

Министерство образования и науки Украины
Российская ассоциация искусственного интеллекта
Национальный технический университет Украины "КПИ"
Факультет прикладной математики
Институт прикладного системного анализа
Издательство "Просвіта"

***Международная научная конференция
имени Т. А. Таран***

**Интеллектуальный анализ информации
ИАИ-2014**

Киев, 14-16 мая 2014 г.

Сборник трудов

Рекомендовано ученым советом
факультета прикладной математики

**«Просвіта»
Киев
2014**

УДК 004.8/.9+001.102](06)

ББК 32.973я43+73я43

I-73

Редакционная коллегия:

д.т.н., проф. Валькман Ю.Р., д.т.н., проф. Гладун В.П., д.т.н., проф. Голенков В.В., к.т.н., доц. Григорьев А.В., д.т.н., проф. Дичка И.А., д.ф.-м.н., проф. Донской В.И., д.т.н., проф., академик НАНУ Згуровский М.З., д.т.н., проф. Кузнецов О.П., д.т.н., проф. Молчанов А. А., д.ф.-м.н., проф. Осипов Г.С., д.т.н., проф. Стефанюк В. Л., д.т.н., проф. Тарасенко В.П., д.т.н., проф. Финн В.К., д.т.н., проф. Хорошевский В.Ф., к.т.н., доц. Чертов О.Р.

Главный редактор к. т. н. Сирота С. В.

Ответственный редактор Темникова Е.Л.

Ответственный за выпуск Копычко С. Н.

Рекомендовано ученым советом факультета прикладной математики НТУУ

“КПИ” протокол №10 от 26 мая 2014 г.

«Интеллектуальный анализ информации», междунар. науч. конф. им. Т.А.Таран (14 ; 2014 ; Киев).

I-73 Международная научная конференция имени Т. А. Таран "Интеллектуальный анализ информации" ИАИ-2014, Киев, 14–16 мая 2014 г. : сб. тр.– К. : Просвіта, 2014. – 260 с. : ил.

ISBN 978–617–7010–03–5

В сборнике опубликованы доклады, представленные на конференции по следующим направлениям: интеллектуальный анализ данных и машинное обучение, правдоподобные рассуждения, интеллектуальные методы классификации, интеллектуальный поиск и анализ информации в локальных и глобальных сетях, прикладные системы интеллектуальных данных, интеллектуальный анализ данных в социальной сфере и гуманитарных исследованиях, инструментальные средства интеллектуального поиска и анализа информации.

УДК 004.8/.9+001.102](06)

ББК 32.973я43+73я43

Використання матеріалів збірки можливе за умови обов'язкового посилання.

Использование материалов сборника возможно при условии обязательной ссылки.

ISBN 978–617–7010–03–5

© ПТФ Просвіта, 2014

Предисловие

Научная конференция "Интеллектуальный анализ информации" носит имя своего основателя и организатора доктора технических наук, профессора Татьяны Архиповны Таран. Конференция начала проводиться с 2001 года как российско-украинский научный семинар. С мая 2005 года она проводится в формате международной конференции. На протяжении четырнадцати лет конференция вызывает интерес специалистов в области искусственного интеллекта и информационных технологий своей неформальной атмосферой живого обсуждения.

Основные научные темы, представленные на ИАИ-2014: знания и рассуждения, онтологический инжиниринг, анализ данных, сетевые и многоагентные модели, мягкие вычисления, обработка естественного языка и речи, социальные проблемы и образование и другие прикладные проблемы.

Доклады в сборнике расположены в алфавитном порядке по фамилии первого соавтора. Исключение сделано для четырех пленарных и одного заключительного докладов.

Выражаем особую благодарность участникам, которые нашли возможность приехать на конференцию, несмотря на сложную геополитическую обстановку.

Желаем успехов участникам конференции и надеемся на плодотворное сотрудничество в будущем.

Программный комитет

УДК 681.3:519.68(076)

Памяти В.П.Гладуна



1936 - 2014

*Святогор Л.А., Величко В.Ю. Институт кибернетики им. В.М. Глушкова
НАН Украины, г. Киев*

24 апреля 2014 года скончался **Виктор Поликарпович Гладун** – доктор технических наук, профессор, старший научный сотрудник Института кибернетики имени В.М. Глушкова Национальной Академии Наук Украины.

Виктор Поликарпович Гладун был ярким представителем талантливой научной молодёжи, которая в начале шестидесятых пришла в кибернетику по зову разума и сердца. С тех пор и до конца дней он оставался верным своему призванию, которое осознавал как научный долг.

Родился Виктор в Киеве в 1936 году в учительской семье. В сорок первом отец ушёл на фронт и погиб под Харьковом. Мама с сыном пережили оккупацию, потом была школа, которую Виктор закончил с золотой медалью.

В 1953 году **Виктор Гладун** поступил в Киевский Государственный Университет имени Тараса Шевченко. В этом году был образован новый факультет – радиофизический, преподавание велось на высоком уровне, соответствующем потребностям страны, в которой бурно развивалось ракетостроение и вскоре был сделан первый шаг в Космос. **Виктор Поликарпович**, благодаря блестящей учёбе, дружелюбию и юмору сразу же завоевал авторитет среди однокурсников, стал признанным комсомольским лидером. Учась в Университете, он завоевал высокие спортивные разряды по гимнастике и альпинизму. Был принят в коммунистическую партию. Ему доверили руководить студенческим отрядом на целине в

горячее время уборки урожая летом 1956 года, и этот добровольный трудовой порыв был отмечен рядом благодарностей.

После получения «красного диплома» об окончании Киевского государственного университета молодой специалист был направлен по распределению на работу в престижный Научно-исследовательский институт радиоэлектроники (г. Киев). Сразу же был замечен и поставлен во главе группы разработчиков системы автоматического управления радиотехническим комплексом. Здесь он научился доводить работы до их приёма Государственной комиссией, приобрёл навыки управления инженерами и техниками в условиях жёсткой производственной дисциплины. Это был его вклад в укрепление обороноспособности страны.

И всё же, несмотря на отличные научные перспективы, **Виктора Гладуна** неумолимо тянуло в Институт кибернетики (в 1962 году – Вычислительный Центр Академии Наук УССР), который основал выдающийся учёный и организатор, основатель кибернетики на Украине Виктор Михайлович Глушков. Поэтому после трёхлетнего инженерного стажа в НИИ **Виктор Поликарпович** поступил на работу в ВЦ АН УССР – в лабораторию Теории цифровых автоматов, которую возглавлял талантливый учёный, впоследствии доктор технических наук, профессор Зиновий Львович Рабинович. На всю оставшуюся жизнь творческая дружба связала их вместе, в жарких дискуссиях между учителем и учеником рождались идеи, алгоритмы, смелые публикации. Под руководством З.Л. Рабиновича в 1967 году **Виктор Поликарпович Гладун** защитил кандидатскую диссертацию «Автоматизированные системы обработки информации».

В атмосфере научного поиска и свободного общения, на бурно протекающих в лаборатории научных семинарах, в которых очень часто принимал участие и директор Вычислительного Центра В.М. Глушков, рождались самые фантастические проекты, которые потом претворялись в жизнь. В.М. Глушков ставил перед молодым коллективом задачу разработки электронно-вычислительных машин для проведения сложных инженерных и теоретических расчётов, которые, в отличие от универсальных ЭВМ, могли бы выпускаться крупносерийно и насытить передовой вычислительной техникой проектно-конструкторские организации, отраслевые институты и конструкторские бюро СССР. В Институте кибернетики были разработаны теоретические и инженерно-конструкторские предпосылки для выполнения этой задачи и практически созданы малые электронно-вычислительные машины. От этой государственной стратегии – создание крупных серий малых ЭВМ – оставался один небольшой

шаг до разработки советских персональных компьютеров, но страна не успела сделать этот шаг.

В период с 1962-го по семидесятые годы **Виктор Поликарпович Гладун** разрабатывал теорию дискретных автоматов и теорию алгоритмов, включая создание машинных языков высокого уровня, ориентированных на «дружественный» интерфейс пользователя. Параллельно он размышлял о принципах обработки информации машинами «неймановского типа» и сопоставлял их с информационными процессами в биологических системах. Это привело его к мысли, что можно создать алгоритмическую модель, которая в какой-то степени воспроизводила бы процессы преобразования информации в мозге человека. Так родилась фантастическая по научной смелости идея – Растущие Пирамидальные Сети (РПС). Спустя годы мы увидели, что это был прорыв в области создания кибернетических устройств и Искусственного Интеллекта.

В основу РПС положена архитектура нейроподобной сети. Алгоритм работы такой сети предельно прост и потому чрезвычайно эффективен. Повторяющиеся регулярно сигналы (стимулы) на входе нейронной сети образуют в ней новые структуры с устойчивыми связями. Если некоторая комбинация входных сигналов повторяется, то в сети «вырастает» отдельный «нейрон» (узел с входящими и выходящими связями), который эту комбинацию запоминает и впоследствии на неё реагирует. Нейронные структуры нижних уровней РПС постепенно – под воздействием обучающей выборки – объединяются в более сложные структуры верхнего уровня. Таким образом, создаётся иерархическая растущая и развивающаяся организация памяти. После конца обучения образуется сетевая структура, сохраняющая в себе «образ» множества входных сигналов. Этот образ может быть представлен в виде множества логических функций, которые формально отображают закономерности порождения множества сигналов.

Как и многие новые и революционные идеи, Растущие Пирамидальные Сети поначалу были встречены научным сообществом с известным скептицизмом. Это, в частности, проявилось в том, что **Виктору Поликарповичу** пришлось защищать свою докторскую диссертацию дважды. Помогли в защите такие качества молодого учёного как упорство в достижении цели и уверенность в созданном им научном продукте. Защита диссертации «Теоретические основы процессов формирования понятий и планирования действий в автоматизированных системах научных исследований», которая состоялась в Киеве, в Институте кибернетики им. В.М. Глушкова в 1983 году, прошла триумфально.

Правильность своей научной теории **Виктор Поликарпович Гладун** доказал не только теоретически, но и на практике. Под его руководством коллектив талантливых исследователей в течение 1970 – 2007 гг. разработал ряд инструментальных программных комплексов: КОДЭКС (для построения экспертных систем), КОНФОР (для обнаружения и анализа закономерностей, на основе индуктивного обобщения данных об известных объектах), АНАЛОГИЯ (для решения задачи классификации на основе вывода по аналогии для объектов, имеющих внутреннюю структуру) и других. КОНФОР способен решать задачи обработки данных, в которых значения целевого свойства зависят от многих параметров без априорных предположений о характере такой зависимости. Наряду с этими комплексами разработан уникальный Язык представления знаний – ЭКСПРЕСС, в котором предусмотрены средства для записи формулировок задач, правила преобразования ситуаций, формы записи планов решений и другие макропроцедуры высокого уровня.

Эффективность новых компьютерных технологий была апробирована в таких областях прикладных исследований, как медицинская диагностика, техническая диагностика радиоэлектронных устройств, определение жизненного цикла механических узлов и некоторых других. С помощью программных комплексов на основе РПС решались разноплановые задачи: прогнозирование мест залегания полезных ископаемых, предсказание солнечной активности, задачи выделения закономерностей, которые присущи экономическим ситуациям и технологическим процессам, и другие. Более трех десятков лет (с 1976 г.) система АНАЛИЗАТОР и ее развитие – КОНФОР используются в исследованиях Института металлургии им. А.А.Байкова (г. Москва) для прогнозирования существования новых материалов с заданными свойствами; кроме этого система показала свою эффективность при решении задачи планирования последовательности действий для синтеза химических соединений (Ин-т органического синтеза, г. Рига). Системы планирования решений – APROS, CODEX применялись в экспериментальном робототехническом комплексе для планирования действий роботов. Прямой народнохозяйственный эффект был получен от внедрения метода прогнозирования миграции рыбы в Баренцевом море (НПО «Севрыба», г. Мурманск, Россия).

В последующие годы деятельности внимание учёного, обладавшего широким кругозором и научным предвидением, сосредоточилось на практических задачах, решаемых человеком на интеллектуальном уровне. Им был сделан шаг от процессов планирования и принятия ре-

шений к проблемам понимания языковых сообщений, которые представлены в виде текстов или дискурсов сложной семантической структуры.

Одним из первых был построен программный комплекс КОНСПЕКТ (2000 – 2007 гг.). Его интеллектуальная работа заключается в том, чтобы для заданного текста произвольной тематики составить автоматически конспект, то есть – дать краткий и грамотный пересказ содержания текста с минимальными искажениями его связности. Использование данной информационной технологии даёт пользователю возможность не только хранить массивы документов в сжатом виде (с сохранением их семантики и основного содержания), но и быстро отбраковывать ненужные, выбирать актуальные файлы.

В последние годы жизни **Виктор Поликарпович Гладун** поставил перед собой амбициозную научную задачу – построить общение человека и компьютера на принципах взаимопонимания. С самого начала он установил, что такое общение невозможно без представления и оформления системы знаний – онтологии. Развитие этих идей привело к созданию концептуального алгоритма, в базу знаний которого заложены, с одной стороны, знания человека об окружающем его Мире и знания о предметных областях деятельности, а с другой стороны – процедуры синтаксического, семантического и прагматического анализа «живого» текста. Предполагается, что понимание текста достаточно сложной когнитивной структуры достигается через извлечение из него базовой категории сознания и коммуникации – «Смысла». Трудность состояла именно в формализации самого понятия «Смысл». Работы в этом направлении продолжаются. В случае успеха компьютеры следующих поколений, андройды и аватары будут обладать элементами антропоморфного понимания и мышления.

Виктор Поликарпович Гладун проявил себя также и как хороший организатор науки. Много лет он, будучи доктором технических наук и профессором, возглавлял лабораторию исследователей в НПО «Горсистемотехника» и в Институте кибернетики имени В.М. Глушкова НАН Украины (г. Киев). Особенный период его жизни связан с началом девяностых годов прошлого века, когда в результате распада СССР региональная наука оказалась на грани деградации. В обстановке отсутствия финансирования он не только нашёл в себе силы сохранить лабораторию, продолжать научную работу, но и сумел объединить учёных Украины, России, Беларуси, Болгарии, Польши, Великобритании, США и других стран в единую организацию, которая действует как «Ассоциация создателей и пользователей систем искусственного интеллекта» (АСПИС).

Международная конференция, регулярные заседания которой проводятся под названием «Знания – Диалог – Решение» (англ. «KDS» – Knowledge-Dialogue-Solution), известна в научном мире на разных континентах; её значимые труды публикуются на страницах журнала “International Journal on Information Technologies and Applications” (“IJ ITA”).

В период 1987 – 1991 гг. **Виктор Поликарпович** был председателем международной рабочей группы WG26, а в 1988 – 1992 гг. – сопредседателем международной рабочей группы WG25, являлся членом Совета ассоциации по проблеме «Искусственный интеллект», принимал участие в организации многих научных конференций как руководитель программного комитета или в составе организационных комитетов, входил в редакционные советы международных научных изданий (в том числе – журнала IJ ITA).

Он вёл преподавательскую работу в ведущих высших учебных заведениях Украины – в Киевском Национальном Университете им. Т.Г. Шевченко, а также в Национальном Университете «НТТУ КПИ». Под его руководством защитились девять соискателей учёной степени кандидата технических наук.

Высокий уровень культуры **Виктора Поликарповича**, знание языка и общественное доверие позволили ему в шестидесятых и семидесятых годах пройти научную стажировку в Англии и в Италии, сотрудничать с учёными США. Именно поэтому он легко находил общий язык с зарубежными исследователями, которые всегда высоко ценили его эрудицию, интеллигентность, уважение к партнёру. Результаты его научных исследований отражены в пяти монографиях и многочисленных статьях.

Увлечённость наукой и высокое трудолюбие **Виктора Поликарповича** гармонично сочетались в его характере с разносторонними увлечениями историей и культурой своей Родины, музыкой, путешествиями. В компаниях друзей он излучал энергию и самобытный юмор, оставаясь в то же время «в тени». Запомнился нам исключительным и чистым стремлением прийти на помощь в трудную минуту, поделиться в походе последним куском хлеба. Был прекрасным семьянином, отцом двух сыновей, воспитывал двух внуков.

В наших сердцах останется глубокое сожаление о потере Учителя и Друга, и в то же время – светлое чувство причастности к его жизни. Сохраним тепло дружеского рукопожатия **Виктора Поликарповича Гладуна**.

Актуальність теми обумовлена високою складністю робіт по створенню і впровадженню автоматизованих інформаційно-керуючих систем в різних галузях народного господарства в поєднанні з повною непрацездатністю теоретичних основ, моделей и методів формалізації та автоматизації розробки оптимальних модульних систем обробки даних. Тому на даний час однією з основних задач дослідження в цій галузі стала розробка теоретичних основ, формалізованих моделей і, разом з тим, методів аналізу і синтезу оптимальних модульних систем обробки даних саме в автоматизованих інформаційно-керуючих системах різного класу і призначення.

Список монографий В.П. Гладуна

1. Глушков В.М., Гладун В.П., Погребинский С.В. Обработка информационных массивов в автоматизированных системах управления. Наукова думка, К, 1970.
2. Гладун В.П. Эвристический поиск в сложных средах. Наукова думка, К, 1977.
3. Гладун В.П. Планирование решений. Наукова думка, К, 1987.
4. Гладун В.П. Процессы формирования новых знаний. София, СД “Педагог”, 1994.
5. Гладун В.П. Партнерство с компьютером К.: “Port-Royal”, 2000. - 128 с.

УДК 007:681.512.2

О гештальтах в мышлении: определения, классификация и свойства

Валькман Ю. Р., зав. отделом, Международный научно-учебный центр информационных технологий и систем НАН и МОН Украины, г. Киев, yur@valkman.kiev.ua

Показано, что семиотика и семиозис во многом имеют когнитивный характер. Рассматриваются определение, свойства и классификация гештальтов. Рассмотрены классификации знаков с ориентацией на построение схем семиозиса. С помощью этих схем производится интерпретация знаков.

Введение

Фундаментальными принципами естествознания является принятие Данная работа является продолжением исследований семиотических структур и когнитивных процессов (см., в частности, [1-3]). *Объектом* этих исследований являются когнитивные процессы. Здесь, *предмет исследования* – гештальты (их свойства и характеристики) в когнитивных процессах. *Общая цель исследования* – моделирование когнитивных процессов в вычислительной среде.

1. Когнитивные структуры, знаки и семиотика: некоторые определения

Основные обозначения, которые будут здесь использоваться, введены и определены в [2]. Здесь рассмотрим их кратко.

Когнитивная система человека как иерархическая совокупность *когнитивных структур* позволяет составить представление о мире, опираясь исключительно на внутренние репрезентации. В любом случае все эти репрезентации имеют знаковую структуру. Известно [2] высказывание Ч. С. Пирса "*мы думаем только в знаках*". Таким образом, мы приходим к семиотике. Широко известен семиотический треугольник, который связывает **S** (знак), **C** (концепт) и **D** (денотат). Для исследования когнитивных процессов целесообразно рассматривать *четырёхугольник*. Схема его представлена на рис. 1.

В этом четырёхугольнике вершина «концепт» заменяется отношением **S''-D''**, в котором **S''**- образ знака в голове интерпретатора (ментальная

модель знака), а D'' - ментальный образ денотата. Будем его называть *десигнатом* (см. в [7]). Обозначением $''$ (см. в [5]) помечаются все объекты,

которые представлены в голове интерпретатора (ментальные модели).

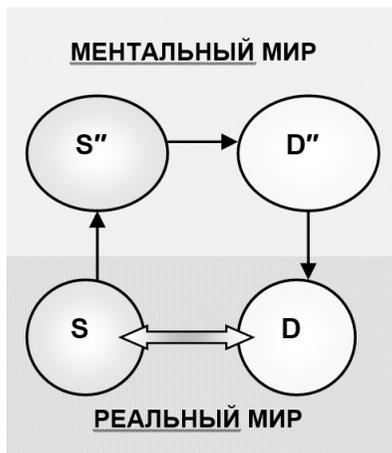


Рис. 1. Отношения знака - денотаты в семиотике.

Очевидно, важнейшим свойством знака является его функция отображения некоторого объекта, т.е. *любой знак всегда знак некоторого объекта*. Заметим, один и тот же объект может быть денотатом в одной структуре и знаком в другой. Известна диаграмма Г. П. Мельникова [5] (см. рис. 1):

$$S \rightarrow S'' \rightarrow D'' \rightarrow D \leftrightarrow S.$$

По сути – это определение знака. Хорошая интерпретация этого определения приведена в [6].

Заметим, что основные семиотические процессы $S'' \rightarrow D''$ или $D'' \rightarrow S''$, восприятия знаков $S \rightarrow S''$ или их порождения $S'' \rightarrow S$, восприятия денотатов $D \rightarrow D''$ проходят в ментальном мире (см. рис. 1) Тем самым показано, что семиотика во многом имеет когнитивный характер

Тем самым показано, что семиотика во многом имеет когнитивный характер

2. Схемы семиозиса: определение и разновидности

Пирс мыслил семиозис как постоянное движение знака. Семиозис у него включает в себя две одинаково важные части: *производство* знаков и их *интерпретацию*. Для нашей когнитивной семиотики (см. в [1-3]) такое понимание семиозиса соответствует анализу и синтезу структур образной информации и образных знаний.

Для Пирса именно понятие *семиозиса* было центральным понятием его семиотической теории. По мысли Пирса, знак не функционирует как знак до тех пор, пока он не осмысливается как таковой. Иначе говоря, знаки должны быть интерпретированы, чтобы быть знаками. Моррис ввел [7] следующее определение: Семиозис – «Процесс, в котором нечто функционирует как знак».

Ранее мы уже использовали понятие *интерпретатор*. Теперь определим его более точно. Интерпретатор – то, что (или кто) воспринимает и/или воспроизводит знаконосители.

Происходящие в интерпретаторе в случае простейшей знаковой коммуникации между интерпретаторами **A** и **B** [5] минимально необходимые звенья процесса коммуникации будут [8] следующими.

1) В интерпретаторе A:

- Возбуждение десигната D_A'' ;
- Возбуждение ассоциаций между десигнатом и образами соответствующих знаков ($D_A'' \rightarrow S_A''$);
- Возбуждение образов знаков S_A'' ;
- Обратимое отражение (по терминологии Мельникова [5]), в результате чего во внешней среде появляется физический носитель знаков S_a .

Схема семиозиса синтеза знака приведена на рис.2.

2) В интерпретаторе B:

- Восприятие органами рецепции интерпретатора **B** физических носителей знаков S_a ;
- Возбуждение образов S_B'' носителей знаков S_a ;
- Возбуждение ассоциаций между образами знаков S_B'' и соответствующим десигнатом D_B'' ;
- Возбуждение десигната D_B'' .

Схема семиозиса интерпретации знака приведена на рис. 3.

3) Схема семиозиса общения ($D_A'' \rightarrow S_A'' \rightarrow S_a \rightarrow S_B'' \rightarrow D_B''$) приведена на рис. 4. Обратим внимание на начало и конец этой цепи, соответственно,

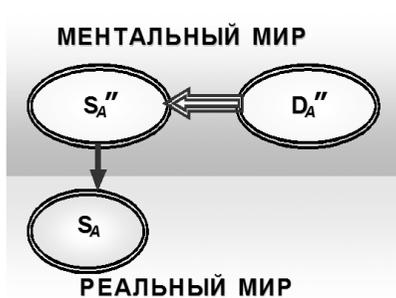


Рис. 2. Схема семиозиса порождения знака (*интерпретатор A*)

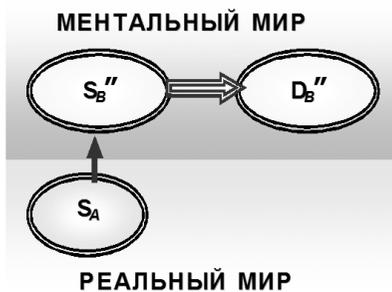


Рис. 3. Схема семиозиса восприятия знака (*интерпретатор B*)

D_a'' – «мысли интерпретатора-передатчика сообщения» и
 D_e'' – «мысли интерпретатора-приемника сообщения».

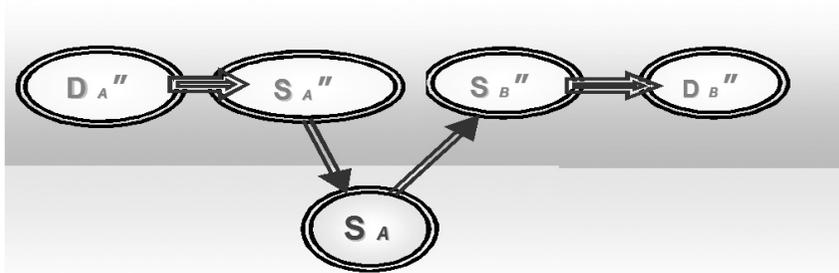


Рис. 4. Схема (семиозиса) общения $D_A'' \rightarrow S_A'' \rightarrow S_A \rightarrow S_B'' \rightarrow D_B''$

Соответственно семиозисам, представленным на рис. 2 и рис.3, можно выделить два частных вида семиозиса: *моторный семиозис* – процесс от возбуждения десигната до генерации знака (знаков) и *сенсорный семиозис* – процесс от восприятия знака (знаков) до возбуждения десигната.

Указанные виды семиозиса сформулированы относительно «внешней» (по отношению к мозгу субъекта) функциональности: производства и восприятия знаков.

Характеризуя указанные виды семиозиса относительно «внутренней» (по отношению к мозгу) функциональности, необходимо признать, что эти виды семиозиса всегда (в той или иной степени) сопровождаются психической деятельностью. Поэтому имеет смысл выделить два частных вида психосемиозиса, соответственно связанных с мотосемиозисом и сенсосемиозисом: *психомотосемиозис* и *психосенсосемиозис* [8].

Итак, объект, или референт, принципиально субъективен и зависит от воспринимающего. Вот знаменитое определение знака, данное Пирсом: «Знак, или репрезентант, есть нечто, что соотносено с иным в каком-либо отношении или по какому-либо признаку. Знак адресован, то есть создает в уме человека эквивалентный или, возможно, более сложный знак. Последний есть интерпретант первого. Знак замещает нечто, а именно свой объект. Он замещает его не во всех отношениях, но в соотношении с той идеей, которую я называю базисом, или основанием знака».

Таким образом, для Пирса интерпретант – это S'' . Проблема «адресованности» знака разрабатывается в теории речевых актов; «сложный» знак указывает на многосоставные акты интерпретации, например, в научном анализе, что же касается базиса, то это основание любой интерпретации,

которое стоит ближе всего к общепринятому понятию. Схемы семиозиса (в [3] введено и обосновано понятие *цепи семиозиса*) могут быть намного сложнее. Пример представлен на рис. 5.

Так как каждый знак производит интерпретанту, которая, в свою очередь, является репрезентантом следующего знака, процесс семиозиса оказывается «*последовательным рядом интерпретант*» ad infinitum [4]. В бесконечном процессе семиозиса ($S_1'' \rightarrow D_1'' \rightarrow S_2'' \rightarrow D_2'' \rightarrow \dots \rightarrow S_i'' \rightarrow D_i'' \rightarrow \dots$) нет *первого* и нет *последнего* знака.

Однако мысль о бесконечном семиозисе не означает отсутствия семиотических ориентиров, но отсылает к актуальному взгляду, согласно которому «*мышление всегда осуществляется в форме диалога - диалога между различными фазами „я“, так что мышление, поскольку оно диалогично, в сущности, состоит из знаков*» [4]. Так как «*каждая мысль должна быть обращена к другой мысли*», непрерывный процесс семиозиса (или мышления) может быть только прерван, но не может быть окончен.

Заметим, что когнитивная семиотика в такой интерпретации исследует только процессы возбуждения образов, их отражения, трансформации и интерпретации. Но существует еще много когнитивных операций (см., например, в [6, 9]).

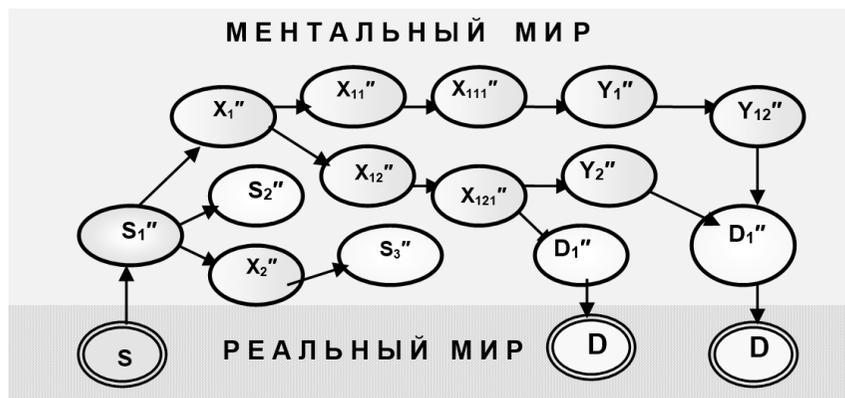


Рис. 5. Пример схемы сложной цепи семиозиса

Однако, с нашей точки зрения, схемы различных видов семиозиса являются реальной основой разработки моделей синтеза и анализа образной информации для построения компьютерных технологий.

3. Об определении гештальта

Гештальт основное понятие гештальтпсихологии [10]. Он обозначает *цельные, несводимые к сумме своих частей, образования сознания*. Понятие о гештальте зародилось при изучении процессов *восприятия* и *структурирования ощущений* человека. Примеры гештальта: инсайт; восприятие мелодии, не сводимое к сумме ощущений звуков этой мелодии и т. д. Возникнув в психологии, сегодня термин «гештальт» используется также для обозначения некоторых целостных и неделимых физических, физиологических и социальных феноменов.

В буквальном переводе с немецкого Gestalt означает «*форма, вид, фигура*». Другие более или менее подходящие по смыслу русские аналоги — «*целостность*», «*структура*», «*система*», «*образ*», «*архетип*» и «*модель*».

Еще И. Кант говорил о том, что мы не можем воспринимать физический мир непосредственно. Человек всегда взаимодействует с информацией, полученной от органов чувств — дорабатывая ее в своем сознании. Таким образом, *любое целое для нас больше суммы его частей, потому что мы вкладываем в него и свое восприятие*. Образно говоря, когда мы смотрим на картину Ренуара или другого импрессиониста, мы не замечаем и не оцениваем каждый мазок, а видим единое целое и именно это общее сочетание цветов и форм нас впечатляет. Такое целостное восприятие и стали называть «*гештальтом*». Интересные эксперименты проводились с музыкой — Эрнст Мах в своей работе «Анализ ощущений» доказал, что изменение тональности и темпа мелодии не мешает идентифицировать мотив. Похожий эксперимент — но только с текстом — сейчас широко тиражируется в соцсетях: благодаря способности мыслить гештальтами вы можете понять предложение, даже если в каждом слове поменять порядок букв и оставить на месте только начальные и конечные. *Пример, вы легко можете почтнуть эту фразу.*

В наших обозначениях гештальт определяется, как D_i'' ($i \in I$, где I — множество всех гештальтов). Поскольку переход (отображение, операция) $S_j'' \rightarrow D_i''$ является основной в процессах мышления, то для нас гештальты представляют особый интерес. И мы намерены исследовать их детально. В данной работе лишь определены некоторые концепции будущих исследований.

Здесь мы хотим показать, что, *во-первых*, гештальты «участвуют» не только в процессах восприятия, но и во многих других процессах нашего мышления и жизнедеятельности. Поэтому, можно классифицировать их по различным критериям.

И, во-вторых, они обладают многими интересными структурными (системными) свойствами, что позволит нам разрабатывать их модели для представления операций мышления в вычислительной среде.

При этом целостность остается главным свойством гештальта (который всегда является отражение объекта, процесса, свойства, отношения – *первый критерий классификации*). Структура гештальта – это взгляд на его «внутренность», или результат анализа сущности, им отражаемой. Так как гештальт что-то отражает, то это – модель, а, следовательно, и система, и образ (см. выше).

4. О классификации гештальтов

Теперь, понятно, что использование понятия *гештальта* [10] чрезвычайно важно в семиозисе и вообще в когнитивных процессах. Рассмотрим некоторые признаки классификации гештальтов.

1. Тот внутренний врожденный (или встроенный в автомат) *интенциональный образ*, который позволяет опознавать определенные внешние объекты (знаки *S* и денотаты *D*) без предварительного научения, только благодаря правильной работе рецепторов, назовем априорным гештальтом [5].

Наличие априорного гештальта позволяет не только обеспечивать отнесение внешних отражаемых объектов к определенному универсальному множеству, но и ускорять процесс выработки *новых интенциональных образов* для различения подмножеств этого множества и даже индивидуальных его представителей. Такие образы можно бы назвать апостериорными гештальтами [5].

Без априорных гештальтов невозможно замыкание цепочек семиозиса. Естественно предположить, что, всегда, *любой акт опознания (в семиозисе) должен завершаться возбуждением априорного гештальта* под влиянием возникновения образов опознаваемого объекта.

2. Очевидна соотнесенность априорного гештальта с конкретными рецепторами например, обоняние, и зона обонятельной рецепции и т.д. Естественно, что априорный гештальт, представляя отражательный опыт вида в индивиде, не имеет пробраза в виде конкретного объекта, т. е. уникального объекта в уникальных обстоятельствах (такие объекты мы называем *окказиональными* – см. в [2]).

Являясь обобщенным и не возникающим, а лишь воспроизводимым для определенного функционального использования, априорный гештальт должен быть отнесен к числу *узуальных образов*. Однако среди *окказиональных образов* (т. е. образов *окказиональных объектов*), а также обобщенных знаков, ставших *узуальными образами*, выработанными в

практике самого интерпретатора и поэтому уже не априорными, а *апостериорными*, априорные гештальты остаются полноправными членами во всех отношениях.

3. По аналогии с образами мы можем рассматривать: *гештальт восприятия, гештальт представления и гештальт воображения*.

4. В чувственном образе может быть воплощено любое содержание. Поэтому мы можем говорить о: *зрительных гештальтах, слуховых, тактильных, мышечных, вестибулярных, вкусовых, обонятельных* и т. д.

Кроме этого, могут быть гештальты различных комбинаций чувственных образов.

5. В [11] Дж. Лакофф предполагает что: *мысли, восприятия, эмоции, процессы познания, моторная деятельность и язык* организованы с помощью одних и тех же структур, которые он называет *гештальтами*.

6. Пирс выделяет четыре класса отношений: *ассоциации по сходству (иконы), по смежности (индексы), по сходству и смежности одновременно (иконические индексы), договорные отношения (символы)*.

Эти отношения являются основными для моделирования отношений $S'' \leftrightarrow D''$ (см. рис. 1-4) в вычислительной среде. А значит, фактически, речь идет о классификации гештальтов. Классификатор Пирса классифицирует не знаки, а отношения их к гештальтам.

5. Некоторые свойства гештальтов

В [11] Дж. Лакофф, отказываясь от определения гештальта, рассматривает некоторые его свойства. С нашей точки зрения, по-сути, это свойства системы, т. е. гештальт рассматривается как система, с соответствующей структурой

1. Гештальты являются одновременно целостными и анализируемыми. Они состоят из частей, но не сводимы к совокупности этих частей. Гештальты обладают определенными дополнительными свойствами, и их части могут располагать дополнительными свойствами, если рассматривать их в рамках гештальта в целом.

В системном анализе эти свойства называют эмерджентными или свойствами целостности.

Как известно, гештальт не делится на части. Но, можно ли составлять из отдельных гештальтов новый гештальт?

Содержание восприятия вообще *гештальтно*; оно таково изначально, а не в результате акта соединения отдельных ощущений. Однако бесспорно и то, что в восприятие любого предмета входит многое такое, что само может быть предметом самостоятельного восприятия.

Например, хотя бы вот это слово, которое мы сейчас читаем. *В качестве слова оно является отдельным гештальтом, предметом отдельного восприятия.* Однако очевидно и то, что в то же время мы можем воспринять и отдельные буквы; *в этом случае каждая из них также будет представлять собой предмет отдельного восприятия и иметь собственный гештальт.*

Рассмотрим «дерево». Как дерево оно является отдельным гештальтом; но предмет восприятия составляет и вот то яблоко, что висит на нижней ветке слева, — ведь и оно, как таковое, дано в содержании восприятия в виде гештальта. *Одним словом, предметы как целостные гештальты состоят из частей, в свою очередь также представляющих собой отдельные гештальты* [10].

2. Гештальты разложимы более чем одним способом, в зависимости от принятой точки зрения. Это означает, что возможны различные способы «правильного» разложения гештальта на части.

В системном анализе этот факт давно известен, как зависимость построения состава системы от решаемых задач, проблем.

3. Части гештальта связаны внутренними отношениями, которые бывают различных классов (типов – у Лакоффа). Способ отношений частей в рамках гештальта входят в содержание самого гештальта.

В системном анализе система этих отношений называется структурой. Именно они обеспечивают целостность системы.

Эти отношения в принципе принадлежат к разным типам, и при описании гештальта типы отношений должны быть указаны. Свойства гештальта и типы отношений между его частями могут быть различными в зависимости от того, какая точка зрения принята.

4. Гештальт может быть связан внешними отношениями с другими гештальтами. Он может составлять часть другого гештальта или проецироваться на другой иным способом (см. 1 пункт данного раздела).

Хороший пример понимания приведен в [13]. «Предложение *«Книга лежит на столе»* непосредственно понимается, потому что *книга* и *стол* – понятия базового уровня (гештальты – примечание мое), а *на* образовано схемами (схемы, мы полагаем, - тоже являются гештальтами) *над*, *контакт* и *опора*. Непосредственное понимание ситуации «Книга лежит на столе», означает гештальтное восприятие книги, стола и отношений между ними».

При проецировании части одного гештальта отображаются на части другого гештальта. Результатом проецирования будет «наследование» частями второго гештальта свойств и отношений первого. Существуют различные типы проецирования, например умозаключения (они транзи-

тивны) или проецирование, отражающее произвольные, символические или культурные ассоциации (они обычно не транзитивны).

5. Внешние соотношения гештальта с другими гештальтами являются свойством гештальта в целом.

Эти внешние отношения также могут быть различными в зависимости от принятой точки зрения.

6. Проецирование гештальта на другой гештальт может быть частичным. Гештальты могут также иметь частичное сходство друг с другом, *степень сходства зависит от степени близости связи между ними.*

7. Гештальт, включенный в качестве части в другой гештальт, может приобрести новые свойства.

8. Одним из свойств гештальта может быть то, что он находится в отношении оппозиции к другому гештальту.

9. Некоторые свойства гештальта могут выделяться как базисные свойства. Гештальты, находящиеся в отношении оппозиции, обычно имеют одинаковые базисные свойства.

10. Гештальты — это структуры, используемые в процессах — языковых, мыслительных, перцептуальных, моторных или других.

В [14] вводится и обосновывается когнитивная структура, названная *предвосхищающей схемой*. Эти схемы подготавливают индивида к принятию информации строго определенного, а не любого вида и, таким образом, управляющие зрительной активностью. Поскольку мы способны видеть только то, что умеем находить глазами, именно эти схемы (вместе с доступной в данный момент информацией) определяют, что будет воспринято. Мы полагаем, что эти схемы правомерно назвать гештальтами.

11. Процессы сами по себе можно рассматривать как гештальты.

12. При анализе гештальтов не выдвигается требования необходимости подразделения анализируемой сущности на элементарные части. Анализ, осуществляемый человеком, ограничен возможностями, намерениями и точкой зрения человека. При разных возможностях, намерениях и точках зрения результаты анализа будут различаться не только степенью детальности, но и самими выделенными частями.

Так, например, в семантике нет необходимости доводить анализ до элементарных, атомарных предикатов.

13. Гештальты могут пересекаться.

14. В гештальтах должны быть разграничены прототипические и непрототипические свойства (см. в [12]).

15. Гештальты часто относятся к смешанному типу. Это означает, что свойства гештальтов неоднородны. Так, сенсорно-моторные гештальты включают как перцептуальные, так и моторные свойства. *Лингвистические*

гештальты будут включать несколько типов свойств — грамматические, семантические, фонологические, функциональные.

В заключение Лакофф подчеркивает, что сказанное выше не является определением гештальта, а лишь дает представление об этом понятии (точнее, отражает точку зрения Лакоффа на гештальт). И, далее он подчеркивает, что возможны поправки, если это по каким-либо соображениям потребуется. Для нас это тоже стартовый набор характеристик гештальта.

И, еще, он как лингвист рассматривает почти исключительно явления языка. Рассуждения же о том, что ограничения на язык, предлагаемые теорией гештальтов, легко могут быть перенесены в другие, внеязыковые, сферы, не более чем его рабочая гипотеза.

Подчеркнем два существенных, с нашей точки зрения вывода.

Во-первых, термин «гештальт», как он здесь используется, несколько напоминает одноименное понятие, в гештальтпсихологии, однако очевидным образом во многих отношениях отличается от последнего.

Во-вторых, предлагаемое понятие гештальта весьма недоопределенно. И эта недоопределенность соответствует нашей концепции моделирования когнитивных процессов посредством компьютерных технологий (см. [1-3]).

Заключение

В работе показано, что семиотика и семиозис во многом имеют когнитивный характер. Обосновано, что знаки должны быть интерпретированы, чтобы быть знаками.

Исследованы принципы построения схем семиозиса с целью построения формального аппарата для реализации компьютерных технологий образного мышления. Начат процесс классификации этих схем. Исследовано определение гештальта, его свойства.

Рассмотрены понятия априорных и апостериорных гештальтов. Показано, что они являются необходимым атрибутом семиозиса. Гештальты «участвуют» не только в процессах восприятия, но и во многих других процессах нашего мышления и жизнедеятельности. Поэтому, можно классифицировать их по различным критериям. Они обладают многими интересными структурными (системными) свойствами, что позволит нам разрабатывать их модели для представления операций мышления в вычислительной среде.

Литература

1. *Валькман Ю. П.* О когнитивной семиотике // Сборник трудов X Междунар. конференции «Интеллектуальный анализ информации (ИАИ-2012)», НТУУ КПИ Киев: Просвіта, 2012, с. 24-38.
2. *Валькман Ю. П.* Когнитивность семиотики // Сборник трудов X Междунар. конференции «Интеллектуальный анализ информации (ИАИ-2013)», НТУУ КПИ Киев: Просвіта, 2013 с. 48-63.
3. *Валькман Ю. П.* Когнитивная семиотика // Сб. научн. тр. 1 Междунар. симпозиума «Гибридные и синергетические интеллектуальные системы: теория и практика», Москва, Калининград, «Наука. Физматлит», май, 2012, с. 98-116.
4. *Пирс Ч.С.* Логика как семиотика: теория знаков. // Метафизические исследования. Вып. 11. Язык. СПб, 1999. С. 199-217.
5. *Мельников Г. П.* Системология и языковые аспекты кибернетики. Москва: Советское радио, 1978.
6. *Рыков В. В.* Извлечение знаний – реализация когнитивной семиотики. <http://rykov-cs.narod.ru/dlg9.html>.
7. *Моррис Ч. У.* Основания теории знаков. // Семиотика, Москва: «Радуга», 1983.
8. Основные виды семиозиса. <http://www.cit.bsu.by/site1/ovs.htm>
9. *Шадриков В.Д.* Интеллектуальные операции. – Москва: Логос, 2006.
10. *Вертгеймер М.* Продуктивное мышление. Москва, 1987.
11. *Лакофф Дж.* Лингвистические гештальты // Новое в зарубежной лингвистике. Выпуск X Лингвистическая семантика. Москва: «Прогресс» 1981, стр. 337-353
12. *Лакофф Дж.* Женщины, огонь и опасные вещи: Что категории языка говорят нам о мышлении, Москва, 2004.
13. *Кузнецов О.П.* Когнитивная семантика и искусственный интеллект.// Искусственный интеллект и принятие решений, 2012, №4, с.32-42.
14. *Найссер У.* Познание и реальность. Смысл и принципы когнитивной психологии — Москва: Прогресс, 1981.

УДК 004.8

Смысловые модели представления и обработки знаний

Голенков В.В., д.т.н., проф., Гулякина Н.А., к.ф.-м.н., доц., Белорусский государственный университет информатики и радиоэлектроники, г. Минск, golen@bsuir.by

В работе в качестве формальной основы проектируемых интеллектуальных систем предлагается использовать графодинамические модели специального вида, ориентированные на параллельную и асинхронную обработку информации

Важнейшей задачей искусственного интеллекта в настоящее время является построение общей комплексной теории интеллектуальных систем, в рамках которой бы сочетались самые разные направления искусственного интеллекта – и теория представления знаний, и теория решения задач (в том числе различные исчисления, эвристики, стратегии), и теория программ (процедурных, декларативных, параллельных, последовательных), и архитектуры интеллектуальных систем, в том числе детализированные до уровня аппаратной поддержки, и теория интеллектуальных пользовательских интерфейсов.

Сейчас эпицентром развития искусственного интеллекта является не столько разработка отдельных его направлений, сколько их глубокая семантическая интеграция, целью которой должна быть не только общая теория интеллектуальных систем, но и общая, доступная технология их комплексного проектирования.

В качестве формальной основы проектируемых интеллектуальных систем, в качестве основы абстрактных логико-семантических моделей интеллектуальных систем предлагается использовать графодинамические модели специального вида – смысловые модели представления и обработки знаний, в основе которых лежат семантические сети [1-5].

Фактически, речь идет о создании формальных средств представления смысла различных видов знаний и формальных средств описания обработки знаний на смысловом уровне [6,7].

Семантическая сеть – это графовая структура G , задаваемая пятеркой $\langle V, C, K, M, I \rangle$ и удовлетворяющая следующим требованиям, которые дополняют свойства множеств V, C, K, M, I , указанные в определении графовой структуры:

- каждая вершина $v_j \in V$ является знаком одного из объектов, описываемых семантической сетью;
- каждая ключевая вершина $k_j \in K$ является знаком соответствующего класса элементов графовой структуры G ;
- каждая метка $m_j \in M$ также является знаком соответствующего класса элементов графовой структуры G ;
- каждая пара инцидентности, принадлежащая любому отношению инцидентности $ij \in I$, является парой принадлежности, связывающей знак некоторого множества элементов семантической сети с одним из этих элементов;
- в семантической сети вершины $v_j, vk \in V$ могут быть инцидентны друг другу, но только в том случае, если по крайней мере одна из них (например, vk) является ключевой ($vk \in K$), а вторая является вершиной, принадлежащей множеству, обозначаемому ключевой вершиной vk (т.е. $v_j \in vk$);
- каждый элемент множества I ($ij \in I$) является знаком некоторого подмножества отношения принадлежности, задающего определенную роль, выполняемую соответствующими элементами семантической сети в рамках соответствующих множеств таких элементов. Указанные подмножества отношения принадлежности будем называть ролевыми отношениями;
- каждый связующий элемент $c_j \in C$ является знаком некоторого фрагмента графовой структуры G , а точнее знаком некоторого подмножества множества всех элементов графовой структуры G ;
- среди элементов графовой структуры G нет пар синонимичных друг другу знаков, т.е. знаков, обозначающих один и тот же объект (один и тот же внешний описываемый объект, либо одно и то же множество элементов графовой структуры);
- среди элементов графовой структуры G нет омонимичных знаков, которые в разных контекстах, в разных обстоятельствах могут обозначать разные сущности.

Следовательно, все (!) элементы (атомарные фрагменты) семантической сети являются знаками различных сущностей (объектов). Такими сущностями могут быть всевозможные внешние описываемые объекты, а также различные множества, состоящие их элементов (атомарных фрагментов) этой же семантической сети.

Таким образом, семантическая сеть – это абстрактная знаковая конструкция "рафинированного вида", в которой нет ничего кроме знаков и инцидентности этих знаков. В частности, в семантической сети отсутствуют элементарные незначащие фрагменты (символы), имена описываемых объектов, слова, из которых эти имена состоят, всевозможные раз-

делители и ограничители, обеспечивающие структуризацию текста. В отличие от текстов традиционного вида, семантическая сеть имеет в общем случае нелинейный характер, поскольку каждый элемент семантической сети может быть инцидентен более чем двум другим элементам.

Семантическую сеть можно трактовать как абстрактный текст, который является семантическим инвариантом соответствующего максимального множества семантически эквивалентных текстов, принадлежащих всевозможным языкам.

На основе понятия семантической сети вводится понятие языка семантических сетей в заданном алфавите и с заданным набором ключевых узлов.

Семантические сети как модели представления знаний известны давно. Но, в отличие от фреймовых, продукционных и логических моделей, для семантических сетей не были разработаны достаточно удобные и практически используемые языки представления знаний, достаточно удобные языки программирования, специально ориентированные на обработку семантических сетей. И, как следствие этого, не были созданы широко используемые комплексные технологии проектирования интеллектуальных систем, в основе которых лежат семантические сети. Причин тому много. Одна из них – это не совсем привычный, нетрадиционный характер таких моделей и возникший на этой основе миф о сложности их реализации. Если такие графодинамические семантические модели реализовывать "в лоб" и без ориентации на последующую аппаратную поддержку, то, конечно, это будет неэффективно. На фоне бурного развития микроэлектронных технологий подобного рода мифы выглядят все менее и менее убедительными. Более того, развитие Internet-технологий привело к необходимости формализации смысла информации, обрабатываемой в сети Internet, что вызывало бурное развитие целого направления – Semantic Web.

Чем же хороши семантические сети и в чем достоинство семантических моделей обработки информации:

- Представление знаний в виде семантических сетей позволяет существенно упростить процедуру интеграции знаний и свести эту процедуру к выявлению и склеиванию синонимичных элементов интегрируемых семантических сетей;
- Специфика обработки баз знаний заключается в том, что порождаемые (генерируемые) новые фрагменты знаний необходимо не просто построить, но и погрузить, интегрировать в текущее состояние базы знаний, т.к. в этих порождаемых фрагментах знаний могут появиться знаки, синонимичные тем, которые уже присутствуют в текущем состоянии базы знаний. Таким образом, процедура интеграции порождаемых

фрагментов обрабатываемой базы знаний является процедурой, постоянно используемой в ходе обработки знаний. Следовательно, представление знаний в виде семантических сетей, благодаря упрощению процедуры интеграции знаний, позволяет упростить не только ввод новых знаний из вне, но и интеграцию в состав текущего состояния базы знаний новых знаний, порождаемых в ходе решения задач;

- База знаний интеллектуальной системы, представленная в виде корректно построенной семантической сети, полностью исключает дублирование информации в рамках такой базы знаний – каждая информация, представленная соответствующим фрагментом семантической сети, должна находиться в рамках этой семантической сети там и только там, где она должна находиться, и нигде больше;
- Представление знаний в виде семантических сетей позволяет существенно упростить процедуру ассоциативного доступа к различным видам фрагментов хранимой базы знаний, а также существенно расширить типологию запросов (вопросов) к базе знаний;
- Семантические модели обработки знаний не просто хорошо приспособлены к поддержке параллельной асинхронной обработки информации, но и обеспечивают обмен информацией через общую графодинамическую память между различными параллельно (одновременно) протекающими (выполняемыми) процессами, что может существенно ускорить каждый из этих процессов. Примером такого взаимодействия параллельно протекающих процессов является одновременная реализация разных стратегий и тактик, направленных на поиск пути решения заданной задачи;
- С помощью семантических моделей представления и обработки знаний можно проинтерпретировать все известные виды моделей представления обработки знаний (фреймовые, продукционные, логические), а также все известные модели решения задач различного вида и все известные модели рассуждений. Это дает возможность рассматривать перечисленные модели не как альтернативные, а как дополняющие друг друга модели, которые могут сосуществовать в разных сочетаниях в разных интеллектуальных системах.

Семантическая модель обработки знаний представляет собой абстрактную многоагентную систему, состоящую из абстрактной семантической памяти в которой хранятся семантические сети, и из множества агентов, ориентированных на обработку семантических сетей, хранимых в указанной семантической памяти.

Семейство абстрактных агентов, работающих над семантической памятью вместе с этой семантической памятью можно трактовать как семантическую модель решения задач, используемую в соответствующей

интеллектуальной системе, или как операционную семантику этой интеллектуальной системы. Подчеркнем, что семантическую модель обработки информации можно построить для любой компьютерной системы (как для интеллектуальной системы, так и для компьютерной системы традиционного вида), обеспечивая, тем самым, семантическую совместимость (на абстрактном уровне) не только интеллектуальных систем, но и компьютерных систем любого уровня интеллектуальности.

Всю семантическую сеть (максимальную семантическую сеть), хранимую в семантической памяти абстрактной логико-семантической модели интеллектуальной системы, будем называть абстрактной семантической моделью базы знаний этой интеллектуальной системы.

База знаний должна содержать в себе всю информацию, необходимую агентам, работающим над семантической памятью, для организации коллективной деятельности по решению задач, с которыми должна справляться интеллектуальная система (сюда, в том числе входит и описание блокировок, задаваемых разными процессами в семантической памяти).

Фундаментальными принципами естествознания является принятие Данная работа является продолжением исследований семиотических структур и когнитивных процессов (см., в частности, [1-3]). *Объектом* этих исследований являются когнитивные процессы. Здесь, *предмет исследования* – гештальты (их свойства и характеристики) в когнитивных процессах. *Общая цель исследования* – моделирование когнитивных процессов в вычислительной среде.

Литература

1. *Кузнецов И.П.* Семантические представления. – М : Наука, 1986.
2. *Лозовский В.С.* Семантические сети // Представление знаний в человеко-машинных и робототехнических системах. – М.: ВИНТИ, 1984. – С. 84-121.
3. *Плесневич Г.С.* Представление знаний в ассоциативных сетях // Изв. АН СССР. Техн. кибернет. - 1982. – N 5. - с.6-22.
4. *Скороходько Э.Ф.* Семантические сети и автоматическая обработка текста. – Киев: Наук. думка, 1983.
5. *Шенк Р.* Обработка концептуальной информации. – Москва: Энергия, 1980.
6. *Представление и обработка знаний в графодинамических ассоциативных машинах / В.В.Голенков, О.Е.Елисеева, В.П.Ивашенко и др.; Под ред. В.В.Голенкова.* – Мн.: БГУИР. – 2001. – 412 с.
7. *Электронный ресурс.* Открытая семантическая технология проектирования интеллектуальных систем [Электронный ресурс]. – 2012. - Режим доступа: <http://ostis.net>. – Дата доступа: 015.03.2014.

УДК 517.977.52

Еще одна концепция когнитивной семантики

Кузнецов О.П., д.т.н., проф., Институт проблем управления им. В.А. Трапезникова РАН, г. Москва olpkuz@yandex.ru

В работе обсуждаются основные положения концептуальной семантики, предложенной известным лингвистом С.Пинкером. Как и другие концепции когнитивной семантики, эта концепция подчеркивает фундаментальную роль физической природы человека в формировании его когнитивных процессов. Важный тезис Пинкера - утверждение о наличии языка мышления, первичного по отношению к вербальному языку.

Введение

Одно из фундаментальных понятий современной информатики – понятие формализации – возникло задолго до появления не только компьютерных наук, но и самих компьютеров. Оно появилось в результате начавшихся во второй половине XIX века исследований оснований математики, которые ставили своей целью повышение строгости математических доказательств.

Решение формализованной задачи должно сводиться к чисто формальным операциям над ее описанием. Идея формального подхода заключалась в том, чтобы исключить из процесса решения такие субъективные характеристики решателя задачи, как знания, опыт и сообразительность, меняющиеся от человека к человеку, и тем самым максимально унифицировать процесс решения. Именно такой подход привел в 30-х гг. XX-го века к представлениям о том, что процесс решения задачи должен быть механической процедурой, которую можно передать машине. Тем самым к моменту появления первых вычислительных машин концептуальная база для их создания и использования была полностью подготовлена.

Формальный подход не означает отказа от рассмотрения содержательной стороны задачи, т.е. ее смысла. Однако он четко разделяет синтаксис (т.е. правила построения выражений языка и операций над ними) и семантику, т.е. смысл выражений языка. Более того, он утверждает, что решение задачи может быть осуществлено чисто синтаксическими методами; семантика присутствует лишь в начале (при постановке задачи) и в конце – при осмыслении формального решения. Иначе говоря, синтаксис существует независимо от семантики в виде формальной системы или (что то же самое) формального языка.

Первая успешная реализация этого подхода была осуществлена в логических исчислениях (формальных теориях). В работах А.Тарского 30-х гг. (см. [1]) отношения между синтаксисом и семантикой в формальных теориях были четко сформулированы благодаря точному понятию интерпретации как однозначного отображения элементов формального языка на элементы предметной области, соответствующей содержанию, т.е. семантике теории. Успеху здесь способствовало то, что предметными областями, для которых строились формальные теории, были обычные математические теории с хорошо структурированным строением; формализация этих теорий заключалась в их аксиоматизации, т.е. в их превращении в логические исчисления. Классическим примером такой аксиоматизированной теории является формальная арифметика.

Второй успешной реализацией формального подхода, сыгравшей решающую роль в компьютеризации различных сфер человеческой деятельности, явились возникшие в 50-х гг. средства автоматизации программирования на основе универсальных языков программирования. Процедурами интерпретации (в логическом смысле) здесь являются процедуры трансляции, а семантикой – программы в машинных кодах.

Одновременно появилась развитая лингвистом Н.Хомским теория формальных языков и грамматик, которая заодно оказалась теоретическим обобщением средств описания языков программирования (нормальной формы Бэкуса-Наура). Эта теория, сыграв значительную роль в создании математических основ структурной (а в последствии и компьютерной) лингвистики, вселила большие надежды на возможность построения формальной модели естественных языков, которая, в частности, смогла бы решить проблему машинного перевода чисто синтаксическими средствами.

Действительность, однако, оказалась намного сложнее. Довольно скоро пришлось убедиться в недостаточности формального подхода, о чем хорошо сказано в обзоре [2]:

«в процессе построения формальной алгоритмической модели языка на каждом ее этапе усложняется алгоритм и правила. И, несмотря на это, всякий раз оказывается, что какие-то фрагменты естественного языка все равно не порождаются, а то, что порождается, наоборот, не существует в языке, так что требуются новые и новые усовершенствования. Поэтому именно рамки узкой алгоритмической модели заставляют однажды задать вопрос: а что, если человек думает и говорит совершенно иначе — т.е., не алгоритмически.»

Этот вывод означает следующее. Процедуры формального (алгоритмического) подхода механистичны и не зависят от того, кто их исполняет – человек или компьютер. Если же человек мыслит не алгоритмически, то

свойства естественного языка должны быть связаны с особенностями его когнитивных процессов. Так возникла когнитивная лингвистика, в которой основную роль стала играть семантика.

Когнитивная семантика Дж.Лакоффа

В когнитивной системе человека синтаксис уже не играет решающей роли. В отличие от компьютера и формальных систем, люди не оперируют незначимыми символами.

«Когда человек доказывает утверждение, он опирается на его смысл, не строя формального доказательства. Разумный человек, не изучавший логику, способен точно выразиться и быстро определять противоречия, но часто не может построить формальное доказательство» [3].

Наиболее развернутую концепцию когнитивной семантики предложил Дж.Лакофф [4]. Ее основные принципы изложены в [5]. Здесь мы сформулируем их следующим образом.

1. Значения возникают раньше, чем формируются концептуальные структуры: они возникают из нашего доконцептуального телесного опыта. Наш опыт доконцептуально структурирован на базовом уровне. Доконцептуальные структуры – это гештальты и образно-схематические структуры, порожденные нашим опытом, такие как вместилище, верх-низ, часть-целое, центр-периферия и т.д.

2. Базовый уровень понятий (уровень прототипа в концепции Э.Росх [6]) находится в середине иерархии абстрактного-конкретного и характеризуется наличием целостно воспринимаемого ментального образа – гештальта.

3. Образно-схематические структуры являются основой человеческих рассуждений, во многом заменяя правила логического вывода. В частности, схема «вместилище» - это образный аналог *Modus Ponens*: если вместилище *A* находится во вместилище *B* и *X* находится в *A*, то он находится в *B*.

3. В терминах гештальтов и образно-схематических структур формулируется проблема понимания, крайне существенная при описании когнитивных процессов, но принципиально невыразимая в рамках формального подхода. Можно сказать, что формальные рассуждения – это «рассуждения без понимания».

Очевидно, что проблемы, затронутые когнитивной семантикой, выходят за пределы собственно лингвистики, и представляют несомненный интерес для искусственного интеллекта.

Здесь мы рассмотрим еще одну концепцию когнитивной семантики, изложенную в книге [7].

Концептуальная семантика С.Пинкера – язык мышления предшествует вербальному языку

Один из главных тезисов концепции Пинкера, которую он называет концептуальной семантикой, заключается в том, что «значения слов репрезентированы в мозгу в виде совокупностей базовых понятий, представленных на языке мышления». Иначе говоря, язык мышления – это язык семантики, предшествующий вербальному языку. Понятия хранятся в форме, гораздо более абстрактной, чем конкретные предложения. Это подтверждается, в частности, тем, что люди гораздо хуже запоминают конкретные предложения, чем смысл, который они из них извлекли (и который может не совпадать со смыслом, который вкладывал в них говорящий).

Следует заметить, что этот тезис прямо противоречит известной гипотезе Сепира-Уорфа, которая утверждает, что структура языка определяет мышление и способ познания реальности.

Хорошей иллюстрацией тезиса Пинкера о первичности языка мышления являются два свидетельства Эйнштейна. Одно из них было высказано в беседе с Вертгеймером.

«Я вообще очень редко думаю словами. Приходит мысль, а потом я могу попытаться выразить ее словами.» [8]

Другое содержится в замечательной книге Адамара [9]:

«Слова, написанные или произнесенные, не играют, видимо, ни малейшей роли в механизме моего мышления. Психическими элементами мышления являются более или менее ясные знаки или образы, которые могут быть «по желанию» воспроизведены или скомбинированы. ... Элементы, о которых я только что говорил, у меня бывают обычно визуального или изредка двигательного типа. Слова или другие условные знаки приходится подыскивать (с трудом) только во вторичной стадии, когда эта игра ассоциаций дала некоторый результат, и может быть при желании воспроизведена».

Базовые компоненты языка мышления по Пинкеру – это совокупность абстрактных концептуальных структур, организующих наш опыт, – пространство, время, материя, причинность, сила, число. Более конкретно, состав этих компонент выглядит так:

- Система базовых понятий: событие, состояние, предмет, направление, место, свойство;
- Система отношений, связывающих эти понятия: действие, движение, существование, обладание;

- Система пространственных понятий, определяющих место и направление и выражаемых в предлогах «на», «у», «в», «к», «над», «под»;
- Линия времени, выстраивающая события и разграничивающая события а) мгновенные, б) ограниченные промежутками, и в) имеющие неограниченную протяженность;
- Совокупность причинно-следственных отношений: принуждение, позволение, способствование, предотвращение, препятствование, поощрение;
- Понятие цели и различие средства и цели.

Второй важный тезис концепции Пинкера – мышление и язык навязывают нам структуру мира. «Наши когнитивные модели представляют собой данные, считанные с основных аспектов природы человека. Каждая из этих моделей осмысления отвечает отчетливо человеческим целям, и они позволяют нам членить материю, пространство, время, причинно-следственные связи так, как это важно в первую очередь для реализации целей человека в природе и обществе.»¹

Наши когнитивные модели – это наш здравый смысл, наше обыденное мышление. Нашу картину мира строит наш мозг, и эта картина (интуитивная геометрия, интуитивная физика) может сильно отличаться от того, каким мир является на самом деле, т.е. каким его представляет научное знание.

Самый простой пример – хорошо изученные иллюзии зрения, с которыми, впрочем, взрослый мозг учится справляться на основе приобретаемого опыта. Описание пространства в нашем языке сильно отличается от представлений геометрии. С точки зрения геометрии трехмерное тело ограничено двумерной поверхностью, поверхность – одномерной линией, линия – нульмерной точкой. Однако слово «конец» может означать короткую линию, ограничивающую ленту, и двумерную поверхность, ограничивающую брусок. Здесь проявляется еще одна особенность человеческого сознания – способность «структурировать даже самые непримечательные повседневные события более чем одним способом». В момент, когда мы говорим о конце ленты, бруска, дороги, наше мышление структурирует их как одномерные объекты, пренебрегая другими их измерениями, хотя в других ситуациях их ширина имеет значение.

¹ В дальнейшем все тексты, взятые в кавычки, - это цитаты из русского перевода книги [7].

Но, кроме того, мы говорим и о конце процесса: рабочего дня, домашней уборки, футбольного матча. Это связано с тем, что время в мышлении репрезентируется пространственными метафорами. Время – это линия (см. выше состав базовых компонент мышления); но относительно наблюдателя возможны три вида метафор, т.е. три вида структурирования отношения время-наблюдатель (см. также [10]). В метафоре ориентации во времени время образует неподвижный ландшафт, наблюдатель также неподвижен. Он находится в настоящем времени, прошлое расположено позади, а будущее – перед ним: *надо смотреть вперед, а не оглядываться назад; все еще впереди*. В метафоре движущегося времени время проходит мимо движущегося наблюдателя: *пришло время обеда; Новый год приближается*. В метафоре движущегося наблюдателя время – снова ландшафт, по которому движется наблюдатель: *этот этап мы прошли успешно*.

Всю историю физики можно рассматривать как историю разоблачения интуитивных представлений о физическом мире. Только изучая физику, человек с удивлением узнает, что всякое тело продолжает удерживаться в состоянии покоя или равномерного и прямолинейного движения, пока внешние силы не изменят это состояние, и что все тела (тяжелые и легкие) падают с одинаковым ускорением. Что уж говорить о теории относительности и квантовой физике, которые даже многие профессионалы при первом знакомстве называли их насилием над здравым смыслом.

Семантика дискурса

Термин «дискурс» стал популярным в последние десятилетия, хотя часто употребляется не по делу, а «чтобы свою ученость показать». Наиболее конструктивно дискурс можно определить как текст или речь, смысл которых нельзя адекватно понять без учета социально-психологических, профессиональных и других контекстов. Некоторые лингвистические характеристики определенного дискурса (например, специфический словарь) также определяются этими контекстами. Дискурс может характеризовать определенную социокультурную группу, политическую идеологию и т.д. Сам Пинкер термин «дискурс» не употребляет, однако существенную часть его концепции удобно назвать семантикой дискурса.

В основе семантики дискурса, обсуждаемой Пинкером, лежит типология общественных взаимоотношений между людьми, развитая антропологом Аланом Фиске [11], состоящая из четырех типов отношений. Первый тип основан на солидарности, второй – на власти, третий – на социальном обмене, четвертый – на рыночных расценках.

Тип, основанный на солидарности, – это общественное совладение, «коммунальность». В его основе – желание того, чтобы другие хотели для вас того же, что вы желаете себе сами. Коммунальность возникает

естественным образом между кровными родственниками. Кроме того, в коммунальное сообщество людей могут объединять общие вкусы или интересы, общие враги, землячества в чужой стране. Взаимные обязательства, накладываемые коммунальностью, не обсуждаются, а принимаются по умолчанию. Хороший пример – брачный контракт: люди, как правило, не любят заключать брачные контракты именно потому, что явно прописанные в них обязательства могут разрушить предполагаемый дух коммунальности в будущей семье. Коммунальности присуща специфическая лексика и типичные словесные обороты: братство, единство, солидарность, преданность, «мы – одна семья» и т.д.

Второй тип взаимоотношений – «ранжирование авторитетов»; это власть, статус и господство. Это ранжирование фиксировалось в дворянских званиях Европы, табели о рангах России XVIII-XIX веков, соответствующей лексике обращений («ваше величество», «ваша светлость», «ваше высокоблагородие») и в различных ритуалах. В наше время это ранжирование сохраняется в военной, бюрократической, церковной иерархиях, а ритуалы – в протоколах официальных приемов, отдавании чести старшему по званию в армии и т.д.

Третий тип взаимоотношений – «соблюдение равенства», основанное на обмене и общих представлениях о справедливости. Сюда относятся согласованные правила распределения ресурсов, установление очередей, обмен услугами или одолжениями и т.д. В отличие от первых двух типов, язык обмена, как правило, явный, поскольку он требует явных договоренностей.

Если первые три типа присущи всем человеческим цивилизациям, включая самые примитивные, то четвертый тип – «рыночная расценка» – характерен только для развитых цивилизаций. Он содержит компоненты современных рыночных экономик: валюта, цены, зарплаты, социальные пакеты, ренты, процентные ставки, кредит и т.д. Средством общения являются цифры, математические операции, цифровая бухгалтерия, а также язык официальных контрактов. Многие механизмы рыночной экономики выглядят неестественными с точки зрения обычных когнитивных процессов, поскольку они не укладываются в когнитивно понятный третий тип «обмена лицом к лицу». Поэтому «люди склонны считать, что у каждого предмета есть некая присущая ему внутренняя правильная цена, что посредники – паразиты (несмотря на то, что они привозят товары из отдаленных точек Земли) и что взимание процентов – это аморальный акт (несмотря на то, что деньги стоят по-разному в зависимости от времени)».

Человек на протяжении своей жизни – а часто и в течение одного дня – участвует в разных дискурсах, и умение распознавать адекватный

социальный контекст и соответствующий дискурс – важная часть социального опыта. Неформальный стиль общения в ближнем кругу неуместен в официальной обстановке. Человек, привыкший общаться среди чиновников с четким ранжированием авторитетов, плохо понимает, как себя вести в гораздо более свободной академической среде. А для бывшего офицера даже атмосфера гражданского чиновничества кажется слишком свободной.

Навязывание нужного дискурса и соответствующей лексики – важная составная часть различных видов манипулирования сознанием в рекламных и политических кампаниях, а также в информационных войнах. Успех таких кампаний объясняется тем, что лексика агрессивного дискурса, как правило, носит сильно оценочный характер (революционер vs экстремист, стабильность vs застой и т.д.) и формирует у принявшего дискурс нужное отношение к событиям, взглядам и т.д.

Заключение

Основные тезисы концептуальной семантики Пинкера сводятся к следующим.

- Наши когнитивные модели обусловлены природой человека. Каждая из этих моделей отвечает человеческим целям.

- Концептуальная семантика – язык мышления первичен по отношению к собственно языку, т.е. к вербальному языку.

- Базовые компоненты языка мышления – это совокупность абстрактных концептуальных структур, организующих наш опыт, – пространство, время, материя, причинность, сила, число.

- Язык структурирует мир. Наши когнитивные модели позволяют нам членить мир так, как это важно для реализации целей человека в природе и обществе.

- Мышление и язык навязывают структуру мира, в результате чего интуитивная картина мира – как физического, так и социального – может сильно отличаться от научной. С одной стороны, исследования языка позволяют раскрыть особенности когнитивных моделей обыденного сознания (отсюда – подзаголовок в названии книги [7] «Язык как окно в человеческую природу»). С другой стороны, ограниченность интуитивных моделей говорит о важной роли образования, позволяющего компенсировать эту ограниченность.

- Наша речь имеет не только лингвистические, но и социальные характеристики, образуя определенный дискурс. Правильное распознавание дискурса ведет к адекватному социальному поведению.

Литература

1. Черч А. Введение в математическую логику. – М: ИЛ, 1960.
2. Рахилина Е.В. Когнитивная семантика: История. Персоналии. Идеи. Результаты. // Семиотика и информатика, 1998, вып. 36, 274-323.
3. Sowa J.F. Conceptual Structures - Information Processing in Mind and Machines. Addison-Wesley Publ.Comp. 1984.
4. Lakoff J. Women, Fire, and Dangerous Things: What Categories Reveal About the Mind. – University of Chicago Press, 1987. (Русский перевод: Лакофф Дж. Женщины, огонь и опасные вещи: Что категории языка говорят нам о мышлении, М. 2004).
5. Кузнецов О.П. Когнитивная семантика и искусственный интеллект.// Искусственный интеллект и принятие решений, 2012, №4, с.32-42.
6. Rosch E. Cognitive representations of semantic categories. Journal of Experimental Psychology, 1975. 104, pp.192-233.
7. Pinker S. The Stuff of Thought, 2007. Русский перевод: Пинкер С. Субстанция мышления: Язык как окно в человеческую природу. М.: Книжный дом «ЛИБРОКОМ», 2013.
8. Вертгеймер М. Продуктивное мышление. М.:Прогресс, 1987.
9. Адамар Ж. Исследование психологии процесса изобретения в области математики. М.:Сов. радио, 1970.
10. Boroditsky L. Metaphoric structuring: understanding time through spatial metaphors. Cognition, 2000, 75 (1), p.1-28.
11. Fiske A.P. The four elementary forms of sociality: framework for a unified theory of social relations. // Psychological Review, 99, 689-723.

УДК 007:681.512.2

Интеллектуализация удаленного процесса обучения иностранному языку

Бабенко А.Е., аспирант, Международный научно-учебный центр ин-формационных технологий и систем НАН и МОН Украины

Abstract. The most problem of current researching is intellectualization of remote tutoring. Almost we are concentrated on communications between student, server and tutor-expert and give diagrams of organization of studying process.

Key words : intellectual tutoring, foreign languages, quality of knowledge.

В работе рассматривается интеллектуализация процесса удаленного обучения иностранному языку, работа в схеме обучаемый-сервер-эксперт, представлены схемы организации процесса обучения.

Ключевые слова: интеллектуальное обучение, иностранные языки, качество знаний

Введение

Объектом исследования является процесс интеллектуализации удаленного обучения иностранному языку. *Предметом* исследования является взаимосвязь между обучаемым, сервером, экспертом, при схеме обучения человек-машина-эксперт.

Удаленное обучение позволяет учиться в любое время и в любом месте, когда у учащегося появляется свободная минута. С каждым днем мир все более и более глобализирован, что приводит к доминированию макро-региональных языков в каждом уголке планеты, как например английского в целом, испанского в большей части Американского континента, немецкого - в странах центральной и Восточной Европы и т.д. Знание иностранных языков необходимо, чтобы идти в ногу со временем, однако традиционные методики изучения языка с уровня А0 до В2(С1), позволяющего вести бизнес-переговоры занимает не менее 18 месяцев при занятиях на курсах с интенсивностью 3-4 раза в неделю. Такое количество времени не под силу выделить большинству людей, на помощь которым приходит удаленное интеллектуальное обучение.

1. Интеллектуальная деятельность и удаленное обучение

1970-е годы дали мощный толчок к развитию интеллектуальных систем, основанных на знаниях, что породило большие надежды, связанные с их дальнейшим практическим применением. Однако за последние годы, несмотря на теоретические успехи и успешных проектов таких систем, не произошло их массовое внедрение.

Процесс удаленного интеллектуального обучения иностранным языкам мы можем рассматривать в качестве интеллектуальной деятельности, в которой машина, основываясь на знания об изучаемом предмете, самостоятельно принимает решения [1,2].

В данных условиях на первое место выходит борьба за качество интеллектуальной деятельности.

Качество интеллектуальной деятельности является метой риска ошибочных решений и характера ущерба от последствий таких ошибочных решений. Правильность решений определяется степенью надежности процесса применения знаний, что в свою очередь зависит от нескольких факторов – *правильности знаний* (частота правильного принятия решений), *точности знаний* (однозначность полученного решения). *Процесс управления качеством* интеллектуальной деятельности состоит, прежде всего, в повышении *качества (правильности и точности) знаний*, используемых системой и экспертом, который играет непосредственную роль в ее функционировании.

На рисунке 1 показана схема первичной работы между удаленной системой интеллектуального обучения и обучаемым. Благодаря первичному тесту происходит сбор информации и генерирование учебного плана

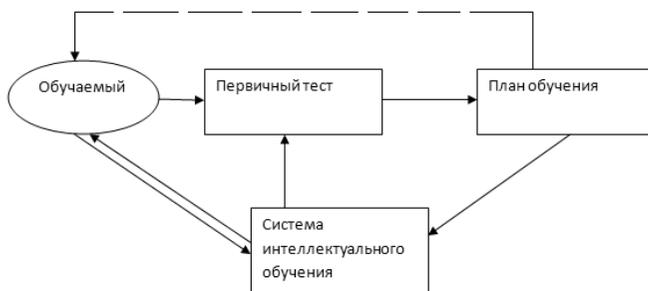


Рис. 1. Первичный сбор информации и генерирование учебного плана

На рисунке 2 показана схема интеллектуальной деятельности в процессе удаленного обучения иностранным языкам. Эксперт в данной схеме постоянно работает над улучшением системы знаний, чтобы снизить риск принятия неверного решения и написания заведомо неверного плана обучения. Для оценки текущего уровня качества интеллектуальной деятельности и управления этим качеством можно использовать отчеты о результатах деятельности, включающие информацию о решениях, принимаемых системой интеллектуального обучения касательно необходимого подхода к обучаемому и планов обучения.

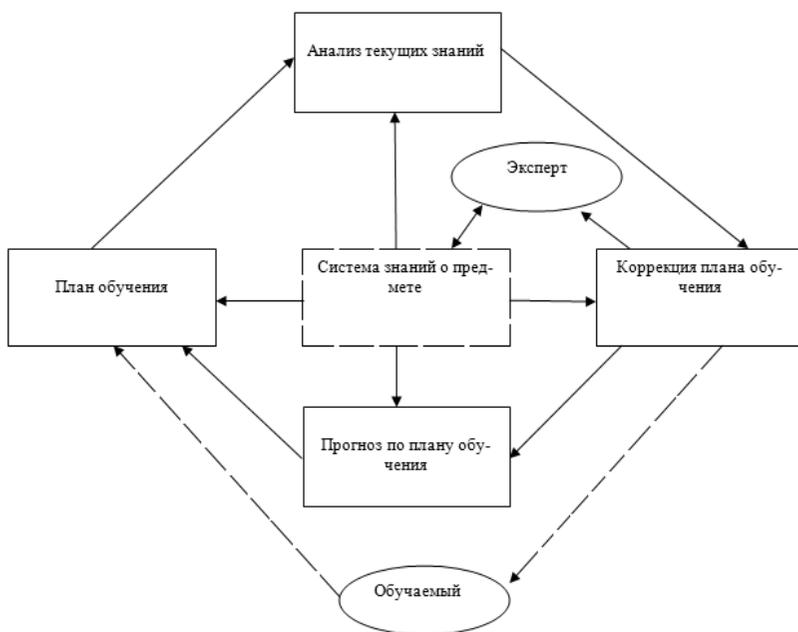


Рис.2. Схема интеллектуальной деятельности в интеллектуальном обучении

В нашей системе три действующих лица : обучаемый, цель которого изучить необходимый материал(в данном конкретном случае иностранный язык), удаленная интеллектуальная обучающая система, которая, опираясь на базу знаний в области иностранных языков дают по плану учебный материал и принимает решение касательно успеш-

ности прохождения данного материала или корректировки плана обучения. Третьим действующим лицом выступает эксперт(группа экспертов), который отслеживает деятельность интеллектуальной обучающей системы в плане принятия решений, занимается усовершенствованием базы знаний для снижения риска принятия неправильного решения.

2. Поддержка принятия решений

Распишем подробнее механизм принятия решений.

В работе [3] была рассмотрена функция знаний, когда каждому направлению присваивалась переменная X_i , которая может принимать значения от 0 до 1 в зависимости от степени усвоения материала обучаемым(формализация результатов обучения).

На рисунке 3 представлена схема принятия решения удаленной интеллектуальной обучающей системой. На первых этапах работы риск принятия неверного решения сохранится достаточно высоким, так как невозможно включить все возможные варианты и, чтобы минимизировать риск неверного дальнейшего обучения необходимо иметь прямую и обратную связь с экспертом (группой экспертов)

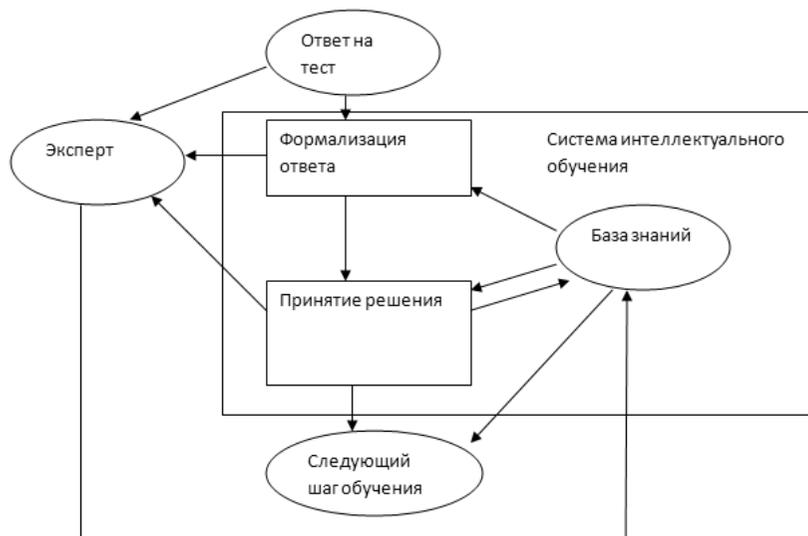


Рис.3. Схема работы по принятию решения

Также в базу знаний, помимо правок, внесенных экспертом закладывается информация обо всех возможных инцидентах с неверным оценением для того, чтобы избежать принятия неверного решения в дальнейшей работе удаленной интеллектуальной обучающей системы.

3. Основные затраты на проектирование и автоматизацию интеллектуальной деятельности

Трудозатраты на автоматизацию интеллектуальной деятельности удаленной интеллектуальной обучающей системы могут быть направлены на *системную инженерию* [4,5], на *программное обеспечение* и на *базы знаний* (включая управление ими) [6], а также на вспомогательные и организационные процессы, например, на процесс создания и сопровождения инфраструктуры (работа серверов и т.п.).

Трудозатраты на системную инженерию могут в себя включать:

- проведения системного анализа профессиональной деятельности в предметной области,
- концептуального проектирования системы автоматизации,
- интеграции всех подсистем в единую базу знаний.

Трудозатраты на программное обеспечение состоят в разработке, а затем в сопровождении следующих видов программных средств :

- Система принятия решений,
- редактор знаний (с встроенным контролем внутренних свойств знаний),
- подсистема документирования и архивирования,
- подсистема модификации базы знаний,
- подсистема редактирования базы знаний.

В свою очередь разработка каждого из этих программных средств складывается из трудозатрат на:

- анализ (уточнение) требований к программному средству,
- проектирование его программной архитектуры (в том числе баз данных и других хранилищ информации),
- техническое проектирование программных средств,
- реализацию и тестирование программных средств;
- сборку программных средств;
- квалификационные испытания программных средств.

Трудозатраты на формирование базы знаний складываются из затрат:

- на разработку первого варианта каждой базы знаний (с помощью редакторов знаний и подсистемы оценивания варианта базы знаний);
- на сопровождение и управление базой знаний.

Таким образом, автоматизация интеллектуальной деятельности (даже в рамках одного раздела предметной области) связана с типичным спектром трудозатрат на системную инженерию и разработку программного обеспечения, а также с особым рода затратами на разработку баз знаний, поскольку роль базы знаний и способ организации отличаются от других хранилищ данных.

Заключение

В статье проведено обоснование интеллектуализации удаленного обучения иностранным языкам, работа с базой знаний, которую формируют обучаемый и эксперт. Происходит формирование базы знаний, которая доступна всем обучающимся из любой точки мира.

Позитивным моментом внедрения удаленной интеллектуальной обучающей системы для изучения иностранного языка является то, что интеллектуальное удаленное обучение позволит изучать иностранный язык быстрее и дешевле, чем традиционным способом. Также снижаются расходы на обучение, так как система глобальна и каждый случай обучения сохраняется в истории, что приведет к минимально-возможному риску неправильной подачи материала в процессе обучения. Минусом безусловно будут первоначальные высокие затраты на внедрение и на экспертную поддержку для мониторинга процесса принятия решений.

Литература

1. *Клещев А.С., Черняховская М.Ю., Шалфеева Е.А.* Парадигма автоматизации интеллектуальной профессиональной деятельности. Часть 1. Особенности интеллектуальной профессиональной деятельности
2. *Клещев А.С., Черняховская М.Ю., Шалфеева Е.А.* парадигма автоматизации интеллектуальной профессиональной деятельности. Часть 2. Парадигма автоматизации отрасли

3. *Бабенко О.С.* Интеллектуальная система когнитивного обучения иностранному языку и алгоритм ее работы // Інтелектуальний аналіз інформації: наук.праці XIII міжн.наук.конф. ІАІ-2012 ім. Т.А.Таран (15-17 травня 2013р., Київ).-К., 2013 с.4-124.ГОСТ Р ИСО/МЭК 15288-2005 Информационная технология. Системная инженерия. Процессы жизненного цикла систем. – М.: Стандартиформ, 2005. – 57 с.
5. *Клецев А.С., Шалфеева Е.А.* Системный анализ при автоматизации интеллектуальной профессиональной деятельности // XIII Национальная конференция по искусственному интеллекту с международным участием «КИИ-2012», 16–20 октября 2012 г. ISBN: 978-5-361-00182-8 Труды конференции, т.2. - Белгород: Изд-во БГТУ, 2012. - С.128–135.
6. *Клецев А.С., Грибова В.В.* Управление интеллектуальными системами. Известия РАН. // Теории и системы управления. 2010. № 6. - С. 122-137.

УДК 519.688

Модель оцінювання ймовірності виникнення розладання у нестационарних процесах з множинними розладаннями

д.т.н., проф., Байбуз О.Г., Луценко О.П., Дніпропетровський національний університет імені О.Гончара, Дніпропетровськ,

lutsenkoop@yandex.ua

Описана розроблена авторами модель оцінки ймовірності розладання нестационарного процесу, заснована на непараметричному відновленні розподілу вектору характеристик розладань з використанням В-сплайн апроксимації. Модель є частиною системи інтелектуального аналізу фінансового ринку Forex.

Вступ

Задача визначення моментів зміни статистичних властивостей випадкових сигналів актуальна в багатьох прикладних напрямках, пов'язаних з обробкою часових рядів, зокрема у геології, економіці, контролі технологічних процесів, тощо.

Послідовні методи визначення розладання характеризуються тим, що при надходженні кожного нового спостереження або приймається рішення про наявність розладання, або про продовження спостережень. Очевидно, що для прийняття обґрунтованого рішення про розладання необхідно отримати деяке число спостережень, що описуються новою моделлю, при цьому виникає затримка виявлення. При зменшенні порогу чутливості алгоритмів затримка зменшується, але збільшується кількість помилкових сигналів про розладання.

В той же час, існують предметні області, в яких важливим є не тільки швидке реагування на подію розладання, але і знання ризику такої події. Прикладом такої предметної області можуть бути, зокрема, часові ряди біржових котирувань, де розладання супроводжує зміну напряму діючого тренду. Враховуючи, що в предметній області фінансової торгівлі знання величини ризику може використовуватися для гнучкого управління об'ємами торговельних позицій, можна стверджувати, що для даної галузі задача має особливу актуальність, і наявність підсистеми аналізу ризику зміни тренду в найближчий час стане необхідною умовою для створення сучасної інтелектуальної системи аналізу фінансового ринку.

Авторами пропонується модель, що може бути застосована до процесу з множинними розладнаннями і дозволяє оцінити ймовірність виникнення розладнання на певному проміжку часу в майбутньому, а також ймовірність належності значення процесу в момент розладнання до заданого числового інтервалу.

Опис пропонованої моделі

Нехай після розбиття графіка маємо n відрізків стаціонарності. Кожен з відрізків характеризується вектором параметрів v , що включає його тривалість у часі (довжину відрізка по осі $x - xS$) і різницю між цінами на початку і в кінці відрізка (різниця по осі $y - yS$). Таким чином, процес описується вектором випадкових величин (xS, yS) .

Пропонована авторами модель заснована на припущенні, що даний вектор є випадковою величиною, вигляд функції розподілу якої повільно змінюється з часом. Відтворюючи функцію щільності розподілу на певному інтервалі часу у минулому і оновлюючи її з надходженням нових даних про розладнання, отримуємо можливість судити про ймовірність певних сценаріїв поведінки процесу у майбутньому.

Відтворення щільності розподілу вектору властивостей розладнань

Завдання непараметричного відтворення щільності розподілу за гістограмою полягає в побудові апроксимуючої функції на прямокутній сітці з відновленням проміжних значень. Перші результати в цьому напрямку отримані в роботах [1, 2], де в якості оцінки невідомої щільності ймовірності була розглянута гістограма. В [1] була встановлена майже напевно рівномірна збіжність гістограми до безперервної щільності ймовірності, в [2] отримано граничний закон розподілу для максимуму абсолютної величини нормованого відхилення гістограми від теоретичної гладкої щільності.

Форма розподілу вказаного раніше вектору може варіюватися від процесу до процесу, а також у залежності від використовуваного методу та його налаштувань. Зокрема, розподіл, отриманий при застосуванні алгоритму кумулятивних сум до ряду біржових котирувань, характеризувався вираженою бімодальністю. У зв'язку з наявністю таких відмінностей, а також з тим, що ключові особливості розподілу не завжди зберігаються при використанні параметричних методів, для відтворення щільності розподілу було обрано апроксимацію розподілу B-сплайнами за гістограмою. Даний метод є універсальним щодо форми функції щільності і порівняно швидким за рахунок локальності B-сплайнів. При низькій об-

числювальній складності [3, 4] В-сплайни мають гарні апроксимуючі властивості. Крім того, В-сплайни мають властивість локальності, що дозволяє перераховувати тільки частину функції при зміні вхідних даних. Останнє особливо актуально з урахуванням того, що в процесі отримання нових ринкових даних необхідно проводити оновлення функції щільності.

Побудуємо гістограму розподілу вектору (x_s, y_s) . Точки гістограми будуть служити вузлами сплайна. Двовимірну В-сплайн функцію запишемо у вигляді тензорного добутку одновимірних В-сплайнів, побудованих на кожній з двох осей координат [5]:

$$x(s, t) = \sum_{i=0}^n \sum_{j=0}^m P_{i,j} B_i(s) B_j(t),$$

$$B_{k,1} = \begin{cases} 1, & t_k < t < t_{k+1} \\ 0 & \text{в іншому випадку} \end{cases}$$

$$B_{k,d} = \left(\frac{t - t_k}{t_{k+d-1} - t_k} \right) B_{k,d-1}(t) + \left(\frac{t_{k+d} - t}{t_{k+d} - t_{k+1}} \right) B_{k+1,d-1}(t)$$

де s, t – координати точки,

$P_{i,j}$ – значення функції в точці i, j ;

$B_{k,d}(t)$ – значення стикувальної функції порядку d в точці t , що належить до інтервалу $(t_k; t_{k+1})$;

$t_k; t_{k+1}$ – пари сусідніх вузлових точок сплайну.

Приклад результату відтворення функції щільності розподілу вектору (x_s, y_s) наведений на рисунку 1.

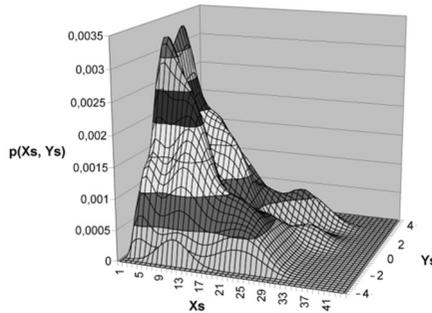


Рис. 1. Відтворена функція щільності $p(x_s, y_s)$

Розрахунок функцій ризику розладнання

Знаючи щільність розподілу вектору, а також поточне (з моменту останнього зафіксованого розладнання до теперішнього моменту) значення $(x_s, \Delta y)$, можемо обчислити функцію ризику, що виражає ймовірність розвороту тенденції на наступному кроці спостережень. Ймовірність розвороту тренду визначимо як відношення ймовірності розладнання на наступному кроку спостережень P_1 до ймовірності виникнення розладнання на всіх інших інтервалах, до яких величина, що спостерігається, може належати на наступних кроках спостереження P_2 з урахуванням неспадаючої природи x .

$$P = \frac{P_1}{P_2},$$

$$P_1 = \int_x^{x+h} \int_{-\infty}^{\infty} p_1(x, y) p_{2,h}(y) dy dx, \quad (1)$$

$$P_2 = \int_{x+h-\infty}^{\infty} \int_{-\infty}^{\infty} p_1(x, y) p_{2,h}(y) dy dx,$$

де $p_1(x, y)$ – відновлена щільність розподілу вектору координат розладнань,

$p_{2,h}(y)$ – щільність розподілу прирощень ряду котирувань за h проміжків часу.

Для окремих задач має значення не тільки ймовірність розладнання, але також інтервал, до якого в момент розладнання належатиме величина, що спостерігається. В цьому випадку, можливі інші інтегральні функції, зокрема, ймовірність потрапляння значення ряду при розладнанні до інтервалу $[b_1; b_2]$:

$$P = \int_x^{b_2} \int_{b_1}^{\infty} p_1(x, y) p_{2,h}(y) dy dx, \quad (2)$$

Висновки

Запропонована модель була включена до програмного пакету інтелектуального аналізу фінансового ринку Forex, де була використана в якості основи для систем автоматичної торгівлі (торгівельних роботів).

Знання точної величини ризику розвороту ринкового тренда, обчисленої за формулами (1) дозволяє формалізувати прийняття рішень про відкриття і закриття торгівельних позицій.

Формула (2) дають можливість автоматизувати вибір величини тейк-профіт (ціни, при досягненні якої поточна позиція закривається), виходячи з математичного очікування прибутку при різних значеннях тейк-профіт.

Отримуючи на вході ці сигнали, на виході підсистема аналізу стану ринку повертає сигнали до відкриття/закриття позицій, об'єм позиції і величини тейк-профіт і стоп-лосс, дозволяючи ефективно проводити торгівлю.

Окрім предметної області фінансового ринку, отримані функції можуть бути використані у інших сферах, де важливий час прийняття рішення, і ризики розладнання мають бути оцінені до появи чергового розладнання.

Перспективи подальших досліджень у даному напрямку: визначення оптимальних величин вибірок для відновлення щільності; підбір оптимальних параметрів відновлення залежно від тимчасового масштабу даних; аналіз ефективності торгових стратегій, заснованих на застосуванні отриманої функції.

Література

1. Гливенко, В. И. Курс теории вероятностей: учебник для физико-математических факультетов государственных университетов / В. И. Гливенко. — М., Л.: ГОНТИ НКТП СССР, Редакция технико-теоретической литературы — 1939. — 220 с.
2. Смирнов, Н. В. Приближение законов распределения случайных величин по эмпирическим данным / Н. В. Смирнов // *Успехи Матем.Наук* — 1944 — N 10 — С. 179-206.
3. Завьялов, Ю. С. Методы сплайн-функций / Ю. С. Завьялов, Б. И. Квасов, В. Л. Мирошниченко — М.: "Наука" — 1980. — 352 с.
4. De Boor, C. Package for Calculating with B-Splines / C. De Boor // *SJAM J. On Numer. Anal.* — 1977 — vol. 14, N3 — p. 441—472.
5. Rogers D.F. An Introduction to NURBS with Historical Perspective — San Francisco: Morgan Kaufmann Publishers — 2000 — 344p.

УДК 004.01

Разработка онтологии документооборота выполнения заказа на предприятии

К.т.н., проф. Боргест Н.М.^{1,2}, Бурдюгова В.В.¹

*¹Самарский государственный аэрокосмический университет
(национальный исследовательский университет), г. Самара, Россия,*

*²Институт проблем управления сложными системами РАН
borgest@yadex.ru, damnkid@yandex.ru*

В статье рассматривается онтологическое моделирование документооборота, сопровождающего подготовку заявки и выполнения государственного заказа. На примере НКП «Разумные решения» (г. Самара) анализируется структура отдела, который выполняет задачи, связанные с организацией документооборота по подготовке и выполнению госзаказа.

Введение

Для организации взаимодействия различных отделов какого-либо предприятия, а также взаимодействия с внешними информационными системами необходимо обеспечить решение задачи организации её документооборота. Отсутствие универсальных механизмов управления документооборотом исполнения государственного заказа или контракта и систем мониторинга его реализации является причиной недостаточно эффективной организации взаимоотношений между поставщиком и заказчиком, а впоследствии мешает гибкому и конструктивному управлению отношениями между ними в процессе выполнения заключенного контракта.

Метод решения

Перспективной технологией решения задачи организации документооборота между поставщиком и заказчиком является онтологическое моделирование. На основе этой технологии можно создать информационные системы, основанные на онтологиях (Ontology-Driven Information Systems, ODIS) [1], которые позволят решить следующие группы задач:

- создание единого словаря терминологии предприятия/отдела;
- построение иерархии понятий (таксономии), характеризующих деятельность предприятия/отдела;
- оптимизация поиска и навигации в электронных ресурсах;
- решение проблемы интеграции разнородных баз и хранилищ данных;

- обеспечение интеграции информационной системы предприятия с системами других предприятий и глобальным информационным пространством на основе общих принципов и стандартов.

Управление исполнением государственного контракта в общем виде представляет собой мониторинг хода выполнения контракта с целью обеспечения «уверенности в успешности процедур конкурса» [2]. По определению О.Б. Ломакиной данный процесс это – «выполнение уполномоченными государственными служащими функций, связанных с принятием всех технических и административных решений с момента присуждения контракта до его успешного выполнения или расторжения, контроль за оплатой поставленного товара и урегулирование споров» [2]. Такое управление необходимо для координации действий заказчика и поставщика товаров/услуг, выявления проблем и рисков по исполнению госзаказа на условиях заключаемого контракта, а также для решения этих проблем и минимизации степени рисков.

Создание онтологии позволяет при организации документооборота решить или облегчить решение ещё ряд важных задач:

- анализ документооборота предприятия;
- определение групп сотрудников, которым следует рассылать внутренние или внешние документы для ознакомления и дальнейшей работы с ними;
- формирование групп специалистов при разработке или рассмотрении документов.

Главной задачей при создании онтологии документооборота госзаказа является идентификация предметной области деятельности отдела предприятия, занимающегося госзаказами и контрактами. Основой создания такой онтологии конкретного предприятия являются анализ, моделирование и оптимизация бизнес-процессов организации и, в частности, данного отдела. Исследование моделей бизнес-процессов позволяет не только выделить множество понятий, характеризующих объекты и процессы организации и отдела, но и определить их принадлежность к конкретным исполнителям, определить предметную область. Изучаемые предметные области различаются: по физической сути процессов; по виду структуры связей элементов, что позволяет их классифицировать [3].

Описание онтологии

Для примера, рассмотрим отдел «Аналитический центр» самарской научно – производственной компании «Разумные Решения» («Smart Solutions» - <http://www.smartsolutions-123.ru/>), который непосредственно работает с госзаказами и контрактами. Структура отдела «Аналитический

центр» представлена на рисунке 1. Отдел возглавляет директор аналитического центра. В его состав входит группа государственных контрактов, группа коммерческих контрактов, группа технической документации и группа международный рынок. В своей деятельности отдел руководствуется уставом компании [4].

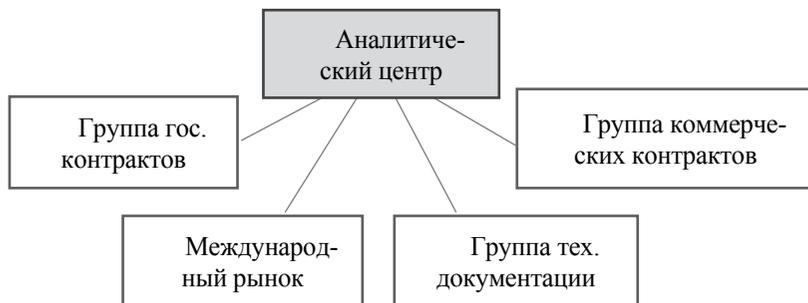


Рис. 1. Структура отдела «Аналитический центр»

Для построения онтологии используется конструктор онтологий, который представляет собой комплекс инструментальных средств для управления онтологиями, моделями, сценами и агентами.

В конструкторе онтологий для удобства построения различных элементов концептуальной модели выделяются три уровня описания предметной области: «онтология» - «модель» - «сцена»:

- онтология описывает понятия и отношения, необходимые для описания знаний в любой предметной области (аэрокосмические организации, биотехнологии, медицина, наносистемы, живые системы и т.д.); на основе онтологии в дальнейшем строится модель деятельности;
- модель описывает устойчивые конфигурации объектов, упрощающие создание формализованных описаний сцен в конкретных ситуациях (рисунок 2);
- сцена описывает экземпляры понятий и отношений в заданный момент времени (набор фактов) [5].

Заказчик, в данном случае государственный, (онтологический объект) объявляет конкурс на заказ, который обладает своими атрибутами, такими как тема, сроки, финансирование, требования. Участники конкурса - компании, организации, объекты, которые оформляют заявку на участие и дальнейшее выполнение государственного контракта.

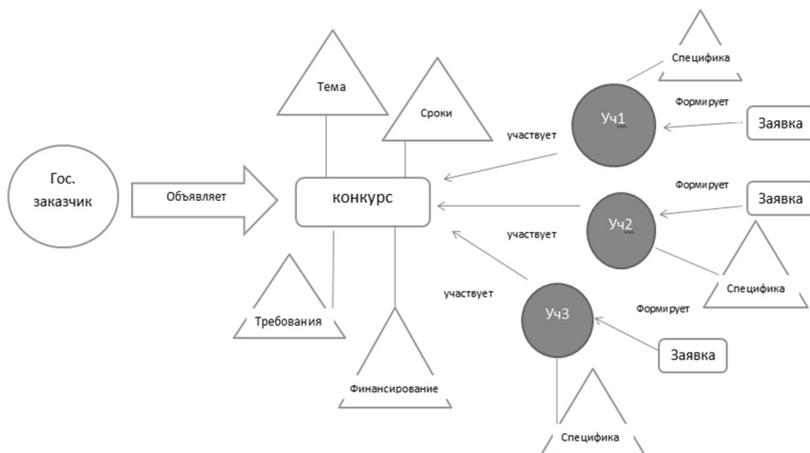


Рис.2. Модель «Государственный заказчик и потенциальные поставщики»

Сцена описывает экземпляры понятий («сведения», «титальный лист») и отношения («оформить», «приложить»). Заявка обладает условиями (1...n - условий), которые должны быть соблюдены, иначе заявка не пройдет по конкурсу (см. рисунок 3).

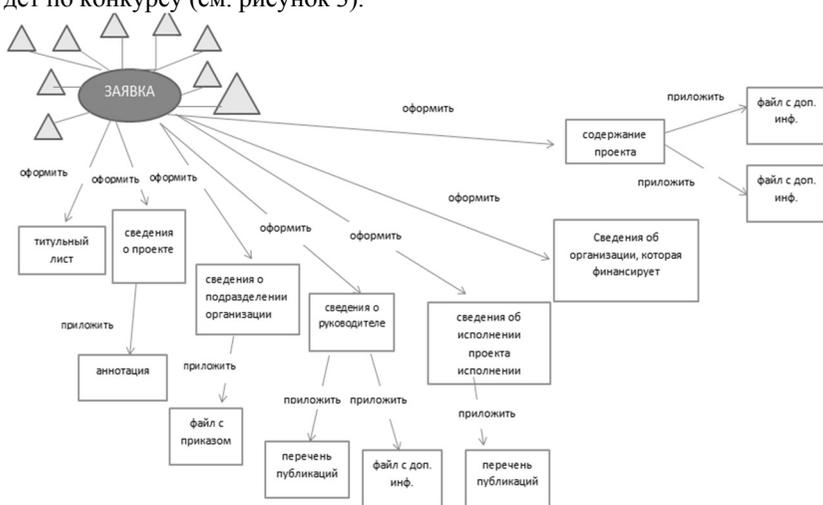


Рис.3. Сцена формирования заявки на выполнение государственного контракта

Выводы

С помощью онтологического моделирования может быть решена задача управления документооборотом при подготовке и выполнении государственного заказа и других контрактов как внутри предприятия, так и между поставщиком (исполнителем) и заказчиком, что позволяет построить более чёткие и эффективные отношения между ними.

Благодарности

Работа выполнена в рамках Государственного контракта № 07.524.12.4022 от 11.07.2012 года по созданию распределенной интеллектуальной системы согласованного управления производственными цехами корпораций машиностроительных предприятий, построенных по сетевому принципу.

Литература

1. *Burcu Yildiz and Silvia Miksch*. Ontology-Driven Information Systems: Challenges and Requirements [Электронный ресурс]. - http://publik.tuwien.ac.at/files/pub-inf_4601.pdf
2. *Ломакина О.Б.* Подготовка, контроль и управление исполнением контракта при проведении конкурсных торгов - М.: Tacis-International Training Center of the ILO, - 2003. - 318 с.
3. *Кузьменко В.В.* Онтология формально-логического моделирования семиотических систем электронного документооборота / Гилея научный вестник. Сборник научных трудов /Гл. ред. В.М. Вашкевич. - К.: ВИР УАН 2011. - Выпуск 44. - с.48-54.
4. *Скобелев П.О.* Презентационные материалы «Основные принципы реформирования компании в 2013 году». – Самара: НКП «Разумные решения». - 2013. – 16 с.
5. *Вакурина Т.Г., Котеля В.В., Лахин О.И., Матюшин М.М., Скобелев П.О.* Онтология российского сегмента Международной космической станции и ее практическое использование в интеллектуальных аэрокосмических приложениях // Труды IV Международной научно-технической конференции «Открытые семантические технологии проектирования интеллектуальных систем» Open Semantic Technologies for Intelligent Systems (OSTIS-2014), 20 – 22 февраля 2014 г., Минск, Республика Беларусь. – 2014. – С.221-226.

УДК 004.822

Разработка онтологии нештатных ситуаций на космической станции

К.т.н., проф. Боргест Н.М.^{1,2}, Сокова Н.А.¹

¹*Самарский государственный аэрокосмический университет (национальный исследовательский университет), г. Самара, Россия,*

²*Институт проблем управления сложными системами РАН
borgest@yadex.ru, natalyasokova@mail.ru*

В статье рассмотрен подход к разрабатываемым методам и алгоритмам построения онтологий в сложной предметной области на примере функционирования международной космической станции и возможное применение таких прикладных онтологий при парировании аварийных и нештатных ситуаций.

Принятые сокращения

МКС – международная космическая станция.

ЦУП – центр управления полётами.

НШС – нештатная ситуация.

СППР – система поддержки принятия решений.

КА – космический аппарат.

КО – конструктор онтологии.

Введение

МКС – сложный, многоцелевой космический исследовательский орбитальный комплекс, управление которым происходит в основном со стороны ЦУП. Так как космос является неблагоприятной средой для человека, то возникновение НШС в опасной для жизни экипажа среде весьма нежелательно, и реагирование на её устранение должно быть немедленным и адекватным. В зависимости от конфигурации МКС (количества и состава пристыкованных КА, количества членов экипажа, находящегося на станции, и мн.др. факторов) действия при парировании НШС и их последовательность могут быть разными.

Основой для разрабатываемых инструментов информационной поддержки принятия решений может являться онтология конкретной предметной области, включающая в себя в рассматриваемом случае МКС со всеми её элементами и системами. Разрабатываемые и совершенствуемые

СППР в российских компаниях (ЗАО «Космос комплект», ОАО «Российские космические системы», ОАО РКК «Энергия», НПК «Разумные решения») позволяют справляться с постоянно растущим объёмом информации, осуществляя работу автономной системы обнаружения и устранения НШС для КА с длительным сроком активного существования [1-4].

Подходы к построению онтологии

Для создания онтологии МКС предлагается использовать КО (в данной работе применен КО, разработанный в НПК «Разумные решения» [1, 2]) и набор сопутствующих инструментов, в рамках которых пользователь может построить модель предметной области и концептуальную модель МКС, описать исходную ситуацию, создать (вручную или автоматически) сценарий действий и исполнить его по шагам, моделируя поведение «назначенных» агентов систем и компонентов для достижения результатов.

В модели КО для построения плана парирования НШС за основу могут быть взяты алгоритмы, описанные в бортовой документации с учетом последних изменений. Регламентированные инструкции к выполнению описываются в онтологии в виде последовательных или параллельных шагов в зависимости задач.

Требуемые к выполнению задачи зависят от текущей ситуации. Поэтому в разрабатываемой онтологии введено понятие состояние. Для описания алгоритмов в модели КО имеется описание различных состояний, которые могут сложиться в системе в результате протекания той или иной НШС. Все задачи обязательно привязаны к определенному перечню состояний. Именно активность этих состояний определяет необходимость выполнения того или иного набора задач. Упрощенная схема описания состояний и задач в онтологии приведена на рисунке 1. Каждое состояние «активизирует» условия возникновения состояния. Также состояние «предшествует» какому-либо действию. Действия, в свою очередь, «требуют» ресурса.

Каждому действию, вопросу, состоянию, условию «срабатывания» состояния прописываются свои атрибуты и проставляются соответствующие связи между ними. Шаблон описания атрибутов объектов в модели представлен на рисунке 2. Количество субъектов зависит от количества экипажа, находящихся в данный момент времени на станции. В выпадающем окне заполняются формы редактирования значения атрибутов, например, «действие» (функция эффективности - ФЭ) и «условие» (см. рисунок 3).

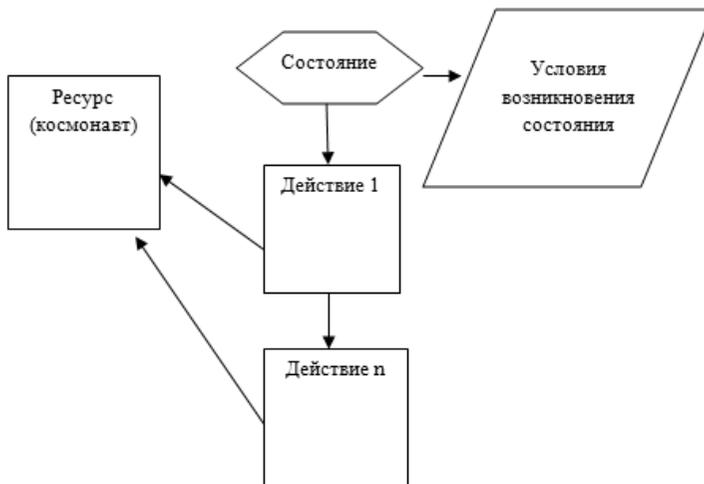


Рис.1. Схема идентификации состояния и задач в онтологии

Название	Значение	Единица измерения
Эффективность	1	б/р
Выполнена		б/р
Описание		б/р
Функция эффективности	ФЭ_ЗначениеЭффективности	б/р
Приоритет		б/р
Функция процентного выполнения	ФЭ_ВсеСубъекты	б/р
Количество субъектов	3	б/р
Функция времени	ФЭ_ФункцияВремениВыполненияЗадачи	б/р
Время выполнения		минуты
Количество объектов		б/р
Штрафная функция		б/р

Рис.2. Шаблон описания атрибутов

Меньше (<Атрибут Объекта Модели = [Отсек][Концентрация аммиака в ppm]>,100)

Каждый обрабатываемый алгоритм проверяется и отлаживается в системе СППР. Вводя условия «срабатывания» состояния в командную строку, алгоритм выбирает необходимую ветку действий парирования НШС и отображает ее в виде действий. Например, пользователь вводит в командную строку параметры ситуации:

- Корабль С – «Используется в текущей конфигурации» - Да;
- Сектор А выброс аммиака – «Тип обнаружения» - Экипаж;

- Сектор А выброс аммиака – «Состояние аммиак»;
- Тип датчика – «Используется в текущей конфигурации» - Да;
После ввода данных СППР выводит на экран пользователю перечень необходимых действий:
- надеть кислородную дыхательную маску;
- срочно покинуть Сектор А;
- закрыть задний люк и т.д.

Выводы

Приведенные примеры разработанных алгоритмов реализуются в мультиагентном аэрокосмическом приложении Smart Tower Control и могут быть использованы при парировании НШС на КА.

Литература

1. *Матюшин М.М., Потоцкий С.И., Скобелев П.О., Потанов В.И., Лахин О.И.* Автоматизированная система поддержки принятия решений в аварийных ситуациях // Программные продукты и системы. – 2013. №3. С. 62–69.
2. *Вакурина Т.Г., Котеля В.В., Лахин О.И., Матюшин М.М., Скобелев П.О.* Онтология российского сегмента Международной космической станции и ее практическое использование в интеллектуальных аэрокосмических приложениях // Труды IV Международной научно-технической конференции «Открытые семантические технологии проектирования интеллектуальных систем» Open Semantic Technologies for Intelligent Systems (OSTIS-2014), 20 – 22 февраля 2014 г., Минск, Республика Беларусь. – 2014. – с 221-226
3. *Димитров Д.М., Данилин Н.С.* Интеллектуальные системы управления на борту отечественных спутников: фантастика или реальность // Труды IV Международной научно-технической конференции «Открытые семантические технологии проектирования интеллектуальных систем» Open Semantic Technologies for Intelligent Systems (OSTIS-2014), 20 – 22 февраля 2014 г., Минск, Республика Беларусь. – 2014. - с. 217-220
4. *Скобелев П.О.* Онтологии деятельности для ситуационного управления предприятием в реальном времени // Онтология проектирования. – 2012. №1(3). С. 26–48

УДК 004.04

Анализ средств организации хранилищ данных в информационно-аналитических системах статистических данных

Валькман Ю. Р., зав. отделом, Жолнарская А. А., аспирантка, Международный научно-учебный центр информационных технологий и систем НАН и МОН Украины, г. Киев

В этой статье исследованы известные средства организации хранилищ данных. Определены их достоинства и недостатки. Обоснованы требования к средствам использования хранилищ данных. Определена специфика статистических данных

Введение

Данная работа содержит результаты исследований, посвященных разработке теоретико-методологических основ создания информационных хранилищ для информационно-аналитических систем.

Объектом исследования является процесс организации хранилищ данных.

Предмет исследования здесь – средства использования хранилищ данных в информационно-аналитических системах.

Цель исследования – анализ средств использования хранилищ данных в информационно-аналитических системах.

Ожидаемые результаты – определение методов и принципов организации хранилищ данных в информационно-аналитических системах статистических данных.

Характеристика статистической информации

Статистическая информация есть совокупность различных сведений, подлежащих фиксации, передаче, преобразованию, хранению и использованию для принятия решений в различных областях. Для статистической информации характерны большие объемы, ограниченные сроки обработки, многократное получение и анализ, абсолютные выражения исходных данных и относительные результаты расчетов, единство расчитываемых показателей для различных сфер деятельности, длительное хранение данных.

Статистические показатели описывают характеристики объекта статистического наблюдения [3].

Статистическая информация может быть организована иерархически, т. е. на каждом уровне данные могут быть укрупнены (агрегированы) на основе некоторых признаков группировки (территориального, функционального).

Любая система статистики призвана организовывать движение потоков статистической информации от мест возникновения к потребителям. В государственной статистической системе источниками первичной информации являются предприятия и организации различных отраслей, а потребителем — Госкомстат Украины, который формирует сводную статистическую отчетность в соответствующих разрезах и предоставляет ее в органы управления различного уровня.

Статистическая обработка данных является одной из сфер, требующих выполнения большого объема сложных и трудоемких математических расчетов, работы с большими массивами исходных данных. К основным предпосылкам применения современных информационных технологий в этой области можно отнести следующие:

- большое количество объектов статистического наблюдения, многомерность данных;
- необходимость отслеживания динамики массива показателей во времени, формирования сводок по различным разрезам;
- трудность получения достоверных данных в доступном для анализа виде;
- высокие материальные и трудовые затраты на проведение статистических работ;
- низкая оперативность обработки данных;
- территориальная разобщенность исходных данных, необходимость их интеграции и одновременной обработки;
- объемность хранимых данных и трудность доступа к ним;
- сложность математических методов анализа данных;
- относительная стабильность форм статистической отчетности.

Статистические информационные системы предназначены для автоматизации работ в области статистики и позволяют собирать, хранить и обрабатывать разнородные массивы данных с использованием единой информационной базы. Структура статистической информационной системы соответствует определенному уровню в организационной структуре органов государственного управления и на этом уровне статистические данные должны обеспечивать интересы потребителей наилучшим образом [3, 8].

С точки зрения состава выполняемых функций, статистическая информационная система обеспечивает комплекс работ по сбору, фиксации динамики изменения и анализу взаимосвязанных показателей, характеризующих конкретный объект статистического наблюдения. Совокупность статистических работ с конкретной группой показателей может составлять основу для выделения функциональной подсистемы статистической информационной системы.

Для организации однократного ввода исходных данных и их многократного использования при решении различных задач, для поддержания полноты и достоверности данных, исключения дублирования информации в основе статистической системы лежит единая информационная база [8].

Технология многоуровневой территориально-распределенной обработки информации с использованием статистических информационных систем включает следующие основные этапы:

- запуск статистической информационной системы, загрузка информационной базы и библиотек;
- при необходимости — корректировка существующих данных;
- выбор форм первичных статистических документов и их визуальный контроль;
- ввод исходной информации с первичных документов в информационную базу и выдача протокола об ошибках;
- контроль и корректировка введенных данных;
- при необходимости — объединение введенных данных с существующим архивом;
- резервное копирование информационной базы;
- формирование промежуточных матриц с показателями (без применения справочников и словарей) и сводных таблиц;
- предварительная печать таблиц;
- контроль таблиц и пересчет показателей в случае корректировки первичных данных;
- повторная печать таблиц;
- подготовка промежуточных числовых матриц и сводных таблиц для передачи в электронном виде на вышестоящий уровень;
- печать контрольных и справочных сведений о передаваемой информации;
- передача комплекта электронных материалов, справок и копий первичных отчетов на вышестоящий уровень;
- оценка справочной информации и контроль таблиц, полученных с каждого из объектов нижнего уровня;

- при необходимости корректировка и повторная передача материалов с нижнего уровня;
- объединение данных по всей совокупности таблиц;
- формирование каталога по территориальным и другим разрезам;
- расчет сводных таблиц по сформированным разрезам;
- контроль сводных таблиц этого уровня;
- печать сводных таблиц по всем разрезам (корректировка и повторная печать — при необходимости);
- формирование и печать сводных таблиц по уровню в целом;
- печать контрольных и справочных сведений о структуре сводной информации;
- передача электронных материалов на федеральный уровень;
- сбор и агрегация данных с нижних уровней, формирование и выдача сводных статистических таблиц;
- передача агрегированных данных на нижние уровни.

Сочетание указанных этапов технологического процесса обработки статистической информации позволяет выбрать наилучший способ решения любой статистической задачи. В одних случаях может выполняться лишь часть перечисленных этапов, а в других цепочки некоторых этапов могут многократно повторяться. Конкретный перечень этапов технологического процесса в каждом случае зависит от разобщенности исходных данных, от потребителя сводных отчетов и необходимой степени агрегации итоговых показателей.

Технологии и концепции организации хранилищ данных в информационно - аналитических системах

Хранилище данных - предметно-ориентированный, интегрированный, неизменяемый, поддерживающий хронологию набор данных, организованный для целей поддержки управления.

Подход построения хранилища данных, для интеграции неоднородных источников данных принципиально отличается от подхода динамической интеграции разнородных БД. Реально строится новое крупномасштабное хранилище, управление данными, в котором происходит по другим правилам, чем в исходных оперативных БД [1].

В основе концепции хранилища данных лежат две основные идеи:

1. Интеграция разьединенных детализированных данных (детализированных в том смысле, что они описывают некоторые конкретные факты, свойства, события и т.д.) в едином хранилище. В процессе интеграции должно выполняться согласование рассогласованных детализиро-

ванных данных и, возможно, их агрегация. Данные могут поступать из исторических архивов, оперативных баз данных, внешних источников.

2. Разделение наборов данных и приложений, используемых для оперативной обработки и применяемых для решения задач анализа.

В свою очередь информационно-аналитические системы рассматривают такие основные вопросы, как:

1. Вопрос накопления данных и их моделирование на концептуальном уровне.
2. Вопрос эффективной выгрузки данных из нескольких независимых источников.
3. Вопрос анализа данных [2].

Интеллектуальный анализ информации в информационно-аналитических системах часто рассматривают, как естественное развитие концепции хранилищ данных, в котором накапливаются все необходимые данные для осуществления определенных задач стратегического управления разных периодов [1].

Ключевыми компонентами такой организации являются технологии OLAP и Data Mining.

Проблемы интеграции данных

Остановимся на некоторых проблемах реализации хранилища данных:

- Неоднородность программной среды
- Распределенный характер организации
- Повышенные требования к безопасности данных
- Необходимость наличия многоуровневых справочников метаданных
- Потребность в эффективном хранении и обработке очень больших объемов информации [2, 5]

Неоднородность программной среды

Хранилище данных практически никогда не создается на пустом месте. Почти всегда конечное решение будет разнородным, т.е. в нем будут использоваться автономно разработанные программные средства. Прежде всего, это касается формирования интегрированного согласованного набора данных, которые могут поступать из разнородных баз данных, электронных архивов, публичных и коммерческих электронных каталогов, справочников, статистических сборников. При построении хранилища данных приходится решать задачу построения единой, согласованно функционирующей информационной системы на основе неоднородных программных средств и решений. При выборе средств реализации хранилища данных приходится учитывать множество факторов, включающих

уровень совместимости различных программных компонентов, легкость их освоения и использования, эффективность функционирования и т.д.

Распределенный характер организации

В концепции хранилища данных предопределено то, что операционная аналитическая обработка может выполняться в любом узле сети независимо от места расположения основного хранилища. Хотя при аналитической обработке данные только читаются, и потребность в синхронизации отсутствует, для достижения эффективности необходимо поддерживать репликацию данных в разных узлах сети [5].

Повышение требований к безопасности данных

Собранная вместе согласованная информация об истории развития корпорации, ее успехах и неудачах, о взаимоотношениях с поставщиками и заказчиками, об истории и состоянии рынка дает возможность анализа прошлой и текущей деятельности корпорации и построения прогнозов для будущего. Эта информация настолько ценна для корпорации, что нельзя допустить возможности ее утечки (на самом деле, если хранилище данных одной корпорации попадет в руки аналитиков другой корпорации, то все аналитические прогнозы первой корпорации сразу станут неверными). В системах, основанных на хранилищах данных, оказывается недостаточной защита данных в стиле языка SQL, которую обеспечивают обычные коммерческие СУБД (этот уровень защиты соответствует классу C2 в соответствии с классификацией Оранжевой Книги Министерства обороны США) [5]. Для обеспечения должного уровня защиты доступ к данным должен контролироваться не только на уровне таблиц и их столбцов, но и на уровне отдельных строк (это уже соответствует классу B1 Оранжевой Книги). Приходится также решать вопросы аутентификации пользователей, защиты данных при их перемещении в хранилище данных из оперативных баз данных и внешних источников, защиты данных при их передаче по сети.

Необходимость наличия многоуровневых справочников метаданных

Если роль метаданных (обычно содержащихся в таблицах-каталогах) в оперативных информационных системах достаточно ограничена, то для OLAP-систем наличие развитых метаданных и средств их предоставления конечным пользователям является одним из основных условий успешной реализации. Например, прежде, чем задать системе свой вопрос, необходимо понимать, какая информация имеется, насколько она актуальна, можно ли ей доверять, сколько времени может занять формирование ответа и т.д. Для пользователя OLAP-системы требуются метаданные, по крайней мере, следующих типов:

1. Описания структур данных, их взаимосвязей.
2. Информация о хранимых на хранилище данных и поддерживаемых им агрегатах данных.
3. Информация об источниках данных и о степени их достоверности. Одна и та же информация могла попасть в хранилище данных из разных источников. Пользователь должен иметь возможность узнать, какой источник был выбран основным, и каким образом производились согласование и очистка данных.
4. Информация о периодичности обновлений данных. Желательно знать не только то, какому моменту времени соответствуют интересующие его данные, но и когда они в следующий раз будут обновлены.
5. Информация о владельцах данных. Пользователю OLAP-системы может оказаться полезной информация о наличии в системе данных, к которым он не имеет доступа, о владельцах этих данных и о действиях, которые он должен предпринять, чтобы получить доступ к данным.
6. Статистические оценки времени выполнения запросов. До выполнения запроса полезно иметь хотя бы приблизительную оценку времени, которое потребуется для получения ответа, и объема этого ответа [3, 5].

Потребность в эффективном хранении и обработке очень больших объемов информации

Уже сейчас известны примеры хранилищ данных, содержащих терабайты информации. По данным консалтинговой компании Meta Group, около половины корпораций, использующих или планирующих использовать хранилища данных, предполагает довести их объем до сотен гигабайт. Проблемой таких больших хранилищ является то, что накладные расходы на внешнюю память возрастают нелинейно при возрастании объема хранилища [5]. Исследования, проведенные на основе тестового набора TPC-D, показали, что для баз данных объемом в 100 гигабайт потребуется внешняя память объемом в 4.87 раза большая, чем нужно собственно для полезных данных. При дальнейшем росте баз данных этот коэффициент увеличивается.

OLAP-системы

Термин «OLAP» неразрывно связан с термином «хранилище данных».

Можно определить OLAP, как совокупность средств многомерного анализа данных, накопленных в хранилище. Такая технология базируется

на построении и визуализации многомерных кубов данных с возможностью произвольного оперирования данными.

В основе OLAP системы лежит многомерное концептуальное представление данных. Такая система, основана на технологии аналитической обработки данных в режиме реального времени (On-Line Analysis Processing) [6].

Методика работы с информацией, организованной с помощью OLAP-хранилищ, призвана обеспечить вывод результатов в требуемой форме, ускорить анализ данных по запросам пользователей. OLAP-технологии позволяют формировать сложные запросы, ответом на которые будут многомерные таблицы, представленные в форме, удобной для восприятия и последующей обработки [6, 10]. Такие выборки принято называть кубами OLAP, на оси которых «наизываются» параметры отбора, а в ячейках содержатся соответствующие данные.

Существует огромное многообразие OLAP-систем. Разработано несколько классификаций продуктов этого типа: например, классификация по способу хранения данных, по месту нахождения OLAP-машины, по степени готовности к применению. Рассмотрим первую из приведенных классификаций.

Существует три способа хранения данных в OLAP-системах или три архитектуры OLAP-серверов:

- MOLAP (Multidimensional OLAP);
- ROLAP (Relational OLAP);
- HOLAP (Hybrid OLAP).

Таким образом, согласно этой классификации OLAP-продукты могут быть представлены тремя классами систем [10].

В случае MOLAP, исходные и многомерные данные хранятся в многомерной БД или в многомерном локальном кубе. Такой способ хранения обеспечивает высокую скорость выполнения OLAP-операций. Но многомерная база в этом случае чаще всего будет избыточной. Куб, построенный на ее основе, будет сильно зависеть от числа измерений. При увеличении количества измерений объем куба будет экспоненциально расти. Иногда это может привести к "взрывному росту" объема данных, парализующему в результате запросы пользователей.

В ROLAP-продуктах исходные данные хранятся в реляционных БД или в плоских локальных таблицах на файл-сервере. Агрегатные данные могут помещаться в служебные таблицы в той же БД. Преобразование данных из реляционной БД в многомерные кубы происходит по запросу OLAP-средства. При этом скорость построения куба будет сильно зависеть от типа источника данных, и поэтому время отклика системы порой становится неприемлемо большим [6].

В случае использования гибридной архитектуры, т.е. в HOLAP-продуктах, исходные данные остаются в реляционной базе, а агрегаты размещаются в многомерной. Построение OLAP-куба выполняется по запросу OLAP-средства на основе реляционных и многомерных данных. Такой подход позволяет избежать взрывного роста данных. При этом можно достичь оптимального времени исполнения клиентских запросов.

Следующая классификация - по месту размещения OLAP-машины. По этому признаку OLAP-продукты делятся на OLAP-серверы и OLAP-клиенты.

В серверных OLAP-средствах вычисления и хранение агрегатных данных выполняются отдельным процессом - сервером. Клиентское приложение получает только результаты запросов к многомерным кубам, которые хранятся на сервере. Многие современные OLAP-серверы поддерживают все три способа хранения данных: MOLAP, ROLAP и HOLAP. OLAP-клиент устроен по-другому. Построение многомерного куба и OLAP-вычисления выполняются в памяти клиентского компьютера.

OLAP - удобный инструмент анализа

В OLAP реализуется сложный интеллектуальный анализ данных. Возможность использования хорошо зарекомендовавших себя методов математической статистики, нейронных сетей, машинного обучения, визуализации данных, индукции правил, нечеткой логики, генетических алгоритмов.

В OLAP-системах основное внимание уделено гибкости доступа и манипулирования информацией [6, 10].

Технология Data Mining

Средство анализа хранилищ данных в информационно-аналитических системах, основанное на методах искусственного интеллекта и инструментах поддержки принятия решений – называется Data Mining. Функционирование Data Mining состоит в выявлении скрытых правил и закономерностей в больших массивах данных.

В отличие от OLAP, в Data Mining основную задачу формулирования гипотез и выявления необычных шаблонов выполняет компьютер [4].

Хранилище данных, поддерживающие технологию Data Mining (метод «добычи данных») и Exploration (метод исследования данных), являются гибридом классических Хранилищ. Такие Хранилища используются для выполнения мощной статистической обработки данных, т.к. являются:

очень детальными, глубоко историческими и оптимизированными для статистического анализа.

Кроме того, для таких Хранилищ характерна ориентация на какой-либо проект. Это означает, что, в отличие от всех других типов ХД, их перестают использовать сразу по завершении анализа, ради которого они создавались.

Методы Data Mining позволяют обнаружить стандартные закономерности. Выделяют пять стандартных типов закономерностей: ассоциация, последовательность, классификация, кластеризация и прогнозирование.

Ассоциация имеет место в том случае, если несколько событий связаны друг с другом [4].

Если существует цепочка связанных во времени событий, то говорят о последовательности.

С помощью классификации выявляются признаки, характеризующие группу, к которой принадлежит тот или иной объект. Это делается посредством анализа уже классифицированных объектов и формулирования некоторого набора правил.

Кластеризация отличается от классификации тем, что сами группы заранее не заданы. С помощью кластеризации средства Data Mining самостоятельно выделяют различные однородные группы данных [9].

Основой для всевозможных систем прогнозирования служит историческая информация, хранящаяся в БД в виде временных рядов. Если удастся построить найти шаблоны, адекватно отражающие динамику поведения целевых показателей, есть вероятность, что с их помощью можно предсказать и поведение системы в будущем.

Классы систем Data Mining

Data Mining является мультидисциплинарной областью, возникшей и развивающейся на базе достижений прикладной статистики, распознавания образов, методов искусственного интеллекта, теории баз данных и др. Отсюда обилие методов и алгоритмов, реализованных в различных действующих системах Data Mining. Многие из таких систем интегрируют в себе сразу несколько подходов. Тем не менее, как правило, в каждой системе имеется какая-то ключевая компонента, на которую делается главная ставка.

Различные методы Data Mining характеризуются определенными свойствами, которые могут быть определяющими при выборе метода анализа данных. Методы можно сравнивать между собой, оценивая характеристики их свойств.

Среди основных свойств и характеристик методов Data Mining рассмотрим следующие: точность, масштабируемость, интерпретируемость, проверяемость, трудоемкость, гибкость, быстрота и популярность.

Масштабируемость - свойство вычислительной системы, которое обеспечивает предсказуемый рост системных характеристик, например, быстроты реакции, общей производительности и пр., при добавлении к ней вычислительных ресурсов.

В таблице приведена сравнительная характеристика некоторых распространенных методов [4, 9].

Оценка каждой из характеристик проведена следующими категориями, в порядке возрастания: чрезвычайно низкая, очень низкая, низкая/нейтральная, нейтральная/низкая, нейтральная, нейтральная/высокая, высокая, очень высокая (таб.1).

Алгоритм	Точность	Масштабируемость	Интерпретируемость	Пригодность к использованию	Трудоемкость	Разносторонность	Быстрота	Популярность, широта использования
классические методы (линейная регрессия)	нейтральная	высокая	высокая / нейтральная	высокая	нейтральная	нейтральная	высокая	низкая
нейронные сети	высокая	низкая	низкая	низкая	нейтральная	низкая	очень низкая	низкая
методы визуализации	высокая	очень низкая	высокая	высокая	очень высокая	низкая	чрезвычайно низкая	высокая / нейтральная
деревья решений	низкая	высокая	высокая	высокая / нейтральная	высокая	высокая	высокая / нейтральная	высокая / нейтральная
полиномиальные нейронные сети	высокая	нейтральная	низкая	высокая / нейтральная	нейтральная / низкая	нейтральная	низкая / нейтральная	нейтральная
k-ближайшего соседа	низкая	очень низкая	высокая / нейтральная	нейтральная	нейтральная / низкая	низкая	высокая	низкая

Таблица 1. Сравнительная характеристика методов Data Mining

Как видно из рассмотренной таблицы, каждый из методов имеет свои сильные и слабые стороны. Но ни один метод, какой бы не была его оценка с точки зрения присущих ему характеристик, не может обеспечить решение всего спектра задач Data Mining.

Большинство инструментов Data Mining, предлагаемых сейчас на рынке программного обеспечения, реализуют сразу несколько методов, например, деревья решений, индукцию правил и визуализацию, или же нейронные сети, самоорганизующиеся карты Кохонена и визуализацию.

В универсальных прикладных статистических пакетах (например, SPSS, SAS, STATGRAPHICS, Statistica) реализуется широкий спектр разнообразнейших методов (как статистических, так и кибернетических). Следует учитывать, что для возможности их использования, а также для интерпретации результатов работы статистических методов (корреляцион-

ного, регрессионного, факторного, дисперсионного анализа и др.) требуются специальные знания в области статистики [4, 8].

Универсальность того или иного инструмента часто накладывает определенные ограничения на его возможности. Преимуществом использования таких универсальных пакетов является возможность относительно легко сравнивать результаты построенных моделей, полученные различными методами. Такая возможность реализована, например, в пакете Statistica, где сравнение основано на так называемой "конкурентной оценке моделей". Эта оценка состоит в применении различных моделей к одному и тому же набору данных и последующем сравнении их характеристик для выбора наилучшей из них [9].

Интеграция OLAP и Data Mining

Интеграция технологий OLAP и Data Mining "обогащает" функциональность и одной, и другой технологии. Эти два вида анализа должны быть тесно объединены, чтобы интегрированная технология могла обеспечивать одновременно многомерный доступ и поиск закономерностей. По словам N. Raden, "многие компании создали ... прекрасные хранилища данных, идеально разложив по полочкам горы неиспользуемой информации, которая сама по себе не обеспечивает ни быстрой, ни достаточно грамотной реакции на рыночные события" [7].

В настоящее время немногие производители реализуют Data Mining для многомерных данных. Кроме того, некоторые методы Data Mining, например, метод ближайших соседей или байесовская классификация, в силу их неспособности работать с агрегированными данными неприменимы к многомерным данным.

Выводы

В статье продемонстрированы технологии OLAP и Data Mining, которые являются ключевыми компонентами организации хранилищ данных. В Data Mining, в отличие от OLAP основную задачу формулирования гипотез и выявление необычных шаблонов выполняет компьютер. Data Mining позволяет выявить новые знания, которые невозможно получить методами статистического, регрессионного анализа или эконометрики. Преимуществом этих технологий в том, что они постоянно развиваются, привлекают к себе все больший интерес со стороны научного мира, так и со стороны применения достижений технологий в различных отраслях.

Литература

1. *Inmon W.* Building the Data Warehouse. – New York: John Wiley & Sons, 1992.
2. *Спирли, Эрик.* Корпоративные хранилища данных. Планирование, разработка, реализация. Том. 1: Пер. с англ. – М.: Издательский дом "Вильямс", 2001.
3. Статистический словарь: М. "Финансы и статистика", 1989.
4. *Дюк В, Самойленко А,* Data mining: учебный курс. – СПб: Питер, 2001.
5. *Ralph Kimball:* The Data Warehouse Toolkit: Practical Techniques for Building Dimensional Data Warehouses. John Wiley 1996.
6. *Codd E.F., Codd S.B., Salley C.T.* "Providing OLAP (On-line Analytical Processing) to User-Analysts: An IT Mandate". Codd & Date, Inc, 1993. Retrieved on 2008.
7. http://www.olap.ru/basic/olap_arch.asp
8. *Айвазян С. А., Бухштабер В. М., Юнжуков И. С., Мешалкин Л. Д.* Прикладная статистика: Классификация и снижение размерности. - М.: Финансы и статистика, 1989.
9. *B. de Ville.* Microsoft Data Mining. Digital Press, 2001.
10. E. F. Codd, S.B.Codd. Providing OLAP. On-line Analytical Processing to User-Analysts: An IT

УДК 004.93

Моделирование зрительного анализатора человека для выделения контуров объектов изображений

Вовк О.Л., к.т.н., доц.,

*Донецкий национальный технический университет, г. Донецк
vovk.olga@gmail.com*

В работе предложен метод для выделения контуров объектов изображений, основанный на моделировании возбуждения нейронов сетчатки человека. Рассматриваются несколько основных слоев: фоторецепторы, горизонтальные, биполярные, амакриновые и ганглиозные клетки. Предлагаемый подход основан на принципах формирования зрительного образа в углублении сетчатки глаза человека.

Введение

Постоянное расширение областей внедрения автоматического распознавания объектов изображений (системы видеонаблюдения, медицинской диагностики, слежения, дефектоскопического контроля, контекстного поиска изображений и др.) требует разработки эффективных методов выделения контуров объектов изображений (как с точки зрения качества, так и - быстродействия).

В рассматриваемой работе предлагается выделить особенности строения сетчатки глаза человека, которые могут использоваться при создании модели зрительного анализатора для распознавания контуров объектов изображений.

Цель работы – разработка метода выделения контуров объектов изображений, основанного на имитации основных слоев сетчатки человека.

Для достижения цели исследования были решены следующие задачи: рассмотрено строение зрительного анализатора человека, выделены особенности отдельных слоев сетчатки, разработан метод распознавания контуров.

Механизм функционирования зрительного анализатора человека

Глаз – орган зрения, весьма сложный орган чувств, воспринимающий действие света [1]. Глаз человека раздражается лучами определенной части спектра (на него действуют электромагнитные волны длиной от 400 до 800 нм).

Глаз состоит из двух основных систем: оптической системы светопреломляющих сред (роговица, водянистая влага передней камеры глаза, хрусталик и стекловидное тело) и рецепторной системы сетчатки [1].

Упрощенно формирование зрительного образа можно записать в виде [2-4]: хрусталик глаза фокусирует свет, падающий на матрицу фоторецепторов в сетчатке, в которой фотоны преобразуются в электрические сигналы, передаваемые по зрительному нерву в мозг.

Стоит отметить, что значительная часть предварительной обработки зрительной информации проводится на уровне сетчатки. Специализированные нейроны, размещенные внутри сетчатки, выделяют контуры объектов, реагируют на движение, передают информацию о свете и тенях [2,4].

Рассмотрим подробнее строение сетчатки зрительного анализатора. Основные понятия и определения приведены в [2,3], остановимся на базовых моментах.

Сетчатка находится на внутренней поверхности глазного яблока и представляет собой сложную сеть фоторецепторов и нервных клеток, которые преобразуют световую энергию в нейронную активность. В основе сетчатки фоторецепторы двух типов: палочки (в основном находятся на периферии, отвечают за информацию без учета цвета) и колбочки (основное расположение – центральная ямка, цветная информация).

До попадания в фоторецепторы свет проходит через сплетение нервных волокон, кровеносных сосудов и эпителиальных клеток. Сетчатка «вывернута наизнанку».

Центральный участок сетчатки с углублением, через который проходит зрительная линия, называется областью фовеа. В этой области от каждой колбочки отходит одно волокно к зрительному нерву (в периферийной части сетчатки одно волокно соединяется с рядом палочек и колбочек). Эта ямка содержит 4-5 тыс. колбочек [2].

Общая схема размещения клеток слоев сетчатки в фовеа приведена на рис.1 [5]. На рис.1 представлена структура центральной ямки сетчатки – области наиболее ясного видения.

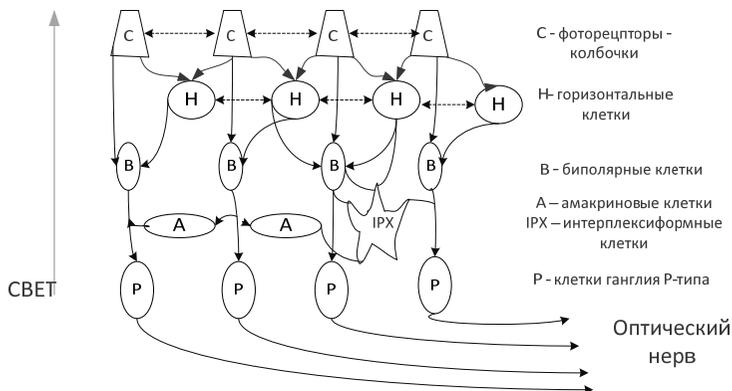


Рис.1. Общая структура центральной части сетчатки

Стоит отметить, что на периферии сетчатки количество биполярных и ганглиозных клеток значительно меньше, чем количество фоторецепторов, т.е. каждая клетка получает сигнал от большого числа палочек и происходит суммирование сигналов (результатирующий сигнал меньше суммы единичных сигналов).

По вертикали фоторецепторы связаны с биполярными клетками, связывающими фоторецепторы и ганглиозные клетки. В зависимости от того какой сигнал получили фоторецепторы – освещения или затемнения, возможно два варианта рецептивных полей вокруг биполярных клеток – с оп-центром или с off-центром (рис. 2).

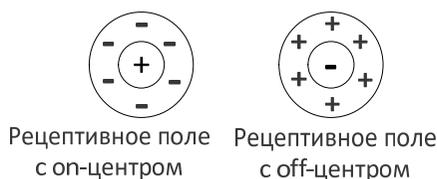


Рис.2. Варианты рецептивного поля биполярной клетки

Ответ биполярных клеток определяется тем – каков их вход: короткий (от одного или нескольких фоторецепторов) или длинный (через горизонтальные клетки) [6-8]. Биполярная клетка получает сигнал от рецептора (рецепторов, в случае периферии сетчатки) и от окружения рецептора через горизонтальные клетки. Горизонтальные клетки (как и амакриновые)

принято считать тормозными нейронами, которые обеспечивают латеральное торможение между биполярными и ганглиозными клетками.

Латеральное торможение – нестандартный тип торможения, который позволяет выделить края тени, если есть перепад освещенности.

Амакриновые клетки – клетки без аксонов, классифицируются по типам ветвления их дендритных деревьев.

Интерплексиформные клетки опосредуют взаимодействие между внешней и внутренней частями сетчатки, они обеспечивают обратную связь, чтобы уменьшить спаривание горизонтальных клеток [8].

Стоит отметить, что в рассмотренной схеме (рис.1) представлены ганглиозные клетки только Р-типа, однако, также существуют ганглиозные клетки М-типа, которые находятся на периферии сетчатки и при возбуждении демонстрируют короткий выброс активности и быстро «затухают». Вероятное назначение этих клеток – идентификация движения [3].

Модель зрительного анализатора для выделения контуров объектов изображений

Пусть исходное изображение I (размером M на N пикселей) задано матрицей характеристик яркости (для упрощения будем использовать модель без учета характеристик цвета). Предполагаем, что каждому пикселю изображения соответствует фоторецептор на центральной ямке сетчатки.

На рис.3 приведена условная схема предлагаемого метода моделирования основных слоев сетчатки для выделения контуров объектов.

Рассмотрим обозначения, введенные на рис.3:

- i_0 – яркость анализируемого пикселя;
- i_j - i_4 – яркость окружения (соседних пикселей);
- b_0 – выходной сигнал биполярной клетки для анализируемого пикселя;
- b_j - b_4 – выходные сигналы биполярной клетки пикселей окружения;
- g_0 – выходной сигнал ганглиозной клетки для анализируемого пикселя;
- eps – параметр метода, подбираемый экспериментально в зависимости от разброса яркостей и энтропии анализируемого изображения.

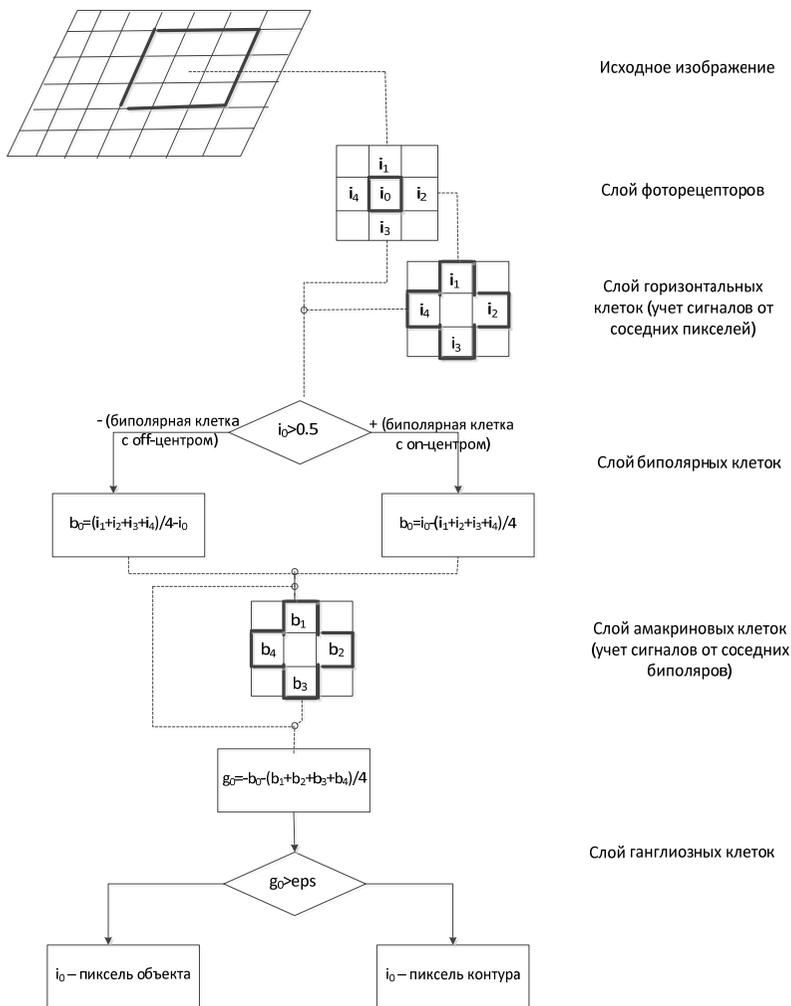


Рис.3. Схема метода выделения контуров объектов изображений

Выводы

Рассмотренный метод выделения контуров объектов изображений основан на модели сетчатки человека, в основе которой возбуждение нейронов на слоях фоторецепторов, горизонтальных, биполярных, амакриновых и ганглиозных клеток. В отличие от методов

пространственной фильтрации Собеля, Превитта и др. предложенный подход учитывает связи между соседними пикселями изображений. Кроме того, рассматривается разделение биполярных клеток на два класса – с off-центром и с off-центром.

В дальнейшем планируется добавить в модель аналог интерплексиформных клеток для минимизации горизонтального спаривания клеток – минимизации шумов контуров объектов изображений.

Результаты исследований могут применяться для выделения заданных областей изображений (к примеру, области автомобильных номеров) и сегментации изображений.

Литература

1. *Гальперин С.И.* Физиология человека и животных. Учебное пособие для университетов и пед.институтов. – М: Высшая школа. – 1977. – 653 с.
2. *Боюн В.П.* Зоровий аналізатор людини як прототип для побудови сімейства проблемно-орієнтованих систем технічного зору // Матеріали Міжнародної науково-технічної конференції «Искусственный интеллект. Интеллектуальные системы. ИИ-2010». – 2010.Т.1. – С. 21-26.
3. *Харви Ричард Шиффман.* Ощущение и восприятие. – СПб.: Питер, 2003. – 928 с.
4. *Верблин Ф., Роска Б.* Кино в наших глазах // В мире науки. – №7, 2007. – с. 28–35.
5. *Samir Shah, Martin D. Levine* . Visual Information Processing in Primate Cone Pathways – Part I: A Model// IEEE Transactions on Systems, Man and Cybernetics – Part B: Cybernetics. – vol.26, no.2, 1996. – P. 259-274.
6. *Golish T., Meister M.* Eye Smarter than Scientists Believed: Neural Computations in Circuits of the Retina //Neuron. – vol.65, IS.2. – 2010. – P.150-164.
7. *Boycott B.B., Dowling J.E.* Organization of the Primate Retina: Light Microscopy // Philosophical Transaction of the Royal Society of London. – Series B, Biological Sciences. – vol. 255, is.799. – 1969. – P.109-184.
8. . Сетчатка //Электронный журнал «Биология и медицина». - [Электронный ресурс] – Режим доступа: http://http://medbiol.ru/medbiol/plus_ner/0008db2b.htm#0004a91e.htm

УДК 681.3: 519.68

Класифікація руху людини з використанням методу швидкого відключення для рекурентних нейронних мереж

Гаврилов С.В., студент, Дробішев Ю.П., д.т.н., проф., Національний технічний університет України «КПІ», м. Київ segii.gavrylov@gmail.com

В роботі розглянуто метод швидкого відключення для рекурентних нейронних мереж та його застосування до задачі класифікації даних зібраних системою 'motion capture'. Отримана нова апроксимація розподілу випадкової величини яка є гіперболічним тангенсом від гаусівської величини. Дана апроксимація зберігає перші два моменти справжнього розподілу, що є достатнім для практичного використання методу. Проведений порівняльний аналіз показав, що запропонований метод має найкращу точність класифікації на розглянутому наборі даних.

Вступ

Задача розпізнавання рухів людини отриманих з систем 'motion capture' [1] є дуже важливою в комп'ютерній анімації, а також є основою для багатьох технологій, таких як альтернативні комп'ютерні інтерфейси, робототехніка та ін. Одним із недоліків сучасних методів розв'язку даної задачі є висока залежність якості отриманої системи від процесу пошуку релевантних характеристик для класифікатора, що потребує значної кількості апріорних знань [2].

Існуючі методи розпізнавання руху можна розділити на дві категорії. До першої відносяться алгоритми які всю послідовність руху представляють одним вектором ознак. Так як рухи різної тривалості відображаються в вектори різної розмірності часто використовують метод головних компонент, для їх приведення до однієї розмірності. Зазвичай єдиного вектора не достатньо для опису внутрішньої та міжкласових варіацій. Другий тип систем декомпозує рух на терми зі словника створеного вручну або на основі кластеризації. Точність цього підходу залежить від кількості класів через потенційно великий розмір словника (дуже розріжене кодування).

Незважаючи на успіх застосування підходу вивчення репрезентації в останні роки до багатьох практичних задач [3], задачі розпізнавання рухів людини приділялось не так багато уваги. Так наприклад існуючі алгоритми [2] основою яких є вивчення високорівневого представлення вхідних

даних, ігнорують часову природу руху людини припускаючи незалежність кадрів. Це потребує додаткової розробки характеристик які враховують еволюцію руху в часі, так як на рівні кадрів деякі класи є дуже схожими. Для подолання подібних обмежень в даній роботі запропоновано модель на основі рекурентної нейронної мережі (РНМ), які зазвичай застосовуються для задач класифікації, де вхідними та вихідними даними є послідовності.

Рекурентні нейронні мережі займають чільне місце серед моделей аналізу багатовимірних часових рядів. Дана модель володіє великою репрезентативною силою, оскільки з довільною точністю може апроксимувати будь-яку вимірну функцію з множини послідовностей [4]. Процес навчання таких моделей є досить складним через ефекти зникання та вибуху градієнту [5]. Сигнал помилки збільшується або зменшується експоненційно для подій рознесених в часі, що робить майже неможливим застосування рекурентної нейронної мережі для більшості задач. Недавні дослідження показали що застосування методів оптимізації другого порядку [6] або ретельно сконструйовані модифікації методу градієнтного спуску Нестерова [7] можуть подолати ці обмеження.

Зважаючи на складність навчання РНМ проблемі перенавчання (англ. overfitting) приділялось мало уваги [8]. Дана проблема часто виникає при малих розмірах тренувальної вибірки. Її можна розв'язати за допомогою нещодавно запропонованого методу регуляризації – швидкого виключення [9]. Даний тип регуляризації передбачає наявність активаційних функцій, які перетворюють розподіл вхідної випадкової величини. Оскільки вихідний розподіл є досить складним, на практиці використовують його апроксимацію. В оригінальній роботі такий вивід зроблений лише для логістичної функції. В даній роботі отримана апроксимація для гіперболічного тангенсу. Це становить інтерес, так як його застосування в якості функції активації покращує остаточну якість моделі [10].

Опис методу

Метод швидкого відключення для рекурентної нейронної мережі

Рекурентна нейронна мережа – це клас штучних нейронних мереж що мають зворотній зв'язок між нейронами. Типова РНМ може бути описана наступними рівняннями:

$$h_t = f_h(u_t, h_{t-1}), \quad (1)$$

$$\check{y}_t = f_o(h_t), \quad (2)$$

Де \check{y}_t передбачене значення в момент часу t , f_h та f_o перехідна та вихідна функції відповідно. Кожна з них параметризується множиною параметрів θ_h, θ_o та вектором початкового прихованого шару h_0 . Маючи навчальну вибірку послідовностей $\mathcal{D} = \{(u_1^{(n)}, y_1^{(n)}), \dots, (u_{T_n}^{(n)}, y_{T_n}^{(n)})\}_{n=1}^N$, можна оцінити параметри РНМ шляхом мінімізації цільової функції:

$$\mathcal{L} = \frac{1}{N} \sum_{n=1}^N \sum_{t=1}^{T_n} e(y_t, \check{y}_t, f_o(h_t^{(n)})), \quad (3)$$

де $e(y, \check{y})$ міра розбіжності між істинним та передбаченим значенням. Зазвичай для задачі класифікації за вихідну функцію обирають функцію софтмаксу $f_o(z)_j = \frac{e^{z_j}}{\sum_{k=1}^K e^{z_k}}$, гіперболічний тангенс як функцію переходу і крос ентропію як функцію розбіжності.

Безпосереднє застосування класичних методів регуляризації (L1, L2 регуляризації) нейронних мереж прямого розповсюдження (НМПР) має негативний ефект для РНМ [8]. Дані підходи можна розглядати як апріорний розподіл для параметрів з єдиною оптимальною точкою. В випадку глобального атрактора в нульовій точці, параметри РНМ будуть зміщені в цьому напрямку, що приведе до неможливості передавання інформації про вхідні дані на великі проміжки в часі шляхом її збереження в прихованому шарі.

За допомогою нещодавно запропонованого методу регуляризації НМПР, що має назву метод відключення, були отримані найкращі результати на класичних наборах даних [11]. Суть методу полягає в випадковому відключенні, викиданні нейронна з мережі, що реалізується шляхом занулення певного вихідного значення. Це приводить до зменшення коадаптації нейронів тобто до менш залежних детекторів ознак. Проте безпосереднє застосування цього методу до РНМ є неможливим так як процедура виключення нейронна еквівалента зануленню всіх синаптичних вагів інцидентних заданому нейрону. Це перетворення має значний вплив на результуючу динаміку РНМ, що призводить до дестабілізації навчання. Метод швидкого відключення [9], що апроксимує значення нейрона нормальною випадковою величиною не тільки прискорює процес навчання але й відкриває можливість його використання в РНМ [8]. Оскільки апроксимація детерміністична, можна припустити що дана процедура не призведе до дестабілізації навчання. Також метод швидкого відключення не створює єдиного атрактора в просторі параметрів, що дозволяє використання конфігурацій системи з нетривіальною динамікою.

При застосуванні методу відсіву до РНМ її можна описати наступними рівняннями:

$$q_t = (d_u \circ u_t)W_{uh} + (d_h \circ h_{t-1})W_{hh} + d_b b_h, \quad (4)$$

$$h_t = \tanh(q_t), \quad (5)$$

$$s_t = (d_{h_y} \circ h_t)W_{h_y} + d_{b_y}b_y, \quad (6)$$

$$\check{y}_t = \text{softmax}(s_t) \quad (7)$$

Формула (4) та (5) визначають пресинаптичне та постсинаптичне значення активації нейронів у прихованому шарі відповідно, формули (6) та (7) – розподіл ймовірності для класів в момент часу t . Функція переходу параметризується ваговою матрицею вхідних даних W_{uh} , рекурентною ваговою матрицею W_{hh} та вектором зміщень b_h . Вихідна функція параметризується матрицею W_{hy} та вектором зміщень b_y . Символ \circ означає по елементне множення векторів, $d_{u_k} \sim \mathcal{B}(p_u)$, $d_{h_k}, d_{b_k} \sim \mathcal{B}(p_h)$, $d_{h_{y_k}}, d_{b_{y_k}} \sim \mathcal{B}(p_{h_y})$ випадкові величини розподілені за законом Бернуллі, реалізації яких і визначають порядок відключення нейронів. В алгоритмі швидкого відсіву [9] замість використання безпосередніх реалізацій цих випадкових величин їх залишають незмінними, натомість q_t апроксимують випадковою величиною $\hat{q}_t \sim \mathcal{N}(E[q_t], V[q_t])$ з нормальним розподілом, що має маточікування $E[q_t] = (E[d_u] \circ E[u_t])W_{uh} + (E[d_h] \circ E[h_{t-1}])W_{hh} + E[d_b]b_h$, та дисперсію $V[q_t] = (V[d_u] \circ E[u_t]^2)W_{uh}^2 + (V[d_h] \circ E[h_{t-1}]^2 + E[d_h] \circ V[h_{t-1}])W_{hh}^2 + V[d_b]b_h^2$. Так як h_t має досить складний ймовірнісний розподіл також використовується апроксимація $\hat{h}_t \sim \mathcal{N}(E[h_t], V[h_t])$ нормальним законом, що зберігає перші два моменти.

Апроксимація для гіперболічного тангенсу

В попередніх роботах [9] масштабована та зміщена логістична функція використовувалась як аналог гіперболічного тангенсу. Проте її похідна в нулі не дорівнює одиниці, саме ця властивість зменшує проблему зникаючого градієнта [10], що покращує результативність моделі в цілому. Тому щоб зберегти цю властивість ми отримали наступні апроксимації:

$$E[\hat{h}_t] = \int_{-\infty}^{\infty} \tanh(x) \mathcal{N}(x|E[q_t], V[q_t]) dx \approx \tanh\left(\frac{E[q_t]}{\sqrt{1 + \frac{\pi V[q_t]}{2}}}\right), \quad (8)$$

$$V[\hat{h}_t] = \int_{-\infty}^{\infty} (\tanh(x) - E[\hat{h}_t])^2 \mathcal{N}(x|E[q_t], V[q_t]) dx \approx \left(1 - \frac{1}{\sqrt{1 + \frac{\pi V[q_t]}{2}}}\right) \left(1 - \tanh^2\left(\frac{E[q_t]}{\sqrt{1 + \frac{\pi V[q_t]}{2}}}\right)\right) \quad (9)$$

Система класифікації руху людини

В даній роботі була використана база даних HDM05 [12]. Оскільки координати залежать від конкретного виконавця та його положення в просторі тож вони є непорівнюваними навіть в межах одного класу. Для уникнення цієї проблеми була проведена попередня обробка даних шляхом нормалізації довжин всіх з'єднань між маркерами, а також зміщення кожного маркеру таким чином щоб особливий маркер 'root' був на початку координат. Деякі класи рухів в оригінальному наборі даних були об'єднанні через їх логічну ідентичність. Наприклад в цій роботі рухи, що відрізняються лише тривалістю виконання, або тим з якої кінцівки починається рух були віднесені до одного класу.

Граденти цільової функції по параметрам моделі були отримані за допомогою методу зворотного поширення помилки в часі [5]. Навчання виконувалось методом rmsprop [13] комбінованим з прискоренням Нестерова [7]. Гіперпараметри були відібрані за допомогою крос валідації. Досягнута точність є порівнюваною з найкращим методом, гібридним багат шаровим перцептроном [2]. Проте потребує значно меншої кількості попередньої обробки даних та пошуку релевантних ознак, оскільки модель РНМ імпліцитно представляє еволюцію руху в часі. В таблиці 1 наведені результати розробленої моделі та попередніх підходів взятих з [2], в дужках зазначене середнє квадратичне відхилення.

Таблиця 1. Оцінка якості існуючих алгоритмів та запропонованого

	extreme learning machine	support vector machine	multilayer perceptron	hybrid multilayer perceptron	fast dropout recurrent neural network
classification accuracy	91.57%(0.88)	94.95%(0.82)	95.20%(1.38)	95.59%(0.76)	96.02%(0.91)

Висновок

Одним із недоліків сучасних методів класифікації руху людини є підбір релевантних характеристик. В даній роботі ця проблема вирішена шляхом використання алгоритму який автоматично вивчає високорівневе представлення даних.

Для подолання проблеми перенавчання був використаний метод регуляризації швидким відключенням. Для використання гіперболічного тангенсу у якості перехідної функції були отримані відповідні апрокси-

мації що зберігають перші два моменти справжнього розподілу. Отримана точність класифікації є кращою в порівнянні з попередніми методами.

Література

1. *Chen, X., & Koskela, M.* (2013). Classification of RGB-D and motion capture sequences using extreme learning machine. In *Image Analysis* (pp. 640-651). Springer Berlin Heidelberg.
2. *Cho, K., & Chen, X.* (2013). Classifying and Visualizing Motion Capture Sequences using Deep Neural Networks. arXiv preprint arXiv:1306.3874.
3. *Bengio, Y.* (2009). Learning deep architectures for AI. *Foundations and trends® in Machine Learning*, 2(1), 1-127.
4. *Schäfer, A. M., & Zimmermann, H. G.* (2006). Recurrent neural networks are universal approximators. In *Artificial Neural Networks–ICANN 2006* (pp. 632-640). Springer Berlin Heidelberg.
5. *Pascanu, R., Mikolov, T., & Bengio, Y.* (2012). On the difficulty of training recurrent neural networks. arXiv preprint arXiv:1211.5063.
6. *Martens, J., & Sutskever, I.* (2011). Learning recurrent neural networks with hessian-free optimization. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)* (pp. 1033-1040).
7. *Sutskever, I., Martens, J., Dahl, G., & Hinton, G.* (2013). On the importance of initialization and momentum in deep learning. In *Proceedings of the 30th International Conference on Machine Learning (ICML-13)* (pp. 1139-1147).
8. *Bayer, J., Osendorfer, C., Chen, N., Urban, S., & van der Smagt, P.* (2013). On Fast Dropout and its Applicability to Recurrent Networks. arXiv preprint arXiv:1311.0701.
9. *Wang, S., & Manning, C.* (2013). Fast dropout training. In *Proceedings of the 30th International Conference on Machine Learning (ICML-13)* (pp. 118-126).
10. *LeCun, Y. A., Bottou, L., Orr, G. B., & Müller, K. R.* (2012). Efficient backprop. In *Neural networks: Tricks of the trade* (pp. 9-48). Springer Berlin Heidelberg.
11. *Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. R.* (2012). Improving neural networks by preventing co-adaptation of feature detectors. arXiv preprint arXiv:1207.0580.
12. *Müller, M., Röder, T., Clausen, M., Eberhardt, B., Krüger, B., & Weber, A.* (2007). Documentation mocap database hdm05.
13. *Graves, A.* (2013). Generating sequences with recurrent neural networks. arXiv preprint arXiv:1308.0850.

УДК 681.325.59

Методика аналізу та оцінки достовірності функціонування логічних мереж на основі суперпозицій булевих функцій

*Дробязко І.П., Клятченко Я.М., Тарасенко-Клятченко О.В., к.т.н., доц., Тарасенко В.П., д.т.н., проф., Тесленко О.К., к.т.н., доц., Національний технічний університет України «КПІ», м. Київ
teslenko@scs.ntu-kpi.kiev.ua*

Проведено аналітичне обґрунтування методики визначення достовірності функціонування логічних мереж, які реалізують багаторозрядні булеві функції шляхом суперпозиції функцій меншої розрядності.

Вступ

Вирішення проблем забезпечення надійного функціонування комп'ютерних систем безпосередньо пов'язане з оцінкою (перевіркою) достовірності функціонування апаратних та програмних засобів. Достовірність функціонування комп'ютерних засобів взагалі – це їх властивість, що визначає безпомилковість виконуваних ними перетворень інформації та характеризується закономірностями появи помилок внаслідок випадкових збоїв. Поки що не існує загальної математико-модельної основи, яка дозволяла б забезпечити аналіз достовірності функціонування будь-яких комп'ютерних засобів. Тому задачі визначення достовірності функціонування програмних і апаратних засобів доцільно вирішувати для окремих їх класів.

В комп'ютерних системах досить широкий клас їх апаратних засобів складають логічні мережі (ЛМ), практичне застосування яких значно розширилось завдяки розвитку технології ПЛІС, зокрема ПЛІС FPGA. Висока надійність сучасних інтегральних середовищ дає підстави для більш глибокого дослідження методів визначення достовірності функціонування за умови наявності детермінованих (тобто таких, які приймають значення з наперед визначеної множини можливих значень) спотворень вхідних даних. Кількісною оцінкою достовірності служить ймовірність отримання правильного результату (ймовірність правильної роботи) або ймовірність помилки. Якщо вважати, що апаратні засоби не схильні до відмов та збоїв, то традиційно ймовірність помилки прирівнюють до ймовірності спотворення вхідних даних. стисло викладається

суть роботи.

В той же час в роботах [1-4] досліджувалися латентні можливості булевих функцій по виправленню детермінованих спотворень вхідних даних – явище автокорекції. В [5] показані можливості суттєвого підвищення оцінок достовірності роботи логічних функціональних перетворювачів інформації в умовах дії детермінованих вхідних спотворень завдяки автокоригуючим властивостям булевих функцій. Визначення абсолютних та відносних оцінок автокоригуючих властивостей має, згідно з [5], трудомісткість, яка пропорційна експоненті від кількості змінних булевої функції. В [6] запропонована методика обчислень достовірності з врахуванням реальної реалізації ЛМ, яка була апробована на конкретних прикладах. Задачею даної роботи є теоретичне обґрунтування вказаної методики.

Аналіз попередніх результатів та уточнення постановки задачі

Нехай довільна булева функція $f(x_1, x_2, \dots, x_n)$ реалізована тими чи іншими апаратними засобами (наприклад, ПЛІС), де x_i – розряди n -розрядного вхідного набору (далі – операнда). Тоді будемо вважати, що $p_i^0 = p(x_i=0)$, $p_i^1 = p(x_i=1)$ – ймовірності значень $x_i=0$ та $x_i=1$. Крім того, допускаємо, що в загальному випадку на i -й розряд операнда діють b детермінованих спотворень. Тоді $g_{0i} + g_{1i} + \dots + g_{bi} = 1$, де g_{0i} – ймовірність відсутності будь-яких спотворень, g_{li} – ймовірність спотворення типу l ($l=1, 2, \dots, b$). Очевидно, що

$$(g_{0i} + g_{1i} + \dots + g_{bi})(p_i^0 + p_i^1) = 1, \quad (1)$$

Детермінованість спотворень дозволяє подати вираз (1) в наступному вигляді

$$g_{0i} + g_{1i} + \dots + g_{bi} = (g_{0i} p_i^0 + g_{c1}^0 p_i^0 + g_{e1}^0 p_i^1) + (g_{0i} p_i^1 + g_{c1}^1 p_i^1 + g_{e1}^1 p_i^0) = p_{gi}^0 + p_{gi}^1, \quad (2)$$

де g_{c1}^0 – сума ймовірностей детермінованих спотворень, при яких нульове значення x_i не змінюється (вхідна автокорекція); g_{e1}^0 – сума ймовірностей детермінованих спотворень, при яких одиничне значення x_i змінюється на нульове; g_{c1}^1 – сума ймовірностей детермінованих спотворень, при яких одиничне значення x_i не змінюється (вхідна автокорекція); g_{e1}^1 – сума ймовірностей детермінованих спотворень, при яких нульове значення x_i змінюється на одиничне; p_{gi}^0, p_{gi}^1 – ймовірності значень 0 та 1 на вході ЛМ в результаті спотворень.

Із (2) випливає, що кожна з ймовірностей p_{gi}^0 та p_{gi}^1 складається із 3 компонент: ймовірності при відсутності спотворень, ймовірності вхідної

автокорекції та ймовірності вхідного хибного значення. Визначимо множину $M(f)$ кортежів (впорядкованих наборів значень змінних) $A_j = \langle a_{1j}, a_{2j}, \dots, a_{nj} \rangle$. Позначимо як $M^0(f)$ множину кортежів значень змінних функції f , де вона приймає значення 0 , і як $M^1(f)$ – множину кортежів значень змінних функції f , де вона приймає значення 1 .

Розглянемо довільний кортеж $A_j = \langle a_{1j}, a_{2j}, \dots, a_{nj} \rangle \in M^0(f)$, $j \in \{0, 1, \dots, 2^n - 1\}$, який одержано в результаті дії спотворень. Для функції $f(x_1, x_2, \dots, x_n)$ згідно з [6], маємо

$$P^0(A_j) = \prod_{i=1}^n (g_{0i} p_i^{a_{ij}} + g_{ci} p_i^{a_{ij}} + g_{ei} p_i^{not a_{ij}}) \quad (3)$$

Після розкриття дужок в (3) добуток із n співмножників перетвориться в суму із 3^n добутоків вигляду

$$g_{r_1}^{\alpha_1} p_1^{\alpha_1} g_{r_2}^{\alpha_2} p_2^{\alpha_2} \dots g_{r_n}^{\alpha_n} p_n^{\alpha_n} \quad (4)$$

Тут буква r в індексах може приймати значення 0 , c або e , а буква α_i – значення i -го розряду кортежу, якщо використовується $r=0$ чи $r=c$, або його інверсії, якщо використовується $r=e$. Кожен з добутоків (4) є значенням ймовірності перетворення відповідного кортежу в кортеж A_j .

Згідно з [6] сукупність із 3^n добутоків можна розділити на наступні класи:

- клас $K_{j0}(A_j)$, який містить лише один добуток, коли спотворення відсутні ($r=0$ для всіх елементів добутку);
- клас $K_{jc}(A_j)$, який містить добутки, де хибні значення відсутні ($r \neq e$ для всіх елементів добутку), а вхідна автокорекція ($r=c$) відбувається принаймні в одному розряді;
- клас $K_{je}(A_j)$, який містить добутки, де принаймні в одному розряді внаслідок спотворень одержано хибне вхідне значення ($r=e$).

Кількість добутоків класу $K_{jc}(A_j)$ дорівнює $2^n - 1$, а кількість добутоків класу $K_{je}(A_j)$ дорівнює $3^n - 2^n$.

Нехай $G_{j0}^0(A_j)$ - ймовірність нульового значення булевої функції на кортежі A_j за відсутності спотворень. Очевидно, що значення G_{j0}^0 дорівнює значенню елемента класу $K_{j0}(A_j)$. Позначимо далі як $G_{jc}^0(A_j)$ ймовірність нульового значення булевої функції на кортежі A_j за наявності вхідних автокорекцій. Очевидно також, що значення $G_{jc}^0(A_j)$ дорівнює сумі значень елементів класу $K_{jc}(A_j)$. Елементи класу $K_{je}(A_j)$ можна розділити на два підкласи, а саме: підклас $K_{jec}(A_j)$, де містяться елементи вихідної автокорекції; підклас $K_{jee}(A_j)$, де містяться елементи хибних значень. Формування вказаних підкласів полягає в наступному. Для будь якого елемента класу $K_{je}(A_j)$ утворюємо кортеж \tilde{A}_j шляхом інверсії значень розря-

дів, де $r=e$. Якщо $\tilde{A}_j \in M^0(f)$, то даний елемент класу $K_{je}(A_j)$ віднесемо до підкласу $K_{jec}(A_j)$, якщо ж $\tilde{A}_j \in M^1(f)$ – то до підкласу $K_{jee}(A_j)$. Суму значень елементів підкласу $K_{jec}(A_j)$ позначимо як $G_{jec}^0(A_j)$. Це значення буде відповідати ймовірності вихідної автокорекції булевої функції $f(x_1, x_2, \dots, x_n)$ при спотворенні відповідних кортежів в кортеж A_j . Суму значень елементів підкласу $K_{jee}(A_j)$ позначимо як $G_{jee}^0(A_j)$, що відповідає ймовірності хибних значень функції при спотворенні відповідних кортежів в кортеж A_j . Позначимо як $G_{jcc}^0(A_j) = G_{jc}^0(A_j) + G_{jec}^0(A_j)$ загальну ймовірність автокорекції за наявності детермінованих спотворень. Таким чином, маємо

$$P^0(A_j) = G_{j0}^0(A_j) + G_{jcc}^0(A_j) + G_{jee}^0(A_j)$$

У відповідності з [6] можна записати

$$\sum_{\substack{\text{по всіх кортежах,} \\ \text{де } f=0}} P^0(A_j) = G_0^0(f) + G_c^0(f) + G(f)$$

де

$$G_0^0(f) = \sum_{\substack{\text{по всіх кортежах,} \\ \text{де } f=0}} G_{j0}^0(A_j), \quad G_c^0(f) = \sum_{\substack{\text{по всіх кортежах,} \\ \text{де } f=0}} G_{jec}^0(A_j), \quad (5)$$

$$G_e^0(f) = \sum_{\substack{\text{по всіх кортежах,} \\ \text{де } f=0}} G_{jee}^0(A_j)$$

Аналогічно можна обчислити значення $G^1_o(f)$ – ймовірності одиничного значення булевої функції за відсутності спотворень, $G^1_c(f)$ – ймовірності правильного одиничного значення за наявності спотворень (повна автокорекція) та $G^1_e(f)$ – ймовірності хибного одиничного значення за наявності спотворень.

Складність обчислення вказаних ймовірностей пропорційна експоненті від кількості змінних, що суттєво ускладнює їх практичне застосування. Але враховуючи, що, по-перше в ЛМ багаторозрядні булеві функції реалізують шляхом суперпозиції функцій меншої розрядності, та, по-друге, система ймовірностей (5) для значень булевої функції аналогічна системі ймовірностей її аргументів, то це дає підстави проводити обчислення у відповідності до конкретної реалізації логічної мережі. В [6] на конкретних прикладах було показано, що значення ймовірностей, обчислене без врахування суперпозицій та з їх врахуванням, співпадають між собою. Доведемо це аналітичним шляхом.

Основна частина

Нехай булева функція подана наступним чином.

$$f(x_1, x_2, \dots, x_n) = f_1(x_1, x_2, \dots, x_t, f_2(x_{t+1}, x_{t+2}, \dots, x_n)) \quad (6)$$

Розглянемо функцію $f_1(x_1, x_2, \dots, x_t, u)$. Позначимо як $M^0(f_1)$ множини кортежів значень аргументів функції f_1 , де вона приймає значення 0, як $M^1(f_1)$ множини кортежів значень аргументів функції f_1 , де вона приймає значення 1, як $M^0(f_1|u=0)$ множини кортежів значень змінних x_1, x_2, \dots, x_t за умови, що $u=0$. Очевидно, що всі кортежі множини $M^0(f_1|u=0)$ попарно різні. Аналогічно утворимо множини $M^0(f_1|u=1)$, $M^1(f_1|u=0)$ та $M^1(f_1|u=1)$. Далі позначимо як $M^0(f_2)$ множини кортежів значень змінних функції f_2 , де вона приймає значення 0, а як $M^1(f_2)$ множини кортежів значень змінних функції f_2 , де вона приймає значення 1. Очевидно, що

$$M^0(f) = M^0(f_1|u=0) \times M^0(f_2) \cup M^0(f_1|u=1) \times M^1(f_2), \quad (7)$$

$$M^1(f) = M^1(f_1|u=0) \times M^0(f_2) \cup M^1(f_1|u=1) \times M^1(f_2), \quad (8)$$

де \times – позначення декартового добутку, а як операцію множення виконують операцію конкатенації кортежів. Позначимо як $M_{t+1,n}$ множини кортежів значень змінних $x_{t+1}, x_{t+2}, \dots, x_n$. Відношення рівності дочірніх функцій $f(x_1, x_2, \dots, x_t, a_{(t+1)}, a_{(t+2)}, \dots, a_n)$, де $\langle a_{(t+1)}, a_{(t+2)}, \dots, a_n \rangle$ – довільний кортеж значень змінних $x_{t+1}, x_{t+2}, \dots, x_n$, розбиває згідно з (6) множини $M_{t+1,n}$ на два класи еквівалентності $M^1_{t+1,n}$ та $M^2_{t+1,n}$. Це означає, що серед всіх дочірніх функцій $f(x_1, x_2, \dots, x_t, a_{(t+1)}, a_{(t+2)}, \dots, a_n)$ існують лише дві різні, тобто для будь-якого кортежу $\langle a_{(t+1)}, a_{(t+2)}, \dots, a_n \rangle \in M^1_{t+1,n}$.

$$f(x_1, x_2, \dots, x_t, a_{(t+1)}, a_{(t+2)}, \dots, a_n) = f^1(x_1, x_2, \dots, x_t),$$

а для будь-якого кортежу $\langle a_{(t+1)}, a_{(t+2)}, \dots, a_n \rangle \in M^2_{t+1,n}$

$$f(x_1, x_2, \dots, x_t, a_{(t+1)}, a_{(t+2)}, \dots, a_n) = f^2(x_1, x_2, \dots, x_t)$$

Для функції f_2 , згідно з попереднім, обчислимо значення ймовірностей $G^0_o(f_2)$, $G^0_c(f_2)$, $G^0_e(f_2)$, $G^1_o(f_2)$, $G^1_c(f_2)$, $G^1_e(f_2)$.

Позначимо як $C_{h(u=0)} = \langle a_{1h}, a_{2hj}, \dots, a_{th}, 0 \rangle$ довільний кортеж значень змінних функції f_1 ($h=0, 1, \dots, 2^t-1$) при значенні змінної $u=0$. Припустимо, що $f_1(C_{h(u=0)})=0$. Для такої функції маємо

$$P^0(C_{h(u=0)}) = \prod_{i=1}^t (g_{0i} p_i^{a_{ij}} + g_{ci}^{a_{ij}} p_i^{a_{ij}} + g_{ei}^{a_{ij}} p_i^{not\ a_{ij}}) \times$$

$$\times (G^0_o(f_2) + G^0_c(f_2) + G^1_e(f_2))$$

Згідно з попереднім значення $P^0(C_{h(u=0)})$ подамо наступним чином

$$P^0(C_{h(u=0)}) = (L^0_{h0(u=0)} + L^0_{hc(u=0)} + L^0_{hce(u=0)} + L^0_{hee(u=0)})$$

Обчислимо

$$G_0^0(f_{1(u=0)}) = \sum_{\substack{\text{по всіх кортежах} \\ C_h(u=0), \text{ де } f_1=0}} L_{h0(u=0)}^0, \quad G_c^0(f_{1(u=0)}) = \sum_{\substack{\text{по всіх кортежах} \\ C_h(u=0), \text{ де } f_1=0}} L_{hcc(u=0)}^0,$$

$$G_e^0(f_{(u=0)1}) = \sum_{\substack{\text{по всіх кортежах} \\ C_h(u=0), \text{ де } f_1=0}} L_{hee(u=0)}^0$$

Позначимо як $C_{h(u=0)} = \langle a_{1h}, a_{2hj}, \dots, a_{th}, l \rangle$ довільний кортеж значень змінних функції f_l ($h=0, 1, \dots, 2^l-1$) при значенні змінної $u=1$. Далі аналогічно вищевикладеному визначимо $P^0(C_{h(u=1)})$.

$$P^0(C_{h(u=1)}) = \prod_{i=1}^l (g_{0i} p_i^{a_{ij}} + g_{ci} p_i^{a_{ij}} + g_{ei} p_i^{not a_{ij}}) (G_0^1(f_2) + G_c^1(f_2) + G_e^0(f_2))$$

або

$$P^0(C_{h(u=1)}) = (L_{h0(u=1)}^0 + L_{hc(u=1)}^0 + L_{hce(u=1)}^0 + L_{hee(u=1)}^0)$$

Обчислимо

$$G_0^0(f_{1(u=1)}) = \sum_{\substack{\text{по всіх кортежах} \\ C_h(u=1), \text{ де } f_1=0}} L_{h0(u=1)}^0, \quad G_c^0(f_1) = \sum_{\substack{\text{по всіх кортежах} \\ C_h(u=1), \text{ де } f_1=0}} L_{hcc(u=1)}^0,$$

$$G_e^0(f_1) = \sum_{\substack{\text{по всіх кортежах} \\ C_h(u=1), \text{ де } f_1=0}} L_{hee(u=1)}^0$$

Розглянемо функцію $f_l(x_1, x_2, \dots, x_h, u)$. Позначимо як $C_h = \langle a_{1h}, a_{2hj}, \dots, a_{th}, a_{uh} \rangle$ довільний кортеж значень змінних функції f_l ($h=0, 1, \dots, 2^{l+1}-1$). Припустимо, що $f_l(C_h)=0$. Для цієї функції маємо

$$P^0(C_h) = \prod_{i=1}^l (g_{0i} p_i^{a_{ij}} + g_{ci} p_i^{a_{ij}} + g_{ei} p_i^{not a_{ij}}) \times \\ \times (G_0^{a_{uh}}(f_2) + G_c^{a_{uh}}(f_2) + G_e^{not a_{uh}}(f_2))$$

Відповідно до вищевикладеного подамо $P^0(C_h)$ наступним чином

$$P^0(C_h) = (L_{h0}^0 + L_{hc}^0 + L_{hce}^0 + L_{hee}^0)$$

Обчислимо

$$G_0^0(f_1) = \sum_{\substack{\text{по всіх кортежах} \\ C_h, \text{ де } f_1=0}} L_{h0}^0, \quad G_c^0(f_1) = \sum_{\substack{\text{по всіх кортежах} \\ C_h, \text{ де } f_1=0}} L_{hcc}^0,$$

$$G_e^0(f_1) = \sum_{\substack{\text{по всіх кортежах} \\ C_h, \text{ де } f_1=0}} L_{hee}^0$$

Очевидно, що

$$G_0^0(f_1) = G_0^0(f_{1(u=0)}) + G_0^0(f_{1(u=1)}),$$

$$G^0_c(f_i) = G^0_c(f_{i(u=0)}) + G^0_c(f_{i(u=1)}),$$

$$G^0_e(f_i) = G^0_e(f_{i(u=0)}) + G^0_e(f_{i(u=1)}).$$

Аналогічно для кортежів, де функція f_j приймає одиничне значення

$$G^1_0(f_i) = G^1_0(f_{i(u=0)}) + G^1_0(f_{i(u=1)}),$$

$$G^1_c(f_i) = G^1_c(f_{i(u=0)}) + G^1_c(f_{i(u=1)}),$$

$$G^1_e(f_i) = G^1_e(f_{i(u=0)}) + G^1_e(f_{i(u=1)}).$$

Твердження. Якщо функція f подана у формі (6), то

$$G^0_0(f) = G^0_0(f_i), \quad G^0_c(f) = G^0_c(f_i), \quad G^0_e(f) = G^0_e(f_i),$$

$$G^1_0(f) = G^1_0(f_i), \quad G^1_c(f) = G^1_c(f_i), \quad G^1_e(f) = G^1_e(f_i).$$

Доведення. Розглянемо довільний кортеж $A_j = \langle a_{1j}, a_{2j}, \dots, a_{nj} \rangle \in M^0(f)$, $j \in \{0, 1, \dots, 2^n - 1\}$, який одержано в результаті дії детермінованих спотворень. Подамо, далі, кортеж A_j як конкатенацію двох кортежів $A_j = C_j || S_j$, де $C_j = \langle a_{1j}, a_{2j}, \dots, a_{ij} \rangle$, $S_j = \langle a_{(i+1)j}, a_{(i+2)j}, \dots, a_{nj} \rangle$.

Згідно з (3) та (4) та у відповідності з дистрибутивним законом ймовірність $P(A_j)$ можна подати наступним чином

$$P(A_j) = \left(\sum_{s=1}^{3^i} g_{r_s 1}^{\alpha_1} g_{r_s 2}^{\alpha_2} \dots g_{r_s i}^{\alpha_i} \right) \left(\sum_{s=1}^{3^{n-i+1}} g_{r_s (i+1)}^{\alpha_{i+1}} \dots g_{r_s n}^{\alpha_n} \right)$$

Тут, як і раніше, буквою r_s позначено θ , c або e , а буквою α_i – значення i -го розряду кортежу A_j , якщо використовується $r_s = \theta$ чи $r_s = c$, або його інверсія, якщо використовується $r_s = e$.

Сукупність добутків в другій сумі відповідає всім можливим спотворенням, які перетворюють кортежі значень змінних $x_{i+1}, x_{i+2}, \dots, x_n$ в кортеж S_j , при цьому існує функція f_2 , визначена на цих кортежах, яка дозволяє визначати елементи підкласів $K_{jcc}(S_j) + K_{jee}(S_j)$. Тому, якщо $f_2(S_j) = \theta$, то

$$P(A_j) = (G_{j0}^0(S_j) + G_{jcc}^0(S_j) + G_{jee}^1(S_j)) \left(\sum_{s=1}^{3^i} g_{r_s 1}^{\alpha_1} g_{r_s 2}^{\alpha_2} \dots g_{r_s i}^{\alpha_i} \right)$$

або якщо $f_2(S_j) = I$, то

$$P(A_j) = (G_{j0}^1(S_j) + G_{jcc}^1(S_j) + G_e^0(S_j)) \left(\sum_{s=1}^{3^i} g_{r_s 1}^{\alpha_1} g_{r_s 2}^{\alpha_2} \dots g_{r_s i}^{\alpha_i} \right)$$

Розглянемо суму значень $P(A_j)$ при всіх $A_j = \langle a_{1j}, a_{2j}, \dots, a_{nj} \rangle \in M^0(f)$ та $f_2(S_j) = \theta$.

$$\sum_{\substack{\text{по всіх } A_j, \text{ де } f(A_j) = \theta, \\ \text{та } f_2(S_j) = \theta}} P(A_j) = \sum_{\substack{\text{по всіх } A_j, \text{ де } f(A_j) = \theta, \\ \text{та } f_2(S_j) = \theta}} (G_{j0}^0(S_j) + G_{jcc}^0(f_2) + G_e^1(f_2)) \times \left(\sum_{s=1}^{3^i} g_{r_s 1}^{\alpha_1} g_{r_s 2}^{\alpha_2} \dots g_{r_s i}^{\alpha_i} \right) \quad (9)$$

Згідно (7), для функції f має місце рівність $M^0(f) = M^0(f) \setminus u=0 \times M^0(f_2)$. Тому (9) можна подати наступним чином

$$\sum_{\substack{\text{по всіх } A_j, \text{ де } f(A_j)=0, \\ \text{та } f_2(S_j)=0}} P(A_j) = \sum_{\text{по всіх } S_j, \text{ де } f_2(S_j)=0} (G_{f_0}^0(S_j) + G_{f_{cc}}^0(f_2) + G_e^1(f_2)) \times \\ \times \left(\sum_{\substack{\text{по всіх } A_j, \\ \text{де } f_1(C_j)=0}} \sum_{s=1}^{3^l} g_{r_s^1}^{\alpha_1} g_{r_s^2}^{\alpha_2} \dots g_{r_s^l}^{\alpha_l} \right)$$

або

$$\sum_{\substack{\text{по всіх } A_j, \text{ де } f(A_j)=0, \\ \text{та } f_2(S_j)=0}} P(A_j) = (G_0^0(f_2) + G_c^{01}(f_2) + G_e^1(f_2)) \left(\sum_{\substack{\text{по всіх } A_j, \\ \text{де } f_1(C_j)=0}} \sum_{s=1}^{3^l} g_{r_s^1}^{\alpha_1} g_{r_s^2}^{\alpha_2} \dots g_{r_s^l}^{\alpha_l} \right)$$

Отже

$$\sum_{\substack{\text{по всіх } A_j, \text{ де } f(A_j)=0, \\ \text{та } f_2(S_j)=0}} P(A_j) = G_0^0(f_{1(u=0)}) + G_c^0(f_{1(u=0)}) + G_e^0(f_{(u=0)1})$$

Аналогічно для випадків, коли $f_2(S_j)=1$, та $A_j \in M^l(f)$. Доведення закінчено.

Висновок

Аналітично доведена тотожність результатів обчислення ймовірностей функціонування ЛМ за наявності детермінованих спотворень вхідних даних без врахування суперпозицій булевих функцій та з врахуванням суперпозицій. Одержані результати легко узагальнюються на випадок подання функції f у вигляді

$$f(x_1, x_2, \dots, x_n) = f_1(x_1, x_2, \dots, x_t) f_2(x_{t+1}, x_{t+2}, \dots, x_n), \dots, f_{2r}(x_{t+1}, x_{t+2}, \dots, x_n)$$

та використання багаторівневих суперпозицій.

Напрямки подальших досліджень полягають в визначенні спільних параметрів для оцінки достовірності функціонування ЛМ в цілому.

Література

1. *Тарасенко В.П., Тарасенко-Клятченко О.В.* Метод оценки автокорректирующих свойств поразрядных логических операций // Радиоэлектроника и информатика. – 2001. – № 1 (14). – с. 83-86.
2. *Михайлюк А.Ю., Тарасенко-Клятченко О.В.* Автокоригуючі властивості логічних операцій // Вимірювальна та обчислювальна техніка в технологічних процесах. – 2000. – № 7. с. 196-198.
3. *Тарасенко-Клятченко О.В.* Сравнительный анализ корректирующих свойств переключательных функций // Правове, нормативне та метрологічне забезпечення системи захисту інформації в Україні. – 2002. – № 5. – с. 189-194.
4. *Tarasenko-Klyatchenko O.V.* The Comparative Analysis of Correcting Properties of Switching Functions // Proceedings of the 7-th International Conference “The experience of designing and application of CAD Systems in Microelectronics”. – 2003. – p. 232-234.
5. *Тарасенко-Клятченко О.В.* Автокоригуючі властивості та достовірність роботи логічних функціональних перетворювачів інформації. // Дисертація на здобуття наукового ступеня кандидата технічних наук – Інститут проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України. – Київ – 2004.
6. *Я.М.Клятченко, О.В. Тарасенко-Клятченко, В.П.Тарасенко, О.К.Тесленко.* Достовірність функціонування логічних мереж в умовах дії вхідних спотворень. // Комп'ютерно-інтегровані технології: освіта, наука, виробництво. – 2012 – №8 – с. 47-52.

UDC 004.023

An artificial bee colony algorithm for data clustering

*Yu.Zorin, Ph.D, Assoc. Prof., National Technical University of Ukraine,
Kyiv, yzorin@gmail.com*

The paper presents a new metaheuristic algorithm for data clustering based on Artificial Bee Colony optimization. Data clustering is one of the most challenging optimization problems that belongs to NP-hard class. For this reason a heuristic approach has to be applied to get at least a quasi-optimal solution. As the results of computational experiment have shown, due to the introduction of two fundamental distinctions a proposed algorithm outperforms basic Bee Algorithm as well as k-means algorithm.

INTRODUCTION

Clustering is a typical methodology of unsupervised machine learning. Clustering algorithm groups together data objects in accordance with the information found in the objects themselves. The goal of clustering is creation of groups inside which the objects are more similar among themselves and different from the objects in other groups. The bigger similarity of objects inside the groups and the bigger dissimilarity between groups, the higher quality of clustering result is. To date there are dozens of clustering algorithms and their modifications [1].

The mathematical definition of clustering is as follows. Let $X \in R^{m \times n}$ is a set of data items representing a set of m points x_i in R^n . The goal is to partition X into K groups C_K such that the points that belong to the same group are more "alike" than points in other groups. Each of the K groups is called a cluster. The result of the algorithm is an injective mapping $X \rightarrow C$ points x_i in C_K .

As a similarity measure of points that belong to the same cluster often is chosen a p-norm, defined as

$$\|X\|_p = \left(\sum_{i=1}^n |x_i|^p \right)^{1/p}$$

When $p = 1$ we obtain the so-called Manhattan norm, for $p = 2$, we have the Euclidean norm and for p approaching to infinity, p -norm is close to the maximum norm. In the case of the Euclidean norm clustering result is evaluated using the sum of squared errors (SSE), which is defined as

$$SSE = \sum_{i=1}^K \sum_{x \in C_i} dist(x, c_i)^2,$$

where C_i is i -th cluster, c_i - its centroid, $dist(x, c_i)$ - the Euclidean norm from the point x to c_i . The cluster centroid is defined as

$$c_i = \frac{1}{m_i} \sum_{x \in C_i} x,$$

where m_i is the number of points in cluster C_i .

There are hierarchical and partitional clustering algorithms. Among the latter, the most often used is the k-means algorithm. The gist of k-means algorithm is as follows.

Randomly choose K points in X as initial centroids.

Do

- a) form K clusters by assigning each points to the nearest centroid;
- b) recalculate centroids for each cluster till centroids stop moving.

The biggest drawback of the k-means algorithm is that its results are significantly different after each run. This is mainly due to the fact that the initial centroids are chosen randomly. Furthermore k-means algorithm has a peculiar effect of "splitting" the large clusters. Somewhat better results can be obtained using the algorithm called k-means ++ [2].

The clustering problem itself is known to be NP-hard, and thus the common approach is to search only for quasi-optimal solutions. This can be done with the use of heuristic algorithms, in particular the so-called swarm algorithms

[3-5]. One of these algorithms is the Artificial Bee Colony algorithm (ABC-algorithm). This is one of the latest multi-agent optimization algorithms inspired by the behavior of bees in nature [6].

THE MULTI-AGENT ALGORITHM FOR DATA CLUSTERING

The goal of the ABC-algorithm is to find the optimal (quasi-optimal) solution to some optimization problem. That is, it comes down to finding the extremum (minimum or maximum) of the objective function $f(x)$. The colony of artificial bees consists of three varieties: employed bees, onlooker bees and scouts. Every bee represents a vector containing the encoded solution to the problem. At the beginning of the algorithm the first two categories are initialized with solutions (nectar sources) generated randomly.

At each iteration the search for optimal solution consists of three steps: sending the employed bees to the nectar sources and then measuring nectar amount (calculation of the fitness function); choice of nectar sources for onlooker bees based on information exchange with the employed bees on the nectar amount; finding scout bees and sending them to search for new sources of nectar. After returning to the hive and information exchange each employed bee returns to a place that was visited at the previous iteration and selects a new source of nectar based on information received from the other bees. Onlooker bees select the new source on a probabilistic basis, based on information collected by employed bees. The choice of a new source of nectar by onlooker bee is described by the equation

$$p_k = \frac{fit_k}{\sum_{n=1}^N fit_n} \quad (1),$$

where P_k is the probability of choosing k -th nectar source, fit_k - the fitness function value of k -th solution calculated in accordance with nectar amount (objective function), N - the number of nectar sources equal to the number of employed bees. The choice of a new nectar source by employed and onlooker bee is described by the equation

$$v_{ij} = x_{ij} + \phi_{ij}(x_{ij} - x_{kj}) \quad (2),$$

where $k \in \{1, 2, \dots, SN\}$, $j \in \{1, 2, \dots, m\}$, m - dimension of the problem to be optimized, ϕ_{ij} - random value in the range $[-1; 1]$. The value of k for the employed bees is chosen randomly, while for the onlooker bees its choice is based on equation (2), j is chosen randomly in both cases. The fitness function fit_i of i -th bee in the case when the maximum of the objective

function $f(x)$ is sought coincides with the latter. When the minimum of objective function is sought, fit_i is defined as

$$fit_i = \frac{1}{1 + f(x_i)}, \quad f(x_i) \leq 0, \quad fit_i = 1 + \text{abs}(f(x_i)), \quad f(x_i) < 0$$

Besides that there must one more bee `best_ever`, which stores the best solution found by all bees to this moment. After the algorithm finishes, this solution represents the algorithm's result.

When the problem to solve is data clustering, every bee is a vector $\{x_1, x_2, \dots, x_K\}$ of centroids' coordinates. Thus the ABC-algorithm can be applied to the clustering problem solution as it was described. It worth to mention, that the ABC-algorithm has one significant drawback.

As it follows from (2), a new nectar source is placed in some random neighborhood of current solution and it differs from it by a single random coordinate of m -dimensional vector, and it's possible that precisely that coordinate belongs to the optimal solution. At the same time, since the solution's fitness function is evaluated with the use of the whole m -dimensional vector, its value with a significant probability may be less than of that of current solution. Thus that "fit" coordinate found will be lost. For instant, suppose we have an objective function $\|x\| = 0$ with the evident global minimum $x(0; 0)$ and the current solution is $x(6; 8)$ with the objective function value equal to $36 + 64 = 100$, while the `best_ever` bee stores solution $(6;6)$ with the objective value of 72. Assume that as a result of transformation in accordance with (2) solution $(6;8)$ becomes $x(4; 8)$. So its objective function value is 80 which is bigger than 72, and, thus, the `best_ever` solution won't change. At the same time it's evident that the first coordinate of $x(4; 8)$ is nearer to the optimal solution than first coordinate of the `best_ever`, but it will be lost in further solution modifications.

Thus the following improvement of the basic ABC-algorithm is proposed. After modification of nectar source accordingly to (2) by each employed bee and onlooker bee, one solution component is chosen randomly and its value is put into the `best_ever`. If it leads to higher value of fitness function it is stored in the `best_ever` and discarded otherwise.

The second fundamental difference from the basic ABC-algorithm is that the transformation accordingly to the equation (2) is performed not over a single coordinate but over all the coordinates of the centroid.

Hence the following algorithm is proposed.

1. Initialize the nectar sources.

2. Evaluate nectar amount (fitness function value) for all nectar sources.
3. Repeat until termination condition not met.
 - 3.1 For each employed bee calculate new nectar source accordingly to (2).
 - 3.1.2. Change randomly chosen k-th coordinate of the best_ever to the value of k-th component of the current employed bee. If it leads to higher fitness value, keep it.
 - 3.1.3. If trials limit to solution improvement has expired initialize the employed bee with the random solution.
 - 3.2. For each onlooker bee
 - 3.2.1 Evaluate p_k accordingly to (1) and a new nectar source by (2).
 - 3.2.2. Change randomly chosen k-th coordinate of the best_ever to the value of k-th component of the current onlooker bee. If it leads to higher fitness value, keep it.
 - 3.2.3. If trials limit to solution improvement has expired initialize the onlooker bee with the random solution.

The termination criterion is maximal number of iterations. As an additional criterion may be used the maximal number of iterations, during which the solution improvement did not occur. As the fitness function the value reciprocal to SSE was used.

CLUSTERING VALIDATION CRITERIA

There is a lot of clustering validation criteria [7]. All of them take into consideration two factors – cluster density and centroids separation. Among the most often used are Silhouette method and Xie-Beni index.

In the Silhouette method each point x_i in cluster C_j is assigned a value known as the silhouette width calculated as

$$s(i) = (b(i) - a(i)) / \max\{a(i), b(i)\} \quad (3),$$

where $a(i)$ is average distance between x_i and all points included in C_j , $b(i)$ is the minimum average distance between x_i and all points clustered in C_k ($k = 1, 2, \dots, m, k \neq m$). From formula (3) it follows that $s(i)$ has a value between -1 and 1. When $s(i)$ is close to 1, one may infer that x_i is assigned to appropriate cluster. If $s(i)$ is close to zero, it suggests that x_i could be assigned to the nearest neighboring cluster, i.e. this point lies equally far from both clusters. If $s(i)$ is close to -1, one may assume that x_i was

misclassified. Thus, for each cluster C_j a cluster silhouette S_j can be calculated as a sum of all points' silhouette widths in C_j . The Global Silhouette (GS) as an effective validity index is calculated as

$$GS = \frac{\sum_{j=1}^m S_j}{m}$$

It's evident that the bigger GS value, the better clustering quality is.

The Xie-Beni index (XB) is known as the compactness and separation validity function. It defines the intra-cluster compactness as the mean square distance between each data object and its cluster center and the inter-cluster separation as the minimum square distance between cluster centers.

$$XB = [\sum_i (\sum_{x \in C_i} d^2(x_i, c_i)) / n_i] / (n \cdot \min_{i, j \neq i} d^2(c_i, c_j))$$

The best clustering validity is obtained when XB reaches its minimum value.

COMPUTATIONAL EXPERIMENT RESULTS

The synthetic data sets (Data Set1 – Data Set6) of different dimensions (from 500 to 10000 samples) and configurations have been generated to compare k-means and basic ABC-algorithm with the proposed algorithm. The average results on 50 runs of each algorithm are given in Table 1.

Data set	Algorithm	SSE	GS	XB
Data Set1	k-means	3406386.32	430.40	1.88
	ABC-algorithm	1173727.00	538.12	0.05
	Proposed algorithm	927372.00	541.76	0.03
Data Set2	k-means	19257763.78	553.91	0.18
	ABC- algorithm	16461029.00	606.16	0.15
	Proposed algorithm	15569264.00	687.84	0.14
Data Set3	k-means	30543545.39	1832.66	1.11
	ABC- algorithm	13708951.00	2208.12	0.05
	Proposed algorithm	10788121.00	2258.12	0.04
Data Set4	k-means	121737978.32	5126.16	0.24
	ABC-алгоритм	133099598.00	5187.14	0.35
	Proposed algorithm	108519652.00	5420.93	0.19
Data Set5	k-means	20296101.00	1297.00	0.11
	ABC- algorithm	20557622.00	1318.12	0.17
	Proposed algorithm	16139439.00	1341.88	0.08
Data Set6	k-means	12040962.18	212.80	0.53
	ABC- algorithm	5871974.00	412.26	0.09
	Proposed algorithm	5340428.00	409.60	0.08

Table 1. The comparison of algorithms performance

CONCLUSIONS

The results of computational experiment have shown that the proposed algorithm in comparison with the k-means algorithm and the ABC-algorithm allows to obtain better solutions for all criteria. Especially notable advantages of the suggested algorithm are in the case of data sets with a large number of well separated clusters (Data Set1 and Data Set5). Perspective seems the introduction of the scheme, according to which the number of bees increases when the algorithm sticks in the local minimum.

A proposed algorithm is less dependent than the ABC-algorithm on such parameters as the colony size and the number of iterations.

REFERENCES

1. *Jain, A.K., Murty, M.N. and Flynn, P.J.* Data clustering: A review.// ACM Computing Survey. — 1999. — № 31 (3). — P. 264-323.
2. *David Arthur, Vassilvitskii.* k-means++: The advantages of careful seeding. //In Proc. of the 18th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA). — 2007. — P. 1027-1035.
3. *Y. Kao and K. Cheng,* “An ACO-based clustering algorithm” // Proceedings of the 5th International Workshop on Ant Colony Optimization and Swarm Intelligence. — 2006. — P. 340–347.
4. *V. D. Merwe and A. P. Engelbrecht,* “Data clustering using particle swarm optimization” //Proceedings of IEEE Congress on Evolutionary Computation (CEC '03). — 2003. — P. 215–220.
5. *Зорін Ю. М., Подольський С. В.* Генетичний алгоритм кластеризації множини відповідно до заданого критерію // ІІ наук.-техн. конф. «Прикладна математика та комп'ютеринг». Тези доповідей. – К.: НТУУ «КПІ», 14–16 квітня 2010. – С. 335–338.
6. *D. Karaboga and C. Ozturk,* “A novel clustering approach: artificial bee colony (ABC) algorithm” //Applied Soft Computing. — 2010. — № 11, P. 652–657.
7. *M. Halkidi , Y. Batistakis, M. Vazirgiannis.* On Clustering Validation Techniques.// Journal of Intelligent Information Systems. — 2001. — №17. — P.107–145.

УДК 519.711.74

Сетевая парадигма в когнитивных науках

Жилякова Л.Ю., к.ф.-м.н., Институт проблем управления им. В.А. Трапезникова РАН, г. Москва zhilyakova.ludmila@gmail.com

В работе дается характеристика сложных сетей; приводится обзор моделей случайных графов, порождающих сети с различными свойствами. Описываются методы построения структурных и функциональных сетей мозга и рассматривается применение аппарата сложных сетей в когнитивных исследованиях.

Введение

Сети различной природы – от искусственных нейронных сетей до моделей функциональных сетей участков мозга – уже давно играют ключевую роль в когнитивной науке. Однако именно появление теории сложных сетей произвело прорыв в исследованиях структурных и функциональных сетей мозга [1-3].

Сетевая наука появилась как результат интеграции междисциплинарных усилий в описании свойств сложных систем различной природы. Развитие теории сложных сетей обусловлено по большей части тем, что, благодаря совершенствованию вычислительной техники и программного обеспечения, появилась возможность сбора и обработки больших наборов данных в разных предметных областях. Исследование построенных на их основе сетей привело к фундаментальному открытию: существенно отличающиеся друг от друга сложные системы часто имеют общие ключевые организационные принципы, и могут быть количественно охарактеризованы одними и теми же параметрами. Таким образом, сетевая наука предоставила унифицированные инструменты для исследования различных систем.

На макроскопическом уровне сложность в поведении проявляется при сочетании статистической случайности с регулярностью. Многие сложные системы демонстрируют удивительно сходное макроскопическое поведение, несмотря на глубокие различия в микроскопических деталях каждой системы и в механизмах взаимодействия их элементов. В [4] анализируются сложные сети в таких областях как биология, химия, экология, лингвистика, а также описываются структурные и функциональные сети мозга. В [5] рассматриваются информационные сети и связь теории информационного поиска и сложных сетей. Книга [6] посвящена исследованию свойств социальных и экономических сетей. Обзорные

работы [1-3] описывают состояние дел в области когнитивных исследований, а также исследований структурных и функциональных сетей мозга и из взаимного влияния с помощью сетевой математики. В книгах [4, 7] и многих других работах рассматриваются динамические процессы в сложных сетях.

Свойства сложных сетей

Многие свойства сложных сетей были выявлены непосредственно из эмпирических наблюдений, и некоторые – еще до появления сетевой математики.

1. Эффект тесного мира. Это свойство впервые наблюдалось в эксперименте социального психолога Стэнли Милгрэма. Он заключался в следующем. Случайно выбранных людей в Омахе (Небраска) попросили передать письма некоторым незнакомым получателям в Бостоне (Массачусетс), используя только сеть своих знакомств. Среднее количество людей в цепочках, по которым прошли письма, привело к известному афоризму «шесть рукопожатий». Хотя число 6 не универсально, среднее расстояние между парами вершин в реальных сложных сетях обычно очень мало в сравнении с размером сети.

2. Высокая транзитивность. Концепция транзитивности в социальных сетях относится к тому факту, что, например, два друга любого взятого индивида с большой вероятностью также дружат между собой.

Транзитивность может быть измерена численно посредством коэффициента кластеризации, который имеет большие значения почти во всех реальных сетях.

Коэффициент кластеризации: c_i вершины i определен как отношение между e_i – фактическим числом ребер между его ближайшими соседями и максимально возможным числом ребер, $k_i(k_i - 1)/2$, т.е.

$$c_i = \frac{2e_i}{k_i(k_i - 1)}.$$

Коэффициент кластеризации численно оценивает транзитивность сети, измеряя вероятность, что две вершины с общим соседом сами являются соседями. Средний коэффициент кластеризации обозначается через $\langle c \rangle$ и определяется как среднее арифметическое по всем вершинам. В реальных сетях $\langle c \rangle$ принимает значения порядка единицы, в полной противоположности со случайными сетями, в которых коэффициент кластеризации обратно пропорционален размеру сети.

3. Безмасштабность (степенное распределение степеней вершин).

Эмпирически было установлено, что многие естественно возникающие сети – социальные, коммуникационные, биологические, графы цитирований, ссылок в WWW, и другие системы – являются безмасштабными.

Безмасштабная сеть – сеть, в которой степени вершин распределены по степенному закону, $P(k) \sim k^{-\gamma}$, т.е. доля вершин со степенью k асимптотически пропорциональна $k^{-\gamma}$.

γ – характеристическая степень, обычно между 2 и 3.

Критерий безмасштабности лежит в основе многих уникальных свойств сложных сетей, таких как их крайняя устойчивость к случайному удалению вершин, в совокупности с крайней чувствительностью к направленному удалению наиболее связанных вершин.

4. Наличие хабов. Хаб – это вершины сети с наибольшими степенями. Безмасштабность предполагает наличие вершин с очень большими степенями. Именно они поддерживают связность сети и обеспечивают ее малый диаметр.

5. Ассортативность – склонность вершин соединяться со сходными вершинами. Ассортативность по степени означает, что узлы со сходными степенями склонны формировать связи преимущественно друг с другом. В большинстве социальных сетей узлы с высокими степенями связаны с другими такими же узлами, а слабо связанные вершины имеют тенденцию соединяться друг с другом. Свойство, обратное ассортативности, – дисассортативность.

6. Феномен клуба богатых – свойство, наблюдаемое во многих реальных сетях, в которых хабы имеют сильную тенденцию соединяться между собой, нежели с узлами с малыми степенями.

7. Разреженность. Относительно небольшое количество ребер, достаточное, тем не менее, для поддержания связности сети, свойства тесного мира и высокой кластеризации.

8. Модулярность или наличие сообществ с более плотными связями внутри и менее плотными связями между сообществами.

Сетевые модели

Для порождения графов, обладающих перечисленными и некоторыми другими свойствами, было предложено множество различных моделей [8 – 10].

Модель Эрдёша–Реньи

Долгое время парадигмой порождения сетей служила модель случайного графа Эрдёша–Реньи [8, 9]. В этой модели дано множество $V_n = \{1, \dots, n\}$ – множество n изолированных узлов. Каждое ребро между двумя любыми узлами устанавливается с независимой вероятностью $p \in [0, 1]$. Считается, что конструируемый граф не имеет петель и кратных ребер, поэтому максимальное количество его ребер равно:

$$C_n^2 = \frac{n(n-1)}{2}.$$

Из этих условий следует, что построение случайного графа Эрдёша–Реньи представляет собой схему Бернулли с количеством испытаний C_n^2 . Результирующий граф имеет биномиальное распределение степеней, центрирован на средней степени и имеет низкую кластеризацию.

Данные о крупномасштабных сетях ясно показали, что для объяснения наблюдаемых свойств, в частности, большой коэффициент кластеризации и степенной закон распределения степеней, нужны другие модели.

Модели Ваттса–Строгатца и Барабаши–Альберт

Одной из попыток примирить высокую кластеризацию, характерную для регулярных решеток, и малую длину путей, наблюдаемую в сложных сетях, стала модель Ваттса–Строгатца [11]. В случайном графе, сгенерированном по модели Ваттса–Строгатца присутствуют свойства тесного мира (малый диаметр) и высокая кластеризация. В этой модели для n вершин изначально конструируется регулярная круговая решетка, каждый из n узлов которой имеет k соседей – по $k/2$ с каждой стороны. После этого некоторые ребра случайно «перемонтируются» – один конец ребра перебрасывается на другую вершину с некоторой вероятностью p . Вместо ребра (i, j) появляется ребро (i, l) , где l выбирается с равной вероятностью из всех остальных вершин, кроме i , поскольку петли не допускаются. Для малой вероятности p высокая кластеризация, присущая решетке, сохраняется, в то время как введение небольшого количества коротких путей очень сильно уменьшает диаметр сети.

Модель Барабаши–Альберт [9, 12, 13] представляет первое объяснение распределения степеней вершин по степенному закону, обнаруживаемому во многих сложных сетях. Она основана на принципе роста и предпочтительного прикрепления. На каждом шаге времени в сеть попадает новый узел и связывается с существующими узлами с вероятностями, пропорциональными их степеням. Таким образом, «богатые узлы»

становятся богаче. Это правило приводит к степенному закону с масштабom $P(k) \sim k^{-3}$. Показатели степени, отличные от 3, могут быть получены, например, если позволить ребрам «перемонтироваться», как в модели Ваттса–Строгатца.

Перечень существующих на сегодняшний день моделей построения больших сетей, конечно, не исчерпывается описанными, однако, насколько известно, пока не предложено модели, генерирующей граф, обладающий сразу всеми свойствами реальных сложных сетей.

Приложение теории сложных сетей к когнитивной науке

Сетевая основа обеспечивает естественный способ описания нейронной организации. Приход сетевой науки предполагает изменение традиционной «компьютерной метафоры» для мозга на «Интернет-метафору», где неокортекс выполняет задачи «перераспределения пакетов» [1].

Более широко, сетевая теория позволяет произвести сдвиг от редукционистской точки зрения на организацию мозга к точке зрения сложных систем. В этих рамках оптимальное функционирование мозга требует баланса между локальной обработкой и глобальной интеграцией. В частности, кластеризация способствует локальной обработке, а короткая длина путей в нервной сети требуется для глобальной интеграции информации между участками мозга. Два этих фактора могут придавать форму структуре и поведению нервной сети.

Структурные и функциональные сети мозга

Отображению структурной сети человеческого мозга в масштабе как областей мозга так и нейронов посвящено множество исследований ([1, 2] и библиографии к этим работам). Структурные сети человеческого мозга отображаются с помощью диффузной томографии, трактографии и др. Были получены структурные паттерны соединений из кросс-корреляции в толще коры, которые могут косвенно указывать на наличие кортикальных путей [14]. Графовый анализ показал присутствие свойства тесного мира и существование локальных сообществ сетях головного мозга. Более детальный анализ корреляции наборов данных о связях, полученных из толщи коры, для выявления модулярности или определения структуры сообществ, выявил значительное перекрытие между анатомическими сетевыми модулями и функциональными системами в коре.

Хотя анализ структурных сетей помогает нам понять фундаментальную архитектуру связей между областями, необходимо также непосредственно рассмотреть функциональные сети, чтобы выяснить, как эта

архитектура поддерживает нейрофизиологическую динамику. Несмотря на значительную неоднородность методологических подходов, наблюдается достаточно сильная конвергенция исследований функциональных сетей головного мозга. Свойство тесного мира в функциональных сетях было подтверждено с помощью исследований, основанных на данных функциональной МРТ (фМРТ), электроэнцефалографии (ЭЭГ), магнитоэнцефалографии (МЭГ), мультиэлектродного массива (МЕА).

Структурно-функциональные отношения в сетях мозга

В настоящее время неизвестно, претерпевают ли крупномасштабные кортикальные сети мозга взрослых людей структурные перестройки в быстрых временных масштабах. Большинство наблюдаемых изменений были связаны со старением, развитием болезней, или приспособляемостью, обусловленной опытом [2].

В отличие от этого паттерны функциональной связности между областями мозга испытывают спонтанные флуктуации и очень чувствительны к возмущениям, которые, например, вызваны сенсорными входами или когнитивными задачами, на временных шкалах порядка сотен миллисекунд. Эти быстрые реконфигурации не влияют на стабильность глобальных топологических характеристик. Постоянство функциональных сетей, связанных с состоянием покоя, обеспечивает возможность исследовать, насколько паттерн функциональных соединений определяется лежащей в основе структурной сетью.

Все результаты показывают, что структурные и функциональные сети мозга очень тесно взаимосвязаны и имеют общие топологические свойства.

Изменение топологии. Перколяция

Сетевой подход предлагает объяснение быстрых и даже внезапных появлений новых когнитивных функций в процессе развития, так же, как и деградации этих функций с возрастными изменениями или при нейродегенеративных заболеваниях.

Такие внезапные скачки могут появляться из гладких изменений, если система переходит порог перколяции [1, 15], т.е. ключевую точку, в которой сеть внезапно становится связной (в процессе развития) или не связной (возрастные изменения или болезнь). Существование такой точки продемонстрировано на семантических сетях, извлеченных из Википедии, развивающихся с помощью добавления новых страниц. Концеп-

ция перколяции также инспирировала недавнее объяснение гиперпрайминга у пациентов с болезнью Альцгеймера на ранних стадиях [1].

Многие работы посвящены установлению характера изменений топологии структурных и функциональных сетей при различных патологиях (болезнь Альцгеймера, шизофрения, аутизм) [16,17]. Сетевые свойства могут использоваться как диагностические маркеры для выявления этих и других заболеваний. Будущие исследования могут быть направлены на использование теоретикографовых характеристик для определения терапевтической эффективности лечения при восстановлении топологически оптимальной конфигурации сетей у пациентов.

Заключение

Теория сложных сетей обещает стать значительной интегративной основой в когнитивных исследованиях для понимания сознания и поведения в широком диапазоне масштабов, от процессов в мозге до паттернов социальных и культурных взаимодействий. В общем, сетевая математика может помочь когнитивной науке повысить внутреннюю согласованность и выявить ее связи со многими областями, в которых ее аппарат оказался плодотворным.

Литература

1. *Baronchelli, A., Ferrer-i-Cancho, R., Pastor-Satorras, R., Chater, N., and Christiansen, M.H.* Networks in Cognitive Science. // Trends in Cognitive Sciences July 2013, Vol. 17, No. 7.
2. *Bullmore E. and Sporns O.* Complex brain networks: graph theoretical analysis of structural and functional systems. // Nature, March 2009.
3. *Park H.-J. and Friston K.* Structural and Functional Brain Networks: From Connections to Cognition. Science 342, (2013).
4. *Dehmer, M., Emmert-Streib F.* Analysis of complex networks. From biology to linguistics. Wiley-VCH Verlag GmbH & Co. KGaA. 2009.
5. *Ландэ Д. В., Снарский А. А., Безсуднов И. В.* Интернетика: Навигация в сложных сетях: модели и алгоритмы. М.: Либроком, 2009.
6. *Jackson, M.O.* Social and Economic Networks. Princeton University Press. 2008.
7. *Barrat, A., Barthelemy, M, Vespignani, A.* Dynamical Processes on Complex Networks. Cambridge University Press. 2008.
8. *Колчин В.Ф.* Случайные графы. — М.: Физматлит, 2004. — 256 с.
9. *Райгородский А.М.* Модели случайных графов и их применения // Труды МФТИ. — 2010. — Том 2, № 4

10. *Bollobás, B.* Random Graphs. Cambridge University Press. 2001.
11. *Watts, D.J. and Strogatz, S.H.* (1998) Collective dynamics of ‘small-world’ networks. *Nature* 393, 440–442.
12. *Barabási, L.-A., Albert R.* Emergence of scaling in random networks // *Science*. — 1999. — V. 286. — P. 509–512.
13. *Barabási, L.-A., Albert R. Jeong H.* Scalefree characteristics of random networks: the topology of the world-wide web // *Physica*. — 2000. — V. A281. — P. 69–77.
14. *Chen, Z. J., He, Y., Rosa-Neto, P., Germann, J. & Evans, A. C.* Revealing modular architecture of human brain structural networks by using cortical thickness from MRI. *Cereb. Cortex* 18, 2374–2381 (2008).
15. *Bollobás, B. Riordan, O.* Percolation. Cambridge University Press. 2006.
16. *Wright, I. C. et al.* Supra-regional brain systems and the neuropathology of schizophrenia. *Cereb. Cortex* 9, 366–378 (1999).
17. *Barabási, L.-A. et al.* (2011) Network medicine: a network-based approach to human disease. *Nat. Rev. Genet.* 12, 56–68.

УДК 004.822

Представлення правових знань на основі онтологій

*Карасюк В.В., к.т.н., доц.,
Національний юридичний університет імені Ярослава Мудрого,
м. Харків vl_karasuk@ukr.net*

В роботі розглядається методологія та програмний засіб для створення онтології правових знань. Особливості правової інформації наклали свої вимоги на структуру бази знань. Сучасна реалізація пакету програм на основі запропонованої методики передбачає web-інтерфейс і можливість наповнення бази багатьма користувачами.

Вступ

Правники основним об'єктом своєї праці мають документи. Вони використовують різні документи, що відносяться до певних юридичних ситуацій: нормативні документи, підзаконні акти, правозастосовчі документи, рішення, ухвали, поточні документи тощо. Доступний обсяг поширених правових баз, які пропонуються користувачам, містить біля 500 тисяч документів. Це дуже велика кількість. Тому не дивно, що серед них мають місце колізії, досить заплутана структура підпорядкування понять в окремих галузях права, невідповідність документів один одному. Ці проблеми ставлять задачу формалізації опису правових знань. До того ж, рухаючись у напрямку побудови інформаційного суспільства, ми мусимо мати впорядковане доступне електронне представлення системи правових знань.

У багатьох країнах світу вже створені бази правових знань для їх практичного використання. За основу їх опису майже у всіх випадках взята онтологічна структура. Як приклади, можна назвати такі відомі правові онтології.

Цікавими системами є FOlaw (Functional Ontology of Law) [1], LRI Core [2], Frame-based Ontology, CLO (Core Legal Ontology), Jurwordnet [3]. Слід також згадати системи юридичного виводу на основі прецедентів LCBR (legal case-based reasoning) [4,5]. Для владнання онлайн суперечок використовується платформа Ontomedia, яка працює з Core Mediation Ontology і онтологічний субдомен використовується для посередницьких функцій з окремими субдоменами (наприклад, проблемна область споживача, проблемна область моделі сім'ї, здоров'я і т.ін.). Ця модель є осно-

вою для представлення понять і відношень у посередництві [6]. В Україні відомо про роботи у напрямку створення онтології правового спрямування на базі технології METHONTOLOGY [7] та правової онтології учбового спрямування JurOnt [8].

В цілому завдання побудови онтологічних структур, що описують знання обраної предметної області, є привабливою і актуальною для багатьох сфер діяльності і досліджень, особливо для тих, які мають наповнення, що динамічно змінюються.

Мета даної роботи – розглянути особливості онтологічних моделей правової інформації для систем електронного навчання та надати характеристику підходу до наповнення системи.

Онтологія як основа формування структури правових знань

Є низка причин, за яких доцільно використовувати онтологічне уявлення знань [9]: сумісне використання людьми або програмними агентами загального розуміння структури інформації; можливість повторного використання знань проблемної області; створення явних припущень у предметній області; виокремлення знань у предметній області від оперативних знань; можливість об'єктивного аналізу знань у предметній області.

У роботі [10] вказано, за рахунок чого онтології в праві мають серйозне обґрунтування для свого застосування. Це взаємозв'язок правових норм між собою; важливість юридичних рішень; ступінь мінливості правових норм; подібність різних гілок права та інші. Тому онтологічний інжиніринг у правовій сфері має характерні риси: 1) велика кількість загальноприйнятих понять зі своєю специфікою застосування; 2) відмінність в структурах різних областей права; 3) наявність проміжного загальнотеоретичного правового рівня між онтологією верхнього рівня і онтологією предметної області; 4) велика кількість теоретичних припущень і абстрактних конструкцій, що залежать від специфіки правових поглядів. Тому варто вважати, що найбільш ефективним є підхід, заснований на вивченні природної мови та побудові онтологій на великих текстових корпусах.

При створенні онтологій важливе значення має оцінка їх придатності та якості на основі системи критеріїв. Всебічна типологія критеріїв якості онтології передбачає [11]: епістемологічну адекватність (епістемологічна ясність, епістемологічна інтуїція, епістемологічна доречність, епістемологічна закінченість); експлуатаційна адекватність (послідовність, обчислюваність); можливість багаторазового використання (завдання і методу,

предметної області). Ці критерії доцільно використовувати як стандарт для опису всіх онтологій.

Аналіз розглянутих проєктів, в яких застосовані онтологічні принципи опису правових знань та методик їх побудови наводить на думку, що в даний час не напрацьовані оптимальні принципи побудови онтологічних інформаційних систем.

Особливості структури правової онтології

Пропонується структуру правових знань уявити онтологією, яка описує зв'язок пов'язаних понять і належних до них описів. Для проєктування структури онтології правової інформації слід врахувати ряд особливостей, притаманних правовій інформації, і в особливості звернути увагу на наступні:

- синонімічність визначень (вузлів онтології);
- обмеженість конкретних формулювань нормативних документів у часі;
- наявність обов'язкового зв'язку визначень (вузлів онтології) зі строгими формулюваннями (законодавчими визначеннями) у нормативних документах.

Тобто база даних, що реалізує онтологію, включає такі частини:

- поняття і зв'язки;
- зв'язки між групами понять;
- тексти - джерела;
- словесне відображення понять і зв'язків;
- індекси вживання понять і зв'язків у тексті.

Розглянемо кожну частину.

- поняття та зв'язки: записуються з унікальними ідентифікаторами, з рядком назви для виведення в графічному інтерфейсі.

- тексти - джерела: зберігаються у вигляді набору рядків - речень, з фіксацією належності до розділу і тексту, тобто серед них можна виділити наступні сутності:

- текст - упорядкована множина розділів, тип, назва, список авторів та коментарі;

- розділ - назва, упорядкована множина підрозділів і впорядкована множина речень;

- речення - текст у вигляді рядка і типу речення (з вказівкою належності до основного тексту розділу або його заголовка).

- словесне відображення понять і зв'язків схематично можна зобразити так, як показано на рис. 1.

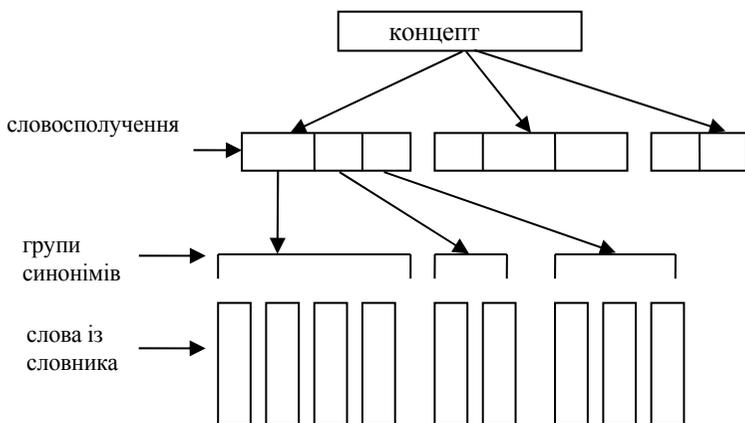


Рис.1. Структурне відображення понять і їх зв'язків у базі даних онтології.

Практична реалізація онтології правової інформації

Запропонована структура моделі знань реалізована у вигляді програмного комплексу, названого JurOnt. Комплекс виконаний у вигляді чотирьох підсистем, з використанням сучасних технологій об'єктно-орієнтованого візуального програмування, в середовищі Eclipse 3.4 на мові Java з підтримкою JDK версії 1.6. Передбачений web інтерфейс користувача і автоматизований режим роботи з базою знань, у тому числі автоматизоване наповнення онтології з доступних текстових документів.

Передбачаються наступні режими роботи з системою.

- Режим створення онтологічної системи. У цьому режимі фахівці - когнітологи розробляють ядро онтологічної системи і наповнюють середу первинними знаннями .

- Режим експлуатації. Передбачає одночасне використання системи багатьма користувачами з доступом до знань в єдиній базі.

- Режим індивідуального доповнення, розвитку та інтеграції. Передбачає роботу в системі з доповненням існуючої бази знань і створенням власної бази. Надалі передбачається інтеграція баз знань, створених окремими користувачами, в єдину систему, з перевіркою можливих суперечностей у тлумаченнях понять.

• Навчання та оцінювання знань. У цьому режимі створюється система оцінювання на основі послідовностей взаємодії понять і формування онтології «питання - відповідь».

• Онтологічний аналіз. В основі онтологічного аналізу лежить опис предметної області в термінах понять, їх взаємозв'язків, а також перетворення понять в процесі прийняття рішень.

Система розробляється таким чином, що можлива робота користувача (експерта) з кожним реченням висхідних текстів послідовно. Тобто послідовність роботи буде наступною:

- користувач читає чергове речення і усвідомлює його зміст;
- виділяє поняття, які згадуються в реченні;
- визначає зв'язки між групами понять;
- встановлює додаткові зв'язки, які уточнюють зміст основних зв'язків або описують деяким чином складні поняття, у тому числі пов'язує їх з законодавчими визначеннями.

Далі йде технічний етап внесення знань у базу даних.

Основною підсистемою, що визначає межі використання системи, є аплікація користувача. Глобальними завданнями аплікації користувача є: 1) навігація в онтології; 2) пошук фрагментів текстів - джерел, що відповідають елементам онтології; 3) перегляд текстів - джерел у повному обсязі, у вигляді тексту, поділеного по розділам, маркованого тексту.

Кожне із зазначених завдань припускає реалізацію певних сценаріїв роботи з аплікацією.

Система JurOnt в даний час прийнята в дослідну експлуатацію в локальній мережі Національного юридичного університету імені Ярослава Мудрого. Іде процес наповнення бази знань інформацією з області кримінального права України. Експерти та користувачі системи будують онтологію на основі змісту підручника з кримінального права, який є базовим для вивчення цієї дисципліни в університеті і з будь-яких інших джерел. Надалі в базу знань буде включена інформація з суміжних галузей права.

Слід наголосити на тому, що процес наповнення онтології буде виконуватись великою кількістю користувачів. Це, з одного боку, дає можливість швидко просуватися у наповненні онтології, а з іншого – створює проблеми, пов'язані із узгодженням фрагментів онтології, створених різними користувачами і включенням їх до основної онтології. Поки що це відбувається за рахунок розмежування гілок онтології для заповнення окремими користувачами. У наступному планується побудова автоматизованої (а у перспективі – і автоматичної) процедури злиття фрагментів

онтології. Тобто передбачається використання принципів краудсорсінгу при наповненні онтології.

Висновок

У результаті дослідження, аналізу переваг і недоліків різних систем подання знань обрана онтологічна модель знання. Сформована структура бази даних, яка реалізує базу знань; розроблені програмні додатки, які орієнтовані на технологію "клієнт-сервер" і забезпечують побудову семантичної мережі на сервері з можливістю багатокористувальницької роботи. Система прийнята в дослідну експлуатацію в Центрі інформаційних технологій Національного юридичного університету.

Перспективними для подальших досліджень є такі напрямки:

- розробка засобів автоматичного порівняння онтологій, це важливий інструмент для оцінки повноти, непротиворіччя онтологій, створених різними користувачами, а також для порівняння онтологій різних напрямків юридичної діяльності або різних країн, за рахунок розробки метрики близькості;

- розробка інтерфейсу системи на природній мові, це важлива частина створення інтелектуальних систем, яка дозволить залучати до роботи непрофесійних користувачів, або використовувати систему для автоматичного наповнення баз знань довільних текстів;

- створення інтернет-порталу для надання консультаційних юридичних послуг без участі фахівця при використанні запропонованої системи в юридичній клініці;

- дослідження впливу принципів самоорганізації на якість створюваної множиною користувачів онтології в предметній області правознавства.

Література

1. *Breuker J., Hoekstra R.* Epistemology and ontology in core ontologies: FOLaw and LRI-Core, two core ontologies for law // Proceedings of the EKAW04 Workshop on Core Ontologies in Ontology Engineering, Northamptonshire, UK. - p. 15 – 27. Режим доступу: <http://dare.uva.nl/document/8751>.
2. *Gangemi A., Prisco A., Sagri M.T., Steve G., Tiscornia D.* Some ontological tools to support legal regulatory compliance, with a case study // Workshop WORM Core, LNCS: Springer Verlag. – 2003. - p. 607–620. Режим доступу: <http://www.loa.istc.cnr.it/Papers/WORM-CORE.pdf>

3. *Sagri M. T., Tiscornia D., Bertagna F.* Jur-WorNet // Second International Wordnet Conference - GWC - 2004. Brno: Masaryk University. - p. 305-310.
4. *Henderson J., Bench-Capon T.* (2001) Dynamic arguments in a case law domain. // ICAIL '01: proceedings of the 8th international conference on artificial intelligence and law. New York: ACM Press. - 2001. - p 60–69. DOI:10.1145/383535.383542
5. *Zeng Y., Wang R., Zeleznikow J., Kemp E.A.* (2005) Knowledge representation for the intelligent legal case retrieval. // KES (Part 1). Lecture Notes in Computer Science, Springer. – 2005. – Vol. 3681. – p. 339 – 345. DOI: 10.1007/11552413_49
6. *Poblet M., Casanovas P. López-Cobo J-M. Castellás N.* ODR, Ontologies, and Web 2.0. // Jurnal of Universal Computer Science. – 2011. – Vol. 17(4). – p. 618-634. Режим доступа: http://www.researchgate.net/publication/258046605_ODR_Ontologies_and_Web_2.0
7. *Хала Е.* О построении онтологии для правовой области с применением технологии METHONTOLOGY / Е.А. Хала // Збірник наукових праць. К.: Інститут проблем моделювання в енергетиці ім. Г.Є.Пухова. – 2012. – Випуск 64. – с. 64 – 71.
8. *Tatsyi V., Getman A., Ivanov S., Karasiuk V., Lugoviy O., Sokolov O.* Semantic network of knowledge in science of law // Automation, Control, and Information Technology (ACIT 2010): Proceedings of the IASTED International Conference on Automation, Control, and Information Technology, Novosibirsk, Russia / The International Association of Science and Technology for Development. – Anaheim, USA, Zurich, Switzerland: ACTA Press. - 2010. p. 218 – 222.
9. *Noy N., McGuinness D.* Ontology development 101. Stanford Knowledge Systems Laboratory Technical Report KSL-01-05. – 2001. Режим доступа: <http://www.ksl.stanford.edu/people/dlm/papers/ontology-tutorial-noy-mcguinness.pdf>
10. *Bench-Capon T., Visser P.* (1997) Ontologies in Legal Information Systems; The Need for Explicit Specifications of Domain Conceptualisations // Proc. of the 6th Int. Conf. on Artificial Intelligence and Law. Melbourne, Australia. – 1997. – p. 132-141. Режим доступа: <http://www.egov.ufsc.br/portal/sites/default/files/anexos/5718-5710-1-PB.pdf>
11. *Visser P., Bench-Capon T.* A Comparison of Four Ontologies for the Design of Legal Knowledge Systems // Artificial Intelligence and Law. – 1998. – № 6. – p. 27-57.

УДК 007: 304: 659

Новые подходы к извлечению знания: архитектоническая парадигма

Комаров-Ермолов А. И., студент, Херсонский национальный технический университет, г. Ялта, phylosvet@gmail.com

В докладе представлена система взглядов на процесс извлечения знания (Knowledge transfer), алгоритм работы когнитолога с экспертом как новый метод архитектурного дифференциала.

Введение

Настоящее исследование ставит своей задачей разработку основ интегрального холистического подхода к извлечению и передаче знаний с помощью архитектурных схем сенсорного восприятия. Объектом исследования является современная парадигма моделирования структур образной информации передачи знания, предметом – извлечение и передача знания методом архитектурного дифференциала. Цель исследования – разработка принципов холистического конвергентного подхода извлечения и передачи знания на основе образов (архитектонических схем) сенсорного восприятия.

Что такое извлечение знания?

Сегодня существование в корпорациях эффективной системы передачи знаний является ключевым преимуществом в конкурентной среде [1]. Действительно современная рабочая среда наполнена пластом неявного знания, извлечение и передача которого сопряжена с трудностями слабой формализации. Научные исследования директора Института передачи знаний профессора Роберта Хоулетта (Йорк, Великобритания) указывают на отсутствие универсальных методов извлечения знания и необходимость выработки адекватной методологии процесса передачи знания [2]. Одним из самых перспективных направлений передачи знаний являются подходы к созданию моделей представления структур образной информации [3], в конструировании которых, по нашему мнению, может быть сделан акцент на создании образа (архитектонической схемы) с использованием формализации восприятия элементов активной сенсорики.

Термины «извлечение знания» и «передача знания» в данной работе суть тождественны, с той лишь разницей, что при извлечении знания условный вектор активного действия направлен от когнитолога к эксперту, а при передаче знания – от эксперта к когнитологу, не изменяя при этом качественных характеристик данного процесса. Указанные термины используются в контексте спектра коммуникационных процессов, обеспечивающих передачу явного и неявного знания совокупными вербальными, и невербальными методами, а также процесс формализации знания, обеспечивающий тождественность представлений коммуникатора и реципиента через информационный продукт (архитектоническую схему), конструируемую когнитологом.

Архитектонический дифференциал

Архитектоника - естественно упорядоченная искусственная система [4].

Нами разработан метод архитектурного дифференцирования, являющийся собой четыре этапа извлечения и передачи знания с использованием формализации восприятия элементов активной сенсорики (Рис. 1). На Рис.1 представлен реальный прибор с соответствующими функциями, придуманный автором данного доклада, успешно апробированный на практике и показавший свою эффективность.

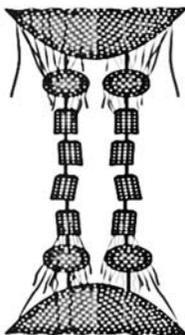


Рисунок 1. Архитектонический дифференциал

На первом этапе «Создание онтологии предметной области»:

- а) определяем однозначность интерпретируемых экспертом терминов;
- б) осуществляем первичный информационно-теоретический синтез. Первичный соотносно-категориальный анализ. Дифференцирование базовых, взаимовложенных категорий для дальнейшего их структурного раз-

ложения при помощи фрактально используемого по отношению к ним метода Архитектонического дифференциала;

в) осуществляем практическое использование метода Архитектонического дифференциала.

Архитектонический дифференциал являет собой конструкцию, состоящую из двух симметричных систем элементов, расположенных параллельно друг другу, друг под другом и представляющих собой четвертную симметрию, символизирующих последовательную этапность порядков абстрагирования элементов передаваемого знания (причина и следствие предстают параллельно в данном процессе абстрагирования), обеспечивающую процесс объективного познания и здоровых семантических реакций, как у индивидуума описывающего данное знание, так и у индивидуума воспринимающего данное описание.

Процесс тактильного восприятия экспертом элементов Архитектонического дифференциала позволяет более эффективно задействовать операционные рефлекторно-импульсные нейронные цепи в коре больших полушарий головного мозга, восстановить старые рефлекторно-импульсные нейронные цепи и создать новые. Наглядное представление целевого конечного тождества знания эксперта со знанием когнитолога обуславливает концентрацию максимальных усилий эксперта для эффективности данного процесса. Доктор Ю. Херик считает, что практика мышления, принятия решений, чувствования, различения и сочувствия в некоторой степени формирует личность мыслителя. Предположительно, стабильные паттерны корковых ассоциаций изменяются при выполнении этих действий, также как на низшем уровне мышцы меняются при систематических упражнениях. Из этого следует, что, хотя, при применении данного метода эксперту будет необходимо потратить больше времени и энергии для сосредоточения, формулирования и объяснения, но при этом будет иметь место благоприятное воздействие данной процедуры на функциональные способности его (эксперта) коры больших полушарий головного мозга, что, в свою очередь, даст положительный эффект для развития его же (эксперта) собственного интеллектуально-творческого потенциала и укрепления психологической устойчивости;

На втором этапе «Моделирование теоретических описаний и практических действий эксперта»:

а) осуществляется практическое представление, если таковое возможно, экспертом своего теоретического описания;

б) осуществляется сегментарный характер передачи знаний. Посегментарная теоретическая и практическая интерпретация когнитологом полу-

ченной информации, уточнение и исправление экспертом информации усвоенной когнитологом;

На третьем этапе «Формализация полученного знания методом архитектурного моделирования» мы применяем системологический подход, а именно метод архитектурного моделирования [4], что, по нашему мнению есть целесообразно и рационально для исследования предметной области передачи знания. Профессор Е. Соловьёва обосновывает стратегическую необходимость системологического, естественного классифицирования каждой отдельной исследуемой предметной области, что, по её мнению, является адекватной методологией формализации и дальнейшего качественно устойчивого познания формализованного в преемственности поколений [5];

На четвертом этапе «Верификация переданного знания» мы осуществляем процесс передачи извлечённого из эксперта знания когнитологом реципиенту (третьему лицу) при строгом наблюдении за данным процессом эксперта, без его (эксперта) активного участия, что, в свою очередь, наглядно продемонстрирует коэффициент усвоенной когнитологом информации и, даст возможность эксперту (после демонстрации процесса в целом) скорректировать усвоенное знание когнитологом методом конструктивной критики теоритического и практического аспектов. Повторяемость выше описанного процесса должна иметь место до момента наступления качественно адекватной (в понимании эксперта) передачи знания когнитологом реципиенту.

Приведенный выше алгоритм предусматривает индивидуальное коммуникационное взаимодействие когнитолога с экспертом.

Архитектоника понятия «Intelligence»

В качестве верификации предлагаемый метод использован для извлечения знания о понятии «Intelligence». Нами приведена архитектурная схема понятия «Intelligence» научного руководителя настоящего исследования Г. Брахмана. В представлении Г. Брахмана, архитектурное понимание термина «Intelligence» снимает проблематику происхождения и понимания термина «Искусственный интеллект» как устоявшегося значения перевода английского термина «Artificial Intelligence».

По данным Г. Брахмана англоязычный термин «Intelligence» не имеет точных аналогов в украинском и русском языках, соответственно, не существует и однозначного перевода именуемого им понятия. На сайте Центрального разведывательного управления США можно найти следующие определение «Intelligence»: «информация, в которой нуждаются лидеры

нации с целью обеспечения безопасности страны». Это слишком простое определение термина «Intelligence» совершенно не выражает сущности «Intelligence» как надинформационного понятия.

Сущностной причиной трудностей, возникающих в процессе перевода термина «Intelligence», является разница в ментальном восприятии мира, связанная с культурной моделью интеллекта, в англоязычном аналоге которой мысль не отделена от эмоций. Данный термин несет как когнитологическую, так и эпистемологическую нагрузку одновременно, что выражается в именовании широкого понятия, значение которого находится в пределах обозначения, как и процесса превращения данных в информацию, и знание, так и результата такого процесса – знания, как эпистемы. Следовательно, трудности в понимании исходного значения данного термина связаны, по мнению Г. Брахмана, именно с проблемами симультанного конвергентного осознания этих двух значений (Рис. 2).

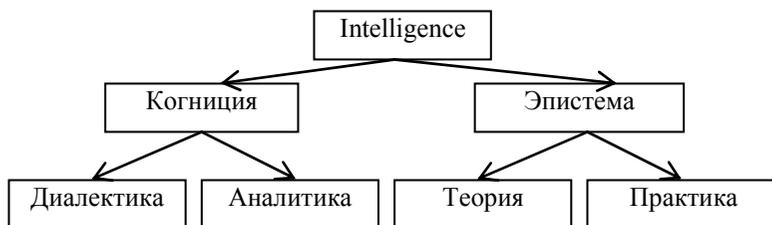


Рисунок 2. Архитектоника понятия «Intelligence»

В системе принципов классической научной рациональности следует осознавать рельефность полярных оппозиций терминов «когниция» и «эпистема» как взаимозависимость концептов антиномической пары, выражающей сущность понятия «Intelligence», - сложное понятие, гипероним концептов интегрированных когнитивных, эмоциональных и социальных процессов постижения разумом, сущностью которого является итеративное осмысление материализуемого *in vivo* знания, постижение которого возможно только в поле действия знание-ориентированной коммуникации. По мнению Г. Брахмана, представление сущности термина «Intelligence» лежит в Аристотелевском понимании энтелехии, как осуществленности, одновременно заключающей в себе смысл, цель и соответствующую ей реализацию. Таким образом, когниция – это совокупность процессов мен-

тальных репрезентаций и категоризаций элементов цели и смысла в иную структуру (образы, фреймы), индивидуальная формализация, облегчающая обработку информации. Когниция – всегда схема, отлична от знания, это когнитологическая эвристичность, обеспечивающая эффективность последующей рациональной деятельности. В отличие от когниции, эпистема (в переводе с греческого – знание) – дискурс-формирование, парадигмально определяющее систему способов, модусов, технологий, подходов онтологизации картины мира. Понимание сущности когниции открывает новые подходы формализации когнитологической эвристичности как гештальта, обеспечивающего эффективность создания и применения нового знания (эпистемы).

Создание таких структур образной информации всегда основано на операциях сравнения. Еще Аристотель антиномически противопоставлял диалектику и аналитику, как исследование того, что бывает не всегда и не везде, но является правилом и, того, что существует всегда и везде. Применение метода архитектурного дифференциала позволило выявить сенсорические представления Г. Брахмана о диалектике, как способе сравнения своего образа с чужим, при котором в начале происходит полное заимствование чужого образа (диалектика (Рисунок 2)), «делание» его своим с дальнейшим сравнением двух (своего-изначально и своего-заимствованного) равнозначных по приоритету выбора (свое-чужое) образов, что обеспечивает большую адекватность сравнения вне приоритетности своего над чужим, а также выработку системы общих правил. В отличие от диалектики, аналитическое сравнение подразумевает конкурентное противопоставление своего и чужого образа, в конкурентно-естественном процессе вытеснения и замещения одного из них, сверхприоритет отдается своему образу. Таким образом, диалектику в создании интеллектуальных систем следует использовать в конструировании образов продукционных правил, обеспечивающих эффективную редукцию существующего к фреймам – информационным структурам, обеспечивающим эффективную интеллектуальную деятельность.

Такое понимание процесса познания (когниции) обеспечивает получение нового знания (эпистемы), формализованной парадигмы, использование которой приносит стабильный положительно прогнозируемый результат. Очевидно, что антиномическими элементами знания-эпистемы являются элементы теории и практики.

Прогностическая сила архитектурной схемы (Рисунок 2) определяет использование диалектики в создании теоретической парадигмы, в то время, как в конструировании практических элементов интеллектуальной

деятельности следует использовать аналитику (теория – свойство объекта «диалектика», практика – свойство объекта «аналитика»).

Заключение

Предложенная концепция процесса передачи знания позволяет зафиксировать чётко описанный алгоритм работы когнитолога с экспертом, являющийся собой универсальный инструмент извлечения (передачи) знания в любой предметной области. Применение данного алгоритма обладает многогранным полезным эффектом: передача знания экспертом когнитологу (и персонифицированным в его лице будущим поколениям); системологически сформулированная, не допускающая возможности к разночтению и личностной псевдоинтерпретации формализованная модель знания в данной предметной области; активизация функциональных способностей коры больших полушарий головного мозга эксперта, что, в свою очередь, даёт положительный эффект для развития его же (эксперта) собственного интеллектуально-творческого потенциала и укрепления психического здоровья; эвристическое воздействие на творческий потенциал когнитолога и реципиента.

Литература

1. *Гаврилова, Т.* Извлечение знания: психологический аспект / Т. А. Гаврилова. – СПб.: Enterprise Partner, 2001, № 8 (25).
2. *Howlett, R.* Innovation through Knowledge Transfer / R. Howlett. – Berlin: Springer – Verlag Berlin Heidelberg, 2010. – 369 p.
3. *Валькман, Ю.* О когнитивной семиотике / Интеллектуальный анализ информации: матер. XII Международ. науч. конф. им. Т. А. Таран ИАИ-2012 / Ю. Р. Валькман. – К., 2012.
4. *Брахман, Г.* Архітектонічне моделювання елементів ітеративності консолідації інформації: дисертація кандидата наук із соціальних комунікацій: 27.00.07 / Г. О. Брахман. - К., 2012. – 200 с.
5. *Соловьёва, Е.* Естественная классификация: системологические основания / Е. А. Соловьёва. – Харьков: ХНУРЭ, 1999. – 222 с.

УДК 004.048

Інтелектуальна система автоматизованої ідентифікації клієнтів мобільної мережі

Копичко О.С.¹, Копичко С.М.², Мороченець О.І.², Рибачок Г.І.¹

¹Служба безпеки України

*²Національний технічний університет України «КПІ», м. Київ
aris100tel@gmail.com*

В статті запропоновано ідею створення системи ідентифікації клієнтів мережі стільникового зв'язку на основі інформації про їх мобільні пристрої та місцезнаходження абонентів в конкретний проміжок часу. Також в роботі наведено опис алгоритму ідентифікації абонентів та рекомендації для його програмної реалізації в подальшому.

Вступ

Одним із затребуваних і важливих напрямків роботи науково-аналітичних підрозділів різноманітних спецслужб та операторів мобільного зв'язку є обробка й аналіз інформації щодо абонентів мобільних пристроїв за заданим критерієм. Проте великі об'єми даних неможливо коректно структурувати для їх аналізу без відповідної автоматизації. Саме тому виникає необхідність у створенні інструментарію та окремих систем автоматизованого аналізу даних, які б дали змогу підвищити продуктивність праці та забезпечити прийняття своєчасних і обґрунтованих рішень.

Постановка задачі

Метою даної роботи є опис алгоритму для отримання даних про місцезнаходження абонентів в конкретний проміжок часу та їх мобільні пристрої. А також надання рекомендацій, які можуть бути застосовані при програмній реалізації даного алгоритму.

Для досягнення даної мети необхідно розв'язати такі основні групи задач:

1. Ознайомитися з існуючими рішеннями та виділити їх сильні та слабі сторони.
2. Провести огляд математичних методів, які можуть бути використані для реалізації системи.
3. Розробити і описати систему автоматизованої ідентифікації клієнтів мобільної мережі.

4. Навести рекомендації для подальшого розвитку і впровадження системи.

На сьогоднішній день існує достатньо програмних рішень для збору даних та приведення їх до потрібного формату, але аналіз і обробка уже відформатованих даних часто виконується вручну за допомогою робочого персоналу і нерідко займає тривалий період часу. Такі аналіз і обробка є економічно не вигідними та неефективними, тому що потребують значних ресурсів, крім того, вихідні дані можуть стати неактуальними, ще задовго до завершення повного аналізу.

Практична цінність даної роботи полягає у створенні робочого алгоритму для системита написанні програмного продукту, який повинен автоматизувати і прискорити процес аналізу даних.

Основна частина

Поняття інтелектуального аналізу даних

Інтелектуальний аналіз даних (англ. DataMining) — виявлення прихованих закономірностей або взаємозв'язків між змінними у великих масивах необроблених даних [1].

Data Mining знаходиться на перетині баз даних, статистики і штучного інтелекту[2], що дозволяє об'єднати здобутки і можливості цих галузей для аналізу даних і зробити його дійсно інтелектуальним.

Завдання, які вирішуються методами інтелектуального аналізу даних, прийнято розподіляти на описові і передбачувальні.

У описових завданнях найголовніше — це дати наочний опис наявних прихованих закономірностей, в той час як в передбачувальних завданнях на першому плані стоїть питання передбачення тих випадків, для яких даних ще немає.

Єдиної думки щодо того, які завдання слід відносити до інтелектуального аналізу даних немає. Більшість авторитетних джерел перераховують наступні: класифікація, кластеризація, прогнозування, асоціація, візуалізація, аналіз і виявлення відхилень, оцінювання, аналіз зв'язків, підведення підсумків[3].

Огляд існуючих програмних рішень

Багатосистем, які виконують інтелектуальний аналіз, пов'язаний з ідентифікацією особи, використовуються спецслужбами різних країн, і тому інформація про них є закритою. Але, скоріш за все, подібний алгоритм використовувався Агентством національної безпеки (АНБ), (англ. NationalSecurityAgency(NSA)) — агентством криптологічної розвідки Сполуче-

них Штатів Америки, що є частиною Міністерства оборони США і відповідає за збір та аналіз іноземної розвідувальної інформації і за захист інформаційних систем та комп'ютерних мереж уряду США[4]. Використовувався він у рамках PRISM — секретної програми електронного спостереження та збору розвідувальної інформації. Програма PRISM є головним джерелом збору первинної інформації, яка використовується АНБ у своїх аналітичних звітах та спеціальних брифінгах для Президента США[5, 6].

Також подібна система могла бути використана в глобальній системі радіоелектронної розвідки «Ешелон», головним оператором якої також є Агентство національної безпеки США. У цій системі подібний алгоритм міг застосовуватися для аналізу даних, які отримувалися в результаті перехоплення та обробки телефонних розмов, факсів та інших інформаційних потоків[7].

Огляд методів інтелектуального аналізу даних

Основу DataMining складають різні методи класифікації, моделювання і прогнозування, що ґрунтуються на використанні дерев прийняття рішення, штучних нейронних мереж, генетичних алгоритмів, еволюційного програмування, асоціативної пам'яті, нечіткої логіки. До методів інтелектуального аналізу нерідко відносять статистичні методи – описовий аналіз, кореляційний і регресійний аналіз, факторний аналіз, дисперсний аналіз, компонентний аналіз, дискримінантний аналіз, аналіз числових рядів і аналіз зв'язків [3].

Більшість методів, що застосовуються для ідентифікації об'єктів або їх параметрів, вирішують завдання класифікації. Але при цьому потрібна навчальна вибірка об'єктів, для яких заздалегідь відомо до якого з класів вони належать. Такі методи відносять до класу «навчання з учителем».

У задачі, що поставлена в даній роботі, необхідно виявити потенційних клієнтів, не володіючи інформацією до якого класу належать ті чи інші спостереження.

Кластеризація, на відміну від класифікації, не вимагає наявності інформації про належність до класу і відповідно відноситься до класу «навчання без учителя».

Завдання кластеризації вирішується на початкових етапах дослідження. Її рішення допомагає краще зрозуміти дані, їх природу[8].

Важливою особливістю кластерного аналізу є те, що він дозволяє проводити розбиття об'єктів не по одному, а по цілому набору ознак.

У нашій задачі, необхідно виявити потенційних власників мобільних пристроїв серед клієнтів мережістільникового зв'язку, не володіючи інформацією до якого класу належать ті чи інші спостереження.

Тому в нашому випадку доцільно використовувати на різних етапах аналізу даних різні методи (кластеризації і класифікації), що дозволить

- автоматично визначити оптимальну кількість кластерів;
- працювати з великими масивами даних;
- не потрібна навчальна вибірка об'єктів, для яких заздалегідь відомо до якого з класів вони належать.

Технологія обробки даних

Крок 1. Першим етапом роботи алгоритму є введення потрібних для роботи даних: інформації про мобільні пристрої клієнтів телефонної мережі та місцезнаходження абонентів в конкретний проміжок часу та перевірка отриманих даних на коректність і повноту, а також приведення їх до формату, прийнятого для подальшої обробки.

Дані вводяться у систему як матриці — відбувається табуляція даних, де в стовпцях містяться параметри мобільних пристроїв або атрибути абонентів, а в рядках — значення для конкретного мобільного пристрою або клієнта мобільної мережі. Введення даних і їх табуляція може здійснюватися вручну або ж за допомогою спеціальних комп'ютерних програм. На цьому етапі уможливорюється сортування, форматування, відбір та інші операції оброблення даних.

Введення даних є процесом трудомістким і часто вимагає залучення спеціально підготовлених операторів, тому доцільно визначитися з оптимальною (а краще — мінімальною) кількістю необхідних даних для запобігання труднощам під час аналізу отриманих даних. Проблеми, які виникають під час введення даних, складніше виявляти і попереджувати, коли вручну доводиться вводити сотні тисяч рядків текстової інформації. Проте їх можна вирішити, якщо автоматизувати процес введення даних [9].

Крок 2. Після перевірки даних на коректність і приведення їх до стандартного формату виконується виділення лише вагомих для роботи алгоритму та виконання аналізу характеристик мобільних пристроїв. Зокрема, такими характеристиками можуть виступати унікальні параметри для кожного мобільного пристрою — IMEI та IMSI. Також серед характеристик мобільних пристроїв, для коректної роботи алгоритму, обов'язковою є наявність моменту часу в який було знято параметри даного пристрою.

IMEI (International Mobile Equipment Identity) — серійний номер мобільного пристрою (п'ятнадцятизначне число, тобто 14 цифр коду плюс 15-та

контрольна цифра), який встановлюється заводом-виробником та є унікальним для кожного мобільного телефону[10].

IMSI (InternationalMobileSubscriberIdentity) — міжнародний ідентифікатор користувача мобільного зв'язку. Це унікальний номер, що ставиться у відповідність будь-якому користувачеві мобільного телефона. Цей номер зберігається у модулі ідентифікації користувача (SubscriberIdentityModule, SIM). IMSI відсилається мобільним терміналом та використовується для отримання інформації про абонента з бази даних HLR (HomeLocationRegister)[11].

Крок 3. Виконується групування клієнтів за їх місцезнаходженням конкретний проміжок часу.

Крок 4. Здійснюється кластеризація клієнтів на основі унікальних параметрів мобільних пристроїв —IMEI, IMSI, але при цьому розглядаються окремо кожна з груп клієнтів мобільної мережі, які у вказаний період часу мали спільне місцеперебування.

Крок 5. Здійснюється класифікація клієнтів на основі унікальних параметрів мобільних пристроїв – IMEI, IMSI для всіх абонентів мобільної мережі лише з урахування класів, які були сформовані при виконанні кроку 4.

Крок 6. У кожному класі виконується видалення дублікатів абонентів.

Крок 7. Проводиться вибірка тих класів, які містять клієнтів, що знаходилися в заданий період часу в одному місці. Виконується обробка, яка передбачає розробку висновків та гіпотез, проводиться перевірка репрезентативності результатів, їх здатності до перенесення на всю генеральну сукупність.

Крок 8. Після виконання зазначених вище кроків отримується певна вибірка, що містить дані користувачами мобільної мережі, і як результат виводяться потрібні власники мобільних пристроїв.

Висновок

У даній роботі була запропонована ідея реалізації системи ідентифікації клієнтів мобільної мережі на основі аналізу даних по мобільним пристроях та місцезнаходженню абонентів у конкретний проміжок часу. Було розроблено базовий алгоритм для роботи системи, який на відміну від існуючих може працювати без початкової вибірки і давати більш точний результат за рахунок ретельної фільтрації даних.

У подальшому планується проведення робіт по адаптації розробленого алгоритму для вирішення більш широкого кола задач і підвищення

швидкості роботи алгоритму за рахунок введення більш скрупульозних перевірок на першому етапі роботи — етапі введення даних.

Система поступово реалізується програмно і впроваджується для використання Службою Безпеки України.

Література

1. Журавл'єв Ю.І., Рязанов В.В., Сенько О.В. Распознавание. Математические методы. Программная система. Практические применения. — М.: Изд. «Фазис», 2006. — 176 с.
2. Пятецький-Шапиро Г. DataMining и перегрузка информацией // Вступительная статья к книге: Анализ данных и процессов / А. А. Барсегян, М. С. Куприянов, И. И. Холод, М. Д. Тесс, С. И. Елизаров. 3-е изд. перераб. и доп. СПб.: БХВ-Петербург.: — 2009. — 512 с.
3. Дюк В., Самойленко А. DataMining: учебный курс (+CD). — СПб.: Изд. Питер, 2001. — 368 с.
4. Gellman B., Miller G. (August 29, 2013). "U.S. spy network's successes, failures and objectives detailed in 'black budget' summary". The Washington Post. p. 3. Retrieved August 29, 2013.
5. Prism program taps in to user data of Apple, Google and others. The Guardian. 2013-06-07. Процитовано 11 червня 2013.(англ.)
6. Intelligence chief blasts NSA leaks, declassifies some details about phone program limits. The Washington Post. 2013-06-07. Архів оригіналу за 2013-06-15. Процитовано 11 червня 2013.(англ.)
7. Vamford J. The Shadow Factory: The Ultra-Secret NSA from 9/11 to the Eavesdropping on America. — Doubleday, 2008. — ISBN 0-38-5521-32-4.
8. Анализ клиентских баз данных. Выявление мошенничества (fraud-detection) на базе STATISTICA DataMiner. [Електронний ресурс]. — Режим доступу: <http://www.statsoft.ru/solutions/ExamplesBase/tasks/>
9. Дистанційний курс Тараса Григорчука. [Електронний ресурс]. — Режим доступу: <https://sites.google.com/site/marketingdistance/>
10. 3GPP TS 22.016: International Mobile Equipment Identities (IMEI) (ZIP/DOC; 36 KB). 2009-10-01. Retrieved 2009-12-03.
Режим доступу:
http://www.3gpp.org/ftp/Specs/archive/22_series/22.016/22016-900.zip
11. 3GPP specification: International Mobile Subscriber Identity 23.003.
Режим доступу: <http://www.mib.net.ua/2008/03/gsm-numbering-plans-en.html>

УДК 681.5.013

Моделі штучних нейронних мереж при їх апаратно-програмній реалізації на FPGA

*Кравець П.І., к.т.н., доц., Шимкович В.М., Зубенко Г.А.
Національний технічний університет України «КПІ», м. Київ
kravets@acts.kiev.ua*

В роботі розроблені математичні моделі штучного нейрону та штучних нейронних мереж прямого розповсюдження при їх апаратно-програмній реалізації на програмованих логічних інтегральних схемах за розробленим раніше методом, наведені графіки що відображають адекватність розроблених моделей.

Розвиток теорії автоматичного управління, як і розвиток будь-якого іншого напрямку науки, характеризується ускладненням розв'язуваних завдань і підвищенням якісних показників необхідних рішень [1]. Традиційні методи управління в основному спираються на теорію лінійних систем, в той час як реальні об'єкти є, за своєю природою, нелінійними. Нейромережеві системи управління являють собою новий високотехнологічний напрямок в теорії управління та відносяться до класу нелінійних динамічних систем [2]. Висока швидкодія за рахунок розпаралелювання вхідної інформації в поєднанні зі здатністю до навчання нейронних мереж робить цю технологію вельми привабливою для створення пристроїв управління в автоматичних системах.

На даний час основним методом реалізації нейромережевих систем управління є програмний, з використанням комп'ютерної техніки чи спеціалізованих контролерів, побудованих на її основі, що значно звужує коло практичної реалізації систем управління із-за значної вартості таких регуляторів і робить їх практично недоцільними та недоступними для використання в простих системах керування, крім того вони мають обмежену швидкодію та потребують значних затрат часу на навчання. Рекурентність і послідовність дій процедури навчання нейромережі при її реалізації на всій множині налагоджуваних параметрів не дозволяє повністю вирішити проблему швидкодії процедури навчання нейромережевих структур в темпі з динамікою об'єкта управління. Єдиною альтернативою цьому є розпаралелення процедури навчання та роботи внутрішніх елементів нейромережевих структур. Такі можливості з'являються при апаратно-програмній реалізації нейромережевих структур побудованих на нейрочіпах чи програмованих логічних інтегрованих схемах (ПЛІС-FPGA) [3].

Засоби реалізації нейромережевих систем управління повинні орієнтуватися на широке застосування в промислових умовах, бути універсальними і гнучкими, навчатися і адаптуватися в реальному часі, бути простими і дешевими, тому найбільш перспективними засобами можна вважати FPGA [3,4]. З появою FPGA проектування цифрових мікросхем перестало бути долею виключно великих підприємств з обсягами випуску в десятки і сотні тисяч кристалів. Проектування і випуск невеликої партії унікальних цифрових пристроїв став можливий в умовах проектно-конструкторських підрозділів промислових підприємств, в дослідницьких і навчальних лабораторіях і навіть в умовах домашніх радіоаматорських місць. Промислово випускаються «заготовки» програмованих мікросхем з електричним програмуванням і автоматизованим процесом перекладу схеми користувача в послідовність імпульсів програмування роблять проектування нових цифрових пристроїв порівнянним з розробкою програмного забезпечення [5-7].

Метою даної роботи є розробка математичних моделей штучних нейронних мереж при апаратно-програмній реалізації нейронних мереж на FPGA для синтезу нейромережевих систем управління.

При апаратно-програмній реалізації штучного нейрону та штучних нейронних мереж на FPGA використовуються метод, розроблений нами раніше та описаний в [8]. Даний метод розроблений для реалізації штучного нейрону з сигмоїдальною функцією активації, а саме:

$$f(x) = \frac{1}{1 + e^{-x}} \quad (1)$$

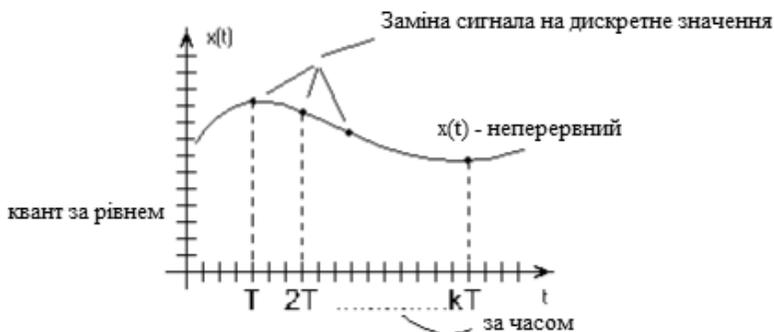


Рис.1. Квантування неперервного сигналу за часом і рівнем

В методі апаратно-програмної реалізації використовується принцип квантування неперервного сигналу за часом та рівнем. На рис. 1 T – крок квантування за часом, $x(t)$ – неперервний сигнал, що квантується.

Згідно [8] вхідне значення нейрона:

$$lin = \frac{\left(\sum_{i=1}^n in_i \cdot W_i \right)}{128} \quad (2)$$

тобто:

$$t = \frac{\sum_{i=1}^n w_i r_i}{128} \quad (3)$$

Якщо $x(t)$ – сигмоїда, то згідно [8] вихідне значення нейрона дорівнює:

$$f(x) = 0.5 + 0.01N \quad (4)$$

де N – значення, що визначається з таблиці, відповідно до lin (з формули (2)). Тобто $N = k(t)$, де t – значення з формули (3). $k(t)$ – функція, значення якої розраховується за таблицею і являє собою квантоване за рівнем значення сигмоїдальної функції. Область значень функції k – множина цілих чисел. 0.01 у формулі (4) є періодом квантування за часом, тоді у загальному випадку (4) прийме вигляд

$$f(t) = 0.5 + T \cdot k(t) \quad (5)$$

де T – період квантування за часом. Область значень функції $f(x)$ – множина дійсних чисел.

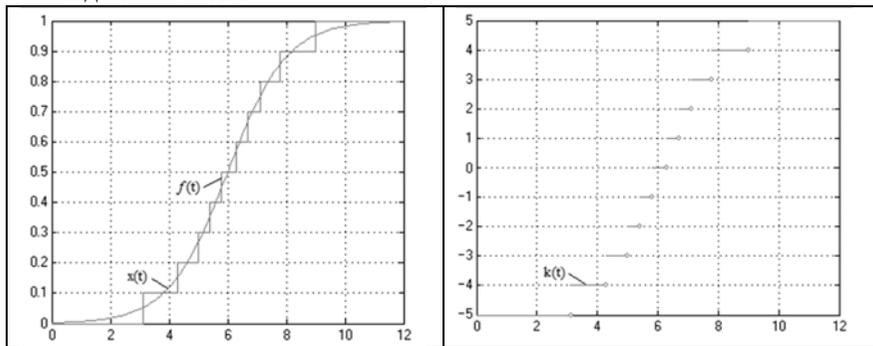


Рис.2. Функції $f(t)$, $k(t)$, $x(t)$ при $T = 0.1$

Згідно вищеописаного, математична модель штучного нейрону при апаратно-програмній реалізації на FPGA приймає вигляд:

$$q = f(r) = 0.5 + 0.01k \left(\frac{\sum_{i=1}^n w_i r_i}{128} \right) = f(t) = 0.5 + Tk(t) \quad (6)$$

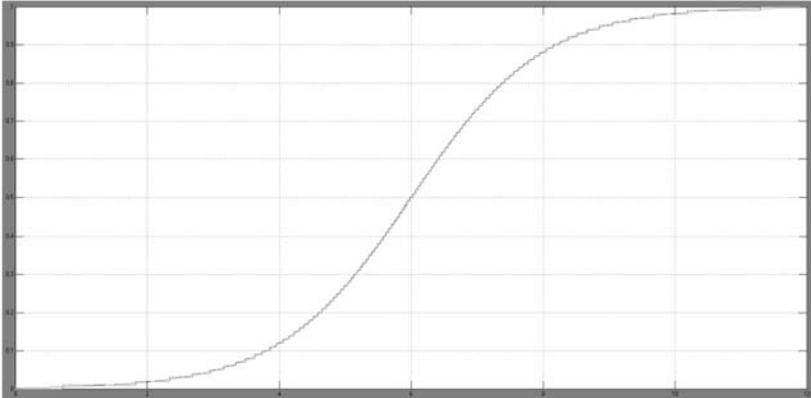


Рис.3. Сигмоїдальна функція

На рис. 3 зображений графік за формулою (1) та (6) з кроком квантування 0.01, як видно модель є адекватною, похибка не перевищує крок квантування.

Розглянемо статичну багатшарову нейронну мережу, структура і математична модель якої описана в [2], вихід i -го елемента в l -му шарі визначається за формулою:

$$q_i^{(l)} = f \left(\sum_{j=1}^{n^{(l-1)}} w_{i,j}^{(l)} \cdot q_j^{(l-1)} + w_{i,0}^{(l)} \cdot q_0^{(l-1)} \right) \quad (7)$$

Враховуючи (6), (7) прийме вигляд:

$$q_i^{(l)} = 0.5 + 0.01k \left(\frac{\sum_{j=1}^{n^{(l-1)}} w_{i,j}^{(l)} \cdot q_j^{(l-1)} + w_{i,0}^{(l)} \cdot q_0^{(l-1)}}{128} \right) = f(t) = 0.5 + Tk(t_i^{(l)}) \quad (8)$$

$$\text{де } t_i^{(l)} = \frac{\sum_{j=1}^{n^{(l-1)}} w_{i,j}^{(l)} \cdot q_j^{(l-1)} + w_{i,0}^{(l)} \cdot q_0^{(l-1)}}{128} .$$

Формула (8) є математичною моделлю статичної багатозарової нейронної мережі при її реалізації на FPGA.

Висновок

В даній роботі розроблені та описані математичні моделі штучного нейрону та штучних нейронних мереж при їх реалізації на FPGA. Моделі є адекватними, похибка не перевищує крок квантування. Розроблений метод та математичні моделі мають змінний крок квантування, що дозволить адаптувати їх до конкретних задач з відповідною точністю та дозволить скоротити займаний ресурс в FPGA.

Література

1. *Егунов Н.Д.* Методы робастного, нейро-нечеткого и адаптивного управления. – М.: Изд-во МГТУ им. Н.Э. Баумана, 2001. – 744 с.
2. *Терехов В.А.* Нейросетевые системы Управления: Учеб. пособие для вузов. – М: Высшая школа. 2002. -183с.
3. *Соловьев В.* Проектирование цифровых систем на основе ПЛИС. – М.: Радио и связь, 2003. – 376с.
4. *Гильгурт С.Я.* Анализ применения реконфигурируемых вычислителей на базе ПЛИС для реализации нейронных сетей // Моделирование та інформаційні технології. Зб. наук. пр. ІПМЕ НАН України. – Вип. 37. – Київ: 2006. – С. 168-174 с.
5. *Логовский А.* Технология ПЛИС и ее применение для создания нейрочипов. // Открытые системы. – 2000. – № 4. – С. 100-102 с.
6. *Тарасов И.Е.* Разработка цифровых устройств на основе ПЛИС Xilinx® с применением языка VHDL. – М.: Горячая линия – Телеком, 2005. – 252 с.
7. *Сергиенко А.М.* VHDL для проектирования вычислительных устройств – К ЧП «Корнейчук», «ТИД «ДС», 2003. – 208 с.
8. *Кравець П.І., Шимкович В.М., Зубенко Г.А.* Технологія апаратно-програмної реалізації штучного нейрона та штучних нейронних мереж засобами FPGA / Вісник НТУУ «КПІ». Інформатика, управління та обчислювальна техніка: Зб. наук. пр. – К.: Век+, – 2012. – №55. 174-180с.
9. *Туманов М.П.* Теория импульсных, дискретных и нелинейных САУ: Учебное пособие. – МГИЭМ. М., 2005, 63 с.

УДК 681.3: 519.68

Межкластерная многопутевая маршрутизация в сложной сети

Кулаков Ю. А., д.т.н., проф., Национальный технический университет Украины «КПИ», г. Киев, ya.kulakov@gmail.com;

Воротников В. В., к.т.н., доц.; Данелюк Д.О.,

Житомирский военный институт им. С.П. Королева Государственного университета телекоммуникаций, г. Житомир, vvorotnik@ukr.net, donatos_premium@ukr.net

Работа посвящена вопросам управления в сложных сетях с применением новых методов теории перколяции и фрактальной геометрии. Для перколяционного кластера сложной сети приведен пример использования LoCoop-многопутевой маршрутизации. Показано, что использование слабо пересекающихся маршрутов позволяет повысить надежность сети для случая использования ненадежных соединений, которые имеют место в беспроводных технологиях. Доказано, что добавление к множеству путей многопутевого маршрута любого дополнительного слабосвязанного резервного пересекающегося пути, с одной стороны повышает надежность сети, с другой – увеличивает среднюю задержку маршрута передачи данных. Поэтому, задача выбора количества резервных путей является оптимизационной и решается известными методами.

Введение

Одним из перспективных направлений моделирования сложных сетей является использование метода мультифрактальной динамики. В настоящее время разработаны фрактальные подходы для анализа архитектуры таких сетей [1].

• Основная концепция формирования такой структуры получила название парадигмы TMN (Telecommunication Management Network) – сети управления телекоммуникациями. Концепция TMN поддерживает три уровня управления:

• управление сервисом сети (обеспечение предоставления пользователям необходимых услуг с гарантированным качеством);

• управление сетью непосредственно (управление топологией, осуществление маршрутизации потоков информации и управление перегрузками в сети);

- управление элементами сети (контроль состояния и управление линиями связи и узловых пунктов, своевременное переключение на резервное оборудование и выявление аварийной ситуации).

- Разнообразие аппаратуры, среды передачи, телекоммуникационных технологий обуславливают большое количество возможных реализаций сегментов сети и интерфейсов между ними. В частности, в сложных информационных системах в последнее время широкого распространения получили сегменты, построенные на основе беспроводных mesh технологий.

В такой ситуации актуальным является вопрос повышения надежности маршрута передачи данных в сложной сети за счет использования многопутевой маршрутизации.

Обзор последних исследований и публикаций

В беспроводных ad hoc сетях реализуются механизмы маршрутизации и борьбы с задержками отличные от механизмов проводных сетей. В проводных сетях маршруты имеют большую «продолжительность жизни», что используется в процедурах управления потоками; управление сетью обычно существенно менее затратное, чем управление данными, и время построения пути меньше из-за использования проактивных протоколов (например, OSPF[2]). В беспроводных ad hoc сетях из-за подвижности узлов и взаимной интерференции соединения непродолжительны, а поэтому непригодны для процедур управления потоками данных. Это связано с тем, что в протоколах маршрутизации реализуются слишком жесткие требования к выбору маршрутов: если количество утраченных пакетов за время передачи больше порогового значения, маршрут не используется. Время построения маршрута в ad hoc сети при помощи реактивного протокола больше и зависит от потерь пакетов и задержек в сети.

Так, в работе [3], посвященной управлению потоками в сетях под фиксированной маршрутизацией понимается использование единственного пути доставки сообщений от источника к адресату. Альтернативная маршрутизация (многопутевая) предполагает наличие нескольких путей. В работах по оптимальной (минимальной задержке) маршрутизации ограничиваются управлением потоками по непересекающимся (независимым) путям.

В работе [4] исследован статический протокол маршрутизации, который не перенаправляет потоки мимо неисправных участков. Показано, что в случае краткосрочного отказа узла или связи, ремаршрутизация не является необходимой при наличии нескольких путей.

Различают несколько типов многопутевых маршрутов по степени связности [4]:

1) непересекающиеся (несвязанные, независимые) пути, которые не пересекаются, не имеют ни общих транзитных узлов, ни общих транзитных ребер, кроме начального (источника) и конечного (адресата) узлов;

2) LoCouр-пересекающиеся (Low Coupling, слабосвязанные, «слабо-зависимые») пути, которые не имеют общих ребер, но могут иметь общие транзитные узлы;

3) зоннопересекающиеся (реберно-зависимые) пути, которые могут иметь общие транзитные ребра в пределах определенной зоны (или числа сегментов сети).

С другой стороны, в [5] показано, что стоимость поддержания двух непересекающихся путей при маршрутизации от источника в реактивном протоколе SMR, выше, чем построение двух путей. Поэтому, время необходимое для определения новых маршрутов после обнаружения отказов, может быть уменьшено за счет большей избыточности свободных от петель путей между источником и адресатом.

Целью работы является повышение надежности сети передачи данных в масштабно-инвариантной сети с использованием многопутевой маршрутизации, за счет использования резервных LoCouр-пересекающихся путей.

Изложение основного материала

Под «сложными сетями» понимаются системы, состоящие из реальных объектов и связей между ними. Сложная сеть моделируется графом, однако этот граф, как правило, имеет определенную структуру и обладает характерными признаками. Такие сети принято называть безмасштабными (scale-free), поскольку средняя степень вершины в них не является характерной, т.е. отсутствует характерный масштаб [2]. Для безмасштабной топологии характерно наличие малого числа хабов – вершин наибольшей степени – и большого числа вершин малой степени. Сложные сети имеют хорошо выраженную структуру естественных сообществ: вершины сети разделены по группам, которые слабо связаны между собой, но имеют большую плотность ребер внутри.

Для повышения надежности связи между кластерами сложной сети, реализованной по архитектуре mesh-соединений целесообразно иметь возможность многопутевой маршрутизации. Наличие альтернативных путей для реактивных методов маршрутизации позволяет сократить время, затраченное на построение нового маршрута, за счет переключения трафика в обход поврежденного участка (ребра).

Формулировка задачи исследования

Пусть перколяционный кластер [1,3] сети задан графом $G(V, E, d(e))$, где: V – множество узлов графа; E – множество ребер графа, $d(e)$ – вес пути (сумма весов ребер вдоль данного пути) от начального узла v_s до узла конечного v_d ; $w(v_s, v_d)$ – вес ребра между v_s -м и v_d -м узлами.

Для графа G определено множество возможных маршрутов $\Pi = \{\pi_1, \dots, \pi_N\}$ между узлами v_s, v_d .

Необходимо: из множества всех путей $\Pi = \{\pi_1, \dots, \pi_N\}$ сформировать следующие два множества путей и сравнить эффективность их применения для многопутевой маршрутизации:

1) множество $\Pi_{disj}^{s,d} \in \Pi$, состоящее из k -альтернативных непересекающихся путей π_i между узлом-источником v_s и узлом-адресатом v_d :

$$\Pi_{disj}^{s,d} = \{\pi_i\}, \pi_i = \{V_s, \{\overline{V} \pi_i\}, V_d\}, \quad (1)$$

где $i, j = \overline{1, k}$, $i \neq j$, k – количество альтернативных путей; $\overline{V} \pi_i$ – вектор транзитных узлов пути π_i мощностью $l-2$, l – количество узлов в маршруте.

Непересекающиеся многопутевые маршруты [6] должны удовлетворять условию:

$$\forall \pi_i: \pi_i \cap \pi_j = \{V_s, V_d\}. \quad (2)$$

2) множество LoCoop-пересекающихся путей $\Pi_{mesh}^{s,d} \in \Pi$, удовлетворяющее условию:

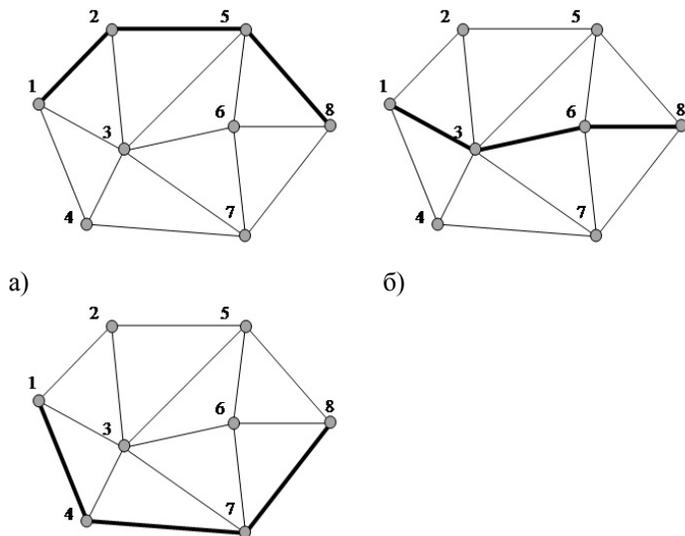
$$\forall \pi_i \in \Pi_{mesh}^{s,d}: (\pi_i \cap \pi_j) / \{v_s, v_d\} \neq \emptyset \quad (3)$$

Эффективность использования $\Pi_{disj}^{s,d}$ и $\Pi_{mesh}^{s,d}$ для многопутевой маршрутизации оценим при помощи полинома надежности [5,6].

Пример. Пусть задана mesh сеть G , состоящая из 8 узлов (рис. 1).

Оценим эффективность многопутевой маршрутизации из узла 1 в узел 8 при помощи использования двухузловых непересекающихся и резервных LoCoop-пересекающихся путей.

Определим множество непересекающихся путей многопутевого маршрута $\Pi_{disj}^{1,8} = \{\pi_1, \pi_2, \pi_3\}$ (рис. 1). В него попадают три пути:



в)

Рис. 1. Непересекающиеся пути многопутевого маршрута:

а) - $\pi_1 = \{1,2,5,8\}$; б) - $\pi_2 = \{1,3,6,8\}$; в) - $\pi_3 = \{1,4,7,8\}$

Введем ограничение, что все ребра графа равнонадежны, т.е. вероятность функционирования каждого ребра равна p .

Пусть каждое соединение имеет случайное независимое значение «времени жизни» L , с $\lambda = 1/L$ [4]. Тогда совместная функция распределения времени формирования маршрута определяется как $P(E_i) = 1 - \exp(-n\lambda t)$, где n - число прямых связей в E_i (событие работоспособности i -ого пути).

Определим надежность каждого пути, используя метод включения [6]. Маршруты $\pi_1 = \{1,2,5,8\}$, $\pi_2 = \{1,3,6,8\}$, $\pi_3 = \{1,4,7,8\}$ состоят из трех ребер и надежность каждого равна p^3 . Представим полином надежности многопутевого маршрута, состоящий из трех непересекающихся путей:

$$P_{disj}^{1,8}(\pi_1, \pi_2, \pi_3) = P_{disj}^{1,8}(\pi_1, \pi_2) + P_{disj}^{1,8}(\pi_3), \text{ где}$$

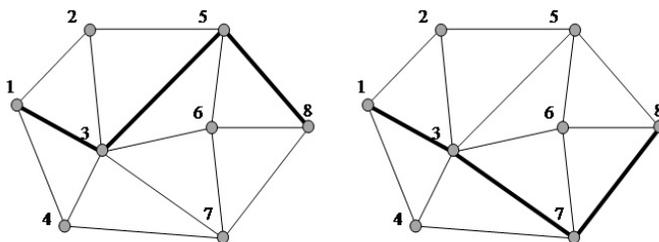
$$P_{disj}^{1,8}(\pi_1, \pi_2) = P(\pi_1) + P(\pi_2) - P(\pi_1) * P(\pi_2). \text{ При условии}$$

$$P(\pi_1) = P(\pi_2) = P(\pi_3) = p^3, \quad P_{disj}^{1,8}(\pi_1, \pi_2) = 2p^3 - p^6,$$

$$P_{disj}^{1,8}(\pi_1, \pi_2, \pi_3) = P_{disj}^{1,8}(G, p) = 3p^3 - p^6 - p^9.$$

Для того же графа определим полином надежности многопутевого маршрута, состоящего из LoCoop-пересекающихся путей. В данном примере ограничимся рассмотрением путей без петель, состоящих не более чем из трех ребер; резервные пути не должны иметь с основными непересекающимися путями более одного общего узла.

Множество трехреберных путей многопутевого маршрута $\Pi_{mesh}^{1,8}$, при условии $\Pi_{disj}^{1,8} \in \Pi_{mesh}^{1,8}$, дополнительно будет состоять из двух следующих резервных путей (рис. 2).



а) б)

Рис. 2. LoCoop-пересекающиеся пути многопутевого маршрута:

а) - $\pi_4 = \{1,3,5,8\}$; б) - $\pi_5 = \{1,3,7,8\}$

При расширении многопутевого маршрута двумя трехреберными LoCoop-пересекающимися путями полином надежности будет иметь вид:

$$P_{mesh}^{1,8}(G, p) = P_{mesh}^{1,8}(\pi_1, \pi_2, \pi_3, \pi_4, \pi_5) = 5p^3 - 9p^6 + 7p^9 + 2p^{14}$$

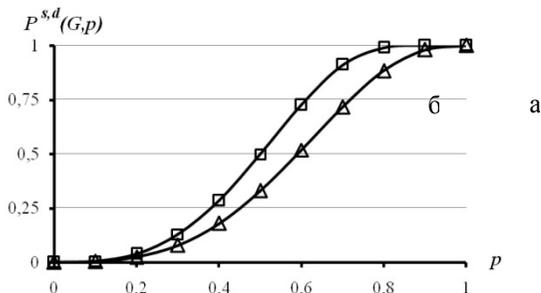


Рис. 3. Зависимость надежности многопутевого маршрута от p при использовании: а) трех непересекающихся путей;

б) пяти LoCoop-пересекающихся путей

На рис. 3 показана зависимость надежности непересекающейся (disjoint) и петлевой (mesh) конфигураций сети G . Как и ожидалось, mesh сеть имеет большее значение надежности.

Заключение

Классическая многопутевая маршрутизация использует для построения маршрута только непересекающиеся пути. Однако их нахождение и поддержание в актуальном состоянии в условиях динамического изменения топологии сети является сложной задачей. Поэтому использование резервных LoCoop-пересекающихся маршрутов, не требующих чрезмерных дополнительных затрат на поддержание их актуальности, по сравнению с непересекающимися, позволяет повысить как надежность mesh сети так и ее пропускную способность.

Полученные результаты свидетельствуют о преимуществах узлово-пересекающейся конфигурации mesh сети по сравнению с чисто непересекающейся конфигурацией, особенно для ненадежных соединений.

Литература

1. *Тарасевич Ю.Ю.* Перколяция: теория, приложения, алгоритмы. – Ю.Ю. Тарасевич, – М.: УРСС, 2002. – 109 с.
2. *Додонов А.Г.* Живучесть информационных систем / А.Г. Додонов, Д.В. Ландэ. – Киев: Наукова думка, 2011. – 256 с.
3. *Кулаков Ю.А., Воротников В.В.* Энергоэффективная иерархическая маршрутизация в самоорганизующихся динамических сетях. Управляющие системы и машины: междунар. науч. журн.–2014.–№1.– С 70-76.
4. *Marc Mosko, J.J. Garcia-Luna-Aceves.* Multipath Routing in Wireless Mesh Networks. 1st IEEE Workshop on Wireless Mesh Networks, IEEE WiMesh 2005, Held in conjunction with IEEE SECON 2005, 26 September 2005, Santa Clara, California USA, pp.64-70.
5. *K. Valarmathi, Dr.N. Malmurugan* Reliable multipath routing for 802.16 wireless mesh networks - Indian Journal of Computer Science and Engineering Vol. 2 No. 6 Dec 2011-Jan 2012 pp. 966-976.
6. *Кулаков Ю.А., Воротников В.В.* Многопутевая маршрутизация в mesh сетях с использованием резервных слабосвязанных путей. Системы управления и информационные технологии: междунар. науч.-техн. журн.– 2014.– №1(85). – С. 68–77.

УДК 681.3

К оценке качества базового словаря в системе автоматического обнаружения ошибок пользователя и возможностей его улучшения

*Литвинов В.А., д.т.н., проф., Майстренко С. Я., к.т.н.
Институт проблем математических машин и систем НАНУ, г. Киев
maistrsv@ukr.net*

Разработана имитационная модель искажений слов и обнаружения ошибок пользователя. Приводятся результаты моделирования для словарей русского и украинского языков. Полученные данные иллюстрируют возможную результативность решения задачи оптимизации словаря допустимых слов с учетом частот использования слов.

Вступление

Основой систем автоматического обнаружения ошибок пользователя в естественно-языковых словах (в общем случае - нерегулярных алфавитно-цифровых кодах) является базовый словарь БС допустимых («правильных») слов. Качество БС в первую очередь определяется двумя факторами:

- способностью обнаруживать как можно больше из наиболее вероятных (типовых) ошибок;
- способностью выдавать как можно меньше ложных сообщений об ошибке.

Первый фактор может быть оценен относительным количеством ρ слов БС, искаженных определенными ошибками и совпавших с реальными допустимыми словами.

Второй фактор оценим суммарной вероятностью π отсутствия требуемого слова в БС.

Словарь естественно-языковых слов и типовые ошибки пользователя имеют нерегулярную структуру, поэтому здесь упрощенные оценки, основанные на соотношениях мощностей множеств разрешенных (N) и всевозможных (q^n) комбинаций n символов в алфавите q практически неприемлемы [1].

Доклад посвящен вопросам экспериментальной оценки указанных факторов для некоторых словарей русского и украинского языков. Отдельные примеры оценки контролирурующих свойств БС были приведены в [1, 2].

Результаты имитационного моделирования

В табл.1 приведены результаты полномасштабного моделирования трех словарей русского языка [3, 4] и адаптированных украинских версий указанных словарей, сформированных путем русско-украинской конвертации, с целью оценки соответствующих значений ρ .

Таблица 1. Результаты сравнения словарей различного объема

k	P_k	СЗр	СЛр	СРЛр	СЗу	СЛу	СРЛу
1	0.56	0.39	0.41	1.20	0.28	0.28	1.00
2	0.16	0.06	0.07	0.27	0.04	0.04	0.15
3	0.12	2.14	2.16	8.80	1.39	1.40	5.20
4	0.06	0.95	1.55	1.20	0.91	1.22	1.10
\bar{P}_k	0.90	0.54	0.6	1.84	0.38	0.41	0.77

Исследованы следующие основные классы типовых ошибок тайпинга: однократные транскрипции ($k = 1$), добавление символа ($k = 2$), выпадение символа ($k = 3$), транспозиция соседних символов ($k = 4$).

Для ориентировочных оценок вероятностей P_k ошибок этих классов взяты соответствующие значения, приведенные в [5].

Приняты следующие обозначения для исследованных словарей:

СЗр – «Словарь Зализняка» оригинальный, $N = 92555, \bar{n} = 9.61$;

СЛр – «Словарь Лопатина» оригинальный, $N = 150213, \bar{n} = 10.06$;

СРЛр – «Словарь русской литературы» оригинальный, $N = 161730, \bar{n} = 8.44$.

СЗу – украиноязычная версия СЗр, $N = 84575, \bar{n} = 9.49$;

СЛу – украиноязычная версия СЛр, $N = 135401, \bar{n} = 9.93$;

СРЛу – украиноязычная версия СРЛр, $N = 129244, \bar{n} = 8.31$.

Как видно из данных табл.1, содержащей масштабированные (для большей наглядности) значения $\rho 10^2$, различные словари имеют заметно отличающиеся контролирующие свойства. Так из 1000 случайных

ошибочных слов словарей, искаженных ошибками 1, 2, 3, 4 (в указанной пропорции) в среднем не обнаруживаются 5.4 ошибки для Словаря Зализняка и 18.4 ошибки для Словаря русской литературы. Исходя из полученных данных, мы можем предположить, что разброс значений $\bar{\rho}_k$ для исследованных словарей определяется как чисто лингвистическими факторами (язык, структура), так и разницей в объемах. Последнее свойство отмечено и иллюстрировано примерами в [2].

Для ориентировочной оценки второго фактора (значения π в связи с варьируемым объемом словаря) примем во внимание следующее.

Приложение известного принципа Парето к описанию востребованности информационных ресурсов дает основание утверждать, что в любой информационной системе, как правило, наиболее активно используется сравнительно небольшая часть ресурсов. Так, например, большое количество пользователей Интернета наиболее часто посещает сравнительно небольшое количество сайтов. Убывающую функцию плотности распределения вероятностей обращения к элементам множества информационных ресурсов, обладающую подобными свойствами, удобно представить в виде непрерывной экспоненциальной функции

$$p(x) = c\lambda \exp(-\lambda x),$$

где c - нормирующий множитель.

При этом для соотношений Парето $\frac{20}{80}$ и $\frac{10}{90}$ значения λ оказываются примерно равными $\frac{8}{N}$ и $\frac{24}{N}$ соответственно.

В табл.2,3 в качестве примера сведены значения ρ, π для предварительно нормированных (по объемам) словарей СЗр,СЗу, СРЛр, СРЛу в зависимости от объема Δ их «усечения» путем исключения наименее востребованных слов; для исходных объемов ($N = 84575, \Delta = 0$) принято $\pi = 0$.

Таблица 2. Словарь Зализняка

$\Delta 10^{-3}$	$(N - \Delta) 10^{-3}$	СЗ _{р,норм}		СЗ _{у,норм}		$\pi^{(\Delta)} 10^4$ $\lambda = 8 / N$	$\pi^{(\Delta)} 10^9$ $\lambda = 24 / N$
		$\rho^{(\Delta)} 10^2$	$\rho^{(\Delta)} / \rho^{(N)}$	$\rho^{(\Delta)} 10^2$	$\rho^{(\Delta)} / \rho^{(N)}$		
0	84.57	0.50	1	0.39	1	0	0
5.08	79.50	0.48	0.96	0.37	0.95	2.1	0.12
10.16	74.42	0.45	0.90	0.35	0.90	5.4	0.63
15.24	69.35	0.42	0.84	0.33	0.86	10.8	2.80
20.32	64.27	0.40	0.80	0.31	0.81	19.5	11.96
...
N	0	0	0	0	0	10^4	10^9

Таблица 3. Словарь русской литературы

$\Delta 10^{-3}$	$(N - \Delta) 10^{-3}$	СРЛ _{р,норм}		СРЛ _{у,норм}		$\pi^{(\Delta)} 10^4$ $\lambda = 8 / N$	$\pi^{(\Delta)} 10^9$ $\lambda = 24 / N$
		$\rho^{(\Delta)} 10^2$	$\rho^{(\Delta)} / \rho^{(N)}$	$\rho^{(\Delta)} 10^2$	$\rho^{(\Delta)} / \rho^{(N)}$		
0	84.57	1.12	1	0.84	1	0	0
5.08	79.50	1.05	0.94	0.79	0.94	2.1	0.12
10.16	74.42	0.99	0.88	0.75	0.89	5.4	0.63
15.24	69.35	0.92	0.82	0.70	0.83	10.8	2.80
20.32	64.27	0.86	0.77	0.65	0.77	19.5	11.96
...
N	0	0	0	0	0	10^4	10^9

Выводы

Из приведенных данных видно, что уменьшение объема БС противоположным образом влияет на факторы качества. Приемлемость компромиссного парето-оптимального решения зависит как от абсолютных значений ρ , π , так и от их относительной значимости в конкретном случае. Поэтому моделирование, подобное проведенному выше, для реальных дискретных значений ρ_j конкретного словаря может дать полезную информацию для принятия обоснованных решений относительно задачи выбора словаря и его объема.

Что касается видимых из приведенных данных «преимуществ» украинскоязычных словарей относительно значений ρ (не имеющих, впрочем, практического значения, т.к. язык БС не является предметом выбора), то одно из возможных объяснений заключается в принятом способе их формирования путем русско-украинской конвертации. Авторам пока не удалось найти подходящих украинских словарей в свободном доступе в формате, подходящем для проведения соответствующих исследований.

Литература

1. *Литвинов В.А., Майстренко С.Я., Юденко О.П.* Экспериментальная оценка эффективности автоматического обнаружения типовых ошибок пользователя по словарям русского и украинского языков. // Міжнародна науково-технічна конференція «Системний аналіз та інформаційні технології» SAIT 2012 (System Analysis and Information Technologies SAIT 2012, April 24, 2012 Kyiv, Ukraine). С. 374.
2. *Литвинов В.А., Майстренко С.Я.* Контролирующая способность методов автоматического обнаружения типовых ошибок пользователя по словарям русского и украинского языков // Матеріали науково-практичної конференції з міжнародною участю «Системи підтримки прийняття рішень. Теорія і практика». – Київ. – 2013. – 3 червня. С.46-48.
3. Словари русского языка, <http://speakrus.ru/dict/>.
4. Словарь Лопатина, http://royallib.ru/book/lopatin_vladimir/russkiy_orfograficheskiy_slovar.html
5. *Литвинов В.А., Крамаренко В.В.* Контроль достоверности и восстановления информации в человеко-машинных системах.-Київ: Техніка, 1986.-200 с.

УДК 681.3: 519.68

Прогнозування волатильності акцій на основі гетероскедастичних моделей

*Олефір О. С., к. т. н., доцент, Гусак Б. І., студент,
Національний Технічний Університет України «КПІ»*

Abstract

Forecasting of stock volatility based on heteroscedastic models.

The main problem of the analysis and forecasting of the financial-economic factors on stock market is researched in this work. This problem is actual and complex since the nature of the processes is non-stationary and chaotic. Review of the facilities of the technical analysis is made in the work when working on stock market, as well as review of existing methods of building volatility forecasts.

Вступ

Курс акцій при виконанні торгівельних операцій на фондовій біржі характеризується складною неоднозначною поведінкою процесу, при аналізі якого дуже важливим являється складання найбільш вірогідного прогнозу поведінки цінних паперів. Прогнозування – це ключовий момент при прийнятті рішень в управлінні. Можливість передбачити некеровані аспекти подій перед прийняттям кінцевого рішення дозволяє зробити найкращий вибір, який в іншому випадку міг бути невдалим.

Робота присвячена дослідженню та прогнозуванню фінансово-економічних показників при роботі на фондовому ринку за допомогою теорії гетероскедастичних процесів. Ця задача є актуальною та складною у зв'язку із нестационарним та хаотичним характером процесів на ринку. В роботі виконано огляд засобів аналізу при роботі на фондовому ринку та існуючих методів прогнозування волатильності акцій. Для розв'язання поставленої задачі, намагатимемося спрогнозувати волатильність акцій на фондовій біржі за допомогою нейронних мереж, авторегресійної умовно гетероскедастичної моделі та авторегресійної моделі з ковзним середнім та екзогенною складовою.

Вибір методу прогнозування

Гетероскедастичні процеси відносяться до широкого класу нестационарних процесів, який включає в себе наступні процеси:

- процеси із детермінованими і стохастичними трендами;
- процеси із змінною дисперсією;
- процеси, які характеризуються змінним в часі математичним сподіванням та змінною дисперсією одночасно.

Дисперсію та стандартне відхилення часто використовують як міру ризику при дослідженні фінансово-економічних процесів, а тому цій проблемі приділяється значна увага в спеціальній літературі. При дослідженні фінансових процесів стандартне відхилення називають волатильністю (мінливістю) процесу.

Розглянемо узагальнену авторегресійну модель гетероскедастичного процесу (УАРУГ). При використанні такої моделі умовна дисперсія процесу представляється за допомогою моделі авторегресії з ковзним середнім (АРКС). Нехай похибки моделі описуються рівням:

$$\varepsilon(k) = v(k)\sqrt{h(k)}, (1)$$

де $v(k)$ – процес білого шуму з одиничною (для простоти) дисперсією $\sigma_v^2 = 1$;

$$h(k) = a_0 + \sum_{i=1}^q \alpha_i \varepsilon^2(k-i) + \sum_{i=1}^p \beta_i h(k-i) \quad (2)$$

Оскільки процес $\{v(k)\}$ визначено в даному випадку як білий шум, то умовне та безумовне середнє процесу будуть дорівнювати нулю.

Математичне сподівання для $\varepsilon(k)$:

$$E[\varepsilon(k)] = E\{v(k)\sqrt{h(k)}\} = 0 \quad (3)$$

Важливим моментом є те, що умовна дисперсія процесу $\{\varepsilon(k)\}$ є залежною від часу: $E_{k-1}[\varepsilon^2(k)] = h(k)$, оскільки $E_{k-1}[v^2(k)] = 1$. Для того щоб ця умовна дисперсія була скінченою, необхідно щоб корені характеристичного рівняння, записаного для (2) знаходились всередині кола одиничного радіуса. Таким чином, основою відмінною властивістю моделі УАРУГ є те, що умовна дисперсія збурень, які діють на процес $\{y(k)\}$, є процесом авторегресії з ковзним середнім.

Припустимо, що $\{y(k)\}$ – це процес авторегресії з ковзним середнім. При побудові моделі процесу в даному випадку можливі наступні варіанти:

- якщо вдається побудувати адекватну модель АРКС, то похибки моделі будуть мати властивості білого шуму;
- якщо не вдається побудувати адекватну модель АРКС, то, використовуючи автокореляційну функцію для квадратів залишків, необхідно побудувати

модель УАРУГ, яка дозволяє виконати аналіз поведінки дисперсії процесу; корелограма процесу $\{\varepsilon^2(k)\}$ дає можливість визначити гетероскедастичність.

Оскільки $E_{k-1}[\varepsilon(k)] = \sqrt{h(k)}$, то рівняння (2) можна записати у формі:

$$E_{k-1}[\varepsilon^2(k)] = a_0 + \sum_{i=1}^q \alpha_i \varepsilon^2(k-i) + \sum_{i=1}^p \beta_i h(k-i) \quad (4)$$

За своєю структурою це рівняння схоже на рівняння АРКС (q, p) для послідовності $\{\varepsilon^2(k)\}$.

Також ми розглядаємо метод групового урахування аргументів [4].

Цей метод використовує ідеї самоорганізації і механізми живої природи – схрещування (гібридизацію) і селекцію (добір).

Нехай є вибірка з N спостережень вхідних векторів $X(i)$ та вихідних $Y(i)$:

$$\begin{aligned} &\{X(1) \quad Y(1)\} \\ &\{X(2) \quad Y(2)\} \\ &\dots\dots\dots \\ &\{X(N) \quad Y(N)\} \end{aligned}$$

За результатами спостережень треба визначити $F(x)$. Причому навіть структура моделі $F(x)$ невідома.

Найбільш повна залежність між входами $X(i)$ і виходами $Y(i)$ може бути представлена за допомогою узагальненого полінома Колмогорова-Габора.

Переваги:

- можна відновити невідому довільно складну залежність по обмеженій вибірці. Число невідомих параметрів моделі може бути більше, ніж число точок навчальної послідовності.
- можливість адаптації параметрів моделі при одержанні нових даних експериментів.

Також було досліджено каскадну нейронну мережу на ортонеїронах.

Архітектуратакої мережі наведено на рис. 1.

Представлена конструкція реалізує відображення вигляду:

$$\widehat{y}_1 = \sum_{i=1}^n \sum_{j=0}^h \omega^{[1]}_{ji} \varphi_{ji}(x_i) \quad (5)$$

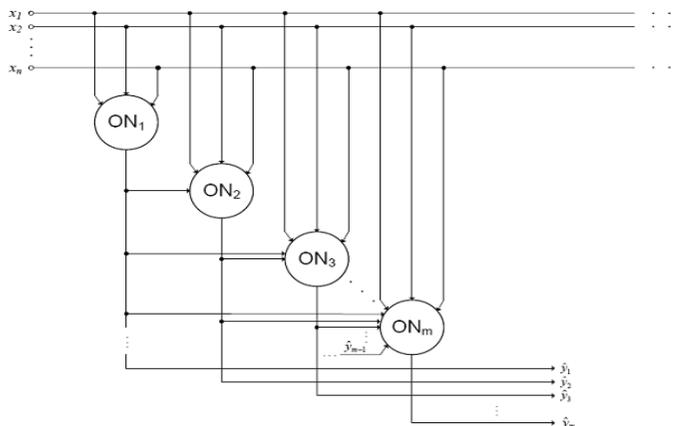


Рис. 1 Каскадна неймережа

- нейрон m -го каскада обчислюється за наступною формулою:

$$\widehat{y}_m = \sum_{i=1}^n \sum_{j=0}^h \omega_{ji}^{[m]} \varphi_{ji}(x_i) + \sum_{i=n+1}^{n+m-1} \sum_{j=0}^h \omega_{ji}^{[m]} \varphi_{ji}(\widehat{y}_{i-n}), \quad (6)$$

де \widehat{y}_m - вихід m -го каскаду неймережі; $\omega_{ji}^{[m]}$ - j -й синаптичний ваговий

коэффициент в i -му орто-синапсі орто-нейрона m -го каскаду; φ_{ji} - ортогональний поліном i -го порядку в j -му орто-синапсі; h - кількість ортогональних поліномів в орто-синапсах, по яким відбувається розкладання вхідних сигналів; n - кількість орто-синапсів в орто-нейроні першого каскаду; $(n + m - 1)$ - кількість орто-синапсів в орто-нейроні m -го каскаду.

Каскадна ортогональна неймережа складається з $(h + 1)(n + \sum_{l=n+1}^{n+m-1} l)$ параметрів, що настраюються, що дуже важливо, всі вони лінійно входять до висп-го каскада (6).

Вводячи лінійні вектори ортогональних поліномів $(h + 1)(n + m - 1) \times 1$ m -го орто-нейрона:

$$\varphi^{[m]} = (\varphi_{01}(x_1), \varphi_{11}(x_1), \dots, \varphi_{h1}(x_1), \varphi_{02}(x_2), \dots, \varphi_{h2}(x_2), \dots, \varphi_{ji}(x_1), \dots, \varphi_{hn}(x_n), \varphi_{0,n+1}(\widehat{y}_1), \dots, \varphi_{h,n+1}(\widehat{y}_1), \dots, \varphi_{h,n+m-1}(\widehat{y}_{m-1}))^T$$

і відповідний йому вектор синаптичних ваг $\omega^{[m]} = (\omega_{01}^{[m]}, \omega_{11}^{[m]}, \dots, \omega_{h1}^{[m]}, \omega_{02}^{[m]}, \dots, \omega_{h2}^{[m]}, \dots, \omega_{ij}^{[m]}, \dots, \omega_{hn}^{[m]}, \omega_{0,n+1}^{[m]},$

$\dots, \omega_{h,n+1}^{[m]}, \dots, \omega_{h,n+m-1}^{[m]})^T$ тієї ж розмірності, можна представити (6) у більш компактній формі:

$$\hat{y}_m = \omega^{[m]T} \varphi^{[m]} \quad (7)$$

Узагальнений алгоритм прогнозування

Крок 1. Зробити попередню обробку експериментальних даних (нормування, логарифмування, заповнення пропусків даних) і застосувати до них тести на гетероскедастичність. Якщо процес містить тренд, то перед побудовою моделі необхідно його видалити. Для підвищення надійності тестування необхідно застосувати не менше двох тестів. Зокрема, досвід побудови моделей свідчить, що, необхідно користуватись спрощеним тестом, який є досить наглядним і відносно простим. Досить часто візуальний аналіз дозволяє отримати суттєву інформацію щодо присутності гетероскедастичності. Разом з візуальним аналізом корисно розглядати параметри описової характеристики, які полегшують визначення структури моделі.

Крок 2. Користуючись автокореляційною функцією та частково автокореляційною функцією для експериментальних даних побудувати модель АР(p) або АРКС(p, q) для процесу $\{y(k)\}$ та обчислити ряд з квадратів залишків $\{\varepsilon^2(k)\}$, де $\{\varepsilon(k) = e(k)\}$. Обчислити вибірккову дисперсію σ_ε^2 , збурення $\varepsilon(k)$: $\sigma_\varepsilon^2 = \frac{1}{N-1} \sum_{k=1}^N \varepsilon^2(k)$, де N – число залишків після побудови моделі АР чи

АРКС.

Крок 3. Обчислити та побудувати графік вибіркової автокореляційної функції для квадратів залишків:

$$\rho(s) = \frac{\sum_{k=s+1}^N [\varepsilon^2(k) - \sigma_\varepsilon^2][\varepsilon^2(k-s) - \sigma_\varepsilon^2]}{\sum_{k=1}^N [\varepsilon^2(k) - \sigma_\varepsilon^2]} \quad (8)$$

Якщо існують такі значення $\rho(s)$, які відрізняються від нуля в статистичному сенсі, то це свідчить про присутність процесу АРУГ або УАРУГ. Для того щоб переконатись у присутності гетероскедастичності, використовують Q -статистику Люнга-Бокса, яка обчислюється за виразом:

$Q = N(N+2) \sum_{i=1}^n \rho(i)/(N-i)$, де $n = \frac{N}{4}$ (емпірично встановлене значення). Якщо

значення $\varepsilon^2(k)$ некорельовані, то Q -статистика повинна мати розподіл χ^2 з n ступенями свободи.

Крок 4. Використовуючи ряд значень $\varepsilon^2(k)$, побудувати модель УАРУГ (або іншу модифікацію):

$$h(k) = a_0 + \sum_{i=1}^q \alpha_i \varepsilon^2(k-i) + \sum_{i=1}^p \beta_i h(k-i) \quad (9)$$

Якщо в цій моделі хоча б один із коефіцієнтів $\alpha_i, i \in 1, \dots, q$ є значимий (для якого відхилена гіпотеза про рівність 0), то процес є дійсно гетероскедастичним. Оскільки модель (9) описує залишки моделі з деяким наближенням, то в загальному випадку доцільно продовжити процес уточнення моделей, що описують вихідний процес в цілому. Тобто, можна уточнити початкову модель $AP(p)$ чи $APKC(p, q)$, а також модель типу (2).

Крок 5. Скористатись моделлю (9) для того, щоб отримати дійсні значення залишків, які описуються цією моделлю, тобто згенерувати ряд $\{\varepsilon_1(k)\}$. Згенерувати ще один ряд $\{y_1(k)\}$, де $y_1(k) = y(k) - \varepsilon_1(k)$, за допомогою якого процес уточнення моделей може бути продовжений.

Висновки

Дана робота присвячена аналізу та моделюванню фінансово-економічних показників з метою прийняття правильного рішення при виконанні торговельних операцій на фондовій біржі.

Дослідивши та проаналізувавши авторегресійну та узагальнену авторегресійну модель з умовною гетероскедастичністю, метод групового урахування аргументів та каскадну нейронну мережу на ортонеїронах, їх було обрано для виконання короткострокового прогнозування волатильності акцій.

Література

1. *Сохацька О.М.* Біржова справа [Текст] / О.М.Сохацька: Підручник. – Тернопіль: Карт-бланш, 2003. – 602с.
2. *Волохова Л.* Основні індикатори фондового ринку і їхнє економіко-статичне моделювання [Текст] / Л. Волохова, О. Евсеєнко // «Ринок цінних паперів України». - 2005. - №5-6. - С. 3-8.
3. *Кузнєцова Н. М.* Фінансові ринки [Текст] / Н. М. Кузнєцова. - Київ. - 2002 р. - 523 с.
4. *Івахненко А. Г.* Прийняття рішень на основі самоорганізації / А. Г. Івахненко, Ю. П. Зайченко, В. Д. Димитров: «Сов. радио» 1976, - 280 с.

УДК 519.2 (477)

Інтелектуальний аналіз небезпеки виникнення природних пожеж на основі геоінформаційних технологій

Путренко В.В., к. геогр. н.

*Національний технічний університет України «КПІ», м. Київ
putrenko@wdc.org.ua*

В роботі визначено методика побудови оціночної моделі небезпеки виникнення природних пожеж з використанням інтелектуального інструментарію ГІС-аналізу та використано методи факторного аналізу, растрової та булевої алгебри, математичного картографування.

Вступ

В структурі надзвичайних ситуацій природного походження природні пожежі традиційно займають провідні позиції. В зв'язку з цим актуальним є завдання оцінки небезпек виникнення природних пожеж на території України. Створення геоінформаційної моделі для інтелектуального аналізу природних пожеж включає проведення факторного аналізу чинників, які впливають на вірогідність виникнення пожеж, оцінку кожного із цих чинників та визначення інтегральної оцінки.

Використання елементів інтелектуального аналізу при прогнозуванні природних пожеж досліджено в роботах Колодяжного О.А., Luhmann E., Машковського А.Г., Потапенко Л.С., Ясинський Ф.Н., Потемкина О.В., Сидоров С.Г. [1, 3]. Цей напрям активно розвивається в роботах Christopher B. O Neal, John D. Stuart, Radmila Jovanović, Željko Bjeljac та інших дослідників [4, 5, 6].

Метою дослідження є розробка інструментарію інтелектуального аналізу для оцінювання небезпеки природних пожеж на основі геопросторових даних та інструментів ГІС-аналізу. Завданнями дослідження є визначення підходів до методики оцінки небезпек, збір, обробка та аналіз геопросторових даних, оцінка результатів інтелектуального ГІС-аналізу.

Основні факторів виникнення природних пожеж

Виникнення природних пожеж пов'язано з чисельними збитками та загрозою для сталого розвитку і безпеки життя. Ці пожежі мають різний характер в залежності від умов виникнення пожежі, рослинного та ґрунто-

вого покриття. Тому необхідною умовою оцінки небезпек є вибір факторів, які впливають на можливість пожежі. До основних груп природних факторів відносяться рельєф, рослинний покрив та кліматичні ресурси, які разом визначають особливості ландшафтної структури, які при певній вірогідності можуть сприяти виникненню пожеж.

В результаті проведення процедури факторного аналізу було визначено низку чинників, які потребують кількісного відображення в моделі. До групи чинників рельєфу відноситься висота фізичної поверхні над рівнем моря, яка визначає інсоляцію та розчленованість рельєфу, ухил поверхні, експозиція схилів. Кліматичні фактори відіграють провідну роль в оцінці ризиків природних пожеж. Традиційно для моделювання ризиків пожеж враховують такий набір чинників як температурний режим, опади, баланс вологи, сонячна радіація, напрямок та швидкість вітру. При аналізі було враховано тільки основні чинники: середньорічну температуру в градусах Цельсія та середньорічну кількість опадів у міліметрах. Відповідно фактор температури враховується в моделі як стимулятор. Із його збільшенням небезпека природних пожеж збільшується. Фактор опадів є дистимулятором і з його збільшенням небезпека зменшується.

Основним фактором виникнення природних пожеж виступає рослинний та ґрунтовий покрив. В залежності від його характеру ранжується потенційна небезпека пожеж. В залежності від співвідношення ступеня покриття ґрунтового покриття та видів рослинності можливо виділення різних просторових угруповань за ступенем небезпеки виникнення пожеж. Основними ознаками для рослинності виступають типізація за показниками лісовий покрив, чагарник чи трав'яний покрив; вічнозелений або сезонний покрив лісів, хвойні чи листяні ліси; співвідношення видів рослинності та ступінь проектного покриття ґрунтів. Для ґрунтового покриття важливими факторами є відкриті території, вологість ґрунтів та умови їх формування – водно-болотні угіддя та торфові родовища. Також велике значення мають форми землекористування: сільськогосподарські території різних типів, урбанізовані території, інші типи землекористування. Лісові та степові території мають найбільший ступінь небезпеки. В тому числі вічнозелені хвойні ліси та степова територія з різним типом рослинності.

Усі чинники моделі нормалізуються у діапазоні від 0 до 1 одночасно з врахуванням їх впливу як стимуляторів та дестимуляторів у моделі.

Реалізація методики інтелектуального геоінформаційного аналізу

Загальна методика оцінки небезпек складається з п'яти кроків:

1. Визначення відповідних даних для кожної небезпеки та їх характеристик.
2. Перекласифікація та трансформація даних у растровий формат.
3. Зважування вхідних даних та додавання стандартизованих значень.
4. Математична обробка шарів даних.
5. Аналіз шару з даними про небезпеки.

В якості первинних даних для моделювання показників рельєфу було використано дані з топографічних карт масштабу 1:200 000. За допомогою модуля Spatial Analyst у складі ПЗ ArcGIS було проведено моделювання GRID-поверхні за допомогою алгоритму обернено зважених відстаней. Ця модель була використана для побудови похідних карт ухилів рельєфу та експозиції схилів, а також їх пере класифікації згідно з визначеними схемами групування значень. Кліматичні дані були розраховані шляхом інтерполяції даних багаторічних спостережень на метеопостах на території України. Дані про рослинний покрив та типи землекористування були отримані з топографічних карт та карт розподілу видового складу порід деревини за даними Міністерства екології та природних ресурсів України [2].

Важливим етапом розробки методики є визначення вагових коефіцієнтів під час операцій растрової алгебри, яка дозволяє зіставляти нормалізовані значення чинників та визначати їх сумарний вплив. Вагові коефіцієнти були визначені за результатами факторного аналізу. В даному випадку нормалізовані значення підсумовуються за допомогою інструменту Raster Calculator та методів растрової алгебри, який дозволяє обчислити нові растри на підставі тих, що існують, використовуючи ряд функцій і операторів. Числення растрів може виконуватися комплексними виразами, що включають арифметичних операторів, оператори порівняння і логічні, на підставі їх табличних даних (Overlays using a decision table).

Небезпека пов'язана зі складом рослинного покриву та особливостями рельєфу (N_1) розраховується за формулою:

$$N_1 = 0,54Z + 0,32R_2 + 0,09R_3 + 0,06R_1,$$

де Z – значення типу рослинного покриву,

R_1 – висота рельєфу, м,

R_2 – ухил поверхні рельєфу, градуси,

R_3 – експозиція поверхні рельєфу.

Небезпека пов'язана з кліматичними факторами (N_2) розраховується за формулою:

$$N_2 = 0,7375 + 0,2625 O,$$

де T – значення середньорічної температури,

O – значення середньорічної кількості опадів.

Інтегральна оцінка небезпеки виникнення природних пожеж (N) розраховується за формулою:

$$N = 0,81N_1 + 0,19 N_2$$

Результуючі значення отримано в діапазоні від 0 до 1, де значення 0 унеможливають можливість природних пожеж (наприклад, водні поверхні), значення близькі до 1 свідчать про високу небезпеку потенційного виникнення природних пожеж.

На останньому етапі аналізу було використано методи зональної статистики із використанням адміністративно-територіального поділу території України за районами. В цьому випадку для кожної одиниці було розраховано середнє зважене значення небезпеки природних пожеж. Ця інформація може бути використана в управлінні територіальним розвитком та попередженні надзвичайних ситуацій [Рис. 1].

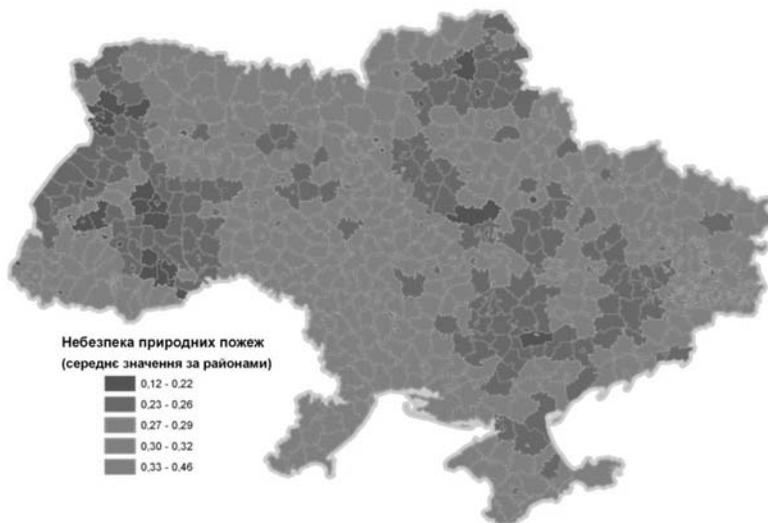


Рис. 1. Оцінка небезпеки природних пожеж за адміністративними одиницями України

Надання оцінки в розрізі районів дає більш генералізовану ситуацію, яка вказує загальні тенденції в розподілі небезпеки природних пожеж.

Висновок

Геоінформаційне моделювання дозволяє будувати оціночні моделі різного ступеня складності. В тому числі оцінка небезпек природних та техногенних надзвичайних ситуацій є важливим завданням, яке потребує вирішення на основі використання географічних даних. Інтелектуальний аналіз факторної моделі небезпеки виникнення природних пожеж показав доцільність та ефективність використання операцій інтерполяції растрових поверхонь та їх подальшого аналізу за допомогою методів перекласифікації, нормалізації, зваженого числення та зональної статистики.

Список літератури

1. *Колодяжний О.А., Лихтманн Е., Машковський А.Г., Потапенко Л.С., Штена Ю.М.* Організація космічного моніторингу пожеж.// Матеріали першої всеукраїнської конференції з космічних досліджень, 2001.
2. Національна доповідь про стан навколишнього природного середовища в Україні у 2011 році. – К. : Міністерство екології та природних ресурсів України, LAT & K. – 2012. – 258 с.
3. *Ясинский Ф.Н., Потёмкина О.В., Сидоров С.Г., Евсеева А.В.* Прогнозирование вероятности возникновения лесных пожаров с помощью нейросетевого алгоритма на многопроцессорной вычислительной технике // Вестник ИГЭУ, Вып. 2, 2011. – С. 1- 4.
4. Atlas of natural hazards & risks of Georgia // Caucasus Environmental NGO Network, 2012 <http://drm.cenn.org/index.php/en/>
5. *Christopher B. Oneal, John D. Stuart, Steven J. Steinberg,* Geographic analysis of natural fire rotation in the California redwood forest during the suppression era // Fire Ecology, Volume02, Issue01, Spring, 2006, <http://fireecology.org/docs/Journal/pdf/Volume02/Issue01/073.pdf>
6. *Radmila Jovanović, Željko Bjeljac, Olgica Miljković, Aleksandra Terzić* Spatial analysis and mapping of fire risk zones and vulnerability assessment – case study mt. Stara Planina // Prevention and Education in Natural Disasters, 2013, <http://www.doiserbia.nb.rs/img/doi/0350-7599/2013/0350-75991303213J.pdf>

УДК 681.3

Разработка методов самообучения поисково-рекомендующей системы с целью усовершенствования ее поведения

Рогущина Ю.В., к.ф.-м.н., доц., Институт программных систем НАН Украины, г. Киев, ladamandraka2010@gmail.com

Предложено использование онтологических моделей пользователя, информационных ресурсов и соответствующих предметных областей для формирования параметров, по которым стоит индуктивное обобщение, направленное на самообучение поисково-рекомендующей системы с целью усовершенствования ее поведения.

Введение

Значительная часть современных приложений интеллектуальны и ориентированы на использование и преобразование знаний об интересующей пользователя предметной области (ПрО). Большинство из них динамичны, т.е. ориентированы на функционирование в открытой информационной среде, в частности, на поиск и использование ресурсов Web. Таким образом, поиск в Web оказывается составной частью самых разнообразных информационных систем, а общая тенденция указывает на интеллектуализацию средств поиска. Одно из основных направлений интеллектуализации информационных систем связано с их способностью к самообучению, то есть с наличием средств и методов, позволяющих на основе собственного опыта усовершенствовать собственное поведение и получаемые результаты. Поэтому важно разрабатывать методы самобучения, специфичные для поисково-рекомендующих систем (ПРС).

Семантический поиск

При традиционном информационном поиске производится сопоставление запроса пользователя, характеризующего его информационную потребность, со сведениями о контенте проиндексированных информационных ресурсов (ИР), и в ответ на запрос пользователя формируется группа информационных объектов, по тем или иным параметрам сопоставленных с запросом.

Анализ исследований в области информационного поиска показал, что дальнейшее усовершенствование средств описания и алгоритмов

сопоставления запросов и ИР практически не улучшает ситуацию, т.к. для более эффективного поиска нужно использовать дополнительные знания – о ПрО, интересующей пользователя, о качестве ИР, о самих пользователях и о сообществах пользователей с подобными интересами.

При персонифицированном поиске субъектами обработки являются тройки (запрос, пользователь, ИР). При этом сопоставление производится уже по-разному для различных пользователей в зависимости от их персональных предпочтений.

При простом знание-ориентированном поиске при обработке учитываются не только формальные сведения о запросе, пользователе и ИР, но и более сложно структурированные знания о них. Тогда при их сопоставлении необходимо будет оценивать степень подобия этих знаний. В частности, при онтологическом подходе к представлению знаний для каждого ИР может указываться онтология ПрО, характеризующая его контент, а для пользователя – онтология интересующей его ПрО, а при сопоставлении ИР и запроса необходимо будет выполнить сопоставление этих двух онтологий.

Семантический поиск определяется как метод информационного поиска, в котором релевантность документа запросу определяется семантически (по близости смысла), а не синтаксически (по встречаемости ключевых слов в документе). То, какие именно знания используются, как они представлены и как они обрабатываются, зависит как от специфики разрабатываемой ИПС, так и от концепции, выбранной ее разработчиками, но в общем случае.

Постановка задачи

Чтобы обеспечить для различных интеллектуальных приложений, функционирующих в открытой информационной среде Web, необходимо обеспечить этим приложениям эффективный доступ к ресурсам Web, т.е. разработать средства семантического поиска. Чтобы повысить pertinентность такого поиска, нужны методы самообучения поисковой системы на основе собственного опыта. Использование алгоритмов продуктивного вывода для этой цели требует создание структурированных моделей пользователей и информационных ресурсов, на основе которых можно описать их взаимодействие. Использование онтологического подхода для формирования параметров, по которым строится индуктивное обобщение, обеспечивает интероперабельность знаний и возможность их экспорта из Web.

Онтологическая модель взаимодействия пользователей и ресурсов при семантическом поиске в Web

Семантический поиск представляет собой надстройку над традиционным информационным поиском, в котором с целью повышения pertinентности поиска используется обработка знаний, которые касаются как самого пользователя и его информационных потребностей (персонализация поиска) [1], так и об информационных ресурсах, среди которых осуществляется поисковая процедура. Таким образом, семантический поиск состоит из формирования информационных моделей пользователя, интересующей его предметной области, задачи, которую решает пользователь, и информационных моделей доступных информационных ресурсов (ИР), которые характеризуют их семантику, и их дальнейшего сопоставления.

Сейчас для интероперабельного представления различных знаний в Web широко применяются онтологии, обеспечивающие явное формализованное представление семантики представленной информации и обеспечивающие возможность логического вывода на них. Поэтому представляется целесообразным для поддержки семантического поиска разрабатывать именно онтологические модели взаимодействия пользователей и ИР в информационном пространстве Web, а также методы их сопоставления и пополнения.

В онтологической модели взаимодействия пользователей и ИР используются такие классы, как *онтология ПрО*, которая описывает область, к которой относятся информационные потребности пользователя; *тезаурус задачи* – термины онтологии, совокупность которых характеризует ту конкретную задачу из ПрО, которую в данный момент решает пользователь, и их вес; *запрос* – множество ключевых слов, характеризующих одну их информационных потребностей пользователя, связанный с конкретной задачей при помощи тезауруса; *тема* – множество запросов, связанных с одной информационной потребностью разных пользователей, позволяющее объединять семантически связанные запросы; *результат запроса* – ссылки на ИР и их оценки; *информационный ресурс*; *пользователь* и *группа пользователей*.

Класс *Пользователь* несколько групп свойств:

1. регистрационная информация: идентификатор пользователя; пароль для доступа к ИПС;

2. опыт взаимодействия ИПС с пользователем: списки онтологий, которые пользователь применял для описания своих информационных интересов; тезаурусов, которые пользователь применял в поисковых запро-

сах; ранее выполненных запросов и результатов выполненных запросов с оценками пользователя для найденных результатов;

3. сведения, импортируемые из внешних источников – социальных сетей, Википедии и других вики-ресурсов; сайта пользователя и его публикаций в Web;

4. собственные характеристики пользователя, характеризующие сферу его компетенций в различных Про;

5. биографические данные о пользователе;

6. психофизиологические свойства пользователя, влияющие на его способность к восприятию информации

В большинстве ПРС последняя группа свойств не используется, поэтому рассмотрим ее более детально. Сейчас существует много психологических тестов, позволяющих достаточно четко классифицировать личность (например, тест Айзенка, ММРІ), а затем на основе такой классификации охарактеризовать (с точки зрения формы представления информации, а не собственно контента) те ИР, которые в большей степени могут удовлетворить такого пользователя. Например, достаточно распространена классификация личностей по скорости возбуждения и торможения, в соответствии с которой выделяю четыре типа – холерик, сангвиник, флегматик и меланхолик, подразделение людей на экстравертов и интровертов и т.д. Социопсихофизиологические исследования обеспечиваются рядом компьютерных систем, которые используются для получения индивидуального адаптационного социопсихо-физиологического портрета человека в различных сферах его жизнедеятельности и рассматривают проблемы создания для пользователя комфортного индивидуального адаптационно-информационного пространства [2].

Такие индивидуально-типологические свойства, как сила, подвижность, уравновешенность нервных процессов, тип реагирования, эмоциональный фон, скоростные свойства анализаторных систем, особенности высших психических функций (восприятие, внимание, память, мышление, интеллект); поведение в стрессе, тип оптимальной работоспособности, могут в значительной мере определить выбор пользователя при восприятии информации – и, соответственно, при оценке предложенных ИР. Например, для экстраверта положительными чертами ИР являются наличие иллюстраций, использование коротких простых предложений, эклектичный дизайн, применение ярких цветов для выделения важных элементов и т.д., а для интроверта такими положительными чертами являются сложная структурированность текста, сложноподчиненные предложения, пастельная цветовая палитра. Поэтому можно предположить, что близкие по контенту ИР с различным оформлением по-разному будут оценены экстравертами и интавертами. Включение в число анализируемых при

индуктивном выводе свойств психофизиологических параметров пользователя позволит находить более точные обобщающие правила.

Класс *группа пользователей* описывает множество пользователей, объединенных по тем или иным причинам. Построение групп пользователей может осуществляться как вручную (на основании имеющихся у пользователей знаний), так и автоматически: в частности при помощи индуктивного обобщения опыта взаимодействия пользователя с системой.

Использование индуктивного вывода для формирования групп пользователей

При интеллектуальном поиске используются знания, которые могут быть извлечены из опыта работы поисково-рекомендующей системы (ПРС) различными методами Data Mining - например, на основе индуктивного (от частного к общему) и традиционного (от частного к частному) обобщения. Современные вычислительные возможности позволяют системам Data Mining извлекать полезные для пользователя закономерности из описаний, содержащих большое количество параметров. Но для этого необходимо наличие структурированного описания ситуаций, для которых строятся такие закономерности. Значительное количество современных рекомендующих систем [3,4] не использует структурированный подход к описанию пользователей и рекомендуемых объектов, и потому при выработке рекомендаций может применять только традиционные правила, построенные на основе опыта взаимодействия с пользователями: если есть два пользователя, u_1 и u_2 , и при оценке предложенных им ИР $r_i, i = \overline{1, n}$ оценки пользователей совпадали $o(u_1, r_i) = o(u_2, r_i) \forall r_i, i = \overline{1, n}$ (или хотя бы достаточно часто были близки), то можно предположить, что для некоторого РО x , который оценивал пользователь u_1 , но не оценивал пользователь u_2 , оценка пользователем u_2 совпадет (или хотя бы будет близка) с оценкой пользователя u_1 . При этом, если структурированное описание пользователей и ИР отсутствует, то нельзя вывести какие-то более общие закономерности, так как отсутствуют параметры, по которым можно группировать ИР и пользователей.

В качестве источника знаний для такого структурированного описания предлагается использовать онтологическую модель пользователя, описанную выше.

Существуют независимые подходы к реализации индуктивного и традиционного извлечения знаний из данных: ID3, ACLS, CART и т. д. Наиболее интересным, в связи со спецификой проводимой работы, оказался алгоритм ID3 [5], который специально разработан для извлечения ценной информации из больших объемов слабо структурированных данных. При работе этого алгоритма время вычислений зависит линейно от числа введенных примеров, числа атрибутов, используемых для описания примеров, и числа узлов в строящемся дереве решений. Методы, подобные ID3 (C4.5, ID4 и т. д.), предназначены для обобщения опыта экспериментов, параметры и результаты которых описаны через лингвистические переменные. ID3 принадлежит к невозрастающим алгоритмам. ID3 предназначен для построения бинарного дерева решений, а ID3m [6] – модификация ID3 для произвольного (конечного) количества решений.

На вход алгоритма ID3m поступает обучающая выборка H – набор из n классифицированных (получивших одну из возможных оценок) примеров одинаковой размерности. Каждый пример из выборки представляет собой упорядоченную последовательность значений s атрибутов и результирующего атрибута.

Если обучающая выборка содержит примеры, в которых все значения атрибутов одинаковы, а решения различны, то введенная информация недостаточна для построения классификационного правила. Если множество примеров пустое, то можно произвольно связать его с любым решением. Если все примеры относятся к одному классу, строится один лист дерева решений, связанный с этим классом. В противном случае необходимо выбрать один из атрибутов и разделить множество атрибутов на подмножества в зависимости от значения этого атрибута и применить алгоритм к каждому из полученных подмножеств.

На каждом шаге работы алгоритма вычисляется, какой атрибут несет наибольшее количество информации о результате. В результате работы алгоритма ID3m формируется дерево решений, в котором каждый лист связан с одним из решений, каждый узел – с именем одного из атрибутов, а выходящие из такого узла ветви – со значениями этого атрибута. В обучающей выборке элементами являются описания того, как пользователь поступил при получении ИР, атрибуты – свойства классов *пользователь* и *информационный ресурс* из соответствующих онтологических моделей, а результат – оценка пользователем ИР. Такое дерево решений позволяет по параметрам ИР прогнозировать, как именно оценит его пользователь. На основании дерева решений формируются группы пользователей, похоже оценивающих ИР.

Выводы

Использование онтологических моделей пользователя, информационных ресурсов и соответствующих предметных областей для формирования параметров, по которым стоит индуктивное обобщение, направленное на самообучение поисково-рекомендующей системы с целью усовершенствования ее поведения, позволяет более эффективно обобщать опыт системы при меньшем объеме обучающей выборки за счет того, что часть знаний об окружающем мире закладывается в систему явным образом.

Литература

1. *Kobsa A.* User modeling: recent work, prospects and hazards. – <http://zeus.gmd.de/~kobsa/papers/1993-aui-kobsa.pdf>.
2. *Кузьмина К.И., Оноприенко В.Н., Козак Н.С., Семик Т.М., Андон Т.А.* Семейная медицина сегодня и проблема ее дальнейшей интеллектуализации с помощью информационных технологий и компьютерных систем // Теорія і практика управління соціальними системами, № 2, 2012. – С.56-67
3. *Middleton S., De Roure D., Shadbolt N.* Ontology-Based Recommender Systems // in Handbook on Ontologies, Edt. by S.Staab, R.Studer, Springer, 2009. – P.779-796.
4. *Розушина Ю.В.* Использование онтологических знаний в рекомендующих системах // Проблеми програмування, 2013, №2. – С.71-86.
5. *Quinlan J.R.* Discovery rules from large collections of examples: a case study // Expert Systems in the Microelectronic Age. – Edinburg, 1979. – P. 87-102.
6. *Розушина Ю.В.* Знание-ориентированные средства поддержки семантического поиска в Web // Материалы IV международной научно-технической конференции «OSTIS-2014, Минск, БГУИР, 2014. – С.339-352.

УДК 519.254

Кластерний аналіз в задачах обробки інформації

Савеленко О.К., Кіровоградський національний технічний університет, м. Кіровоград kolodochkinaek@mail.ru

В статті розглядаються характерні особливості рішення задач кластерного аналізу та особливості розвитку відповідної прикладної області рішення задач цього класу на основі використання автоматизованих систем обробки і аналізу інформації. Наводяться результати проведеного порівняльного аналізу основних методів кластерного аналізу найбільш уживаних методів кластеризації в середовищі MatLab.

Вступ

Характерною рисою сучасної світової спільноти є обробка, використання та накопичення величезних обсягів інформації. Одним з основних завдань по обробці і аналізу інформації є задача її кластерного аналізу або кластеризації. Актуальність рішення цієї задачі особливо висока у зв'язку з усе зростаючим обсягом інформації, що накопичується в різних галузях знань. Потреба продуктивної переробки величезних потоків даних, без якої неможливе їх ефективне використання, породжує необхідність підвищення якості рішення задач кластеризації за рахунок розробки та впровадження систем автоматичної обробки і систематизації інформації.

Предметна область інтелектуального аналізу даних і задачі кластерного аналізу

Переробка інформації – це інтелектуальний аналіз даних і витягання знань, які спрямовані на виявлення певних закономірностей в даних [1]: прогнозування, класифікації, кластеризації тощо.

Основою для систем прогнозування служить інформація, що зберігається у вигляді тимчасових послідовностей зв'язаних в часі подій. Якщо вдається виявити функції, адекватно відбиваючі динаміку їх поведінку, то за їх допомогою можна передбачити значення такої послідовності в майбутньому. Рішення задачі прогнозування припускає знаходження певних моделей, що відбивають взаємозв'язки в послідовностях даних, а задача прогнозування формально може бути зведена до задачі класифікації.

При класифікації шляхом аналізу вже класифікованих об'єктів з навчальної послідовності здійснюється формування деякого набору вирішальних правил, по яких можна рознести об'єкти, як відомі з навчальної вибірки, так і нові невідомі об'єкти по заданих класах. Кластеризація відрізняється від класифікації тим, що самі групи заздалегідь не задані. Спочатку самою системою виробляються вирішальні правила, по яких формуються групи (кластери), а потім буде вирішуватися задача класифікації. Таким чином, кластеризація є підзадачею множини задач, які вирішуються у напрямі Data Mining (вилучення знань).

Задача кластеризації відноситься до найбільш складного класу задач - «навчання без учителя», оскільки відсутні відповіді до навчальних прикладів, а віднесення об'єктів до того або іншого кластера має бути виконане автоматично. При цьому в загальному випадку апріорно не відомі ні топологія кластерів, ні число кластерів. В результаті рішення задач кластеризації виявляється прихована структура даних та неочевидні закономірності в даних.

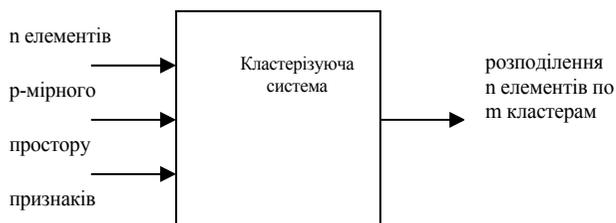
Актуальність задачі кластеризації

Попри те, що задача розробки автоматичних систем кластеризації поставлена давно та досягненні певні на успіхи в області розпізнавання образів, рішення її з високими показниками якості доки не отримане. Отже виникає необхідність залучення нових підходів до рішення задач кластерного аналізу і розгляду їх з системних позицій, враховуючи обчислювальну складність для багатовимірного простору ознак з одного боку, і наявність великого числа об'єктів з іншого, оскільки розбір і детальний облік окремих характеристик задачі не приносять бажаних результатів. Це підтвердив і проведений аналіз наявних джерел літератури. Таким чином, новий метод кластеризації повинен ґрунтуватися на нових принципах обробки інформації.

Постановка задачі кластеризації

Кластеризація - це процес розбиття [1,7] безлічі об'єктів на задане або невідоме число кластерів на основі деякої міри схожості, яка характеризується будь-якою загальною властивістю. У загальному випадку завдання кластеризації зводиться до наступного (рисунок 1).

Рис. 1 - Загальне представлення задачі кластеризації.



На вході системи, що реалізовує метод або групу методів кластеризації, є деякий набір з n елементів. Система повинна виявити наявну подібність або відсутність її між n елементами, і розташувати їх по m кластерах ($m < n$) так, щоб на виході системи кожен кластер характеризувався деяким розподілом елементів, які потрапили до нього. Як впливає з визначення, число кластерів, по яких має бути проведене розбиття, може бути невідомим апіорі. Вхідні n елементи складають образ, який подається на вхід системи кластеризації.

Кожен з n елементів задаємо ознаковим вектором для опису ступені вираженості кожної окремої ознаки, поєднання яких визначає унікальність елементу. Простір ознак може бути багатовимірним, допускається вкладеність ознак, при цьому матрицю завжди можна розгорнути у вектор. Міра схожості (метрика або відстань) між вхідними елементами визначається по схожості поєднання ознак. Поля ознак – це їх різновид і кількість, відомі системі апіорно.

Простір ознак в загальному випадку є безперервним. Це означає, що важлива не лише наявність або відсутність ознаки, але й ступінь її вираження. Тому може відбуватися об'єднання в кластери не лише по наявності групи ознак, але і в залежності від ступені їх вираженості.

Враховуючи вище зазначене, формуємо задачу кластеризації, як задачу пошуку розбиття K безлічі об'єктів X на задалегідь задане або невідоме число m підмножин безлічі X , які не перетинаються, відповідно до критерію міри схожості між об'єктами [2]:

$$K = \{K_1, K_2, \dots, K_m\},$$

$$K_i \subseteq X, i = 1, m \quad (1)$$

$$\bigcup_{i=1}^m K_i = X, \quad (2)$$

$$K_i \cap K_j = 0, i \neq j,$$

де $X = \{x_1, x_2, \dots, x_n\}$, $x_i = \{x_{ij}, x_{i2}, \dots, x_{in}\}$, n -число об'єктів кластеризації, p -число ознак опису кожного об'єкта, який представлений точкою у багатовимірному просторі ознак. Результати кластеризації можуть бути представлені у вигляді вектора відповідей $n \times 1$, або у вигляді матриці результатів кластеризації $m \times n$.

Кількість методів кластеризації зростає у міру збільшення задач, адже у кожному конкретному випадку часто відбувається об'єднання методів або використання набору різних метрик, оскільки ступінь схожості між об'єктами може задаватися різними способами. Розглянемо класичні методи кластеризації та проведемо їх порівняльний аналіз.

Класичні методи кластеризації діляться на ієрархічні (агломеративні) і неієрархічні (ітераційні, розділені, засновані на алгоритмах таксономії) [1, 2].

Ієрархічна кластеризація реалізується шляхом послідовного об'єднання малих груп даних у великі або - розбиттям великих груп на менші. Кінцевий результат кластеризації - дендрограма, яка показує, як групи співвідносяться одна з одною.

При збільшенні порогу ухвалення рішення про об'єднання двох або більше об'єктів в один кластер, відбувається об'єднання кластерів в один кластер. На самому початку процесу кластеризації кожен об'єкт утворює свій кластер. При малому значенні порогу усі об'єкти потрапляють в один кластер. В якості основних етапів в алгоритмах ієрархічної кластеризації можна виділити наступні: визначення схожості чи відмінності між кожною парою об'єктів на основі обчислення ступені схожості; побудова двійкового ієрархічного дерева кластеризації ітераційним об'єднанням близьких кластерів; ухвалення рішення про те, який рівень дендрограми містить шуканий розподіл.

При неієрархічній кластеризації здійснюється розподіл даних в групи за найбільш схожими ознаками (щільні області в просторі даних). Інший можливий метод кластеризації полягає в зниженні значення певної міри відмінностей даних в межах кожної групи зі збільшенням міри відмінностей самої групи від інших груп за рахунок переміщення об'єктів з кластера в кластер. І в першому, і в другому випадках здійснюється пошук типових представників кластерів - центрів кластерів за геометричним критерієм або по групі критеріїв. В якості основних етапів в алгоритмах неієрархічної кластеризації можна виділити наступні: розбиття усієї кількості на групи, найбільш близькі до одного з центрів кластерів (первинні m центри кластерів вибираються випадковим чином, кількість кластерів m апіорі задана); ітеративне уточнення центрів кластерів шляхом міграції точок з кластера в кластер до тих пір, поки центри кластерів не перестануть мінятися.

Методи кластеризації називаються по-різному, але суть їх однакова, за винятком нюансів. Це пов'язано з тим, що при кластеризації можуть вирішуватися різні задачі [1, 2]: кластеризація самих об'єктів або кластеризація ознак об'єктів, при виявленні взаємного впливу ознак.

Найчастіше в якості міри схожості (а точніше - міри відмінностей) використовуються різні метрики відстані між елементами (об'єктами). Значення взаємних відстаней, обчислених по певній метриці, заносяться в симетричну матрицю відстаней D :

$$D = \{d_{rs}\}, \quad (3)$$

де d_{rs} відстань між r -ім і s -ім об'єктами, а $r, s = 1, 2, \dots, n$.

Використання декількох метрик необхідне для визначення відстані між об'єктами усередині кластера, відстані між об'єктами з різних кластерів,

відстані між самими кластерами. Міру схожості між кластерами будемо визначати відповідно до основних метрик. Нехай об'єкти $\{x_r\}$, $r=1,2,\dots,n_r$ належать до одного класу, а $\{x_s\}$, $s=1,\dots,n_s$ - до іншого. Відстань між об'єктами усередині кластера позначимо, як $d_{ist}(x_{ri}, x_{si})$.

Ділення метрик умовне, оскільки ті, що використовуються для визначення відстані між кластерами, застосовуються і для визначення відстаней між елементами усередині кластера, і навпаки. Відстань Евкліда використовується найчастіше. Манхетенська відстань призводить до таких же результатів, як і для звичайної відстані Евкліда. Проте, для цієї метрики вплив окремих великих різниць (викидів) зменшується (оскільки вони не зводяться в квадрат). Відстань Чебишева доцільно застосовувати у випадку, коли два об'єкти необхідно призначити в різні кластери, якщо вони розрізняються по якій-небудь одній координаті (яким-небудь одним виміром). Ступеневу відстань використовують, якщо необхідно прогресивно збільшити або зменшити вагу, для якої відповідні об'єкти сильно відрізняються. Стратегія найближчого сусіда сильно стискає простір початкових змінних і рекомендується для отримання мінімального дерева. Застосовується у разі угруповань, що мають складну, ланцюжкову структуру. Стратегія далекого сусіда сильно розтягує простір початкових змінних і застосовується, коли шукані угруповання утворюють в просторі ознак кулясті хмари. Проміжне місце займають алгоритми, що використовують відстані центрів тяжіння і середнього зв'язку, які краще всього працюють у випадку угруповань еліпсоїдної форми.

Стратегія групового сусіда зберігає метрику ознакового простору. Стратегія Ward's method мінімізує внутрішній кластерний розкид об'єктів і дендрограма виходить з сильно розділеними кластерами [1,3].

На основі перелічених вище метрик працюють наступні ієрархічні методи кластеризації: на основі обчислення середньої відстані між кластерами, на основі обчислення середньої відстані між елементами усередині кластера, найближчого сусіда, на основі обчислення відстані між центрами кластерів, Варда, далекого сусіда, мінімального дерева, оцінки щільності вірогідності, ланцюгової розгортки.

До неієрархічних методів можна віднести методи: k – середніх, метод квадратичної помилки.

Порівняльний аналіз методів кластерного аналізу і невіршені задачі

Проведений аналіз наявних джерел інформації показав, що переважна кількість відомих досить давно методів кластеризації, повного і системного

їх порівняння ні у вітчизняній, ні в зарубіжній літературі не мають. Існують тільки огляди з вибіркоким порівнянням деяких методів [3, 5, 6], чи якісним порівнянням окремих властивостей методів [2, 4]. Тому для проведення порівняльного аналізу найбільш уживаних методів кластеризації в середовищі MatLab було створено програмний засіб, який дозволив вирішити задачу кластеризації зображень, приведених за допомогою відповідних методик.

Зокрема, була проведена кластеризація на основі використання наступних методів:

- ієрархічного методу кластеризації з використанням пари метрик Чебишева та одиночного зв'язування (один правильний варіант кластеризації та два майже правильних варіанта);

- ієрархічного методу кластеризації з використанням метрик Махаланобіса і повного зв'язування (не виявлено правильних варіантів кластеризації);

- ієрархічного методу кластеризації з використанням евклідової метрики та одиночного зв'язування (один правильний варіант кластеризації та два майже правильних);

- ієрархічного методу кластеризації з використанням евклідової метрики та одиночного зв'язування (не виявлено правильних варіантів кластеризації);

Для кожної пари метрик, що визначають метод кластеризації, було побудовано графічне представлення дев'яти варіантів кластеризації, що відповідають дев'яти різним значенням порогу для міри схожості між кластерами. За стійкий результат кластеризації беремо випадок наявності чотирьох правильних з дев'яти варіантів кластеризації. Таке різне поєднання метрик дає правильні результати кластеризації, тобто одне і теж поєднання метрик буде «хорошим» для одного зображення об'єкту та абсолютно непридатним для кластеризації іншого.

Оскільки ієрархічні методи є детермінованими в тому сенсі, що результат кластеризації для заданого набору даних може бути тільки один, а різними можуть бути тільки способи його інтерпретації, то для цих методів немає необхідності в повторних експериментах при рішенні задачі кластеризації на одних і тих же даних.

На відміну від ієрархічних, неієрархічні методи кластеризації мають елемент випадковості, оскільки в них результат кластеризації сильно залежить від початкового розподілу точок (об'єктів) по кластерах.

Аналіз результатів кластеризації неієрархічними методами був проведений на основі використання:

- ітераційного методу кластеризації k -середніх з використанням метрики на основі обчислення евклідової відстані (не виявлено правильних варіантів кластеризації);

- ітераційного методу кластеризації k -середніх з використанням метрики на основі квадратичної евклідової відстані (один правильний варіант при параметрі методу (числі кластерів) рівним трьом).

Проведені дослідження дозволили виявити сильну залежність результатів кластеризації від наявності апріорної інформації про топологію кластерів і про число кластерів. Це підтверджують і наявні джерела літератури, в яких стверджується, що націленість алгоритмів кластерного аналізу на певну структуру груп об'єктів в просторі ознак може призводити до неоптимальних або навіть до неправильних результатів, якщо гіпотеза про тип груп невірна [7]. У разі відмінностей реальних розподілів від гіпотетичних, вказані алгоритми часто «нав'язують» даним не властиву їм структуру і видають велике число помилок. Тому, в умовах апріорної невизначеності, необхідно застосування групи алгоритмів кластерного аналізу і остаточний висновок формується на основі комплексної оцінки результатів роботи цих алгоритмів [2].

Наявні джерела літератури також стверджують, що у разі кластеризації на основі одного з ієрархічних методів є складність ухвалення остаточного рішення про кластеризацію, зважаючи на нестійке розбиття на кластери. В цьому випадку потрібна додаткова інформація про передбачуване число кластерів, яку буде надавати експерт-людина. Сильна залежність результатів кластеризації від апріорної інформації про число кластерів характерна також і для неієрархічних методів кластеризації.

Необхідність експериментального підбору поєднання метрик (між об'єктами і між кластерами) означає відсутність універсальності кожного з методів кластеризації окремо. А коли по методу голосування не можна вибрати остаточний результат кластеризації з усієї безлічі результатів із-за непостійності результату при різних значеннях порогу, отримання правильної відповіді за допомогою розробленого програмного забезпечення взагалі стає неможливим.

Більшість методів припускають компактність образів і можливість виділення центрів кластерів (типових представників), якщо ж це виявляється неможливим, методи кластеризації видають помилкові результати кластеризації. Відмінною особливістю традиційних методів кластеризації є неможливість розпізнавати групи елементів образу, коли останні просторово не рознесені.

Коли групи елементів образу просторово рознесені (групи, що не перетинаються), то кластеризація проводиться успішно, але в такому випадку автоматизованій системі слід надати більше інформації про область перетину. Її відсутність або неповнота призводить до помилки в класифікації, чи до того, що образи взагалі не розділяються.

Практично усі методи кластеризації характеризуються високою обчислювальною складністю. Ієрархічні методи надають структурну інформацію, але можуть застосовуватись тільки для невеликих наборів даних. Зі збільшенням числа об'єктів, обчислювальні витрати ростуть. Розділяючі методи менш вимогливі до ресурсів, але вони залежать від "правильної здогадки про структуру" [2], тобто - кількості і початкове положення центрів. Якщо вибір початкових кластерів далекий від шуканого, ці методи стають дуже ресурсоемними в обчисленні нових центрів кластерів, а це означає, що з їх допомогою стає неможливим вирішити завдання з високою якістю.

Аналіз результатів роботи класичних методів кластеризації дозволяє говорити і про те, що усі розглянуті вище методи кластеризації мають обмеження, пов'язані з необхідністю апріорної інформації про структуру кластерів і про їх число. Кожен з методів не універсальний і видає правильні результати кластеризації тільки для зображень з певною топологією. Тому задача розробки нових методів кластеризації, що дозволяють здійснювати правильну кластеризацію в умовах відсутності якої-небудь апріорної інформації, дійсно актуальна на даний час.

Висновки

1. В результаті аналізу існуючих методів кластеризації були виявлені їх недоліки, до яких можна віднести наступне:

- необхідність апріорної інформації про передбачувану топологію кластерів, що на практиці не завжди здійснено і на даний час задача, як в загальному випадку ці параметри можна вибирати автоматично, не вирішена;

- сильна залежність результатів кластеризації від апріорної інформації про число кластерів;

- більшість методів припускають компактність образів і можливість виділення центрів кластерів, адже ці методи працюють на основі концепції усереднення по вибірці, що призводить до операцій над фіктивними величинами, або розгляду крайніх ситуацій (найкращий і найгірший випадки).

- необхідність експериментального підбору поєднання метрик схожості між об'єктами усередині одного кластера і метрик відмінностей між кластерами (відсутність універсальності кожного з методів окремо). Традиційні методи не дозволяють обробляти умовні зображення, які складають елеме-

нти, що описуються надмірним полем ознак. В цьому випадку знижується ефективність обробки вхідної інформації, оскільки особливістю деяких методів є неможливість відкидати незначущу та дублюючу інформацію;

- складність ухвалення остаточного рішення про кластеризацію, зважаючи на неоднозначне розбиття на кластери (результат залежить від вибраного порогу кластеризації). Як впливає з постановки задачі кластеризації, на вході системи є зображення, що складається з елементів, які необхідно розділити на кластери відповідно до поля ознак. На основі однієї з існуючих метрик по вибраному алгоритму формується матриця взаємних мір схожості між елементами образу. Ця матриця несе в собі усю інформацію про початковий образ, на її основі здійснюється кластеризація з виділенням кластерів, які повинні відбивати розподіл об'єктів по групах. Наскільки успішним виявиться аналіз матриці мір схожості, настільки правильним буде розподіл об'єктів по групах. Відмінною особливістю традиційних методів кластеризації є неможливість розпізнавати групи елементів образу, коли ті просторово не рознесені.

2. Аналіз наявних джерел літератури дозволяє виділити основні підходи до вдосконалення методів кластерного аналізу. Вони пов'язані із залученням нейронних мереж і нечіткої логіки. Проте, при детальному розгляді виявляється, що задачі розробки нових методів кластеризації залишається актуальними в силу невирішеності на сьогоднішній день проблем, означених в п. 1 висновків.

3. Необхідність при побудові автоматизованих систем кластеризації залучення таких підходів, які дозволили б ще більше наблизитися до обробки інформації по принципах, близьких до нейробіологічного прототипу універсального вирішального пристрою - мозку людини. Це дозволить здійснювати розробку дійсно сучасних систем автоматичної обробки інформації і ухвалення рішень.

Література

1. Дюк В. Data mining: учебный курс - СПб.: Питер - 2001. - 370 с.
2. Журавлев Ю.И. Об алгебраическом подходе к решению задач распознавания или классификации / Проблемы кибернетики. - М.: Наука - 1978.- вып. 33.- 23-56 с.
3. Загоруйко Н. Г. Прикладные методы анализа данных и знаний. — Новосибирск: ИМ СО РАН - 1999. - 203 с.
4. Мандель И. Д. Кластерный анализ. - М.: Финансы и статистика - 1988. - 254 с.
5. Шлезингер М., Главач В. Десять лекций по статистическому и структурному распознаванию. - Киев: Наукова думка - 2004. - 163 с
6. Потапов А. Б., Али М. К. Нелинейная динамика обработки информации в нейронных сетях Сб. «Новое в синергетике: Взгляд в третье тысячелетие».- М.: Наука - 2002. - 320-412 с.
7. Журавлев Ю.И. Об алгебраическом подходе к решению задач распознавания или классификации Проблемы кибернетики. - М.: Наука - 1978.- вып. 33. 23-56 с.

УДК 004.415

Інтелектуальний аналіз діагностичної інформації складних технічних комплексів

Савчук О.В., к.т.н., с.н.с., Кривенко К.С.

Національний технічний університет України «КПІ», м. Київ
savchuk@acts.kiev.ua

Пропонується інтелектуальний аналіз діагностичної інформації при декомпозиції складних технічних комплексів до рівня елементної бази. Розбирання елементів за їх фізичним та технічним станом виконано як класифікацію у базисі Карунена-Лоева за допомогою нейронних мереж у середовищі MATLAB.

Вступ

Для діагностування та аналізу відмов сучасних складних технічних комплексів використання інтелектуальних систем діагностування (ІСД) чи окремих компонент штучного інтелекту є актуальною задачею [1].

Метою статті є вибір типу нейронних мереж, способів опрацювання накопичених даних для підвищення якості й достовірності результатів діагностування та їх дослідження на прикладах реалізації процесу діагностування електрорадіокомпонентів (ЕРК).

Опрацювання та стиснення інформації за інтегральними фізичними ефектами

Для фізичних об'єктів функція сприйнятливості $g(\cdot)$, складна функція середовища, являється комплексною функцією

$$g(\cdot) = \text{const Re}[g(\cdot)] + \text{var Re}[g(\cdot)] + j\{\text{const Im}[g(\cdot)] + \text{var Im}[g(\cdot)]\}, \quad (1)$$

де Re – дійсна частина функції; Im – уявна частина; const – постійна частина; var – змінювана частина, яка саме цікавить в діагностиці.

Згідно з формулою (1) та перетворенням Тейлора в [2] отримані характеристики ЕРК, серед яких були розглянені основні, що мають місце для опису інтегральних мікросхем.

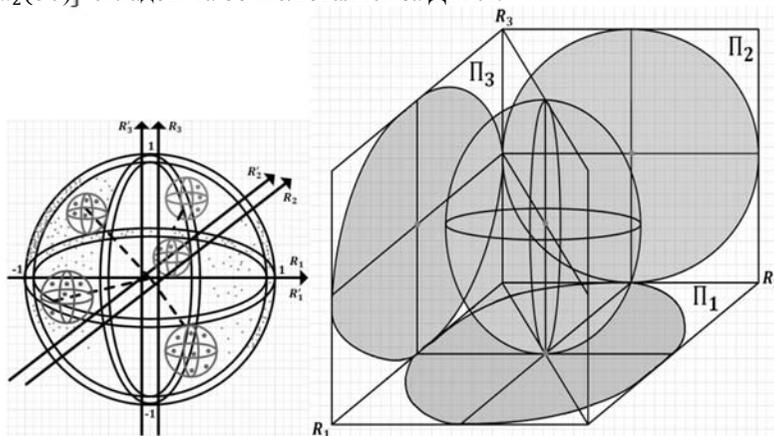
Стиснення первинної діагностичної інформації про стан ЕРК виконано за допомогою дискретного розкладання Карунена-Лоева (ДРКЛ) [3], що є розкладанням ансамбля початкових векторів за власними векторами коваріаційної матриці:

$$z_i = \sum_{i=1}^m v_i r_i, i = \overline{1, n}, n > m, \tag{2}$$

де $z_i = \{z_{ij}\}$, $j = \overline{1, \text{щ}}$; щ – число реалізацій початкових векторів; m – число незалежних базисних векторів; $r_i = \{r_{ij}\}$; $i = \overline{1, n}$; n – кількість досліджуваних початкових векторів; $v_i = \text{const}$ – коефіцієнти ДРКЛ.

Досліджувалися перехідні теплові характеристики різних типів резисторів та характеристики нелінійності інтегральних мікросхем серії K174XA11 [4,5].

Комплексні залежності квадратичної нелінійності за модулем та фазою, отриманих за методом різницевої частоти окремо для придатних та дефектних зразків, перетворювалися на косинусні $F_c[a_2(U_0)]$ та синусні $F_s[a_2(U_0)]$ складові та обчислювалися за ДРКЛ.



а)

б)

Рисунок 1 - Геометрична інтерпретація ДРКЛ: а) у тривимірному ортонормованому просторі векторів (ліва півсфера – для косинусної складової, права – для синусної) [4]; б) декомпозиція простору за проєкціями (з накладанням осей для косинусної та синусної складових)

Результатом природної класифікації дефектних та потенційно-ненадійних мікросхем за типами дефектів виявилися 5 класів у вигляді малих сфер в середині двох головних півсфер (рисунок 1,а). Канонічна форма декомпозиції тривимірного ортонормованого простору ДРКЛ (рисунок 1,а) за проєкціями П1, П2 та П3 (рисунок 1,б) дозволяє проводити класифікацію у двовимірних евклідових ортонормованих просторах П1, П2 та П3.

Спостереження за проєкціями дозволяє використовувати найпростіші типи нейронних мереж для розбавки ЕРК.

Класифікація електрорадіокомпонентів за допомогою нейронних мереж

Розглянемо багатовходовий перцептрон, нейрони якого мають активаційну функцію у вигляді одиничного стрибка. Робота перцептрона зводиться до класифікації вхідних сигналів, що належать k -вимірному гіперпростору ($k=2, \dots, 8$), по деякому числу класів ($w=4, \dots, 19$). З математичної точки зору це відбувається шляхом розбивки гіперпростору гіперплощинами.

Навчання багат шарового перцептрона у базисі Карунена-Лоева з двома базисними векторами зводиться до формування ваг зв'язків між першим і другим шарами за відомим алгоритмом у MATLAB. Збіжність алгоритму встановлюється теоремами [4]. Випробувано на 14 векторах для 4-х класів метало-діелектричних резисторів, число факторів - ознак пророблялося від 5 до 20. Спочатку навчали двошарову нейронну мережу з зворотнім поширенням помилки: MLP з двома шарами та 2-ма нейронами в другому шарі. Обидва шари активувалися логістичною функцією logsig та навчалися за методом градієнтного узвозу (функція traingd).

Таблиця 1 - MLP з двома шарами для резисторів

Код класу	Тип обрахунку помилки			
	Тип помилки, 10^{-2}			
	MSE	SSE	MSEREG	MAE
11	1.532	15.319	1.379	2.714
00	2.833	28.333	2.550	4.091
10	2.445	24.451	2.201	3.578
01	4.687	46.868	4.218	5.820

Результати навчання виявилися не досить приємними (таблиця 1).

Але після подальшого дослідження багат шарових перцептронів для резисторів класу 00 з різними типами навчальних функцій найкращий результат отриманий для MLP з чотирма шарами за алгоритмом пружного зворотного поширення з логістичною функцією активації trainpr (таблиця 2).

Таблиця 2 - MLP з чотирма шарами для резисторів

Навчальна функція	Функція активації прихованого шару logsig			
	Тип помилки, 10^{-2}			
	MSE	SSE	MSER EG	MAE
Traingd	1.90	26.56	1.71	7.92
Traingda	0.89	12.46	0.80	7.24
Traingdm	1.55	21.75	1.40	8.47
Traingdx	0.92	12.91	0.83	6.42
Trainlm	1.48	14.32	1.45	7.32
Trainrp	0.19	2.63	0.17	2.82

Для мікросхем з використанням декомпозиції ортонормованого простору ДРКЛ (рисунок 1,а,б) задача спрощується через те, що класифікація виконується для лівої й правої півкулі окремо, тобто число класів 2 або 3.

Дослідження на мережах Кохонена

Задача розбиття на групи (кластерний аналіз) може розглядатися як задача класифікації в контексті використання нейромереж з двох різних боків. Мережі Кохонена не потребують задання виходів нейромережі на навчальній множині, тобто навчання виконується тільки по вхідних даних, що називається навчанням без вчителя.

Цей алгоритм є нейромережним аналогом відомого алгоритму k середніх. При цьому процес навчання залежить від вибору метрики.

В загальному випадку алгоритм навчання мережі Кохонена виглядає наступним чином. На кожному кроці навчання на вхід мережі подається черговий вектор x_n та шукається нейрон, вага якого найменше відрізняється від цього вектора. Знайдений нейрон оголошується переможцем, і вектор його ваг ω оновлюється за формулою:

$$(k + 1) = \omega_i(k) + \eta(x_n - \omega_i(k)) + \alpha\omega_i(k)(1 - \|\omega_i\|^2). \quad (3)$$

де параметр η відповідає за швидкість навчання та змінює свої значення в проміжку (0; 1). Усі навчальні вектори обробляються по черзі, доки не відбудеться стабілізація або не виконається інша умова зупинки.

Якщо вибрана звичайна евклідова метрика $\|\omega\|^2 = 1$, то переможець буде нейрон, для якого скалярний добуток (x_n, ω_i) максимальний. Для підтримання такого нормування у цю формулу вводиться штраф за відхилення від 1.

Наведемо приклад навчання нейромережі Кохонена для чотирьох кластерів резисторів.

Отримані після навчання результати в nntool (рисунок 2,а,б).

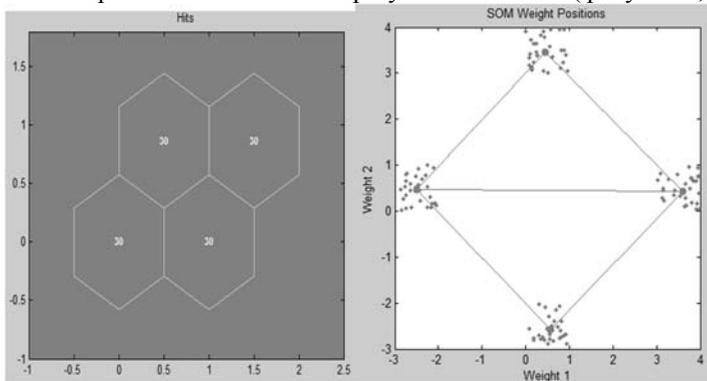


Рисунок 2 – а) кількість точок, що попали в кластери; б) формування центрів ваги

Подальше дослідження планується на радіально-базисних (RBF) мережах. Головна відмінність RBF-мереж від звичайних статичних багатопарових мереж прямого поширення полягає у функції нейронів прихованого шару. В звичайній багатопаровій мережі кожен нейрон робочого шару реалізує у багатовимірному просторі гіперплощину, а RBF-нейрон - гіперсферу. Для мікросхем (рисунок1,а), де дані близькі до утворення кругової симетрії, це дозволяє зменшити число нейронів. Відстань може обчислюватися на основі заданої метрики як:

$$d_j = \sqrt{\sum_{i=1}^{n(j)} (\omega_{ji} - x_{ji})^2}, \quad (4)$$

де $n(j)$ – кількість входів j -го нейрона другого шару.

Планується використання спеціальних типів радіально-базисних нейромереж: мережі GRNN (Generalized Regression Neural Networks) для розв'язку задач узагальненої регресії та мережі PNN (Probabilistic Neural Networks) для розв'язку імовірнісних задач.

Висновки

Дискретне розкладання Карунена-Лоева дозволило застосувати нейронні мережі прямого розповсюдження для задач діагностування електро-радіокомпонентів у середовищі MATLAB, а використання нейромережових технологій - збільшити ефективність технічних засобів діагностики ЕРК по виявленню прихованих дефектів, потенційної нестабільності і позапланових процесів деградації.

Література

- 1 *Berezhnoy V.P., Serdjuk G.B. and other.* The Technical Diagnosis of the elements of electroradio equipment. – М; Preprint of the Central Scientific-Research Institute “Electronica”, 1990. – 303 p. (in Russian).
- 2 *Сердюк Г.Б., Усатенко В.Г., Миненко А.П.* Автоматизированные комплексы для электрофизического диагностирования ИС и других изделий электронной техники по критериям качества и надежности. К.: КПИ, НПП АМАТЕК, 1992.-С. 55-59.
- 3 *Солодовщиков А.Ю., Платонов А.К.* Исследование метода Карунена-Лоева./ Электронный ресурс. Режим доступа: www/keldysh.ru/papers/2006/prep19/prep2006-19.html
- 4 *Савчук О.В., Кривенко К.С.* Інтелектуальний аналіз діагностичної інформації електро-радіокомпонентів в умовах невизначеності // Інтелектуальний аналіз інформації ІАІ-2013 / Зб. праць. – К.: Просвіта. – 2013. – С.211-217
- 5 *Telenyk S., Savchuk O., Pokrovskyy E., Domaskina N., Krivenko K.* Usage of Neural Networks for Processing Diagnostic Information of Electric Radio Components // Proc. of the 6-th Int. Conf. ACSN-2013. - P. 224-225.

УДК 621.391 : 004.052.3 : 004.052.4

О некоторых методах “борьбы с неустранимой неоднозначностью” в сетях телекоммуникаций

Северин Н. В., Леоненко Л. Л., к.ф.н., доц.

*Одесская национальная академия связи им. А. С. Попова, г. Одесса
Leonid.Leonenko@gmail.com*

Обсуждаются проблемы внедрения в сети телекоммуникаций средств распознавания “искаженных” сообщений, в том числе с неустранимыми искажениями, обусловленными представлением одной и той же информации различными кодовыми последовательностями. Показано, что распознать некоторые типы таких сообщений возможно с применением методов теории подобия конечных последовательностей.

Введение

Нередки ситуации, когда на вход информационной системы (ИС) подаются сообщения, содержащие искажения. Эти ситуации распространены в системах, где источником формирования сообщений является человек. Ниже мы рассмотрим следующие примеры подобной “неустранимой неоднозначности” сообщений в ИС:

- случайные ошибки в адресах e-mail, нуждающиеся в исправлении;
- намеренные искажения электронных адресов в коммерческих целях;
- искажения в запросах интернет-ресурсов, вносимые с целью обхода систем фильтрации;
- поиск в сетях и базах данных по “нечеткому” запросу.

Как мы увидим, общим для всех перечисленных примеров является то, что искажения сообщений, критичные для их идентификации ИС, не являются таковыми для распознавания этих же сообщений человеком.

Мы предлагаем использовать для идентификации сообщений, нечеткость которых связана с описанной выше “неустранимой неоднозначностью”, методы теории подобия конечных последовательностей (ТПКП).

Кратко о теории ТПКП

ТПКП предлагает меры, оценивающие подобие конечных иерархических систем с линейным порядком на каждом уровне иерархии [1; 2]. Таковыми являются, в частности, натуральные или искусственные языки,

исходящие из некоторого алфавита M_0 . Из элементов M_0 ("букв") строятся "слова" – конечные последовательности, ограниченные некоторыми спец-символами из M_0 . Далее строятся "предложения" как "слова второго уровня", состоящие из "букв-слов"; и "тексты", буквами которых выступают предложения.

Меры ТПКП позволяют выявлять подобие указанных последовательностей, то есть давать количественную оценку их "сходства", или "близости". Принимается, что на M_0 определена двухместная функция $\approx(x,y)$ со значениями в $[0,1]$, которая называется "функцией подобия (сходства)". Если $\approx(x,y)=p$, говорят, что x и y "подобны на уровне p ". При этом $\approx(x,x)=1$, т.е. тождественные символы подобны на уровне 1. Но допустимо и $\approx(x,y)>0$ при $x \neq y$, так, для русских букв e и \ddot{e} можно задать ненулевой уровень подобия, скажем $\approx(e, \ddot{e})=0.8$. Аналогично можно положить, скажем $\approx(v, \phi)=0.4$.

Функция $\approx(x,y)$ служит базисом для введения различных численных мер подобия для объектов следующих уровней иерархии ("предложений", "текстов" и т.д.). Некоторые из этих мер используют дополнительное предположение о различной значимости, или "весе", букв в слове либо слов в тексте. Такова, в частности, так наз. мера *взвешенного узкого подобия* G_s . Она оценивает сходство слов, отпрявляясь от их *максимально весомой* (значимой) *подпоследовательности общих входящих символов* (сравн. [3]).

В дополнение к мерам подобия, базирующимся на функции $\approx(x,y)$, вводятся численные меры сравнения кодов по их *структуре*. Предполагается, что в предложениях по некоторым основаниям выделяются особые *именные группы* (ИГ). Например, если речь идет о текстах русского языка, это могут быть группа подлежащего и группа сказуемого, etc. Принимается следующий "*принцип сплоченности*" для групп: *слова, образующие группу, обычно соседствуют; и перестановки слов внутри групп "разрушают" структуру текста в меньшей степени, чем чередование слов, принадлежащих разным группам* (сравн. [4]). Предложены меры оценки сходства текстов, комбинирующие меры их лексического "*взвешенного*" подобия с мерами *сплоченности* входящих в текст ИГ. Последние оценивают, насколько узнаваемость ИГ искажается от ее "засорения чужеродными словами".

Доказано, что предлагаемые ТПКП алгоритмы вычисляют степени подобия за полиномиальное время. Таким образом, они достаточно быстро оценивают сходство "длинных" текстов. Сравнение этих мер с другими известными мерами оценивания "схожести" текстов выявило ряд их преимуществ [5].

Искажения сообщений в сетях телекоммуникаций и методы их корректировки

В сети Интернет довольно распространены случаи ошибочного набора электронного адреса. Опечатки такого рода при записи URL приводят к потере потенциальных клиентов сайта, а при пересылке писем (e-mail) – к недоставке письма адресату.

Для ряда разработчиков исследование "проблемы опечаток" в сети может представлять коммерческий интерес. Так, в [6] рассматриваются вопросы, связанные с привлечением посетителей на сайт за счет использования опечаток при ручном наборе URL адреса. Для этого регистрируется домен, близкий по названию с популярным доменом (его адрес содержит наиболее вероятные ошибки при наборе). Такой метод получил название "type-in-traffic".

Не менее "популярны" и ошибки запросов в поисковых системах. Они также используются для привлечения клиентов на свой сайт [7]. Статистика поисковых систем говорит о большом количестве орфографических ошибок или опечаток вида: А) искажения символа (нескольких символов), В) пропуска символа(ов), С) вставки "лишнего" символа(ов), D) перестановки нескольких символов. В поисковых системах применяются меры, направленные на устранение таких ошибок. При вводе "искаженного" текста система предлагает пользователю возможные варианты "правильных" слов из словаря. Это снижает нагрузку на поисковый сервер и улучшает качество предоставляемых им услуг.

Ниже предлагаются методы распознавания искаженных сообщений, базирующиеся на мерах подобия "текстов", предлагаемых ТПКП.

Коррекция адресов электронной почты. Сейчас здесь применяют следующие приемы обработки "неправильных" адресов. Агент передачи сообщений МТА (Mail Transfer Agent) при отсутствии указанного отправителем адресата формирует возвращаемое письмо (bounce message) с описанием причины, по которой письмо не могло быть доставлено. Bounce message отсылается обратно отправителю.

В традиционной почтовой связи существует требование принятия всех мер по доставке письма адресату, в т. ч. и по выяснению его нового места пребывания. Качество сервиса электронной почты должно быть, по крайней мере, "не хуже". Ввиду этого можно предложить внедрение в МТА модуля, позволяющего выполнять коррекцию ошибочного или же поиск "близкого" адреса. В случае возможности однозначного определения адресата, модуль позволит выполнить корректировку адреса и доставку письма. При невозможности однозначно определить адресата в bounce message может быть включен перечень "близких" адресов.

Рассмотрим пример оценок близости адресов e-mail, полученных при применении упомянутой выше меры подобия $G_s(a, b)$ ТПКП. В качестве a возьмем `n_severin@ukr.net`. В качестве b берем различные искаженные адреса. Оценку близости выполним только для первой части адреса (до символа `@`), поскольку здесь подразумевается, что сообщение находится на сервере адресуемого домена, и на почтового агента возложена задача только поиска адресата в своем домене. Веса всех символов в сравниваемых адресах вначале примем равными 1.

Таблица 1. Оценка близости электронных адресов мерой подобия G_s

Искаженный адрес b	$G_s(a, b)$	Искаженный адрес b	$G_s(a, b)$
nseverin	0,889	n sveerin	0,889
m severin	0,889	n siverin	0,889
n-severin	0,889	n sivirin	0,778
nv severin	0,900	n civerin	0,778

Из табл. 1 видно, что незначительные искажения (одиночные ошибки, перестановки символов) "мало" искажают электронный адрес. Установив порог меры близости, при котором "близкие" к правильному адресу допустимо считать результатами "опечаток", равным 0.88, первые 6 из приведенных адресов можно автоматически "исправить". Конечно, на практике все адреса в приведенном списке могли бы принадлежать другим пользователям сети. Если такая ситуация считается высоковероятной, первые 6 адресов можно включить в bounce message.

Таким образом, bounce message становится более информативным. В итоге применение методов ТПКП позволит как повысить качество работы сервиса, так и снизить "ошибочный" трафик сети.

Как отмечалось выше, мера $G_s(a, b)$ позволяет учесть различную *относительную значимость* различных частей электронного адреса. Например, можно считать, что пропуск/замена символа `@` в адресе e-mail является более "грубой" ошибкой, чем пропуск/замена какого-либо иного символа. Если, допустим, приписать `@` вес 5, а всем остальным символам адреса – вес 1, мера узкого подобия $G_s(a, b)$ даст следующие значения подобия адресов $b = n_severi@ukr.net$ и $c = n_severin_ukr.net$ относительно правильного адреса $a = n_severin@ukr.net$: $G_s(a, b) \approx 0.952$; $G_s(a, c) \approx 0.762$. Итак, адрес c с заменой `@` знаком подчеркивания, с "точки зрения" меры $G_s(a, b)$, намного менее похож на правильный адрес, чем адрес c с пропуском отличной от `@` буквы.

Распознавание "злонамеренных искажений" в запросах интернет-ресурсов. Намеренные искажения запросов с целью обхода систем фильтрации, препятствующих получению доступа к ресурсам, содержащим порнографический, экстремистский или иной подлежащий запрету кон-

тент, являются широко распространенным явлением. Это явление подробно обсуждается авторами в статье [8], поэтому здесь мы ограничимся краткой характеристикой особенностей применения мер ТПКП для борьбы с ним.

Во-первых, отметим: распознавание запросов запрещенного контента, как правило, предполагает различия в значимости разных частей слов, текстов и т.п. Так, часть *порно* в слове *порнография*, очевидно, более значима. Таким образом, здесь уместно применение взвешенных мер подобия, вроде *Gs*.

Полезным может оказаться использование упоминаемой выше особенности функции подобия $\approx(x,y)$: возможности $\approx(x,y)>0$ при $x\neq y$. Так, если задать $\approx(o,0)=0.9$, подобие слов *порно* и *п0рн0* (часто встречающаяся в реальных запросах подмена буквы *o* нулем) будет равно 0.96.

Когда запрос выражается словосочетанием, уместно использовать меры ТПКП, оценивающие *сплоченность* этого словосочетания. Например: *девочки по вызову* – если в контенте интернет-ресурса эти слова *не* соседствуют, его запрещенность является сомнительной.

В [8] показано, что применение мер ТПКП позволяет существенно сократить объемы "черных списков", применяемых в системах фильтрации нецелевого контента.

Поиск информации по "нечетким" запросам. Задача нечеткого поиска становится все более востребованной в системах телекоммуникации [9; 10]. В современных ИС часто возникает задача определения схожести полученного сообщения с некоторым имеющимся набором сообщений. Так, по статистике Яндекса, около 12% поисковых запросов содержат ошибки и опечатки [11]. "Умная" система могла бы предложить пользователю коррекцию его нечеткого запроса.

Задачи, предполагающие оценку подобия сообщений некоторому "образцу", ставятся и решаются в рамках различных ИС. Пример: браузер Chrome от корпорации Google Inc обрабатывает информацию, вводимую пользователем в адресную строку, и, используя асинхронный теневой режим, делает запрос к серверу, получая в ответ набор предполагаемых адресов, в т.ч. варианты коррекции исходного адреса [12]. Такой подход уменьшает число ошибочных запросов и тем самым снижает нагрузку как линии связи, так и серверного оборудования.

Для оценки сходства текстовых сообщений применяются различные методы, основанные обычно на тех или иных функциях расстояния (метриках). В зависимости от области использования выбирается тот или иной метод, либо используется комбинация некоторых методов.

Рассмотрим результаты применения ряда методов оценки подобия сообщений в некоторых задачах нечеткого поиска ИС, и сравним их с оценками, полученными посредством использования меры G_s ТПКП.

Целью системы нечеткого поиска может являться как предоставление пользователю доступа к ресурсам, содержащим "похожие" на его запрос сообщения, так и *запрет* такого доступа – когда задачей системы контроля является сравнение запроса с запрещенными текстами или адресами. Ниже рассматривается второй случай. Для соблюдения приличий приводимые примеры не относятся к запрещенному контенту, но они достаточно наглядно показывают характер проблем, возникающих в данной области. Для сравнения возьмем три простых сообщения:

$$a = \textit{fastmail} \quad b = \textit{emailing}, \quad c = \textit{fastfood}$$

Мы будем считать *mail* "запрещенным" буквосочетанием и, следовательно, a – запрещенным сообщением. Предположим, что b и c – сообщения, поступившие на вход некоей системы нечеткого поиска.

Расчет подобия и сравнение полученных результатов выполним для метрик Хемминга [3], Левенштейна [3], функций Soundex и Metaphone (они оценивают *фонетическое* сходство звучания слов – см. [13]), а также для меры G_s . Поскольку ТПКП предлагает меры не для расстояния, а для подобия сообщений, ниже на базе перечисленных функций расстояния вводятся соответствующиеим функции подобия $H(a, b)$, $L(a, b)$, $S(a, b)$ и $M(a, b)$.

Учитывая, что в сообщении $a = \textit{fastmail}$ подстрока *mail* считается запрещенной, т.е. более значимой, установим веса (они отображены верхними индексами) следующим образом: $f^0 a^0 s^0 t^0 m^1 a^1 i^1 l^1$. Результаты оценок подобия будут следующими:

Таблица 2. Оценки подобия для сравниваемых методов

по- добие	H	L	S	M	G_s
a и b	0,000	0,125	0,250	0,400	0,667
a и c	0,500	0,500	0,750	0,600	0,250
b и c	0,000	0,000	0,000	0,000	0,222

Таким образом, сравнительный анализ показывает перспективность применения метода взвешенного подобия G_s : его оценки оказались наиболее приближены к "человеческим".

Заключение

Метод ТПКП эффективны для оценки подобия сообщений с искажениями типов Левенштейна-Дамерау, в ситуациях значительных смещений суб-сообщений, а также с учетом различной "смысловой значимости" суб-сообщений. Перспективно их применение в сервисах телекоммуни-

кационных систем, выполняющих динамический контроль нецелевого трафика, идентификацию полученного электронного адреса, поиск информации в базах данных, и других системах, где оправданно использование нечеткого поиска.

Литература

1. *Леоненко Л.Л., Поддубный Г.В.* Теория подобия конечных последовательностей и ее приложения к распознаванию образов // Автоматика и телемеханика. – 1996. – №8. – С. 119 – 131.
2. *Леоненко Л. Л.* Алгоритмы оценки аналогичности текстов и их применение в компьютерном тестировании // Сб. трудов VII международной конф. "Интеллектуальный анализ информации" (ИАИ-2007). – К.: Просвіта, 2007. – С. 210 – 220.
3. *Смит У.* Методы и алгоритмы вычислений на строках. – М.: "И.Д. Вильямс", 2006. – 496 с.
4. *Гладкий А.В.* Математические методы изучения естественных языков // Труды МИАН им. В.А.Стеклова. – 1973. – Том 133. – С. 95 – 108.
5. *Северин Н.В.* Методы нечеткого поиска в системах контроля нецелевого контента // Вісник Східноукраїнського національного університету ім. В. Даля. – 2012. – №8 (179), Ч. 2. – С. 199 – 205.
6. *Жарков С.* Кривые пальцы, приносящие трафик // www.linkz.ru/promotion/krivuee_palmztcue_prinosyascie_trafik.html
7. *Кокшаров С.* Роль опечаток в SEO // <http://devaka.ru/articles/seo-misprints>
8. *Северин Н.В., Леоненко Л.Л.* Теория подобия конечных последовательностей в задачах фильтрации Интернет-контента // Сб.трудов XIII международ. конф. "Интеллектуальный анализ информации" (ИАИ-2013). – К.: Просвіта, 2013, с. 285 – 291.
9. *Тодорико О.А., Добровольский Г.А.* Оценка сигнатурных алгоритмов поиска по сходству в словаре // Вестник ХНТУ – 2011. – №2(41) – С. 250 – 254.
10. *Ломакин А.А.* Совершенствование методов идентификации персональных данных в автоматизированной информационной системе «Электронный социальный регистр населения» Санкт-Петербурга // Электронный научный журнал «Исследовано в России» – С. 156 – 163; <http://zhurnal.ape.relaru/articles/2011/015.pdf>
11. Пресс-релизы Яндекса за 2011 год: Яндекс отвечает на два вопроса // http://company.yandex.ru/press_releases/2011/0902/index.xml.
12. Google Chrome // <https://www.google.com/chrome?hl=ru>
13. Фонетические алгоритмы // <http://habrahabr.ru/post/114947/>

УДК 004.596

Застосування паралельних обчислень для побудови зростаючих пірамідальних мереж

*Кк.т.н., доц. Величко В.Ю., к.т.н., Сирота С.В., Ліскін В.О.
Національний технічний університет України «КПІ», м. Київ*

В даній роботі запропоновано модель представлення знань в інтелектуальних системах на основі зростаючих пірамідальних мереж (ЗПМ) з використанням паралельних обчислень. Та приведено експериментальні результати роботи. Розглянуто основні операції ЗПМ та запропоновано алгоритм та їх реалізацію з використанням паралельних обчислень.

Вступ

Постійне збільшення обсягів наявної інформації робить майже неможливим її ручну обробку й освоєння людиною та висуває вимоги щодо створення комп'ютерних засобів для автоматизації подання, збереження, систематизації, пошуку та іншої обробки інформації [1]. При цьому домінуючою тенденцією у розвитку інтелектуальних систем, є використання у комп'ютерах природних притаманних людині принципів моделювання середовищ, ситуацій, задач. Особлива потреба виникає у створенні систем, що здатні узагальнювати та аналізувати інформацію, адаптуватися до її змін, спілкуватися з людиною-користувачем природною мовою, приймати рішення в умовах неповної, ненадійної та суперечливої інформації [1]. З цієї причини дуже важливим є підвищення рівня інтелектуалізації комп'ютерних систем, використання логіко-лінгвістичних моделей в задачах логічного аналізу даних та знань [2]. Одним із підходів до розширення існуючих моделей знань є зростаючі пірамідальні мережі.

Визначення

Зростаючою пірамідальною мережею [3] називається ациклічний орієнтований граф, в якому немає вершин, що мають одну дугу, що заходить. Вершини, які не мають дуг, що заходять, називаються рецепторами, решта вершини – концепторами. Підграф пірамідальної мережі, що включає вершину a і всі вершини, від яких є шлях до вершини a , називається пірамідою вершини a . Вершини, що входять в піраміду вершини a , утворюють її субмножину. Множина вершин, до яких є шляхи від вершини a ,

називається її супермножиною. У субмножині та супермножині вершин виділяються 0-субмножину і 0-супермножину, що складаються з тих вершин, які пов'язані з нею безпосередньо.

Рецептори відповідають значенням ознак. Концептори відповідають описам об'єктів в цілому і перетинів описів або ситуацій, які пов'язані з певним набором рецепторів. Вхідний інформацією служать набори значень ознак, на виході можна визначити ситуацію, яка відповідає даній комбінації ознак.

Побудова ЗПМ на основі паралельних обчислень

В [4] наведено алгоритм паралельної побудови ЗПМ. Побудова ЗПМ складається з двох основних етапів. Перший етап складається з пошуку перетинів значень ознак об'єктів. На другому етапі відбувається впорядкування знайдених концепторів, які відповідають перетинам значень ознак об'єктів.

На першому етапі для кожного нового об'єкта a_i , який додається до мережі, та об'єкта $a_j, j = 1, l - 1$, який вже існує в мережі, визначається множина рецепторів $R_{c_k^l}: R_{c_k^l} = R_{a_j} \cap R_{a_i}$, яка відповідає деякому концептору c_k^l . Якщо концептор c_k^l , який відповідає перетину множин рецепторів об'єктів, існує в мережі, то він додається до 0-субмножини об'єкта a_i . В іншому випадку у мережу вводиться новий концептор c_k^l , який з'єднується дугами, що виходять з вершин множини $R_{c_k^l}$ та дугами, що заходять до вершин a_i та a_j . Дуги від рецепторів з 0-субмножини нового концептора, які безпосередньо йшли до вершин a_i та a_j розриваються. Для кожного концептора c_k^l визначається його рівень l у мережі, який дорівнює кількості рецепторів з його 0-субмножини:

$$l = \text{Card} \left(R_{c_k^l} \right).$$

На другому етапі структурування мережі виконується перевірка пірамід концепторів з множини $C_Q \setminus C_A$ на вкладеність за множиною рецепторів. При обробці кожного концептора c_k^l починаючи з рівня $l = (|R_a| - 2)$ до рівня $l = 2$ його рецептори $R_{c_k^l}$ переводяться в стан збудження $-R_{c_k^l}^+$. Збудження розповсюджується по мережі та для кожного частково збудженого концептора $\forall c_m^n \left[l < n < (|R_a| - 1), 0 < m < |C_Q| \right]$,

який має шляхи до збуджених рецепторів, підраховується ступінь збудження за формулою $|R_{c_m^n}^+|$. Концептори, які входять до супермножини концептора c_k^l не розглядаються. Якщо ступінь збудження частково збу-

дженого концептора c_m^n дорівнює l , концептор c_k^l додається до 0-субмножини концептора c_m^n . На цьому етапі використання асоціативних властивостей ЗПМ дозволяє зменшити кількість операцій порівнянь концепторів [4].

Запропонований спосіб дозволяє виконати розподілену реалізацію побудови мережі. За рахунок використання розподіленої реалізації та розширенню можливостей мережевих структур розширюється простір реальних складних задач, де можливе використання зростаючих пірамідальних мереж. А також задачі більшого об'єму будуть розв'язуватися за прийнятний час за допомогою багатопотокових комп'ютерних систем.

Для реалізації паралельних обчислень були використані бібліотеки OpenMP.

За допомогою даних бібліотек було реалізовано паралельний пошук перетинів рецепторів, тобто пошук концепторів. Використання паралельних обчислень реалізовано за допомогою директиви `#pragma omp parallel` з ініціалізацією та вказанням конкретних дій для кожного потоку.

В загальному випадку, алгоритм пошуку перетинів на основі паралельних обчислень можна описати наступним чином:

- 1) аналіз кількості можливих або необхідних потоків;
- 2) паралельний запуск N потоків;
- 3) ініціалізація (створення списку рівнів та виділення пам'яті для нього в кожному потоці);
- 4) виконання кожним потоком своєї задачі по знаходженню спільних рецепторів та запис їх в список рівнів потоку;
- 5) після завершення виконання всіх потоків, відбувається злиття та перетворення списків рівнів кожного потоку в список концепторів.
- 6) звільнення виділеної пам'яті.

Вказання конкретних дій та використання `#pragma omp parallel`, визвано тим, що дані в пам'яті зберігаються у вигляді списків, а не масивів, тому використання `#pragma omp for` не можливе для списків.

Список рівнів потоків оптимізує та пришвидшує пошук спільних концепторів за допомогою спеціальної організації пам'яті для зберігання концепторів «по полицкам» та в відсортованому порядку.

Результати тестування

Для того, щоб обґрунтувати коректність роботи реалізованих алгоритмів та доцільність і ефективність запропонованої моделі в рамках даної роботи проведено декілька видів практичних експериментів:

- 1) побудова ЗПМ без паралельних обчислень з використанням одного потоку;
- 2) побудова ЗПМ на основі паралельних обчислень з використанням двох потоків;
- 3) побудова ЗПМ на основі паралельних обчислень з використанням чотирьох потоків.

Вхідні файли для тестування були взяті з бази даних *uci machine learning data* [5].

На рисунку 1 показано результати побудови ЗПМ в секундах в залежності від кількості використовуваних потоків.

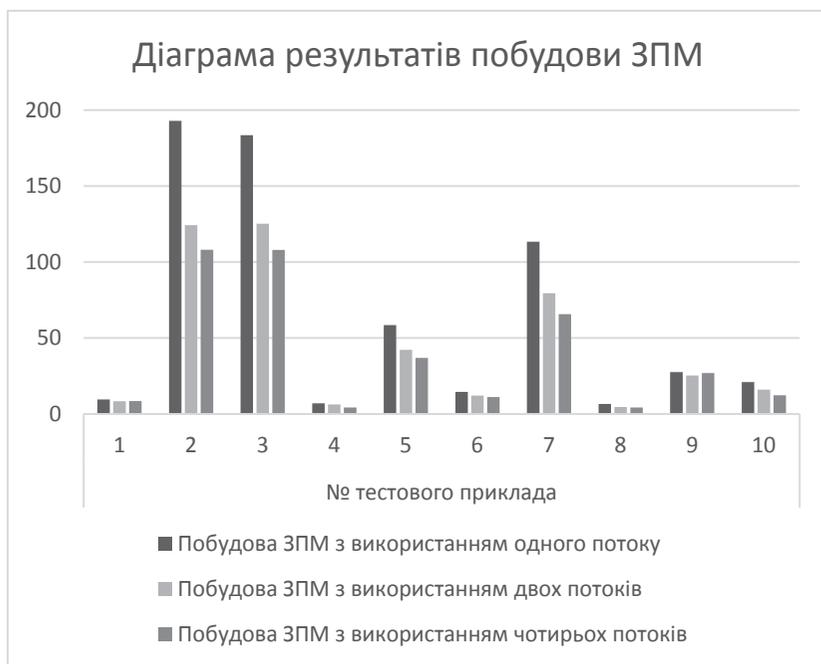


Рис.1. Діаграма результатів побудови ЗПМ

Експериментальні дослідження, представлені в цьому розділі показали швидкодії алгоритму побудови зростаючих пірамідальних мереж за рахунок паралельних обчислень.

Незалежно від кількості задіяних потоків, алгоритм що використовує паралельні обчислення проявляє себе краще ніж алгоритм побудови ЗПМ без паралельних обчислень.

Висновки

Основним результатом даної роботи є введення принципово нової моделі з використанням паралельних обчислень для побудови зростаючих пірамідальних мереж.

За результатами виконаної роботи можна зробити наступні висновки:

- при малій кількості вхідних даних використовувати паралельні обчислення недоцільно;
- швидкість побудови ЗПМ з'являється при великій кількості даних та використанні паралельних обчислень;
- швидкість побудови мережі збільшується приблизно в 1,4–2 рази при використанні декількох потоків.
- використання паралельних обчислень для побудови зростаючих пірамідальних мереж дозволяє збільшити розмір оброблюваних даних для побудови мережі за заданий час.

Таким чином, модель побудови зростаючих пірамідальних мереж на основі паралельних обчислень є більш ефективною. Це підтверджують експериментальні результати.

Література

1. Суббогін С.О. Подання й обробка знань у системах штучного інтелекту та підтримки прийняття рішень: Навчальний посібник, - Запоріжжя: ЗНТУ, 2008. – 341 с.
2. Поспелов Д.А. Логіко-лінгвістичні моделі в системах керування.- Москва: Энергоиздат.- 1981.
3. *Гладун В.П.* Партнерство с компьютером. Человеко-машинные целеустремленные системы. К. : Port-Royal. 2000, – 128с.
4. Величко В. Ю. Алгоритм побудови зростаючих пірамідальних мереж у паралельному обчислювальному середовищі. – «Комп'ютерні засоби, мережі та системи №10», 2011. – 50-57 с.
5. [Електронний ресурс]. – Режим доступу : URL : <http://archive.ics.uci.edu/ml/>

УДК 681.3:004.7

Эффект «взрывной перколяции» в контентной сети

*Снарский А.А.^{1,2}, д.ф.-м.н, проф., Ландэ Д.В.^{2,1}, д.т.н., с.н.с.
1НТУУ «Киевский политехнический институт», Киев, Украина
2Институт проблем регистрации информации НАН Украины,
Киев, Украина
dwlande@gmail.com, asnarskii@gmail.com*

Предлагается метод построения контентной сети (CNC, Content Network Connections), являющийся развитием алгоритма «взрывной перколяции». Изучается поведение данной растущей сети, а также фазовый переход первого рода, возникающий при построении такой сети.

Введение

Исследование динамики сложных сетей (Complex Networks) [1-2], изменения количества связей, уровней кластерности, других параметров, является актуальной научной и практической задачей, так как большинство реальных сетей (Интернет, WWW, социальные, биологические и др. сети) являются динамичными, растущими. Для таких сетей, в которых течением времени увеличивается число узлов и/или связей, очень важным является эффект возникновения так называемого гигантского кластера (GC, Giant Cluster), при котором количество узлов самого большого кластера в ней становится равным по порядку общему количеству узлов. Условия возникновения GC и его свойства хорошо изучены для перколяционных [3] и некоторых других сложных сетей, например, для сетей Эрдоша-Реньи. В частности, для таких сетей условие возникновения GC определяется степенями узлов и задается критерием Моллоя-Рида [4]: $\langle k^2 \rangle - 2\langle k \rangle \geq 0$, где k – степень узла, а $\langle \dots \rangle$ означает усреднение по всем узлам.

На фазе возникновения и роста GC, как в перколяционных, так и в ER и в других сложных сетях, например, в безмасштабных (SF, Scale Free Networks) [1], некоторые параметры этих сетей (например, размеры максимальных групп связанных узлов – кластеров), ведут себя аналогично параметрам порядков в теории фазовых переходов второго рода [5].

«Взрывная перколяция»

В работе [6] был предложен алгоритм построения растущей сети, который приводит к так называемой взрывной перколяции (EP, Explosive Percolation). В отличие от алгоритма Эрдоша-Реньи, т.е. стандартного случайного порождения (путем «набрасывания») новых связей, этот алгоритм предполагает альтернативу. В соответствии с ним на каждом шагу случайным образом порождается не одна, а две связи, из которых только одна остается (реализуется), а вторая отбрасывается. Критерий отбора реализованной связи определяется сравнением произведений размеров кластеров (количества узлов), соединяемых связями. Из двух выбирается та связь, для которой это произведение наименьшее. Как оказалось, при реализации такого алгоритма, формирование и рост GC принципиально отличается, его размер изменяется скачкообразно, аналогом чего является фазовый переход первого рода, а не второго.

Модель контентной сети

Авторами предлагается следующая модель контентной сети. Предполагается, что сеть состоит из N узлов. Существует M различных документов, отдельные экземпляры которых (копии) распределены в узлах контентной сети. Пусть i -й узел сети содержит n_i различных документов, причем только по одному экземпляру каждого из них. Узлы можно интерпретировать как библиотеки, веб-сайты, файловые хранилища. Соответственно, документы можно интерпретировать как книги, веб-страницы, файлы, фрагменты файлов и т.п. Предполагается, что распределение количества документов по узлам соответствует закономерности Парето (аналогично распределению богатства в социальных системах), т.е. степенное.

В этом случае для построения растущей сети предложен новый алгоритм порождения связей в контентной сети (CNC, Content Network Connections), являющийся развитием алгоритма EP. В соответствии с ним на каждом шагу случайным образом из всего многообразия документов задаются два различных документа, а затем случайным образом выбираются две пары узлов, содержащих данные документы. В данном случае если хотя бы один выбранный узел не содержит необходимого документа, то происходит случайный выбор следующего узла. Эта процедура выполняется циклически до достижения цели, естественно, должны учитываться условия выхода из «зацикливания», например, путем перехода к выбору другой пары документов.

После этого между выбранными узлами, содержащими разные документы, также как и в [6] устанавливаются две связи, из которых также остается (реализуется) только одна, а вторая отбрасывается. Критерий отбора реализованной связи в этом случае определялся сравнением общего количества документов, содержащихся в кластерах, узлы которых соединяются связями с выбранными узлами. Из двух связей выбирается та, для которой сумма документов в соответствующих кластерах оказывается наименьшей. Таким образом, документы являются основой построения связей между узлами-библиотеками. На рис.1 представлена абстрактная модель сети из 12 узлов-библиотек, содержащих документы, обозначенные различными фигурами (закрашенными и не закрашенными треугольниками, квадратами, ромбами и т.д.).

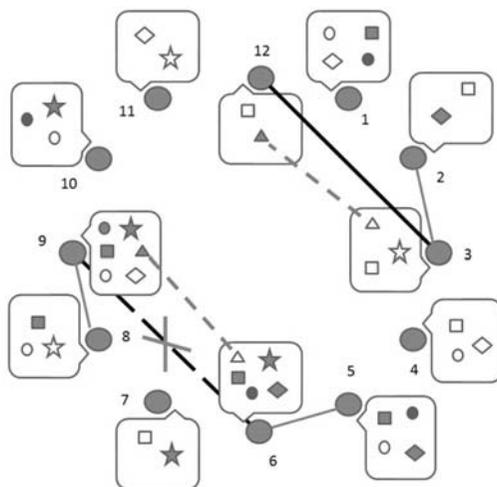


Рис. 1 – Установление связи в модели контентной сети

Здесь первое условие, задаваемое при построении новой связи, заключается в установлении связей между узлами, содержащими документы-треугольники – закрашенный и не закрашенный. Случайным образом выбраны две пары узлов, содержащих требуемые документы: первая – 12 и 3, вторая – 6 и 9. Очевидно, кластер первой пары, содержащий документы из 12-го, 3-го и 2-го узлов (всего 7 документов), значительно меньше по количеству документов кластеру второй пары, содержащей документы из 6-го,

9-го, 5-го и 8-го узлов (всего 18 документов). Таким образом остается только первая связь (между 12-м и 3-м узлами), а вторая отбрасывается.

Выбор двух документов в алгоритме CNC может интерпретироваться, например, как установление связей между библиотеками при подборе многотомного издания, или как сборка целого файла при объединении различных фрагментов, расположенных в различных узлах пириновой сети.

Изменение размера гигантского кластера

Авторами были проведены эксперименты для различных N и M с целью определения зависимости числа узлов (библиотек) наибольшего кластера p от числа связей для предложенного алгоритма CNC. На рис. 2 показана типичная зависимость для эталонного примера, соответствующего алгоритму Эрдоша-Реньи, когда устанавливается связь между первыми выбранными узлами, содержащими требуемую пару документов (рис 2a), и для CNC (рис 2b) при $M = N = 250$.

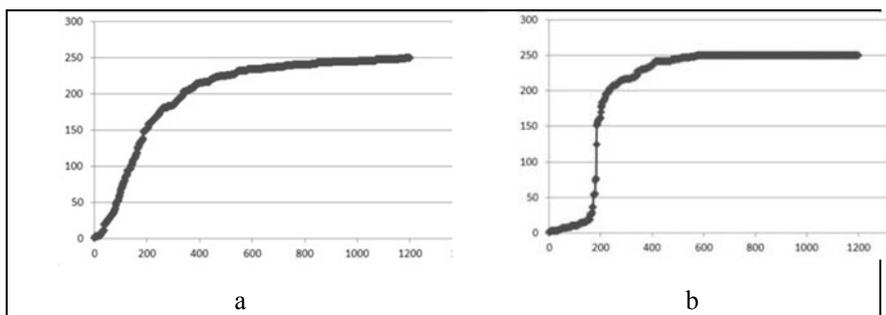


Рис. 2 – Зависимость размера наибольшего кластера от числа связей при $M = N = 250$

Оценка величины скачка

Как видно на рис. 2, при определенном значении p , в случае предложенного алгоритма CNC происходит скачкообразный рост наибольшего кластера. Усредненная по реализациям зависимости величины скачка η от

размера контентной сети N в случае $M = N$ приведена на рис. 3. Как показывают измерения, тренд этой зависимости близок к линейному.

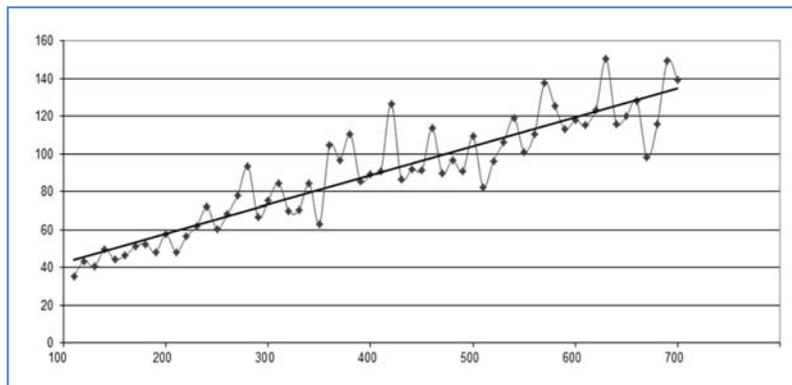


Рис. 3 – Зависимость величины скачка η от размера контентной сети N

Пусть C – размер самой большой компоненты связности, t_0 – наибольшее количество связей в растущей контентной сети, при которых $C < \sqrt{N}$, а t_1 – наименьшее количество связей, при которых $C > N/2$. В случае непрерывных переходов, разность $D = t_0 - t_1$ линейно зависит от N , в частности, $D \approx 0.19N$ для сетей Эрдоша-Реньи. В отличие от данного поведения, в случае CNC зависимость не является линейной, а может аппроксимироваться зависимостью $D \sim AN^\alpha$, ($A \approx 0,84$, $\alpha \approx 0,79$), что соответствует поведению, описанному в [6].

Выводы

При реализации алгоритма CNC, образование и рост GC соответствует поведению, наблюдаемому при реализации алгоритма EP, его величина также изменяется скачкообразно, что также соответствует фазовому переходу первого рода, т. е. так называемой «взрывной перколяции». Как и в [6] небольшие изменения в правилах формирования связей приводит к принципиально новому характеру в протекании процесса формирования сети, что необходимо учитывать в процессе формирования и управления контентными сетями.

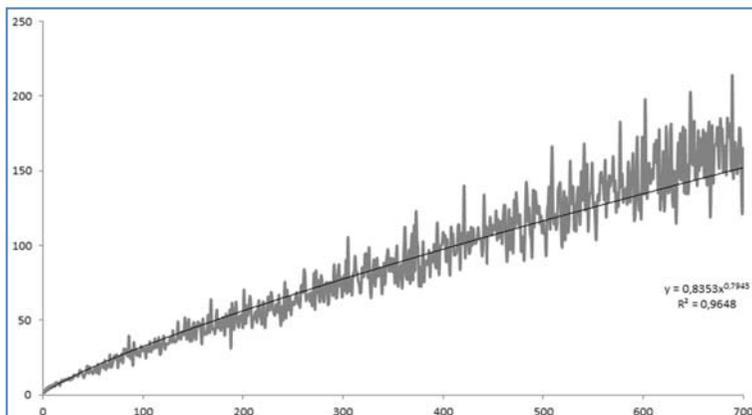


Рис. 4 – Зависимость величины D от размера контентной сети N

Литература

1. *Albert R., Barabási A.-L.* Statistical mechanics of complex networks // *Rev. Mod. Phys.*, 2002 – 74. – P. 47-97.
2. *Dorogovtsev S.N., Mendes J.F.F.* Evolution of Networks. From Biological Nets to the Internet and WWW. – Oxford: Oxford University Press, 2003. – 280 pp.
3. *Stauffer D., Aharony A.* Introduction to Percolation Theory. London: Taylor&Francis, 1992. – 180 pp.
4. *Dorogovtsev S.N., Goltsev A.V., Mendes J.F.F.* Critical phenomena in complex networks // *Rev. Mod. Phys.*, 2008. – 80. – P. 1275-1335.
5. *Ландау Л.Д., Лифшиц Е.М.* Статистическая физика. Ч.1. – М.: Наука, 1976. – 584 с.
6. *Achlioptas D., D'Souza R.H., Spenser J.* Explosive Percolation in Random Networks // *Science*, 2009. – 323. – P. 1453-1455.

УДК 004.42

Усунення мовної неузгодженості в тестових завданнях, згенерованих на основі понятійно-тезисної моделі

*Танченко С. С., к.т.н., доц. Титенко С. В., к.т.н., доц. Гагарін О.О.
Національний технічний університет України «КПІ», м. Київ
serg.tanchenko@gmail.com*

У роботі запропоновано розширення понятійно-тезисної моделі (ПТМ) з метою подолання проблеми мовної неузгодженості в тестових завданнях, що будуються автоматично. Запропоновано використання узагальненого мовного класифікатора як додаткової характеристики структурних елементів ПТМ. Удосконалено алгоритм побудови тестових завдань на базі ПТМ.

Вступ

В умовах стрімкого розвитку технологій та неспішною зміною університетських програм, значну частину професійних знань фахівцеві отримує не в межах навчального закладу. Із підвищенням ступеня інформатизації суспільства та освіти проблема ефективного контролю знань набуває особливої значущості. Однією з найбільш поширених форм перевірки знань є комп'ютерне тестування. У попередній роботі [1] було здійснено огляд методів генерації тестових завдань та обґрунтовано ПТМ як доцільний напрямок з точки зору реалізації та практичного використання. Завданням даної роботи є розширення семантики ПТМ для усунення проблеми мовної неузгодженості тестових завдань, що часом виникають при генерації тестів.

Проблема мовної неузгодженості в тестових завданнях

Побудова тестових завдань на основі ПТМ відбувається шляхом вибору із сукупності понять та тез певного набору елементів, що стають структурними одиницями тестового завдання [2].

Розглянемо проблему мовної неузгодженості тестів, згенерованих на основі ПТМ. Проблема полягає у невідповідності роду або числа понять та тез, що беруть участь у конструюванні тестового завдання. Це суттєво знижує якість завдання, а також ефективність перевірки засвоєння матеріалу. Нижче наведені приклади згенерованих завдань із мовними неузгодженостями:

1. У чому полягає сутність поняття *session.name*?
 - i) **являють** собою розширення протоколу HTTP;
 - ii) повертає ім'я поточної сесії;
 - iii) є інформаційною моделлю предметної області;
 - iv) являє собою рядок з 32 символів.
 2. До якого поняття наступне твердження підходить найбільше?
Найбільш очевидно вона проявляє себе в систематичному переповненні каналів грошового обігу масою надлишкових грошей, що веде до їх знецінення:
 - a) конкретно-економічні відносини; b) пропонування;
 - c) **інфляція** d) засіб обігу.
 3. У чому полягає сутність поняття: *Internet*?
 - a) це система, що безперервно розвивається людьми, які використовують її послуги;
 - b) ідентифікує та встановлює наявність передбачуваних партнерів для зв'язку;
 - c) **дозволяла** будь-якому з комп'ютерів зв'язуватись з будь-яким іншим;
 - d) представлений протоколом TCP і UDP;
 - e) можна розділити на мережі з комутацією потоків і мережі з комутацією каналів.
 4. Яка особливість поняття: *канальний рівень*?
 - a) нижче цього рівня тільки апаратний рівень;
 - b) **охопила** більше ніж 100 країн, **об'єднала** більше 40 тис. окремих мереж;
 - c) його політика припускала, що нова мережа буде орієнтована на різноманітних користувачів;
 - d) у 1986 році **зв'язувала** менше 6000 тис. комп'ютерів.
 - e) використовує процес експертної оцінки документів.
- У тесті №2 є очевидним, що лише один варіант відповіді підходить за синтаксисом.

Вирішення проблеми мовної неузгодженості в тестових завданнях, згенерованих на основі ПТМ

Генерація тестів з використанням ПТМ відбувається на основі понятійно-тезисної бази, структурними одиницями якої є: «поняття» та «тези». Позначимо множину понять як:

$$C = \{c_1, \dots, c_{n1}\}$$

Тези – це відомості про поняття. Зазвичай теза – це одне або декілька речень, в яких мова йде про поняття, але словесно поняття не фігурує в ній. Позначимо множину тез як:

$$T = \{t_1, \dots, t_{n2}\}$$

У свою чергу кожне поняття може мати будь-яку кількість тез. Зв'язок між поняттями та тезами:

$$CT: T \rightarrow C, TC: C \rightarrow 2^T$$

ПТ-елементи виділяються із тексту навчального фрагменту, що відноситься до певного предмету. Позначимо множину предметів як:

$$S = \{s_1, s_2, \dots, s_{n4}\}$$

Зв'язок між предметами та тезами: $TS: S \rightarrow 2^T$. Відповідно до цього, поняття предмету визначаються як:

$$CS(s) = \{c | TS(s) \cap CT(c) \neq \emptyset\}$$

Шляхом подолання проблеми мовної неузгодженості є введення додаткової характеристики в опис структурних одиниць ПТМ, що отримала назву «узагальнений мовний класифікатор» (УМК). Множина УМК для поняття:

$$MC: \{mc_1, \dots, mc_{n3}\},$$

де mc_1 – «чоловічий»; mc_2 – «жіночий»; mc_3 – «середній»; mc_4 – «множина». Для кожного поняття та тези додається свій УМК. Особливістю є те, що теза може лексично відповідати поняттям які мають відмінне значення УМК, ніж сама теза. Зв'язок між поняттям і УМК: $MCC: C \rightarrow MC$. Позначимо множину УМК для тез:

$$MT = MC \cup mt',$$

де $mt' = (\{mc_1, mc_2, mc_3\})$ і означає «довільну одиницю». Зв'язок між тезою і УМК як: $MTT: T \rightarrow MT$.

На рис. 1 зображено тестове завдання для поняття «константа» – жіночого роду, проте в якості дистракторів виступають тези як жіночого так і чоловічого родів, наприклад теза поняття «клас».

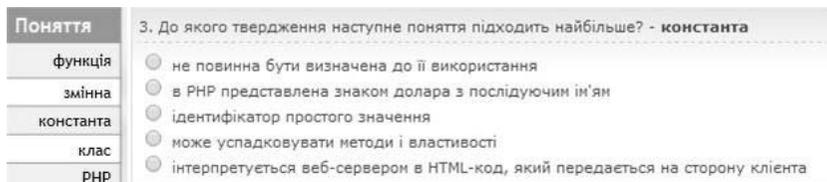


Рис. 1. Згенероване тестове завдання

Розроблено алгоритм побудови тестового завдання $task_k$ на основі ПТ-елементів, що містить наступні кроки:

1) пошук контрольного поняття c_k на основі якого буде згенероване тестове завдання:

- a) вибір випадкового поняття c'_k з контексту контрольного предмету s_k : $c'_k: [c'_k \in C \wedge c'_k \in CS(s_k)]$;
- b) пошук множини альтернативних варіантів відповідей (тез) T'_k , що не відносяться до поняття c'_k :
 - $T'_k = \{t | (t, c'_k) \notin CT \wedge [MTT(t) = MCC(c'_k) \vee MTT(t) \in MCC(c'_k)]\}$
- c) перевірка, чи задовольняє обране поняття вимозі $|T'_k| \geq 4$, тобто контрольним поняттям може бути тільки поняття, ще має 4 або більше альтернативних варіантів відповідей. Якщо умова виконується, прийняти $c_k = c'_k$, інакше повернутись до п. 1.а;

2) вибір контрольної тези: $t_k: t_k \in T_k$, де T_k – множина тез, що відносяться до поняття c_k : $T_k = \{t | (t, c_k) \in CT\}$.

3) візуалізація тестового завдання.

В результаті тестове завдання складається із: $Tasks(task_k) = \langle c_k, t_k, T'_k \rangle$, де: c_k - контрольне поняття; t_k - контрольна теза; T'_k - альтернативні варіанти відповідей.

На рис. 2 показано алгоритм побудови тестового завдання.

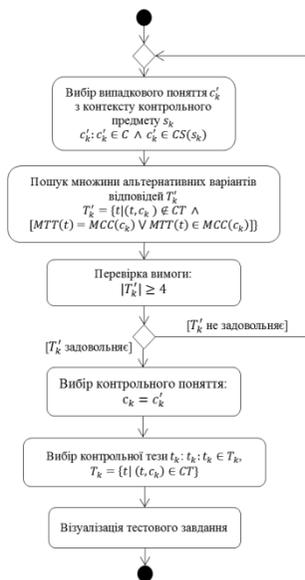


Рис. 2. Діаграма діяльності у нотатції UML “Алгоритм побудови тестового завдання”

Висновки

В роботі було описано метод подолання мовної неузгодженості згенерованих завдань. Суть методу полягає в розширенні онтології на основі понятійно-тезисної моделі шляхом введення в опис структурних одиниць ПТМ характеристики узагальненого мовного ідентифікатора. Удосконалено алгоритм генерації тестового завдання з урахуванням вимог лексичної узгодженості понятійно-тезисних елементів.

Література

1. *Танченко С.С., Титенко С.В.* Аналіз методів генерації тестових завдань // Международная научная конференция имени Т.А. Таран «Интеллектуальный анализ информации ИАИ-2013», Киев, 15-27 мая 2013 г.: сб. тр./ ред. кол.: С.В. Сирота (гл. ред.) и др. – К.: Просвіта, 2013. – С. 218-225.
2. *Титенко С. В.* Генерація тестових завдань у системі дистанційного навчання на основі моделі формалізації дидактичного тексту / С. В. Титенко // Наукові вісті НТУУ "КПІ". – 2009. – № 1(63). – С. 47–57.

УДК 004.93

Построение системы аутентификации операторов и распознавания слов по голосу

*Темников В.А., к.т.н., доц., Темникова Е.Л., Конфорович И.В.
Национальный технический университет Украины «Киевский политехнический институт», Национальный авиационный университет,
temnikov_v@ukr.net*

В статье описаны принципы построения разработанной авторами системы контроля доступа операторов к информационным ресурсам, предназначенной для проведения автоматической дистанционной аутентификации операторов по непрерывной слитной речи. Система может быть также применена для распознавания речевых фрагментов (слов, словосочетаний) для последующего выделения их из непрерывной речи операторов. Обоснованный выбор значений параметров речевого сигнала и искусственных нейронных сетей, на основе которых построена подсистема классификации, позволил существенно ускорить процессы параметризации и классификации речевых сигналов при обеспечении правильной аутентификации операторов выше 98%.

Введение

Одним из направлений контроля доступа операторов к информационным ресурсам является их аутентификация, позволяющая предотвратить занятие рабочего места несанкционированным лицом.

В качестве биометрического признака человека для проведения аутентификации предлагается использовать его голос [1]. Аутентификация осуществляется по непрерывной слитной речи на основе анализа параметров речевого сигнала, фиксируемого во время работы операторов (например, для авиадиспетчеров – их аудиообмена с членами летных экипажей).

В статье представлены принципы проведения аутентификации операторов, а также основных подсистем разработанной авторами системы аутентификации, применение которых обеспечивает дистанционный автоматический контроль операторов в режиме реального времени.

Разработанная система аутентификации оказывает помощь администратору, осуществляющему визуальный контроль за действиями операторов, в предотвращении доступа к информационным ресурсам лиц, занявших рабочее место оператора несанкционированно, путем подачи сигнала при обнаружении возможного нарушения, а также позволяет получить документальное свидетельство о нарушении при расследовании причин аварий, аварийных ситуаций и летных происшествий.

Принципы проведения аутентификации операторов

Дистанционную аутентификацию операторов в процессе контроля их доступа к информационным ресурсам предлагается проводить по речевым фрагментам, выделенным из непрерывной слитной речи оператора. В качестве указанных выделенных речевых фрагментов используются отдельные слова или словосочетания, часто используемые операторами во время работы, а также входящие в состав их профессиональной фразеологии. Такой подход позволяет свести задачу аутентификации человека по непрерывной слитной речи к задаче аутентификации по отдельным командам.

Указанные речевые фрагменты представляются в разработанном словаре в виде векторов параметров.

Аутентификация проводится на основе методов теории распознавания образов с применением разработанной системы параметров речевых сигналов. В статье описаны принципы построения подсистем параметризации и классификации речевых сигналов.

Принципы построения подсистем параметризации и классификации речевых сигналов

Подсистема параметризации строится на основе деления речевых фрагментов, выделенных из непрерывной слитной речи операторов, сканированной в процессе выполнения ими своих профессиональных обязанностей, на фреймы длительностью несколько десятков миллисекунд и определения на каждом фрейме кепстральных коэффициентов.

Ниже описан алгоритм работы подсистемы параметризации и приведены его основные шаги.

Сигнал, поступивший на вход подсистемы параметризации, сначала делится на фреймы длиной примерно 25 мс. На каждом фрейме рассчитывается по 36 кепстральных коэффициентов линейного предсказания (ККЛП), на основе которых строится матрица C_{nm} , где c_{ij} – j -ый ККЛП i -го фрейма, n – количество фреймов, m – количество ККЛП в каждом фрейме.

Так как получаемое в этом случае количество параметров является слишком большим для эффективного применения в подсистеме классификации, построенной на основе искусственной нейронной сети (ИНС), то проводятся следующие преобразования: матрица кепстральных коэффициентов C_{nm} преобразуется в одномерный вектор (на рис. 1 изображен сигнал, составленный из ККЛП), к которому применяется быстрое преобразование Фурье.

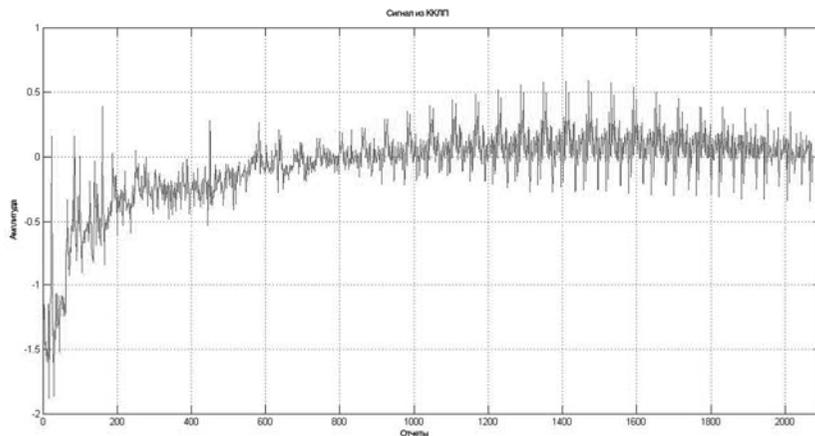


Рис.1. Сигнал, составленный из ККЛП (по оси абсцисс – отсчеты нового сигнала, по оси ординат – амплитуда сигнала)

Результатом преобразований является спектрограмма, представляющая собой зависимость амплитуд спектральных составляющих от частоты и номера ККЛП [2].

Спектрограмма имеет большую размерность, но не все ее части являются одинаково информативными - более информативной является низкочастотная часть спектрограммы. Наиболее же информативным, как показало тестирование, является первый столбец спектрограммы (он соответствует частоте, значение которой есть величина, обратная длительности исходного речевого сигнала).

Использование результирующего вектора параметров (рис.2) позволяет осуществлять аутентификацию человека с высокой степенью достоверности (процент правильной аутентификации достигает 98%), а небольшая размерность вектора параметров при применении в подсистеме классификации ИНС типа многослойного перцептрона – добиться высокой скорости параметризации (порядка 10^{-2} с). Такие высокие показатели достоверности и быстродействия получены благодаря проведенному тестированию подсистем параметризации и классификации.

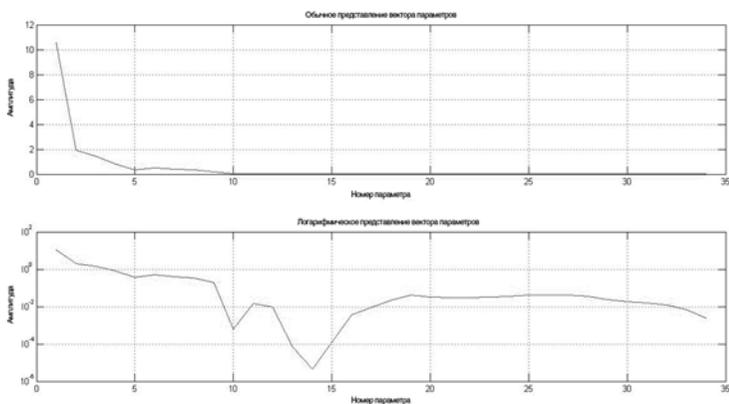


Рис.2. Обычный (сверху) и логарифмический (снизу) вид вектора параметров (по оси абсцисс – порядковый номер параметра, по оси ординат – амплитуда параметра)

Достигнутое быстродействие системы аутентификации позволяет проводить ее в режиме реального времени.

Особенности принятого авторами подхода к построению ИНС, на основе которых строится разработанная система аутентификации:

1. В качестве входных нейронов применяются информативные параметры речевых сигналов.
2. Результатом расчетов, проводимых с применением ИНС, является вектор размерности n , каждый (i -ый) элемент которого представляет собой вероятностное значение принадлежности речевого сигнала i -му диктору.
3. Обязательно должно проводиться обучение ИНС.
4. Изначально количеству эпох и скрытых нейронов присваиваются эмпирические значения, окончательные их значения определяются тестированием ИНС.

В процессе исследований к ИНС предъявлялись следующие требования: возможно меньшее количество применяемых информативных параметров, характеризующих речевые сигналы, при обусловленных значениях ошибок первого и второго рода (процента правильной аутентификации), возможно меньшее необходимое время обучения при обеспечении заданного качества проведения аутентификации и возможно более высокая скорость обработки речевых сигналов.

На основе результатов проведенных исследований авторы предлагают при аутентификации операторов в качестве ИНС применять много-

слойный перцептрон с одним скрытым слоем, а в качестве метода обучения - метод масштабированных сопряженных градиентов [3].

Для обоснованного выбора параметров системы было проведено ее тестирование. Критерием качества работы системы являлся процент правильно классифицированных сигналов (процент правильной аутентификации).

Применение разработанной системы параметров речевого сигнала для распознавания слов

Разработанная система параметров речевого сигнала была проверена на возможность её применения для распознавания речевых фрагментов. Целесообразность этого обусловлена тем, что, в соответствии с алгоритмом работы разработанной системы аутентификации, последняя проводится по командам, представляющим собой речевые фрагменты, выделенные из непрерывной речи операторов.

В результате проведения автоматической оптимизации системы аутентификации по наиболее значимым параметрам (длине фрейма, количеству коэффициентов линейного предсказания, применяемых при расчете кепстральных коэффициентов, количеству ККЛП, слоев, нейронов в каждом слое, эпох) был достигнут процент верно распознанных слов порядка 98% при суммарном времени обработки сигнала порядка 10 мс, что является приемлемым результатом для систем, работающих в режиме реального времени.

Данный результат подтверждает то, что представленный в статье принцип построения подсистемы параметризации позволяет строить подсистемы параметризации не только для решения задачи распознавания человека, являющейся частным случаем задачи распознавания образов, но и для решения более широкого класса задач – в частности, решения задачи распознавания речевых фрагментов (слов и словосочетаний).

Выводы

1. На основе проведенных исследований авторами в качестве параметров речевых сигналов системы голосовой аутентификации предлагается применять амплитуды спектральных составляющих, полученных в результате быстрого преобразования Фурье сигнала, составленного из пофреймово определенных кепстральных коэффициентов. Это позволяет уменьшить

размер векторов параметров более чем на порядок. Разработан алгоритм расчета предложенных параметров речевых сигналов.

2. В статье описана подсистема классификации для проведения голосовой аутентификации, построенная на основе искусственной нейронной сети.

3. Обоснованный выбор типа и значений параметров системы аутентификации позволил довести процент правильной аутентификации до 98%, а время параметризации – до 10^{-2} с.

Литература

1. Темников В.А., Темникова Е.Л. Параметризация автоматического контроля доступа операторов к ресурсам информационных систем по голосу // Вісник Східноукраїнського національного університету імені Володимира Даля. – 2010. – №9 (151). – Ч.1. – С.143-148
2. Темников В.А., Темникова Е.Л., Конфорович И.В. Разработка системы информативных параметров для аутентификации личности по голосу // XIII Международная научная конференция "Интеллектуальный анализ информации" (ИАИ-2013). Сборник трудов. - К.: Просвіта, 2013. – С.225-229
3. Moller M.F. A Scaled Conjugate Gradient. Algorithm for Fast Supervised Learning (preprint) / Denmark: Computer Science Department University of Aarhus. – 1990. – November 13. – 21 p.

УДК 510.2:004.8

Представлення знань за допомогою об'єктно-орієнтованих динамічних мереж

Терлецький Д.О., аспірант 3-го року навчання

*Київський національний університет імені Тараса Шевченка, м. Київ
dmytro.terletskyi@gmail.com*

В роботі пропонується нова модель представлення знань у вигляді об'єктно-орієнтованої динамічної мережі, яка дозволяє переставляти знання, які можуть бути змінені протягом певного часу, що дозволяє говорити про проектування нових типів інтелектуальних інформаційних систем, для вирішення різного роду задач в області штучного інтелекту.

Вступ

На сьогодні до області штучного інтелекту відносяться багато різних напрямків, зокрема експертні системи, представлення знань, автоматичне планування, машинне навчання, робототехніка, комп'ютерне бачення (виявлення), аналіз природньої мови, штучна творчість, когнітивне моделювання. Проте, одним із головних завдань всіх цих напрямків є проектування та розробка інтелектуальних інформаційних систем (ІС) для вирішення конкретних проблем. Важливою складовою більшості ІС є модель представлення знань, яка в певному сенсі визначає природу системи. На сьогодні існує багато різних моделей представлення знань (МПЗ), зокрема семантичні мережі, фрейми, скрипти та логічні моделі [1-4]. Кожна з цих моделей має певні особливості, які обумовлюють її практичне використання. Не дивлячись на це, усі ці моделі мають дещо спільне, зокрема вони описують зв'язки між концептами, об'єктами, класами, при цьому формально не визначаючи самі концепти. Більшість моделей відтворюють певну ієрархію між концептами, при організації якої виникають проблеми з винятками та множинним наслідуванням властивостей, що в свою чергу призводить до парадоксів у логічному виведенні, побудованому на основі зав'язків між концептами. Також більшість МПЗ не є динамічними, тобто вони не пристосовані до опису зміни концептів та зав'язків між ними. У зв'язку з цим, в рамках даної роботи пропонується деяка динамічна модель представлення знань, в основі якої лежать механізми мислення та сприйняття навколишнього світу людського інтелекту.

Об'єкти та класи

Як вже було зазначено вище, моделі представлення знань описують взаємозв'язки між концептами. В якості графічного представлення використовують графи та діаграми. У випадку семантичних мереж (СМ) використовуються графи, вершини яких відображають концепти, об'єкти або класи, а ребра – зв'язки між ними. Концепції об'єкту та класу об'єктів, є базовими для об'єктно-орієнтованого програмування (ООП), де об'єкти мають певну структуру, зокрема поля (властивості) [5,6]. Таке представлення знань використовується у фреймових моделях, де знання описуються через об'єкти та класи, використовуючи ідеї ООП. Розглянемо більш детально поняття об'єкту та класу об'єктів.

Об'єкти

В якості об'єктів можуть виступати будь-які предмети, явища нашої уяви або навколишнього світу. Розглянемо такий об'єкт, як натуральне число. Очевидно, що кожне натуральне число має бути цілим та додатнім, що є власне характеристичними властивостями натуральних чисел. Не важко переконатися, що 10 це дійсно натуральне число, а 4.36 або 5 – не є такими. Аналізуючи вище наведені факти, можна зробити висновок, що будь-який об'єкт має певні властивості, які є для нього характеристичними і визначають його як деяку сутність та дозволяють відрізнити його від інших об'єктів. В загальному, властивості об'єктів можна поділити на *кількісні* та *якісні*.

Кількісна властивість об'єкта A це пара $p(A) = (v(p(A)), u(p(A)))$, де $v(p(A))$ – це кількісне значення властивості $p(A)$, а $u(p(A))$ – одиниці її вимірювання.

Дві кількісні властивості $p_i(A)$ та $p_j(B)$, де $i = \overline{1, n}$, $j = \overline{1, m}$ є еквівалентними, тобто $Eq(p_i(A), p_j(B)) = 1$, тоді і тільки тоді коли $u(p_i(A)) = u(p_j(B))$.

Якісна властивість об'єкта A – це функція верифікації $vf(A)$, яка визначається наступним чином: $vf(A) : p(A) \rightarrow [0,1]$.

Дві якісні властивості $p_i(A)$ та $p_j(B)$, де $i = \overline{1, n}$, $j = \overline{1, m}$ є еквівалентними, тобто $Eq(vf_i(A), vf_j(B)) = 1$, тоді і тільки тоді коли $(vf_i(A) = vf_j(A)) \wedge (vf_i(B) = vf_j(B))$.

Специфікація об'єкта A – це вектор $P(A) = (p_1(A), \dots, p_n(A))$, де $p_i(A)$, $i = \overline{1, n}$ є кількісною або якісною властивістю об'єкта A .

Об'єкт A це пара $A/P(A)$, де A – це ідентифікатор об'єкта, а $P(A)$ – його специфікація. Тобто, об'єкт – це носій певної сукупності характеристичних властивостей, які визначають його як деяку сутність.

Окрім властивостей об'єктів, варто розглянути операції (методи), які можна застосовувати до об'єктів, враховуючи особливості їх специфікації.

Метод об'єкта або операція над об'єктом A – це функція $f(A)$, яку можна застосувати до об'єкту з врахуванням особливостей його специфікації.

Сигнатура об'єкта A – це вектор $F(A) = (f_1(A), \dots, f_m(A))$, де $f_i(A)$, $i = \overline{1, m}$ є методом об'єкта A .

Класи

В загальному об'єкти можна поділити на конкретні (реально матеріально існуючі) та абстрактні. Кожен конкретний об'єкт, незалежно від того, коли і як він був створений, є нічим іншим, як матеріальною реалізацією свого абстрактного образу – прототипу. Кожен прототип є власне абстрактною специфікацією для створення майбутніх реальних об'єктів. В рамках ООП, програмісти оперують властивостями об'єктів та їх методами без самих об'єктів, називаючи це *типом* або *класом* об'єктів [5,6]. Використавши цю ідею, сформулюємо формальне визначення класу об'єктів.

Клас об'єктів T – це пара $T = (P(T), F(T))$, де $P(T)$ – це специфікація класу T , а $F(T)$ – його сигнатура.

Аналізуючи концепцію об'єктно-орієнтованого класу, можна зробити висновок, що клас є прототипом для створення конкретних об'єктів і всі об'єкти певного класу матимуть специфікацію, аналогічну до специфікації їх класу і до них можна буде застосовувати лише ті методи, які передбачені в рамках їх класу. Тобто ми не можемо описати два принципово різних об'єкти, використовуючи для цього один об'єктно-орієнтований клас. Це пов'язано з тим, що концепція об'єктно-орієнтованого класу є прикладом однорідного класу.

Однорідний клас об'єктів – це клас, що містить лише однотипні об'єкти.

Проте існують і неоднорідні класи. Розглянемо такі об'єкти як натуральні числа n_1, \dots, n_m . Вони належать одночасно до класів натуральних,

цілих, раціональних та дійсних чисел, тобто $n_1, \dots, n_m \in N \in Z \in Q \in R$. При цьому, якщо розглянути раціональні числа q_1, \dots, q_m , то їх специфікація буде відмінною від специфікації n_1, \dots, n_m , але всі вони належать до класу дійсних чисел R .

Неоднорідний клас об'єктів T – це кортеж виду

$$T = (\text{Core}(T), pr_1(A_1), \dots, pr_n(A_n)),$$

де $\text{Core}(T) = (P(T), F(T))$ – це ядро класу T , що містить тільки ті властивості і методи, що є спільними для специфікацій $P(A_1), \dots, P(A_n)$ та сигнатур $F(A_1), \dots, F(A_m)$ відповідно. $pr_i(A_i) = (P(A_i), F(A_i))$, $i = \overline{1, n}$ є проєкціями об'єктів A_1, \dots, A_n , які містять властивості та методи, які є характерними лише для цих об'єктів.

Операції над об'єктами

Як було зазначено вище, в рамках ООП об'єкти певного класу мають не лише властивості, а й методи, які можна до них застосовувати, враховуючи особливості їхньої специфікації. В залежності від характеру дії, методи, які застосовуються до об'єктів, можна поділити на два типи – *модифікатори* та *експлуататори*.

Модифікатори – це функції, які можуть змінювати об'єкти, наприклад, деякі їх поля (властивості).

Експлуататори – це функції, що використовують об'єкти в якості параметрів (аргументів) і не можуть змінювати їх.

Проте, більшість методів є локальними щодо об'єктів і не можуть бути застосовані до об'єктів різних типів. Підтвердженням цього є те, що алгоритм множення дробових чисел відрізняється від алгоритму множення цілих чисел. В рамках ООП ця проблема вирішується за допомогою перевантаження операторів, тобто реалізується декілька варіацій одного і того ж методу в залежності від типів об'єктів. Проте, використовуючи перевантаження операторів, варто пам'ятати, що для кожного окремого типу об'єкта та чи інша операція матиме свою семантику, яка відрізнятиметься від інших її реалізацій.

В роботі [7] були запропоновані операції об'єднання, перетину, різниці, симетричної різниці та клонування, які є універсальними і можуть бути застосованими до будь-яких об'єктів.

Об'єктно-орієнтовані динамічні мережі

Як вже було сказано раніше, на сьогодні існує багато різних моделей представлення знань. Однією з найпоширеніших моделей є семантичні мережі. Існує декілька їх типів, зокрема семантичні дерева, дерева пошуку, цільові дерева [1-3]. Як і кожна з моделей, семантичні мережі мають свої переваги і недоліки. З одного боку СМ є дуже простою та інтуїтивною моделлю представлення знань, але з іншого боку вони мають ряд недоліків, зокрема проблеми з організацією ієрархії об'єктів та класів, винятками та множинним наслідуванням [2]. Також вони дозволяють описувати лише статичні знання, тобто знання, які є незмінними в часі, при цьому основний акцент робиться на зв'язках між об'єктами та класами, в той час як визначенню самих об'єктів та класів приділяється дуже мало уваги. Зрозуміло, що з самого початку існування СМ, вони були запропоновані як модель представлення живої мови, динаміка зміни якої не є досить високою. Проте, якщо розглядати СМ як модель представлення знань в рамках деякої ІС, то виникає багато питань щодо її ефективності для розв'язку тих чи інших проблем в області ШІ, які не зовсім або взагалі не пов'язані з обробкою природньої мови. Дане питання постає досить гостро, коли мова йде про динамічні знання, тобто ті, які змінюються у часі. У зв'язку з цим, виникає необхідність в розробці більш динамічних моделей представлення знань.

Об'єктно-орієнтовану динамічну мережу (ООДМ) можна представити у вигляді зв'язного орієнтованого графа $OODN = (Nod, Edg)$, де Nod – це множина вершин, а Edg – множина ребер. Вершинам графа $OODN$ можуть відповідати об'єкти або класи об'єктів, що дозволяє виділити три типи таких мереж – об'єктні, класові та змішані. Кожен об'єкт $A_i \in Nod$, який є вершиною графа $OODN$ має свою специфікацію $P(A_i)$ та належить до певного класу $T(A_i)$, який визначає для нього відповідну сигнатуру $F(A_i)$. Ребра графа $OODN$ можуть відображати відношення між об'єктами і класами об'єктів. Також ребра можуть відображати модифікації об'єктів їх класів та відношень між ними, тобто методи з їхніх сигнатур.

Для прикладу побудуємо ООДМ, використовуючи такі об'єкти, як квадрат $A/(p_1(A), \dots, p_7(A))$ і ромб $B/(p_1(B), \dots, p_6(B))$ (Рис. 1.). Кожен з цих об'єктів належить до відповідного класу опуклих чотирикутників, тобто квадрат A належить до класу квадратів $T(A)$, а ромб B належить до класу ромбів $T(B)$. Властивості $p_1(A)$, $p_1(B)$ – це кіль-

кість сторін фігури, яка рівна 4, властивості $p_2(A)$, $p_2(B)$ – це розміри сторін фігур, властивості $p_3(A)$, $p_3(B)$ – це кількість кутів у фігурі, яка рівна 4, властивості $p_4(A)$, $p_4(B)$ – це градусна міра кутів фігури, властивості $p_5(A)$, $p_5(B)$ – це паралельність протилежних сторін фігури, властивості $p_6(A)$, $p_6(B)$ – це рівність усіх сторін фігури і властивість $p_7(A)$ – це рівність усіх кутів фігури (градусна міра кожного 90°).

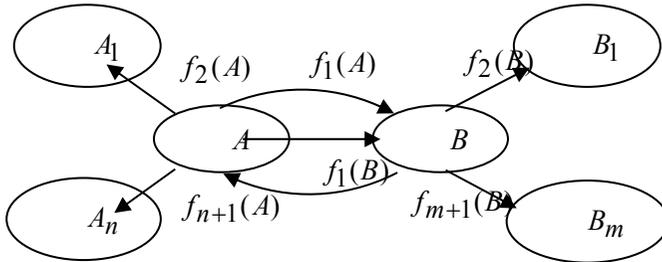


Рис. 1. Частина ООДМ для квадрата і ромба.

В результаті ми отримали частинку ООДМ для квадрата A і ромба B , яка відображає, що квадрат є ромбом і що, якщо застосувати до квадрата операцію $f_1(A)$, яка змінює градусну міру кутів квадрату таким чином, що жоден з кутів не дорівнює 90° , але сума градусних мір всіх кутів при цьому залишається 360° , то він стане ромбом. Операція $f_1(B)$ змінює градусну міру кутів ромба таким чином, що всі кути стають рівними 90° . Операції $f_2(A), \dots, f_{n+1}(A)$ та $f_2(B), \dots, f_{m+1}(B)$ пропорційно змінюють розміри сторін квадрата A і ромба B , таким чином, що ми отримуємо нові квадрати A_1, \dots, A_n та ромби B_1, \dots, B_m , які є меншими або більшими за початкові.

Аналізуючи граф ООДМ, що зображений на Рис. 1, можна переконалися, що такий підхід до представлення знань є більш динамічним, ніж звичайні СМ чи інші моделі. Він дозволяє в певному сенсі моделювати систему знань людини, яку вона модифікує, коли отримує новий досвід і нові знання. Модифікацію знань і переміщення по графу ООДМ можна розглядати як моделі препозиційного міркування, логічного висновку, пошуку потрібного рішення, творчої фантазії, тощо. Застосування операцій над об'єктами може змінювати об'єкти та їх класи, а також породжувати нові об'єкти та класи об'єктів з уже існуючих в мережі. В концепції

ООДМ передбачені можливості конструктивного розширення мережі, тобто додавання нових об'єктів та класів об'єктів, шляхом використання операції об'єднання об'єктів.

Висновки

В даній роботі пропонується об'єктно-орієнтована динамічна мережа як модель представлення знань в ІС та описується її концепція та основні можливості. Ця модель дозволяє представляти знання, які можуть бути змінені протягом певного часу. Вона також дозволяє будувати нові типи зв'язків (відношень) між об'єктами та класами об'єктів, а саме відображати результати змін (модифікації) певних об'єктів або їх класів, таким чином відображаючи можливі результати операцій, які можна до них застосовувати. Такий підхід дає підстави говорити про створення на його основі динамічних ІС, проте, незважаючи на це, він потребує більш глибокого та детального дослідження.

Література

1. *B. Coppin*. Artificial intelligence illuminated. – Jones and Bartlett Publishers, Inc. 2004.
2. *E.C. Way*. Knowledge Representation and Metaphor. – Springer-Science + Business Media, B.V., 1991.
3. *G.F Luger*. Artificial Intelligence. Structures and Strategies for Complex Problem Solving: 6-th edition. – Addison-Wesley, 2008.
4. *M. Ginsberg*. Essentials of artificial intelligence. – Morgan Kaufmann Publishers, Inc., 1993.
5. *R. Pecinovsky*. OOP – Learn Object Oriented Thinking and Programming. – Tomáš Bruckner, Řepín-Živonín, 2013.
6. *M. Weisfeld*. The Object-Oriented Thought Process. Third Edition. – Addison-Wesley Professional, 2008.
7. *Д.О. Терлецький*. Конструктори множин та мультимножин об'єктів // Проблеми програмування. – 2014. – № 1. – С. 18–30.

УДК 004.896

Інформаційно-логічні та архітектурні засади універсальних систем керування web-контентом

Титенко С.В., к.т.н.

*Національний технічний університет України «КПІ», м. Київ
lab@setlab.net*

Робота присвячена проблемам моделювання та розробки універсальних систем керування контентом в веб-середовищі. Подаються ключові концептуальні та архітектурні передумови побудови таких систем.

Вступ

Керування контентом – галузь, з якою користувачі всесвітньої мережі WWW зустрічаються щодня. Системи керування контентом (*англ.* content management systems, CMS) стали базовим і невід’ємним програмним забезпеченням інформаційних веб-ресурсів. Проблема інтелектуалізації та автоматизації процесів керування веб-контентом не втрачає своєї актуальності, незважаючи на безліч промислових CMS-систем.

Універсальні CMS-системи та керування знаннями в Web-середовищі

Детальний огляд та класифікація CMS-систем здійснені в роботі українських авторів [1]. Тут широко висвітлено сукупність технічних аспектів з акцентом на функціональні можливості систем керування контентом. Спроба узагальнити структуру CMS-системи подані в роботі [2]. Дана структура покриває сукупність низькорівневих технічних задач CMS. Натомість керування веб-контентом в широкому сенсі фактично передбачає керування знаннями в веб-середовищі, що вимагає застосування методів інформаційного аналізу, моделювання формалізації знань та інформації [3-7].

Універсальні CMS-системи – це такі системи керування контентом, які дозволяють подавати в веб-середовищі довільні інформаційні структури предметних областей, надаючи зручні механізми конструювання, пуб-

лікації та навігації по інформаційним сутностям предметної області веб-ресурса. Такі задачі зумовлюють потребу в особливій увазі до проблем побудови інформаційних моделей контенту CMS.

Питанням інформаційної моделі CMS та перспективам семантичного моделювання контенту присвячена роботи Олецького О.В., зокрема [8]. В роботі наголошується, що CMS-системи є документо-орієнтованими, пропонується формалізувати відношення документів системи та понять предметної області, а також ввести структуру навігаційних посилань у вигляді графа. У даному графі окрім динамічних відношень, пов'язаних з моделлю предметної області, пропонується використовувати ієрархічні відношення між елементами контенту. Натомість, у роботі не подано конкретних шляхів реалізації програмної системи та прикладів дослідної апробації запропонованих ідей. У подальших роботах розвинуто формальний теоретико-множинний апарат для запропонованого підходу. З іншого боку значна кількість робіт з галузі побудови навчальних веб-систем також детально розглядають питання формалізації контенту у поєднанні з моделюванням предметної області [9,10]. Узагальнюючи, можна підсумувати, що склався підхід до моделювання контенту інтелектуальних веб-систем, що передбачає семантичну структуру концептів предметної області, структуру елементів контенту та зв'язки між концептами та контентом. Актуальними є роботи, що разом з концептуальними пропозиціями акцентують увагу на працюючих проектних рішеннях та практичних апробаціях іноваційних методів керування контентом в універсальних CMS.

В попередніх роботах було подано модель контенту та перспективи керування знаннями в межах керування контентом в веб-середовищі [3-7]. Завданням даної роботи є узагальнення інформаційно-логічних та архітектурних вимог до універсальних CMS-систем та їх програмна реалізація.

Інформаційно-логічні та архітектурні засади CMS

Результатом досліджень, проектування та програмної реалізації CMS став перелік інженерно-технічних та модельних рішень, що повинні бути покладені в основу системи керування контентом універсального типу (рис.1).

Об'єктна модель сторінки.

Ключовою інформаційною сутністю систем керування контентом є сторінка або елемент контенту. Об'єкт сторінки інкапсулює в собі усю логіку роботи із сторінкою, надаючи інтерфейс іншим програмним компонентам CMS для отримання доступу до інформації даного елемента конте-

нту. На рівні БД CMS-системи існує спеціальна таблиця, що представляє дану сутність, характеризуючи її певним набором полів. Основними полями об'єкта сторінки є заголовок, контент, анотація та ін. Так в розповсюджених CMS сторінка також моделюється і подається певним чином: в Drupal загальний елемент контенту позначається терміном Node (вузол), у Wordpress – Post (допис), в Joomla – матеріал.

Спираючись на об'єкт сторінки, відбувається процес публікації (рис.2).



Рис.1. Архітектурні та модельні компоненти CMS універсального типу

Ієрархія контенту

Ієрархія – це розташування частин або елементів цілого в певному порядку від вищого до нижчого. Ієрархія – найзручніший і природно зрозумілий спосіб структурування речей, це ж справедливо і для елементів контенту. Незважаючи на те, що ієрархічне структурування в Web викликає труднощі у випадках великої кількості інформації, тим не менше ієрархія, як базова структура, для організації контенту є зручним і практично доцільним способом структурування інформації веб-ресурсу.

Сторінки-контейнери

Сторінки-контейнери – сторінки, головним призначенням яких є подання списку анотованих посилань на дочірні сторінки. Власний контент таких сторінок, як правило, не має великого об'єму і є короткою анотацією до розділу або набору статей. Сторінки-контейнери займають друге місце після простих інформаційних сторінок по частоті застосування в веб-ресурсах. Наявність сторінок-контейнерів як стандартної функціональності системи є необхідною передумовою для універсальності CMS-системи. За допомогою сторінок-контейнерів моделюються найрізноманітніші розділи веб-ресурсу – добірки статей, розділи новин, категорії

товарів та будь-які інші набори інформаційних об'єктів предметної області веб-ресурсу.

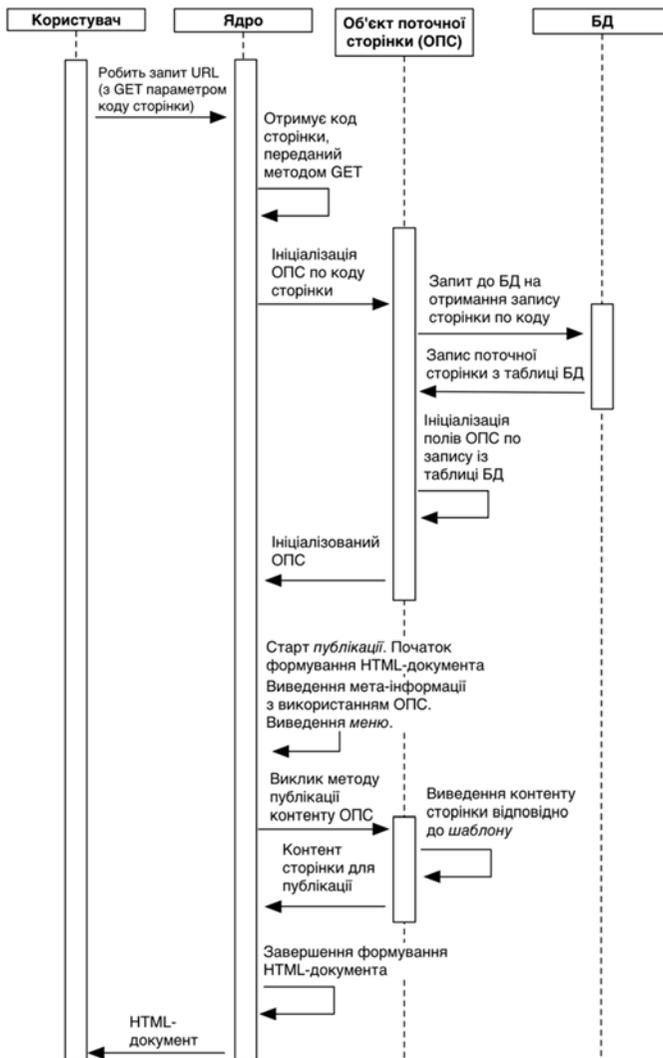


Рис.2. Діаграма послідовності CMS

Ациклічний орграф контенту

Тенденція примноження інформації ускладнює структурування інформаційних елементів у строгій деревоподібній формі. Крім цього строга ієрархія часто не задовольняє навігаційним вимогам веб-ресурсу. Потреба одночасної присутності одного і того ж елемента контенту в різних розділах сайту – досить розповсюджене і актуальне завдання. Прикладом цього може служити новина, яка по своєму змісту одночасно задовольняє різним розділам новин, товар, що підходить до різних категорій, деяка службова інформаційна сторінка сайту, яку доцільно розмістити в різних частинах сайту.

Подібні об'єктивні неоднозначності вимагають від системи функціональної можливості розташування сторінки сайту в різних місцях ієрархії. Доцільною структурою для такого завдання є ациклічний орієнтований граф. Ациклічний орграф є узагальненням дерева. Для оптимізації інформаційно-логічної структури CMS-системи ациклічний орграф доцільно реалізовувати на базі відношень псевдонімів [4] (рис. 3).

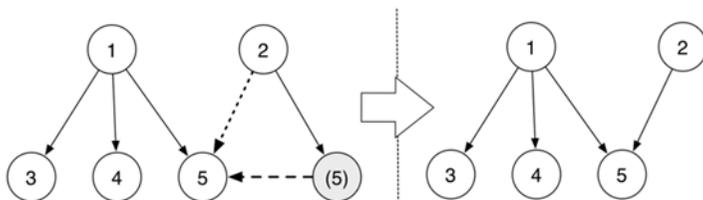


Рис. 3. Принцип роботи відношення псевдонімів

Спеціалізовані типи контенту

Актуальним завданням розвинених CMS-систем залишається дослідження, розробка та вдосконалення засобів універсального керування інформаційними об'єктами веб-ресурсів, що дозволить спростити створення нових ресурсів різного призначення, а також забезпечить ефективні механізми їх супроводження та налаштування. В межах даної задачі потрібно вирішувати наступні завдання: створення спеціалізованих інформаційних об'єктів із користувацьким набором полів, керування відображенням таких об'єктів, сортування, фільтрація та навігація по сховищу створених об'єктів, інтеграція сховища інформаційних об'єктів із загальним сховищем контенту, керування ієрархією об'єктів, класифікація та групування об'єктів тощо. Огляд інформаційно-логічних методів моделювання спеціалізованих типів контенту подано в попередній роботі [5]

Макромова керування контентом.

Макромова керування контентом в CMS – це спеціальна мова логічного керування контентом, оператори якої вбудовуються безпосередньо в інформаційний html-текст веб-сторінки. Такі команди можуть використовуватись на рівні контент-менеджера або архітектора веб-сайту.

Робота з контентом часто передбачає багаторазове виконання однотипних задач оформлення інформації. Виконання подібних задач вручну призводить до необхідності повторного формування html-розмітки, додаткового програмування та нагромадження в контенті сайту технічно-залежного коду. Ефективним розв'язком проблеми є створення системи *макрокоманд контенту*, що дозволять зосередити логіку оформлення повторюваних фрагментів контенту в одному місці [6].

Тематичні таксономії предметних областей контенту

Тематичні групи служать для організації різноманітних міжпредметних і внутрішньопредметних зв'язків між елементами контенту. Тематичні групи використовуються для моделювання предметних областей, каталогізації, групування і вибірки асоціативного контенту. Організація тематичних груп відбувається у ієрархічній структурі. Це дозволяє вибудовувати таксономію предметних областей. Кожен елемент контенту може брати участь у довільній кількості тематичних груп [4].

Семантична база понять контенту

Для реалізації керування контентом на рівні семантики предметної області необхідно створювати сховище понять предметної області контенту та формалізувати відношення між ділянками контенту і поняттями предметної області. Така задача вирішується на основі онтологій та семантичних моделей предметних областей [7,8,10].

Висновок

У роботі представлено сукупність архітектурних, інженерно-технічних та інформаційно-логічних рішень для побудови на їх ґрунті універсальної системи керування контентом. Запропонований комплекс дозволяє на програмному рівні реалізувати ключові завдання універсальних CMS щодо конструювання, публікації та навігації по інформаційним сутностям предметної області веб-ресурсу. Прототип системи апробований на порталі znannya.org. Публікації та лекційний відео-матеріал щодо теми дослідження розміщено на сайті setlab.net.

Література

1. *Курилов М.А.* Классификация систем управления содержимым web-ресурсов и их использование для разработки сайта дистанционного обучения / М.А. Курилов, С.В. Терещенко // Штучний інтелект. — 2010. — № 3. — С. 648-654.
2. *Любченко С.С.* Формирование обобщенной структуры системы управления содержимым web-сайта / С. С. Любченко // Искусств. интеллект. - 2005. - № 2. - С. 109-115.
3. *Титенко С.В.* Моделювання області знань в системі безперервного навчання на основі інтеграції моделі контенту Tree-Net і понятійно-тезисної моделі / С. В. Титенко, О. О. Гагарін// VIII міжнародна конференція «Інтелектуальний аналіз інформації ІАІ-2008», Київ, 14-17 мая 2008г. : Сб. тр./ Ред. кол. : С.В. Сирота (гл.ред.) и др. – К.: Просвіта, 2008. – С. 475-484.
4. *Титенко С. В.* Модель навчального Web-контенту Tree-Net як основа для інтеграції керування знаннями і безперервним навчанням / С. В. Титенко, О. О. Гагарін // Системні дослідження та інформаційні технології. – 2009. – № 1. – С. 74–86.
5. *Титенко С. В.* Моделювання спеціалізованих інформаційних об'єктів в універсальних системах керування Web-контентом / С. В. Титенко // Вісник Східноукраїнського національного університету імені Володимира Даля – 2012. – №8 (179), Ч.2. — С. 235-239.
6. *Титенко С. В.* FreshKnowledge – система управління навчальним Веб-контентом на семантичному рівні / С. В. Титенко, О.О.Гагарін//VII міжнародна конференція «Інтелектуальний аналіз інформації ІАІ-2007», Київ, 15-18 мая 2007г. : Сб. тр./ Ред. кол. : С.В. Сирота (гл.ред.) и др. – К.: Просвіта, 2007. – С. 342-352
7. *Титенко С. В.* Структурные основы онтологически-ориентированной системы управления информационно-учебным Web-контентом / С. В. Титенко // Управляющие системы и машины: информационные технологии: междунар. науч. журн. - 2012. - № 2. - С. 35-42. ISSN 0130-5395
8. *Олецький О.В.* Застосування формальних моделей онтологій для формалізації інформаційних потоків у системах управління контентом / О. В. Олецький // Теоретичні та прикладні аспекти побудови програмних систем. Матеріали міжнародної конференції ТАAPSD'2005, Київ, 7-9 грудня 2005 р. — С. 26-29.
9. *Семикин, В. А.* Семантическая модель контента образовательных электронных зданий: Автореф. дис. ... канд. тех. наук: 05.13.18 / Семикин Виктор Алексеевич ; Тюменск. гос. ун-т. – Тюмень, 2004. – 21 с.
10. *Brusilovsky, P.* Adaptive hypermedia, an attempt to analyze and generalize / P. Brusilovsky // Multimedia, Hypermedia, and Virtual Reality (Lecture Notes in Computer Science). [P. Brusilovsky, P. Kommers, N. Streitz (Eds.)]. – Berlin: Springer-Verlag, 1996. – Vol. 1077. – P. 288-304.

УДК 681.3: 519.68

Сжатие данных на основе бинарного дерева

Трифонов Е.О. Белорусский Государственный Университет Информатики и Радиоэлектроники «БГУИР», г. Минск, naffer@iut.by

В данной работе описан новый метод и принципы сжатия информации на основе бинарного дерева, которые могут быть использованы как самостоятельно, так и вместе с существующими методами сжатия информации. Предложено полное теоретическое обоснование работоспособности нового метода сжатия.

Вступление

Сжатие информации всегда являлось актуальной задачей в сфере информационных технологий. За последние несколько лет выросли объемы жестких дисков компьютеров, но тем не менее передача данных по сети до сих пор использует сжатие данных.

В данной работе получено математическое обоснование работоспособности метода сжатия данных на основе бинарного дерева, в дальнейшем результаты работы необходимо проверить на реальных данных и собрать статистическую информацию о результатах применения нового метода сжатия информации. Базируясь на результатах, в ближайшем будущем будет проведено исследование сжатия данных с разбиением на части по 4 бита каждая.

Обоснование сжатия данных

Базовая структура сжатия данных

Все данные в компьютере представляют собой определенную последовательность бит. Каждый бит такой последовательности может принимать одно из двух значений: единица или ноль. Благодаря этому такую последовательность можно представить веткой бинарного дерева, в корне дерева будет располагаться первый бит последовательности, а листом будет последний бит. И не будет иметь значения, ноль или единица поставлены в корень, так как всегда вправо будут откладываться единицы, а влево – нули.

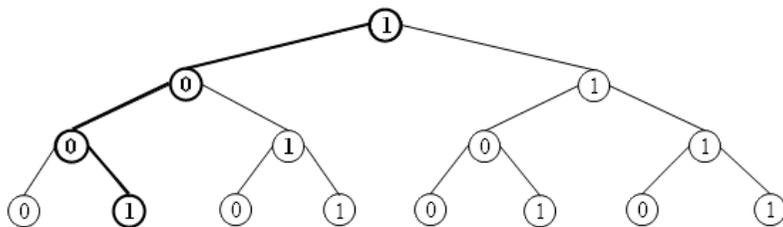


Рис. 1. Пример построения дерева для 4 бит данных, начинающихся с единицы

На рисунке 1 выделенная ветка дерева соответствует последовательности бит **1011**, в данном случае последний бит последовательности является вторым по счету на третьем уровне бинарного дерева – уровни дерева нумеруются с нуля. На любом уровне дерева присутствуют позиции соответствующие числам, которые можно получить при возведении двойки в определенную степень.

Из таблицы 1 очевидно, что количество позиций m зависит от уровня дерева n как: $m = n + 1$

Выделенная строка в таблице соответствует примеру, показанному на рисунке 1, и, как можно заметить, позиция последнего бита последовательности **1011** является степенью двойки, а именно 1.

Таблица 1. Количество и номера позиций степеней двойки на уровнях от 0 до 5 бинарного дерева

Уровень дерева (степень двойки), n	Количество позиций соответствующих степени двойки, m	Номера позиций
0	1	1
1	2	1, 2
2	3	1, 2, 4
3	4	1, 2, 4, 8
4	5	1, 2, 4, 8, 16
5	6	1, 2, 4, 8, 16, 32

Благодаря тому, что дерево бинарное, а значит ориентированное, достаточно знать значение первого бита, число уровней в дереве и позицию последнего бита, чтобы проложить «путь» от первого до последнего бита, такой же, как показан на рисунке 1. Пройдя через такой «путь» можно восстановить последовательность данных без потерь. Таким образом, последовательность бит можно сохранить в следующей структуре.

Таблица 2. Структура хранения последовательности данных

Значение первого бита	Длина последовательности, n	Позиция последнего бита, pos
-----------------------	-------------------------------	--------------------------------

В структуре, описанной таблицей 2, размер значения первого бита постоянный и составляет 1 бит, размер длины последовательности динамический, а позиция последнего бита зависит от длины последовательности следующим образом: $size(pos) = 2^{n-1}$.

Размер исходной последовательности данных соответствует сумме размеров значения **первого бита** и **позиции последнего бита**. Таким образом, **длина последовательности** является избыточной информацией, и размер структуры превышает размер исходных данных. Но если учесть тот момент, что **позиция последнего бита** может оказаться степенью двойки, тогда в **позиции последнего бита** достаточно сохранить лишь степень, в которую необходимо возвести двойку, а это значит, что размер этого поля будет зависеть от количества позиций соответствующих степени двойки в на определенном уровне бинарного дерева, и его можно рассчитать следующей зависимостью от **длины последовательности n**:

$$size(pos) = \begin{cases} \sqrt{n+1} - \lfloor \sqrt{n+1} \rfloor = 0, \sqrt{n+1} \\ \sqrt{n+1} - \lfloor \sqrt{n+1} \rfloor > 0, \sqrt{n+1} + 1 \end{cases}, \quad (1)$$

Таким образом, при соответствии **позиции последнего бита** степени двойки, структура может сжимать информацию, так как для хранения такой структуры может понадобиться меньше бит, чем исходной последовательности данных.

Применение особенностей бинарного дерева

Для получения последовательностей, которые бы заканчивались в позиции степени двойки, исходная последовательность может быть разделенная на части, и эти части могут быть сжаты предложенной структурой по отдельности.

Таблица 3. Процентное соотношение числа позиций соответствующих значениям степеней двойки к общему числу позиций на заданном уровне бинарного дерева

Уровень дерева (степень двойки), n	Количество позиций соответствующих степени двойки, m	% покрытия позиций, c
0	1	100,00
1	2	100,00
2	3	75,00
3	4	50,00
4	5	31,25
5	6	18,75

Очевидно, что процент покрытия может быть рассчитан по следующей формуле: $c = \frac{m}{2^n} \times 100\% = \frac{n+1}{2^n} \times 100\%$. А при $n \rightarrow \infty$, процент покрытия $c \rightarrow 0$.

В таблице 4 не представлены данные для уровней ниже 6 по той причине, что результаты сжатия были получены отрицательные, и не представляют интереса. Колонка, показывающая необходимое количество бит

для хранения структуры, в таблице 4 в скобках имеет расписанное число бит для каждой части структуры: **значение первого бита, длина последовательности, позиция последнего бита** – соответственно.

Таблица 4. Процентное соотношение сжатия от уровня бинарного дерева при использовании предложенной структуры

Уровень дерева (степень двойки), n	Число бит исходных данных	% покрытия позиций, с	Число бит необходимое для хранения структуры	% сжатия
6	7	10,94	7 (1 + 3 + 3)	0,00
7	8	7,03	7 (1 + 3 + 3)	12,50
8	9	3,91	9 (1 + 4 + 4)	0,00
9	10	2,15	9 (1 + 4 + 4)	10,0
15	16	0,05	9 (1 + 4 + 4)	43,75
31	32	1,48e-8	11 (1 + 5 + 5)	65,63

Структура сжатия данных с константной величиной длины последовательности данных

Если предположить, что сжатие данных будет происходить по фиксированной длине последовательности или фиксированной длине частей исходных данных, тогда эту величину будет правильным вынести из структуры, и таким образом, структура станет выглядеть следующим образом.

Таблица 5. Структура хранения данных с константной величиной длины последовательности данных

Значение первого бита	Позиция последнего бита
-----------------------	-------------------------

При такой структуре, результаты заметно улучшаются. В таблице 6 видно, что диапазон сжатия уровней бинарного дерева расширился, а процент сжатия данных увеличился при использовании структуры с константной величиной длины последовательности данных. Это значит, что структура с константной величиной длины последовательности имеет право на существование. Можно предположить разумным изучение исходной последовательности бит для вычисления наиболее подходящего константного значения **длины последовательности**, а затем сохранения этого константного значения в заголовке файла или последовательности бит, где будет храниться сжатая информация.

Таблица 6. Процентное соотношение сжатия от уровня бинарного дерева при использовании структуры с константной величиной длины последовательности данных

Уровень дерева (степень двойки), n	Число бит исходных данных	% покрытия позиций, с	Число бит необходимое для хранения структуры	% сжатия
1	2	100,00	2 (1 + 1)	0,00
2	3	75,00	3 (1 + 2)	0,00
3	4	50,00	3 (1 + 2)	25,00
4	5	31,25	4 (1 + 3)	20,00
5	6	18,75	4 (1 + 3)	33,(33)
6	7	10,94	4 (1 + 3)	42,85
7	8	7,03	4 (1 + 3)	50,00
8	9	3,91	5 (1 + 4)	44,(44)
9	10	2,15	5 (1 + 4)	50,00
15	16	0,05	5 (1 + 4)	68,75
31	32	1,48e-8	6 (1 + 5)	84,38

Классификация

Данный алгоритм основан на избыточности информации при хранении последовательности бит, которую возможно вычислить базируясь на построении бинарного дерева и позиций бит в нем. Эта модель избыточности является главным отличием данного алгоритма от существующих.

Кодирование информации предполагается кодами фиксированной длины, при условии блочной обработки информации. Блоки переменной длины замедлят скорость алгоритма, а так же потребуют дополнительной информации при сжатии.

Алгоритм сможет быть повторно применен к уже сжатой информации, это следует из того, алгоритм сжимает информацию выборочно, в виду чего и есть основания предполагать, что он не будет однократным, а значит наврядли сможет применяться в потоковой обработке или передаче данных с хорошими показателями сжатия.

При теоретическом обосновании сложно указать какой тип информации будет сжиматься лучше, поэтому предварительно алгоритм не является специализированным и может применяться для любого типа информации.

Заключение

Как и всем алгоритмам сжатия информации, базируемым на бинарных деревьях, данному алгоритму будет присуща высокая скорость обработки данных, а конкретно, поиск последовательностей для сжатия и построение исходной последовательности из сжатой.

Алгоритм предполагает использование блоков статической длины, что также содействует увеличению производительности. К тому же, такой алгоритм будет способен работать в несольких потоках одновременно.

Хотелось бы отметить, что разбиение исходной последовательности на части даже по 4 бита могут быть сжаты с вероятностью 50 процентов, что является интересной темой для изучения.

Таким образом, можно заключить, что алгоритм способен сжимать данные, а степень сжатия зависит как от предварительного изучения исходных данных, так и от реализации самого алгоритма.

Литература

1. *Балашов К.Ю.* Сжатие информации: анализ методов и подходов. – Минск: Институт технической кибернетики НАН Беларуси. – 2000. – 42 с.
2. *Blleloch E.* Introduction to Data Compression: Carnegie Mellon University. – 2001. – P. 54.

УДК 340.11: 004.94

Функциональность и методы прикладной правовой онтологии

*Хала К. А., аспирантка, Международный научно-учебный центр информационных технологий и систем НАН и МОН Украины, г. Киев
cecerongreat@ukr.net*

Эта статья представляет некоторые решения, на основе контента шаблонов проектирования онтологий, которые предназначены, чтобы сделать жизнь дизайнеров правовых онтологий легче. Обзор типовых задач и услуг для представления юридических знаний, вводится понятие шаблона проектирования онтологий, и некоторые выдержки из базовой правовой онтологии (БПО) и включая связанные с ней модели.

Введение

Онтологии могут быть разработаны с помощью различных методологий [1], охватывая нисходящий сбор информации экспертных знаний от людей, восходящее приобретение знаний из документов и среднее применение контента шаблонов (специализированного для независимых от области онтологий, выявляемых нисходящим способом, или обучаясь через шаблоны, найденные в документах экспертов), которые можно назвать шаблонами проектирования онтологий (КШПО, концептуальные шаблоны проектирования онтологий) [1].

Правовая область является очень сложной по сравнению с другими, потому что она включает знания о физических и социальных мирах, а также типовые правовые знания, которые фактически создают новый слой по социальному миру [1,2].

Из-за автономии (с одной стороны) и зависимости (с другой стороны) правового знания и о физических, и о социальных знаниях, правовые задачи рассуждения развивались специфическим образом, который включают, например, структуру нормы основанную на КШПО как требование \rightarrow следствие (если фактические знания P , то правовые знания Q), обязательство \rightarrow право (если у A есть обязательства по отношению к B , то B имеет право к A), норма \leftrightarrow случай (если ситуация выполняет условия для нарушения нормы, это становится правовым случаем (судебным делом), Сценарий Преступления (преступление совершено преступником и привлекает внимание властей, которые преследуют преступный процесс) и др. КШПО который принят правовыми экспертами может быть формализован, специализируя или составляя другие существующие шаблоны для социального мира.

Прикладная правовая онтология: функциональность и методы

Разработчик онтологии имеет дело с проектированием, управлением и эксплуатацией онтологий в пределах информационных систем. Обычно онтологии смешиваются с другими компонентами, чтобы построить семантически-явные приложения; например, когда используются совместно с:

- доказательством теорем, тогда проверка согласованности может быть выполнена, чтобы логически утвердить ряд допущений, закодированных в онтологии

- категоризацией и классификаторами экземпляра логического языка известной и управляемой сложности, автоматические выводы могут быть получены из таксономического рассуждения, а также для классификации случаев и фактов [2]

- вычислительными словарями, с NLP инструментами и алгоритмы машинного обучения, правовые онтологии могут повысить извлечение информации из слабоструктурированных и неструктурированных данных, добавляя новое измерение в управление знаниями и открытия в области права [2,3]

- алгоритмами планирования, онтологии могут помочь автоматизировать или обсуждение условий, или выполнение, например для соглашений, положений, услуг и т.д. [1,3]

- случаем основанным на рассуждениях, онтологии могут формализовать абстракции случая в более общих рамках, или могут классифицировать случаи согласно предварительно разработанным описаниям [4]

- системами основанными на правилах, где факты могут выводиться, например, для причинной оценки ответственности, проверки соответствия, обнаружения конфликта и в целом для состава факта [3,4].

Концептуальные шаблоны проектирования онтологии

Семантически-явные приложения в правовой области дают нам концептуальный анализ и проблемы интеграции, которые требуют соответственно разработанных правовых формальных онтологий. Часть задач проектирования можно упростить путем создания или извлечения «концептуальных шаблонов проектирования онтологий» (КШПО) для прикладной области [5]. Интуитивно понятная характеристика КШПО обеспечивается:

- КШПО является шаблоном для представления, и, возможно, решения проблем моделирования. Например, норма ↔ случай КШПО облегчает моделирование правовых норм и случаев (а также их компонентов и зави-

симостей) в логических языках, которые требуют ограничительной материализации. Например, в OWL, взаимоотношения с арностью=2 не допускаются, поэтому OWL моделирование требует материализации этих отношений. Словарь для материализации был разработана в Описаниях и Ситуациях онтологии (рис. 4), которая специализируется на шаблоне Норма \leftrightarrow Случай.

- КШПО "извлекает" связный фрагмент справочной онтологии, которая составляет его «фон». Например, взаимосвязь двух отношений и трех классов ($A(x) \exists B(y) \exists C(Z) \exists R(x, y) \exists S(Y, Z)$) может быть извлечена в виде подтеории из онтологии O из-за своей актуальности в области. Поэтому КШПО существует в справочной (эталонной) онтологии, которая обеспечивает его таксономическим и аксиоматическим контекстом.

- Отображение и композиция шаблонов требуют справочной онтологии для того, чтобы проверить непротиворечивость композиций, или сравнить наборы аксиом, которые должны быть отображены. Операции на КШПО зависят от операций из справочных онтологиях. Тем не менее, для пользователя шаблона, эти операции должны быть (почти) невидимым.

- КШПО может быть представлен на языке представления онтологии (в зависимости от его справочной онтологии), но его интуитивная и компактная визуализация является важным требованием. Это требует критического размера, так, чтобы его схематическая визуализация была эстетически приемлема и легко запоминающаяся. Например, норма \leftrightarrow случай КШПО включает только восемь классов, с несколькими, систематическими отношениями между ними: это делает их плотным, но управляемым.

- КШПО может быть элементом в частном порядке, где отношение упорядочивания требует, чтобы, по крайней мере, один из классов или отношений в шаблоне был специализирован. Иерархия КШПО может быть построена путем специализации или обобщения некоторых элементов (или классов, или отношений). Например, шаблон участия (объекта в событии) может быть специализирован к принятию участия в шаблоне государственного предприятия (агента в общественной деятельности с общедоступной значимостью).

- КШПО должен интуитивно иллюстрироваться и уловить соответствующие "базовые" понятия области. Независимо от общности, в которой выбран КШПО, он должен содержать центральные понятия и методы наиболее успешной практики, которые "делают ход рациональному мышлению" для эксперта в данной области для данной задачи.

- КШПО может часто создаваться из неформальных или упрощенных схем, используемых экспертам в этой области, вместе с поддержкой другого допускающего повторное использование КШПО или справочных онто-

логий и методологий для анализа онтологии предметной области. Как правило, эксперты спонтанно разрабатывают схемы, чтобы улучшить своего дело, а также для хранения необходимых ноу-хау. Эти схемы могут быть повторно спроектированы с использованием надлежащих методов [4,5,6].

- КШПО могут быть похожи на схемы базы данных, но шаблон определяется по отношению к справочной онтологии, и имеет общий характер, независимо от конструкции системы. В этом смысле она ближе к так называемому шаблонному моделированию данных [5,6], но КШПО изучается в контексте в справочной онтологии, делая ее более совместимой, чем шаблон моделирования данных.

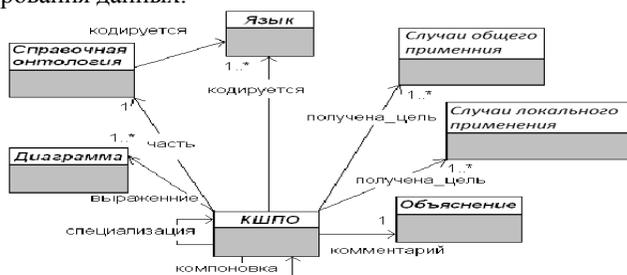


Рисунок 1. Аннотация шаблонов КШПО

Концептуальные шаблоны проектирования онтологий (КШПО являются ресурсом и методом технического содержания онтологии по Semantic Web. Шаблон (рис. 1) может быть использован для аннотирования КШПО как подтеории эталонных онтологий, чтобы совместно использовать их в предварительно отформатированных документах, а также для описания, визуализации, и выполнения операций над ними надлежащим образом. Шаблон КШПО состоит из:

- Два слота для общего использования, и локальных случаев использования, которые включают описание контекста, проблем и ограничений/требований.

- Два слота для адресуемой логики и справочной онтологии, используемых в качестве фона для шаблона.

- Два слота для «если любой» специализированного шаблона и составных шаблонов (наследованием и обратным наследованием, можно получить закрытые специализированные и расширяющиеся шаблоны).

- Два слота для максимального соотношения, которое кодирует пространство случая и его намеченную аксиоматизацию: принимается вся логика первого порядка с метауровнем, но слот может быть пустым, не влияя на функциональность фрейма КШПО.

- Два слота для объяснения подхода и его кодирования в предпочтительной логике.

- Последний слот для диаграммы классов, которая визуальнo воспроизводит подход.

Текущая работа направлена на укрепление инструмента, который помогает развитию, обсуждению, поиску и обмену КШПО по Semantic Web, и на установление образцово-теоретических и операционных основ манипулирования КШПО и обоснования. В частности, для КШПО чтобы быть реальным преимуществом в жизненном цикле онтологии, следующая функциональность должна быть доступной:

- Классификация КШПО, базируется или на прецедентах, которые она поддерживает, или на понятиях, которые она кодируют.

- Алгоритмы сопоставления с шаблоном для получения КШПО, например, спецификации естественного языка или эскиза онтологии.

- Поддержка специализации и структуры КШПО. КШПО p_2 специализирует другой p_1 , когда по крайней мере один из классов или свойств от p_2 является подклассом или подсвойством некоторого класса соответствующему свойству из p_1 , в то время как остаток от КШПО идентичен. КШПО p_2 расширяет p_1 , когда p_2 содержит p_1 , добавляя некоторый другой класс, свойство или аксиому. КШПО p_3 составляет p_1 и p_2 , когда p_3 содержит и p_1 , и p_2 . Формальная семантика этих операций обеспечивается базовой (справочной) моделей онтологии [6].

- Взаимодействие КШПО для визуализации, обсуждения, и создания базы знаний.

- Богатый набор метаданных для манипулирования КШПО и эксплуатации в приложениях.

Базовая правовая онтология и связанные с ней шаблоны

Потребность в расширенной типологии правовых объектов становится тяжелым грузом, даже для традиционных «восходящих» методов. Например, потребность соединить доказательную аргументацию с онтологией первых принципов следует исследовать для того, чтобы представить два вида структур, используемых в рассуждении: абстракция из случаев, и удовлетворенный набор ограничений (например, норм) [4,7].

Две опоры БПО являются: стратификация (наслаивание) и материализация.

Исходя из принципа стратификации, БПО обеспечивает модели и отношения для разнородных объектов из правовой области, будь они из материального, когнитивного, социального, или собственно правового мира [7]. Согласно стратификации, объекты из разных слоев могут быть совме-

щены пространственно-временным образом, но все же быть совершенно различными и (взаимно или односторонне) зависимыми. Например, объект принадлежит к материальному миру как биологический организм, но свойства организма не являются достаточными, чтобы характеризовать его как социальное лицо. В свою очередь, свойства социального лица не являются достаточными, чтобы характеризовать его как юридическое лицо.

Основанный на принципе материализации, БПО позволяет инженеру онтологии определить количественный анализ на правовых нормах или на отношениях (типовая материализации) [8], или на правовых фактах (символическая материализация)[8]. БПО расширяет словарь Описаний и Ситуаций для материализации. Например, интенциональные технические требования как нормы, соглашения, предметы и нормативные тексты могут быть представлены в той же самой области как их пространственная реализация случаев, выполнение соглашений, агентов, физические документы.

Для того чтобы описать часть онтологической выразительности БПО, вводится комплекс КШПО, в котором БПО в качестве справочной онтологии. Норма ↔ случай КШПО (рис. 2) используется в качестве шаблона для представления судебных дел, специализированных для типовых норм и случаев, например для расследования преступления, и составлено с другими шаблонами, например для проверки нормы конфликта.

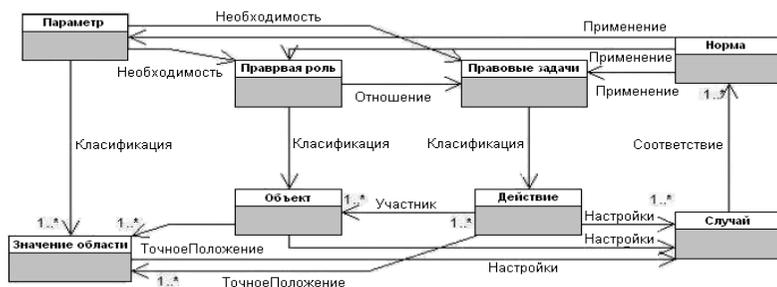


Рисунок 2. КШПО

Выводы

Был представлен обзор актуальности онтологии шаблонов для правовой инженерии знаний (ПИЗ).

Для некоторых универсальных вариантов использования ПИЗ (проверка конфликта, извлечение информации, и т.д.), были упомянуты некоторые проблемы и решения, и по обоснованию, и по моделированию. В то время как обоснованный аспект ПИЗ, является стремительной целью с интересными решениями, исходя из, например, смешанного различными механизмами вывода и классификаторами, аспект моделирования намного

менее разработан, несмотря на огромное количество литературы, которая сосредоточена на юридической и судебной практике, не говоря уже о работе в формальной онтологии и за ее пределами

Были коротко изложены несколько шагов к созданию репозитория правовых КШПО, вместе с некоторыми возможными направлениями для дальнейшего развития.

Литература

1. *Gangemi, A., Sagri M.T., Tiscornia D.: A Constructive Framework for Legal Ontologies.* In R. Benjamins, P. Casanovas, J. Breuker, A. Gangemi (eds.): *Law and the Semantic Web*, Springer, Berlin (2004).
2. *Gangemi, A., F. Fisseha, J. Keizer, J. Lehmann, A. Liang, I. Pettman, M. Sini, M. Taconet: A Core Ontology of Fishery and its Use in the FOS Project,* in (Gangemi and Borgo, 2004) (2004).
3. *Sagri M.T., Tiscornia D., Gangemi A., An Ontology-based Approach for Representing “Bundle-of-rights”,* M. Jarrar & A. Gangemi (eds.), 2nd International Workshop on Regulatory Ontologies, in *OTM Workshops*, Springer, 2004.
4. *Motta, E., Domingue, J., Cabral, L., Gaspari, M. (2003) IRS-II: A Framework and Infrastructure for Semantic Web Services.* 2nd International Semantic Web Conference (ISWC2003) 20-23 October 2003, Florida, USA (2003).
5. *McGuinness D.L. and van Harmelen F. (editors), OWL Web Ontology Language Overview,* W3C Recommendation, 10 February 2004, <http://www.w3.org/TR/2004/REC-owl-features-20040210/> (2004).
6. *Moore M.S. Legal Reality: A Naturalist Approach to Legal Ontology.* *Law and Philosophy* 21: 619–705, 2002.
7. *Domingos, P., Richardson, M. Markov Logic: A Unifying Framework for Statistical Relational Learning.* *Proceedings of the ICML-2004 Workshop on Statistical Relational Learning and its Connections to Other Fields* (pp. 49-54), 2004.
8. *Gangemi A., Mika P. Understanding the Semantic Web through Descriptions and Situation,* Meersman R, et al. (eds.), *Proceedings of ODBASE03*, Springer, Berlin, 2003.

УДК 004.81

Щодо деяких методів підвищення часової ефективності при фрактальному стисненні зображень

*Хімиченко І.В., Міжнародний науково-навчальний центр інформаційних технологій і систем НАН і МОН України, м. Київ,
stryker@ukr.net*

В даній статті наведені математичні основи фрактального стиснення з використанням пошуку найближчого сусіднього елементу в багатовимірному метричному просторі; розглянуті основні сучасні підходи до розв'язку задачі пошуку найближчого сусіда: аксіома трикутника, метод Орчарда, метод AESA, метричні дерева, метод просторово-чутливих хеш-функцій. Проведений аналіз і порівняння часової ефективності даних алгоритмів дозволяє виділити метод просторово-чутливих хеш-функцій як найбільш перспективний для подальшого дослідження з погляду його застосування при фрактальному стисненні зображень. Розглянуті питання адаптації методу просторово-чутливого хешування для фрактального стиснення зображень і викладені переваги даного методу в порівнянні з іншими методами пошуку найближчого сусіда.

Вступ

У роботі P.Indyk і R.Motwani представлений метод вирішення задачі наближеного пошуку найближчого елементу, заснований на просторово чутливому хешуванні (метод LSH - locality sensitive hashing). Автори LSH пропонують використовувати просторове хешування для організації пошуку в додатках баз даних, розпізнаванні образів, пошуку в архівах документів. Пропонується також модифікувати метод просторово чутливого хешування для організації пошуку найближчого сусіднього елементу при фрактальному стисненні зображень.

Мета

Метою статті є розгляд методу підвищення часової ефективності фрактального стиснення зображень на основі використання просторово-чутливого хешування.

Проблема

Однією з основних проблем при фрактальному стисненні зображень є розробка ефективного за часом методу зіставлення доменно-рангових областей.

Шляхи вирішення

На основі аналізу існуючих методів прискорення фрактального стиснення зображень зробити висновок про те, що найбільш перспективним і відповідним для подальшого поліпшення буде метод Саупа, який зведе задачу фрактального стиснення до задачі пошуку найближчого сусіднього елемента в багатовимірному метричному просторі за допомогою просторово-чутливого хешування.

Ідея

Для множини точок S , що містить проєкції доменних блоків $D_k^\perp = \phi(D_k)$ та рангових блоків $R_m^\perp = \phi(R_m)$ на $\mathfrak{R}^n \setminus \mathfrak{S}$, з мірою відстані (у нашому випадку Евкліда), сімейство LSH функцій визначене таким чином:

Множина хеш-функцій $H = \{h : S \rightarrow U\}$ називається (r_1, r_2, p_1, p_2) -чутливими для $d(\cdot, \cdot)$, якщо для будь-якої проєкції рангового і доменного блоку $R_m^\perp, D_k^\perp \in S$ на $\mathfrak{R}^n \setminus \mathfrak{S}$ виконується

$$\text{якщо } D_k^\perp \in B(R_m^\perp, r_1), \text{ то } \Pr_H [h(R_m^\perp) = h(D_k^\perp)] \geq p_1,$$

$$\text{якщо } D_k^\perp \notin B(R_m^\perp, r_2), \text{ то } \Pr_H [h(R_m^\perp) = h(D_k^\perp)] \leq p_2.$$

де $B(x, r)$ - гіперсфера радіусом r в сенсі міри відстані $d(\cdot, \cdot)$ з центром в точці x , а $\Pr_H [h(R_m^\perp) = h(D_k^\perp)]$ - вірогідність того, що хеш-функція утворює колізію для даних проєкцій рангового і доменного блоку R_m^\perp та D_k^\perp .

Застосування просторово-чутливого хешування для вирішення задачі фрактального стиснення зображень

Для того, щоб просторово-чутлива функція хешування була корисною з погляду її застосування до фрактального стиснення зображень, вона повинна задовольняти нерівностям $p_1 > p_2$ та $r_1 < r_2$.

У спеціальній літературі прийнято позначати задачу наближеного пошуку сусіднього елемента як (τ, c) -NN задачу, при цьому $r_1 = \tau$ та $r_2 = c \cdot \tau$. Покажемо, як просторово-чутливі функції можуть бути використані для вирішення (τ, c) -NN задачі при фрактальному стисненні:

для даного сімейства H хеш-функцій, що володіють параметрами (r_1, r_2, p_1, p_2) , збільшимо розрив між «високою» вірогідністю p_1 і «низькою» вірогідністю p_2 шляхом з'єднання декількох функцій. Зокрема, для параметра K , значення якого буде встановлено нижче, визначимо сімейство функцій $\Psi = \{g : S \rightarrow U^K\}$ таким чином, що $g(D_k^\perp) = (h_1(D_k^\perp), \dots, h_K(D_k^\perp))$, де $h_i \in H$. Для цілого числа L виберемо L функцій g_1, \dots, g_L з Ψ , незалежно і рівномірно, випадковим чином. Під час кроку передобчислень, збережемо кожну проекцію доменного блоку $D_k^\perp \in S$ в наборі $g_j(D_k^\perp)$ для кожного $j = 1, 2, \dots, L$. Оскільки загальна кількість таких множин може бути великою, залишимо тільки непорожні набори шляхом повернення до класичного хешування. Для того, щоб обробити ранговий блок R_m , проводимо пошук серед всіх множин $g_1(R_m^\perp), \dots, g_L(R_m^\perp)$. Оскільки можливо (хоча і маловірогідно) що спільна кількість доменів, збережених в цих множинах, є великою, то пошук домена припиняється після знаходження $3L$ елементів (включаючи дублікати). Нехай $D_1^\perp, \dots, D_t^\perp$ - знайдені елементи. Для кожного домена $D_{1, \dots, t}$ повертаємо відповідь «ТАК» (тобто даний домен є потенційним кандидатом побудови перетворення в ранговий блок R_m), якщо $D_{1, \dots, t} \in B(R_m^\perp, r_2)$, інакше повертаємо відповідь «НІ». Параметри K та L вибираються так, щоб гарантувати, що наступні дві властивості виконуються із заданою вірогідністю:

Якщо існує $D_k^* \in B(R_m, r_1)$, то $g_j(D_k^*) = g_j(R_m^\perp)$ для деякого $j = 1, \dots, L$

Загальна кількість колізій R_m^\perp з точками з $S - B(D_k^\perp, r_2)$ менша, ніж $3L$ (параметр визначений емпіричним шляхом) тобто:

$$\sum_{j=1}^L \left| (S - B(D_k^\perp, r_2)) \cap g_j^{-1}(g_j(D_k^\perp)) \right| < 3L$$

Якщо виконуються обидві властивості, то алгоритм є коректним. Звідси випливає, що якщо встановити $K = \log(1/p_2)$, і $L = n^\rho$, де

$$\rho = \frac{\ln(1/p_1)}{\ln(1/p_2)}, \quad (1)$$

то властивості (1) і (2) виконуються з постійною вірогідністю. Таким чином, отримуємо наступну теорему, яка стосується залежності ефективності вирішення $(\tau, c) - NN$ задачі при фрактальному стисненні зображень від параметрів LSH:

Передбачимо, що існує $(\tau, c \cdot \tau, p_1, p_2)$ - чутливе сімейство функцій хешування H для міри відстані $d(\cdot, \cdot)$. Тоді існує алгоритм для вирішення $(\tau, c) - NN$ задачі для міри $d(\cdot, \cdot)$, який використовує $O(dn_d + n_d^{1+\rho})$ пам'яті, з часом обробки одного запиту, визначеним $O(n_d^\rho)$ обчислень відстані і $O(n_d^\rho \log(1/p_2) \cdot n_d)$ обчислень хеш-функцій з H , де ρ визначений як (1).

Адаптація просторово-чутливих хеш-функцій до фрактального стиснення зображень на основі p -стійких розподілів

Стойкі розподіли визначаються як границі нормалізованих сум незалежних рівномірно розподілених випадкових змінних (нижче дано альтернативне визначення). Найбільш відомий приклад стійкого розподілу - це нормальний (гаусовий) розподіл. Проте, клас таких розподілів є значно ширшим. Розподіл μ над \mathcal{R} називається p -стійким, якщо існує $p \geq 0$ таке, що для будь-яких d , дійсних чисел v_1, \dots, v_d та змінних X_1, \dots, X_d з розподілом μ , випадкова змінна $\sum_i v_i X_i$ має такий же розподіл, як і змінна $(\sum_i |v_i|^p)^{1/p} X$ де X - випадкова змінна з розподілом μ .

Відомо, що стійкі розподіли існують для будь-якого $p \in (0, 2]$. Зокрема, гаусовий (нормальний) розподіл μ_G , визначений як функція щільності вірогідності

$$g(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2},$$

є 2-стійким.

Зауважимо, що з практичної точки зору, не дивлячись на недолік закритих форм функцій розподілу щільності, відомо, що можна створити p -стійку випадкову змінну з двох незалежних випадкових змінних, рівномірно розподілених на інтервалі $[0,1]$.

Використовуючи p -стійкі розподіли, побудуємо сімейство хеш-функцій H , адаптоване до фрактального стиснення зображень. Інтуїтивно, сімейство хеш-функцій має бути просторово-чутливим, тобто якщо два вектори R_m^\perp, D_k^\perp близькі один до одного (значення $\|R_m^\perp - D_k^\perp\|_p$ відносно невелике), то вони повинні з великою вірогідністю викликати колізію (мати одне і те ж значення хеш-функції) і, якщо вони розташовані далеко один від одного, то вірогідність колізії має бути мала. Нехай a - вектор розмірності d , елементи якого вибираються незалежно з p -стійкого розподілу. Скалярний добуток $\langle a, R_m^\perp \rangle$ проєктує кожен вектор на множину дійсних чисел. З p -стійкості випливає, що для двох векторів R_m^\perp, D_k^\perp відстань між їх проєкціями $\left| \langle a, R_m^\perp \rangle - \langle a, D_k^\perp \rangle \right|$ розподілено як $\|R_m^\perp - D_k^\perp\|_p X$, де X - це випадкова змінна, що вибрана з p -стійкого розподілу. Якщо «розділити» множину дійсних чисел на підмножини рівного розміру r та визначити значення хеш-функції для вектора залежно від того, на яку підмножину він проєктується, то інтуїтивно ясно, що така хеш-функція буде просторово чутлива в описаному вище сенсі.

Формально, кожна хеш-функція $h_{a,b}(v): \mathfrak{R}^d \rightarrow N$ відображує вектор v розмірності d (проєкція рангового R_m^\perp або доменного блоку D_k^\perp) на множину цілих чисел. Кожна хеш-функція в сімействі однозначно визначається вибором випадкового вектора a , визначеного вище, і дійсного числа b , вибраного випадковим чином з діапазону $[0, r]$. Для даних a, b визначимо хеш-функцію $h_{a,b}$ таким чином:

$$h_{a,b}(v) = \left[\frac{\langle a, v \rangle + b}{r} \right] \quad (2)$$

Тепер обчислимо вірогідність колізії хеш-функції, вибраної випадково згідно описаним вище міркуванням, для двох векторів R_m^\perp, D_k^\perp . Нехай $f_p(t)$ позначає щільність вірогідності абсолютного значення p -стійкого розподілу. Для двох векторів R_m^\perp, D_k^\perp Нехай $c = \|R_m^\perp - D_k^\perp\|_p$. Для випадково вибраного вектора a , елементи якого узяті з p -стійкого розподілу, $\left| \langle a, R_m^\perp \rangle - \langle a, D_k^\perp \rangle \right|$ розподілене як cX , де X - випадкова змінна, вибрана з p -стійкого розподілу. Оскільки b вибране випадково з діапазону $[0, r]$, легко побачити, що

$$p(c) = \Pr_{a,b} [h_{a,b}(R_m^\perp) = h_{a,b}(D_k^\perp)] = \int_0^r \frac{1}{c} f_p\left(\frac{t}{c}\right) \left(1 - \frac{t}{r}\right) dt$$

Для фіксованого параметра r вірогідність колізії монотонно зменшується із зростанням $c = \|R_m^\perp - D_k^\perp\|_p$. Згідно визначення, що було дане на початку параграфа, сімейство хеш-функцій $h_{a,b}(v)$ є (r_1, r_2, p_1, p_2) -чутливим для $p_1 = p(1)$ та $p_2 = p(c)$ при $r_2 / r_1 = c$.

Параметр r не був чітко визначений, оскільки він залежить від значень c та p . Для кожного даного c задача полягає у виборі такого кінцевого значення r , яке робить ρ якомога меншим.

Висновок

У даній статті розглянуті аспекти застосування просторово-чутливого хешування при фрактальному стисненні зображень. Описані достоїнства даного методу, його природна пристосованість для вирішення задач, що виникають при фрактальному стисненні. Стаття може бути використана для наведення оцінок вимог алгоритму до пам'яті і вірогідності негативної відповіді на запит для кожної рангової області.

Результати проведеної роботи говорять про доцільність застосування просторово-чутливого хешування при вирішенні задач фрактального стиснення зображень.

Література

1. *Indyk P., Motwani R.* Approximate nearest neighbor: towards removing the curse of dimensionality. Proceedings of the Symposium on Theory of Computing, 1998.
2. *Indyk P.* Stable distributions, pseudorandom generators, embeddings and data stream computation. Proceedings of the Symposium on Foundations of Computer Science, 2000.
3. *Saupe D., Hamzaoui R.* A guided tour of the fractal image compression literature, in: *New Directions for Fractal Modeling in Computer Graphics*, J. Hart (ed.), ACM SIGGRAPH'94 Course Notes 13, 1994.
4. *Friedman J.H., Bentley J.L., Finkel R.A.* An algorithm for finding best matches in logarithmic expected time, *ACM Trans. Math. Software* 3,3 (1977) 209-226.
5. *Darrell T., Indyk P., Shakhnarovich G. (eds.)*, *Locality-sensitive hashing using stable distributions*, *Nearest Neighbor Methods in Learning and Vision: Theory and Practice*, MIT Press, 2006.
6. *Boss R.D., Jacobs E.W.* Archetype classification in an iterated transformation image compression algorithm, in: *Fractal Image Compression Theory and Applications*, Y. Fisher (ed.), Springer-Verlag, New York, 1994.
7. *Arya S., Mount D.M., Netanyahu N.S., Silverman R., Wu A.* An optimal algorithm for approximate nearest neighbor searching, *Proc. 5th Annual ACM-SIAM Symposium on Discrete Algorithms* (1994) 573-582.

УДК 004.62:004.023

Метод штрафных функций для учета ограничений на решения задачи обеспечения групповой анонимности

*Чертов О.Р., к.т.н., доц., Тавров Д.Ю.,
Национальный технический университет Украины «КПИ», г. Киев
{chertov, dan.tavrov}@i.ua*

Предлагается метод учета нечетких ограничений, накладываемых на решения задачи обеспечения групповой анонимности с использованием меметических алгоритмов. Предлагается модификация функции приспособленности особей в популяции с помощью штрафных функций. Метод иллюстрируется решением задачи, основанной на реальных данных.

Введение

С развитием современных информационных технологий стало возможным анализировать большие объемы первичных данных, открытое распространение которых может, в частности, привести к утечке информации об определенной группе людей. Защита соответствующей информации предполагает выполнение групповой анонимизации данных.

В литературе предложен ряд методов обеспечения групповой анонимности данных. Большинство из них предполагают решение задачи в два этапа [1, с. 285]: на первом определяется, как обеспечить сокрытие уязвимых особенностей распределения информации о группе лиц, на втором — выполняется модификация данных с целью приведения их в соответствие с новым распределением. При модификации данных искажения, вносимые в исходные данные, должны быть минимальными.

В общем случае, обеспечение групповой анонимности можно выполнять в один этап. Соответствующую задачу можно рассматривать как задачу условной оптимизации, в которой допустимость решения определяется уровнем защищенности особенностей данных о группе респондентов, а оптимальность — объемом вносимых искажений.

Задачу обеспечения групповой анонимности (ЗОГА) с минимальным искажением обрабатываемых данных можно считать NP-полной [2, с. 12]. Для решения таких задач необходимо применять эвристические подходы, позволяющие получить удовлетворительный результат за допустимое время. В данной работе для обеспечения групповой анонимизации предлагается использовать меметические алгоритмы (МА) [3], кото-

рые обычно реализуют как эволюционные с элементами локального поиска [4, с. 173]. Выделяют четыре основных метода учета ограничений на решение задачи условной оптимизации при использовании МА:

1. Применение штрафных функций (penalty functions) [5].
2. Применение функций восстановления (repair functions) [6].
3. Сведение пространства поиска к пространству допустимых решений с помощью особого алфавита их кодирования [4, с. 215–216].
4. Применение дешифрующих функций (decoder functions) [7].

Применение методов учета 2–4 предполагает выполнение преобразования решений с целью приведения их в допустимую область. В большинстве практических случаев решения ЗОГА можно разбить на несколько непересекающихся допустимых областей (обеспечить сокрытие данных можно несколькими эквивалентными способами), которые могут содержать локальные оптимумы различной величины (уровень вносимых искажений в данные может отличаться для различных областей). Таким образом, невозможно явно преобразовать решение ЗОГА в допустимое без заведомого увеличения вносимых искажений.

В данной работе оценку допустимости решения предлагается выполнять неявно путем учета ограничений при оценивании качества каждого решения в форме штрафных функций.

Меметический алгоритм для обеспечения групповой анонимности

Схема обеспечения групповой анонимности

Пусть данные, анонимность которых нужно обеспечить, собраны в деперсонифицированный микрофайл \mathbf{M} . Каждая запись микрофайла r_i , $i = \overline{1, \rho}$, содержит значения атрибутов w_j , $j = \overline{1, \eta}$. Обозначим через \mathbf{w}_j множество всех значений атрибута w_j .

Обозначим через w_p параметризирующий атрибут микрофайла. Параметризирующее значение P — значение $P \in \mathbf{w}_p$ этого атрибута. Обо-

значим множество параметризирующих значений через $\mathbf{P} = \{P_1, \dots, P_{l_p}\}$.

При помощи этих значений можно разделить микрофайл на подмикрофайлы $\mathbf{M}_1, \dots, \mathbf{M}_{l_p}$. Например, если в качестве параметризирующего атрибута w_p взять территориальную принадлежность респондента, то под-

микрофайлы дадут соответствующее территориальное распределение респондентов.

Обозначим через w_{v_j} , $v_j \neq p \quad \forall j = \overline{1, t}$, *сущностные* атрибуты микрофайла. *Сущностная комбинация значений* V — элемент декартового произведения $\mathbf{w}_{v_1} \times \dots \times \mathbf{w}_{v_t}$. Обозначим множество сущностных комбинаций значений через $\mathbf{V} = \{V_1, \dots, V_{l_v}\}$. Записи микрофайла, значения сущностных атрибутов которых принадлежат \mathbf{V} , назовем *сущностными*.

Обозначим через $G(\mathbf{V}, \mathbf{P})$ *группу* респондентов микрофайла, уязвимые свойства распределения сущностных комбинаций значений которых нужно защитить при обеспечении групповой анонимности. *Задача обеспечения групповой анонимности* [8] — модификация микрофайла \mathbf{M} с целью маскирования уязвимых свойств данных. Общая схема обеспечения групповой анонимности [1] предполагает выполнение следующих шагов:

1. Подготовить деперсонифицированный микрофайл \mathbf{M} .
2. Определить группы респондентов $G_i(\mathbf{V}_i, \mathbf{P}_i)$, $i = \overline{1, k}$, данные о которых требуется защитить.
3. Для каждого i от 1 до k :
 - а) определить *целевое представление* данных — набор данных произвольной структуры $\Omega_i(\mathbf{M}, G_i)$, представляющий особенности $G_i(\mathbf{V}_i, \mathbf{P}_i)$ в удобном для их модификации виде;
 - б) определить *модифицирующий алгоритм* $A: \Omega_i(\mathbf{M}, G_i) \rightarrow \Omega_i^*(\mathbf{M}^*, G_i)$, с помощью которого получить *модифицированное целевое представление* и соответствующий ему *модифицированный микрофайл* \mathbf{M}^* .
4. Подготовить модифицированный микрофайл к опубликованию.

В работе в качестве целевого представления рассмотрим *количественный сигнал* $q = (q_1, \dots, q_l)$, элементом q_i которого является число сущностных записей в подмикрофайле \mathbf{M}_i , $i = \overline{1, l_p}$. Модификация сигнала эквивалентна обмену сущностных и несущностных записей между подмикрофайлами. Обмен должен быть попарным, чтобы сохранить общее число записей в подмикрофайлах. Искажения, вносимые при обмене, можно оценить с помощью *определяющей метрики* [2]

$$\text{InfM}(r_1, r_2) = \sum_{l=1}^{n_{ord}} \omega_l \left(\frac{r_1(I_l) - r_2(I_l)}{r_1(I_l) + r_2(I_l)} \right)^2 + \sum_{k=1}^{n_{nom}} \gamma_k \chi^2(r_1(J_k), r_2(J_k)), \quad (1)$$

где I_l (J_k) — l -ый порядковый (k -ый номинальный) *определяющий атрибут* (атрибут, значение которого принимается во внимание при внесении искажений в микрофайл), $r(\cdot)$ — оператор, возвращающий значение своего аргумента как атрибута записи r , $\chi(v_1, v_2)$ — функция, равная χ_1 в случае, если ее аргументы принадлежат одной категории, χ_2 — в противном случае, ω_l и γ_k — неотрицательные весовые коэффициенты.

Общая структура меметического алгоритма

Рассмотрим меметический алгоритм для решения ЗОГА:

1. Случайным образом сгенерировать популяцию $P = \{U_i\}$ из μ особей, применить оператор локального поиска $S(U_i) \forall i = \overline{1, \mu}$.
 2. Рассчитать значение функции приспособленности $f(U_i) \forall i = \overline{1, \mu}$.
 3. Если выполняется условие завершения, остановиться. В противном случае — перейти на шаг 4.
 4. Отобрать λ пар родительских особей, поместить их в популяцию P' .
 5. Применить оператор скрещивания $R(U_{i_1}, U_{i_2})$ к каждой паре родительских особей $\langle U_{i_1}, U_{i_2} \rangle$ из P' , $i_1 \in \overline{1, \lambda}$, $i_2 \in \overline{1, \lambda}$, $i_1 \neq i_2$, получая после каждого скрещивания по два потомка. Поместить потомков в популяцию P'' .
 6. Применить оператор мутации $M(U_j)$ ко всем особям U_j из популяции P'' , $j = \overline{1, \lambda}$.
 7. Применить оператор локального поиска $S(U_j) \forall j = \overline{1, \lambda}$.
 8. Рассчитать значение функции приспособленности $f(U_j) \forall j = \overline{1, \lambda}$.
 9. Отобрать μ наиболее приспособленных особей из популяции $P \cup P''$ и поместить их в популяцию P вместо текущих.
 10. Перейти на шаг 3.
- Каждая особь в популяции представляет собой матрицу U :

1. Элементы первого столбца матрицы $u_{i1} \forall i = \overline{1, Q}$ — индексы подмикрофайлов, из которых необходимо перенести существенные записи.

2. Элементы второго столбца матрицы $u_{i2} \forall i = \overline{1, Q}$ — индексы записей из подмикрофайлов $\mathbf{M}_{u_{i1}}$, которые необходимо перенести.

3. Элементы третьего столбца матрицы $u_{i3} \forall i = \overline{1, Q}$ — индексы подмикрофайлов, в которые необходимо перенести существенные записи.

4. Элементы четвертого столбца матрицы $u_{i4} \forall i = \overline{1, Q}$ — индексы записей из подмикрофайлов $\mathbf{M}_{u_{i3}}$, на которые необходимо заменить записи, определяемые элементами u_{i2} .

Количество строк Q в особях может отличаться от особи к особи. Число вхождений индекса i в первый столбец любой особи не должно превышать q_i , а каждая пара значений $\langle u_{i1}, u_{i2} \rangle$ ($\langle u_{i3}, u_{i4} \rangle$) не может встречаться в некоторой особи более одного раза.

Особенности учета ограничений в алгоритме

В данной работе предлагается использовать функцию приспособленности особи в меметическом алгоритме вида

$$f(U) = \Theta(U) + \Phi(U) + \Psi(U), \quad (2)$$

где $\Theta(U)$ — целевая функция оценки качества решения ЗОГА с точки зрения минимизации вносимых в микрофайл искажений, $\Phi(U)$ — штрафная функция, оценивающая качество решения ЗОГА с точки зрения маскирования уязвимых особенностей сигнала, $\Psi(U)$ — штрафная функция, уменьшающая приспособленность особей с большим количеством строк. Все три приведенные функции принимают значения в интервале $[0, 1]$. В качестве целевой функции предлагается использовать

$$\Theta(U) = \frac{C_{\max} - \sum_{i=1}^Q \text{InfM}(\mathbf{M}_{u_{i1}}(u_{i2}), \mathbf{M}_{u_{i3}}(u_{i4}))}{C_{\max}}, \quad (3)$$

где C_{\max} — наибольшее значение, достигаемое суммарной метрикой (1), $\mathbf{M}_i(j)$ — оператор, возвращающий j -ую запись \mathbf{M}_i . Функция (3) позволяет оценить качество решения ЗОГА, поскольку является обратной к

суммарному значению метрики (1) — уровню вносимых в микрофайл искажений.

Ограничения, накладываемые на решение ЗОГА, можно сформулировать как *нечеткие ограничения* вида

$$\begin{aligned} R(X_{i_1}) : X_{i_1} \text{ is } A_{i_1}, \\ \dots \\ R(X_{i_k} | q_{i_1}^*, \dots, q_{i_{k-1}}^*) : X_{i_k} \text{ is } A_{i_k}, \\ R(X_{\bar{J}} | q_J^*) : \sum_{i \in J} q_i^* = Q_0 - \sum_{j \in J} q_j^*, \end{aligned} \quad (4)$$

где $\mathbf{q}^* = (q_1^*, \dots, q_{l_p}^*)$ — значение переменной $X = (X_1, \dots, X_{l_p})$, $R(X_{i_k} | q_{i_1}^*, \dots, q_{i_{k-1}}^*)$ — нечеткое ограничение на переменную X_{i_k} , обусловленное указанными значениями модифицированного количественного сигнала, A_{i_k} — нечеткое множество, соответствующее ограничению, $J = (i_1, \dots, i_k)$ — последовательность индексов, являющаяся собственной подпоследовательностью $I = (1, \dots, l_p)$, \bar{J} — дополнение J до I , Q_0 — общее число сущностных записей в микрофайле.

Первую штрафную функцию предлагается определить как функцию, показывающую степень выполнения указанных ограничений. При этом последнее ограничение из (4) автоматически учитывается с помощью особенностей структуры особой в меметическом алгоритме.

Практические результаты

Рассмотрим задачу маскирования территориального распределения военных, работающих в штате Массачусетс (США), на основе данных переписи населения США [9]. Возьмем в качестве сущностного атрибута «Пребывание на военной службе», сущностного значения — «Активную службу», параметризирующего атрибута — «Место работы», параметризирующих значений — коды статистических областей штата Массачусетс от 25010 до 25120 с шагом 10. Исходный количественный сигнал для данной задачи представлен на рис. 1.

Чтобы маскировать экстремальные значения количественного сигнала в отсчетах 2, 7, 9 и 12, штрафная функция должна соответствовать таким ограничениям, степень выполнения которых наиболее низка для сигналов со значениями в этих отсчетах, равных исходным, и повышается с их

уменьшением до некоторых предопределенных значений. Возьмем в качестве таких ограничений следующие:

$$R(X_2): X_2 \text{ is } A_2,$$

$$R(X_7): X_7 \text{ is } A_7,$$

$$R(X_9): X_9 \text{ is } A_9,$$

$$R(X_{12}): X_{12} \text{ is } A_{12},$$

$$R(X_{\bar{J}} | q_J^*): \sum_{i \in J} q_i^* = 186 - \sum_{j \in J} q_j^*,$$

где $J = \{2, 7, 9, 12\}$, а нечеткие множества определяются функциями

$$\mu_{A_2}(u_2) = \begin{cases} 1, & u_2 \leq 20 \\ 1 - 2\left(\frac{u_2 - 20}{47}\right)^2, & 20 \leq u_2 \leq 43,5 \\ 2\left(\frac{u_2 - 67}{47}\right)^2, & 43,5 \leq u_2 \leq 67 \\ 0, & u_2 \geq 67 \end{cases}, \mu_{A_7}(u_7) = \begin{cases} 1, & u_7 \leq 25 \\ 1 - 2\left(\frac{u_7 - 25}{5}\right)^2, & 25 \leq u_7 \leq 27,5 \\ 2\left(\frac{u_7 - 30}{5}\right)^2, & 27,5 \leq u_7 \leq 30 \\ 0, & u_7 \geq 30 \end{cases},$$

$$\mu_{A_9}(u_9) = \begin{cases} 1, & u_9 \leq 25 \\ 1 - 2\left(\frac{u_9 - 25}{3}\right)^2, & 25 \leq u_9 \leq 26,5 \\ 2\left(\frac{u_9 - 28}{3}\right)^2, & 26,5 \leq u_9 \leq 28 \\ 0, & u_9 \geq 28 \end{cases}, \mu_{A_{12}}(u_{12}) = \begin{cases} 1, & u_{12} \leq 25 \\ 1 - 2\left(\frac{u_{12} - 25}{13}\right)^2, & 25 \leq u_{12} \leq 21,5 \\ 2\left(\frac{u_{12} - 38}{13}\right)^2, & 21,5 \leq u_{12} \leq 38 \\ 0, & u_{12} \geq 38 \end{cases}.$$

В вышеуказанных ограничениях значение во втором отсчете уменьшается до 20, а в остальных — до 25, что гарантирует сокрытие наибольшего максимума в исходном количественном сигнале.

Возьмем определяющие номинальные атрибуты «Пол», «Возраст», «Латиноамериканское происхождение», «Семейное положение», «Образование», «Гражданство», «Совокупный доход в 1999 г.». Положим, что в (1) $\gamma_k = 1 \quad \forall k = \overline{1, 7}$, $\chi_1 = 1$, $\chi_2 = 0$, т.е. определяющая метрика показывает общее число значений атрибутов микрофайла, которые надлежит изменить, чтобы скрыть особенности количественного сигнала.

Функция приспособленности (2) примет вид

$$f(U) = \frac{1099 - \sum_{i=1}^Q \sum_{k=1}^7 \text{sign} |\mathbf{M}_{u_{i1}}(u_{i2}, B_k) - \mathbf{M}_{u_{i3}}(u_{i4}, B_k)|}{1099} +$$

$$+ \prod_{j \in J} \mu_{A_j}(|U|^{(j)}) + \frac{1}{1 + e^{0,5(Q-90)}},$$

где B_k — k -ый определяющий атрибут, $\mathbf{M}_j(i, B_k)$ — оператор, возвращающий значение атрибута B_k i -ой записи подмикрофайла \mathbf{M}_j , $|U|^{(j)}$ — число вхождений индекса j в столбцы 1 и 3 особи U .

Параметры используемого меметического алгоритма, такие, как метод скрещивания, мутации, отбора и пр., были взяты аналогичными представленным в [1]. В качестве критерия окончания работы алгоритма была взята генерация 1000 поколений особей.

По результатам выполнения 30 запусков алгоритма было получено 1454 решений ЗОГА. Средняя суммарная метрика (1) по этим решениям равна 59,560, что меньше, чем получено в [1]. Два решения с наименьшими метриками представлены на рис. 1.

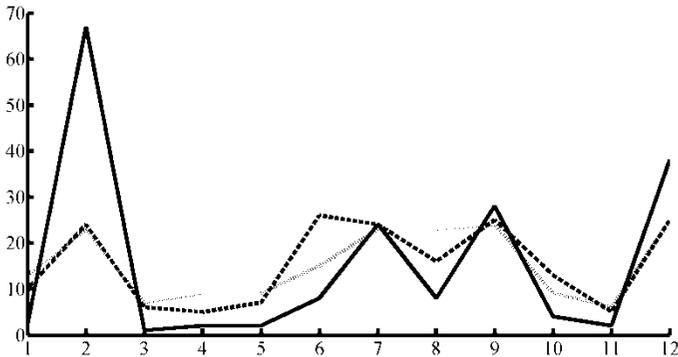


Рис. 1. Исходный (сплошная линия) и модифицированные количественные сигналы с суммарными метриками 44 и 45

Выводы

В работе предложен метод учета ограничений, накладываемых на решения задачи обеспечения групповой анонимности данных с использованием меметических алгоритмов. Для этого разработан особый вид функции приспособленности индивидов в популяции, обеспечивающий учет нечетких ограничений и тем самым позволяющий повысить качество получаемых решений.

Перспективными являются дальнейшие исследования в направлении повышения производительности предложенного алгоритма в зависимости от значений его параметров.

Литература

1. *Chertov O., Tavrov D.* Memetic Algorithm for Solving the Task of Providing Group Anonymity // Advance Trends in Soft Computing [ed. M. Jamshidi, V. Kreinovich, J. Kacprzyk]. – Studies in Fuzziness and Soft Computing, vol. 312. – P. 281–292.
2. *Chertov O.* Group Methods of Data Processing. – Raleigh: Lulu.com. – 2010. – 156 p.
3. *Moscato P.* On evolution, search, optimization, genetic algorithms and martial arts: Toward memetic algorithms // C3P Report 826: Caltech Concurrent Computation Program. – Caltech, CA, 1989. – P. 33–48.
4. *Neri F., Cotta C.* A Primer on Memetic Algorithms // Neri F., Cotta C., Moscato P. (eds.): Handbook of Memetic Algorithms. – Berlin, Heidelberg: Springer-Verlag, 2012. – P. 43–52.
5. *Smith A.E., Coit D.W.* Penalty Functions // Evolutionary Computation 2. Advanced Algorithms and Operators [ed. T. Bäck, D.B. Fogel, Z. Michalewicz]. – Bristol, Philadelphia: Institute of Physics Publishing, 2000. – P. 41–48.
6. *Michalewicz Z.* Repair algorithms // Evolutionary Computation 2. Advanced Algorithms and Operators [ed. T. Bäck, D.B. Fogel, Z. Michalewicz]. – Bristol, Philadelphia: Institute of Physics Publishing, 2000. – P. 56–61.
7. *Michalewicz Z.* Decoders // Evolutionary Computation 2. Advanced Algorithms and Operators [ed. T. Bäck, D.B. Fogel, Z. Michalewicz]. – Bristol, Philadelphia: Institute of Physics Publishing, 2000. – P. 49–55.
8. *Chertov O., Tavrov D.* Data Group Anonymity in Microfiles // Вісник інженерної академії України, 2010. – №2. – С. 159–164.
9. U.S. Census 2000. 5-Percent Public Use Microdata Sample Files, <http://www.census.gov/census2000/PUMS5.html>

УДК 681.3: 581.51

К 15-летию киевской лаборатории искусственного интеллекта

*Казаков А.И., Обухов Л.И., Ткаченко Н.А.,
Хоружий В.П., Хоружая И.А.*

*Киевская лаборатория искусственного интеллекта, г. Киев,
kazakov.alex@gmail.com*

В данной статье кратко представлены некоторые из этапных разработок и проектов осуществленных в киевской лаборатории искусственного интеллекта (КЛИИ) за эти 15 лет. Идеологически, КЛИИ можно считать наследницей и продолжателем разработок в области биомедикибернетики и искусственного интеллекта созданных в школе академика Н.М. Амосова и д.т.н. Э.М.Куссуля.

Введение

В 2014 году киевская лаборатория искусственного интеллекта будет отмечать свое 15-летие. Целью создания нашей лаборатории было проведение исследований и разработок в интересных и стратегически ключевых, по нашему мнению, направлениях развития в таких областях знания как искусственный интеллект и медицинская кибернетика. Проведение исследований и разработок в этих областях знания является достаточно трудоемким делом, но оно опиралось на наш опыт и разработки, сделанные в предшествующее 10-летие в рамках проекта «Нейротехника». В 2000 г. в сборнике ИОИ-2000 вышли наши программные тезисы «На пути к машине общения» о планах работ КЛИИ на ближайшие годы [1].

В данной статье мы кратко представим некоторые из наших этапных разработок и проектов осуществленных за эти 15 лет. Наша лаборатория была основана в 1999 году. С 2000 г. мы приступили к выполнению проекта по созданию ядра системы искусственного интеллекта (ИИ) в сильной форме. Основными этапами в этом проекте были следующие 4 этапа: 1. Создание базовой среды программирования. 2. Создание «Машины общения». 3. Создание «Машины понимания». 4. Создание «Машины познания».

Описание этапов работ по проекту

1. Создание базовой системы языков программирования и общей для них программной среды на основе собственных вариантов языков про-

граммирования с необходимыми на наш взгляд свойствами. Были разработаны языки МикроЛисп-13 и Пролог-XXI (LastProlog), в которых были доработаны их логические структуры. Подробнее можно ознакомиться на нашем сайте «NeuroTechnica.info». В качестве общей среды была создана система «Вавилон», в которой средообразующими языками были выбраны языки Перл и Си. Эти работы в основном завершили в 2002 г.

2. На втором этапе работ было создано лингвистическое обеспечение для «Машины общения». К 2003 году были созданы такие системы как «Конкорданс текстов на русском языке», система грамматических анализаторов русского языка и другие инструментальные и словарные системы, которые обеспечивали возможность работы с предложениями на русском языке из открытого множества текстов.

3. На третьем этапе главной целью было создание «Машины понимания» текстов на русском языке. Было осознано как необходимое условие, что для этого требуется создание самой среды программирования на русском языке и обеспечением ее совместимости с англоязычной средой программирования. Эти задачи были решены на основе разработанной нами программной среды «Велиар». Основным свойством которой, была способность к буквальному пониманию выражений на естественном языке. Этот уровень понимания был создан на основе сильной формы ИИ, а не слабой, в которой только имитируется способность к пониманию. Этот этап работ был в основном завершён к 2005 году и описан в статье [2].

4. Последним этапом проекта было создание «Машины познания» (синонимом здесь является понятие «самообучающаяся система»). Была разработана теория когнитронов, как механизмов способных познавать новую информацию, поступающую из программной среды и символизировать ее, а также была создана их конструктивная реализация к 2008 году и описана в статье [3].

Перспективные разработки

Последующие исследования результатов данных разработок привели к пониманию, что на их основе открывается возможность осуществить работы по следующим новым проектам:

1. Создание нового языка и системы программирования, еще более приближенных, по своим свойствам к естественным языкам и понятийной системе человека.

2. Создание системы знания в форме «Древа Знания». Такая система будет способна взаимодействовать с человеком посредством общения и инструментально поддерживать стремления данного человека к овладению

знаниями и умениями на разных этапах его обучения и практики, а также обеспечивать непрерывность его самообразования в течении всей его жизни. Система типа «Древа Знания» может стать носителем и гарантом нового типа авторского права, которое не будет служить тормозом для инноваций в области информатики.

3. Создание системы формирования информационной копии человека и обеспечения его виртуального присутствия в социальных системах в форме системы типа «Аватарум». Создание технологий, которые будут способны интеллектуально помогать человеку и даже замещать его, когда это нужно данному человеку в его деятельности, как в виртуальности, так и в расширенной реальности данного социума (проект «Аватарум»).

О проекте «Эмбрио»

Одним из наиболее важных направлений разработок является создание нового языка программирования, который был бы базисно адекватен представляемой модели реальности в программных системах. Новый язык должен быть, по нашему мнению, таким, чтобы он мог единообразно представлять структуры и программ, и данных, и самих программных процессов. Такую единообразность можно обеспечить посредством древообразной структуры и поэтому, язык ЛИСП является естественным прототипом данного языка. В качестве рабочего названия языка мы пока приняли имя «Эмбрио». Разработка ведется на основе минимальной версии языка ЛИСП, содержащей всего 12 внутренних операторов и 1 внешний оператор-выдачи на печать. Такой минимальный ЛИСП мы называем микролиспом или М-Лисп.

Язык Эмбрио позволит адекватно представлять все четыре базовых аспекта пространства-времени, а именно: для времени - это его событийность и течение, а для пространства – это место и протяженность. Для человека пространственно-временные соотношения даются ему через чувства и ощущения. Чувства формируются как сплетение аспектов течения во времени и протяженности в пространстве, а ощущения формируются как сплетение аспектов событийности во времени и места в пространстве. Чувства и ощущения представляются в сознании человека посредством значений и образов. Человек получает в опыте жизни контекстно зависящую систему знаний в форме, как многообразия данного значения, так и многозначия данного образа.

Для фиксации таких форм знания и конструктивной работы с ними необходимы языки программирования со свойствами, подобными свойствам понятийной системы человека и реализуемой посредством естественных языков. В настоящее время таких языков программирования пока

нет. Мы надеемся, что нам удастся полнее воплотить эти свойства в языке типа Эмбрио.

О проекте «Велиар»

Одной из наших попыток создать подобную систему программирования, демонстрирующую свойства естественного языка, была разработанная нами к 2005 году система «Велиар», благодаря которой нам удалось создать систему понимания текстов на естественном языке (русском). Тогда удалось достигнуть первых двух уровней понимания, а именно, буквального и идиоматического уровней понимания [2]. Всего же, по нашей классификации, мы выделяем 6 уровней понимания и нужно, в особых случаях, достигать по очереди все 6 уровней, требуемых для полного понимания данного текста, т.е. от буквального и идиоматического уровней понимания и далее ассоциативно переносного и т. д., вплоть до герменевтического уровня, - где сообщение зашифровано в тексте посредством иносказания.

Система понимания «Велиар» была способна отвечать на любые обстоятельственные вопросы по данному тексту, а именно: Что происходило? Кто это делал? Когда это происходило? В каком количестве? С какой целью? Почему это произошло? Каким образом это происходило? и другие варианты основных декларативных вопросов. Совокупность ответов на весь спектр возможных вопросов по данному тексту и определяло меру формального понимания системой данного текста. Система «Велиар» продемонстрировала очень высокую степень понимания для буквального и идиоматического уровней понимания. В качестве дополнения к данной системе в рамках проекта создания «Машины общения» были разработаны еще две системы, которые обеспечивали речевой уровень общения с ней: система распознавания слитной речи на основе НММ-технологии и система синтеза речи на основе трифонных моделей фонем. При вопросоответном характере общения с системой по теме данного текста достигался вполне приемлемый для лабораторной практики уровень распознавания речи, уровень понимания вопроса и уровень разборчивости речи, синтезированной по текстовому ответу системы «Велиар», с возможностью учета интонационного и эмоционального содержания сообщения.

Для продвижения к следующим уровням понимания текстов стали видны следующие две ключевые проблемы:

1. Необходимость в способности системы распознавать самоподобные структуры в собственных для системы программах, данных и процессах;
2. Необходимость в собственной познавательной способности у системы понимания.

Решение первой проблемы будет обеспечиваться новым языком программирования типа «Эмбрио». Для решения 2-й проблемы нами в 2007 году были начаты работы по созданию ядра системы познания. В результате была создана первая версия системы познания, которую мы назвали «Когнитрон» (см. статью [3]). Работы в этих направлениях продолжаются.

О проекте «КАНТ»

Другим важным направлением работ было создание экспертной системы диагностики состояния человеческого организма «КАНТ» и развития ее до системы автоматической диагностики и последующей ее стыковки с системой понимания «Велиар». Ниже дается краткое описание системы «КАНТ», разработка которой началась в 1994 году, а развитие продолжается и поныне. В основе экспертной системы «КАНТ» лежат следующие составляющие:

- система диагностики состояния организма человека, базирующаяся на методе Накатани, современной нозологической системе классификации болезней и на собственных разработках исследовательской группы: Казаков А.И., Ткаченко Н.А.

- система измерений электрической активности в биологически активных точках организма (БАТ),

- технология измерения токов в БАТ посредством деполяризующихся электродов, эта технология была разработана ранее в объединении ФИЗЛИ Инт.

- методика для проведения лечебных процедур посредством биофореза лекарственных веществ за счет собственных управляющих токов организма человека созданная на основе разработок ФИЗЛИ Инт. и собственного опыта применения биофореза.

Практика работы по диагностике и лечению заболеваний (преимущественно желудочно-кишечного тракта) на протяжении многих лет (более 18 лет), показала высокую точность диагностики КАНТ и эффективность лечения на ее основе. Диагностика и лечение производились в лаборатории диагностики и лечения, созданной на базе лечебно-профилактического санатория «Феофания» НАН Украины.

Точность работы экспертной системы диагностики «КАНТ» обеспечивается:

- высокой чувствительностью метода измерений на основе деполяризующихся электродов,

- разработанной авторами системы классификации состояний органов и систем организма по соотношениям значений величин токов, генерируе-

мых в организме человека его симпатической и парасимпатической системами, которые управляют состояниями организма человека,

- учетом в системе более 800 нозологических форм заболеваний, распознаваемых посредством соответствующих систем уравнений,

- большим практическим опытом врача-диагноста, участвующего в развитии системы КАНТ,

- постоянным развитием метода на основе сопоставления результатов авторской системы диагностики и диагностики проводимой на основе других методов исследования состояний организма (УЗИ, ЭКГ, анализы и др.).

Все это позволило создать в 2008 г. на основе системы КАНТ экспертную систему автоматической диагностики состояний органов и систем организма (САД КАНТ). Данная экспертная система «КАНТ» может лечь в основу и будущего интернет-сервиса по удаленной и автоматической диагностике, необходимой в системах телемедицины. Основанием для этого являются технологии, разработанные в киевской лаборатории искусственного интеллекта, а именно: система защиты личной и медицинской информации с управляемым доступом к данным, система аутентификации по физической и цифровой подписям, система понимание текстов на естественном языке для организации более эффективного и качественного взаимодействия экспертной системы с врачом-куратором и пациентом для составления анамнеза заболевания путем направленного опроса пациента, проведение консультационных и профилактических бесед в соответствии с запросом пациента и рекомендациями врача-куратора. Все перечисленное выше составляет необходимую основу также и для создания лечебно-диагностического, профилактического и информационного центра нового типа.

Заключение

Коллектив КЛИИ за эти годы создал и много других разработок и программных продуктов. Среди них можно отметить такие: ряд систем обучения украинскому языку для англоговорящих людей, спеллчекер для азербайджанского языка, встроенный во все продукты Microsoft Office, технологии, обеспечивающие распознавание слитного рукописного текста и ряд других разработок, с некоторыми из них можно ознакомиться на нашем сайте www.neurotechnica.info.

Литература

1. Казаков А.И., Хоружий В.П., Обухов Л.И., На пути к машине общения. // Тезисы докладов Международной научной конференции «Интеллектуализация обработки информации». / Симферополь: Крымский научный центр НАН Украины, Таврический национальный университет, 2000. – С. 34-35.
2. Казаков А.И., Обухов Л.И., Система понимания естественного языка. // Материалы конференции «Интеллектуальный анализ информации». / Сб. трудов. – Изд. «Просвіта», г. Киев. - 2005.- С.157-166.
3. Казаков А.И., Обухов Л.И., Проект теории когнитронов. // Материалы конференции «Интеллектуальный анализ информации». / Сб. трудов. - Изд. «Просвіта», г. Киев. - 2008.- С. 228-237.

Содержание

Святогор Л. А., Величко В. Ю. Памяти В. П. Гладуна	4
Валькман Ю. Р. О гештальтах в мышлении: определения, классификация и свойства	11
Голенков В. В., Гулякина Н. А. Смысловые модели представления и обработки знаний	23
Кузнецов О. П. Еще одна концепция когнитивной семантики	28
Бабенко А. Е. Интеллектуализация удаленного процесса обучения иностранному языку	37
Байбуз О. Г., Луценко О.П. Модель оцінювання ймовірності виникнення розладання у нестационарних процесах з множинними розладаннями	44
Боргест Н. М., Бурдюгова В. В. Разработка онтологии документооборота выполнения заказа на предприятии	49
Боргест Н. М., Сокова Н. А. Разработка онтологии нештатных ситуаций на космической станции	54
Валькман Ю. Р., Жолнарская А. А. Анализ средств организации хранилищ данных в информационно- аналитических системах статистических данных	58
Вовк О.Л. Моделирование зрительного анализатора человека для выделения контуров объектов изображений	71
Гаврилов С.В., Дробышев Ю.П. Класифікація руху людини з використанням методу швидкого відключення для рекурентних нейронних мереж	77
Дробязко І.П., Клятченко Я.М., Тарасенко-Клятченко О.В., Тарасенко В.П., Тесленко О.К. Методика аналізу та оцінки достовірності функціонування логічних мереж на основі суперпозицій булевих функцій	83

Yu.Zorin An Artificial Bee Colony algorithm for data clustering	92
Жилякова Л.Ю. Сетевая парадигма в когнитивных науках	99
Карасюк В.В. Представлення правових знань на основі онтологій.	107
Комаров-Ермолов А. И. Новые подходы к извлечению знания: архитектоническая парадигма	114
Копичко О.С., Рибачок Г.І., Копичко С.М., Мороченець О.І. Інтелектуальна система автоматизованої ідентифікації клієнтів мобільної мережі	121
Кравець П.І., Шимкович В.М., Зубенко Г.А. Моделі штучних нейронних мереж при їх апаратно-програмній реалізації на FPGA	127
Кулаков Ю. А., Воротников В. В., Данелюк Д.О. Межкластерная многопутевая маршрутизация в сложной сети	132
Литвинов В.А., Майстренко С. Я., Хурцилава К.В. К оценке качества базового словаря в системе автоматического обнаружения ошибок пользователя и возможностей его улучшения ...	139
Олефір О.С., Гусак Б.І. Прогнозування волатильності акцій на основі гетероскедастичних моделей.	144
Путренко В.В. Інтелектуальний аналіз безпеки виникнення природних пожеж на основі геоінформаційних технологій	150
Рогущина Ю.В. Разработка методов самообучения поисково-рекомендующей системы с целью совершенствования ее поведения	155
Савеленко О.К. Кластерний аналіз в задачах обробки інформації	162
Савчук О.В., Кривенко К.С. Інтелектуальний аналіз діагностичної інформації складних технічних комплексів	172

Северин Н. В., Леоненко Л. Л. О некоторых методах “борьбы с неустранимой неоднозначностью” в сетях телекоммуникаций	178
Сирота С.В., Величко В.Ю., Ліскін В.О. Застосування паралельних обчислень для побудови зростаючих пірамідальних мереж.	185
Снарский А.А., Ландэ Д.В. Эффект «взрывной перколяции» в контентной сети.	190
Танченко С. С., Титенко С. В., Гагарін О. О. Усунення мовної неузгодженості в тестових завданнях, згенерованих на основі понятійно-тезисної моделі.	196
Темников В.А., Темникова Е.Л., Конфорович И.А. Построение системы аутентификации операторов и распознавания слов по голосу	201
Терлецький Д.О. Представлення знань за допомогою об’єктно-орієнтованих динамічних мереж	207
Титенко С.В. Інформаційно-логічні та архітектурні засади універсальних систем керування web-контентом	214
Трифонов Е.О. Сжатие данных на основе бинарного дерева.....	221
Хала К. А. Функциональность и методы прикладной правовой онтологии.	227
Хіміченко І.В. Щодо деяких методів підвищення часової ефективності при фрактальному стисненні зображень	234
Чертов О.Р., Тавров Д.Ю. Метод штрафных функций для учета ограничений на решения задачи обеспечения групповой анонимности	241
Казаков А.И., Обухов Л.И., Ткаченко Н.А., Хоружий В.П., Хоружая И.А. К 15-летию Киевской лаборатории искусственного интеллекта.....	250

НТУУ «КПІ» Факультет прикладної математики

Збірка праць
XIV міжнародної наукової конференції
імені Тетяни Архипівни Таран
ІНТЕЛЕКТУАЛЬНИЙ АНАЛІЗ ІНФОРМАЦІЇ
IAI-2014

наукове видання

РЕДАКЦІЙНА КОЛЕГІЯ

Д.т.н., проф., академік НАН України Згуровський М.З.,
д.т.н., проф., Молчанов О. А.,
д.ф.-м.н., проф. Осипов Г.С.,
д.т.н., проф., Валькман Ю.Р.
д.т.н., проф., Гладун В.П.,
д.т.н., проф., Голенков В.В.,
д.т.н., проф., Дичка І.А.,
д.ф.-м.н., проф., Донской В.Й.,
д.т.н., проф., Кузнецов О.П.,
д.т.н., проф., Стефанюк В. Л.,
д.т.н., проф., Тарасенко В.П.,
д.т.н., проф., Фінн В.К.,
д.т.н., проф., Хорошевський В.Ф.
к.т.н., доц. Григор'єв О.В.
к.т.н., доц. Чертов О.Р.

Головний редактор к.т.н. Сирота С.В.
Відповідальний редактор Темнікова О.Л.
Відповідальний за випуск Копичко С. М.



Промислово-торговельна фірма «Просвіта»
у формі товариства з обмеженою відповідальністю.
01032, Київ, бульвар Т. Шевченка, 46,
тел. (044) 234-15-86, 234-95-23 (факс).
Свідоцтво ДК № 221 від 16.10.2000 р.

Підписано до друку з оригінального макета 20.06.2013. Формат 60x84 1/16. Гарнітура таймс. Спосіб друку – ризогр. Ум. друк. арк. 15,1 Обл.-вид. арк. 11,8 Наклад 250 пр. (перший завод 100 пр.)

Віддруковано НТУУ «КПІ» ВПІ ВПК «Політехніка»
Свідоцтво ДК 1665 від 28.01.2004 р.
03056, Київ вул політехнічна, 14 корп. 15 тел (044) 406-81-78
Зам. №