НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

Факультет прикладної математики

Кафедра прикладної математики

«На правах рукопису»
УДК 656.073

«До захисту допущено»

Завідувач кафедри
_____ О. Р. Чертов
(підпис)

«___» _____2015 р.

Магістерська дисертація

на здобуття ступеня магістра

зі спеціальності 8.04030101 «Прикладна математика»

на тему: Модель кластеризованої лінгвістичної бази знань для розпізнавання природномовного тексту

Виконала: студентка 2 курсу, групи КМ-31М

Кулик Ольга Олександрівна _____
(підпис)

Науковий керівник доцент, канд. техн. наук Сирота С. В. _____
(підпис)

Консультант із нормоконтролю старший викладач Мальчиков В. В. _____
(підпис)

Консультант зі спеціальних питань зав. відділом, канд. техн. наук Сажок М. М. _____
(підпис)

Рецензент декан ФПМ, д-р техн. наук, проф. Дичка І. А. _____
(підпис)

Засвідчую, що у цій магістерській дисертації немає запозичень з праць інших авторів без відповідних посилань.
Студентка _____
(підпис)

Київ – 2015

Національний технічний університет України
«Київський політехнічний інститут»

Факультет прикладної математики

Кафедра прикладної математики

Рівень вищої освіти – другий (магістерський)

Спеціальність 8.04030101 «Прикладна математика»

ЗАТВЕРДЖУЮ
Завідувач кафедри
_____ О. Р. Чертов
(підпис)
«___» _____2015 р.

ЗАВДАННЯ
на магістерську дисертацію студентці
Кулик Ользі Олександрівні

1. Тема дисертації: «Модель кластеризованої лінгвістичної бази знань для розпізнавання природномовного тексту»,
науковий керівник дисертації Сирота Сергій Вікторович, канд. техн. наук,
затверджені наказом по університету від «20» березня 2015 року № 785-С.

2. Термін подання студентом дисертації: «18» червня 2015 р.

3. Об'єкт дослідження: моделі розпізнавання мовлення, особливості розпізнавання української мови, лінгвістичні моделі у складі генеративної моделі, методи кластеризації, засоби розпізнавання мовлення, програмні продукти та системи побудови лінгвістичних баз.

4. Предмет дослідження: вдосконалена модель кластеризованої лінгвістичної бази знань зі застосуванням skipping n-gram для розпізнавання природномовного тексту українською мовою в процесі розпізнавання мовленнєвого сигналу.

5. Перелік завдань, які потрібно розробити:

– проаналізувати існуючі моделі та методи розпізнавання мовленнєвих сигналів;

– проаналізувати існуючі рішення побудови лінгвістичних баз знань для розпізнавання природномовного тексту на українській мові;

– розробити модель кластеризованої лінгвістичної бази знань зі застосуванням skipping n-gram для розпізнавання природномовного тексту;

– розробити програмне забезпечення для обробки та підготовки даних до кластеризації програмним інструментарієм clustlm;

– проаналізувати результати застосування skipping n-gram при моделюванні послідовності слів для розпізнавання природномовного тексту.

6. Орієнтовний перелік ілюстративного матеріалу:

– таблиця k-skip-3-gram покриття;
– графік покриття k-skip-n-grams;
– таблиця k-skip-3-gram покриття текстів з газет та машинного перекладу;
– діаграма потоків даних (DFD) лінгвістичної бази знань;
– діаграма станів (OSTN) лінгвістичної бази знань;
– UML діаграми послідовності обробки даних;
– таблиця результатів роботи модулів побудови unigrams та k-skip-n-grams;
– таблиця результатів роботи програмного інструментарію clustlm;
– блок-схема алгоритму побудови k-skip-n-grams.

7. Орієнтовний перелік публікацій:

– Міжнародна наукова молодіжна школа «Системы и средства искусственного интеллекта – ССИИ-2013»;

– XV міжнародна наукова конференція «Інтелектуальний аналіз інформації» ім. Т.А. Таран – IAI'2015»;

8. Консультанти розділів дисертації

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Кластеризована лінгвістична модель бази знань зі застосуванням skipping-bigrams Модулі з обробки та підготовки даних для кластеризації лінгвістичної моделі	Сажок М. М., зав. відділом		

9. Дата видачі завдання «25» жовтня 2013 р.

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів магістерської дисертації	Примітка
1	Узгодження тематики МД та напряму досліджень з керівником	15 вересня–30 жовтня 2013	
2	Ознайомлення з існуючими методами рішень поставлених завдань	30 жовтня 2013–15 лютого 2014	
3	Проведення порівняльного аналізу математичних методів побудови лінгвістичної моделі	15 лютого–1 вересня 2014	
4	Підготовка матеріалів першого та другого розділів МД	1 вересня–1 жовтня 2014	
5	Проведення дослідження зі застосування k-skipping-n-grams	1 жовтня–1 листопада 2014	
6	Розробка математичної моделі для кластеризованої лінгвістичної бази знань	1листопада–1 грудня 2014	
7	Підготовка матеріалів третього розділу МД	15 січня–1 лютого 2015	
8	Розробка програмного забезпечення з підготовки даних для кластеризації	1 лютого–1 березня 2015	
9	Переддипломна практика, робота над публікаціями	1 березня–1 квітня 2015	
10	Тестування програмного забезпечення	1 квітня–30 квітня 2015	
11	Завершення роботи над основною частиною МД	4 травня–1 червня 2015	
12	Попередній захист МД	1 червня–15 червня 2015	

Студентка _____ О. О. Кулик
(підпис)

Науковий керівник дисертації _____ С. В. Сирота
(підпис)

РЕФЕРАТ

Магістерська дисертація присвячена проектуванню та розробці кластеризованої лінгвістичної бази знань з використанням skipping n-grams. Лінгвістична модель, що розробляється, є складовою частиною роботи програм з розпізнавання мовленнєвих сигналів і має за мету поліпшення розпізнавання мовлення.

Об'єктом досліджень є моделі розпізнавання мовлення, особливості розпізнавання української мови, лінгвістичні моделі у складі генеративної моделі, методи кластеризації, засоби розпізнавання мовлення, програмні продукти та системи побудови лінгвістичних баз.

Предмет дослідження: вдосконалена модель кластеризованої лінгвістичної бази знань зі застосуванням skipping n-gram для розпізнавання природномовного тексту українською мовою.

В роботі розглянуто та проаналізовано існуючі методи розпізнавання мовлення, визначені переваги ПММ та проаналізовано варіанти побудови лінгвістичних моделей. Проведено ряд експерементів, які показали доцільність використання skipping n-grams для побудови кластеризованої лінгвістичної бази знань.

Результатом дипломної роботи є сукупність модулів, що реалізує математичну модель лінгвістичної бази знань. Розроблене ПЗ впроваджено у використання в МННЦ інформаційних технологій та систем НАН України та МОН України. Основні положення та результати роботи доповідалися на конференціях «ССИИ'2013» та «ІАІ'2015».

Робота складається з вступу, 5 розділів та висновків і налічує 75 сторінок. Містить 10 ілюстративних матеріалів, 4 таблиць, 4 додатки та посилається на 11 літературних джерел.

Ключові слова: приховані марківські моделі, розпізнавання мовленнєвого сигналу, розпізнавання природно-мовного тексту, кластеризація, лінгвістична база, k-skip-n-gram, bigram, unigram.

ABSTRACT

Thesis are dedicated to designing and developing cluster linguistic knowledge base using k-skipping-n-grams. The development of linguistic model is a part of speech signals recognition program and the main aims of it — is speech recognition improving.

Model of speech recognition, speciality of Ukrainian language recognition, linguistic models as a part of generative models, methods of clustering, speech recognition software and linguistic databases building system are the main objects of research.

Research subject: model of clustered linguistic knowledge base based on skipping n-grams for recognizing natural language text.

Most famous methods of speech recognition were considered and analyzed. Benefits of Hiden Markov Models were described and all variants of linguistic models were analyzed. A series of experiment showed the feasibility of using k-skipping-n-gram to build cluster linguistic knowledge base.

The result of the thesis is a set of modules that implements a mathematical model of linguistic knowledge base. These modules are implemented in International Research and Training Center for Information Technologies and Systems of the National Academy of Sciences of Ukraine and Ministry of Education and Science of Ukraine. The results were tested at such conferences «ССИИ'2013» and «IAI'2015».

The work consists of an introduction, 5 sections, includes conclusions and 75 pages. Contains 10 illustrative materials, 4 tables, 4 appendices and has 11 references.

Key words: hiden markov models, speech recognition signal, recognition of natural language text, clustering, linguistic basis, k-skip-n-gram, bigram, unigram.

Зміст

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

Кластеризація — групування, розбиття множини об'єктів на непересічні підмножини, кластери, що складаються зі схожих об'єктів.

Лінгвістична база даних — база слів, які ми отримуємо при розпізнаванні мовленнєвого сигналу після закінчення опрацювання звукового сигналу акустичною моделлю.

Computing minimal edition distance (CMED) — у теорії інформації і комп'ютерній лінгвістиці міра відмінності двох послідовностей символів (рядків).

N-grams — послідовність з N слів.

Skipping n-grams — послідовність n слів з тексту, які допускають виключення одного, або декількох слів з порядку їх слідування.

Інформаційна ентропія — міра невизначеності випадкової величини.

к-ть — кількість.

ПЗ — програмне забезпечення.

ПММ — приховані марківські моделі.

ВСТУП

З розвитком інформаційних технологій широкого застосування набувають системи розпізнавання злитого мовлення. Найчастіше вони застосовуються для введення голосових команд, пошуку аудіофайлів за їх змістом, запису озвученого тексту, ідентифікації диктора та інше. Технології розпізнавання мовлення дають змогу здійснювати взаємодію людини і комп'ютера найприроднішим для людини чином — голосом. Таким чином, будучи посередником взаємодії комп'ютера та людини, дані системи поступово витісняють інші види вводу інформації.

Центральним завданням в галузі розробки інформаційних технологій мовлення є створення систем автоматичного розпізнавання мови і систем синтезу мовлення широкого призначення.

Завдання розпізнавання мовлення складає автоматичне відновлення тексту вимовлених людиною фраз, слів, речень на природній мові. До важливих практичних завдань пов'язаних з процесом розпізнавання мовлення можна віднести розробку систем диктування текстів, систем мовленнєвого керування різним пристроями, систем мовленнєвого діалогу (наприклад по телефону). На данному етапі процес розпізнавання мовлення знаходиться у стадії безприривного росту: десятки крупних комерційних компаній (IBM, Dragon.Phillips, Microsoft) створюють та активно розвивають комерційні системи розпізнання мовлення. Експерти в області комп'ютерних технологій називають завдання розпізнання мовлення одним з найважливіших завдань XXI століття.

Завданя розпізнання мовлення першочергово полягало у відтворенні тексту з окремо вимовлених слів. Тільки у останні десять років комп'ютерна техніка досягла такого рівня, коли стало усвідомненим завдання розпіднавання спонтанного мовлення. На цьому етапі було з'ясовано, що для вирішення завдання розпізнання спонтанного мовлення недостатньо вміти розпізнавати тільки окремі звуки та слова. На практиці було показано, що людина

під час сприйняття мовлення для уникнення неоднозначності використовує знаня про природнє мовлення, а також зміст того що вимовляється.

Завдяки розпізнаванню мовлення вивільняються руки користувача при керуванні комп'ютерними системами, введенні текстової інформації, транскрибуванні (стенографуванні) фонограм тощо. Вже тепер починають з'являтися системи, що допомагають в оволодіванні розмовною іноземною мовою на основі технології розпізнавання мовлення. Велике майбутнє також за системами усного перекладу. В голосових інформаційно-довідкових системах (IVR) розпізнавання мовлення по телефонному каналу дає можливість здійснювати пошук інформації та замовляти різноманітні послуги. Голосові замки надзвичайно перспективні для захисту персональної інформації [1].

В наш час вище зазначені системи існують і повноцінно функціонують лише для англійської мови. Для слов'янських мов загальноприйняті алгоритми та методи розпізнавання злитого мовленнєвого сигналу не дають точного результату, в силу того, що в слов'янських мовах у 8–10 разів більше словоформ, ніж в англійській та достатньо вільний порядок слів у реченні. Таким чином сильно зростає робочий словник та зменшується точність прогнозування в лінгвістичній моделі.

Аналіз публікацій та відкритих для користування програм провідних компаній та наукових центрів з розпізнавання мовленнєвого сигналу показує, що найпоширенішого застосування для розпізнавання мовлення набула схема неявних (прихованих) марківських моделей. Тому взявши дану схему за основу будемо намагатись вдосконалити розпізнавання сигналу українською мовою.

1 ПОСТАНОВКА ЗАДАЧІ

В даній роботі ставиться задача розробки кластеризованої лінгвістичної бази знань для розпізнавання природно-мовного тексту, на основі skipping n-gram. Дана задача включає наступні завдання:

1. аналіз існуючих методів розпізнавання мовленнєвого сигналу;
2. аналіз існуючих методів побудови лінгвістичної моделі;
3. встановлення доцільності застосування skipping-n-grams для розпізнавання природно-мовного тексту;
4. розробка моделі кластеризованої лінгвістичної бази знань;
5. розробка алгоритму перетворення вхідних даних;
6. модифікація програмного модуля кластеризації clustlm;
7. проведення порівняльного аналізу та оцінка роботи ПЗ з вдосконаленою лінгвістичною базою.

Розроблене ПЗ повинно бути багатофункціональним: обробка та підготовка вхідних даних для кластеризації повинна здійснюватись незалежно від кластеризації. Це надасть можливість використовувати розроблені модулі для розв'язання й інших задач.

Функціонал програми не повинен обмежуват користувача — користувачу потрібно надати можливість налаштування параметрів запуску програми та керування процесом кластеризації.

Всі можливі параметри кластеризації в програмі кластеризації clustlm повинні працювати і для роботи з покращеною лінгвістичною моделлю:

- можливість задати наступні параметри: максимальна кількість ітерацій та кластерів;
- можливість збільшення к-ть кластерів починаючи з певної ітерації;
- можливість збільшення к-ть кластерів починаючи з певним інкрементом;
- запис кластерів до файлу;
- зчитування кластерів з файлу;

- зчитування кластерів з лог-файлу;

Користувачу надається можливість управління процесом роботи програми, а саме можливість зупинити кластеризацію в певний момент часу та продовжити з того ж місця в інший час, при цьому, за бажанням, змінивши параметри кластеризації.

2 ОГЛЯД ІСНУЮЧИХ РІШЕНЬ

2.1 Аналіз методів розпізнавання мовленнєвого сигналу.

В наш час існують й розробляються різні методи розпізнавання мовлення, та найширше застосовуються методи, що базуються на генеративній моделі — на прихованих марківських моделей та дискримінативній — нейронних мереж.

2.1.1 Застосування прихованих марківських моделей.

Прихованою марківською моделлю (ПММ) називається модель, що складається з M станів (N-gram модель), в кожному з яких деяка система може приймати одне з N значень якого-небудь параметру.

Ймовірність періодів між станами задається матрицею ймовірностей $A = \{a_{ij}\}$, де a_{ij} — ймовірність переходу з i-го в j-ий стан.

Ймовірність випадіння кожного з M значень параметрів в кожному з N станів задається вектором $B = \{b_j(k)\}$, де $b_j(k)$ — ймовірнісь випадіння k-го значення параметру в j-м стані.

Ймовірність настання початкового стану задається вектором $\pi = \pi_i$, де π_i — ймовірність того, що в початковий момент система опинеться в i-му стані.

Таким чином, прихованою марківською моделлю називают трійку

$\lambda = \{A,B,\pi\}$. Використання прихованих марковських моделей для розпізнавання мови базується на двох засадах:

а) Звуковий сигнал може бути розбитий на фрагменти, що відповідають станам в ПММ, параметри звукового сигналу в межах кожного фрагменту вважаються постійними.

б) Ймовірність кожного фрагменту залежить тільки від поточного

стану системи і не залежить від попередніх станів.

Модель називається «прихованою», так як нас, як правило, не цікавить конкретна послідовність станів, в якій перебуває система. Ми або подаємо на вхід системи послідовність типу $O = \{o_1, o_2, \ldots o_n\}$, де кожне o_i — значення параметру (одне з M), яке приймається в i-тий момент часу, а на виході очікуємо модель $\lambda = \{A, B, \pi\}$ з максимальною ймовірністю генеруючу таку послідовність, або ж навпаки — подаємо на вхід параметри моделі і генеруємо нею породжену послідовність. І в тому і в іншому випадку система виступає як «чорний ящик», в якому приховані поточні стани системи, а пов'язана знею модель заслуговує на назву «прихована».

Формування генеративної моделі починається з наступного: отриманий звуковий сигнал в результаті препроцесингу перетворюється в послідовність акустичних векторів фіксованого виміру $\boldsymbol{Y}_{i:T} = y_1, \ldots, y_T$. Далі декодер намагається встановити послідовність слів $\boldsymbol{w}_{i:L} = w_1, \ldots, w_L$, які найімовірніше утворили Y, тобто декодер шукає

$$\hat{\boldsymbol{w}} = \arg\max_{\boldsymbol{w}} \{P(\boldsymbol{w}|\boldsymbol{Y})\}. \tag{2.1}$$

Такі $\hat{\boldsymbol{w}}$ є основними складовими словника з розпізнавання злитого мовлення.

Насправді завдання змоделювати безпосередньо $P(\boldsymbol{w}|\boldsymbol{Y})$ є досить складним, тому застосувавши правило Байєса, отримаємо еквівалентну задачу пошуку

$$\hat{\boldsymbol{w}} = \arg\max_{\boldsymbol{w}} \{p(\boldsymbol{Y}|\boldsymbol{w})P(\boldsymbol{w})\}. \tag{2.2}$$

Акустична модель визначає міру схожості $p(\boldsymbol{Y}|\boldsymbol{w})$, а ймовірність $P(\boldsymbol{w})$ визначає лінгвістична модель. Ці дві моделі в сукупності формують генеративну модель розпізнавання мовленнєвого сигналу.

В акустичній моделі кожне слово w розкладається на послідовність K_w фонем, тобто послідовність $q_{1:K_w}^{(w)} = q_1, \ldots, q_{K_w}$ є фонемною транскрипцією слова. Зважаючи на те, що вимова може бути різною, міру схожості $p(\boldsymbol{Y}|\boldsymbol{w})$ можна обчислити наступним чином, врахувавши багато фонемних

транскрипцій:

$$p(\boldsymbol{Y}|\boldsymbol{w}) = \sum_{\boldsymbol{Q}} p(\boldsymbol{Y}|\boldsymbol{Q})P(\boldsymbol{Q}|\boldsymbol{w}), \tag{2.3}$$

де сума обчислюється по всім послідовностям вимов $\boldsymbol{w}$, а $\boldsymbol{Q}$ — послідовність вимов, для котрої виконується

$$P(\boldsymbol{Q}|\boldsymbol{w}) = \prod_{l=1}^{L} P(\boldsymbol{q}^{w_l}|w_l), \tag{2.4}$$

де кожна $\boldsymbol{q}^{(w_l)}$ — допустима вимова слова w_l [2].

Кожна фонема q представляється акустичною генеративною моделлю, як показано на рисунку 2.1. Вона має наступні параметри: ймовірності переходу зі стану в стан $\{a_{ij}\}$ та $\{b_j()\}$ — розподіли у просторі первинних ознак для робочих станів [2].

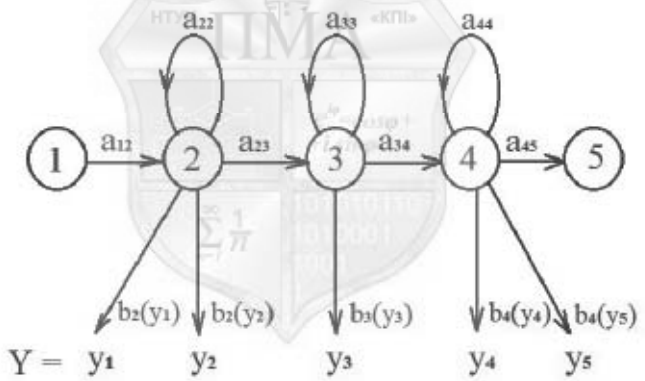

Рисунок 2.1 – Акустична генеративна модель

Ці розподіли фактично апроксимують у просторі первинних ознак ті області, через які проходять траєкторії, що відповідають акустичній реалізації фонеми q. Такий загальний вигляд має базова ПММ. Технічно, перехід від робочого стану генеративної моделі до одного зі станів, з яким робочий стан пов'язаний, здійснюється за одиницю відліку часу, а матриця $\{a_{ij}\}$ залежить від топології ПММ та має вигляд стохастичної матриці, що формує ланцюг Маркова [3].

2.1.2 Використання нейронних мереж.

Цей підхід моделює процес розпізнавання в біологічних системах. Нейронна мережа — апаратні чи програмні засоби, що моделюють роботу мозку людини. Як і будь-яка модель, вони є лише наближенням. Але навіть не дивлячись на те, що в подібних засобах імітуються лише окремі сторони бібліотечного прототипу, вони вже зараз дають змогу досягти певних успіхів в багатьох областях, особливо в пов'язаних з класифікацією і розпізнанням образів.

Як відомо, нервова система людини складається з великої кількості елементів — нейронів, які поєднуються між собою ниткоподібними відростками — дендритами. Збудження чи заторможення передається від нейрона до нейрона по дендритам, де ті приймають сигнали в точках з'єднання, так званих синапсами. Прийняті синапсом сигнали передаються нейрону, де суммуються. Якщо рівень збудження перевищує деяку порогову величину, збудження передається з тіла нейрону у вихідну точку, яка зветься аксоном, звідки по дендритам поступає в інші нейрони.

Саме наведені вище характеристики і стали суттєвими при створенні штучних нейронних мереж.

Основу нейронної мережі складає, як правило, однотипні елементи, імітуючі роботу біологічного нейрону, які так само й називаються. Кожен з нейронів в кожен момент часу знаходиться, як і біологічний нейрон, в деякому стані. Він має групу однонаправлених вхідних зв'язків — синапсів, які проходять від входу в мережу чи від інших нейронів. Крім того він має один однонапрямлений зв'язок — аксон.

Синаптичні зв'язки характеризуються вагою w_i. Поточний стан S нейрона дорівнює зважуваній сумі входів:

$$S = \sum_{i=1}^{n} X_i w_i \tag{2.5}$$

В векторному вигляді це можна записати як $S = XW$, тобто вектор S

є добутком вектора вхідних значень X і матриці вагів W , в якій рядки відповідають шарам, а стовпці — нейронав в середині кожного шару.

Функція S далі перетворюється активаційною функцією F і дає вихідний сигнал Y нейрона.

$Y = F(S)$ — активаційна функція мусить мати таку властивість, як різкий зріст на короткому інтервалі аргументу в околі крайнього значення T, приймати приблизно одне значення до цього інтервалу і приблизно одне (велике) значення — після цього інтервалу. Цим вимогам відповідає, наприклад, функція Y, що дорівнюю 1 при $S > T$, і 0 якщо $S \leqslant T$. Ця функція також називається функцією одиничкого стрибка [4].

2.2 Побудова лінгвістичної моделі

Отриманий звуковий сигнал в результаті препроцесингу перетворюється в послідовність акустичних векторів фіксованого виміру

$$\boldsymbol{Y}_{i:T} = y_1, \ldots, y_T$$

Далі декодер намагається встановити послідовність слів $\boldsymbol{w}_{i:L} = w_1, \ldots, w_L$, які найімовірніше утворили Y, тобто декодер шукає

$$\hat{\boldsymbol{w}} = \arg\max_{\boldsymbol{w}}\{P(\boldsymbol{w}|\boldsymbol{Y})\}. \tag{2.6}$$

Такі $\hat{\boldsymbol{w}}$ є основними складовими словника з розпізнавання злитого мовлення.

Насправді завдання змоделювати безпосередньо $P(\boldsymbol{w}|\boldsymbol{Y})$ є досить складним, тому застосувавши правило Байєса, отримаємо еквівалентну задачу пошуку

$$\hat{\boldsymbol{w}} = \arg\max_{\boldsymbol{w}}\{p(\boldsymbol{Y}|\boldsymbol{w})P(\boldsymbol{w})\}. \tag{2.7}$$

В лінгвістичній моделі ймовірність послідовності слів $\boldsymbol{w} = w_1, \ldots, w_K$,

що зазначалась в (2.7), визначається як

$$P(\boldsymbol{w}) = \prod_{k=1}^{K} P(w_k|w_{k-1}, \ldots, w_1). \tag{2.8}$$

У великих лінгвістичних базах розпізнавання кількість попередніх слів у (2.6) може бути досить великою, тому її скорочують до $(N-1)$ для можливості подальших обчислень і формування N-грамної лінгвістичної моделі:

$$P(\boldsymbol{w}) = \prod_{k=1}^{K} P(w_k|w_{k-1}, w_{k-2}, \ldots, w_{k-N+1}), \tag{2.9}$$

де N лежить в межах 2–4. Ймовірності N-грам оцінюються за текстовим корпусом, підраховуючи кількість входжень N-грам [2].

Якщо задати частоту N-gram $(w_{k-N+1}, \ldots, w_{k-1}, w_k)$ як:

$$C(w_{k-N+1}, \ldots, w_{k-1}, w_k), \tag{2.10}$$

то

$$P(w_k|w_{k-1}, w_{k-2}, \ldots, w_{k-N+1}) \approx \frac{C(w_{k-N+1}, \ldots, w_{k-1}, w_k)}{C(w_{k-N+1}, \ldots, w_{k-1})} \tag{2.11}$$

Застосувавши функцію класифікації $G(w) = g$, де g належить множині класів слів, для bigrams ймовірності в (2.9) можуть бути представлені, як:

$$P_{\text{class'}}(w_k|w_{k-1}) = P(w_k|G(w_k), G(w_{k-1}), w_{k-1}) P(G(w_k)|G(w_{k-1}, w_{k-1})) \tag{2.12}$$

Припустимо, що $P(w_k|G(w_k), G(w_{k-1}), w_{k-1})$ не залежить від $G(w_{k-1})$ і w_{k-1}, а $P(G(w_k)|G(w_{k-1})$ не залежить від w_{k-1}. Тоді маємо наступну модель:

$$P_{\text{class'}}(w_k|w_{k-1}) = P(w_k|G(w_k)) P(G(w_k)|G(w_{k-1})) \tag{2.13}$$

2.3 Побудова кластиризованої лінгвістичної моделі

Нехай W — **послідовність всіх реалізацій слів** $(w_1, w_2, w_3, \dots)$ з вибірки на основі деякого текстового корпусу, V — **словник, або множина слів з** W. Тоді для біграмного контексту найкраще об'єднання слів в класи призводить до максимума

$$P_{\text{class}}W = \prod_{x,y \in V} P_{\text{class}}(x|y)^{C(x,y)}, \tag{2.14}$$

де (x,y) означає пару слів, в котрій слово x слідує за словом y в послідовності W, а функція $C(\cdot)$ визначає частоту спостереження аргументу у вибірці [10]. Для уникнення проблем з малими величинами використовується логарифмування:

$$\log P_{\text{class}}W = \sum_{x,y \in V} C(x,y) \log P_{\text{class}}(x|y) \tag{2.15}$$

Після перетворень над $P_{\text{class}}(x|y)$ маємо:

$$F_G = \sum_{g,h \in G} C(g,h) \log C(g,h) - 2 \sum_{g \in G} C(g) \log C(g), \tag{2.16}$$

де (g,h) означає слідування класу g за класом h.

Ідея пошуку деякого найкращого об'єднання слів в задане число класів полягає в обчисленні зміни критерія F_G при гіпотетичному відношенні кожного зі слів у альтернативні класи з наступним переміщенням до класу з найбільшим критерієм. Таким чином, класи обмінюються словами, тобто здійснюють певний словообмін до тих пір, доки критерій не перестає покращуватись.

Даний алгоритм відноситься до сімейства «жадібних» і не гарантує глобального екстремума [5].

2.4 Побудови лінгвістичної моделі з урахуванням фонетичної близькості слів

Пропонується розвивати підхід попереднього об'єднання слів у кластери з подальшим обчисленням параметрів статистичної лінгвістичної моделі заданого порядку. Опрацьована для української мови, процедура формування такої моделі передбачає однозначне віднесення слова до одного або іншого кластеру за критерієм, що мінімізує ентропію [6]. Аналіз сформованих таким чином кластерів показав значною мірою однорідність розподілу слів за деякими комбінаціями граматичних, семантичних і фонетичних ознак. Разом з тим, однозначність при розподілі слів на кластери суперечить поширеному явищу омонімії (омографії). З використанням орфоепічної бази української мови [8] було проведено дослідження цього явища [9].

У довільному тексті в середньому спостерігається понад 10% слів, що пишуться однаково, але мають різне граматичне та/або семантичне значення. При однозначному відображенні такі слова потрапляють до кластерів, що відповідають більш поширеному значенню. Найбільш частотні омографи формують окремий кластер.

Пропонується спосіб багатозначного розподілу на кластери шляхом введення додаткового підсловника, що містить слова, які відповідають різним граматичним і семантичним значенням слів із базового словника системи розпізнавання. Тоді при формуванні гіпотез відповіді розпізнавання в декодері буде оцінюватись перспективність однієї й тієї самої фонетичної транскрипції слова в контексті приналежності одному або більше кластерам слів.

Формування додаткового підсловника полягає у визначенні слів, які потенційно належать до різних кластерів, тобто міри належності або відстані такого слова до двох або більше класів не повинні суттєво відрізнятися.

Спочатку відбувається перетворення з фонемного тексту на орфографічний за допомогою алгоритму багатозначного перетворення послідовностей символів.

2.4.1 Модель багатозначного перетворення послідовностей символів

Нехай маємо скінченну послідовність символів

$$(a_1, a_2, \ldots, a_n, \ldots, a_N) \equiv a_1^N, a_n \in \mathbf{A}, \tag{2.17}$$

де **A** — алфавіт вхідних символів. Сконструюємо відображення цієї послідовності на множину послідовностей вихідних символів із деякого іншого алфавіту **B**.

Розглянемо функцію f, що відображає послідовність a_1^N, починаючи з її n-го символу, у символ алфавіту **B** або порожню множину:

$$f : a_n^N \to b, b \in \mathbf{B} \cup \varnothing, 1 \leqslant n \leqslant N \tag{2.18}$$

Зауважимо, що (2.18) має місце лише у випадку, коли вхідна послідовність належить області визначення f, тобто $a_n^N \in \mathrm{Def}(f)$. Множина послідовних застосувань таких функцій переводить a_n^N у послідовності символів з алфавіту **B**, утворюючи таким чином мультифункцію:

$$F\left(a_n^N\right) = \{\left(f_1^k(a_n^N)\right), \left(f_2^k(a_n^N)\right), \cdots, \left(f_{L_k}^k(a_n^N)\right) \in \mathbf{B}^{L_k} \cup \varnothing, 1 \leqslant k \leqslant K_F\}, \tag{2.19}$$

де L_k — довжина k-ї вихідної послідовності, загальна кількість яких, K_F, своя для кожного $F \in \mathbf{F}$

Визначимо аналог прямого добутку над множинами, отриманими внаслідок дії мультифункцій з **F**, як перебір усіх варіантів об'єднання скінченних послідовностей символів з алфавіту **B**. Тобто, опускаючи аргументи мультифункцій:

$$F \otimes G = \{\left(f_1^u, f_2^u, \ldots, f_{L_k}^u, g_1^v, g_2^v, \ldots, g_{L_k}^v\right), 1 \leqslant u \leqslant K_F, 1 \leqslant v \leqslant K_G\} \tag{2.20}$$

Припускаємо за визначенням, що якщо результат дії F або G є порожньою множиною, то результатом їх добутку буде порожня множина. На відміну від декартового добутку для визначеного нами аналогу виконується властивість асоціативності.

Розглянемо впорядковану множину $\tilde{F}$ мультифункцій $F \in \mathbf{F}$, які супроводимо додатковими параметрами:

$$\tilde{F} = \left(F_{i,d_i,\delta_i}\right), 1 \leqslant i \leqslant |\tilde{F}|, d_i > 0, \delta_i = \{0,1\}, \tag{2.21}$$

де i є індексом мультифуннкцій у впорядкованій множині $\tilde{F}$; параметр d_i назвемо шириною кроку аналізу, δ_i — **«умовою виключності»**. **Через ці параметри** конструюємо обмеження при обчисленні добутку:

$$\otimes_{i,n} F_{i,d_i,\delta_i}\left(a_n^N\right), 1 \leqslant i \leqslant |\tilde{F}|, 1 \leqslant n \leqslant N \tag{2.22}$$

Припустимо, що ми вже обчислили вираз (2.22) на деяких упорядкованих індексних множинах J і M і отримали деяку непорожню множину:

$$G_{J,M} = \bigotimes_{\substack{u \in J \\ v \in M}} F_{u,d_u,\delta_u}\left(a_v^N\right). \tag{2.23}$$

Нехай j та m є останніми елементами індексних множин J і M відповідно. Тоді при розгляді наступної компоненти добутку, $F_{i,d_i,\delta_i}\left(a_n^N\right)$, проводимо обчислення згідно з визначенням (2.20), якщо виконуються такі умови:

$$\begin{cases} m + d_i = n; \\ \left[\begin{array}{l} \delta_r \neq 1, 1 \leqslant r < 1; \\ \bigotimes\limits_{\substack{u \in J \\ v \in M}} F_{u,d_u,\delta_u}\left(a_v^N\right) \otimes F_{r,d_r,\delta_r}\left(a_v^N\right) \neq \varnothing, 1 \leqslant r < i, \text{якщо } \delta_i = 1 \end{array}\right. \end{cases} \tag{2.24}$$

В іншому випадку, при надходженні наступної компоненти добутку отримуємо порожню множину.

Виразом (2.22) породжуються послідовності вихідних символів за деякою послідовністю вхідних символів. Якщо вхідний алфавіт збігається з алфавітом літер певної мови, а вихідний алфавіт складається з фонем, то маємо багатозначний транскриптор орфографічного тексту. І навпаки, якщо **на вході — фонемний алфавіт, а на виході — алфавіт літер, то отримаємо** багатозначне перетворення з фонемного тексту на орфографічний [7].

Далі визначаємо міру схожості фонем за критерієм CMED. Для пофонемного порівняння двох рядків символів можна застосувати наступний метод: будуємо граф порівняння символів (рис. 2.2) з накладанням «штрафів» за розміром яких і визначаємо CMED.

«Штрафи» будемо накладати згідно наступних критеріїв:

- ступінь огубленості (є, немає);
- місце творення (передня, середня, задня частини ротової порожнини);
- назалізованість;
- палаталізація (пом'якшення);
- вокалізація або наявність «голосу» при творенні.

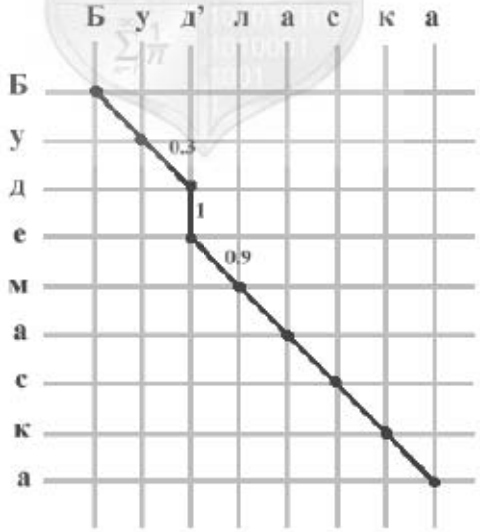

Рисунок 2.2 – Граф порівняння двох фонетичних послідовностей з накладанн «штрафів»

Розглянемо випадок двозначності. Проводиться аналіз розподілу на пло-

щині, де вісі абсцис відповідають мірі належності до кластеру, куди слово було однозначного віднесено, а по вісі ординат відкладаються міри належності до першого найближчого кластеру. Значення розподілу будуть знаходитись у першому квадранті нижче бісектриси (рис. 2.3).

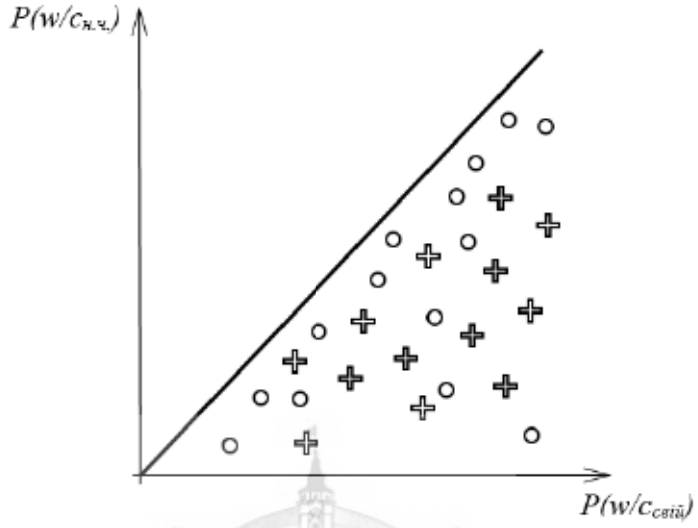

Рисунок 2.3 – Розподіл словоформ на площині

Вочевидь, слова, що належать до різних кластерів, перебувають в області, яка прилягає до бісектриси (рис. 2.4). Цю область слід апроксимувати таким чином, щоб вона містила якомога більшу частку омографів. Аналогічно проводиться оцінювання потенційної належності слова трьом і більше кластерам.

За допомогою запропонованого алгоритму при формуванні кластерів моделюється явище омографії, що дасть змогу більш точно оцінювати параметри лінгвістичної складової моделі розпізнавання мовленнєвого сигналу. Введення додаткового підсловника не збільшує кількість фонемних транскрипцій, а тому не додає навантаження на акустичну складову системи розпізнавання. Описаний спосіб також дає змогу обмежити зростання кількості кластерів.

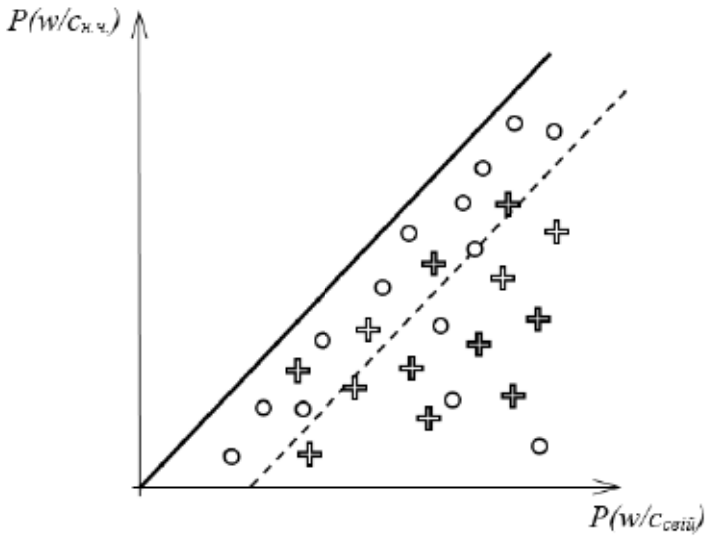

Рисунок 2.4 – Апроксимація області слів з різних кластерів

2.5 Висновок

Як було з'ясовано, більшість алгоритмів нейронних мереж погано працюють або взагалі не працюють з лінійними функціями. На відміну від нейронних мереж, математична структура скритих марковських моделей дуже велика і дозволяє вирішувати математичні проблеми різних областей науки. Правильно спроектована марковська модель дає гарні результати роботи. Перевага моделі заключається в тому, що розмітка і кожний елемент характеризується своїми марковськими випадковими полями, не залежними один від іншого. Це дозволяє легко модифікувати розпізнавальну систему.

При використанні звичайної лінгвістичної моделі для оцінки ймовірності слова w_i за словом w_{i-1} нам потрібно зберігати K^2 параметрів таких відношень (див. рисунок 2.5).

В той час, як при побудові кластеризованої лінгвістичної моделі потрібно зберігати $J^2 + K$ параметрів, де J — кількість класів (див. рисунок 2.6).

Таким чином застосування кластеризації мінімізує витрати ресурсів при розпізнаванні природно-мовного тексту.

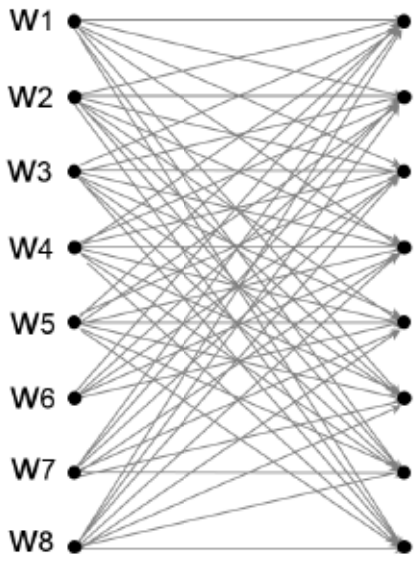

Рисунок 2.5 – Всі можливі пари слів (w_i, w_{i-1})

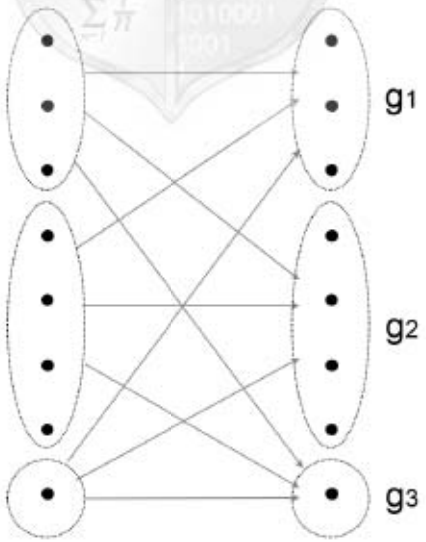

Рисунок 2.6 – Всі можливі пари кластерів (g_i, g_{i-1})

3 МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ

При побудові лінгвістичної моделі пропонується використовувати кластеризацію з врахуванням фонетичної близькості слів [9]. Це дозволить уникнути занесення до одного кластера омонімів та омографів, а також слів, що схожі за звучанням.

3.1 Обробка текстового корпусу для кластеризації

Перед початком роботи з текстовим корпусом, з нього вилучаються всі знаки, окрім дефіса та апострофа, всі цифри переводяться в текст відповідно до їх значення.

Також для кластеризації в текстовому корпусі виділяємо початок речення — <S>, кінець речення — </S>, та слова, що відсутні у словнику — <unk>.

$$\{w : c(w) < d\} = unk \tag{3.1}$$

Речення, які містять певний відсоток слів, відсутніх у робочому словнику, вилучаються:

$$S = \{S : |unk(S)| < d_1 \cdot |S|\} \tag{3.2}$$

$$unk(S) = \{w \in S : w \in unk\}, \tag{3.3}$$

де $|S|$ — к-ть слів у реченні, $|unk(S)|$ — к-ть невідомих слів у реченні.

3.2 Застосування k-skip-n-grams

Skipping n-grams, що допускають k та менше пропусків слів — це k-skip-n-grams.

Для речення $w_1 \dots w_m$ k-skip-n-grams задається, як наступна множина:

$$\left\{ w_{i_1}, w_{i_2} \dots w_{i_n} | \sum_{j=1}^{n} i_j - i_{j-1} < k \right\} \tag{3.4}$$

Так як оцінити лінгвістичну модель досить складно, тому було прийнято рішення провести ряд експериментів зі застосування k-skip-n-grams при розпізнаванні природно-мовного тексту [11].

Для того, щоб встановити доцільність використання k-skip-n-grams для статистичного моделювання послідовності слів ми обраховуємо збільшення skipping n-grams покриття тестових даних. В якості тестових даних було взято тексти з новин (газет) та переклад новин з іноземних мов автоматичною системою перекладу (такий переклад не є точним і дає лише множину слів тієї ж тематики).

Таким чином, підраховуємо всі можливі skipping n-grams в текстовому корпусі та оцінюємо, як багато n-grams вони охоплюють з тестових даних (табл. 3.1).

Таблиця 3.1 – k-skip-3-gram покриття

Skip	К-ть grams	Непокриті 3-grams	Унікальні непокриті 3-grams	Покриття
0	89	50133	38704	68%
1	271	40206	27536	72.12%
2	512	39645	27178	74.56%
3	867	37128	18543	75.32%

Як видно з рисисунка 3.1, зі збільшенням k (пропусків) в skipping n-grams текстового корпусу, покриття зростає: skipping n-grams покривають більше тестових даних, ніж звичайні n-grams ($k = 0$).

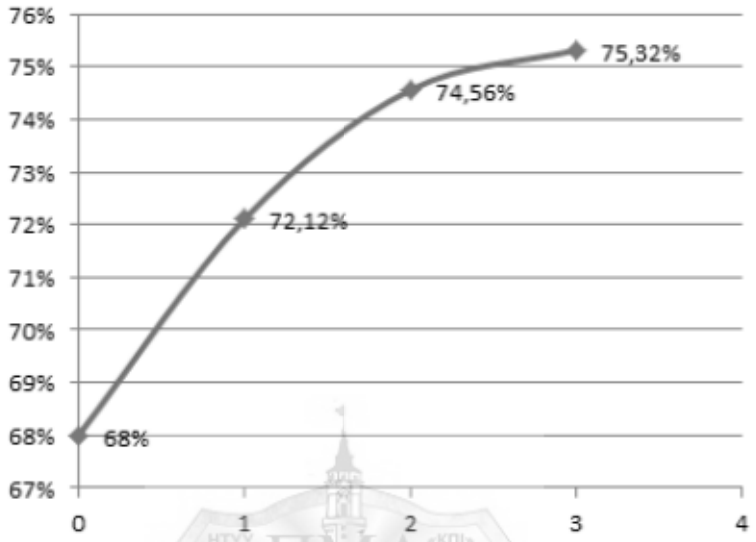

Рисунок 3.1 – Покриття k-skip-n-grams

Також було проведені експеременти показали, що збільшення пропусків слів в skiping n-gram не призводить до значного збільшення покриття тексту, який не відповідає тематиці корпусу (табл. 3.2).

Таблиця 3.2 – k-skip-3-gram покриття текстів новин з газет та текстів машинного перекладу

Середнє значення покриття	0-skip	2-skip	3-skip
новин з газети	64.15%	70.89%	73.12%
текстів перекладу	36.25%	38.13%	39.92%

3.3 Побудова кластеризованої лінгвістичної моделі зі застосуванням 1-skip-bigrams

Ми розглядатимемо k-skip-bigrams з одним і менше пропуском, тобто $k = 1$. Таким чином 1-skip-bigram містить 1 пропуск слів та 0 пропусків — це звичайні bigrams, утворені суміжними словами.

Для подальшої обробки вхідних даних нам потрібно виділити у текстовому корпусі unigrams, bigrams та 1-skip-bigrams з рівно одним пропуском слова.

Визначимо множину unigrams наступним чином:

$$G_1^0 = \{w | w \in W\} \tag{3.5}$$

Тоді множини bigrams та 1-skip-bigrams задаємо відповідно як:

$$G_2^0 = \{(w_i, w_{i+1}) | i \in \mathbb{N}, w_i \in W\} \tag{3.6}$$

$$G_2^1 = \{(w_i, w_{i+2}) | i \in \mathbb{N}, w_i \in W\} \tag{3.7}$$

Визначимо функцію $\text{count}(w_1, w_2, \ldots, w_n)$, як частоту входження послідовності $\{w_1, w_2, \ldots, w_n\}$ в W, таким чином будемо обчислювати частоту спостереження аргументу у виборці.

Для кластеризації використовуватимемо об'єднання множин unigrams, bigrams та 1-skip-ngrams, задаючи їм ваговий коефіцієнт α:

$$G = \alpha_1 G_1 \cup \alpha_2 G_2 \cup \alpha_{1,2} G_2^1 \tag{3.8}$$

$$\alpha_1 + \sum_{k,n} \alpha_{k,n} = 1 \tag{3.9}$$

Таким чином для skipping bigrams найкраще об'єднання слів в класи

призводить до максимума:

$$P_{\text{class}}W = \prod_{x,y \in G} P_{class}(x|y)^{C(x,y)} \tag{3.10}$$

$$C(x,y) = \alpha_2 \text{count}(x,y) + \alpha_{1,2} \sum_{w \in W} \text{count}(x,w,y) \tag{3.11}$$

Для уникнення проблем з малими величинами застосовуємо логарифмування до (3.10):

$$\log P_{\text{class}}W = \sum_{x,y \in G_2^0 \cup G_2^1} C(x,y) \log P_{\text{class}}(x|y) \tag{3.12}$$

$$\begin{aligned}
\log P_{\text{class}}W &= \sum_{x,y \in G} C(x,y) \log \left(\frac{C(x)}{C(G(x))} \times \frac{C(G(x),G(y))}{C(G(y))} \right) = \\
&= \sum_{x,y \in G} C(x,y) \log \left(\frac{C(x)}{C(G(x))} + \sum_{x,y \in G} C(x,y) \log \left(\frac{C(G(x),G(y))}{C(G(y))} \right) \right) = \\
&= \sum_{x \in G} C(x) \log \left(\frac{C(x)}{C(G(x))} + \sum_{g,h \in H} C(g,h) \log \left(\frac{C(g,h)}{C(h)} \right) \right) = \\
&= \sum_{x \in G} C(x) \log C(x) - \sum_{x \in G} C(x) \log C(G(x)) + \\
&\quad + \sum_{g,h \in H} C(g,h) \log C(g,h) - \sum_{g \in H} C(g) \log C(g) = \\
&= \sum_{x \in G} C(x) \log C(x) + \sum_{g,h \in H} C(g,h) \log C(g,h) - 2 \sum_{g \in H} C(g) \log C(g)
\end{aligned} \tag{3.13}$$

Застосувавши (2.11) та (2.13) до $P_{\text{class}}(x|y)$, та відкинувши компоненти, що не залежать від функції класифікації [12], маємо критерій оцінки

належності слова кластеру:

$$F_G = \sum_{g,h \in H} C(g,h) \log C(g,h) - 2 \sum_{g \in H} C(g) \log C(g) \tag{3.14}$$

Таким чином ми шукаємо найкраще об'єднання слів в задану кількість класів. Для цього обчислюємо критерій F_G для гіпотетичного занесення кожного слова до альтернаивних класів, а далі переміщаємо слова до класів з більшим критерієм, допоки критерій F_G не перестає покращуватись.

3.4 Висновок

Проведені досліди зі застосування skipping-n-grams при побудові кластеризованої лінгвістичної бази знань [9] показали, що застосування skip-grams є доцільним: було показано, що використання skip-grams є ефективнішим, ніж збільшення розмірів текстового корпусу майже в 4 рази.

Хоча при застосуванні skipping n-grams можуть утворюватись неіснуючі n-grams, вони як правило не впливають на покриття текстів різних тематик. Недоліком skip-grams моделі може бути великий розмір моделі, що призведе до більших витрат часу. Тому skipping n-grams модель краще використовувати, коли навчальний корпус не може бути збільшений, або немає ресурсів для його розширення.

Було з'ясовано, що тестові дані різної тематики, загалом матимуть менше суміжних n-grams. Таким чином ми можемо виділяти тексти, що схожі по контексту до текстового корпусу, а збільшення пропусків слів в skipping n-gram не призводить до значного збільшення покриття тексту, який не відповідає тематиці корпусу.

4 ПРОЕКТУВАННЯ ПРОГРАМНИХ ЗАСОБІВ

Лінгвістична база знань для розпізнавання природно-мовного сигналу представляє собою базу даних та набір модулів, що задають правила обробки цих даних. Дана база знань розроблена для певної предметної області, а тому є специфічною. На рисунку 4.1 зображено діаграма потоків даних в лінгвістичній базі знань, яка чітко показує процеси побудови нових знань.

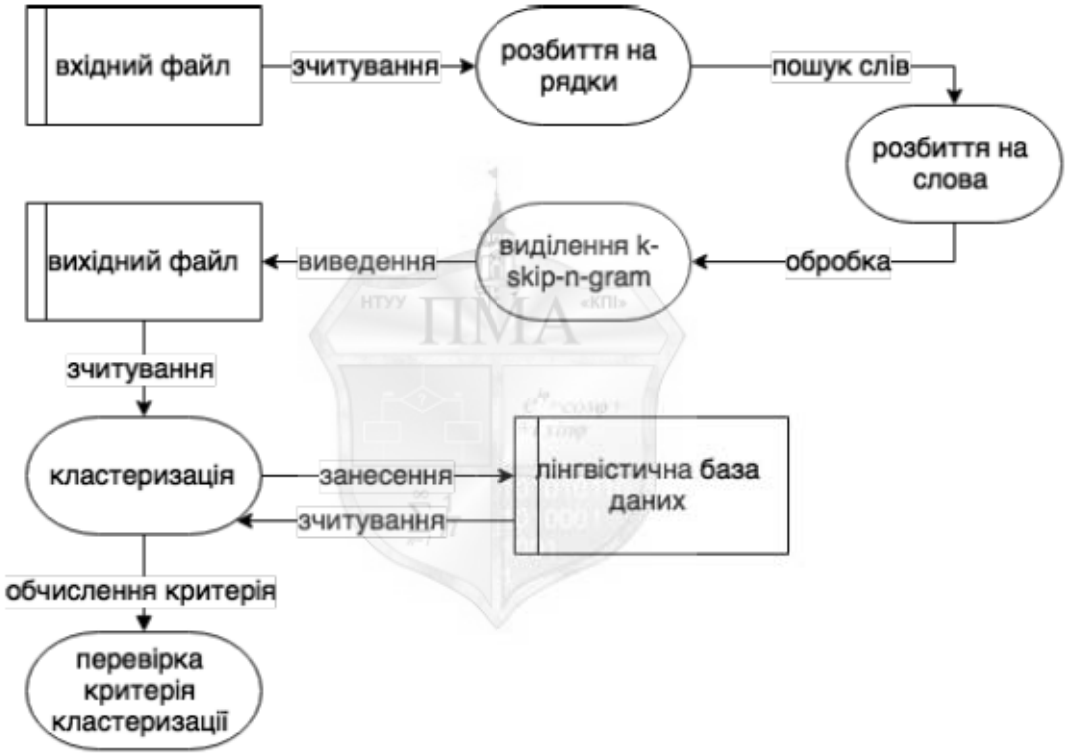

Рисунок 4.1 – DFD-діаграма лінгвістичної бази знань

ПЗ для побудови вхідних даних для кластеризації було реалізоване у вигляді сукупності незалежних модулей. Багатомодульна архітектура дозволяє використовувати модулі окремо іншими ПЗ для отримання результатів, що не були передбачені заздалегідь.

4.1 Опис розроблених програмових засобів

Розроблене програмне забезпечення складається з трьох модулів: модуль побудови unigrams, модуль побудови k-skip-bigrams та модуль підготовки даних для clustlm.

Модуль побудови unigrams аналізує вхідний файл: для кожного речення визначає початок та кінець, як <S> та </S> відповідно, розділяє речення на unigrams та підраховує їх кількість.

Модуль побудови k-skip-bigrams також аналізує файл, визначаючи початок <S> та кінець </S> речень, і розділяє кожне речення на bigrams з відровідним пропуском (skipping), заданим параметрично.

Модуль підготовки даних для clustlm представляє собою скриптовий файл, що виконує послідовність наступних симпонент ОС Linux:

- sort — сортування файлів значних об'ємів;
- uniq — підрахунок кількості однакових gram;
- awk — текстова обробка файлу для подальшого запуску кластеризації цього файлу за допомогою clustlm.

Результати взаємодії модулів відображено на діаграмі станів лінгвістичної бази знань (рис. 4.2).

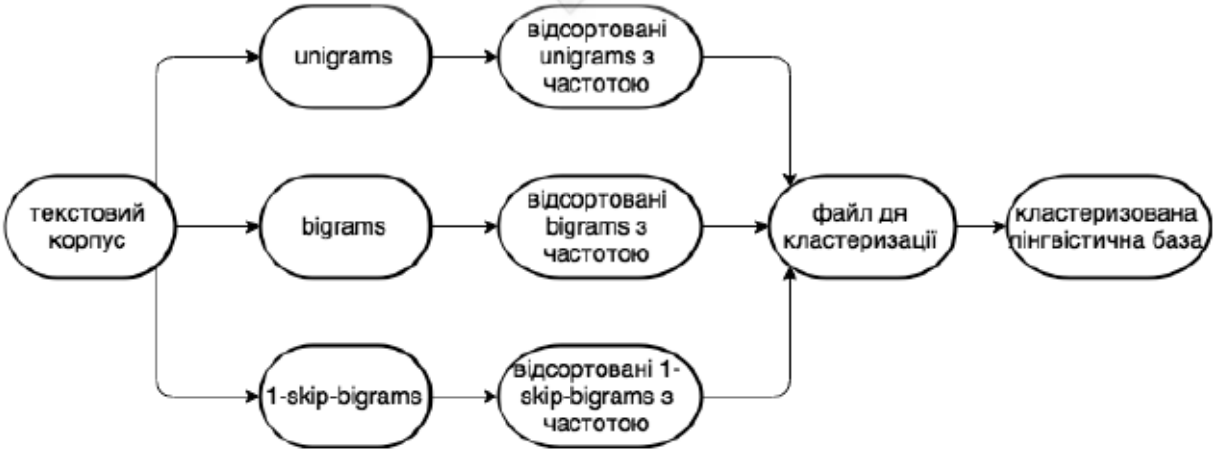

Рисунок 4.2 – OSTN-діаграма лінгвістичної бази знань

4.2 Архітектура розроблених програмових засобів

Для реалізації багатопотоковості у модулях побудови unigrams та k-skip-bigrams використано три види потоків: reader, worker, writer, та дві черги — input, output, які забезпечують синхронізацію між потоками. Взаємодію потоків показано на UML-діаграмі послідовності (рис. 4.3).

Потік reader зчитує файл по реченню та записує речення до вхідної черги input.

Потік worker дістає з вхідної черги речення, будує unigrams при роботі модуля побудови unigrams та надсилає їх у вихідну чергу output. При роботі модуля побудови k-skip-bigrams потік worker параметризується к-тю пропусків (skipping). Кількість таких потоків задається, по замовчуванні — 8 потоків.

Потік writer — потік виводу на екран даних з вихідної черги.

Алгоритм побудови k-skip-bigrams представлено блок-схемою (див. додаток А).

4.3 Засоби керування програмою

Запуск модулів на виконання відбувається з командного рядка.

Для запуску модуля побудови unigrams в командному дярку вказується:

```
do_unigrams n < input.txt > output.txt,
```

де n — к-ть потоків, input.txt — файл вхідних даних, output.txt — файл вихідних даних.

Для запуску модуля побудови k-skipp-bigrams в командному дярку вказується:

```
do_bigrams k n < input.txt > output.txt,
```

де k — кількість пропусків (skipping), n — к-ть потоків, input.txt — файл вхідних даних, output.txt — файл вихідних даних.

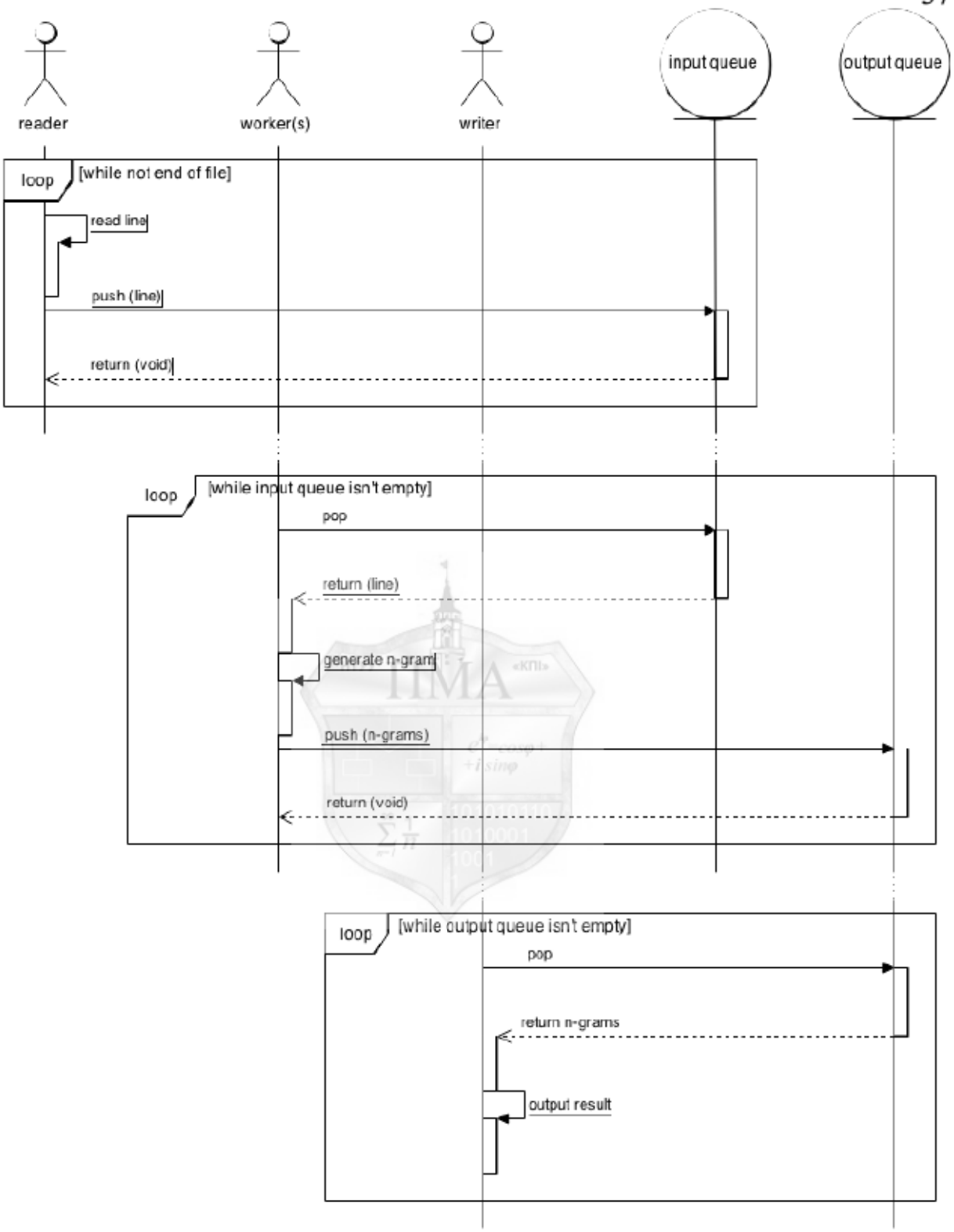

Рисунок 4.3 – UML-діаграма послідовності

Для запуску модуля підготовки даних для кластеризації командному дярку вказується:

```
bash prepare.sh
```

Попередньо потрібно вказати наступні параметри у файлі `prepare.sh`:

- do_unigrams — шлях до модуля побудови unigrams;
- do_bigrams — шлях до модуля побудови k-skip-bigrams;
- input — шлях до вхідного файлу;
- coefficients — масив вагових коефіцієнтів;
- result — ім'я вихідного файлу;
- thread_count — к-ть потоків.

Для запуску програми кластеризації clustlm в командному рядку вказуються наступні параметри:

- -c <str> - file of uni- and bi- gram counters
- -mi int - max iterations
- -mc int - max clusters
- -vc int - vary (grow) classes each iteration starting from int
- -vi int - grow classes each iteration with int increment
- -va float - grow classes each iteration with float accelleration
- -vm float - multiply classes each iteration with float
- -ic int - set initial iteration class count to int
- -rg <str> - read classes from file <str>
- -wg <str> - write classes to file <str>
- -wi <str> - after iterations write classes to file prepended with <str>
- -rl <str> - read classes from a log file

4.4 Засоби розробки ПЗ та вимоги до технічних засобів

Модулі побудови unigrams та k-skip-bigrams розроблено за допомогою Qt-creator — крос-платформений інструментарій розробки ПЗ на мові програмування C++ з використанням бібліотеки boost_thread для забезпечення багатопотоковості в роботі модулів.

Модуль підготовки даних для clustlm — скрипт на мові проограмування

bash, який використовує системні команди Linux — sort, uniq, awk.

Вимоги до персонального комп'ютера, на якому буде використовуватись розроблене програмне забезпечення:

- ОС GNU/Linux;
- Intel Core2Duo 2ГГц або вище;
- 3 Гб ОЗУ;
- 32 Гб вільного місця на диску;
- монітор, клавіатура.

4.5 Висновок

Для побудови кластеризованої лінгвістичної бази знань було модифіковано алгоритм кластеризації програми clustlm: використання 1-skip-bigram змінило обрахунки частот bigram, розроблено та реалізовано три модуля з обробки та підготовки вхідних даних. Модулі побудови n-gram реалізовані з використанням багатопотоковості. Це дозволяє пришвидшити виконання роботи модуля, за рахунок зберігання в пам'яті декількох рядків, доступ до яких в багатопотоковому режимі відбувається швидше, ніж зчитування цих рядків з файлу. Модулі з побудови n-gram є «незалежними», а тому можуть бути використані для інших програм.

5 ТЕСТУВАННЯ СИСТЕМИ

Тестування системи проходило на базі текстового корпусу. В основу якого для лінгвістичної моделі покладено матеріал, завантажений з ряду Інтернет-сайтів, що містять тексти новин та публіцистики (60%), художніх творів (8%), текстів енциклопедичного характеру (24%), текстів юридичного спрямування (8%). Потрібно зазначити, що серед матеріалу, завантаженого з новинних сайтів, містяться коментарі та відгуки відвідувачів, тобто присутні текстові зразки спонтанного типу мовлення.

Нижче наведино приклади результатів роботи модулів побудови unigram та k-skip-bigram для k=0 і k=1 для речень «кругом на вулиці метушня всі кудись поспішають я також влився в потік людей <unk> додому».

Таблиця 5.1 – Результати роботи модулів побудови unigrams та k-skip-n-grams.

unigrams	bigrams	1-skip bigrams
<S>	<S> кругом	<S> на
кругом	кругом на	кругом вулиці
на	на вулиці	на метушня
вулиці	вулиці метушня	вулиці всі
метушня	метушня всі	метушня кудись
всі	всі кудись	всі поспішають
кудись	кудись поспішають	кудись </S>
поспішають	поспішають </S>	
</S>		

Продовження таблиці 5.1

unigrams	bigrams	1-skip bigrams
<S>	<S> я	<S> також
я	я також	я влився
також	також влився	також в
влився	влився в	влився потік
в	в потік	в людей
потік	потік людей	потік <unk>
людей	людей <unk>	людей додому
<unk>	<unk> додому	<unk> </S>
додому	додому </S>	
</S>		

Результатом роботи модуля підготовки даних для кластеризації є файл з unigrams, bigrams, 1-skip bigrams та їх частотами. Файл має наступний вигляд:

```
...
ячмінь  7630
ячменю  6200
яшин    3150
яшина   1320
ящірка  1760
ящірки  2360
ящірок  1680
ященко  2920
```

```
ящик    8380
ящиків  2810
ящика   2430
ящиках  1260
ящики   4110
ящику   2260
<S> <unk>       1515252
<S> і   1336556
<S> і-  260
<S> іі  180
<S> ієн 32
<S> ієрархії    44
<S> ієрархічні  152
<S> ієрархію    12
<S> ієрархія    268
<S> ієрархи     136
<S> ієрогліф    84
<S> ієрогліфами 4
<S> ієрогліфи   172
...
```

Таблиця 5.2 – Результат роботи програми кластеризації clustlm

<0>	<20>	<50>	<100>	<18>
сил	добре	кажуть	президентських	литвин
підприємств	швидко	знають	парламентських	азаров
організацій	легко	говорять	дострокових	луценко
відносин	правильно	знали	олімпійських	мороз
послуг	спокійно	говорили	позачергових	герман
засобів	погано	сказали	чергових	шевченко

Продовження таблиці 5.2

<0>	<20>	<50>	<100>	<18>
досліджень	серйозно	бачили	чесних	тігіпко
партій	відверто	розуміють	неоподатковуваних	іванович
військ	чесно	побачили	зимових	вища
інтересів	щиро	домовилися	пленарних	турчинов
ресурсів	нормально	казали	позачергової	томенко
проектів	глибоко	зрозуміли	повторних	кириленко
закладів	назавжди	пишуть	позачерговій	табачник
установ	чудово	думають	азартнох	симоненко
земель	уважно	забули	одночасних	гриценко
програм	мовчки	вірять	березневих	бойко
груп	повільно	подіваються	всенародних	порошенко
рад	гарно	чули	дочасних	мельниченко
будинків	тяжко	розповідають	паралімпійських	єхануров
систем	обережно	писали	бухарестському	миколайович
банків	умовно	розуміли	січневих	васильович
технологій	сумно	думали	прожиткових	михайлович
зв’язків	неправильно	помітили	пропорційних	штати

5.1 Висновки, шляхи покращення

Не важко пересвідчитись, що в наведеному вище прикладі (табл. 5.1) unigrams, bigrams та 1-skip-bigrams виділено вірно. Як видно з таблиці 5.2 наведені класи містять семантично близькі словоформи, що свідчить про правильне виконання алгоритму кластеризації. Тому можна сказати, що розроблене ПЗ відповідає усім вимогам, поставленим у відповідному розділі даної роботи (розділ 1).

Реалізація частини ПЗ у вигляді «незалежних» модулів дає змогу використовувати дані модулі для обробки текстових файлів у процесі вирішення інших задач.

У майбутньому може виникнути потреба модифікації алгоритму кластеризації для кластеризації 3-gram та k-skip-n-gram. Розроблений модуль побудови bigram дає можливість будувати k-skip-n-gram вже сьогодні, але їх кластеризація потребує окремих досліджень.

ВИСНОВКИ

В магістерській дисертації було проведено порівняльний аналіз методів розпізнавання злитого мовленнєвого сигналу, який показав, що для розпізнавання мовленнєвого сигналу краще використовувати приховані Марківські моделі, так як кожному елементу мовлення у відповідність можна поставити Марківський випадковий процес.

Аналіз проблем розпізнавання українського мовлення показв, що однією з головних проблем є довільний порядок слів в українській мові. Ми намагаємось статистично будувати послідовність слів, що розпізнаються,тому було проведено аналіз методів побудови лінгвістичної моделі. Найкращим варіантом виявилась комбінація кластеризованої лінгвістичної моделі та моделі з урахуванням фонетичної близькості слів.

Так як лінгвістична модель не має однозначної оцінки, тому для обґрунтування застосування skipping-bigram було проведено ряд додаткових дослідів з порівняння skipping-bigram та bigramю. Результати проведених експериментів показали, що застосування skipping-bigram дозволяє краще моделювати ймовірні послідовності слів, а це є особливо важливим за відсутності можливості збільшувати навчальний корпус.

Було розроблено математичну модель кластеризації лінгвістичної бази знань, для цього було модифіковано алгоритм кластеризації. Розроблена модель була реалізована у вигляді модулів, що задають правила обробки даних лінгвістичної бази даних для отримання нових знань. Зазначені бази даних та знань суттєво відрізняються від стандартних баз даних/знань, так як розроблені для певної предметної області.

Реалізовано три модулі для розробленої математичної моделі, для подальшої роботи з програмою кластеризації clustlm. Дані модулі забезпечують коректну обробку і підготовку даних до кластеризації. Модулі реалізовані «незалежно» один від одного, а тому можуть і будуть використовуватись для роботи над іншими задачами розпізнавання природно-мовного тексту.

Згідно проведеного тестування та аналізу розроблений метод кластеризації лінгвістичної бази знань дає задовільні результати: слова розподіляються до кластерів за семантичним значенням, тому кластеризована лінгвістична база знань може бути включена до роботи повного циклу розпізнавання мовленнєвого сигналу.

В подальшому даний алгоритм кластеризації може бути покращений реалізацією через багатопотоковість. Також не зайвим буде проаналізувати розміри лінгвістичних моделей та спробувати зменшити їх.

ПЕРЕЛІК ПОСИЛАНЬ

1. Єлькіна К.С. Стан проблеми систем автоматичного розпізнавання та синтезу мовлення. Завдання та значення // Прикладна лінгвістика 2010: проблеми і рішення. –2010. — с.56

2. Gales M. The Application of Hidden Markov Models in Speech Recognition / Gales M., Young S. — Foundations and Trends in Signal Processing, 2007. — 124p.

3. Робейко В.В. Розпізнавання спонтанного мовлення на основі акустичних композитних моделей слів у реальному часі / Робейко В.В., Сажок М.М. // Штучний інтелект. — №4'2011. — Донецьк, 2011. — 12с.

4. Уосермен Ф. Нейрокомп'ютерна техніка: Теорія і практика/Ф. Уосермен ; [пер. з англ. Ю. А. Зуєв, В. А. Точенов]. – М.ИПРЖ, 1992. – 240 с.

5. Сажок Н.Н. Кластеризация слов при построении лингвистической модели для автоматического распознавания речевого сигнала // Информационные технологии и системы. — 2012. — с.59-66

6. M. Sazhok, V. Robeiko. Language Model Comparison for Ukrainian Real-Time Speech Recognition System. SPECOM 2013, LNAI 8113, pp. 211–218, 2013. Springer International Publishing Switzerland 2013.

7. Сажок М.М. Багаторівнева багатозначна модель перетворення орфографічного тексту на фонемний // Штучний інтелект. — 2012. — с.65-75

8. Український лінгвістичний портал. [Електронний ресурс]. — Режим доступу: http://lcorp.ulif.org.ua/dictua/

9. Сажок Н.Н. Кулик О.О. Врахування фонетичної близькості слів при формуванні лінгвістичних моделей // Системы и средства искусственного интеллекта. — 2013. — с.149-150

10. Martin S., Liermann J., Ney H. Algorithms for bigram and trigram word clustering // Proc. of Eurospeech. — Madrid, 1995. — Vol. 2, — 1293–1256 p.

11. Кулик О.О. Сирота С.В. Застосування skipping n-grams при моделюванні кластеризованої бази знань // Інтелектуальний аналіз інформації. — 2015. — с.127-131

12. The HTK Book Version 3.4 / S.J. Uoung, G. Evermann, M. Gales et al. — Cambridge University, 2006. — 360 p.