Вінницький національний технічний університет

Міністерство освіти і науки України

Кваліфікаційна наукова праця

на правах рукопису

ЯХИМОВИЧ ОЛЕКСАНДР ВІКТОРОВИЧ

УДК 004.891

ДИСЕРТАЦІЯ

ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ ПОШУКУ КЛЮЧОВИХ СЛІВ НА ОСНОВІ ПАРСИНГУ АНГЛОМОВНИХ ТЕКСТІВ

05.13.06 – інформаційні технології

Технічні науки

Подається на здобуття наукового ступеня кандидата технічних наук

Дисертація містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело

_____О. В. Яхимович

Науковий керівник: Бісікало Олег Володимирович

доктор технічних наук, професор

Вінниця – 2021

АНОТАЦІЯ

Яхимович О. В. Інформаційна технологія пошуку ключових слів на основі парсингу англомовних текстів. – Кваліфікаційна наукова праця на правах рукопису.

Дисертація на здобуття наукового ступеня кандидата технічних наук за спеціальністю 05.13.06 «Інформаційні технології». – Вінницький національний технічний університет, Вінниця, 2021.

Завдання пошуку ключових слів тексту виникає у бібліотечній справі, лексикографії та термінознавстві, а також в усіх задачах інформаційного пошуку. В даний час обсяги і динаміка інформації, яка підлягає обробці в цих областях, роблять особливо актуальною задачу автоматичного пошуку ключових слів, які можуть використовуватися для створення і розвитку термінологічних ресурсів, а також для ефективної обробки документів – індексування, реферування, кластеризації та класифікації. Популярність та затребуваність пошукових машин для кожного користувача Інтернету висуває високі вимоги до релевантності результатів пошуку, яка прямо пропорційна якості процесів знаходження ключових слів у текстовій інформації.

Існує значна кількість доступних систем автоматичного пошуку ключових слів, розроблених і орієнтованих на обробку природних мов. В основу роботи таких систем покладено методи пошуку ключових слів тексту, які умовно можна розділити на дві категорії – експертно-лінгвістичні та статистичні. Перша група методів ґрунтується на значеннях слів, отриманих експертним шляхом, зокрема використовують словники з зібраними семантичними даними про кожне слово або онтології предметних областей. Однак, лінгвістичні методи ресурсоємні на ранніх етапах. Тому найбільша кількість відомих напрацювань у напряму експертно-лінгвістичної обробки текстів відома для визнаної мови міжнародного спілкування – англійської. З іншого боку,

традиційні статистичні методи, які є фактично мово-незалежними, супроводжуються значними обсягами «вербального шуму», що суттєво впливає на якість пошуку ключових слів. Внаслідок цього статистичні методи зазвичай супроводжуються емпіричними процедурами, налаштованими на конкретний клас задач, що суттєво звужує їхню область застосування.

Підвищення якості процесу пошуку ключових слів вимагає залучення додаткової інформації, бажано універсального, а не специфічного характеру. Зокрема, відсутня формальна постановка та розв'язок задачі пошуку ключових слів як згортки інформації у тексті. Не враховуються у відомих методах пошуку ключових слів результати аналізу зв’язків між лексичними одиницями тексту, а інформаційна оцінка результатів парсингу тексту не береться до уваги як складова відповідного критерія якості.

Перспективними є гібридні методи пошуку ключових слів, для яких швидкість статистичної обробки тексту підсилюється можливостями сучасних лінгвістичних пакетів, що мають найбільш розвинутий функціонал, у першу чергу, для англійської мови. Тому актуальним науковим завданням є підвищення якості пошуку ключових слів у англомовному тексті шляхом розробки інформаційної технології пошуку ключових слів на основі парсингу англомовних текстів.

Мета дисертаційного дослідження полягає у підвищенні якості пошуку ключових слів у англомовному тексті.

Для досягнення вказаної мети в роботі розв’язуються такі основні задачі:

1. Аналіз й порівняльна характеристика відомих підходів, методів та засобів пошуку ключових слів.

2. Побудова математичної моделі процесу пошуку ключових слів на основі інформаційної оцінки результатів парсингу тексту.

3. Розробка методу пошуку ключових слів на основі визначення зв'язків між словоформами.

4. Розробка методу зменшення впливу вербального шуму на пошук ключових слів.

5. Експериментальне дослідження результатів запропонованих методів у порівнянні з аналогами.

6. Розробка та апробація інформаційної технології пошуку ключових слів англомовного тексту.

Науковою новизною виконаного дисертаційного дослідження визначено:

1. Удосконалено модель пошуку ключових слів, яка, на відміну від існуючих, побудована на основі інформаційної оцінки результатів парсингу тексту та враховує результати аналізу зв’язків між лексичними одиницями тексту, що дозволило формалізувати критерій якості процесу пошуку ключових слів.

2. Уперше розроблено метод пошуку ключових слів, який, на відміну від існуючих, базується на знаходженні синтаксичних зв’язків між словоформами у реченнях англомовного тексту за допомогою технологічних можливостей парсингу сучасних лінгвістичних пакетів. Запропонований метод дає змогу підвищити чисельні характеристики якості пошуку ключових слів, а саме повноту (за Жаккаром) і точність.

3. Удосконалено метод зменшення впливу вербального шуму на пошук ключових слів, який, на відміну від існуючих, побудовано на основі стенфордської класифікації зв'язків між лексичними одиницями речення, що дозволило підвищити якість результатів пошуку ключових слів у порівнянні з основним методом.

4. Набула подальшого розвитку інформаційна технологія пошуку ключових слів, яка, на відміну від існуючих, враховує додаткову інформацію процесів парсингу речень у межах послідовного застосування двох запропонованих методів, що дозволило уточнити чисельні оцінки змістовних параметрів тексту та підвищити якість пошуку його ключових слів.

Практична цінність отриманих в дисертації результатів полягає у наступному: формальному описі методики пошуку ключових слів англомовного тексту, створенні алгоритму її реалізації та розробці програмного забезпечення, що знаходить ключові слова на основі врахування значимих зв’язків між словоформами у реченнях англомовного тексту та подальшою фільтрацією вербального шуму. Створені моделі, алгоритми та програмні засоби можуть бути використані при вирішенні практичних задач комп’ютерної лінгвістики, які потребують знаходження ключових слів, наприклад, для підвищення точності аналізу контенту сайту і підняття позиції сайту в результатах пошуку. Використання мово-незалежних засобів запропонованої інформаційної технології у поєднанні з необхідними, згідно з отриманою специфікацією, технологічними ресурсами лінгвістичного аналізу інших природних мов дозволить розширити область застосування інформаційної технології, зокрема на українську мову.

Основні результати та дисертаційна робота в цілому апробовані на восьми науково-практичних конференціях:

- міжнародній Інтернет-конференції «Молодь в технічних науках: дослідження, проблеми, перспективи» (МТН-2015), м. Вінниця, 23-26 квітня 2015 р.;

- першій міжнародній конференції «Адаптивні технології управління навчанням», м. Одеса, 23-25 вересня 2015 р.;

- третій міжнародній науковій конференції «Вимірювання, контроль та діагностика в технічних системах» (ВКДТС-2015), м. Вінниця, 27-29 жовтня 2015 р.;

- XLIV, XLV, XLVI, XLVII та XLVIII науково-технічних конференціях підрозділів ВНТУ факультету комп'ютерних систем та автоматики, м. Вінниця, щорічно у 2015 – 2019 рр.

Дослідження, результати яких представлено в дисертації, проводились відповідно до пріоритетних тематичних напрямів науково-технічних розробок на період до 2020 року «Технології та засоби розробки програмних продуктів і систем», затверджених постановою Кабінету Міністрів України №556 від 23.08.2016 р. Зокрема, дослідження проводились згідно планів наукових досліджень кафедри автоматизації та інтелектуальних інформаційних технологій Вінницького національного технічного університету, а саме у науково-дослідних роботах «Інтелектуальна інформаційна технологія образного аналізу тексту та синтезу інтегрованої бази знань природно-мовного контенту» (№ ДР 0114U003462) та «Ідентифікація прихованих залежностей в онлайнових соціальних мережах на основі методів нечіткої логіки та комп'ютерної лінгвістики» (№ ДР 0117U000575).

Робота впроваджена на ТОВ НВП «СПІЛЬНА СПРАВА», а також в навчальний процес кафедри АІІТ Вінницького національного технічного університету, що підтверджено виданням навчального посібника “A lexical relationships-based keywords selection in an English text” та актом про результати впровадження від 10.01.2020. Результати експериментів показали, що запропонована інформаційна технологія одночасно збільшує у межах від 8,1% до 12,7% повноту за метрикою Жаккара та від 9,1% до 14,3% абсолютну точність пошуку ключових слів для англомовних текстів обсягом 140-1400 слів у порівнянні з аналогами.

Ключові слова: інформаційна технологія, ключові слова, вербальний шум, лінгвістичний пакет, DKPro Core, синтаксичний аналіз, словосполучення.

ABSTRACT

Yahimovich O. V. Information technology of searching keywords based on parsing English texts. – Qualification research paper, manuscript copyright.

Thesis for the degree of a candidate of technical sciences in specialty 05.13.06 «Information technology». – Vinnytsia National Technical University, Vinnytsia, 2021.

The task of searching keywords in the text arises in the library business, lexicography and terminology, as well as in all tasks of information retrieval. Currently, the amount and dynamics of information to be processed in these areas, make it especially important to the automation process of searching keywords in texts that can be used to create and develop terminological resources, as well as for efficient document processing - indexing, abstracting, clustering and classification. The extraordinary demand and popularity of search engines for every Internet user puts forward high demands on the relevance of search results, which is directly proportional to the quality of the technological process of searching keywords in textual information. Therefore, the topic of the thesis is relevant.

There are a lot of automatic searching keywords systems available, designed and focused on natural language processing. The work of such systems is based on methods of searching keywords in the text, which can be divided into two categories - expert-linguistic and statistical. The first group of methods is based on the meanings of words obtained by experts, in particular, use dictionaries with collected semantic data about each word or ontology of subject areas. However, linguistic methods are resource-intensive in the early stages. Therefore, the largest number of known developments in the direction of expert-linguistic processing of texts is known for the international communication language - English. On the other hand, traditional statistical methods, which are, in fact, language-independent, are accompanied by significant amounts of "verbal noise", which significantly affects the quality of

searching keywords in texts. As a result, statistical methods are usually accompanied by empirical procedures tailored to a specific class of problems, which significantly narrows their scope of its usage.

Increasing the quality of the searching keywords process requires additional information, preferably universal rather than specific knowledge. In particular, there is no formal description and solution of the problem of searching keywords as a convolution of information in the text. The known methods of searching keywords do not take into account the results of the analysis of relationships between lexical units of the text, and the informational evaluation of the results of parsing the text is not taken into account as part of the relevant quality criterion.

The hybrid methods of keyword search, for which the speed of statistical text processing is enhanced by the capabilities of modern linguistic packages that have the most advanced functionality, especially for the English language, are promising. Therefore, the actual scientific task is to increase the quality of searching keywords in English text by developing information technology for searching keywords based on parsing English texts.

The purpose of the qualification research paper is to improve the quality of searching keywords in English text.

In order to achieve the mentioned purpose, the following basic tasks are solved in the work:

1. Analysis and comparative characteristics of known approaches, methods and ways of searching keywords.

2. Formalization of a mathematical model of the searching keywords process based on information evaluation of the results of text parsing.

3. Development of a method for searching keywords based on determining the relationships between word forms.

4. Development of a method to reduce the impact of verbal noise for searching keywords.

5. Experimental research of the results of the proposed methods in comparison with analogues.

6. Development and approbation of information technology for searching keywords in English text.

The scientific novelty of the qualification research paper is:

1. The model of searching keywords has been improved, which, unlike the existing ones, is based on the information evaluation of parsing text results and takes into account the results of analysis of relationships between lexical units of text, which allowed to formalize the quality criterion of searching keywords process.

2. For the first time, searching keywords method has been developed, which, unlike the existing ones, is based on finding syntactic relationships between word forms in sentences of English text with the help of technological capabilities of parsing of modern linguistic packages. The proposed method allows to improve the numerical characteristics of searching keywords quality, namely completeness (according to Jacquard) and accuracy.

3. The method of reducing the impact of verbal noise for searching keywords has been improved, which, unlike the existing ones, is based on the Stanford classification of relationships between lexical units of a sentence, which has improved the quality of results of searching keywords compared to the main method.

4. The information technology of searching keywords has been further developed, which, unlike the existing ones, takes into account additional information of sentence parsing processes within the bounds of the consistent use of the two proposed methods, which allowed to refine numerical estimates of content parameters of the text and improve the quality of searching keywords.

The practical value of the results obtained in the qualification research paper is as follows: formalization of the searching keywords method of the English text, creation of an algorithm for its implementation and development of software that searching keywords based on complex relationships between word forms in sentences

of English text and subsequent filtering of verbal noise. Created models, algorithms and software can be used to solve practical problems of computational linguistics that require searching keywords, for example, to improve the accuracy of site content analysis and raise the position of the site in search results. The use of language-independent means of the proposed information technology in combination with the necessary, according to the obtained specification, technological resources of linguistic analysis of other natural languages will expand the area of information technology, in particular for Ukrainian texts.

The results of the qualification research paper were reported and discussed at 8 scientific and technical conferences:

- international internet conference «Youth in Science: Research, Problems, Prospects» (MTH-2015), Vinnytsia National Technical University, April 23-26, 2015;
- the first international conference «Adaptive technologies of learning management», Odessa, September 23-25, 2015;
- the third international scientific conference «Measurement, control and diagnostics in technical systems» (ВКДТС-2015), Vinnytsia National Technical University, October 27-29, 2015;
- XLIV, XLV, XLVI, XLVII and XLVIII scientific and technical conferences of teaching staff, staff and students of Vinnytsia National Technical University, Faculty of Computer Systems and Automation, annually in 2015 - 2019.

The results of the qualification research paper which are presented, was conducted in accordance with the priority thematic areas of scientific and technical development for the period up to 2020 "Technologies and tools for software and systems development", approved by the Cabinet of Ministers of Ukraine №556 from 23.08.2016. In particular, the research was conducted according to the research plans of the Automation and Intelligent Information Technologies Department of Vinnytsia National Technical University, namely in the research works «Intelligent information technology of figurative analysis of text and synthesis of integrated knowledge base

of natural language content» (№ 0114U003462) та «Identification of hidden dependencies in online social networks based on the methods of fuzzy logic and computational linguistics» (№ 0117U000575).

The results of the qualification research paper were implemented at LLC «Spilna Sprava» and also to the educational process of the Automation and Intelligent Information Technologies Department of Vinnytsia National Technical University. This is confirmed by the publication of a textbook “A lexical relationships-based keywords selection in an English text” and act on the results of implementation from January 10, 2020. The results of the experiments showed that the proposed information technology simultaneously increases in the range from 8.1% to 12.7% the completeness of the Jacquard metric and from 9.1% to 14.3% the absolute accuracy of searching keywords for English texts of 140-1400 words in comparison with analogues.

Keywords: information technology, keywords, verbal noise, linguistic package, DKPro Core, syntactic analysis, phrase.

СПИСОК ОПУБЛІКОВАНИХ ПРАЦЬ ЗА ТЕМОЮ ДИСЕРТАЦІЇ

[1] O.V. Bisikalo, W. Wójcik, O.V. Yahimovich, and S. Smailova, "Method of determining of keywords in English texts based on the DKPro Core", *Proceedings of SPIE 10031, Photonics Applications in Astronomy, Communications, Industry, and High-Energy Physics Experiments 2016*, 100314T, 2016. DOI:10.1117/12.2249225.

[2] O. Bisikalo, A. Lisovenko, O. Jahumovuch, S. Trachenko, and M. Pradivliannyi, "System of computational linguistic on base of the figurative text comprehension", *Proceedings of the 2016 IEEE 1st International Conference on Data Stream Mining and Processing (DSMP 2016)*, pp. 69-74, 2016. DOI: 10.1109/DSMP.2016.7583510.

[3] О.В. Бісікало, та О.В. Яхимович, "Метод визначення ключових слів англомовного тексту на основі DKPro Core", *Технологический аудит и резервы производства: Информационные технологии*, Том 1, № 2(21), с. 26-30, 2015. ISSN 2226-3780.

[4] О.В. Бісікало, А.І. Лісовенко, О.В. Яхимович, та С.С. Траченко, "Визначення змістовних ознак тексту на основі аналізу зв'язків між лексичними одиницями", *Вісник НТУ «ХПІ». Серія "Механіко-технологічні системи та комплекси"*, № 21 (1130), с. 83-89, 2015. ISSN 2411-2798.

[5] О.В. Бісікало, та О.В. Яхимович, "Знаходження ключових слів англомовного тексту за допомогою інструментальних засобів пакету DKPro Core", *Інформаційні технології та комп'ютерна інженерія*, № 2(34), с. 10-14, 2015. ISSN 1999-9941.

[6] О.В. Бісікало, та О.В. Яхимович, "Автоматизоване визначення лексичних технологій з тезаурусу технічного спрямування", *Оптико-електронні інформаційно-енергетичні технології*, № 1 (31), с. 26-38, 2016. ISSN 1681-7893.

[7] О.В. Бісікало, О.В. Яхимович, та Я.В. Яхимович, "Розробка методу фільтрації вербального шуму в процесі пошуку ключових слів англомовного тексту", *Технологический аудит и резервы производства: Информационные технологии*, № 6(44), с. 33–41, 2018. ISSN 2226-3780.

[8] С.Д. Штовба, О.В. Штовба, О.В. Яхимович, та М.В. Петричко, "Вплив синтаксичних зв'язків у реченнях на якість ідентифікації токсичних коментарів в соціальній мережі", *Наукові праці ВНТУ*, № 4, с. 1-8, 2019. DOI: https://doi.org/10.31649/2307-5376-2019-4-35-42.

[9] О.В. Яхимович, "Визначення ключових слів англомовного тексту з використанням технології DKPRO CORE", *Молодь в технічних науках: дослідження, проблеми, перспективи (МТН-2015) : Матеріали міжнародної Інтернет-конференції*, Вінниця, 2015, с. 72-74. ISBN 978-966-924-027-9.

[10] О.В. Бісікало, О.В. Яхимович, А.І. Лісовенко, та Траченко С.С., "Підтримка діалогу з навчальним контентом", *Адаптивні технології управління навчанням: матеріали першої міжнародної конференції*, Одеса, 2015, с. 97-100.

[11] О.В. Бісікало, О.В. Яхимович, А.І. Лісовенко, та Траченко С.С., "Моделювання процесів побудови парадигматичних зв’язків між словоформами на основі вимірювання текстової інформації", *Вимірювання, контроль та діагностика в технічних системах (ВКДТС-2015)*, Вінниця, 2015, с. 119-121.

[12] О.В. Яхимович, "Застосування інструментальних засобів пакету DKPRO CORE для визначення ключових слів англомовного тексту", *XLIV науково-технічній конференції підрозділів ВНТУ: факультет комп'ютерних систем та автоматики*, Вінниця, 2015, с. 7.

[13] О.В. Яхимович, "Визначення ключових слів з тексту повідомлень мікроблогів", *XLV науково-технічній конференції підрозділів ВНТУ: факультет комп'ютерних систем та автоматики*, Вінниця, 2016, с. 1119-1121.

[14] О.В. Яхимович, "Колізія при знаходженні ключових слів", *XLVI науково-технічній конференції підрозділів ВНТУ: факультет комп'ютерних систем та автоматики*, Вінниця, 2017, с. 1312-1314.

[15] О.В. Яхимович, "Зменшення вербального шуму при визначенні ключових слів", *XLVII науково-технічній конференції підрозділів ВНТУ: факультет комп'ютерних систем та автоматики*, Вінниця, 2018, с. 1463-1465.

[16] О.В. Яхимович, "Формалізація задачі визначення ключових слів тексту", *XLVIII науково-технічній конференції підрозділів ВНТУ: факультет комп'ютерних систем та автоматики*, Вінниця, 2019, с. 1136-1138.

[17] О.В. Бісікало, Р.Н. Квєтний, С.Г. Кривогубченко, Л.Є Азарова, та О.В. Яхимович, "Синтез інтегрованої бази знань природно-мовного контенту", ВНТУ, Вінниця, Д/б № 0114U003462, 2015.

[18] О.В.Бісікало, Р.Н. Квєтний, С.Г. Кривогубченко, А.І. Лісовенко, та О.В. Яхимович, "Розв’язання семантико-залежних задач обробки природно-мовних об’єктів на основі бази знань", ВНТУ, Вінниця, Д/б № 0114U003462, 2016.

[19] О. В. Бісікало, А. І. Лісовенко, О. В. Яхимович, та В. В. Шолота, "Спосіб автоматичного пошуку ключових слів з використанням технології DKPro Core", *МПК G06F 17/21, G06F 17/27, G06F 17/28. № и 2019 00016*, Черв. 25, 2019.

[20] O. Bisikalo, and A. Yahimovich. *Keyword search based on lexical relationships in the text*. Beau Bassin-Rose Hill, Mauritius: Lap Lambert Academic Publishing, 2019. ISBN 978-620-0-00314-0.

[21] O. Bisikalo, and A. Yahimovich. *Lexical relationships-based keywords selection in english texts*. Vinnytsia, Ukraine: VNTU, 2020.

ЗМІСТ

ВСТУП

Обґрунтування вибору теми дослідження. Завдання пошуку ключових слів тексту виникає у бібліотечній справі, лексикографії та термінознавстві, а також в усіх задачах інформаційного пошуку. В даний час обсяги і динаміка інформації, яка підлягає обробці в цих областях, роблять особливо актуальною задачу автоматичного пошуку ключових слів, які можуть використовуватися для створення і розвитку термінологічних ресурсів, а також для ефективної обробки документів – індексування, реферування, кластеризації та класифікації. Популярність та затребуваність пошукових машин для кожного користувача Інтернету висуває високі вимоги до релевантності результатів пошуку, яка значно залежить від якості пошуку ключових слів у текстовій інформації.

Значний внесок у розвиток математичних моделей та методів семантичного аналізу природномовних текстів закладено закордонними дослідниками Н. Хомскі, Д. Маккарті, Т. Виноградом, К. Менінгом, Д. Журафски, Колмогоровим А.М., Мельчуком І.А., Апресяном Ю.Д., Поспєловим Д.А., Сосніним П.І., плідно працювали та продовжують дослідження у цьому напряму вітчизняні науковці Шабанов-Кушнаренко Ю.П., Бондаренко М.Ф., Широков В.А., Шлезінгер М.І., Анісімов А.В., Шаронова Н.В., Дарчук Н.П., Бісікало О.В., Хайрова Н.Ф, Шевченко І.В. тощо.

Натепер існує значна кількість доступних систем автоматичного пошуку ключових слів, розроблених і орієнтованих на обробку природних мов. В основу роботи таких систем покладено відомі методи пошуку ключових слів тексту, які умовно можна розділити на дві категорії – експертно-лінгвістичні та статистичні. Перша група методів ґрунтується на значеннях слів, отриманих експертним шляхом, зокрема використовують словники з зібраними семантичними даними про кожне слово або онтології предметних областей. Такі методи ресурсоємні на ранніх етапах – розробка онтологій, наприклад,

вельми трудомісткий процес [1]. Тому найбільша кількість відомих напрацювань у напряму експертно-лінгвістичної обробки текстів відома для визнаної мови міжнародного спілкування – англійської. З іншого боку, традиційні статистичні методи, які є фактично мово-незалежними, супроводжуються значними обсягами «вербального шуму», що суттєво впливає на якість пошуку ключових слів. Внаслідок цього статистичні методи зазвичай супроводжуються емпіричними процедурами, налаштованими на конкретний клас задач, що суттєво звужує їхню область застосування.

Підвищення якості процесу пошуку ключових слів вимагає залучення додаткової інформації, бажано універсального, а не специфічного характеру. Зокрема, відсутня формальна постановка та розв'язок задачі пошуку ключових слів як згортки інформації у тексті. Не враховуються у відомих методах пошуку ключових слів результати аналізу зв’язків між лексичними одиницями тексту, а інформаційна оцінка результатів парсингу тексту не береться до уваги як складова відповідного критерія якості.

Потрібно звернути увагу на гібридні методи пошуку ключових слів, для яких швидкість статистичної обробки тексту підсилюється можливостями сучасних лінгвістичних пакетів, що мають найбільш розвинутий функціонал, у першу чергу, для англійської мови. Тому *актуальним* науковим завданням є підвищення якості пошуку ключових слів у англомовному тексті шляхом розробки інформаційної технології пошуку ключових слів на основі парсингу англомовних текстів.

Зв’язок роботи з науковими програмами, планами, темами.

Дослідження, результати яких представлено в дисертації, проводились відповідно до пріоритетних тематичних напрямів науково-технічних розробок на період до 2020 року «Технології та засоби розробки програмних продуктів і систем», затверджених постановою Кабінету Міністрів України №556 від 23.08.2016 р. Зокрема, дослідження проводились згідно планів наукових

досліджень кафедри автоматизації та інтелектуальних інформаційних технологій Вінницького національного технічного університету. Автор брав участь у виконанні науково-дослідних робіт «Інтелектуальна інформаційна технологія образного аналізу тексту та синтезу інтегрованої бази знань природно-мовного контенту» (№ ДР 0114U003462) та «Ідентифікація прихованих залежностей в онлайнових соціальних мережах на основі методів нечіткої логіки та комп'ютерної лінгвістики» (№ ДР 0117U000575).

Мета і завдання дослідження. Мета дослідження полягає у підвищенні якості пошуку ключових слів у англомовному тексті.

Для досягнення поставленої мети необхідно розв’язати наступні задачі:

1. Аналіз й порівняльна характеристика відомих підходів, методів та засобів пошуку ключових слів.

2. Побудова математичної моделі процесу пошуку ключових слів на основі інформаційної оцінки результатів парсингу тексту.

3. Розробка методу пошуку ключових слів на основі визначення зв'язків між словоформами.

4. Розробка методу зменшення впливу вербального шуму на пошук ключових слів.

5. Експериментальне дослідження результатів запропонованих методів у порівнянні з аналогами.

6. Розробка та апробація інформаційної технології пошуку ключових слів англомовного тексту.

Об’єкт дослідження – процес обробки вербальної інформації для пошуку ключових слів у англомовному тексті.

Предмет дослідження – моделі, методи та технологічні засоби пошуку ключових слів у англомовному тексті.

Методи досліджень. Для побудови моделі пошуку ключових слів використано методи теорії множин та теорії інформації. Розробка методів

пошуку ключових слів та зменшення впливу вербального шуму базувалася на основі поєднання методів лінгвістичного аналізу англомовного тексту та статистичних методів. Під час оцінки релевантності запропонованої інформаційної технології застосовано методи вимірювання у метричних просторах.

Наукова новизна отриманих результатів.

Удосконалено модель пошуку ключових слів, яка, на відміну від існуючих, побудована на основі інформаційної оцінки результатів парсингу тексту та враховує результати аналізу зв’язків між лексичними одиницями тексту, що дозволило формалізувати критерій якості процесу пошуку ключових слів.

Уперше розроблено метод пошуку ключових слів, який, на відміну від існуючих, базується на знаходженні синтаксичних зв’язків між словоформами у реченнях англомовного тексту за допомогою технологічних можливостей парсингу сучасних лінгвістичних пакетів. Запропонований метод дає змогу підвищити чисельні характеристики якості пошуку ключових слів, а саме повноту (за Жаккаром) і точність.

Удосконалено метод зменшення впливу вербального шуму на пошук ключових слів, який, на відміну від існуючих, побудовано на основі стенфордської класифікації зв'язків між лексичними одиницями речення, що дозволило підвищити якість результатів пошуку ключових слів у порівнянні з основним методом.

Набула подальшого розвитку інформаційна технологія пошуку ключових слів, яка, на відміну від існуючих, враховує додаткову інформацію процесів парсингу речень у межах послідовного застосування двох запропонованих методів, що дозволило уточнити чисельні оцінки змістовних параметрів тексту та підвищити якість пошуку його ключових слів.

Практичне значення отриманих результатів. Прикладні результати дисертаційного дослідження полягають у формальному описі методики пошуку ключових слів англомовного тексту, створенні алгоритму її реалізації та розробці програмного забезпечення, що знаходить ключові слова на основі врахування значимих зв’язків між словоформами у реченнях англомовного тексту та подальшої фільтрації вербального шуму.

Створені моделі, алгоритми та програмні засоби можуть бути використані при вирішенні практичних задач комп’ютерної лінгвістики, які потребують знаходження ключових слів, наприклад, для підвищення точності аналізу контенту сайту і підняття позиції сайту в результатах пошуку. Використання мово-незалежних засобів запропонованої інформаційної технології у поєднанні з необхідними, згідно з отриманою специфікацією, технологічними ресурсами лінгвістичного аналізу інших природних мов дозволить розширити область застосування інформаційної технології, зокрема на українську мову.

Робота впроваджена на ТОВ НВП «СПІЛЬНА СПРАВА» (акт про результати впровадження від 10.01.2020), а також в навчальний процес кафедри АІІТ Вінницького національного технічного університету, що підтверджено виданням навчального посібника “A lexical relationships-based keywords selection in an English text” та актом про результати впровадження від 10.01.2020). Результати експериментів показали, що запропонована інформаційна технологія одночасно збільшує у межах від 8,1% до 12,7% повноту за метрикою Жаккара та від 9,1% до 14,3% абсолютну точність пошуку ключових слів для англомовних текстів обсягом 140-1400 слів у порівнянні з аналогами.

Особистий внесок здобувача. Усі результати, які складають основний зміст дисертаційної роботи, отримані здобувачем самостійно. У роботах [146], [142], [125], [126], [130], [94] здобувачеві належать усі теоретичні та практичні результати. У роботах, опублікованих у співавторстві, здобувачу належать: [116] – розробка програмного забезпечення для методу пошуку ключових слів на

основі знаходження синтаксичних зв’язків між словоформами у реченнях англомовного тексту за допомогою технологічних можливостей парсингу лінгвістичного пакету DKPro Core; [147] – запропоновано використання заміни займенників на відповідні до них іменники для зменшення впливу вербального шуму на пошук ключових слів; експериментально підтверджено, що результати пошуку ключових слів розробленим методом містять менше стоп-слів у порівнянні з частотним словником; [101] – запропоновано підхід до інформаційної оцінки процесів парсингу тексту у межах задачі пошуку ключових слів; [10] – запропоновано методику зменшення впливу вербального шуму на пошук ключових слів за допомогою вилучення слів, які відносяться до неінформативних частин мови, а також слів, що відносяться до списку стоп-слів; [156] – експериментально підтверджено доцільність використання розробленого методу пошуку ключових слів, оскільки він забезпечує кращу релевантність результатів пошуку ключових слів у порівнянні з аналогами за критеріями точності і повноти; [45] – запропоновано методику побудови онтологій на основі знаходження відношень між термінами, яка може бути використана для поліпшення якості інформаційного пошуку; [152] – запропоновано метод зменшення впливу вербального шуму на пошук ключових слів на основі стенфордської класифікації зв'язків між лексичними одиницями речення; [144] – запропоновано використовувати інформаційної технології пошуку ключових слів для покращення ідентифікації токсичних коментарів в соціальних мережах; [134], [119] – розробка програмного забезпечення для модулю побудови семантичного графу на основі зв'язків між словами у словосполученнях, отриманих з тексту, а також огляд сучасних лінгвістичних пакетів; [135] – запропоновано використання модульного підходу до фільтрації вербального шуму, що складається з модулів: виключення словосполучень кандидатів з не інформативними типами зв’язків, заміни займенників на іменники в словосполученнях кандидатів, виключення

ключових слів, які належать до попередньо визначеного переліку не інформативних для семантичного аналізу частин мови та списку стоп-слів; [132], [133] – розробка програмного забезпечення на основі пакету DKPro Core для задачі виявлення інформативних ознак тексту; запропоновано метод класифікації тексті, що забезпечує підвищення кількісних характеристик релевантності результатів, а саме повноту і точність; [107], [112] – проведено аналіз зв'язків між лексичними одиницями тексту; реалізація програмного експерименту з використанням лінгвістичного пакету DKPro Core на англомовних текстах різної довжини; здійснено розрахунок чисельних характеристик якості отриманих результатів за критеріями повноти та точності та доведено ефективність використання розробленого методу в порівнянні з аналогами.

Апробація матеріалів дисертації. Основні результати та дисертаційна робота в цілому апробовані на восьми науково-практичних конференціях:

- міжнародній Інтернет-конференції «Молодь в технічних науках: дослідження, проблеми, перспективи» (МТН-2015), м. Вінниця, 23-26 квітня 2015 р.;

- першій міжнародній конференції «Адаптивні технології управління навчанням», м. Одеса, 23-25 вересня 2015 р.;

- третій міжнародній науковій конференції «Вимірювання, контроль та діагностика в технічних системах» (ВКДТС-2015), м. Вінниця, 27-29 жовтня 2015 р.;

- XLIV, XLV, XLVI, XLVII та XLVIII науково-технічних конференціях підрозділів ВНТУ факультету комп'ютерних систем та автоматики, м. Вінниця, щорічно у 2015 – 2019 рр.

Публікації. За темою дисертації з викладенням її основних результатів опубліковано 21 науковій праці, серед яких 1 стаття у закордонному фаховому періодичному виданні, що входить до SCOPUS, 1 стаття у виданні, що входить

до SCOPUS, 6 статей у спеціалізованих фахових виданнях України, що індексуються міжнародними бібліометричними та наукометричними базами даних, 8 публікацій в матеріалах та тезах доповідей, 2 звіти про науково-дослідну роботу. Також, отримано 1 патент України на корисну модель (Пат. 135223 UA), опубліковано закордонну монографію та електронний навчальний посібник (Додаток А).

Структура та обсяг дисертації. Дисертаційна робота складається зі вступу, чотирьох розділів, висновків, списку використаних джерел і додатків. Основний зміст викладено на 137 сторінках друкованого тексту, містить 67 рисунки, 17 таблиць. Список використаних джерел містить 157 найменувань. Загальний обсяг 193 сторінки.

1 АНАЛІЗ ТА ДОСЛІДЖЕННЯ ВІДОМИХ МЕТОДІВ ПОШУКУ КЛЮЧОВИХ СЛІВ

Загальновідомо, що не всі слова в тексті рівнозначні. Множина ключових слів дозволяє представити текст у згорнутому вигляді, який досить точно відображає зміст вихідного тексту. Якщо людям відносно легко конспектувати, наприклад, зміст статті шляхом застосування слів, які можуть представити основний зміст відповідного тексту, то формалізація подібних методів не є простою задачею. Розділ присвячено порівняльному аналізу відомих методів та систем автоматизованого пошуку ключових слів тексту, визначенню їх обмежень і недоліків, а також обґрунтуванню напрямку та задач дослідження.

1.1 Загальна характеристика проблеми

Основний зміст документа (тексту) може бути виражений за допомогою певних слів, узятих безпосередньо з цього тексту [2]. Зазвичай до кожного розгорнутого тексту можна скласти цілий набір ключових слів різного обсягу (найчастіше від 5 до 15 слів). Але взагалі кількість ключових слів може варіюватися в широких межах. Відповідно до думки Л.В. Цукрового і А.С. Штерн, найоптимальнішим є невеликий набір ключових слів і словосполучень – 7-10 слів. В окремих випадках компресія тексту може призвести до знаходження одного основного ключового слова, яке має найбільшу частоту [3], [4].

Ключовим словом будемо вважати таке слово в тексті, яке здатне в сукупності з іншими ключовими словами відображати зміст цього тексту [5]. Набір ключових слів близький до анотації, плану і конспекту, які теж представляють документ з меншою деталізацією, але, на відміну від ключових слів, пов’язані у синтаксичні структури [1].

Термін «ключові слова» значною мірою умовний – як ключові ознаки в тексті можуть виступати не тільки слова, а й словосполучення і, навіть, речення.

Ключове (опорне) слово – це термін, що відноситься до основного змісту тексту [6] і повторюється в ньому кілька разів (з урахуванням всіх можливих синонімів). Ключове словосполучення – це поєднання слів, серед яких є одне або кілька ключових [7], [8]. Ключовим також може вважатися речення, що містить два і більше ключових слова або ключових словосполучення [9].

Ключові слова мають ряд суттєвих ознак:

- високий ступінь повторюваності даних слів у тексті, частотність їх вживання;

- здатність знака концентрувати, згортати інформацію, закладену в увесь текст, об'єднувати «його основний зміст»; ця ознака особливо яскраво проявляється у ключових словах у позиції заголовку.

Однак вибір ключових слів є дуже непростою операцією і вимагає зваженого підходу. Слід вибирати ті ключові слова, які найбільш точно відображають специфіку розглянутої теми. При цьому необхідно уникати випадкових і загальних фраз, не рекомендується повторювати кілька разів одні й ті ж ключові слова. Отже, процес пошуку ключових слів є аналітичним [9].

У процесі попереднього оброблення тексту проводиться видалення неінформативних частин [10]. Найперше до таких відносять стоп-слова – це слова, що не представляють цінності як потенційно ключові. Найчастіше це прийменники, сполучники, вигуки тощо [10].

До уваги беруться також N-грами – це термін з комп'ютерної лінгвістики, що означає послідовність з N елементів тексту (термів), наприклад, слів або їх послідовностей. Кандидати в ключові слова можна відбирати у вигляді N-грам, що не розділені знаками пунктуації (крім дефіса і лапок) і стоп-словами [11], [12]. Ключовими словами можуть бути як поодинокі слова, так і пари слів, трійки тощо.

Інший підхід ґрунтується на аналізі зв'язків між словами в реченнях і в тексті, отриманих або за допомогою тих же N-грам (підхід менш трудомісткий), або на розібраному тексті. Якщо текст пройшов більш затратну обробку на етапах аналізу (морфологічний дасть базові словоформи, синтаксичний – зв'язки між словами, семантичний – смислову карту зв'язків), можна отримати більш точну інформацію про текст, наприклад, на підставі дерев синтаксичного розбору (за наявності синтаксичної розмітки) [13].

Найважливішим етапом в задачі знаходження ключових фраз є розрахунок їх ваг інформативності, який дозволяє оцінити їх значимість по відношенню один до одного в документі. Для кожної з відібраних ключових фраз розраховуються ознаки, які дозволяють судити про важливість кандидата для даного документа. Набір відібраних ключових фраз ранжується за значеннями ознак, наприклад, відповідно до їх частотності та ваги інформативності, розрахованими за однією з методик. Після ранжування проводиться відбір кращих ключових фраз з цього списку або відбираються кандидати, що перевищують встановлений мінімальний поріг значення ознаки [1].

Отже, вибір найбільш інформативної ознаки при пошуку ключових слів та / або фраз має безпосередній вплив на якість результатів їх визначення.

1.2 Аналіз статистики в текстах

В усіх текстових документах, створених людиною, можна спостерігати статистичні закономірності. У будь-якій мові є слова, які зустрічаються частіше, ніж інші, але не мають самостійного значення. Є слова, які зустрічаються рідше, але мають набагато більше смислове значення.

У 1949 році гарвардський професор-лінгвіст і філолог Джордж Ципф, працюючи над принципом найменшого зусилля, сформулював кілька закономірностей, отриманих не на основі математичних висновків, а на основі

аналізу статистики частоти слів в текстах на багатьох мовах, тобто емпірично [14].

У той час, коли Ципф сформулював помічені їм закономірності розподілу частоти слів, законом вони не вважалися – ще не було комп'ютерів і не можна було провести точні розрахунки, що підтверджують виявлені закономірності. У подальшому були проведені масштабні чисельні дослідження, які підтвердили і уточнили помічені закономірності. При цьому провідну роль в обґрунтуванні законів зіграли роботи Б. Мандельброта [14].

Зокрема Ципф вважав, що слова з великою кількістю букв зустрічаються в тексті рідше коротких слів. Ґрунтуючись на цьому постулаті, Ципф вивів два універсальних закони [14].

Виміряємо кількість входжень кожного слова в текст і візьмемо тільки одне значення з кожної групи, що має однакову частоту. Розташуємо частоти по мірі їх спадання і пронумеруємо, а порядковий номер частоти назвемо рангом частоти (позначимо r_i ранг слова i). Слова, що зустрічаються найбільш часто матимуть ранг 1, наступні за ними – 2 і так далі.

Тоді очевидно, що ймовірність зустріти довільне, заздалегідь вибране слово буде дорівнює відношенню кількості входжень цього слова до загального числа слів у тексті (n_i - кількість входжень i слова, $|N|$ - кількість слів у тексті) [14].

$$p = n_i / |N| \tag{1.1}$$

Ципф виявив таку закономірність: добуток ймовірності виявлення слова в тексті на ранг частоти являє собою постійне число (С) [14].

$$\frac{(n_i \cdot r_i)}{|N|} = C \tag{1.2}$$

Закон показує, що поширеність слова в тексті змінюється по гіперболі, залежно від кількості входжень. Наприклад, друге зі слів, що зустрічається найчастіше, зустрічається приблизно в два рази рідше, ніж перше, третє – в три рази рідше, ніж перше і так далі [14].

Значення константи в різних мовах різна, але всередині однієї мовної групи залишається приблизно незмінною, незалежно від тексту. Для російських текстів константа Ципфа приблизно дорівнює 0,08, для англійських текстів – 0,1.

Перший закон не враховує факт того, що різні слова можуть входити в текст з однаковою частотою. Ципф встановив, що частота і кількість слів, що входять в текст з цією частотою, також мають залежність. Якщо побудувати графік, відклавши по осі абсцис частоту входження слова, а по осі ординат – кількість слів у даній частоті, то отримана крива буде зберігати свій вигляд для всіх без винятку текстів [14].

Як і для першого закону, це твердження вірне в межах однієї мови. Однак і міжмовні відмінності невеликі. Якою б мовою текст не був написаний, вигляд кривої Ципфа залишиться незмінною. Може трохи відрізнятися лише коефіцієнт гіперболи [14].

Дж. Ципфом та іншими дослідниками було встановлено, що гіперболічному розподілу підпорядковуються не тільки всі природні мови світу, але й інші явища соціального і біологічного характеру: розподілу вчених за кількістю опублікованих ними статей, міст США за чисельністю населення, населення за розмірами доходу в капіталістичних країнах та інші [14].

Також важливим є той факт, що і документи всередині якої-небудь галузі знань можуть розподілятися відповідно до цього закону.

Дослідження показують, що найбільш значущі для тексту слова лежать у середній частині графіка кривої Ципфа. Цей факт має просте обґрунтування – слова, які трапляються дуже часто, в основному виявляються прийменниками чи займенниками. З іншого боку слова, що зустрічаються рідко, в більшості випадків, не мають вирішального смислового значення.

Від установки ширини діапазону (вікна пошуку ключових слів) залежить статистична якість пошуку значущих слів. Якщо встановити велику ширину діапазону, то в ключові слова будуть потрапляти допоміжні слова; якщо встановити вузький діапазон – можна втратити значущі терміни. Тому, в кожному окремому випадку, необхідно використовувати ряд евристик для визначення ширини діапазону, а також методик, що зменшують вплив цієї ширини.

Одним із способів, наприклад, є попереднє виключення з досліджуваного тексту слів, які за визначенням не можуть бути значущими тому, що складають вербальний «шум». Такі слова (див. пп. 1.1) називаються нейтральними або стоповими (стоп-словами). Наприклад, для російського тексту стоповими словами могли б бути всі прийменники, частки, особисті займенники.

Відомі інші способи підвищити точність оцінки значущості слів за рахунок додаткової інформації. Певні слова можуть зустрічатися майже у всіх документах деякої колекції і, відповідно, давати малий вплив на приналежність документа до тієї чи іншої категорії, тобто не бути ключовими для цього документа. Тому очевидно, що, розглядаючи всю колекцію документів, ми підвищимо інформативність пошуку ключових слів [15].

В задачах аналізу текстів та інформаційного пошуку використовується показник TF-IDF. Його можна застосовувати як один з критеріїв релевантності документа до пошукового запиту, а також при розрахунку міри спорідненості документів при кластеризації.

TF (*term frequency* – частота слова) – відношення числа входжень обраного слова до загальної кількості слів документа. Таким чином, оцінюється важливість слова t_i в межах обраного документа (1.3).

$$TF = \frac{n_i}{\sum_k n_k} \tag{1.3}$$

де n_i у чисельнику є число входжень слова в документ, а сума в знаменнику – загальна кількість слів в документі.

IDF (*inverse document frequency* – обернена частота документа) – інверсія частоти, з якою слово зустрічається в документах колекції. Використання IDF зменшує вагу широковживаних слів (1.4).

$$IDF = log\frac{|D|}{\left|(d_i \supset t_i)\right|} \tag{1.4}$$

де

|D| – кількість документів колекції;

$\left|(d_i \supset t_i)\right|$ – кількість документів, в яких зустрічається слово t_i (коли $n_i \neq 0$).

Вибір основи логарифму у формулі (1.4) не має значення, адже зміна основи призведе до зміни ваги кожного слова на постійний множник, тобто вагове співвідношення залишиться незмінним.

Коефіцієнт TF-IDF дорівнює добутку TF і IDF, в якому TF грає роль підвищувального множника, а IDF – знижувального. Тоді ваговими параметрами векторної моделі деякого документа можна прийняти коефіцієнти TF * IDF слів, що в нього входять [16].

Для того, щоб ваги знаходились в інтервалі [0,1], а вектори документів мали рівну довжину, значення TF * IDF зазвичай нормалізуються по косинусу.

Відзначимо, що ця формула оцінює значимість терміна тільки з точки зору частоти входження в документ, тим самим не враховуючи порядок проходження термінів у документі та їх синтаксичну роль. Отже, семантика документа зводиться до лексичної семантики термінів, що входять до його складу, а композиційна семантика не розглядається.

Ключовими в даному випадку будуть слова, що набрали найбільшу вагу. Слова з малою вагою взагалі можна не враховувати при класифікації.

При використанні цього підходу не виключена ймовірність попадання в ключові слова випадкових спеціальних термінів, рідкісних слів і власних імен та іншого вербального «шуму». Тому необхідно в попередню обробку тексту включати алгоритм, що підвищує якість відбору. Евристики такого відбору частіше залежать від конкретно взятого випадку.

Модель TF * IDF є, мабуть, найбільш популярною з відомих. Однак використовуються й інші індексуючі функції, включаючи ймовірні способи індексування [17] і методики індексування структурованих документів [18]. Інші функції індексації можуть знадобитися в тих випадках, коли спочатку навчальну множину не дано і документну частоту не вдається порахувати. У цих випадках TF * IDF замінюють на емпіричні функції [19].

Отже, статистичні методи мають суттєві обмеження щодо області та умов застосування, при чому супроводжуються значними обсягами «вербального шуму», фільтрація якого вимагає використання спеціальних, налаштованих на особливості текстової колекції евристик.

1.3 Актуальність та практична цінність пошуку ключових слів

Однією з вимог до наукової публікації є наявність списку ключових слів. Варто зауважити, що не тільки наукові статті повинні супроводжуватися

списком ключових слів. Ключові слова – це неодмінний атрибут запису в блозі, графічного зображення, та й будь-якої сторінки в Інтернеті.

Ключові слова є важливим елементом упорядкування масиву інформації, яким є науковий журнал або ж ціла база даних наукових статей. У ключових слів є безліч варіантів використання в процесах пошуку, класифікації та оцінки інформації.

На жаль, здебільшого, видавці демонструють слабкий інтерес до визначення ключових слів. Тому відповідна робота буває зведена до виконання формальностей з мінімальними витратами часу та праці. Але це пов'язано, здебільшого, з нерозумінням цього процесу і потенційного результату – ключові слова можуть значною мірою стати “ключем” до підвищення читацького інтересу до статей і всього періодичного видання в цілому.

У даний час в різних галузях застосовуються різні терміни для позначення ключових слів. Ключові слова (від англ. *Keyword*) – це певні слова з тексту, здатні представити найбільш значущі поняття (терміни) [20], [21], [22], [23], за якими може вестися оцінка і пошук статті.

Тег (від англ. *Tag* – мітка) – це деякий вербальний ярлик, що характеризує статтю [24]. По набору таких ярликів можна швидко зрозуміти, чим відрізняється одна стаття від іншої. Даний термін пояснює ще одне призначення ключових слів. Відмінністю тега є те, що це слово може і не зустрічатися в тексті, однак воно буде характеризувати цей текст. Первісне позначення метатег якраз і показує приставкою «мета», що це зовнішні дані, які не є частиною змісту [24].

Наприклад, стаття, що описує організацію вантажних перевезень, повинна супроводжуватися наступними ключовими словами: вантажні перевезення, організація перевезень, транспортування вантажів.

Тим не менш, для такої статті варто додати й інші слова, які допоможуть зв'язати цю статтю з публікаціями з даної тематики: логістика, транспортна логістика, логістична система.

Зазначені слова можуть не зустрічатися в тексті у зв'язку з особливостями авторського стилю, автор може просто «не любити» ці слова, або ж вони можуть бути замінені більш простими або більш складними синонімами, залежно від рівня підготовки потенційної читацької аудиторії. Тим не менше, ці слова як теги повинні бути присутніми.

Таким чином, ключові слова – це додаткові технічні дані, що дозволяють охарактеризувати статтю.

Класифікація за ключовими словами відрізняється від розподілу статей за рубриками тим, що, як правило, в різних журналах рубрики можуть бути різними. І основне тематичне спрямування одного журналу може бути всього лише підрубрикою для іншого. Ключові слова дозволяють групувати слова незалежно від журналу (періодичного видання), в якому вони опубліковані.

Ключові слова також використовуються для систематизації масиву статей, оскільки дозволяють швидше знайти статтю, групувати схожі статті, класифікувати статті у межах інших систематизованих груп [24].

Розглянемо питання – чи потрібні ключові слова для друкованої версії статті? Чи варто витрачати на них місце в друкованому журналі або обмежитися їх використанням тільки в електронній версії публікації? Ключові слова безумовно дозволяють економити час при першому знайомстві зі статтею навіть у друкованому варіанті. Вони можуть розкрити тематичну спрямованість статті, що відповідає певній рубриці журналу.

Наприклад, статтю «Інноваційне управління персоналом організації» можуть характеризувати абсолютно різні набори ключових слів. По такому набору слів: персонал, управління персоналом, оплата праці, трудовитрати ми

зрозуміємо, що в статті розглядається керування персоналом з позицій класичних підходів.

У той же час статтю зі схожою назвою може супроводжувати інший набір ключових слів: персонал, управління персоналом, інтелектуальний капітал, інтелектуально-креативні ресурси. Для дослідника, що займається цим питанням, відразу стане очевидно, що автор використовував сучасні підходи до дослідження тих же явищ.

Візуальне представлення набору ключових слів у статті все-таки відіб'ється в пам'яті читача і дозволить відразу зрозуміти, чим відрізняються близькі статті одна від одної. Отже, можна зробити висновок, що ключові слова є важливим атрибутом статті в друкованій версії, тому краще розташувати їх на початку статті під анотацією для більшої зручності читача.

Якщо взяти набір всіх ключових слів в деякому журналі і скласти просту таблицю за частотою згадування кожного окремого ключового слова, то найпопулярніші слова якраз і будуть визначати тематичну або, скоріше, термінологічну область журналу.

А якщо взяти, наприклад, 10% найпопулярніших ключових слів і розділити суму їх згадок на загальну суму згадок всіх ключових слів у журналі, то можна буде визначити глибину термінологічної спеціалізації цього журналу. У деяких випадках це дозволяє визначити, наскільки вузька тематика висвітлюється в журналі [25], [24].

У багатьох пошукових системах є можливість шукати статті за ключовими словами. Це дозволяє отримувати результати пошуку набагато швидше. При цьому внаслідок пошуку, як правило, ми зможемо отримати набір статей ближчий до заданого пошукового запиту, оскільки ключові слова відбираються кожного разу вручну саме для цілей більш точного відображення тематичної спрямованості статті.

Найбільш ефективними для пошуку будуть такі слова, які потрапляють у так звані пошукові підказки, тобто варіанти пошукового запиту, які пропонує пошукова система.

Дві статті можна порівняти щодо ступеню схожості по супроводжуючим їх ключовими словами. Чим більша кількість слів відноситься до обох статей, тим більше вони схожі. Для порівняння можна було б використовувати і повні тексти статей, але, по-перше, статті можуть бути схожими за малозначущими словами, а це знизить ефективність даного методу. По-друге, обробка повних текстів статей потребує значних обчислювальних ресурсів, що призводить до значного подорожчання будь-якої бібліотечної системи. Сучасні алгоритми визначення схожих статей все одно на першому етапі будують так звану семантичну хмару, тобто намагаються знайти ті самі ключові слова [26], [27].

Практичний сенс такого підходу полягає в тому, що для однієї, окремо взятої статті, бібліотечна система може підказати читачеві інші близькі статті з цієї тематики.

Отже, наявність правильно підібраного набору ключових слів дозволяє:

а) швидше знайти статтю користувачеві при пошуку по базі даних;

б) побачити статтю при перегляді інших схожих статей;

в) швидше зрозуміти тематичну і термінологічну область як однієї статті, так і журналу в цілому.

Все це ефективно служить одній меті: привернути увагу читачів до статті, що є основним завданням будь-якого засобу масової інформації.

Загалом ключові слова повинні відображати насамперед термінологічну область статті, зокрема:

- які терміни використовуються в статті?
- з якими термінами може бути логічно пов'язана стаття?
- з якими назвами організацій, персон, географічних областей і т.п. асоціюється стаття?

Де краще шукати ключові слова для статті – у першу чергу можуть використовуватися терміни з:

- назви статті;
- анотації до статті;
- вступу та заключної частини тексту статті [28].

Після попереднього відбору ключових слів слід оцінити їх ефективність з різних позицій. Зазвичай редактору ЗМІ доводиться працювати з уже складеним набором ключових слів. Проте варто зробити незалежну оцінку, наскільки підходять для певної статті вже підібрані слова. При визначенні якості окремого ключового слова потрібно відповісти на кілька запитань:

Чи буде хтось шукати статті за цим словом чи словосполученням? Якщо, швидше за все, ні, то таке слово не буде ефективним для пошуку, тобто не буде служити меті залучення читачів.

Чи знайдуться десь ще статті з таким же ключовим словом? Якщо ні, то таке слово буде малоефективним при відборі статей за ключовим словом. ЗМІ втрачає цей спосіб залучення читачів, оскільки навряд чи хтось буде відбирати статті за цим словом з іншої статті.

Також ЗМІ втрачає можливість того, що дана стаття буде запропонована читачеві як «схожа» поруч з якоюсь іншою статтею.

Що ще могли б шукати читачі, які шукали дане слово? По можливості такі слова також варто додати в набір ключових слів для статті, це дозволить знайти статтю читачам зі схожими інтересами. У той же час варто пам'ятати про те, що ключові слова все ж повинні відображати характер саме даної статті. Вказівка популярних, але тих, що не відповідають тематиці статті слів, викличе скоріше негативне відношення до статті і журналу (ЗМІ), в якому ця стаття опублікована.

Отже, задача пошуку ключових слів дійсно актуальна в сучасних умовах, причому після оцінки вже отриманих слів для певного тексту варто задуматися

про розширення їх кількості. Для вже знайдених та нових ключових слів слід пам'ятати про основні принципи підбору слів, практичні поради з цього приводу наведено у Додатку Б.

1.4 Аналіз методів і алгоритмів пошуку ключових слів

Тексти, написані будь-якою природною мовою складаються з двох частин: того, що обов'язково має бути виражене за законами даної мови, і того, що відображає специфіку тематики тексту і стилю автора. Ці складові називаються відповідно тематично нейтральною і тематично маркованою лексикою. Пошук обох груп лексики – крок на шляху до визначення змістовної віднесеності тексту. Даний процес дозволяє як наблизитися до змісту тексту, так і скласти певну думку про своєрідність лексики автора і його мови.

Лексичні одиниці, що зустрічаються в тексті, мають свої частотні характеристики. Чим яскравіші кількісні особливості деяких словоформ у порівнянні з іншими у всьому тексті, тим вища їх тематична маркованість.

Пошук тематично маркованої та тематично нейтральної лексики проводиться методом системного зважування слів за двома функціональними параметрами: прямому (частотному – Q-параметр) і непрямому (довжина слова – F-параметр).

Традиційним способом виявлення тематично маркованої лексики є частота словоформ (або їх множин, що відносяться до одного слова-леми). При цьому передбачається, що чим частіше вживається слово в тексті, тим воно важливіше для його змісту. Це справедливо лише частково: службові слова зазвичай мають більшу частоту, але, тим не менш, специфіки тексту не відображають [28].

Необхідно враховувати і ще одну обставину. Встановлено залежність, що існує між частотою слова (словоформи) і його довжиною (див. пп.1.2): чим

частіше вживається слово, тим воно коротше і навпаки. Але для цього потрібно, щоб слова стійко і тривало були частотними. Отже, якщо коротке слово в тексті є частотним, це буде характеризувати його як коротке слово, але якщо довге слово в тексті буде мати частоту, незвичайну для слів такої довжини, то воно буде відображати специфіку даного тексту. Таким чином, для пошуку тематично маркованої лексики в даному тексті недостатньо інформації про їх довжину і частоту, потрібно співвіднести обидва типи інформації, а для цього зробити дані по довжині та частоті порівнянними.

З цією метою однотипово обчислюються функціональні ваги словоформ: прямий (частотний – *Q*-вага) і непрямий (довжина слова в звуках – *F*-вага). Q-параметр характеризує функціонування слова в цьому тексті. Для його підрахунку після побудови частотного словника вихідного тексту слова розташовуються в порядку зменшення абсолютної частоти. Визначення рангів здійснюється для групи слів з однаковою частотою. Слова з найбільшою частотою отримують ранг 1. При зменшенні частоти слова в тексті ранг збільшується на 1. Таким чином, слова, що мають однакову абсолютну частоту, отримують однаковий ранг.

F-параметр обчислюється за формулою [29]:

$$F_{ri} = \frac{\sum r - R_{i-1}}{\sum r}, \qquad (1.5)$$

де $\sum r$ – сума одиниць усіх рангів.

R_{i-1} – сума одиниць від першого до даного рангу (виключно).

При цьому мова йде про ранги частот, а не про ранги слів.

Довжина слова в звуках – *F*-параметр – характеризує функціонування слова на тривалому відрізку часу, настільки тривалому, щоб функціонування

встигло вплинути на довжину слова. Для його підрахунку після побудови частотного словника вихідного тексту слова розташовуються в порядку зменшення їх довжини в звуках. Визначення рангів здійснюється для групи слів з однаковою довжиною в звуках. Слова з найбільшою довжиною отримують ранг 1. При зменшенні довжини слова ранг збільшується на 1. Таким чином, слова, що мають однакову довжину в звуках, отримують однаковий ранг.

Вводиться індекс тематичної маркування слова (*ІнТеМ*), який обчислюється за формулою *ІнТеМ = Q-вага – F-вага*, де *Q-вага* – вага слова за частотою, *F-вага* – вага слова за довжиною.

Позитивні значення *ІнТеМ* будуть характеризувати тематично марковану лексику, негативні значення – тематично нейтральну лексику. При цьому значення *ІнТеМ* будуть варіюватися в інтервалі від –1 до +1 [30].

Розглянемо алгоритм пошуку тематично маркованої лексики:

а) підраховуємо для кожного слова його абсолютну частоту в тексті і довжину в звуках.

б) призначаємо ранги частот і підраховуємо для кожного слова *Q*.

в) призначаємо ранги для довжин слів в звуках і підраховуємо для кожного слова *F*.

г) обчислюємо для кожного слова *ІнТеМ*.

д) якщо отриманий *ІнТеМ* позитивний, слово відноситься до тематично маркованої лексики, негативний - до тематично нейтральній лексики [31].

Ще одним підходом є виявлення семантичного поля для знаходження ключових слів:

а) знаходимо в тексті *T* для кожної словоформи її частоту – абсолютну і відносну [32] (частка від ділення спостережуваної, абсолютної частоти на кількість словоформ в тексті *T*).

б) знаходимо в *T* ті частини цього тексту, які містять задане слово *C* – речення з цим словом. Для сукупності всіх цих речень *T** побудуємо частотний

словник *V(T*)*, який містить абсолютну і відносну (частка від ділення спостережуваної, абсолютної частоти на кількість слів у *T**) частоти.

в) порівнюємо отримані відносні частоти в *T* і *T**: індекс значимості = відносна частота в *T** / відносна частота в *T* [33].

Якщо отриманий індекс більше одиниці, то словоформу із заданими характеристиками відносимо до семантичного поля слова *C*.

При обробці текстового файлу виникає цілий ряд складнощів, тому існують певні вимоги до вихідних текстів.

У більшості випадків ці вимоги визначаються нормами поліграфії та такими емпіричними міркуваннями:

- вважається, що текст побудований правильно (наприклад, дві коми не можуть йти підряд);
- дефіс відразу після слова в кінці рядка говорить про перенесення слова, а після нього в рядку не можуть йти ніякі символи, у тому числі пропуск;
- тире після пропуску є поодиноким символом і праворуч від нього також повинен бути пропуск;
- тире після символу-літери при наявності наступної літери, розташованої в тому ж рядку, говорить про словосполучення і ніколи не відділяється пропусками.

Прийняття таких правил полегшує обробку текстів, але ускладнює їх набір на комп'ютері, оскільки вимагає постійного контролю матеріалу, що вводиться [34].

Розглянемо інші відомі підходи до пошуку ключових слів. Зокрема одна з перших систем для автоматизованого пошуку термінології LEXTER [35] знаходить терміни з корпусу технічних текстів французькою для подальшої обробки експертом. Перший етап роботи системи – визначення максимальних ланцюжків, що містять терміни. Ці ланцюжки визначаються негативно:

складається список слів і знаків, які не можуть входити до складу терміну, тоді рядки між цими роздільниками розглядаються як кандидати в терміни [36].

Метод k-factor, запропонований в роботі [37], було реалізовано в системі BootCaT, яка призначена для автоматичного формування тематичного корпусу з WEB. Побудова корпусу починається з набору вихідних термінів (seed terms). За допомогою автоматичних запитів, пошукова машина знаходить документи, що містять вихідні терміни; у свою чергу, з цих документів знаходять нові однослівні терміни (на основі порівняння частот у сформованому корпусі зі «стандартним корпусом»), які знову можна використовувати в якості запитів і т.д. Фінальний корпус і список однослівних термінів використовується для ітеративного пошуку багатослівних термінів. Метод можна розглядати як спрощений варіант методу C-value: якщо коротший термін-кандидат зустрічається не набагато частіше, ніж довший термін-кандидат, в який він повністю входить, то «основним» вважається більш довгий варіант. Відбором управляє порогове значення відношення частот термінів k.

Метод пошуку багатослівних термінів, запропонований Frantzi et al. [38], заохочує словосполучення, що не входять до складу інших, більш довгих. Зустрічальність довгих термінів у тексті нижче, ніж коротких, і метод C-value був запропонований для компенсації цього ефекту. Значення термінологічності розраховується так:

$$C - Value(a) = \begin{cases} log_2|a| * freq(a), \text{ якщо не вкладений} \\ log_2|a| * freq(a) - \frac{1}{P(T_a)} * \sum_{b \in T_a} freq(b) \end{cases} \quad (1.6)$$

де a – кандидат в ключові слова,

$|a|$ – довжина словосполучення, вимірюється в кількості слів,

$freq(a)$ – частотність a,

T_a – множина словосполучень, які містять a,

$P(T_a)$ – кількість словосполучень, які містять a.

Легко побачити, що чим більше частота терміна-кандидата і його довжина, тим більше його вага. Але якщо цей кандидат входить у велику кількість інших словосполучень, то його вага зменшується.

Метод, описаний в [39], позначимо Window (в оригінальній статті він позначений TERMC). Ідея методу близька двом попереднім (C-value, k-factor) – нарощувати словосполучення, якщо більш короткі часто зустрічаються у складі більш довгих. Однак, на відміну від інших методів, враховується не тільки частота контактних випадків (слова безпосередньо слідують один за одним), а й спільна зустрічальність у вікні.

На кожній ітерації для кожного елементу списку запам'ятовується його безпосередні сусіди і сусіди в текстовому вікні. Створюються відповідні таблиці, обчислюється частотність зустрічальності пар. Далі, передбачається, що якщо пара елементів (на першому етапі – окремих слів) зустрічається як безпосередні сусіди більш ніж у половині випадків їх появи в одному і тому ж текстовому вікні, то ця пара являє собою термін або фрагмент терміна. Відбувається склейка пари в єдиний елемент, таблиці перераховуються так, як якби цей елемент був відомий з самого початку, до початку обробки тексту, що дає можливість і далі нарощувати термін. Автори наводять приклади довгих термінів, отриманих цим методом: закон про обов'язкове страхування цивільної відповідальності власників транспортних засобів, виконавчий орган місцевого самоврядування. Якщо не накладати обмежень на частоту зустрічальності склеюваних елементів, то метод об'єднає унікальні (з частотою 1) ланцюжки допустимих слів (тобто повторить результат MaxLen) [40].

Відомо, що більшість термінів – це іменні групи (хоча в [41], наприклад, показано, що номінативність не є винятковою характеристикою термінів у багатьох предметних областях). У межах запропонованого у [42] методу в якості термінів-кандидатів розглядаються іменні групи, знайдені за допомогою синтаксичного аналізатора. Метод отримав назву по використовуваному

аналізатору – АОТ. Після первинного аналізу список проріджується шляхом виключення груп з однорідними рядами (з комами, спілками та / або). Отримані рядки перетворюють так, щоб головне слово іменної групи було словникової форми, іншої обробки не проводилось.

Метод на основі АОТ – єдиний з розглянутих, який допускає наявність прийменників у складі кандидатів у ключові слова [43].

Важливим завданням обробки текстових документів, результат рішення якої використовується не тільки для їх класифікації (категоризації) [44], але і для вилучення з них інформації і знань, є координатне індексування, тобто пошуку в текстах документів ключових слів і словосполучень.

Класичний підхід до вирішення даної проблеми полягає у використанні засобів аналізу на основі тезауруса певної предметної області [45]. Але метод пошуку ключових слів і словосполучень, заснований на аналізі тезауруса предметної області [46], має істотний недолік: таким способом не можна виробляти індексування корпусів текстів довільних тематик. Більш того, якщо вести мову про обробку корпусів текстів досить вузьких тематик, то в таких випадках потрібні дуже докладні тезауруси, які є (принаймні, в широкому доступі) далеко не для всіх предметних областей. Підхід же, заснований на виявленні ключових виразів без апріорних обмежень, носить набагато більш універсальний характер, хоча, природно, дещо програє в адекватності індексування.

З огляду на те, що в українській і російській мові іменники і прикметники при відмінюванні змінюють свою форму, розробка ефективного алгоритму автоматизації пошуку ключових слів є нетривіальним завданням, оскільки необхідно враховувати і ті випадки, коли слова, що утворюють термін (тобто ключове слово), знаходяться не тільки в називному, а й в непрямих відмінках.

Для вирішення цього завдання спираються на морфологічний аналіз текстів і пошук ключових словосполучень за морфологічними шаблонами з

використанням програмного продукту компанії Яндекс, який є безкоштовним для некомерційних цілей. При фільтрації і розборі проводиться відсів стоп-слів. Ключові словосполучення відбираються за морфологічними шаблонами з урахуванням словоформ мови.

Для пошуку ключових словосполучень використовуються класичні морфологічні шаблони, які досить якісно визначають шукані ключові вирази:

- (Дієприкметник) (Іменник);
- (Прикметник) (Іменник);
- (Іменник) (Іменник в орудному відмінку);
- (Іменник) (Іменник в родовому відмінку).

Після завершення підрахунку входжень ключових слів і словосполучень в документі необхідно зробити список найбільш значущих слів, що відображають контекстний зміст корпусу. Кількість входжень слів в текст в більшості випадків піддається закону розподілу частот Ципфа (пп.1.2): якщо всі слова впорядкувати за зменшенням частоти їх використання, то частота n-го слова в цьому списку опиниться приблизно обернено пропорційною його порядковому номеру (рангу). Для побудови списку одиночних ключових слів зазвичай використовується саме закон Ципфа [47], [48].

Проте цей закон не працює для частоти розподілу ключових словосполучень.

Для обмеження числа складових ключових фраз, що найбільш точно описують зміст електронного документа, використовується наступна закономірність, помічена емпіричним шляхом, яка перевірялася на досить великій кількості корпусів текстів середньої і великої величини:

$$KeyPhrase(i) = \frac{max(Frequency)}{Frequency(i)} < \frac{word\_num}{3}, \qquad (1.7)$$

де max (Frequency) – максимальна частота 1-го терма (тобто, терма, що зустрічається найчастіше і всіх його словоформ в корпусі текстів); Frequency (i) – частота і-го терма, що перевіряється; word_num – бажана (орієнтовно) кількість відібраних термів.

Зрозуміло, що дана умова (як і закон Ципфа) погано працює на документах невеликого розміру (типу анотацій), оскільки в них частоти всіх однослівних і багатослівних ключових термінів приблизно рівні і прагнуть до одиничного входженню в рамках контексту документа [49], [50], [51]. Ці та інші розглянуті обмеження методів, що побудовані на основі закону Ципфа, якраз і є причиною недостатньої якості пошуку ключових слів. Тому підвищення якості процесу пошуку ключових слів вимагає залучення додаткової інформації, бажано універсального, а не специфічного характеру. Зокрема, не враховуються у відомих методах пошуку ключових слів результати аналізу зв’язків між лексичними одиницями тексту.

1.5 Аналіз підходів до пошуку ключових слів в умовах Web

На сьогоднішній день однією з найважливіших і помітних областей Web, ключовим принципом якої є участь користувачів в роботі сайтів, вважаються мережеві щоденники або веб-логи, скорочено назва – блоги. Перший і найбільш відомий сервіс мікроблогів Twitter був запущений в жовтні 2006 року компанією Obvious з Сан-Франциско. До теперішнього часу постійно зростаюча аудиторія сервісу становить десятки мільйонів людей. Очевидно, що автоматизований пошук найбільш значущих термінів (слів) з потоку повідомлень, що генерується співтовариством Twitter, має практичне значення як для визначення інтересів різних груп користувачів, так і для побудови індивідуального профілю кожного з них.

Виникнення та подальший розвиток ідеї створення сервісу мікроблогів сучасні дослідники Інтернету цілком обґрунтовано вважають результатом

процесу інтеграції концепції соціальних мереж з мережевими щоденниками – блогами [52].

За визначенням Walker в [53], блогами прийнято вважати часто оновлювані веб-сайти, які складаються з записів, що містять різну інформацію, розміщених у зворотному хронологічному порядку.

Характерні риси блогінгу можуть бути описані з використанням 3 ключових принципів [54]:

- вміст блогів являє собою короткі повідомлення;
- повідомлення мають спільне авторство і знаходяться під контролем автора;
- можлива агрегація множини потоків повідомлень від різних авторів для зручного читання.

Ці принципи застосовні також і для мікроблогінгу [55]. Однак у той час, як для блогінгу публікація і агрегація повідомлень є завданнями для різних програмних продуктів, сервіси мікроблогінгу надають всі перераховані можливості відразу.

Крім того, мікроблогінг враховує потребу користувачів у більш швидкому режимі комунікації, ніж при звичайному блогінгу. Заохочуючи більш короткі повідомлення, він зменшує час і роботу, необхідні для створення контенту. Це також є однією з головних, відмінних від блогінгу ознак. Інша відмінність полягає в частоті оновлень. Автор звичайного блогу оновлює його, в середньому, раз на кілька днів, у той час як автор мікроблога може оновлювати його кілька разів на день.

Основні функції найбільш популярного на сьогоднішній день сервісу мікроблогів Twitter являють собою дуже просту модель. Користувачі можуть відправляти короткі повідомлення, або твіти, довжиною не більше 140 символів. Повідомлення відображаються як потік на сторінці користувача. У термінах соціальних зв'язків, Twitter дає можливість користувачам слідувати

(follow) за будь-яким числом інших користувачів, званих друзями. Мережа контактів Twitter асиметрична, тобто якщо один користувач слідує за іншим, то цей інший не зобов'язаний слідувати за ним. Користувачі, які слідкують за іншим користувачем, називаються його послідовниками (followers).

Користувачі мають можливість вказати, чи бажають вони, щоб їх твіти були доступні публічно (з'являються у зворотному хронологічному порядку на головній сторінці сервісу і на сторінці самого користувача) або приватно (тільки послідовники користувача можуть бачити його повідомлення). За замовчуванням, всі повідомлення доступні будь-якому користувачеві.

Для полегшення розуміння внутрішнього устрою Twitter доцільно ввести 2 концепції: елемент як окреме повідомлення і канал як потік елементів, що найчастіше належать одному користувачеві [54].

Основною частиною мікро синтаксису є слештеги, кожен з яких складається з символу «/» і покажчика, який і визначає сенс слеш тега.

Ці елементи використовують для різних цілей. Наприклад, «/ via» - для посилання на автора повідомлення при «ретвіті»; «/ By» - для посилання на автора вихідного повідомлення, якщо воно є результатом ланцюжка «ретвітів»; «/ cc» - для вказівки тих передплатників мікроблога, яким в першу чергу адресоване повідомлення і т.д.

Використання слештегів повністю є ініціативою самих користувачів, тому не існує єдиних правил щодо їх використання.

Наведений опис відображає найбільш популярні способи застосування слештегів до цього моменту. Існують, однак, і інші рекомендації, які теж заслуговують на увагу, тому кожен користувач застосовує елементи мікросинтаксиса на свій розсуд.

Наприклад, можливе об'єднання всіх використаних в повідомленні слештегів без символу «/» в одну групу, обмежену дужками. Деякі користувачі

розміщують слештеги строго в кінці повідомлення, ставлячи попереду лише символом «/» з метою економії символів.

Одним із завдань добування інформації з тексту є пошук ключових термінів, що з певною мірою достовірності відображають тематичну спрямованість документа. Автоматичний пошук важливих тематичних термінів у документі є однією з підзадач більш спільного завдання – автоматичної генерації ключових термінів, для якої знайдені ключові терміни не обов'язково повинні бути присутніми в даному документі [56].

В останні роки було створено багато підходів, які дозволяють проводити аналіз наборів документів різного розміру і знаходити ключові терміни, що складаються з одного, двох і більше слів.

Найважливішим етапом пошуку ключових термінів є розрахунок їх ваг в аналізованому документі, що дозволяє оцінити їх значущість відносно один одного в даному контексті [57]. Для вирішення цього завдання існує значна кількість підходів, які умовно діляться на 2 групи: вимагають навчання і не потребують навчання. Під навчанням мається на увазі необхідність попередньої обробки вихідного корпусу текстів з метою пошуку інформації про частоту зустрічальності термінів у всьому корпусі [58], [59], [60], [61], [62]. Іншими словами, для визначення значущості терміна у цьому документі необхідно спочатку проаналізувати всю колекцію документів, до якої він належить. Альтернативним підходом є використання лінгвістичних онтологій, які є більш-менш наближеними моделями існуючого набору слів заданого мови [45]. На базі обох підходів були створені системи для автоматичного пошуку ключових термінів, однак у цьому напрямку постійно ведуться роботи з метою підвищення точності і повноти результатів, а також з метою використання методів пошуку інформації в тексті для вирішення нових завдань [63], [64], [65], [66], [67], [68], [69].

Найпоширенішими схемами для розрахунку ваг термінів є TF-IDF і різні його варіанти (пп.1.2), а також деякі інші (ATC, Okapi, LTU). Однак загальною рисою цих схем є те, що вони вимагають наявності інформації, отриманої з усієї колекції документів. Іншими словами, якщо метод, заснований на TF-IDF, використовується для створення уявлення про документ, то надходження нового документа в колекцію зажадає перерахунку ваг термінів у всіх документах[70]. Отже, будь-які додатки, засновані на значеннях ваг термінів у документі, також будуть зачеплені. Це значною мірою перешкоджає використанню методів пошуку ключових термінів, що вимагають навчання, в системах, де динамічні потоки даних повинні оброблятися в реальному часу, наприклад, для обробки повідомлень мікроблогів [71].

Для вирішення цієї проблеми було запропоновано декілька підходів, таких як алгоритм TF-ICF [72]. Як розвиток цієї ідеї Mihalcea і Csomai в 2007 році запропонували використовувати в якості навчального тезауруса Вікіпедію [73]. Вони застосували для розрахунків інформацію, що міститься в анотованих статтях енциклопедії з вручну знайденими ключовими термінами. Для оцінки ймовірності того, що термін буде обраний ключовим у новому документі, використовується формула:

$$P(W) \approx \frac{\text{число}(D_{\text{ключове}})}{\text{число}(D_W)}, \qquad (1.8)$$

де W – термін;

$D_{\text{ключове}}$ – документ, в якому термін був обраний ключовим;

D_W – документ, в якому термін з'явився хоча б один раз.

Ця оцінка була названа авторами keyphraseness (в даній роботі визначена як інформативність). Вона може бути інтерпретована наступним чином: чим частіше термін був обраний ключовим з числа його загальної кількості появ, тим з більшою ймовірністю він буде обраний таким знову.

Інформативність може приймати значення від 0 до 1. Чим вона вища, тим вище значимість терміна в аналізованому контексті. Наприклад, для терміна «Of course» у Вікіпедії існує тільки одна стаття, присвячена пісні американського виконавця, тому він рідко вибирається ключовим, хоча зустрічається в тексті дуже часто. Значення його інформативності, таким чином, буде близько до 0. Навпаки, термін «Microsoft» в тексті будь-якої статті майже завжди буде позначиним як ключове, що наближає його інформативність до 1.

Даний підхід є досить точним, оскільки всі статті у Вікіпедії вручну анотуються ключовими термінами, тому запропонована оцінка їх реальної інформативності є результатом узагальнення думок людей.

Разом з тим, ця оцінка може бути ненадійною в тих випадках, коли застосовувані для розрахунків значення занадто малі. Для вирішення цієї проблеми рекомендується розглядати тільки ті терміни, які з'являються у Вікіпедії не менше 5 разів.

Для розрахунку ваги терміна w_i використовувалася формула:

$$w_i = TF_i K_i, \tag{1.9}$$

де i – порядковий номер терміна;

TF_i – частота терміна в аналізованому повідомленні;

K_i – інформативність терміна за даними Вікіпедії.

TF означає частоту терміна (Term Frequency). Значення цього компонента формули дорівнює відношенню числа входження деякого терміна до загальної кількістю термінів повідомлення. Таким чином, оцінюється важливість терміна t_i в межах окремого повідомлення.

Одним з найважливіших етапів розробки системи стала обробка XML-дампа всіх статей англійської Вікіпедії. Метою аналізу був розрахунок інформативності для всіх термінів Вікіпедії за формулою (1.9).

Потрібно відзначити, що для однієї концепції в словнику Вікіпедії може бути кілька синонімів. Наприклад, термін «IBM» має кілька синонімів: «International Business Machines», «Big Blue» і т.д. Так як в розробленій системі відсутній етап дозволу лексичної багатозначності термінів, то було неприпустимо, щоб синоніми мали різні значення інформативності. Тому було прийнято вважати, що інформативність всіх синонімів однієї концепції стає однаковою, виходячи із загальної статистики для всіх них.

Крім того, згідно з рекомендаціями авторів методики розрахунку інформативності [73], були виключені терміни, які були знайдені менш ніж у 5 статтях. Якщо пропустити цей крок, то результуюче значення часто стає недостовірним і не дозволяє коректно оцінити відносну значущість терміна в контексті.

Для отримання інформації про повідомлення користувача з сервера Twitter використовувався Perl-модуль Net :: Twitter. Результатом взаємодії з Twitter API є отримання деякого числа останніх повідомлень його аккаунта, інакше званих timeline.

Для вирішення поставленого завдання був обраний метод statuses / friends timeline, що повертає повідомлення друзів користувача. Для тестування були створені акаунти, підписані на поновлення статусу єдиного друга.

При цьому в самому акаунті не публікувалася ніяких повідомлень. Таким чином, результат виклику даного методу містить лише необхідну кількість повідомлень одного користувача Twitter, що й потрібно в якості вихідних даних.

Однак треба зазначити, що класичні статистичні методи пошуку ключових термінів, засновані на аналізі колекцій документів, малоефективні в даному випадку. Це обумовлено надзвичайно малою довжиною повідомлень (до 140 символів), їх різноманітною тематикою і відсутністю логічного зв'язку між собою, а також великою кількістю рідко використовуваних абревіатур, скорочень та елементів специфічного мікросинтаксису.

Для вирішення цієї проблеми відносна значимість термінів в аналізованому контексті визначається за допомогою даних про частоту їх використання в якості ключових в інтернет-енциклопедії Вікіпедія. Робота алгоритму заснована на розрахунку "інформативності" кожного терміна, тобто оцінки ймовірності того, що він може бути обраний ключовим в тексті. Далі до аналізованого набору термінів застосовується ряд евристик, результатом яких є список термінів, визнаний ключовими.

У процесі попередньої обробки тексту, або препроцесинга, вміст отриманих з сервера Twitter повідомлень перетвориться до формату вхідних даних для алгоритму пошуку ключових термінів.

Крім стандартних для цього етапу операцій, проводиться також видалення слештегов і @-ссилок, а також службових слів «RT» та «RETWEET», тобто тих елементів специфічного синтаксису Twitter, які не несуть смислового навантаження і є стоп-словами в термінах обробки природної мови. На цьому етапі також проводиться пошук хештегів, які в подальшому обробляються однаково з іншими словами.

Очевидно, що для складання списку кандидатів у ключові терміни недостатньо лише вихідної послідовності слів. Виходячи з невеликої довжини повідомлень, представляється можливим зробити алгоритм максимально надлишковим з метою підвищення повноти результатів, тобто не пропустити ні один можливий ключовий термін, зі скількох би слів він не складався. З цією метою проводиться пошук усіх можливих N-грам, тобто послідовностей слів, що йдуть один за одним. Кількість N-грам Q_k, які можна отримати з k слів (включаючи N-грама з одного слова), визначається таким чином:

$$Q_k = \sum_{i=1}^{k} i . \qquad (1.10)$$

Всі отримані на цьому етапі терміни додаються в масив можливих ключових термінів. Завершальним етапом препроцесинга є стоп-лістинг, тобто видалення з отриманого масиву тих слів, які не несуть суттєвого смислового навантаження. Важливо відзначити, що стоп-лістинг виконується лише після знаходження N-грам. Таким чином, стоп-слова можуть входити до складу термінів, але видаляються зі списку кандидатів, якщо зустрічаються в ньому самі по собі. На цьому етапі використовується стоп-лист системи SMART [74].

На етапі розрахунку ваг термінів-кандидатів для кожного з них запитується значення інформативності з бази даних. Другою необхідною для розрахунків показником є частота зустрічальності терміна TF. Для кожного знайденого в БД терміна його вага розраховується за формулою (1.10).

Загальний принцип пошуку ключових термінів полягає в аналізі заданого числа повідомлень і визначенні порогового значення ваги для кожного з них. Ті терміни, ваги яких більше або дорівнюють пороговому значенню, вважаються ключовими [75], [76], [77], [78].

Спочатку пороговим вважається середнє арифметичне значення ваги для всіх термінів-кандидатів. Усі наступні операції покликані уточнити його і поліпшити результати роботи алгоритму в цілому.

Наступним етапом є обробка масиву хештегів, отриманого на етапі препроцесинга. Передбачається, що з їх допомогою користувач явно вказує терміни, що визначають тематику повідомлення. Тому логічно припустити, що граничне значення для всього повідомлення не повинно бути вище мінімальної ваги серед його хештегів. Ґрунтуючись на цьому припущенні, вага кожного з хештегів (якщо він був знайдений в БД) порівнюється з поточним пороговим значенням і знижує його у випадку, якщо воно більше.

Якщо після обробки хештегів [79] порогове значення залишилося рівним 0 (у випадку, коли вони не були вказані або коли жоден з них не був знайдений в БД) або перевищує знайдене середнє значення, воно приймається рівним

середньому. Така ситуація на практиці зустрічається найчастіше, оскільки не часто користувачі явно вказують тематичні терміни. Однак такий підхід істотно покращує результати роботи.

Потрібно відзначити, що повідомлення обробляються в порядку, зворотному надходженню з сервера, тобто в прямому хронологічному. Такий підхід представляється логічним і враховує специфіку сервісу блогів [80] в цілому: користувач може написати повідомлення на якусь тему, а потім повернутися до неї знову. Однак у другому повідомленні, крім тих термінів, які були обрані ключовими у першому, можуть бути інші, більш інформативні терміни, за рахунок яких порогове значення для другого повідомлення буде завищено, і ключові терміни з першого повідомлення не будуть знайдені. Щоб уникнути цього, в системі є окремий масив, що містить всі раніше знайдені ключові терміни. Тоді при обробці чергового повідомлення терміни з цього масиву будуть безумовно знайдені і знизять порогове значення ваги для даного повідомлення [81].

У цьому контексті важливо, що при безпосередньому виборі ключових термінів кандидати обробляються в порядку зростання їх ваг. Таким чином, якщо вага якого-небудь з кандидатів нижче порогового, але він вибирається ключовим через те, що присутній в масиві раніше знайдених термінів, то порогове значення стає рівним його вазі, і всі наступні терміни автоматично потрапляють в список ключових.

Результатом роботи алгоритму є список відсортованих в порядку зменшення ваг ключових термінів [82], тому вибір найбільш інформативної ознаки для визначення ваги має вирішальне значення і в умовах WEB.

1.6 Системи автоматичного пошуку ключових слів

Розглянемо найбільш відомі системи, у першу чергу, OpenCalais – Web-сервіс, призначений для автоматичного пошуку семантичних метаданих в тексті на природній мові. Починаючи з 2007 року, розвитком і підтримкою сервісу займається корпорація Thomson Reuters.

Семантичні метадані представлені в системі у вигляді іменованих сутностей (англ. *Named entity*), а також пов'язаних з ними фактів і подій. Іменовані сутності, в свою чергу, можуть розглядатися як ключові слова і фрази вихідного тексту.

Функціонування системи OpenCalais засноване на методах обробки природної мови, машинного навчання та інших алгоритмах. Для пошуку семантичних метаданих застосовуються попередньо підготовлені онтології різних предметних областей [83], [84], [85] у форматі RDF. Оригінальний текст піддається попередній обробці (графометрична та морфологічна розмітка), потім розмічені словосполучення проходять ідентифікацію за допомогою навченої моделі розпізнавання іменованих сутностей, між якими ведеться пошук семантичних відношень. Отриманий граф сутностей і відношень між ними перетвориться в набір RDF-трійок.

Web-сервіс OpenCalais безкоштовний і доступний для некомерційного та комерційного використання, однак вимагає реєстрації для отримання API ключа.

Сервіс побудований за архітектурі REST і використовує формат XML для обміну даними з користувачами.

Інша широко відома система автоматичного пошуку термінів Extractor функціонує з 2002 року і використовується багатьма організаціями в своїх рішеннях по обробці природної мови.

Робота системи Extractor заснована на застосуванні генетичних алгоритмів в поєднанні з методами машинного навчання і статистичними методами обробки природної мови. Початкове навчання системи ведеться на основі розміченого корпусу текстів.

Ознайомитися з можливостями Extractor можна за допомогою демонстраційного Web-додатку ExtractorLive.com. Для створення власних рішень на основі технології Extractor потрібно придбати набір інструментів розробника.

Система Yahoo! Term Extraction Web Service – це сервіс, який використовується в пошуковій системі Yahoo! Search і призначений для пошуку ключових фраз в тексті на природній мові. Документація Yahoo! Term Extraction Web Service використовує закриту технологію пошуку термінів. Обмін даними з користувачем здійснюється у форматах XML та JSON.

TerMine – Web-сервіс пошуку термінів, розроблений в британському Національному центрі аналізу тексту (англ. *The National Centre for Text Mining*).

Сервіс TerMine працює на основі методу C-value і застосовує аналізатор TreeTagger для попередньої морфологічної розмітки тексту.

Демонстраційний Web-інтерфейс TerMine дозволяє обробляти тексти виключно англійською мовою.

Maui – система тематичної класифікації текстових документів, що працює на основі методів обробки природної мови і машинного навчання.

Схема функціонування системи складається з двох етапів роботи: етапу первісного побудови та навчання моделі і етапу застосування навченої моделі до вирішення завдання тематичної класифікації тексту.

Результати тематичної класифікації, отримані за допомогою Maui, можуть розглядатися в якості тегів (міток) вихідного тексту. Без використання навченої моделі Maui функціонує як система автоматичного пошуку ключових фраз.

Система Maui є вільним кросплатформним програмним забезпеченням і розповсюджується на умовах ліцензії GNU General Public License Version 3. За допомогою Web-додатки maui-indexer можна ознайомитися з основними можливостями системи [86].

TextAnalyst – інструмент для пошуку інформації та аналізу змісту текстів, має можливість пошуку ключових слів.

Функціонування TextAnalyst засноване на застосуванні методів обробки природної мови в поєднанні з методами машинного навчання. Пакет TextAnalyst доступний для ознайомчого використання у вигляді додатку для сімейства операційних систем Windows.

АОТ – проект, спрямований на створення системи автоматичного перекладу «ДІАЛІНГ». В рамках проекту АОТ розроблений комплекс інструментів автоматичної обробки тексту, в тому числі графематичний, морфологічний і синтаксичний аналізатори російської мови.

Всі інструменти, розроблені в рамках проекту АОТ (у тому числі і синтаксичний аналізатор) є вільним кросплатформним програмним забезпеченням і поширюються на умовах ліцензії GNU Lesser General Public License Version 2.1.

ContentAnalyzer – інструмент для аналізу змісту тематичних Web-сторінок у реальному часі, пошуку списків ключових слів і словосполучень, побудови автореферату тексту документа.

Функціонування ContentAnalyzer забезпечується за рахунок обчислення характеристик тексту:

- частота терміна / словосполучення в документі;
- відношення частоти до числа слів документа;
- вага терміна в документі (з урахуванням частоти і вагових коефіцієнтів);
- вага терміна до числа слів документа.

Пакет ContentAnalyzer розповсюджується безкоштовно, доступний для використання в вигляді додатку для сімейства операційних систем Windows.

Семантичне дзеркало – ще одна система тематичної класифікації Web-сторінок, яка розроблена компанією «Ашманов і партнери».

Сервіс «Семантичне дзеркало» обробляє текст Web-сторінки і визначає її тему: аналізує слова, семантичні зв'язки між ними, знаходить найважливіші терміни. Теми визначаються за рубрикатором, де до кожної рубриці приписано деяка множина термінів.

Результат тематичної класифікації можна використовувати в якості списку ключових фраз вихідного тексту або в якості набору тегів (міток). Цю інформацію можна використовувати для показу контекстної реклами та новин на актуальну тему.

На сайті компанії «Ашманов і партнери» доступна демонстраційна версія «Семантичного дзеркала», що має ліміт у 128 звернень з однієї IP-адреси на добу [87].

Seotool – безкоштовний онлайн сервіс, що забезпечує перевірку: чи є написаний текст релевантним списку ключових слів (система автоматично генерує ключі за вказаний текстом). Це сприяє отриманню більш високого рейтингу в пошукових системах Яндекс і Google, так як сторінка матиме ключові слова, відповідні змісту сторінки, на якій розміщені. Також даний сервіс допоможе (генерує семантичне ядро) при складанні семантичного ядра сайту (при включеному режимі потрібно прибрати HTML код). Але при генерації ключових слів і фраз використовуються тільки перші тисячу слів введеного тексту.

У системі присутня можливість відсоткового порівняння слів із шаблоном. Слова аналізованого тексту (контенту), будуть у відсотковому співвідношенні порівнюватися зі списком слів всього шаблону (тексту) із застосуванням морфологічного аналізу цих слів. Певне слово враховується при

відповідності відсоткової рівності з будь-яким зі слів шаблону, інакше – не враховується. Максимальна кількість слів шаблону не повинно перевищувати 250 слів [88].

Ресурс Rise-Top застосовується для складення "начерків" ключових слів для сайту, використовуючи для аналізу вказаний текст. В якості відбору ключових слів використовуються слова з найбільш високою щільністю в порядку зменшення їх щільності до всього тексту [89]. У генерації ключових слів використовуються тільки перші 1000 слів обробленого тексту.

Advego (Адвего) – найбільший в Рунеті постачальник контенту і супутніх послуг для інтернет-сайтів. Ресурс, призначений для оптимізаторів і власників сайтів, підтримує такий функціонал: унікальні статті, відгуки, публікації, просування в пошукових системах та розкрутка в соцмережах. Має можливість пошуку ключових слів [90].

Отже, актуальність напряму досліджень у розвитку методів та засобів автоматизованого пошуку ключових слів стало причиною появи на даний час досить популярних систем та ресурсів, що є інструментами розкрутки та підвищення рейтингу сайтів. Тому є всі підстави вважати, що удосконалення відповідних методів з метою підвищення якості пошуку ключових слів має необхідну інструментальну підтримку і може бути швидко та ефективно впроваджено у SEO-індустрію.

1.7 Вибір напрямку і обґрунтування задач досліджень

За результатами проведеного аналізу можна зробити висновок про високу актуальність розглянутого напряму досліджень, оскільки якість пошуку ключових слів тексту безпосередньо впливає на релевантність результатів пошуку інформації в Інтернеті. Практична цінність обраного напряму досліджень полягає у тому, що отримані ключові слова можна використати не

тільки як вхідні дані для пошукових машин, але й для підвищення точності аналізу контенту сайту і підняття позиції сайту в результатах пошуку.

Показано, що існує велика кількість доступних систем автоматичного пошуку ключових слів, розроблених і орієнтованих на обробку природних мов. Ці системи засновані на значній кількості відомих методів пошуку ключових слів, які діляться на експертно-лінгвістичні та статистичні. При цьому експертно-лінгвістичні методи ресурсоємні на ранніх етапах: розробка онтологій, наприклад, вельми трудомісткий процес [1]. З іншого боку, статистичні методи супроводжуються значними обсягами «вербального шуму», який суттєво впливає на якість пошуку ключових слів. Тому найбільш перспективними для дослідження, на нашу думку, є гібридні методи, для яких швидкість статистичної обробки тексту підсилюється можливостями баз знань та функціоналом сучасних лінгвістичних пакетів.

Найбільшою суттєвою проблемою відомих методів знаходження ключових слів лишається недостатня якість процесу автоматизованої обробки вербальної інформації для пошуку ключових слів в тексті. Дуже важливим, з огляду на це, є вибір найбільш інформативної ознаки (критерія) при пошуку ключових слів як в традиційних умовах обробки колекції текстів, так і в умовах аналізу блогів та мікроблогів для WEB [91], [92].

Підвищення якості процесу пошуку ключових слів вимагає залучення додаткової інформації, бажано універсального, а не специфічного характеру. Зокрема, не враховуються у відомих методах пошуку ключових слів результати аналізу зв’язків між лексичними одиницями тексту, а інформаційна оцінка результатів парсингу тексту не береться до уваги як складова відповідного критерія якості.

Актуальність напряму досліджень у розвитку методів та засобів автоматизованого пошуку ключових слів стала причиною появи на даний час досить популярних систем та ресурсів, що є інструментами розкрутки та

підвищення рейтингу сайтів. Тому можна вважати, що удосконалення відповідних методів з метою підвищення якості пошуку ключових слів має необхідну інструментальну підтримку і може бути швидко та ефективно впровадженим у SEO-індустрію.

Отже, для досягнення мети дослідження, з урахуванням вищезазначеного, зокрема доцільності застосування гібридних методів, необхідно:

1. Побудувати математичну модель процесу пошуку ключових слів на основі інформаційної оцінки результатів парсингу тексту.

2. Розробити метод пошуку ключових слів на основі визначення зв'язків між словоформами.

3. Розробити метод зменшення впливу вербального шуму на пошук ключових слів.

4. Провести експериментальне дослідження результатів запропонованих методів у порівнянні з аналогами.

5. Розробити та апробувати інформаційну технологію пошуку ключових слів англомовного тексту.

1.8 Висновки до розділу

Ключові слова є важливим елементом упорядкування важливого масиву інформації, яким є науковий журнал або ж ціла база даних наукових статей. У ключових слів є безліч варіантів використання в процесах пошуку, класифікації та оцінки інформації. Ключові слова використовуються для систематизації масиву статей, вони дозволяють швидше знайти статтю, групувати схожі статті, класифікувати статті у межах інших структурних групувань.

Найважливішим етапом в задачі знаходження ключових фраз є розрахунок їх ваг інформативності, який дозволяє оцінити їх значимість по відношенню один до одного в документі. Для кожної з відібраних ключових фраз

розраховуються ознаки, які дозволяють судити про важливість кандидата для даного документа.

Статистичні методи, зокрема застосування частотного словника для пошуку ключових слів, мають суттєві обмеження щодо області та умов застосування, при чому супроводжуються значними обсягами «вербального шуму», фільтрація якого вимагає використання спеціальних, налаштованих на особливості обраної текстової колекції евристик.

Вибір найбільш інформативної ознаки при пошуку ключових слів та/або фраз має безпосередній вплив на якість результатів їх пошуку.

Підвищення якості процесу пошуку ключових слів для природно-мовного тексту вимагає залучення додаткової інформації про цей текст, бажано універсального, а не специфічного характеру.

2 РОЗРОБКА МАТЕМАТИЧНОЇ МОДЕЛІ ПОШУКУ КЛЮЧОВИХ СЛІВ

У розділі розглянуто формальну постановку задачі пошуку ключових слів як результату згортки інформації у тексті. Запропоновано математичну модель, що знаходить ключові слова на основі аналізу зв’язків між лексичними одиницями тексту. Проведена інформаційна оцінка результатів парсингу тексту показала переваги запропонованого підходу на відміну від традиційного статистичного аналізу слів тексту, що знайшло своє відображення у запропонованому критерію якості пошуку ключових слів.

2.1 Формалізація задачі пошуку ключових слів тексту

Розглянемо довільний текст як множину синтаксично зв'язаних упорядкованих слів, які, в свою чергу, є підмножиною слів мови: $w \in \{W\} \subset \{\omega\}$, де w – слово, W – множина слів в тексті, ω – множина слів мови [93].

Відомо, що кожне слово має основну форму (канонічна форма, лема), до якої можна звести слово шляхом його морфологічного аналізу. Формально представимо це застосуванням до слова функції нормалізації m морфологічного аналізу: $m(w) = b,\ w \in \{W^F\}$, де $\{W^F\}$ – множина словоформ слова w, b – основна форма слова w. Відомі дві основні властивості функції нормалізації [93]: в результаті нормалізації основної форми слова отримуємо основну форму слова, що показано у формулі (2.1); для будь-якого слова нормалізація дає основну форму слова, яка належить множині словоформ, що зазначено у формулі (2.2):

$$m(b) = b. \qquad (2.1)$$

$$\forall w(w \in \{W\}) \Longrightarrow m(w) = b,\ b \in \{W\}\,. \qquad (2.2)$$

Отже, для задачі пошуку ключових слів, текст T можна представити, як впорядкований набір слів w_i і символів c_i [94]:

$$T = \{w_i, c_i\},\ w_i \in \{W_T\} \subset \{\omega\},\ c_i \in \{C\} \subset \{\varsigma\}\,, \qquad (2.3)$$

де $\{W_T\}$ – множина слів в тексті T, що є підмножиною $\{\omega\}$ – множини слів мови; c_i – знаки пунктуації, цифри, пробіли, переходи на новий рядок та інші символи, що не є буквами [95]; $\{C\}$ – множина символів в тексті, що є підмножиною $\{\varsigma\}$ – множини всіх символів; i – порядковий номер слова або не буквенного символу в тексті [94].

Між словами w і символами c існують зв'язки R в тексті, які можуть бути трьох видів [94]:

- зв'язок між i-м словом та l-м словом: $(w_i, w_l) \in R_j$, де j – довільний порядковий номер зв'язку в тексті;
- зв'язок між i-м словом та l-м символом: $(w_i, c_l) \in R_j$;
- зв'язок між i-м символом та l-м символом: $(c_i, c_l) \in R_j$.

Будемо вважати ключовими словами $K = W^k$ такі, які коротко представляють сутність тексту T [94]:

$$W^k(T) = \{w^k\},\ w^k \in \{\omega\}\,, \qquad (2.4)$$

де w^k – ключове слово, що належить множині слів мови $\{\omega\}$.

Розглянемо процес знаходження ключових слів як згортку текстової інформації за певними критеріями. У цьому випадку ключові слова не завжди можливо формально знайти в тексті. Також, в якості ключових, можуть використовуватися: синоніми, терміни з якими текст може бути пов'язаний

логічно, власні назви, з якими асоціюється текст [25], [24]. Найперше будемо аналізувати варіант, коли всі ключові слова знаходяться в тексті K_T [94]. Наглядно такий випадок представлено спільним сектором на рисунку 2.1.

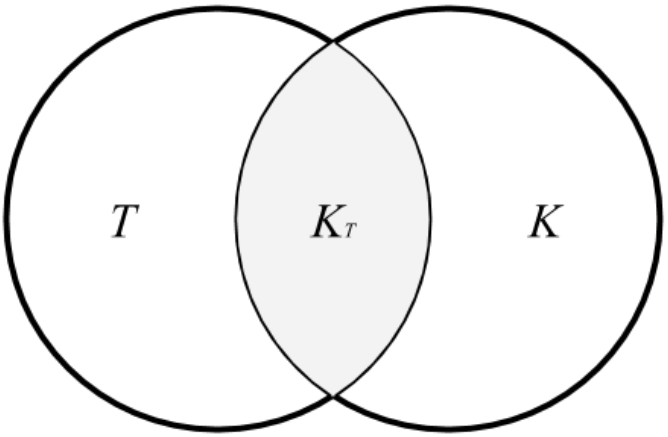


Рисунок 2.1 – Множини слів в тексті і ключових слів

Отже, формально задачу пошуку ключових слів K_T для тексту T можна описати як знаходження таких слів $W^k(T)$, які належать цьому тексту і входять до складу множини слів мови [94]:

$$W^k(T) = \{w^k\},\ w^k \in \{W_T\} \subset \{\omega\}. \qquad (2.5)$$

Вважатимемо, що ключове слово w^k є словом w з тексту T, яке приведено до основної форми b [94]:

$$w^k = b\ = m(w),\ w \in \{W^F\}, \qquad (2.6)$$

де $\{W^F\}$ – множина словоформ одного слова w; $m(w)$ – функція нормалізації морфологічної форми слова w.

З огляду на задачі дослідження позначимо словосполучення як [94]:

$$G \rightarrow [DT] \rightarrow D, \qquad (2.7)$$

де G – головне слово (Governor); $[DT]$ – тип зв'язку (Dependency Type); D – залежне слово (Dependent), при чому для зручності розгляду зв'язків слова з (2.7) краще представляти у формі (2.5), але для пошуку ключових слів – у формі (2.6).

Представлення типових зв'язків Stanford Dependencies було розроблено таким чином, щоб забезпечити простий опис граматичних відношень у реченні [96]. Такий опис можна легко зрозуміти і ефективно використовувати людьми без лінгвістичного досвіду, які хочуть програмно визначити текстові зв’язки. Зокрема, замість фразово-структурних уявлень, які давно домінують у співтоваристві комп’ютерної лінгвістики [97], Stanford Dependencies репрезентують всі зв’язки в реченні рівномірно, як типізовані залежності (фактично, як трійки відношень між парами слів). Це просте, однорідне представлення є цілком природним для фахівців, які не є професійними лінгвістами, але займаються задачами пошуку інформації в тексті, оскільки ефективне для додатків, що забезпечують програмний доступ до зв’язків [94].

Наприклад для речення: «Bell, based in Los Angeles, makes and distributes electronic, computer and building products.», зв’язки будуть мати наступне представлення: nsubj(makes-8, Bell-1) nsubj(distributes-10, Bell-1) vmod(Bell-1, based-3) nn(Angeles-6, Los-5) prep_in(based-3, Angeles-6) root(ROOT-0, makes-8) conj_and(makes-8, distributes-10) amod(products-16, electronic-11) conj_and(electronic-11, computer-13) amod(products-16, computer-13) conj_and(electronic-11, building-15) amod(products-16, building-15) dobj(makes-8, products-16) dobj(distributes-10, products-16). Ці зв’язки представлені у вигляді асоціативного масиву, що наведено на рисунку 2.2, та орієнтованого графу, наведеного на рисунку 2.3.

0	-	-	8 root	-		-	
1	Bell	-	3 vmod	-		-	
2	,	-		-		-	
3	based	-	6 prep_in	-		-	
4	in	-		-		-	
5	Los	-		-		-	
6	Angeles	-	5 nn	-		-	
7	,	-		-		-	
8	makes	-	1 nsubj	-	10 conj_and	-	16 dobj
9	and	-		-		-	
10	distributes	-	1 nsubj	-	16 dobj	-	
11	electronic	-	13 conj_and	-	15 conj_and	-	
12	,	-		-		-	
13	computer	-		-		-	
14	and	-		-		-	
15	building	-		-		-	
16	products	-	11 amod	-	13 amod	-	15 amod
17	.	-		-		-	

Рисунок 2.2 – Приклад асоціативного масиву зв’язків речення

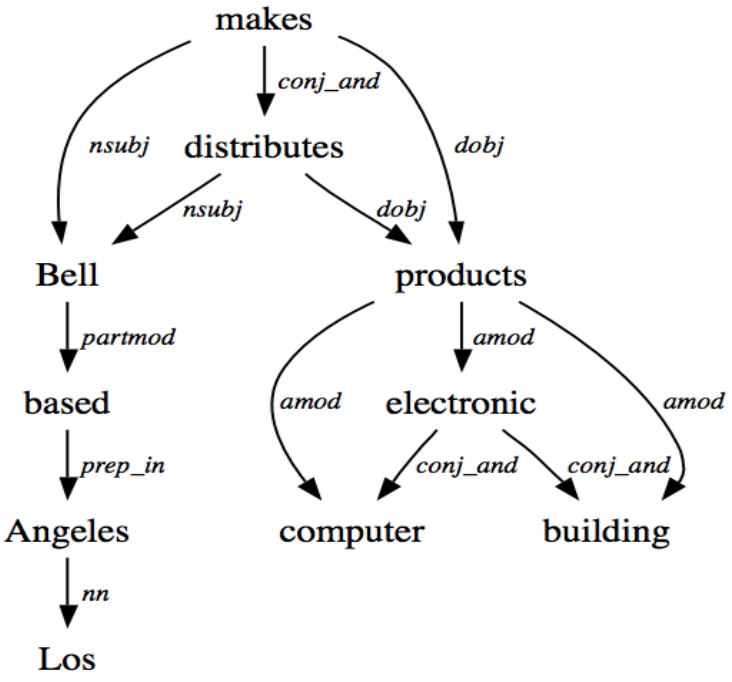


Рисунок 2.3 – Приклад орієнтованого графу зв’язків речення

Зв’язки, відображені на орієнтованому графі розглянутого прикладу, складаються із слів речення, що є вершинами графу, а також позначень типів зв’язків за форматом Stanford Dependencies, що з’єднують вершини [98], [99], [100].

З урахуванням вищезазначеного формалізуємо задачу пошуку ключових слів тексту як параметричну ідентифікацію функції згортки вербальної інформації за критерієм максимума інформації у зв'язках $I(G \to [DT] \to D)$, які поєднують n обраних ключових слів тексту T між собою та з усіма m' значущими словами цього тексту

$$W^k(T) = argmax\, f(W_T), \tag{2.8}$$

де $W^k(T) = \{w_i^k \mid i = \overline{1,n}\}$ – множина n обраних ключових слів тексту з усіх m' значущих слів цього тексту $W_T = \{w_j \mid j = \overline{1,m'}\}$, а

$$f(W_T) = \sum_{i=1}^{n}\sum_{j=1}^{m'} I(G_i \to [DT] \to D_j) + \sum_{i=1}^{n}\sum_{j=1}^{m'} I(G_j \to [DT] \to D_i) \tag{2.9}$$

– функція згортки вербальної інформації в процесі пошуку ключових слів тексту.

2.2 Інформаційна оцінка парсингу тексту для задачі пошуку ключових слів

Розглянемо задачу пошуку ключових слів тексту як певну інформаційну технологію, що має на вході текст, а на виході – множину з l ключових слів $W^k = \{w_1^k, ..., w_l^k\}$. Без применшення загальності будемо вважати, що текст T

складається з m' різних слів, а окреме його *j*-те речення з *k* налічує *n* слів з m' можливих, причому $m' >> n$ та $m' >> l$. Більшість відомих методів пошуку ключових слів тексту беруть за основу частотний словник тексту, який фактично є списком або впорядкованою множиною пар $D' = \{< w_i, f_i >\}$, $i = \overline{1, m'}$, де w_i – одне слово з m', а f_i – його частота ($f_i \geq f_{i+1}$, $i = \overline{1, m' - 1}$), що визначена для *T*. За певною фільтрацією окремих незначущих категорій слів ключовими вважають перші *l* слів зі списку D', тобто, дещо спрощено маємо $W^k = \{w_1, ..., w_l\}$ [101].

Проте результати парсингу природних мов за допомогою сучасних лінгвістичних пакетів дозволяють на доступному програмному рівні [102] оперувати синтаксичними зв'язками між словами окремого речення. Окрім того, можливості цих пакетів дозволяють суттєво зменшити значення *m* шляхом об'єднання слів у словоформи, а останні – у леми та стеми. Отже, необхідно з'ясувати, які формальні переваги для пошуку $W^k = \{w_1^k, ..., w_l^k\}$ надасть нам програмно-лінгвістичне забезпечення процедури парсингу всіх речень тексту *T* [101].

З інформаційної точки зору розуміння сенсу речення окремим суб'єктом супроводжується розпізнаванням а) окремих слів, з яких воно складається та б) зв'язків між парами цих слів з відповідною побудовою дерева таких зв'язків [103]. Вважатимемо, що всі ці процеси відбуваються шляхом порівняльного аналізу та залучення інформації з деякої загальнолінгвістичної бази знань суб'єкта розуміння. Якщо кожен з цих етапів супроводжується збільшенням інформації, то приймаємо робочу гіпотезу [101]:

- рівень загального розуміння тексту *T* може змінюватися від мінімально можливого до максимального в залежності від обсягу та інших параметрів загальнолінгвістичної бази знань суб'єкта;

- якість пошуку $W^k = \{w_1^k, ..., w_l^k\}$ пропорційна рівню загального розуміння тексту, що має підтверджуватися формальними ознаками.

Нехай будь-яке *j*-те речення з *k* складається з *n* різних слів, що не є досить жорстким обмеженням. Тоді зв'язне дерево парних залежностей такого речення налічує або *n–1* гілок, якщо не брати до уваги зворотну залежність між підметом та присудком, або *n* – якщо її врахувати. Відповідно загальна кількість слів цього речення для подальшого поглибленого аналізу збільшується або до *2×n–2* або до *2×n*. Проте таке збільшення відбувається нерівномірно – для всіх не термінальних (кінцевих) вузлів дерева частоти відповідних слів не змінюються, а для термінальних (проміжних) можуть зрости суттєво. В таблиці 2.1 показані випадки зміни частот слів з урахуванням парних залежностей для різних типів речення [101].

Таблиця 2.1 – Аналіз збільшення частоти значущих слів унаслідок урахування парних залежностей для різних типів речення

Склад речення / кількість слів	Тип речення та граф його дерева залежностей	Частотна формула	Кінцева частота
Ab / 2	Словосполучення (Коріння дерева)	A+b	2
Abc / 3	Лінійна трійка (Бережи скарби природи)	A+2b+c	4
Abcd / 4	Лінійна четвірка (Отримав переклад слова дивного)	A+2b+2c+d	6
Abcde / 5	Розгалудження (Густий ліс нізвідки завершився проваллям)	A+2b+c+3d+e	8
Abcdef / 6	Група підмета (Сині примружені очі коханого говорили багато)	A+b+4c+d+2e+f	10
Abcdef / 6	Група присудка (Досвідчений кінь борозну швидко відчує нюхом)	A+2b+c+d+4e+f	10
Abcdef / 6	Обидві групи (Старий дід Еол зобрав всіх вітрів)	A+3b+c+2d+e+2f	10

Навіть такий елементарний аналіз на рівні одного речення показує, що збільшуються частоти саме тих слів, які потенційно можуть належати до множини ключових. Проведемо формальну оцінку такого збільшення для накладених обмежень щодо наявності виключно різних слів у реченні та не врахуванням зворотної залежності між підметом та присудком [101]:

а) мінімальне збільшення відсутнє за умови знаходження *i*-го слова з m' серед не термінальних (кінцевих) вузлів дерева кожного речення, де це слово зустрічається, тобто $f_i^{min} = 0,\ f_i^{new} = f_i,\ i = \overline{1, m'}$;

б) якщо *i*-те слово знаходиться у кожному з *k* речень тексту та, окрім того, відповідає у кожному реченні найбільш розгалуженому термінальному вузлу, то максимальне збільшення частоти складає $f_i^{max} = \sum_{j=1}^{k}(n_j - 2),\ i = \overline{1, m'}$. Відповідно

$$f_i^{new} = f_i + f_i^{max} = k + \sum_{j=1}^{k}(n_j - 2) = \sum_{j=1}^{k}(n_j - 1);$$

в) в загальному та більш реальному випадку $f_i = z \mid z \le k$, тобто *i*-те слово знаходиться у *z* реченнях з *k* маємо $f_i^{new} = z + \sum_{j=1}^{z}(n_j - 2) = \sum_{j=1}^{z}(n_j - 1)$ як оцінку зверху збільшення частоти *i*-го слова.

Отже, експериментальні дослідження мають підтвердити справедливість отриманих формальних оцінок процесу пошуку ключових слів у встановлених межах лінійного зростання частоти значущих слів тексту, з яких обираються ключові.

2.3 Аналіз зв'язків між лексичними одиницями тексту

Розглянемо поняття лексичної одиниці тексту. Лексика, словниковий склад – це матерія мови. Мовлення не може існувати без лексики; граматика в основному відображає взаємозв'язки слів, які в мовленні є носіями значення [104], [105].

Зазвичай лексику ототожнюють з словами, тобто лексика складається із слів, до яких відносяться так звані кореневі, безафіксні слова, складні та похідні слова. Однак, відповідно до лінгвістичних та методичних критеріїв, до лексики можна віднести інші одиниці, що також цілісно відтворюються у мовленні і означають якийсь предмет дійсності. Отже, до розряду лексичних одиниць слід віднести: слова; сталі словосполучення – номінативні та фразеологічні; вирази-кліше [104], [105].

Окрім власних, «внутрішніх» властивостей, слову притаманні також особливі, «зовнішні» властивості – здатність до сполучуваності з іншими словами, завдяки чому утворюються нові словосполучення, а з них — синтагми і фрази [104], [105].

Словосполучення, зазвичай, складаються з головного і залежного слова, які поєднуються зв'язком між ними згідно з (2.7). Відповідно до [96] існують універсальні залежності (зв'язки), які є крос-лінгвістичними, послідовними, деревоподібними анотаціями для багатьох мов. Відповідні специфікації розроблені з метою сприяння розвитку багатомовного аналізатора, міжмовного навчання та досліджень з точки зору типології мови. Зокрема, йде мова про універсальні зв'язки $DT = \{DT_i \mid i = \overline{1,38}\}$ [96] або

$$
\begin{aligned}
DT = \{ & ACL,\ ADVCL,\ ADVMOD,\ AMOD,\ APPOS,\ AUX,\\
& CASE,\ CC,\ CCOMP,\ CLF,\ COMPOUND,\ CONJ,\ COP,\\
& CSUBJ,\ DEP,\ DET,\ DISCOURSE,\ DISLOCATED,\\
& EXPL,\ FIXED,\ FLAT,\ GOESWITH,\ IOBJ,\ LIST,\\
& MARK,\ NMOD,\ NSUBJ,\ NUMMOD,\ OBJ,\ OBL,\\
& ORPHAN,\ PARATAXIS,\ PUNCT,\ REF,\\
& REPARANDUM,\ ROOT,\ VOCATIVE,\ XCOMP\}
\end{aligned}
\tag{2.10}
$$

де $DT_1 = ACL$ – предикативний модифікатор іменника. Приклади *ACL* зв'язків наведено на рисунку 2.4.

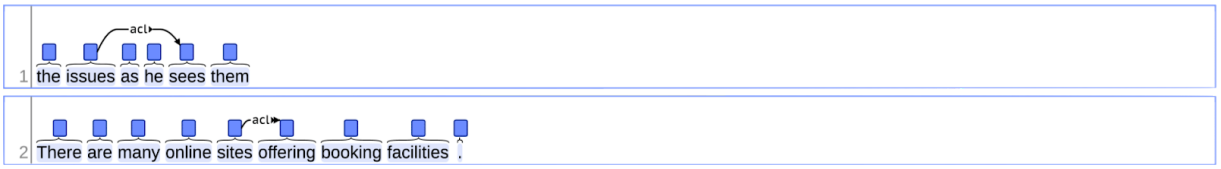


Рисунок 2.4 – Приклади зв'язків *ACL*

$DT_2 = ADVCL$ – зв'язок між дієсловом та дієсловом, якому належить модифікатор доповнювач. Приклади *ADVCL* зв'язків наведено на рисунку 2.5.

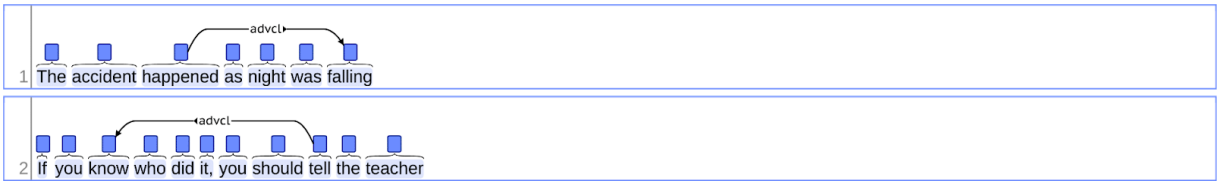


Рисунок 2.5 – Приклади зв'язків *ADVCL*

$DT_3 = ADVMOD$ – зв'язок між словом та прислівником. Приклад *ADVMOD* зв'язку наведено на рисунку 2.6.

Рисунок 2.6 – Приклад зв'язку *ADVMOD*

$DT_4 = AMOD$ – зв'язок між іменником і модифікатором прикметником. Приклад *AMOD* зв'язку наведено на рисунку 2.7.

Рисунок 2.7 – Приклад зв'язку *AMOD*

$DT_5 = APPOS$ – зв'язок між іменником і номіналом, що безпосередньо слідує за першим іменником, та визначає, змінює, називає або описує це іменник. Приклад $APPOS$ зв'язку наведено на рисунку 2.8.

Рисунок 2.8 – Приклад зв'язку *APPOS*

$DT_6 = AUX$ – зв'язок між основним дієсловом і допоміжним дієсловом. Приклад *AUX* зв'язку наведено на рисунку 2.9.

Рисунок 2.9 – Приклад зв'язку *AUX*

$DT_7 = CASE$ – зв'язок, що використовується для позначення будь-якого елемента списку. Приклад *CASE* зв'язку наведено на рисунку 2.10.

Рисунок 2.10 – Приклад зв'язку *CASE*

$DT_8 = CC$ – зв'язок між основним словом та сполучником. Приклад *CC* зв'язку наведено на рисунку 2.11.

Рисунок 2.11 – Приклад зв'язку *CC*

$DT_9 = CCOMP$ – зв'язок між дієсловом або прикметником і доповнювачем (слова, які можуть бути використані для перетворення предиката у підмет або додаток речення). Приклад *CCOMP* зв'язку наведено на рисунку 2.12.

Рисунок 2.12 – Приклад зв'язку *CCOMP*

$DT_{10} = CLF$ – зв'язок між іменником і словом, що його супроводжує у певних граматичних контекстах. Найбільш канонічним використанням є цифрові класифікатори, де слово використовується з числом для підрахунку об'єктів. Приклад *CLF* зв'язку наведено на рисунку 2.13.

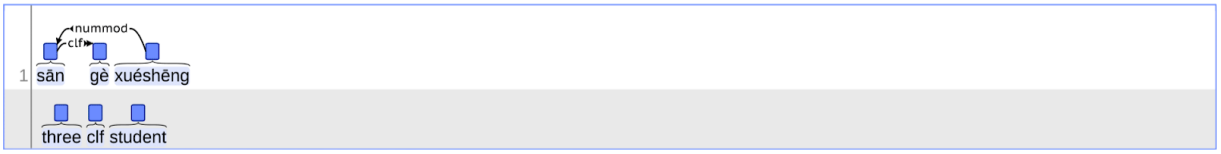


Рисунок 2.13 – Приклад зв'язку *CLF*

$DT_{11} = COMPOUND$ – багатослівні вирази, іменні, а також дієслівні та прикметникові, що є поширеними у мові. Приклад *COMPOUND* зв'язку наведено на рисунку 2.14.

Рисунок 2.14 – Приклад зв'язку *COMPOUND*

$DT_{12} = CONJ$ – зв'язок між двома словами, зв'язаними координаційним сполучником. Приклади *CONJ* зв'язків наведено на рисунку 2.15.

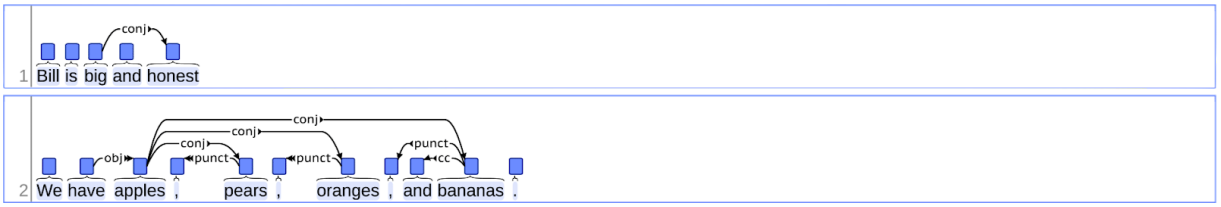


Рисунок 2.15 – Приклади зв'язків *CONJ*

$DT_{13} = COP$ – зв'язок між підметом і невербальним присудком. Приклад *COP* зв'язку наведено на рисунку 2.16.

Рисунок 2.16 – Приклад зв'язку *COP*

$DT_{14} = CSUBJ$ – зв'язок між дієсловом і підметом, що є доповнювачем. Приклад *CSUBJ* зв'язку наведено на рисунку 2.17.

Рисунок 2.17 – Приклад зв'язку *CSUBJ*

$DT_{15} = DEP$ – невизначений зв'язок. Зв'язок може бути позначений як *DEP*, коли неможливо визначити більш точне співвідношення. Це може бути наслідком незвичної граматичної конструкції або обмеження у програмному забезпеченні конвертації чи розбору.

$DT_{16} = DET$ – зв'язок визначника, що існує між номінально головним словом та його визначником. Найчастіше, слово, яке має тег частини мови *DET*, буде мати такий же зв'язок визначника *DET* і навпаки [106]. Відомим винятком є те, що у деяких з наборів даних присвійний визначник (наприклад, такий як

«my») у певний момент отримує тег частини мови *DET*, але зв'язок *NMOD*, що є паралеллю до інших присвійних конструкцій [107]. Але це не повністю однаково для різних мов, у деяких мовах набагато чіткіше, ніж у англійській, виражено те, як присвійні визначники відносяться до прикметників, тому відношення *NMOD* не підлягає сумніву [98], [99], [100]. Приклади *DET* зв'язків наведено на рисунку 2.18.

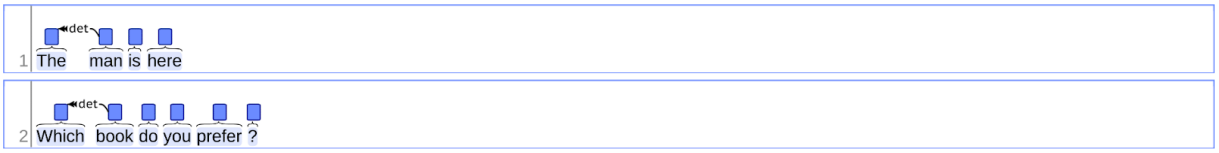


Рисунок 2.18 – Приклади зв'язків *DET*

$DT_{17} = DISCOURSE$ – використовується для позначення включень та інших частинок і елементів дискурсу (які чітко не пов'язані зі структурою речення, крім виразного способу). Приклад *DISCOURSE* зв'язку наведено на рисунку 2.19.

Рисунок 2.19 – Приклад зв'язку *DISCOURSE*

$DT_{18} = DISLOCATED$ – використовується для позначення фронтальних або відкладених елементів, які не відповідають звичайним граматичним зв'язкам у реченні. Ці елементи часто знаходяться на периферії речення, і можуть бути відокремлені комами. Приклади *DISLOCATED* зв'язків наведено на рисунку 2.20.

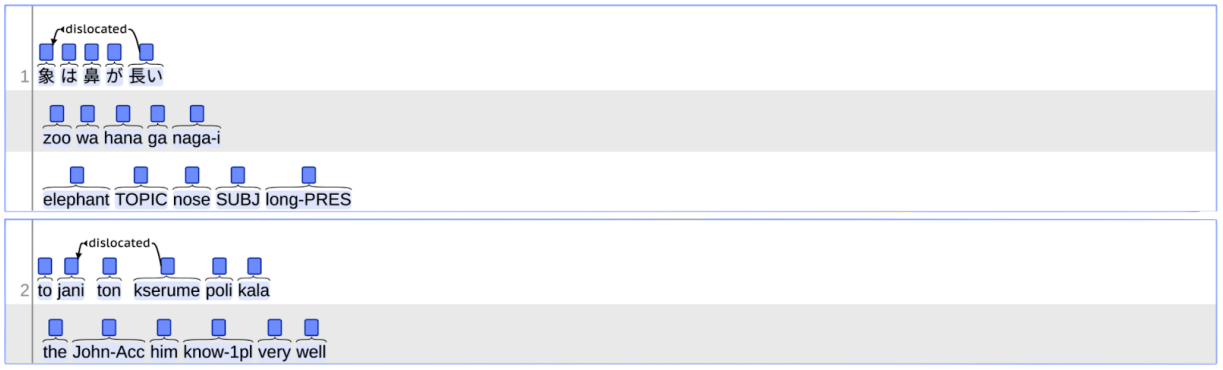


Рисунок 2.20 – Приклади зв'язків *DISLOCATED*

$DT_{19} = EXPL$ – це відношення, що фіксує вставні або плеонастичні номінали. Такі номінали з'являються в аргументній позиції присудка, але не виконують ніякої з семантичних ролей присудка [107]. Основний присудок речення (дієслово або присудковий прикметник чи іменник) є головним словом. В англійській мові це стосується деяких способів використання it та there: екзистенціальне there, а також it при використанні в експозиційних конструкціях [98], [99], [100].

Деякі мови не мають таких, подібних англійському, висловів, це стосується більшості мов pro-drop (мова, в якій певні класи займенників можуть бути вилучені, коли вони прагматично або граматично інерційні). Також це явище часто називають нуль або нульовою анафорою [108]. У мовах з подібними висловами вони можуть бути розташовані там, де зазвичай з'являється основний аргумент: підмет та прямий (і, навіть, непрямий) додаток [98], [99], [100]. Приклади *EXPL* зв'язків наведено на рисунку 2.21.

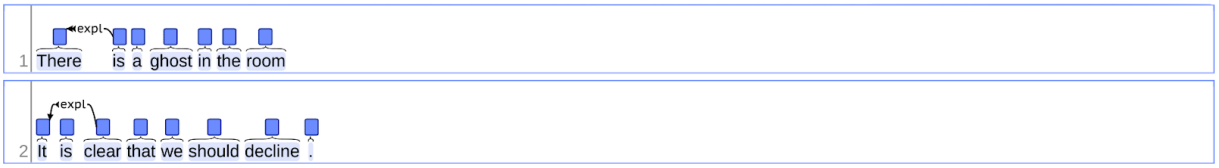


Рисунок 2.21 – Приклади зв'язків *EXPL*

$DT_{20} = FIXED$ – використовується для певних сталих граматичних виразів, які ведуть себе як функціональні слова або короткі прислівники [109].

Сталі багатослівні вирази анотовано у рівній структурі, де всі наступні слова у виразі прикріплені до першого з використанням сталої мітки. Припущення полягає в тому, що ці вирази не мають внутрішньої синтаксичної структури (окрім з історичної точки зору), а також структурна анотація, в принципі, є довільною [107]. Однак, на практиці, дуже важливо використовувати послідовну анотацію всіх сталих багатослівних виразів на всіх мовах [98], [99], [100]. Приклади *FIXED* зв'язків наведено на рисунку 2.22.

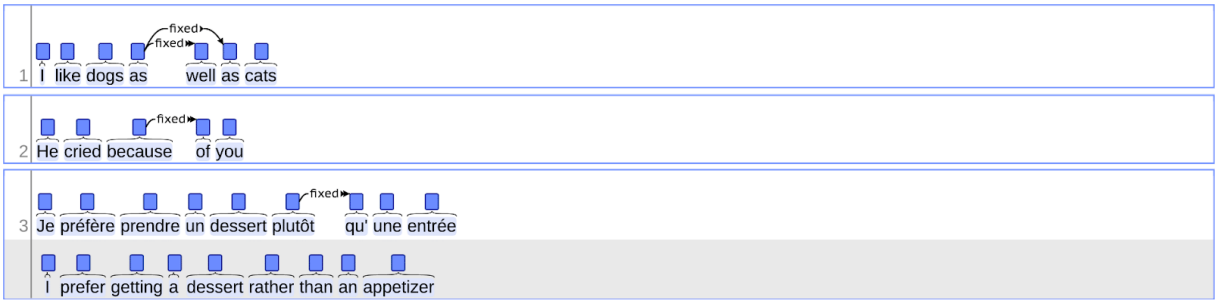


Рисунок 2.22 – Приклади зв'язків *FIXED*

$DT_{21} = FLAT$ – використовується для позначення сталих багатослівних виразів, таких як власні назви або дати. Приклади *FLAT* зв'язків наведено на рисунку 2.23.

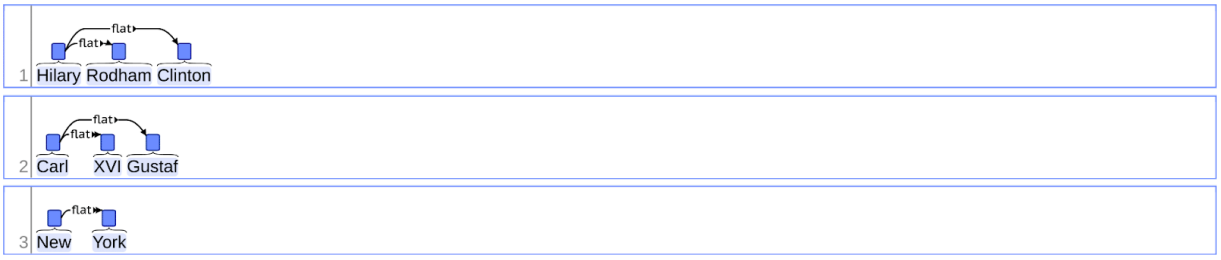


Рисунок 2.23 – Приклади зв'язків *FLAT*

$DT_{22} = GOESWITH$ – зв'язок між двома або більше частинами слова, які розділені у тексті, що не є добре відредагованим. Приклади *GOESWITH* зв'язків наведено на рисунку 2.24.

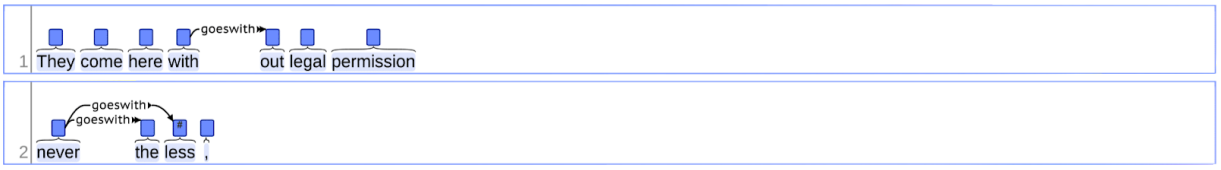


Рисунок 2.24 – Приклади зв'язків *GOESWITH*

$DT_{23} = IOBJ$ – зв'язок між дієсловом та його похідним додатком. Приклад *IOBJ* зв'язку наведено на рисунку 2.25.

Рисунок 2.25 – Приклад зв'язку *IOBJ*

$DT_{24} = LIST$ – використовується для позначення елементів, які можна порівнювати. Приклади *LIST* зв'язків наведено на рисунку 2.26.

Рисунок 2.26 – Приклади зв'язків *LIST*

$DT_{25} = MARK$ – зв'язок між підпорядкованим дієсловом в прислівному зв'язку та субординаційним сполучником, який його вводить. Приклади *MARK* зв'язків наведено на рисунку 2.27.

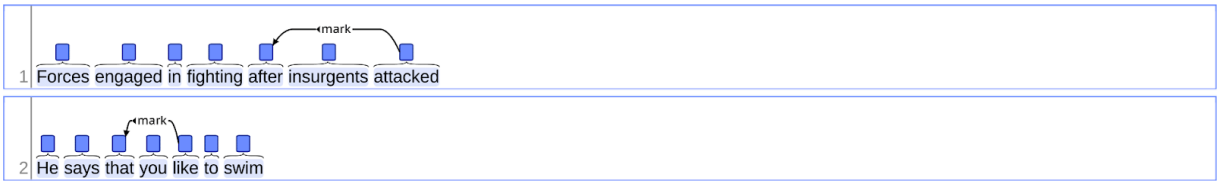


Рисунок 2.27 – Приклади зв'язків *MARK*

$DT_{26} = NMOD$ – відношення використовується для номінальних зв'язків інших іменників або іменних груп, а також функціонально відповідає означенню або доповненню. Приклади *NMOD* зв'язків наведено на рисунку 2.28.

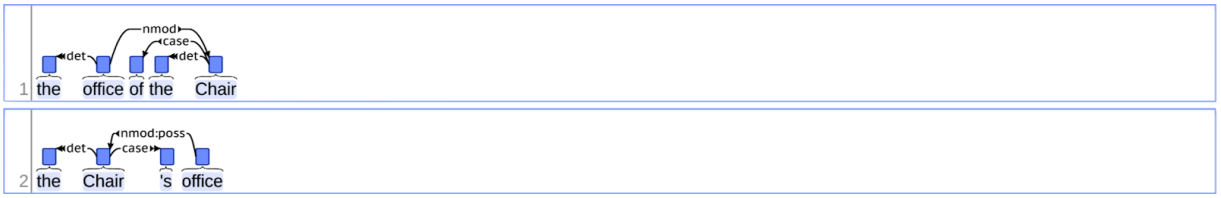


Рисунок 2.28 – Приклади зв'язків *NMOD*

$DT_{27} = NSUBJ$ – зв'язок між дієсловом і підметом іменної групи. Це також зв'язок між пасивним дієприкметником і підметом іменної групи. Приклади *NSUBJ* зв'язків наведено на рисунку 2.29.

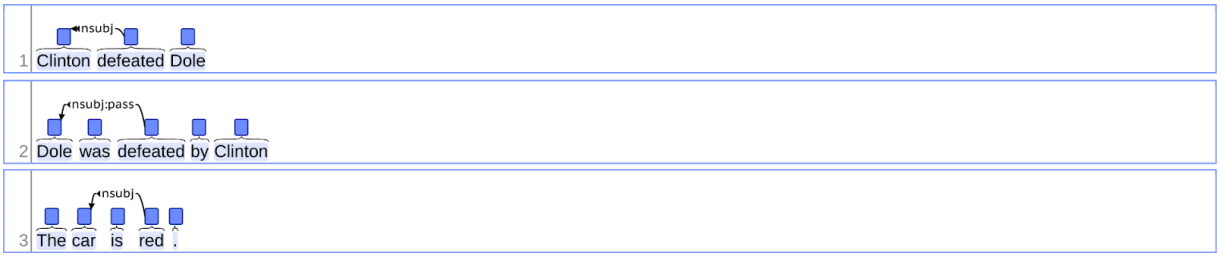


Рисунок 2.29 – Приклади зв'язків *NSUBJ*

$DT_{28} = NUMMOD$ – зв'язок між іменником і будь-якою цифрою, що служить для відображення кількості. Приклади *NUMMOD* зв'язків наведено на рисунку 2.30.

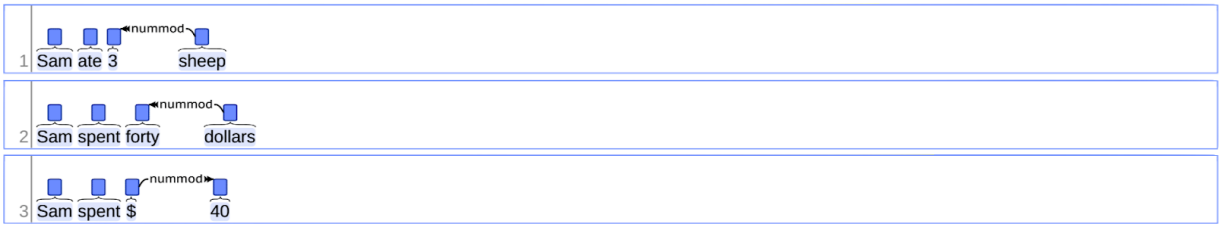


Рисунок 2.30 – Приклади зв'язків *NUMMOD*

$DT_{29} = OBJ$ – зв'язок між дієсловом та одним з його додатків. Приклад *OBJ* зв'язку наведено на рисунку 2.31.

Рисунок 2.31 – Приклад зв'язку *OBJ*

$DT_{30} = OBL$ – зв'язок використовується для номінального іменника / займенника / іменникової групи, що функціонує як непрямий аргумент або обставина. Приклади *OBL* зв'язків наведено на рисунку 2.32.

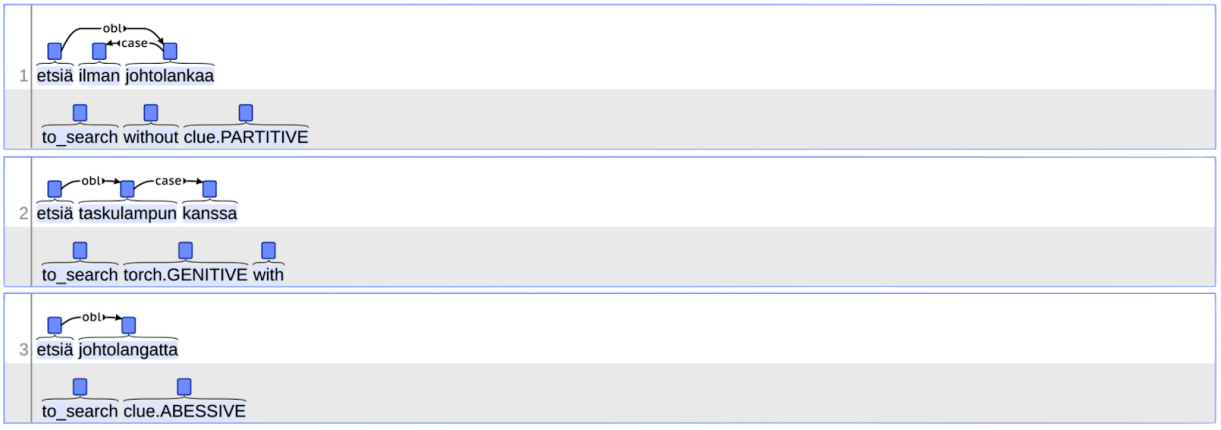


Рисунок 2.32 – Приклади зв'язків *OBL*

$DT_{31} = ORPHAN$ – зв'язок, що застосовується під час пропуску у висловлюванні деяких структурних елементів, які мають домислюватись за контекстом. Приклади *ORPHAN* зв'язків наведено на рисунку 2.33.

Рисунок 2.33 – Приклад зв'язку *ORPHAN*

$DT_{32} = PARATAXIS$ – зв'язок між словом та іншими елементами, такими як вставні слова або предикат після ":" або ";", що розміщені поруч без будь-якої явної координації, підпорядкування або співвідношення аргументів з головним словом. Приклади *PARATAXIS* зв'язків наведено на рисунку 2.34.

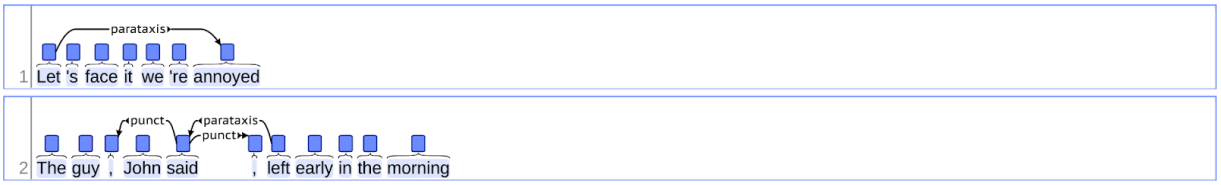


Рисунок 2.34 – Приклади зв'язків *PARATAXIS*

$DT_{33} = PUNCT$ – використовується для позначення будь-якої частини пунктуації в реченні чи частині тексту, якщо пунктуація зберігається в типізованих залежностях [110].

Токени з співвідношенням *PUNCT* завжди прикріплюються до змісту слів і ніколи не можуть мати залежностей. Оскільки *PUNCT* не є нормальним відношенням залежностей, звичайні критерії визначення головного слова не застосовуються. Натомість використовуються такі принципи [107]:

1. Знак пунктуації, що розділяє скоординовані одиниці, додається до наступного зв'язку [107].

2. Знак пунктуації, що передує або слідує за незалежною одиницею, додається до цієї одиниці [107].

3. У межах відповідного підрозділу знак пунктуації прикріплюється до найвищого можливого вузла, який зберігає перспективу [107].

4. Парні знаки пунктуації (наприклад, цитати та дужки, іноді також дефіси, коми тощо) мають бути додані до одного слова, якщо це не порушує перспективу [98], [99], [100].

Приклади *PUNCT* зв'язків наведено на рисунку 2.35.

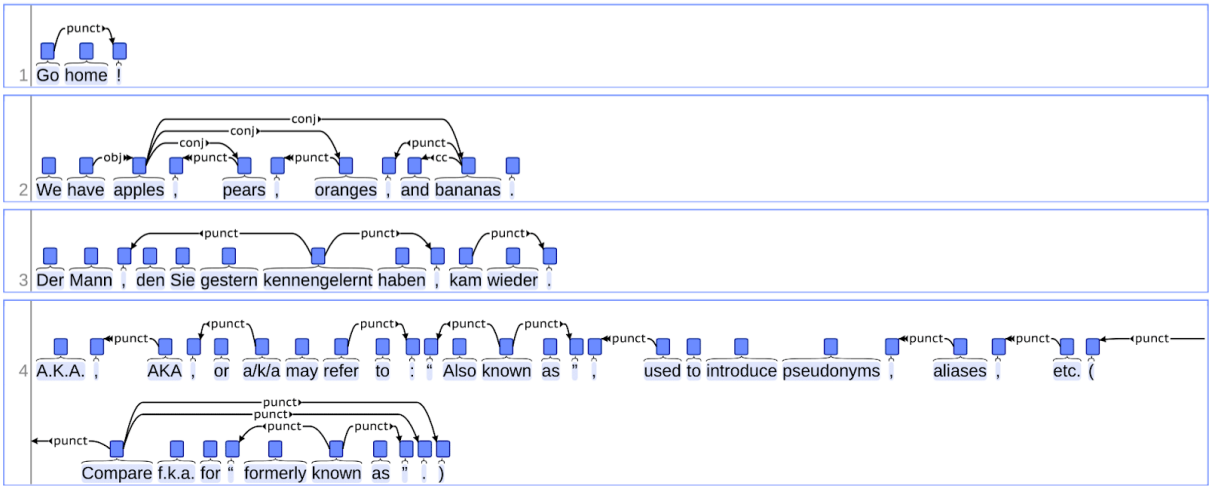


Рисунок 2.35 – Приклади зв'язків *PUNCT*

$DT_{34} = REF$ – референт головного слова іменникового словосполучення, що є відносним словом, яке вводить відносне положення шляхом модифікації іменникового словосполучення [107]. Наприклад для речення: «I saw the book which you bought», зв'язок *REF* буде між словами book та which [96].

$DT_{35} = REPARANDUM$ – використовується для позначення помилок, що були згодом виправленні спікером. Приклад *REPARANDUM* зв'язку наведено на рисунку 2.36.

Рисунок 2.36 – Приклад зв'язку *REPARANDUM*

$DT_{36} = ROOT$ – корневе граматичне відношення, що вказує на корінь речення. Фейковий вузол *ROOT* використовується як головний вузол [111]. Вузол ROOT має індекс 0, оскільки індексація реальних слів у реченні починається з 1 [107]. У кожному дереві повинен бути тільки один кореневий вузол. Якщо основний присудок відсутній, але є багато одиничних залежностей, то одне з них підвищується до положення головного (кореневого), а до нього приєднуються інші одинаки [98], [99], [100]. Приклад *ROOT* зв'язку наведено на рисунку 2.37.

Рисунок 2.37 – Приклад зв'язку *ROOT*

$DT_{37} = VOCATIVE$ – використовується для позначення учасника діалогу, до якого звертаються в тексті (зазвичай в бесідах, діалозі, електронних листах, повідомленнях груп новин, тощо). Приклади *VOCATIVE* зв'язків наведено на рисунку 2.38.

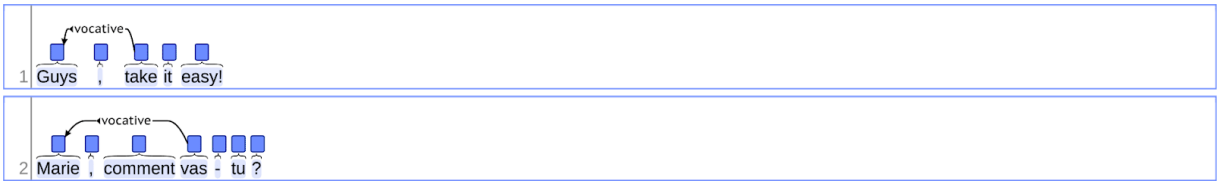


Рисунок 2.38 – Приклади зв'язків *VOCATIVE*

$DT_{38} = XCOMP$ – зв'язок між дієсловом або прикметником і доповненням, що відноситься до дієслівної групи [98], [99], [100]. Приклади *XCOMP* зв'язків наведено на рисунку 2.39.

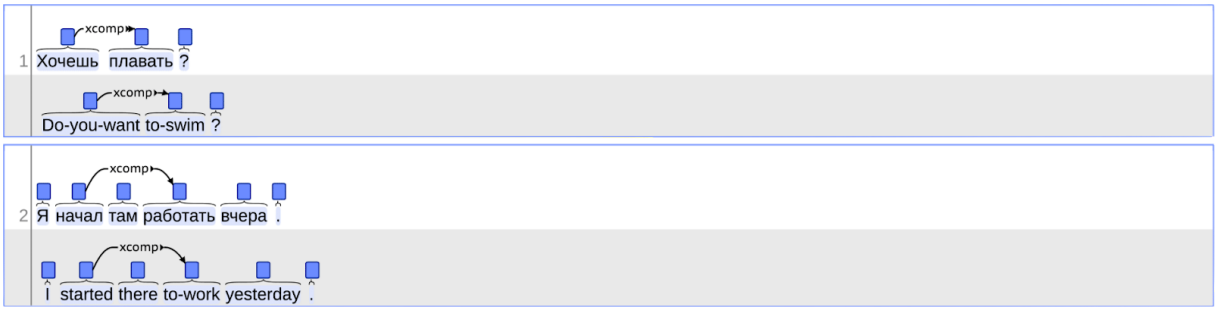


Рисунок 2.39 – Приклади зв'язків *XCOMP*

Також, однією з мовних ознак є віднесеність лексичних одиниць до тих чи інших частин мови [104], [105]. В англійській мові застосовуються такі теги частин мови: CC, CD, DT, EX, FW, IN, JJ, JJR, JJS, LS, MD, NN, NNP, NNPS, NNS, PDT, POS, PRP, PRP$, RB, RBR, RBS, RP, TO, UH, VB, VBD, VBG, VBN, VBP, VBZ, WDT, WP, WP$, WRB, -LRB-, -RRB-, зокрема:

CC – координуючі сполучення: and, but, nor, or, yet, plus, minus, less, times (multiplication), over (division), also for (because), so (i. e., so that), &, 'n, both, either, et, neither, therefore, v., versus, vs., whether [112].

CD – номер, число, кількість: one, two, 2, mid-1890, nine-thirty, forty-two, one-tenth, ten, million, 0.5, forty-seven, 1987, twenty, '79, zero, 78-degrees, eighty-four, IX, '60s, .025, fifteen, 271, 124, dozen, quintillion, DM2,000 [112].

DT – визначник: a, an, every, no, the, another, any, some, all, both, del, each, either, half, la, many, much, nary, neither, such, that, them, these, this, those [112].

EX – екзистенціальне there: ненаголошений there, що викликає інверсію дієслова у відповідній формі та логічного суб'єкта. Наприклад: «There was a party in progress» [112].

FW – іноземні слова, що не належать мові, якою написаний текст [107].

IN – прийменники і сполучники підпорядкування: among, around, astride, atop, behind, below, by, despite, for, if beside, if like, inside, into, near, next, on, out, pro, throughout, towards, until, upon, whether, within [107].

JJ – прикметник, числівник або порядковий номер: battery-powered, calamitous, ectoplasmic, first, fourth, ill-mannered, multi-disciplinary, multilingual, oiled, participatory, pre-war, regrettable, separable, still-to-be-named, third [107].

JJR – прикметник, порівняльний: bleaker, braver, breezier, briefer, brighter, brisker, broader, bumper, busier, calmer, cheaper, choosier, cleaner, clearer, closer, colder, commoner, costlier, cozier, creamier, crunchier, cuter [107].

JJS – прикметник, найвищий ступінь: calmest, cheapest, choicest, classiest, cleanest, clearest, closest, commonest, corniest, costliest, crassest, creepiest, crudest, cutest, darkest, deadliest, dearest, deepest, densest, dinkiest [107].

LS – список, елемент, маркер, цифри та літери, що використовуються як ідентифікатори елементів у списку: A, A., B, B., C, C., D, E, F, First, G, H, I, J, K, One, SP-44001, SP-44002, SP-44005, SP-44007, Second, Third, Three, Two, *, a, b, c, d, first, five, four, one, six, three, two [107].

MD – модальні допоміжні дієслова. Всі дієслова, які не приймають закінчення -s у формі третьої особи однини: can, could, dare, may, might, must, ought, shall, should, will, would, cannot, couldn't, need, ought, shouldn't [112].

NN – іменник, загальні назви, однина або масовий [112].

NNP – іменник, власні назви, однина [112].

NNPS – іменник, власні назви, множина [112].

NNS – іменник, загальні назви, множина [112].

PDT – префіксний визначник. Визначники, як елементи, що передують статті або присвійним займенникам: all, both, half, many, quite, such, sure, this. Наприклад: «all his marbles», «quite a mess» [112].

POS – присвійне закінчення: іменники, що закінчуються маркером ' або 's [107].

PRP – особовий займенник: he, her, hers, herself, him, him, himself, hisself, I, it, itself, me, myself, one, oneself, ours, ourselves, ownself, self, she, she, thee, theirs, them, themselves, they, thou, thy, us, you [107].

PRP$ – присвійний займенник: her, his, its, mine, my, one's, our, ours, their, thy, your [107].

RB – прислівник [107].

RBR – прислівник, вищий ступінь [107].

RBS – прислівник, найвищий ступінь [107].

RP – частка. В основному односкладові слова, що також двоскладові в якості прислівників напрямку: aboard, about, across, along, apart, around, aside, at, away, back, before, behind, by, crop, down, ever, fast, for, forth, from, go, high, i. e., in, into, just, later, low, more, off, on, open, out, over, per, pie, raising, start, teeth, that, through, under, unto, up, up-pp, upon, whole, with you [107].

SYM – символ. Технічні символи або вирази, які не є словами (% & ' " * + , . < = > @ A[fj] U.S U.S.S.R * ** ***) [112].

TO – літерал to, як прийменник або інфінітивний маркер.

UH – вигук: amen, anyways, baby, dammit, diddle, Goodbye, Goody, Gosh, heck, Hey, honey, howdy, Hubba, huh, hush, Jee-sus, Jeepers, Kee-reist, man, my, oh, Oops, please, shucks, sonuvabitch, uh, well, whammo, whodunnit, Wow, yes [107].

VB – дієслово, основна форма.

VBD – дієслово, минулий час.

VBG – дієслово, дієприкметник теперішнього часу або герундій.

VBN – дієслово, дієприкметник минулого часу.

VBP – дієслово, теперішній час, але не третя особа однини.

VBZ – дієслово, теперішній час, третя особа однини.

WDT – wh-визначник: that, what, whatever, which, whichever.

WP – wh-займенник: that, what, whatever, whatsoever, which, who, whom, whosoever [112].

WP$ – присвійний wh-займенник: whose [107].

WRB – wh-прислівник, включаючи when, коли використовується в переносному значенні: how, however, whence, whenever, where, whereby, wherever, wherein, whereof, why [107].

-LRB- – відкрита дужка [107], [113].

-RRB- – закрита дужка [114], [115].

Внаслідок дослідження і аналізу багаточисельних прикладів використання виявлено, що в умовах задачі пошуку ключових слів з аналізу лексичних одиниць тексту можна виключити словосполучення з неінформативними типами зв’язків, а також слова, що відносяться до неінформативних частин мови. Типами зв’язків, які не несуть суттєвого змістовного навантаження є такі 7: CC, DET, EXPL, FIXED, PUNCT, REF, ROOT. Слова, які виключаються з аналізу, відносять до неінформативних частин мови, що мають такі (всього 21) теги: CC, CD, DT, EX, IN, LS, MD, PDT, POS, PRP, PRP$, RP, SYM, TO, UH, WDT, WP, WP$, WRB, -LRB-, -RRB-. Виключення з процесу аналізу тексту саме таких типів зв'язків та частин мови, як і обов'язкове врахування 31 значущого типу зв'язків згідно (2.10) і є головним обмеженням моделі пошуку ключових слів, що пропонується.

2.4 Висновки до розділу

За результатами математичного моделювання формалізовано задачу пошуку ключових слів тексту як параметричну ідентифікацію функції згортки вербальної інформації за критерієм максимума інформації у зв'язках, які поєднують n обраних ключових слів тексту T між собою та з усіма m' значущими словами цього тексту.

На основі проведеної інформаційної оцінки результатів парсингу тексту отримано формальні межі лінійного збільшення кількості інформації для значущих слів тексту, з яких обираються ключові. На відміну від існуючих

моделей, така інформація враховується запропонованим підходом до пошуку ключових слів тексту.

У межах запропонованої математичної моделі обгрунтовано обов'язкове врахування 31 значущого типу зв'язків згідно (2.10) в процесі пошуку ключових слів та виключення з процесу аналізу тексту 7-ми неінформативних типів зв'язків, а також 21-го тегу, якими позначаються неінформативні частин мови. Це і є головним обмеженням моделі, що пропонується.

Удосконалено модель пошуку ключових слів, яка, на відміну від існуючих, побудована на основі інформаційної оцінки результатів парсингу тексту та враховує результати аналізу зв’язків між лексичними одиницями тексту, що дозволило формалізувати критерій якості процесу пошуку ключових слів.

3 РОЗРОБКА МЕТОДІВ ПОШУКУ КЛЮЧОВИХ СЛІВ АНГЛОМОВНОГО ТЕКСТУ

Розділ присвячено побудові двох взаємопов'язаних методів пошуку ключових слів англомовного тексту. У першому методі запропоновано виокремлювати ключові слова не за статистичними ознаками їх вживання у тексті, а на основі кількості зв’язків між ними та іншими лексичними одиницями тексту. Другий метод доповнює перший та покращує результати пошуку ключових слів шляхом фільтрації вербального шуму, характерного для англійської мови. Визначено мовно-незалежні особливості запропонованих методів, розроблено алгоритм пошуку ключових слів на основі застосування двох послідовних методів.

3.1 Концептуальні основи розробки методів пошуку ключових слів англомовного тексту

В загальному, підхід до пошуку ключових слів, з урахуванням зв'язків між членами речення, складається з етапів:

1. Створення багаторівневої розмітки тексту.

2. Застосування синтаксичної розмітки, що враховує складні залежності між парами лем.

3. Зменшення вербального шуму.

4. Вибір перших n слів з найбільшою кількістю зв’язків, де n – кількість потрібних ключових слів.

Створення багаторівневої розмітки тексту і побудову синтаксичної розмітки, що враховує складні залежності між парами лем доцільно здійснювати за допомогою сучасних лінгвістичних пакетів, що мають відповідний функціонал для парсингу текстів [116]. Наразі такі можливості для

англійської мови присутні у цілого ряду відомих лінгвістичних пакетів, що відрізняються, у першу чергу, базовою мовою програмування. Найбільш популярними з них вважаються [117] такі: NLTK для мови Python [118], [119], DKPro Core [120] та Stanford NLP [121] для мови Java, SharpNLP для .NET [122], MeTA для C++ [123], Apache OpenNLP для Java або C++ [124] тощо.

Фільтрацію вербального шуму пропонується забезпечити за допомогою таких операцій: заміна займенників на відповідні до них іменники; вилучення шумових зв'язків; вилучення шумових слів; вилучення стоп-слів.

Розглянемо колізію при знаходженні ключових слів, яка особливо актуальна для невеликих текстів, якими можуть бути відносно невеликі блоги чи повідомлення в соціальних мережах. Концептуальним розвитком блогів, обумовленим їх широкою соціалізацією, є мікроблоги, які мають ряд характерних особливостей: обмежена довжина повідомлень, велика частота публікацій, різноманітна тематика, різні шляхи доставки повідомлень тощо.

Очевидно, що розглянутий у п. 1.5 автоматизований пошук найбільш значущих термінів (слів) з потоку повідомлень, що генерується співтовариством Twitter, має практичне значення як для визначення інтересів різних груп користувачів, так і для побудови індивідуального профілю кожного з них.

Однак треба зазначити, що класичні статистичні методи пошуку ключових термінів, засновані на аналізі колекцій документів, малоефективні в даному випадку. Це обумовлено надзвичайно малою довжиною повідомлень (до 140 символів), їх різноманітною тематикою і відсутністю логічного зв'язку між собою, а також великою кількістю рідко використовуваних та / або маловідомих абревіатур, скорочень, елементів специфічного мікросинтаксису.

Оскільки в Twitter не існує простого і зручного способу для групування «твітів» різних користувачів за тематикою, співтовариство користувачів користується власним рішенням – використанням хештегів. Останні схожі на

інші приклади використання тегів (наприклад, для анотування записів у звичайних блогах) і дозволяють додати «твіти» в якусь категорію.

Хештеги починаються з символу «#», за яким слідує будь-яке поєднання дозволених в Twitter символів без пробілів; найчастіше це слова або фрази, в яких перша буква кожного слова наведена з великої літери. Вони можуть зустрічатися у довільній частині «твіта», часто користувачі просто додають символ «#» перед будь-яким словом. При додаванні в повідомлення хештега воно буде відображатися при пошуку в потоці повідомлень Twitter за цим хештегом.

До неофіційних, але загальноприйнятих правил використання хештегів відноситься вибір для них термінів, релевантних темі повідомлення, а також додавання лише невеликої кількості їх в одне повідомлення. Це дозволяє розглядати хештеги в якості термінів, які з достатнім ступенем імовірності відображають загальну тематику повідомлення.

В останні роки було запропоновано достатньо багато підходів, які дозволяють проводити аналіз наборів документів різного розміру і вилучати з них ключові терміни, що складаються з одного, двох і більше слів.

Найважливішим етапом пошуку ключових термінів є розрахунок їх ваг в аналізованому документі, що дозволяє оцінити їх значущість відносно один одного в даному контексті. Для вирішення цієї задачі існує множина методів, які умовно діляться на 2 групи: потребують навчання і не потребують навчання. Під навчанням мається на увазі необхідність попередньої обробки вихідного корпусу текстів з метою пошуку інформації про частоту зустрічальності термінів у всьому корпусі. Іншими словами, для визначення значущості терміна у певному документі необхідно спочатку проаналізувати всю колекцію документів, до якої він належить. Альтернативним підходом є використання лінгвістичних онтологій, які є більш-менш наближеними моделями існуючого набору слів заданого мови [45]. На базі обох підходів були створені системи для

автоматичного пошуку ключових термінів, однак у цьому напрямку постійно ведуться роботи з метою підвищення точності і повноти результатів, а також з метою використання методів пошуку інформації в тексті для вирішення нових завдань [82].

Краща якість обробки тексту досягається лінгвістичними методами або ж при їх комбінації зі статистичними, тому систему автоматичного пошуку ключових фраз з тексту природною мовою будемо розробляти з використанням морфологічного словника (лексикону) і синтаксичних правил. Зокрема результати парсингу природних мов за допомогою сучасних лінгвістичних пакетів дозволяють на доступному програмному рівні оперувати синтаксичними зв'язками між словами окремого речення [102].

Застосовуючи підхід до пошуку ключових слів, що базується на використанні додаткової інформації про складні залежності між членами англомовного речення, можна знаходити ключові слова в тексті повідомлень мікроблогів і порівнювати їх з хештегами заданими автором повідомлень [125].

Відома у інформатиці та криптографії колізія хеш-функції – це рівність значень хеш-функції на двох різних блоках даних. Колізія при знаходженні ключових слів – це рівність значень частоти для двох чи більше кандидатів в ключові слова, причому вибрати в якості ключових потрібно тільки частину з них. Здебільшого така задача актуальна для текстів невеликого розміру, таких як анотація або записи мікроблогів [126].

Розглянемо приклад – при пошуку ключових слів англомовного тексту для тексту "Variability management in software product lines using adaptive object and reflection" [127] отримано та наведено в таблиці 3.1 список ключових слів, що відсортовані за кількістю зв'язків з іншими словами тексту (частотою зв'язків) за спаданням [126].

Колізія виникає тоді, коли потрібно вибрати зі списку потенційних ключових слів тільки перших N слів з найбільшою частотою, які і будуть

вважатися ключовими словами. Для наведеного прикладу тексту, якщо потрібно 10 ключових слів, то перших вісім вибрати легко. Це будуть слова: model, line, product, reuse, variability, approach, base, software. Тоді необхідні останні два слова потрібно вибрати між трьома словами з однаковою частотою п'ять: increase, management, process. Тому для невеликих текстів розв'язання такої колізії є актуальною задачею [126].

Таблиця 3.1 – Ключові слова і кількість зв’язків для них

Слово	Кількість зв'язків (частота)	Слово	Кількість зв'язків (частота)	Слово	Кількість зв'язків (частота)	Слово	Кількість зв'язків (частота)
model	9	plan	4	challenge	2	savings	2
line	7	vehicle	4	creation	2	support	2
product	7	define	3	derive	2	variant	2
reuse	7	design	3	development	2	activity	1
variability	7	diagram	3	domain	2	adaptive	1
approach	6	extract	3	feature	2	architecture	1
base	6	identify	3	implement	2	brazilian	1
software	6	mechanism	3	key	2	finally	1
increase	5	offer	3	launcher	2	hypothetic	1
management	5	reflection	3	productivity	2	large-scale	1
process	5	step	3	propose	2	object	1
aim	4	study	3	quality	2	space	1
issue	4	benefit	2	satellite	2	specific	1

Для зменшення розглянутої колізії можна застосувати такі підходи [126]:

- Відсортувати множину слів з однаковою частотою за частотою їх появи в певному корпусі. Відносна значимість термінів в аналізованому контексті визначається за допомогою даних про частоту їх використання в якості ключових в інтернет-енциклопедії Вікіпедія [128]. Робота алгоритму заснована на розрахунку "інформативності" кожного терміна, тобто оцінки ймовірності того, що він може бути обраний ключовим в тексті [82]. Такий підхід є досить точним, але потребує попереднього аналізу корпусу.

- Відсортувати множину слів з однаковою частотою за частотою їх появи в частотному словнику словоформ для даного тексту. Такий підхід, що узагальнює слова до словоформ, не потребує попередньої обробки корпусу текстів і легкий в реалізації, але точність його невелика.

- Перевіряти список ключових слів на зв'язність, тобто враховувати парні залежності для різних типів речень [103]. Вибирати ключовими нові слова, які мають більшу сумарну кількість зв’язків з тими словами, що раніше потрапили до списку ключових. Цей підхід не потребує корпусу і його можна реалізувати засобами лінгвістичного пакету.

- Вибирати спочатку іменники, потім дієслова, а потім інші частини мови. Оскільки головні члени речення зазвичай бувають іменниками та дієсловами, то вибиратися будуть саме ті слова, які потенційно можуть належати до множини ключових. Також, для іменників можна спочатку вибирати власні назви, тому що одним із запитань, на які повинні відповідати ключові слова, є таке: з якими назвами організацій, персон, географічних областей тощо асоціюється стаття [19].

Комбінуючи два останні підходи для зменшення колізії можна покращити результати знаходження ключових слів для невеликих за розміром текстів [126].

Пропонується побудувати метод пошуку ключових слів на основі комбінованого підходу, зокрема на першому етапі перевіряти ключові слова з однаковою частотою на зв'язність. На другому етапі, якщо у блоці потенційних ще залишилися ключові слова з однаковою частотою, вибираються спочатку іменники, потім дієслова, а потім інші частини мови. Наступні експериментальні дослідження мають підтвердити доцільність використання такого підходу [126].

Комбінований підхід для зменшення колізії можна використовувати як додатковий модуль, що покращить результати знаходження ключових слів для методу пошуку ключових слів англомовного тексту на основі інструментальних

засобів обраного лінгвістичного пакету, що розробляється, а також для інших алгоритмів знаходження ключових слів [126].

Вербальний шум або шумові слова – термін з теорії пошуку інформації за ключовими словами. Це такі слова, які не несуть смислового навантаження, тому їх користь та роль для пошуку не суттєва [129].

В процесі обробки проводиться виключення з досліджуваного тексту слів, які за визначенням не можуть бути значущими тому, що складають «шум». На відміну від ключових ці слова називаються нейтральними або стоповими (стоп-словами). Такими є слова, що відносяться до службових частин мови, а також займенники [15].

За результатами порівняльного аналізу відомих підходів пропонується досягти зменшення кількості шумових слів у межах додаткового методу, що має доповнювати основний метод, за допомогою чіткої послідовності таких підходів: заміна займенників на відповідні до них іменники; вилучення словосполучень із типами зв’язків, які не несуть суттєвого смислового навантаження; вилучення слів, які відносяться до неінформативних частин мови; вилучення слів, які відносяться до списку стоп-слів [130].

3.2 Мово-незалежні особливості розроблених методів пошуку ключових слів тексту

Тексти на будь-якій природній мові складаються з двох частин: того, що обов'язково має бути виражене за законами даної мови, і того, що відображає специфіку тематики тексту і стилю автора. Ці складові називаються відповідно тематично нейтральною і тематично маркованою лексикою. Знаходження обох груп лексики – крок на шляху до визначення змістовної віднесеності тексту. Даний процес дозволяє як наблизитися до змісту тексту, так і скласти певну думку про своєрідність лексики автора і його мови.

При обробці текстового файлу виникає цілий ряд складнощів, тому існують певні вимоги до вихідних текстів.

У більшості випадків ці вимоги визначаються нормами поліграфії та інтуїтивними міркуваннями:

- вважається, що текст побудований правильно (наприклад, дві коми не можуть йти підряд);

- дефіс відразу після слова в кінці рядка говорить про перенесення слова, а після нього в рядку не можуть йти ніякі символи, у тому числі пропуск;

- тире після пропуску є поодиноким символом і праворуч від нього також повинен бути пропуск;

- тире після символу-літери при наявності наступної літери, розташованої в тому ж рядку, говорить про словосполучення і ніколи не виділяється пропусками.

Прийняття таких правил полегшує обробку текстів, але ускладнює їх набір на комп'ютері, оскільки вимагає постійного контролю матеріалу, що вводиться [32].

Одним із завдань добування інформації з тексту є пошук ключових термінів, що з певною мірою достовірності відображають тематичну спрямованість документа. Автоматичний пошук ключових термінів можна визначити як автоматичний пошук найбільш важливих тематичних термінів у документі. Він є однією з підзадач більш загального завдання – автоматичної генерації ключових термінів, для якої знайдені ключові терміни не обов'язково повинні бути присутніми в даному документі [56].

В останні роки було створено значну кількість підходів, які дозволяють проводити аналіз наборів документів різного розміру і знаходити ключові терміни, що складаються з одного, двох і більше слів. При цьому визначальним етапом пошуку ключових термінів є розрахунок їх ваг в аналізованому

документі, що дозволяє оцінити значущість потенційних термінів відносно один одного в даному контексті.

Закони статистики погано працюють на документах невеликого розміру (типу анотацій, мікроблогів тощо), оскільки в них частоти всіх однослівних і багатослівних ключових термінів приблизно рівні і прагнуть до одиничного входження в рамках контексту документа. Але завдяки заміні займенників відповідними іменниками в запропонованих основному та додатковому методах, причому перед використанням статистичних методів, можливо збільшити частоту потенційних ключових слів.

Ручний аналіз є дорогим процесом. Коли величезні масиви тексту повинні бути проаналізовані або коли можливість обробки мови повинна бути інтегрована в пристрої, автоматичний аналіз є найбільш зручним підходом, який потрібно використати. Після того, як досить великий обсяг аналізованих вручну даних отримано, його можна використовувати для підготовки статистичних моделей з використанням алгоритмів машинного навчання, власне оцінювати їх результати на основі створеного вручну “золотого стандарту”. Навіть для неконтрольованих алгоритмів машинного навчання, тобто алгоритмів, які не потребують будь-яких навчальних даних, оцінка результатів шляхом порівняння зі створеним вручну “золотим стандартом” є незамінною. У контексті досліджень, автоматичний аналіз часто вкладається в експериментальну установку, яка ітеративно повторюється з різними варіаціями компонентів аналізу і їх конфігурацій для досягнення оптимальних результатів [131].

При виконанні ручного аналізу, керівні принципи анотування визначають, як ідентифікувати явище, що має бути анотоване, які категорії використовувати для його класифікації. Ці принципи часто підтримуються у вигляді простих текстових документів, які можуть бути зрозумілі людині анотаторові, але які не є машинозчитуваними. Для автоматичного аналізу, система анотацій типів грає аналогічну роль. Це офіційний документ, який може бути оброблений

комп’ютером, що описує типи анотацій, їх особливості та набори тегів. Типова система анотації типу визначає тип Токена, який несе в собі функцію частини мови, що приймає певне значення, у першу чергу іменник. Системи анотації типів визначається по-різному, в залежності від формалізму, використовуваного для подання фактичних анотацій, зазвичай у залежності від конкретної структури інструменту або від фреймворку, для якого вони написані. Іноді стандартні формати, такі як XML-схеми або мова Web-онтологій (OWL) використовуються для визначення системи типу анотацій [131].

При проектуванні систем анотації типів, деякі аспекти, як правило, не визначені, зокрема, такі аспекти, які явно специфіковані в керівних принципах анотування. Системи анотації типу, однак, часто призначені для багаторазового використання в різних контекстах, наприклад, для різних мов або вченими різних шкіл, які класифікують частини мови по-різному або з використанням різного поділу. Отже, система анотації типу зазвичай визначає, що функція частини мови існує у Токена, але не визначає значення, яке вона може приймати. Якщо формалізм, що лежить в основі системи анотації типу підтримує успадкування або деяку іншу форму розширюваності, деякі розробники можуть ввести спеціалізацію Токена, наприклад, FooToken, який приймає тільки теги частини мови з гіпотетичного набору тегів Foo [131].

У той час, як ручний аналіз, в основному, робиться лінгвістами або іншими експертами з гуманітарних наук, люди, що працюють над автоматичним аналізом, часто мають досвід роботи в області комп'ютерних наук, або, останнім часом, області комп'ютерної лінгвістики [131].

Фреймворк обробки (Processing framework) – це типова частина програмного забезпечення, яке дозволяє створювати інтероперабельні компоненти аналізу і полегшує їх використання. Так як система мови складається з декількох підсистем, таких як синтаксис і морфологія, існують спеціалізовані інструменти та алгоритми для цих різних підсистем. Часто такі

інструменти є одиничними – вони безпосередньо обробляють вхідний документ і знаходять результат аналізу, наприклад, TreeTagger може знаходити частину мови, леми і здійснювати поділу на частини. Stanford Parser може знаходити теги частини мови, синтаксичні складові і відносини залежностей. Обидва інструменти поставляються з основними вбудованими можливостями попередньої обробки, такими як токенізатор і детектор границь речення [131].

Бажано, щоб існувала можливість запустити більше одного інструменту одночасно, наприклад, тому що вони аналізують різні мовні категорії (див. рисунок 3.1). Один компонент може також отримати користь з аналізу, отриманого іншим компонентом (див. рисунок 3.2) [131].

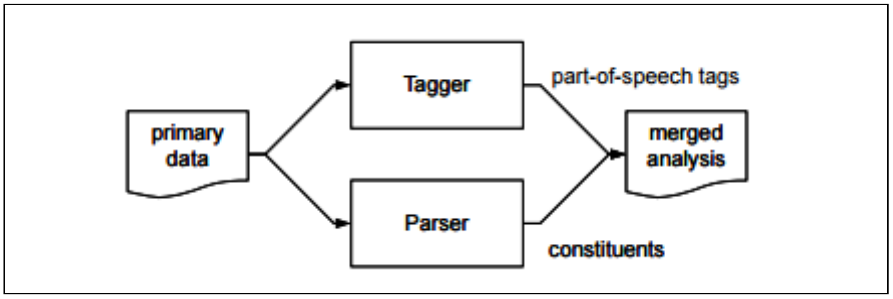


Рисунок 3.1 – Запуск декількох інструментів на одних і тих же даних і злиття результатів в єдину модель

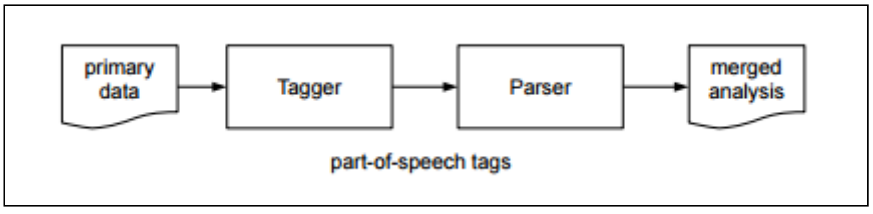


Рисунок 3.2 – Запуск декількох інструментів на одних і тих же даних, один інструмент працює з результатами, отриманими іншим інструментом і злиття результатів в єдину модель

Більшість з цих автономних інструментів аналізу не відразу сумісні. Кожен з них використовує і виробляє різні формати даних, часто навіть категорії

на різних рівнях деталізації. Спрямовувати результат роботи одного інструменту в наступний або, навіть, просто створювати файл аналізу з формою результатів для різних інструментів не просто. Так званий сполучний код повинен бути написаний для перетворення даних, що передаються між інструментами багаторазово. При цьому фреймворк обробки забезпечує загальну модель даних, в якій результати аналізу можуть бути записані і з якої вони можуть бути відновлені. Отже, немає необхідності писати сполучний код для перетворення між специфічними форматами даних довільних інструментів, але тільки перетворення із загальної моделі даних (див. рисунок 3.3) [131].

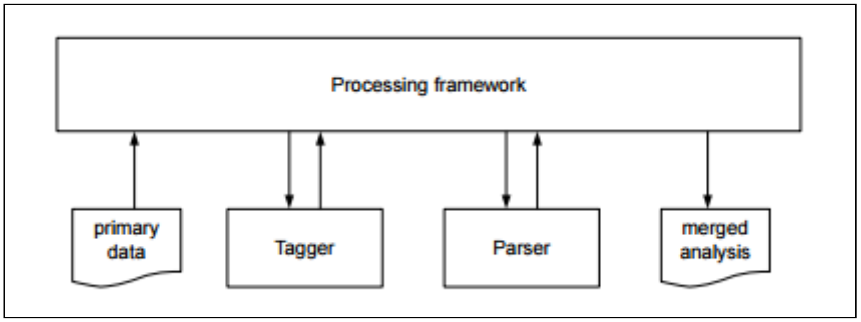


Рисунок 3.3 – Запуск декількох інструментів на одних і тих же даних, інтегрованих у фреймворк обробки і злиття результатів в єдину модель

Коли мовний інструмент інтегрований у такий фреймворк, то він більше не інструмент, а компонент аналізу. Інтеграція здійснюється шляхом реалізації фреймворкової оболонки навколо інструменту, яка піклується про життєвий цикл компоненту (ініціалізація / конфігурація, виконання, знищення), а також перетворення між загальної моделлю даних і форматом даних конкретного інструменту. Після того, як така оболонка буде реалізована, цей інструмент може бути використаний з фреймворком з метою поєднання з іншими інструментами в робочому процесі аналізу. Деякі структури пропонують додаткові функції, такі як паралельне виконання або робочий процес редакторів, на який реалізатор оболонки може не звертати увагу [131].

Фреймворк обробки може забезпечувати різні рівні взаємодії між компонентами аналізу. На найнижчому рівні, він забезпечує набір структур даних, які зберігають вихідні дані разом з результатами аналізу, що використовуються усіма компонентами аналізу. На більш високому рівні, фреймворк може запропонувати підтримку для систем анотації типів. У колекціях компонентів, всі компоненти аналізу повинні дотримуватися загальної системи типу анотацій. На жаль, це також означає, що компоненти з різних колекцій не завжди сумісні один з одним на цьому рівні, якщо самі компоненти не допускають конфігурацію спільних типів. На більш високому рівні аналізатор не тільки повинен знати про тип, але також повинен вміти інтерпретувати конкретний набір тегів, вироблених тагером [131].

Сучасні лінгвістичні пакети використовують динамічний підхід до вибору і отримання ресурсів. Цей підхід реалізується в момент вибору, що дозволяє приймати до уваги характеристики оброблюваних даних тільки під час виконання. При такому підході не потрібно у явному вигляді настроювати компоненти аналізу, що найбільш часто є обов'язковими параметрами для необхідних ресурсів. У більшості випадків компоненти можуть бути використані в робочому процесі аналізу без будь-якої додаткової конфігурації, що значно спрощує збірку робочого процесу, зокрема, для не досвідчених програмістів. Якщо потрібно, конфігурація за замовчуванням може бути змінена, наприклад, щоб вибрати інший варіант ресурсу або використовувати користувацьку локальну модель замість моделі зі сховища. Підхід являє собою крок в напрямку декларативної побудови робочих процесів, вказуючи, що потрібно робити, наприклад, запустити визначення частини мови через OpenNLP, але без зазначення, як це зробити, зокрема, яку модель використовувати, де і як її отримати тощо [131].

Невід'ємною частиною підходу до побудови методів пошуку ключових слів, що пропонуються, є послідовна упаковка ресурсів в артефакти ресурсів,

які можуть бути розповсюджені через сховища артефактів. Пакетні ресурси портативні, що дозволяє легку і автоматичну їх установку на комп'ютер користувача, або в іншу систему, в якій виконується робочий процес аналізу, наприклад, таких як обчислювальний кластер. Упаковка ресурсів розроблена таким чином, що не є специфічною для певного інструменту або фреймворка. Замість цього ресурси можуть бути повторно використані, і якщо є два незалежних пакети ресурсу, то буде використаний один, для економії місця в сховищах [131].

Робочий процес аналізу відтворюється шляхом задання високорівневого опису процесу, який містить інформацію, достатню для побудови експериментального середовища і запуску робочого процесу. Такий підхід є покращеним в порівнянні з відомими фреймворками, які дозволяють будувати робочі процеси аналізу і обмінюватися ними, але залишають створення середовища виконання користувачу. Також існує можливість змінювати структуру робочих процесів динамічно, на основі їх параметризації, наприклад в експерименті, де установка запускається з великою кількістю параметрів налаштувань. Така зміна параметрів може використовуватися, щоб знайти оптимальну конфігурацію для експериментальної установки [131].

З огляду на вищезазначене, лінгвістичний пакет DKPro Lab є досить зручним інструментом для реалізації запропонованого підходу. Існуючий код може бути легко обгорнутий у фреймворк. Користувачі можуть встановлювати фреймворки поступово і застосовувати їх до своєї предметної області, наприклад, для реалізації користувальницьких категорій завдань, розмірів параметрів або звітів. На відміну від інших систем документообігу, така база зосереджена в явному вигляді на програмному створенні складних робочих процесів і, що важливо, дозволяє використання існуючих навичок і інструментів програмування для платформи Java. Це полегшує створення експериментальних робочих процесів, їх налагодження та рефакторинг.

Зокрема, налагодження не обмежується тільки до рівня робочого процесу, але може плавно йти вниз до рівня окремих компонентів аналізу або, навіть, далі, аж до бібліотек [131].

Застосувавши незалежні від структури фреймворка параметри робочого процесу, можна проводити експерименти на кількох фреймворках і включати довільні кроки обробки. Це полегшує автоматизацію допоміжних кроків, які, зазвичай, виконуються вручну і, отже, можуть бути помилковими, наприклад – направити вихід одного етапу експерименту в наступний. DKPro має найбільш повну колекцію портативних компонентів аналізу, тобто компонентів, що не залежать від веб-сервісів. Інструменти аналізу, вбудовані в DKPro зосереджують увагу на різних мовах і напрямках [131]. Отже, DKPro забезпечує мовно-незалежні особливості методів пошуку ключових слів і надає користувачам багатий вибір для реалізації запропонованого підходу.

3.3 Алгоритм пошуку ключових слів на основі двох послідовних методів

З урахуванням самостійної цінності основного та додаткового методів пошуку ключових слів, що було обґрунтовано у пп. 3.1 та 3.2, виникає необхідність у розробці гнучкого алгоритмічного забезпечення. Тому пропонуються такий узагальнений алгоритм:

а) створення багаторівневої розмітки тексту [132], [133];

б) отримання синтаксичної розмітки, що враховує складні залежності між парами лем [132], [133];

в) виключення неінформативних для аналізу типів зв’язків – вилучаються слова, які не відносяться до самостійних частин мови [132], [133];

г) заміна займенників в отриманих парах на відповідні до них іменники [132], [133];

д) отримання пар ключових слів [132], [133];

Примітка: на даному кроці за переліком термінів, отриманих на попередньому кроці, можлива побудова семантичного графа [134]. Семантичний граф являє собою зважений граф, вершинами якого є терміни документа, наявність ребра між двома вершинами означає той факт, що терміни семантично пов'язані між собою, вага ребра є чисельним значенням семантичної близькості двох термінів, які з'єднує дане ребро [135];

е) розбиття пар на окремі слова і визначення кількості зв’язків; даний крок дозволяє збільшити шанси отримання більш вагомих ключових слів, оскільки багато з них замінюються в наступних реченнях займенниками для уникнення повторів [132], [133];

ж) вибір перших n слів з найбільшою кількістю зв’язків, де n – кількість потрібних ключових слів [132], [133].

На рисунку 3.4 представлено схему вищенаведеного алгоритму аналізу тексту.

Схему алгоритму фільтрації вербального шуму і відображення результатів пошуку ключових слів представлено на рисунку 3.5. На вхід алгоритм отримує словосполучення (phrases), що отримано з тексту в результаті його розбору. Далі словосполучення розбиваються на окремі слова і підраховується кількість зв'язків для кожного з них.

Наступні кроки призначенні для фільтрації вербального шуму: заміна займенників на відповідні до них іменники; вилучення словосполучень із типами зв’язків, які не несуть суттєвого смислового навантаження; вилучення слів, які відносяться до неінформативних частин мови; вилучення слів, які відносяться до списку стоп-слів. Після кожного з цих кроків фільтрації ключові слова, що залишилися, виводяться на екран.

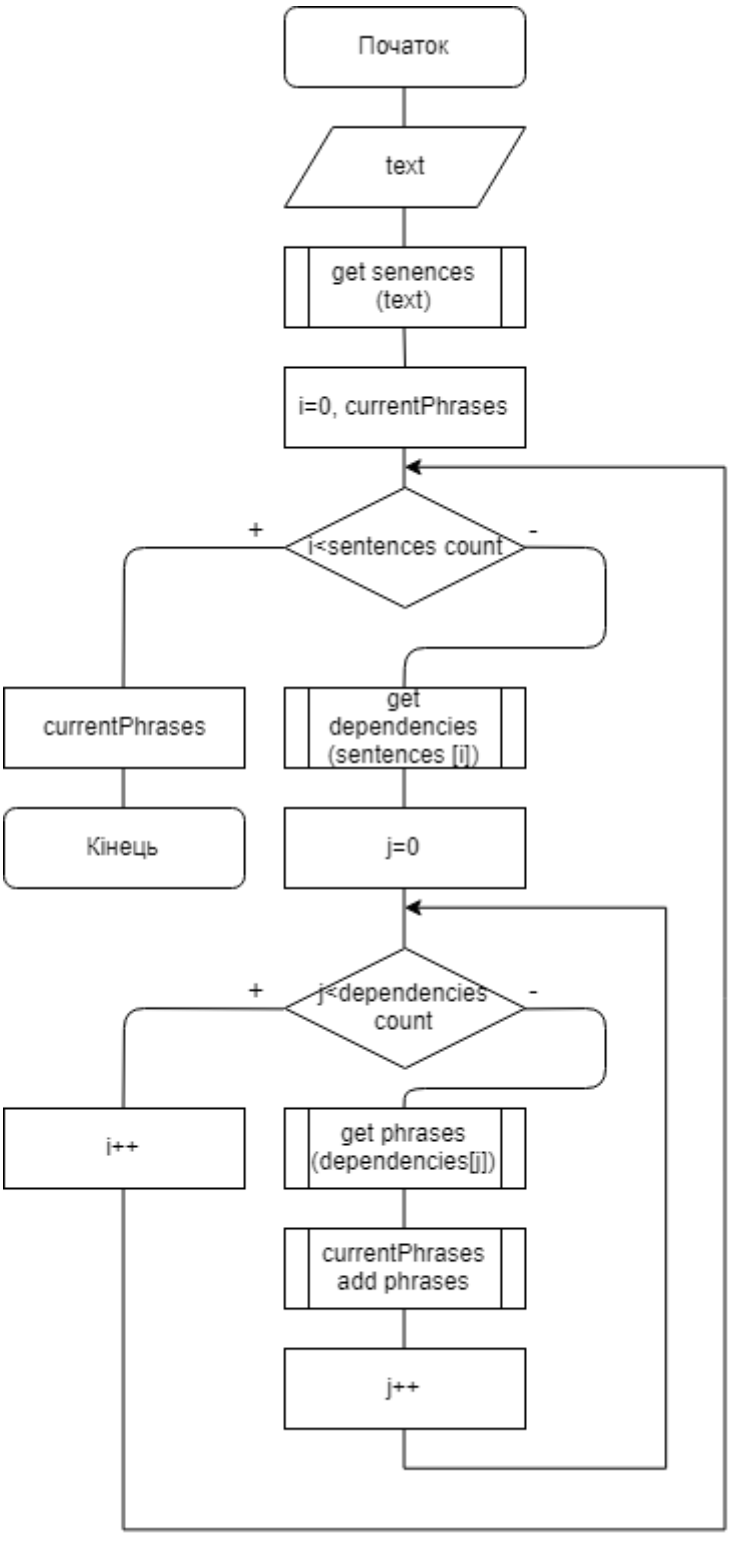


Рисунок 3.4 – Схема алгоритму аналізу тексту

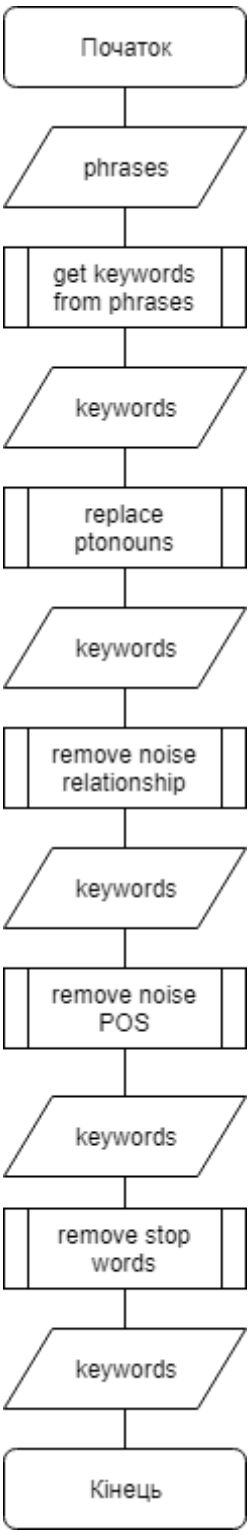


Рисунок 3.5 – Схема алгоритму фільтрації вербального шуму і відображення результатів

На рисунку 3.6 представлено UML діаграму класів. Зокрема, клас NLPKRootScreen відповідає за відображення інтерфейсу користувача і

зв’язаний з класом NLPKTextProcessing, що відповідає за логіку пошуку ключових слів. Як видно з діаграми, клас NLPKTextProcessing зберігає масив словосполучень (currentPhrases) для того, щоб після розбору тексту можна було багаторазово робити пошук ключових слів, при цьому застосовуючи всі модулі фільтрації вербального шуму або тільки деякі з них. Елементами масиву словосполучень є об’єкти типу NLPKPhrase, в яких зберігається інформація про головне і залежне слова, а також тип зв'язку між ними.

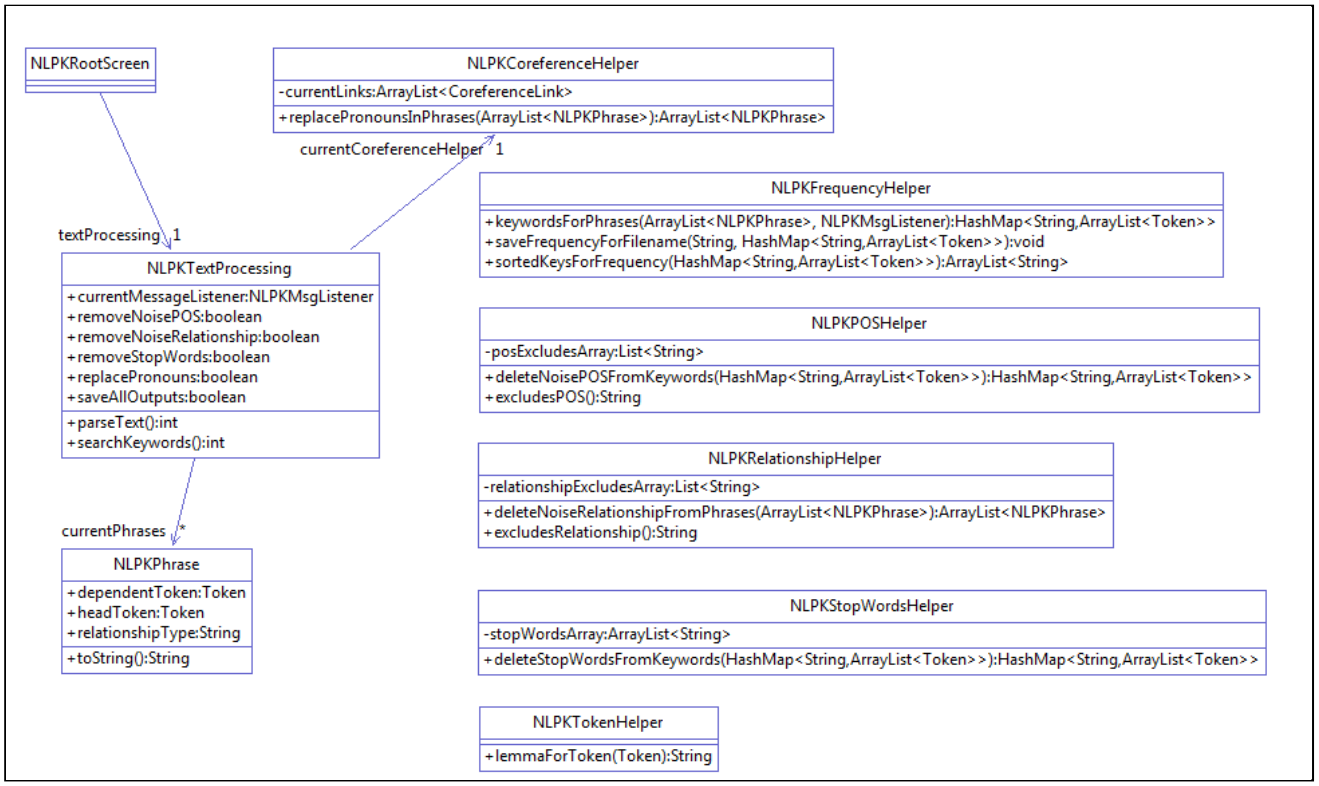


Рисунок 3.6 – UML діаграма класів

Головне і залежне слово є об'єктами типу Token, що зберігає інформацію про слово в тексті. Такою інформацією є: порядковий номер в тексті першого і останнього символу слова; лінк на попереднє і наступне слово; основна форма слова; тег частини мови тощо. Також, клас NLPKTextProcessing створює об'єкт класу NLPKCoreferenceHelper на етапі розбору тексту. У цьому об'єкті зберігається інформація про займенники, які зустрічаються в тексті та слова, з

якими вони пов'язані. Зазвичай пов'язані слова можуть бути іменниками або іншими займенниками. Якщо програмно визначено той факт, що займенник зв'язаний тільки з іншими займенниками, то заміна не відбувається.

На рисунку 3.7 зображена діаграма зв'язків між класами. Основним класом є NLPKTextProcessing, який відповідає за оброблення тексту і пошуку ключових слів.

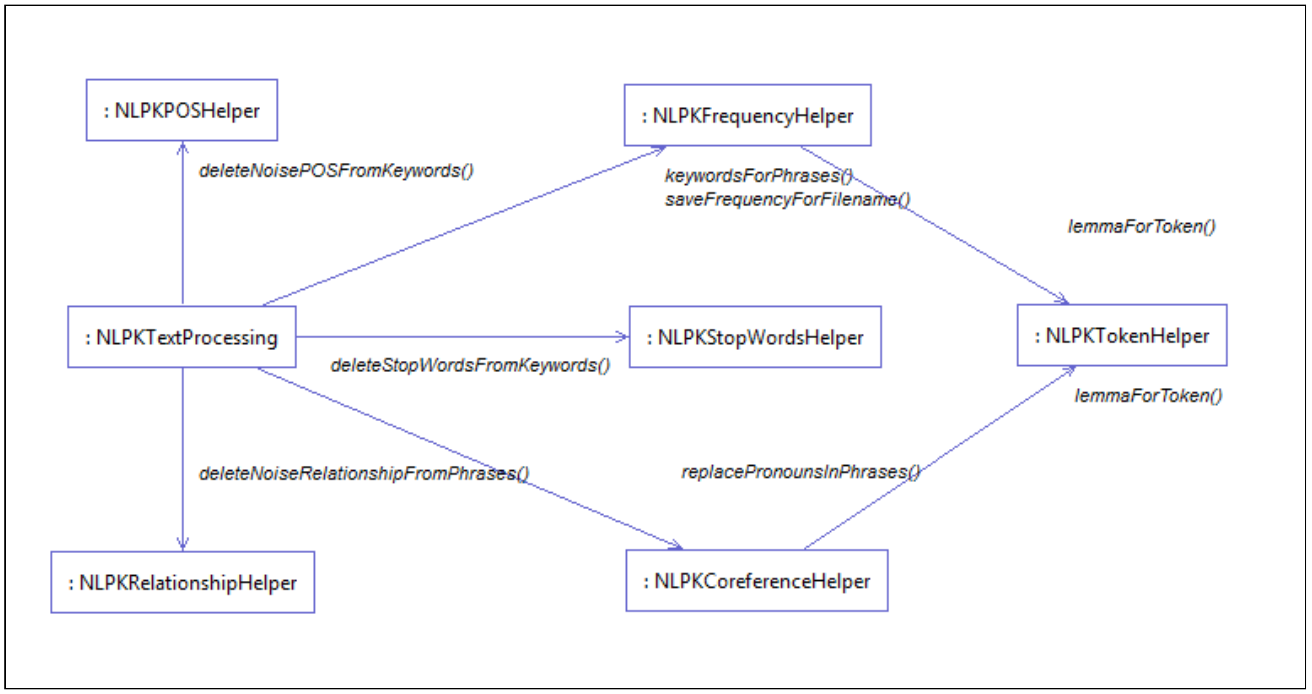


Рисунок 3.7 – Діаграма зв'язків

Клас NLPKTextProcessing використовує клас NLPKFrequencyHelper для підрахунку кількості зв'язків для певного слова з тексту. NLPKFrequencyHelper, в свою чергу, пов'язаний з класом NLPKTokenHelper, робота якого полягає в знаходженні основної форми відповідного слова. Це потрібно для того, щоб зв'язки накопичувалися для унікальних слів, без повторів. Також, NLPKTextProcessing запускає, по черзі, модулі фільтрації вербального шуму:

- модуль заміни займенників (NLPKCoreferenceHelper);
- модуль вилучення словосполучень, які не несуть суттєвого смислового навантаження (NLPKRelationshipHelper);

- модуль вилучення слів, які відносяться до неінформативних частин мови (NLPKPOSHelper);
- модуль вилучення стоп-слів (NLPKStopWordsHelper).

Модуль заміни займенників NLPKCoreferenceHelper пов'язаний з класом NLPKTokenHelper, оскільки займенники у словосполученнях мають замінюватися на іменники в основній формі слова.

3.4 Спосіб автоматичного пошуку ключових слів з використанням технології DKPro Core

На основі розробленого у п.3.3 алгоритмічного забезпечення пропонується спосіб автоматичного пошуку ключових слів з використанням технології DKPro Core [120], який, за рахунок введення нових операцій та їх послідовності дає можливість шукати ключові слова для одного тексту без аналізу всіх текстів корпусу, що призводить до збільшення швидкодії [135].

Вибір лінгвістичного пакету DKPro Core обумовлено перевагами його базової мови програмування Java та досвідом автора у розробці професійних Java-додатків. Варто зазначити, що принципових відмінностей способу автоматичного пошуку ключових слів, що пропонується, при виборі інших лінгвістичних пакетів / програмних платформ не було виявлено.

Спосіб автоматичного пошуку ключових слів з використанням технології DKPro Core включає знаходження словосполучень кандидатів (у ключові слова) зі застосуванням аналізатора DKPro, розбір словосполучень кандидатів в набір слів, побудову семантичного графа між словосполученнями кандидатів, відбір найпопулярніших кандидатів як ключових слів на основі кількості зв'язків між словосполученнями кандидатів, в яких ключові слова отримані шляхом обробки семантики. У запропонованому способі словосполучення кандидатів знаходять з текстового документу, який за допомогою аналізатора DKPro перетворюють в набір речень і зв'язків між членами цих речень, на основі синтаксичного

розбору кожного речення розбивають його на словосполучення кандидатів, для яких визначають головне і залежне слово та зв'язок між ними, будують семантичний граф між словосполученнями кандидатів, виключають словосполучення кандидатів, зв'язки яких внесені у попередньо заданий перелік не інформативних для семантичного аналізу типів зв'язків, займенники в словосполученнях кандидатів замінюють на відповідні до них іменники, словосполучення кандидатів розбивають на окремі слова, для кожного з яких знаходять частину мови і лему, визначають кількість семантичних зв'язків для окремого слова, виключають ключові слова, які належать до попередньо визначеного переліку не інформативних для семантичного аналізу частин мови, виключають ключові слова, які належать до попередньо визначеного списку стоп-слів [135].

Описаний спосіб можна реалізувати на пристрої, представленому на рисунку 3.8 [135].

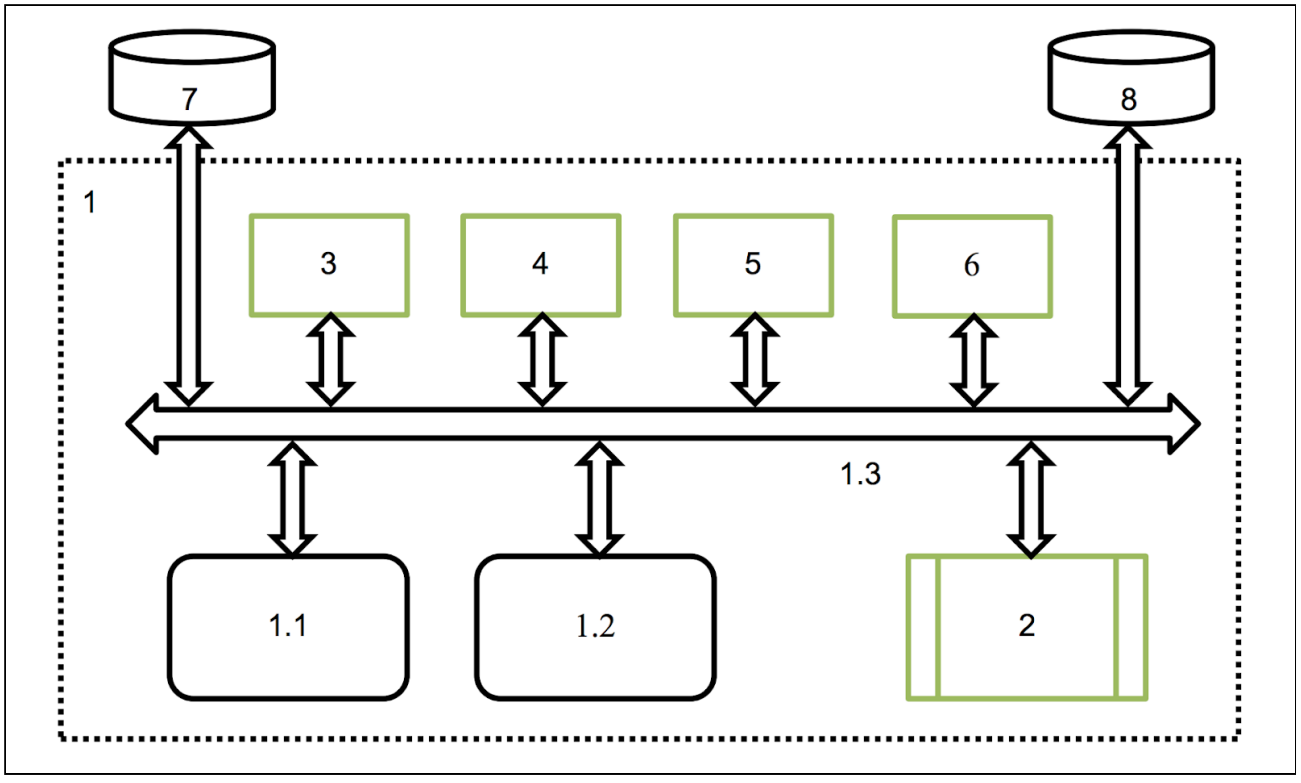


Рисунок 3.8 – Пристрій для пошуку ключових слів

На даному рисунку використані такі позначення [135]:

1. Персональний комп’ютер з апаратними складовими:

1.1. Центральний процесор.

1.2. Запам’ятовуючі пристрої – ОЗП та ПЗП.

1.3. Загальна шина даних (канал зчитування та передачі даних).

Програмні складові, необхідні для реалізації способу [135]:

2. Аналізатор DKPro, здатний виокремити з тексту кожне речення і розбити його на словосполучення кандидатів, для яких визначити головне і залежне слово та зв’язок між ними.

3. Модуль виключення словосполучень кандидатів з не інформативними типами зв’язків.

4. Модуль заміни займенників на іменники в словосполученнях кандидатів.

5. Модуль визначення частини мови, леми і кількості семантичних зв’язків для кожного окремого слова з словосполучень кандидатів.

6. Модуль виключення ключових слів, які належать до попередньо визначених: а) переліку не інформативних для семантичного аналізу частин мови та б) списку стоп-слів.

Зовнішні запам’ятовуючі пристрої, де зберігаються:

7. Колекція текстів.

8. Ключові слова.

На вхід каналу зчитування та передачі даних 1.3 надходить текст, а також попередньо складені переліки не інформативних для семантичного аналізу типів зв’язків і частин мови, список стоп-слів, що розміщені на зовнішньому електронному носії інформації 7 [135].

За допомогою аналізатора DKPro 2, текст перетворюють в набір речень і зв'язків між членами цих речень, на основі синтаксичного розбору кожного речення розбивають його на словосполучення кандидатів, для яких визначають

головне і залежне слово та зв’язок між ними, використовуючи ресурси центрального процесору 1.1, будують семантичний граф між словосполученнями кандидатів, який зберігається на запам’ятовуючих пристроях 1.2 [135].

У модулі виключення словосполучень кандидатів з не інформативними типами зв’язків 3 на рівні центрального процесору 1.1 перевіряється тип зв'язку кожно словосполучення. Якщо тип зв'язку відноситься до попередньо заданого переліку не інформативних для семантичного аналізу типів зв’язків, таке словосполучення видаляється, а ті словосполучення кандидатів, що залишилися, зберігаються на запам’ятовуючих пристроях 1.2 [135].

У модулі заміни займенників 4 на рівні центрального процесору 1.1 знаходять іменники і перелік займенників, які заміняють ці іменники, за допомогою аналізатора DKPro 2. Після цього замінюють займенники в словосполученнях кандидатів, що зберігаються на запам’ятовуючих пристроях 1.2, на відповідні до них іменники [135].

Далі, словосполучення кандидатів, у модулі 5, на рівні центрального процесору 1.1 розбивають на окремі слова, для кожного з яких знаходять частину мови і лему, за допомогою аналізатора DKPro 2, та визначають кількість семантичних зв’язків для кожного окремого слова, що зберігаються на запам’ятовуючих пристроях 1.2 [135].

Після цього у модулі 6, на рівні центрального процесору 1.1 виключають ключові слова, які належить до попередньо визначеного переліку не інформативних для семантичного аналізу частин мови, та слова, які належить до попередньо визначеного списку стоп-слів [135].

Отримані ключові слова, на завершальному етапі, по каналу зчитування та передачі даних 1.3 зберігаються на зовнішньому електронному носії інформації 8 [135].

3.5 Висновки до розділу

Виявлено, що краща якість обробки тексту досягається лінгвістичними методами або ж при їх комбінації зі статистичними, тому систему автоматичного пошуку ключових фраз з тексту природною мовою слід розробляти з використанням морфологічного словника (лексикону) і синтаксичних правил.

Результати парсингу природних мов за допомогою сучасних лінгвістичних пакетів дозволяють на доступному програмному рівні оперувати синтаксичними зв’язками між словами окремого речення.

Застосовуючи підхід до пошуку ключових слів, що базується на використанні додаткової інформації про складні залежності між членами англомовного речення, можна знаходити ключові слова в тексті повідомлень мікроблогів і порівнювати їх з хештегами заданими автором повідомлень.

Розглянуто колізію при знаходженні ключових слів – це рівність значень частоти для двох чи більше кандидатів у ключові слова, причому вибрати в якості ключових з них потрібно тільки частину. Пропонується використовувати комбінований підходу для зменшення колізії, зокрема спочатку перевіряти ключові слова з однаковою частотою на зв'язність. На другому етапі, якщо у блоці потенційних ще залишилися ключові слова з однаковою частотою, вибираються спочатку іменники, потім дієслова, а потім інші частини мови.

Описано вербальний шум – це такі слова, які не несуть змістовного навантаження, тому їх користь та роль для пошуку не суттєва. Зменшення кількості шумових слів можна досягти за допомогою підходів: заміна займенників на відповідні до них іменники; вилучення словосполучень із типами зв’язків, які не несуть суттєвого смислового навантаження; вилучення слів, які відносяться до неінформативних частин мови; вилучення слів, які відносяться до списку стоп-слів.

Розглянуто алгоритми методу пошуку ключових слів і зменшення вербального шуму, а також реалізацію процесу пошуку ключових слів на пристрої з описом відповідних потоків даних між ЦП, ОЗП та ПЗП.

Отже, розроблено метод пошуку ключових слів, який, на відміну від існуючих, базується на знаходженні синтаксичних зв’язків між словоформами у реченнях англомовного тексту за допомогою технологічних можливостей парсингу сучасних лінгвістичних пакетів. Також удосконалено метод зменшення впливу вербального шуму на пошук ключових слів, який, на відміну від існуючих, побудовано на основі стенфордської класифікації зв'язків між лексичними одиницями речення згідно моделі пп. 2.3.

4 РОЗРОБКА ТА АПРОБАЦІЯ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ

У розділі розглянуто головні аспекти проектування та апробації нової інформаційної технології пошуку ключових слів англомовного тексту. На основі вибору сучасних інструментальних засобів запропоновано структуру інформаційної технології. Розроблено програмне забезпечення, орієнтоване на використання лінгвістичного пакету DKPro Core, спроектовано та реалізовано дружній до користувача інтерфейс. Шляхом проведення програмних експериментів, визначення та аналізу кількісних характеристик релевантності отриманих результатів пошуку ключових слів доведено переваги запропонованої інформаційної технології.

4.1 Вибір інструментальних засобів для реалізації інформаційної технології

Для розробки програмного забезпечення було обрано мову програмування Java. При цьому було взято до уваги основні переваги Java: дружній синтаксис, сумісність, незалежність, безкоштовність, велика кількість бібліотек. Розглянемо їх докладніше.

Дружній синтаксис – Java задумана так, щоб допомогти розробникам робити їх роботу легко і з мінімумом зусиль. При мінімумі знань можна навчитися мові Java з нуля і вже через короткий час писати код і, навіть, скомпілювати його у програму з вихідного коду [136]. Оскільки між Java, C і C ++ є багато схожого, програмістам було набагато легше переходити на нову мову. Адже не треба було абсолютно все вчити з нуля, багато конструкцій були їм уже зрозумілі. І це теж сприяло швидкому зростанню популярності Java серед програмістів [137].

Сумісність – компанії програмного забезпечення Sun і Oracle доклали значних зусиль, щоб систематично удосконалювати Java. Код, що написаний на старій версії, без проблем працює на оновленій. Адже немає нічого гіршого, ніж переписувати код, щоб він працював на новій версії – це абсолютно марна трата часу для розробника.

Незалежність – мова програмування Java з часу її проектування не залежить від платформи. Один і той же код можна використовувати для різних операційних систем. Завдяки цьому джава-розробник пише і запускає джава-програму в будь-якому середовищі за допомогою JVM (Java Virtual Machine) [136]. Тому той же самий відлагоджений код буде однаково працювати, наприклад, під операційними системами Windows, Linux і macOS. У той час, як на інших мовах програмування потрібно написати не один, а відразу три різних програми – під Windows, під Linux і під macOS [137].

Мова Java не була першою мовою для написання кросплатформенних додатків, але вона стала найпопулярнішою. Це зовсім не означає повну сумісність на різних платформах – відсутні бібліотеки або несумісні версії бібліотек нівелюють корисність Java коду. Не можна взяти код десктоп додатка, скомпільований під JRE 1.7 і запустити його на телефоні в Java ME. Тому, якщо код не працює, то, як правило, зрозуміло, у чому проблема. Але якщо використовуються правильні версії Java і достатньо пам'яті, код буде працювати. Java розробники можуть розробляти додаток на своєму комп'ютері, а потім розгорнути його на цільовій платформі, будь то телефон або сервер. Якщо для компілятора доступні потрібні бібліотеки, код буде працювати [138].

Безкоштовність – дійсно важливий фактор для багатьох організацій. Адже коли компанія вибирає ту чи іншу технологію, то ціна грає не останню роль. Java – безкоштовна платформа [136]. Компанія Sun завжди була одним з лідерів в Open Source співтоваристві, але вона так і не зважилася повністю звільнити Java. Це не завадило Java програмістам написати значну кількість відмінних

бібліотек і проектів під вільними відкритими ліцензіями. Проект Apache продовжує поставляти безліч проектів на Java під ліцензією, яка не вимагає багато чого взамін [138].

Відома достатньо велика кількість бібліотек, яка дозволяє писати код, слідуючи найкращим практикам програмування. Це досягається завдяки таким фреймворкам як Spring, Struts, Maven. Багато великих компаній, такі як Google, Apache вносять свій фінансовий внесок в бібліотеки і, таким чином, розробка стала швидшою і більш рентабельною [136].

Однією із сильних сторін віртуальної машини Java завжди була її здатність з легкістю жонглювати декількома потоками. JVM оптимізована для великих багатоядерних машин, тому вона без проблем може керувати сотнями потоків. Завдяки цій здатності для JVM запропоновано і інші мови – створюються крос-компілятори і емулятори, що працюють поверх JVM [138].

Google Cloud Natural Language розкриває структуру та зміст тексту завдяки потужним моделям машинного навчання з простою у використанні REST API та користувацькими моделями (рисунок 4.1), які легко створювати за допомогою AutoML Natural Language [139].

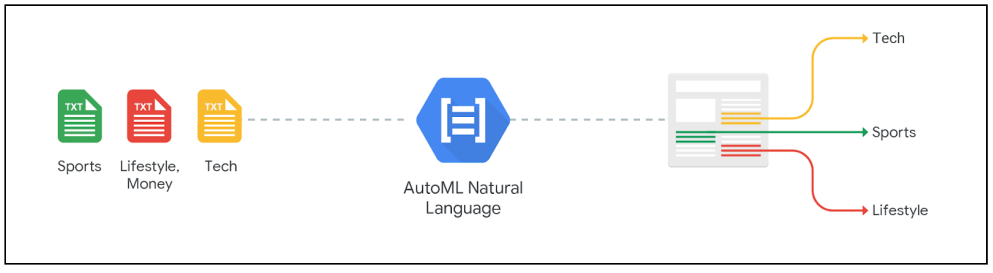


Рисунок 4.1 – Схематичне зображення робочого процесу в Google Cloud Natural Language

Недоліками Google Cloud Natural Language є ліміт на кількість запитів у місяць, щомісячна плата за користування, а також неможливість роботи без з'єднання з Інтернет.

Apple Natural Language фреймворк призначений для виконання завдань, таких як ідентифікація мови та скриптів, токенізація, лематизація, позначення частин мови та розпізнавання назв об'єктів. Також передбачена можливість використовувати цей фреймворк разом з Create ML (фреймворк машинного навчання від Apple) для навчання та створення користувацьких моделей природної мови [140].

Apple NLP API дає можливість виконувати наступні завдання:

- ідентифікація мови – використання методів машинного навчання для визначення мови тексту;
- токенізація – розбиття тексту на слова, речення та абзаци;
- аналіз частини мови – визначення частини мови, до якої належить поточне слово;
- лематизація – визначення леми (словникової форми) слова;
- розпізнавання назв об'єктів – вилучення об'єктів певних типів, наприклад, назв організацій та людей [131].

Недоліком Apple NLP є відсутність можливості визначення зв'язків між членами речення.

Ще одним популярним лінгвістичним пакетом на основі Java є DKPro Core – це набір програмних компонентів для обробки природної мови, що побудовано на основі Apache UIMA framework. Пакет DKPro Core вважають більшим, аніж просто збір компонентів аналізу, які взаємодіють між собою. Він був побудований з метою підвищення продуктивності дослідників, що працюють з автоматичним аналізом мови. Підхід DKPro Core полягає у тому, що дослідники повинні мати можливість зосередитися на своїх реальних наукових питаннях, а не на розробці технологій. Колекція прагне досягти цієї мети, слідуючи таким принципам [142]:

а) Вибір – для більшості кроків аналізу DKPro Core включає кілька різних інструментів від різних постачальників. DKPro Core охоплює такі завдання

аналізу, як визначення мови, лематизація, морфологічний аналіз, синтаксичний розбір, розбір залежностей, сегментація, маркування смислових ролей, перевірка орфографії та морфології. Для кожного завдання було використано до семи різних інструментів. Окрім того, підтримується 19 різних форматів даних [142].

б) Покриття – багато з компонентів аналізу були упаковані та інтегровані у відповідні набори ресурсів для різних мов, наразі DKPro Core об'єднує 94 моделі на 15 природних мовах [142].

в) Взаємозамінність – імена параметрів компонентів однакові, а там, де це можливо, компоненти приймають однакові параметри для різних ресурсів. Наприклад, параметри ручного вибору моделі або перевизначення відображення для різних типів мають ті ж імена, незалежно від компонента. Таким чином, користувачу, що змінив один компонент на інший не потрібно вивчати абсолютно новий набір параметрів. Часом, навіть тільки зміни назви реалізації достатньо, а параметри і значення параметрів залишаються незмінними [142].

г) Переносимість – компоненти аналізу завантажуються та працюють на різних платформах системи або за допомогою віртуальної машини Java, або шляхом використання бінарних файлів, що компілюються для різних операційних систем. DKPro Core інтегровано в Maven інфраструктуру з метою забезпечення належного контролю за версіями артефактів і завантаженням їх на комп'ютер користувача. Це важливий крок на шляху до створення портативних і відтворюваних робочих середовищ. Портативність також важливе питання, бо дає можливість масштабувати з DKPro Core робочі середовища для обчислювального кластера при обробці великих обсягів даних. Для підтримки максимального контролю над процесом веб-сервіси обробки природної мови виключені з DKPro Core. Замість цього, розв’язуються проблеми упаковки і

автоматичного розгортання компонентів обробки на комп'ютерах користувачів [142].

д) Зручність – компоненти аналізу вимагають тільки мінімальної обов'язкової конфігурації і багато компонентів не потребують обов'язкового налаштування взагалі, тому що на основі опрацьованих даних вони можуть автоматично виявити, які ресурси, наприклад, аналізатор чи модуль визначення частин мови, потрібні. Багато з компонентів DKPro Core здатні також автоматично завантажувати ресурси під час виконання, в залежності від даних, що обробляються [131]. Схематичне зображення робочого процесу в DKPro Lab [120] наведено в додатку В.

4.2 Структура інформаційної технології

Інформаційна технологія представляє собою упорядкований набір знань про організацію процесу створення або зміни інформаційних об’єктів. Будь яка інформаційна технологія включає в себе дані про структуру та можливі характеристики інформаційних об’єктів, опис необхідних засобів та перелік методів, які дозволяють досягти бажаних характеристик інформаційних об’єктів. В загальному сенсі, під інформаційним об’єктом розуміють представлення в інформаційній системі реального об’єкту, події, процесу, явища тощо у вигляді опису його структури, атрибутів, обмежень цілісності, можливої поведінки та інших характеристик, важливих для інтерпретації в рамках певних задач конкретної предметної галузі [143].

Структуру інформаційної технології пошуку ключових слів, що пропонується, представлено на рисунку 4.2. Структуру розроблено за модульним принципом з метою подальшого розвитку програмного забезпечення шляхом інтеграції нових модулів в існуючу архітектуру, розширюючи таким чином функціонал системи.

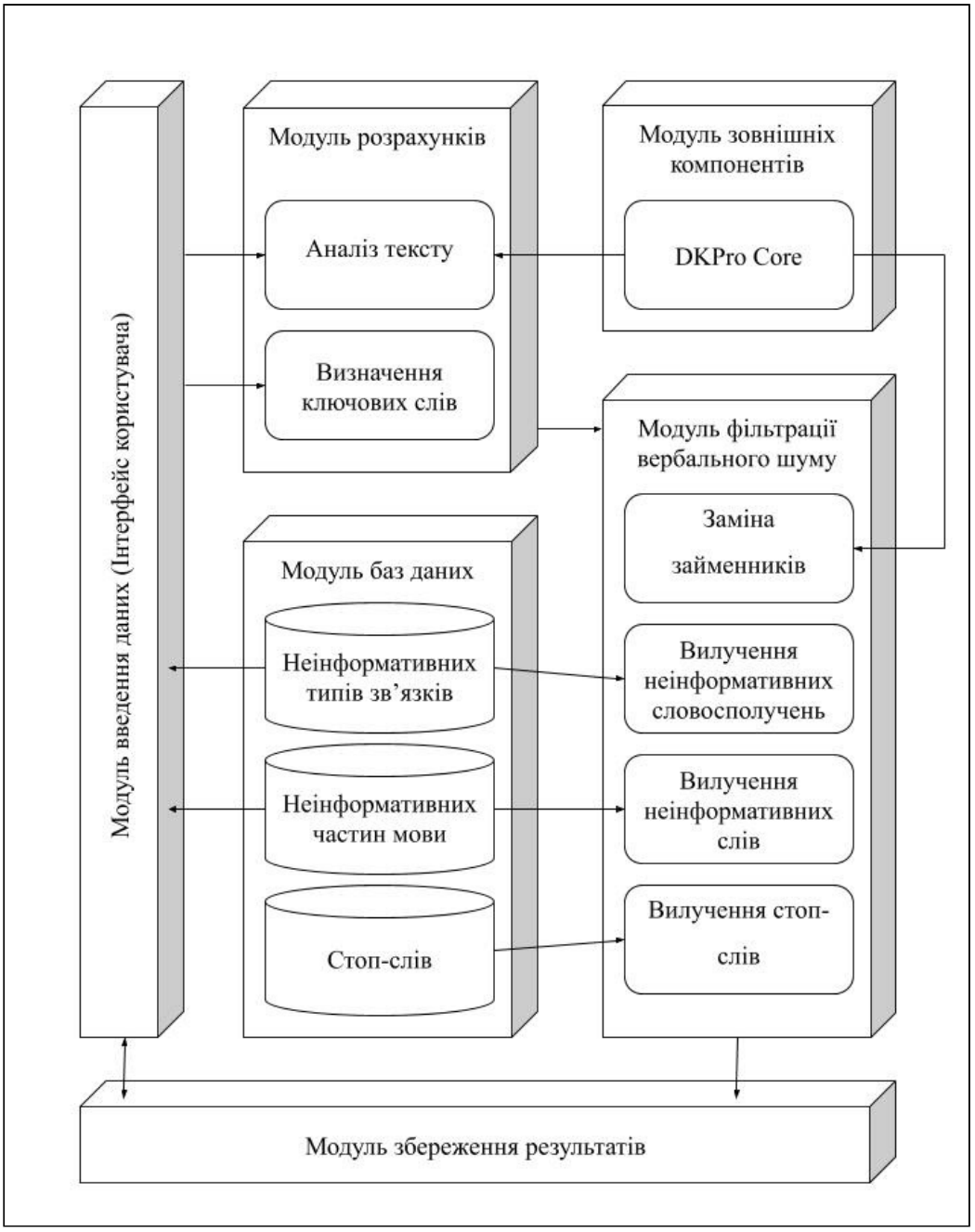


Рисунок 4.2 – Структура інформаційної технології

Модуль введення даних (Інтерфейс користувача) відповідає за:

- вибір з носіїв інформації тексту, для якого необхідно знайти ключові слова;
- відображення типів зв'язків, які не несуть суттєвого смислового навантаження;
- відображення списку неінформативних частин мови;
- запуск аналізу тексту та пошуку ключових слів;

- вибір модулів фільтрації вербального шуму – всіх або тільки деяких з доступних;

- відображення прогресу аналізу тексту та логів виконання.

Модуль розрахунків здійснює аналіз тексту та пошук ключових слів. Аналіз тексту передбачає створення багаторівневої розмітки тексту та отримання синтаксичної розмітки, що враховує складні залежності між парами лем [116]. Для цих операцій використовуються методи DKPro Core, зокрема пошук ключових слів складається з [116]:

- отримання словосполучень як пар слів-кандидатів у ключові слова;

- розбиття пар на окремі слова і визначення кількості зв’язків;

- вибір перших n слів з найбільшою кількістю зв’язків, де n – кількість потрібних ключових слів.

Модуль зовнішніх компонентів відповідає за роботу з DKPro Core і завантаження необхідних для його роботи бібліотек.

Модуль фільтрації вербального шуму здійснює наступні операції: заміна займенників на відповідні до них іменники; вилучення словосполучень із типами зв’язків, які не несуть суттєвого смислового навантаження; вилучення слів, які відносяться до неінформативних частин мови; вилучення слів, які відносяться до списку стоп-слів.

Модуль баз даних відповідає за роботу з інформацію про неінформативні типи зв’язків, а також слова, які відносяться до неінформативних частин мови та список стоп-слів.

Модуль збереження результатів зберігає ключові слова, – що а) знайдені після аналізу зв'язків, і потім, б) після кожного кроку фільтрації вербального шуму, – на носії інформації.

На рисунку 4.3 зображено схему взаємодії користувача з інформаційною технологією.

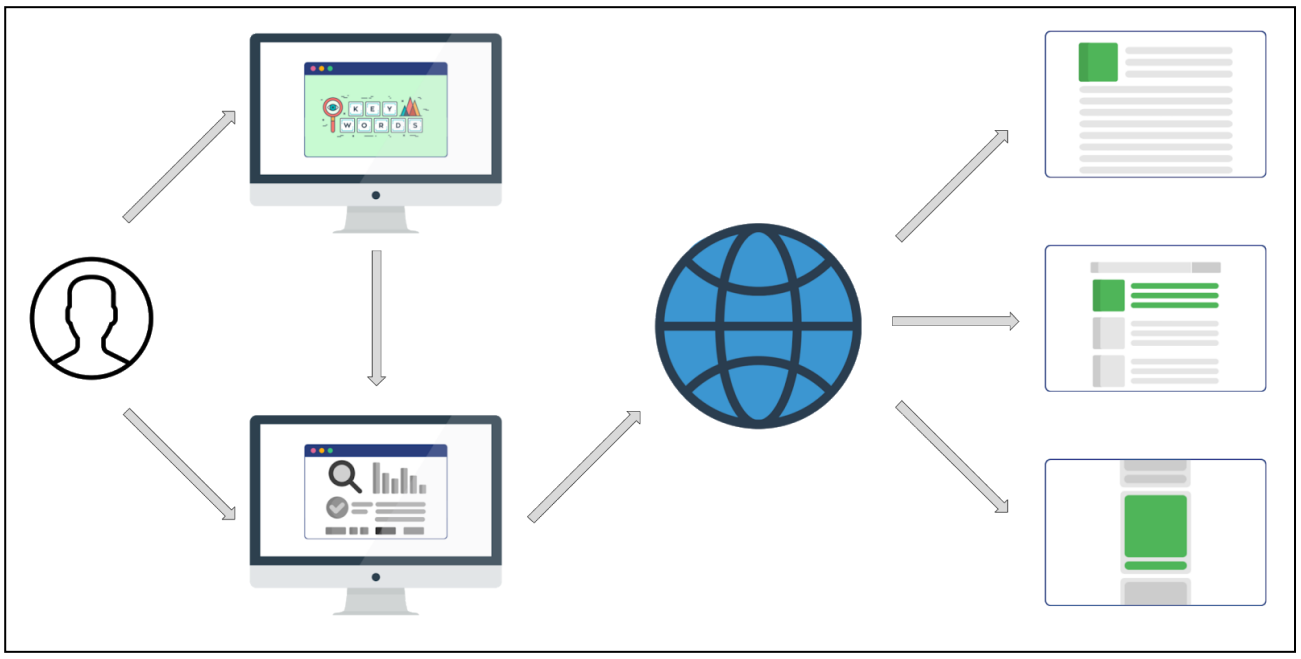


Рисунок 4.3 – Схема взаємодії користувача
з інформаційною технологією

Програмне забезпечення розроблено з використанням об’єктно-орієнтованої технології у середовищі програмування Eclipse. Для роботи із зовнішніми модулями і компонентами використовується Maven.

Пропонується впроваджувати отриману інформаційну технологію для видавництв, де знайдені ключові слова використовуються для індексації публікацій. Корисною буде така інформаційна технологія в SEO оптимізації для підняття позиції сайту в результатах пошуку і для створення контекстної реклами, оскільки саме за ключовими словами відображають контекстну рекламу в пошукових системах та соціальних мережах (Додаток Г).

З рисунку 4.3 видно, що користувач задає текст, який використовується для опису продукту, і отримує ключові слова, використовуючи розроблене програмне забезпечення. Далі, ключові слова вводяться в системи створення контекстної реклами, які в свою чергу показують її на пристроях потенційних клієнтів, зацікавлених в продукції, що рекламується.

Також, запропоновану інформаційну технологію можна використовувати для покращення ідентифікації токсичних коментарів в соціальних мережах [144].

4.3 Реалізація програмного експерименту з використанням лінгвістичного пакету DKPro Core

Для експериментальної перевірки результатів теоретичного аналізу було розроблено програмне забезпечення на основі DKPro Core. Java-програму для пошуку ключових слів засобами DKPro Core було розроблено на основі [145].

Пошук ключових слів відбувається у кілька етапів:

а) створення багаторівневої розмітки тексту;

б) синтаксична розмітка, що враховує складні залежності між парами лем;

в) зменшення вербального шуму.

Сутність підходу, на відміну від відомих аналогів, полягає у визначенні кількості зв’язків для окремих слів і подальшим вибором перших n слів з найбільшою кількістю зв’язків, де n – кількість потрібних ключових слів.

Створення багаторівневої розмітки тексту і синтаксична розмітка, що враховує складні залежності між парами лем досягається засобами DKPro Core [116].

Аналогами розробленої програми можуть бути відомі сайти SEO оптимізації, де є функція пошуку ключових слів. Для даного експерименту вибрані такі популярні сервіси: seotool.by/analiz/seo/key-wordstext.php, rise-top.com/keywordstext.php та advego.ru/text/seo.

Для проведення експерименту було взято текст короткої анотації для тез «Variability management in software product lines using adaptive object and reflection» [127], де ключові слова задані авторами: Software product line,

Variability, Adaptive object model, Reflection, Brazilian Satellites Launcher. Текст анотації складається з 141 слова [146].

Результати знаходження ключових слів для власної розробки і аналогів наведено в таблиці 4.1.

Як видно з результатів пошуку ключових слів (таблиця 4.1) власна розробка знаходить найбільше слів заданих автором – 6 слів. Аналоги знаходять по 5 слів. Власна розробка, так само як і аналоги, знаходить перших 4 слова заданих автором, але не знаходить п’яте – adaptive, проте вона знаходить такі ключові слова: model, reflection, чого не роблять аналоги [146].

Якщо ключові слова не задані автором, то результати роботи можна порівняти з частотним словником для даного тексту [147].

Для тексту «Presidential Address to the Federal Assembly 2013» [148] перші 10 ключових слів за власною розробкою, разом із стоп-словами: we, need, work, I, make, be, system, have, authorities, development, ask, people. Де be і have – стоп-слова. Перші 10 ключових слів із стоп-словами, для частотного словника: that, is, we, will, I, our, be, are, must, it, have, not, all, their, work, Russia, need, should, also, Russian, system. Стоп-слова для частотного словника: that, is, will, be, are, it, have, not, their, should, also. З результатів пошуку ключових слів видно, що при знаходження однакової кількості ключових слів власна розробка показує 2 стоп-слова, а частотний словник – 11. Однаковими ключовими словами є: we, I, work, need, system [147].

Для тексту «Address by President of the Russian Federation 2014» [149] перші 10 ключових слів за власною розробкою, разом із стоп-словами: right, people, Russia, have, be, work, I, do, provide, create, support, make, this, like. Де have, be, do, this – стоп-слова. Перші 10 ключових слів із стоп-словами, для частотного словника: that, we, will, I, be, is, Russia, our, are, have, it, should, not, all, people, must, has, their, was, also, its, they, who, Russian, them, work, national, can, what, course. Стоп-слова для частотного словника: that, will, be, is, our, are,

have, it, should, not, has, their, was, also, its, they, who, them, can, what. З результатів видно, що при знаходженні однакової кількості ключових слів власна розробка показує 4 стоп-слова, а частотний словник – 20. Однаковими ключовими словами є: I, Russia, people, Russian, work [147].

Таблиця 4.1 – Результати пошуку ключових слів

Слова задані автором	seotool	rise-top	advego	Власна розробка
software	+	+	+	+
Product	+	+	+	+
Line	+	+	+	+
variability	+	+	+	+
adaptive	+	+	+	-
Object	-	-	-	-
Model	-	-	-	+
reflection	-	-	-	+
Brazilian	-	-	-	-
Satellites	-	-	-	-
Launcher	-	-	-	-

Окрім того, запропонована інформаційна технологія забезпечує можливість візуалізації графа зв'язків між ключовими словами, де показані типи зв’язків – приклад такого графу представлено в додатку Д.

Для покращення характеристик розробленого методу пошуку ключових слів англомовного тексту на основі інструментальних засобів пакету DKPro Core, було удосконалено метод зменшення впливу вербального шуму на пошук ключових слів. У результаті, за рахунок розроблених додаткових модулів, досягнуто зменшення кількості шумових слів (вербального шуму) на основі

таких підходів: заміна займенників на відповідні до них іменники; вилучення словосполучень із типами зв’язків, які не несуть суттєвого смислового навантаження; вилучення слів, які відносяться до неінформативних частин мови; вилучення слів, які відносяться до списку стоп-слів.

Заміна займенників на відповідні до них іменники (replace pronouns): дозволяє зменшити кількість займенників, а також збільшити кількість іменників, які можуть бути ключовими словами [147]. Для методу пошуку ключових слів англомовного тексту, заміна займенників здійснюється засобами DKPro Core [131].

Вилучення словосполучень із типами зв’язків, які не несуть суттєвого смислового навантаження. Для англомовних текстів, такими типами зв’язків є - визначник (DET): the, which; знаки пунктуації (PUNCT); словосполучення з there або it в експозиційних конструкціях (EXPL), а також: FIXED, REF, ROOT [150].

Вилучення слів, які відносяться до неінформативних частин мови (remove noise words with POS) [10]. Для англійської мови такими частинами мови є: CC, CD, DT, EX, IN, LS, MD, PDT, POS, PRP, PRP$, RP, SYM, TO, UH, WDT, WP, WP$, WRB, -LRB-, -RRB- [98], [99], [100].

Вилучення слів, які відносяться до списку стоп-слів [10]. Список слів для англомовних текстів описаний в [151].

Проілюструємо результати пошуку ключових слів на кожному кроці роботи методу зменшення впливу вербального шуму на пошук ключових слів, що пропонується на невеликому тексті, який складається з двох речень: «Born in Honolulu, Hawaii, Obama is a graduate of Columbia University and Harvard Law School, where he was president of the Harvard Law Review. He was a community organizer in Chicago before earning his law degree» [152]. Знайдені словосполучення і частини мови відповідних слів першого речення наведено в таблиці 4.2, а відповідних слів другого речення наведено в таблиці 4.3.

Таблиця 4.2 – Словосполучення і частини мови відповідних слів першого речення

Born in Honolulu, Hawaii, Obama is a graduate of Columbia University and Harvard Law School, where he was president of the Harvard Law Review			
Головне слово	Тег частини мови головного слова	Залежне слово	Тег частини мови залежного слова
graduate	NN	born	VBN
born	VBN	honolulu	NNP
honolulu	NNP	hawaii	NNP
graduate	NN	obama	NNP
graduate	NN	is	VBZ
graduate	NN	a	DT
university	NNP	columbia	NNP
graduate	NN	university	NNP
school	NNP	harvard	NNP
school	NNP	law	NNP
university	NNP	school	NNP
graduate	NN	school	NNP
president	NN	where	WRB
president	NN	he	PRP
president	NN	was	VBD
university	NNP	president	NN
review	NNP	the	DT
review	NNP	harvard	NNP
review	NNP	law	NNP
president	NN	review	NNP

Таблиця 4.3 – Словосполучення і частини мови відповідних слів другого речення

He was a community organizer in Chicago before earning his law degree			
Головне слово	Тег частини мови головного слова (Governor POS)	Залежне слово	Тег частини мови залежного слова (Dependent POS)
organizer	NN	he	PRP
organizer	NN	was	VBD
organizer	NN	a	DT
organizer	NN	community	NN
organizer	NN	chicago	NNP
organizer	NN	earning	VBG
degree	NN	his	PRP$
degree	NN	law	NN
earning	VBG	degree	NN

Типи зв'язків між головними і залежними словами у словосполученнях, приведених до незмінної, основної форми слова – для першого речення наведено у таблиці 4.4, для другого речення наведено в таблиці 4.5.

Таблиця 4.4 – Зв'язки у словосполученнях першого речення

Born in Honolulu, Hawaii, Obama is a graduate of Columbia University and Harvard Law School, where he was president of the Harvard Law Review					
Головне слово (Governor)	Залежне слово (Dependent)	Тип зв'язку (Dependency Type)	Головне слово (Governor)	Залежне слово (Dependent)	Тип зв'язку (Dependency Type)
graduate	bear	vmod	university	school	conj_and
bear	honolulu	prep_in	graduate	school	prep_of
honolulu	hawaius	appos	president	where	advmod
graduate	obama	nsubj	president	he	nsubj
graduate	be	cop	president	be	cop
graduate	a	det	university	president	rcmod
university	columbium	nn	review	the	det
graduate	university	prep_of	review	harvard	nn
school	harvard	nn	review	law	nn
school	law	nn	president	review	prep_of

Таблиця 4.5 – Зв'язки у словосполученнях першого речення

He was a community organizer in Chicago before earning his law degree		
Головне слово (Governor)	Залежне слово (Dependent)	Тип зв'язку (Dependency Type)
organizer	he	nsubj
organizer	be	cop
organizer	a	det
organizer	community	nn
organizer	chicago	prep_in
organizer	earn	prepc_before
degree	his	poss
degree	law	nn
earn	degree	dobj

Розіб'ємо словосполучення на окремі слова і підрахуємо кількість зв'язків для кожного слова, тобто визначимо, у скількох словосполученнях кожне слово зустрічається [152]. Вісдортувавши слова за кількістю зв’язків отримаємо результати, які наведено в таблиці 4.6.

Таблиця 4.6 – Кандидати в ключові слова після розбиття словосполучень

Слово	Кількість зв'язків	Слово	Кількість зв'язків	Слово	Кількість зв'язків
graduate	6	degree	3	hawaius	1
organizer	6	a	2	community	1
president	5	honolulu	2	the	1
university	4	earn	2	his	1
school	4	bear	2	columbium	1
review	4	harvard	2	where	1
be	3	he	2	chicago	1
law	3	obama	1	–	–

Умовно словосполучення можна позначити як G-[T]->D, де G – головне слово (Governor); T – тип зв'язку (Dependency Type); D – залежне слово (Dependent) [152].

На етапі заміни займенників на відповідні до них іменники (replace pronouns) [152]:

− словосполучення president-[nsubj]->he замінюється на president-[nsubj]->obama;

− словосполучення organizer-[nsubj]->he замінюється на organizer-[nsubj]->obama;

− словосполучення degree-[poss]->his замінюється на degree-[poss]->obama.

Кандидати в ключові слова, після заміни займенників на відповідні до них іменники, наведено в таблиці 4.7.

Таблиця 4.7 – Кандидати в ключові слова після заміни займенників

Слово	Кількість зв'язків	Слово	Кількість зв'язків	Слово	Кількість зв'язків
graduate	6	be	3	harvard	2
organizer	6	law	3	hawaius	1
president	5	degree	3	community	1
university	4	a	2	the	1
obama	4	honolulu	2	columbium	1
school	4	earn	2	where	1
review	4	bear	2	chicago	1

Після заміни займенників, кількість кандидатів в ключові слова зменшилася з 23 до 21. До заміни займенників слово obama мало 1 зв'язок, а після – 4 зв'язки. І навпаки слова he з 2 зв'язками і his з одним зв'язком після заміни займенників мають нуль зв'язків, тому що словосполучення з ними були замінені на еквіваленти з іменниками [152].

Вилучення словосполучень із типами зв’язків, які не несуть суттєвого смислового навантаження (deleting noise relationship). Для даного тексту, видаляються словосполучення: graduate-[det]->a, review-[det]->the, organizer-[det]->a [152].

У результаті кількість кандидатів в ключові слова зменшиться до 19, що відображено в таблиці 4.8.

Вилучення слів, що відносяться до шумових частин мови (deleting noise POS keywords) [152]. На даному кроці видаляється слово where з тегом частини мови WRB. Кандидати в ключові слова будуть мати вигляд, наведений в таблиці 4.9.

Таблиця 4.8 – Кандидати в ключові слова після видалення шумових зв'язків

Слово	Кількість зв'язків	Слово	Кількість зв'язків
graduate	5	honolulu	2
organizer	5	earn	2
president	5	bear	2
university	4	harvard	2
obama	4	hawaius	1
school	4	community	1
be	3	columbium	1
law	3	where	1
degree	3	chicago	1
review	3	–	–

Таблиця 4.9 – Кандидати в ключові слова після видалення шумових частин мови

Слово	Кількість зв'язків	Слово	Кількість зв'язків	Слово	Кількість зв'язків
graduate	5	be	3	bear	2
organizer	5	law	3	harvard	2
president	5	degree	3	hawaius	1
university	4	review	3	community	1
obama	4	honolulu	2	columbium	1
school	4	earn	2	chicago	1

На етапі видалення стоп-слів (deleting stop words) – видаляється стоп-слово be, тому таблиця 4.10 містить 17 кандидатів в ключові слова [152].

Таблиця 4.10 – Кандидати в ключові слова після видалення стоп-слів

Слово	Кількість зв'язків	Слово	Кількість зв'язків	Слово	Кількість зв'язків
graduate	5	law	3	harvard	2
organizer	5	degree	3	hawaius	1
president	5	review	3	community	1
university	4	honolulu	2	columbium	1
obama	4	earn	2	chicago	1
school	4	bear	2	–	–

У результаті, після усіх запропонованих кроків удосконаленого методу, вдалося зменшити кількість кандидатів в ключові слова з 23 до 17, а також повністю видалити шумові слова [152].

Для більш масштабної апробації двох запропонованих методів було проведено експеримент з текстом “Participation, archival activism and learning to learn”[153], який складається з 1588 слів, де ключові слова задані автором: Participatory, Historical geography at large, Archival activism, Canada, Aboriginal rights. Першими дев'ятьма ключовими словами, без використання удосконаленого методу до зменшення шуму, знайдено: the, be, geography, have, work, geographer, participatory, to, history. З використання удосконаленого методу до зменшення шуму, отримано такі перші дев'ять ключових слів: geography, learn, geographer, participatory, history, research, historical, project, community.

Отже, послідовне використання удосконаленого методу до зменшення шуму після основного методу пошуку ключових слів дозволило покращити загальні результати пошуку ключових слів – знайти три слова з заданих автором (participatory, historical, geography), тоді як без зменшення шуму було знайдено два слова, заданих автором (participatory, geography). Окрім того, проведено заміну 4-х шумових слів з 9-ти першого списку на 4 значущих слова.

4.4 Огляд інтерфейсу користувача

Розроблене програмне забезпечення складається з трьох вкладок: Parsing text, Searching keywords і Result. Вкладку Parsing text, яка відповідає за синтаксичний розбір тексту, представлено на рисунку 4.4.

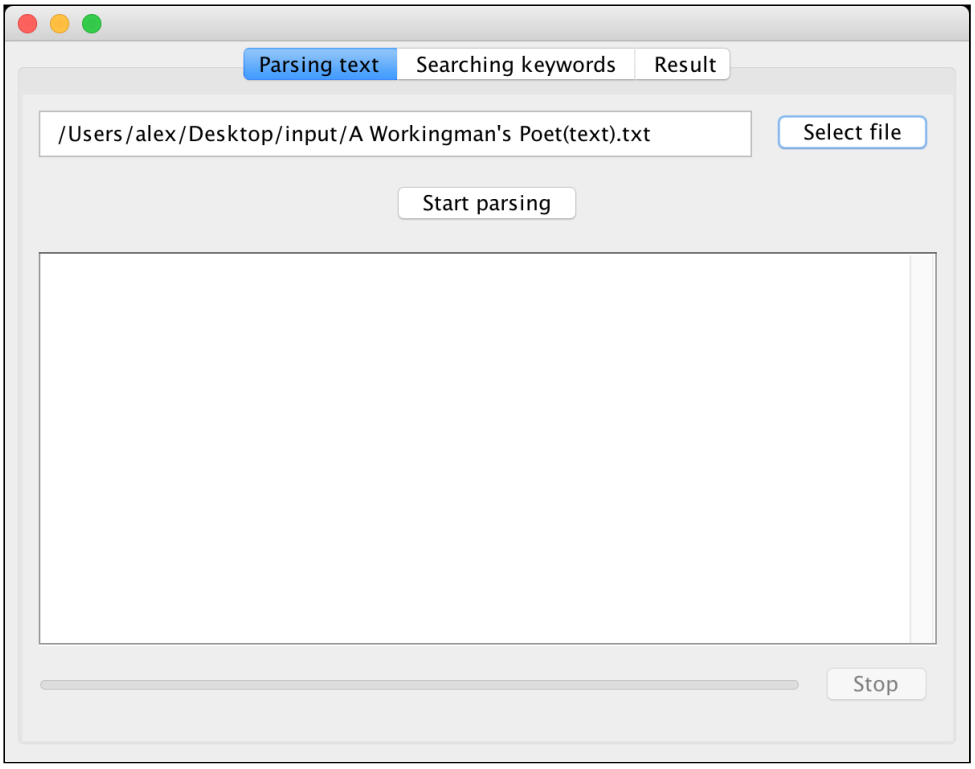


Рисунок 4.4 – Вкладка синтаксичного розбору тексту

Текстовий файл вибирається в діалоговому вікні, загальний вигляд якого представлено на рисунку 4.5.

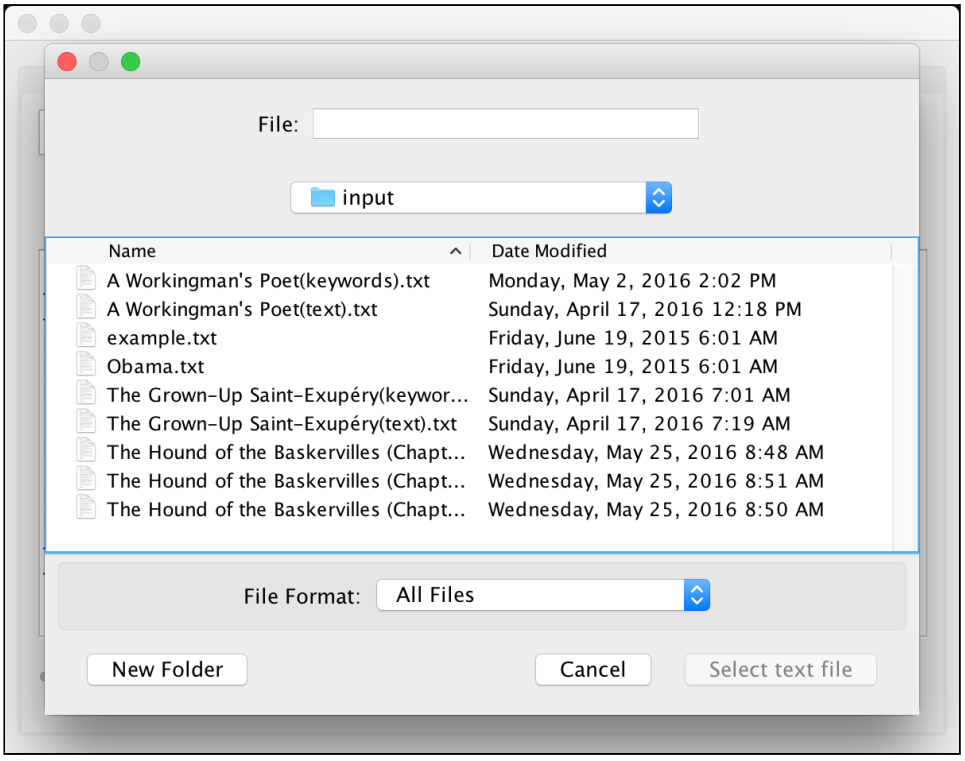


Рисунок 4.5 – Вибір текстового файлу

Елементами вкладки Parsing text є:

- кнопка Select file – відкриття діалогового вікна для вибору текстового файлу;
- кнопка Start parsing – запуск синтаксичного розбору тексту;
- текстова область – виведення логів DKPro Core, інформації про перебіг процесу розбору тексту, помилок, якщо вони виникають;
- прогрес бар – наочно відображає перебіг процесу синтаксичного розбору тексту на основі кількості логів DKPro Core;
- кнопка Stop – зупинка виконання запущеного процесу розбору тексту.

На другій вкладці розміщені елементи, що дозволяють включати і виключати модулі знаходження ключових слів. Вкладка Searching keywords зображена на рисунку 4.6.

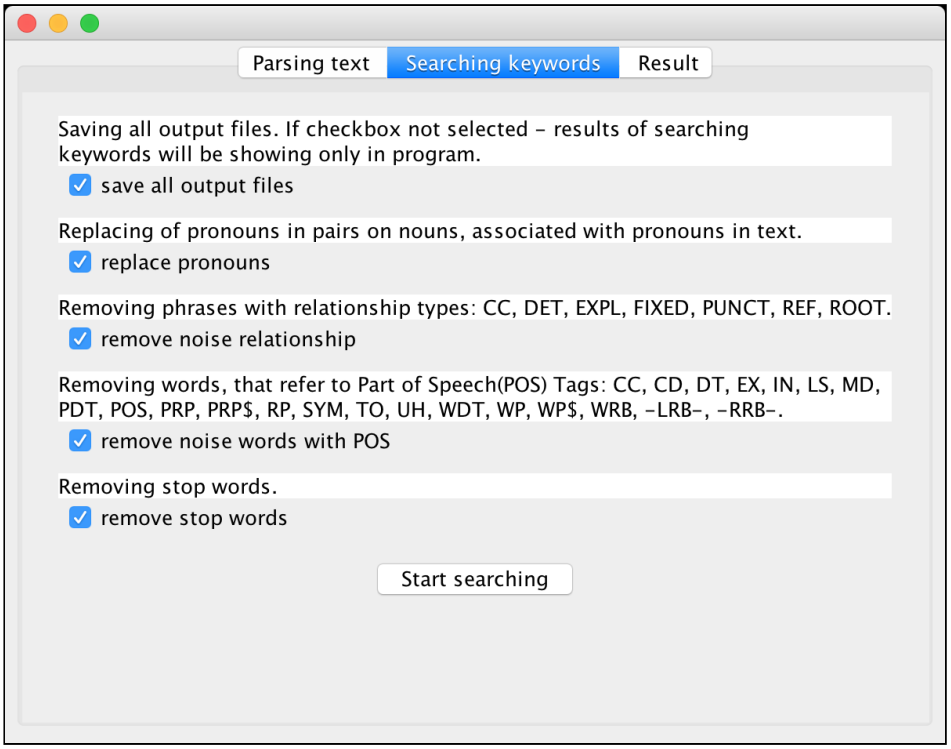


Рисунок 4.6 – Вкладка ввімкнення модулів

Опис модулів знаходження ключових слів:

1) модуль «save all output files» відповідає за збереження вихідної інформації у файли. Якщо виключити його, то знайдені ключові слова і додаткова інформація буде відображатися тільки на третій вкладці. Якщо включити, то крім відображення на третій вкладці, результати роботи зберігаються у файли;

2) модуль «replace pronouns» – заміна займенників у пари, які отримані відповідно до іменників;

3) модуль «remove noise relationship» – видалення зв'язків, які не несуть суттєвого смислового навантаження: CC, DET, EXPL, FIXED, PUNCT, REF, ROOT;

4) модуль «remove noise words with POS» – видалення слів, які відносяться до тегів частин мови: CC, CD, DT, EX, IN, LS, MD, PDT, POS, PRP, PRP$, RP, SYM, TO, UH, WDT, WP, WP$, WRB, -LRB-, -RRB-;

5) модуль «remove stop words» – видаляє стоп-слова.

Синтаксичний розбір тексту відбувається один раз на першій вкладці, але надалі експериментувати з різними комбінаціями модулів для знаходження ключових слів можна багаторазово на другій вкладці.

Вихідні файли:

- output_sentences – файл, в який виводяться всі речення тексту і їхні словосполучення з типами зв'язків. Вихідний файл output_sentences представлено на рисунку 4.7;
- output_links – csv файл з списком зв'язків між словами в тексті, який представлено на рисунку 4.8;

Рисунок 4.7 – Вихідний файл всіх речень тексту

- output_graph – xml файл з даними графа зв'язків між словами в тексті; вихідний файл output_graph зображено на рисунку 4.9;
- output_frequency – файл, в якому зберігаються знайдені ключові слова і кількість їхніх зв'язків. Такі файли створюються для кожного етапу фільтрації вербального шуму, щоб мати можливість порівнювати ключові слова на різних етапах фільтрації. Вихідний файл output_frequency представлено на рисунку 4.10.

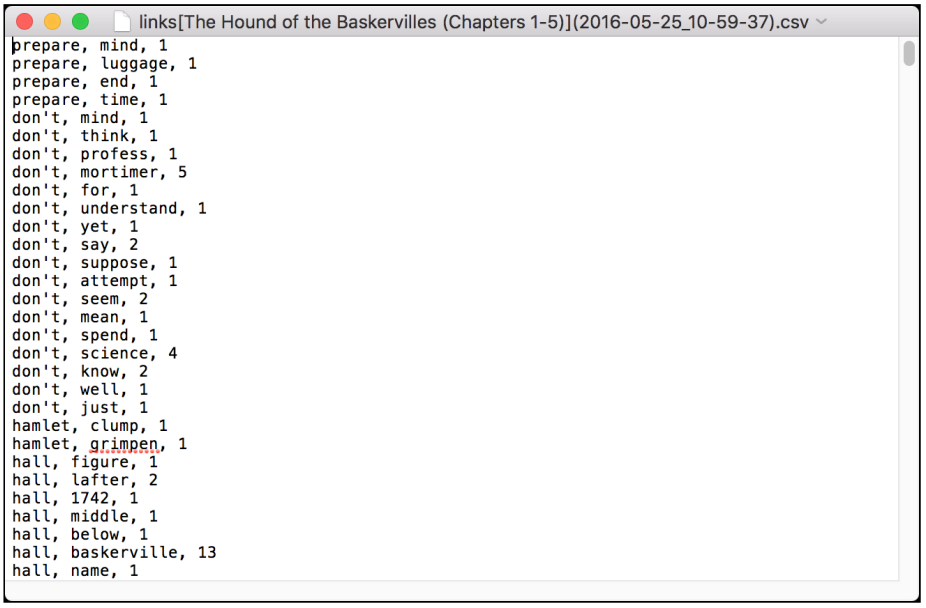


Рисунок 4.8 – Вихідний файл зі списком зв’язків

Рисунок 4.9 – Вихідний файл з даними графа зв’язків

Третя вкладка відображає результати пошуку ключових слів за допомогою раніше увімкнених на другій вкладці модулів. Вкладка з вибраними ключовими словами представлена на рисунку 4.11.

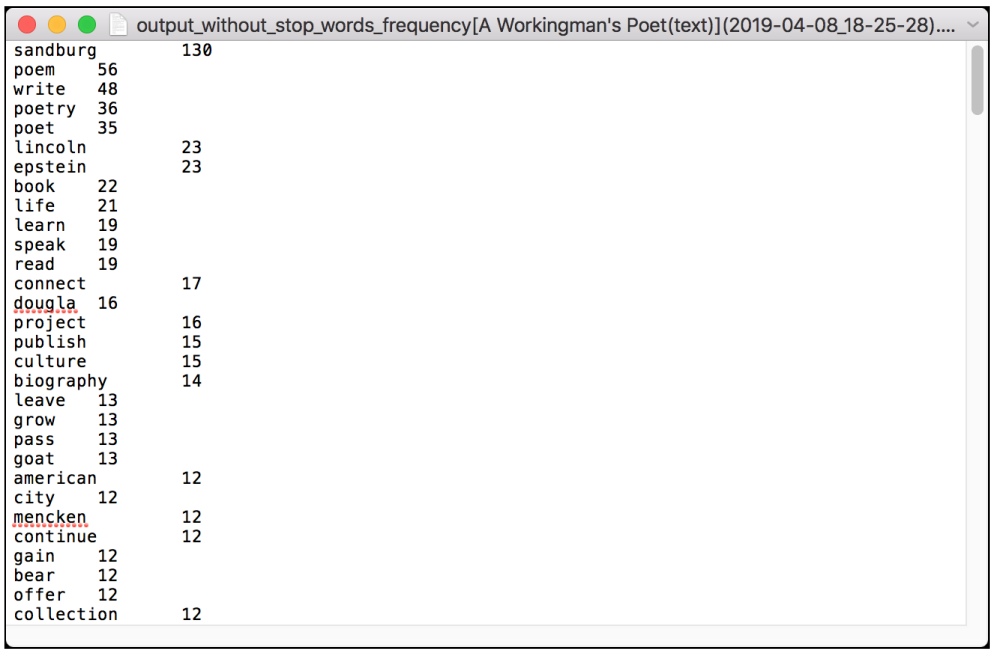


Рисунок 4.10 – Вихідний файл із знайденими ключовими словами

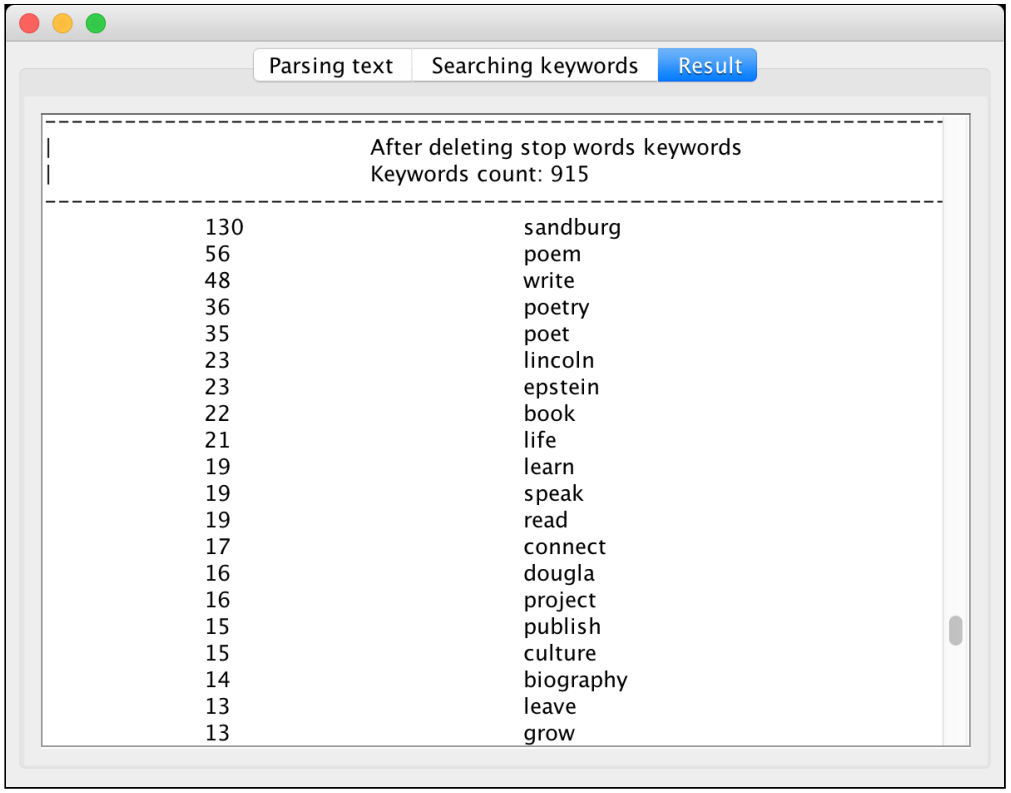


Рисунок 4.11 – Результат пошуку ключових слів

Програма може працювати на різних операційних систем, тому що написана на Java. Тому, її можна запускати в будь-якому середовищі за допомогою JVM (Java Virtual Machine).

4.5 Визначення кількісних характеристик релевантності отриманих результатів

Для підвищення точності пошуку ключових слів у двох запропонованих методах задіяні статистичні методи обробки тексту, швидкість роботи яких підсилюється можливостями сучасних лінгвістичних пакетів. Проте, з огляду на постановку задачі, час отримання результату (знаходження набору ключових слів для вибраного тексту) не має такого вирішального значення, як якість знайдених слів з набору. Тому кількісними характеристиками релевантності отриманих результатів обрано повноту (за Жаккаром) і точність (абсолютну). Проведено інтерпретацію обраних критеріїв до умов задачі пошуку ключових слів.

Повнота за мірою подібності Жаккара, в даному випадку, визначається для двох множин ключових слів – заданої автором (еталонної) та знайденої програмно, і дорівнює відношенню кількості елементів перетину цих множин до кількості елементів їх об'єднання. Тобто, це частка від ділення, де у чисельнику знаходиться кількість правильно знайдених програмою ключових слів, а в знаменнику – загальна кількість усіх елементів (ключових слів) у обох множинах мінус кількість правильно знайдених програмою ключових слів. Отже, повнота за Жаккаром визначається за формулою:

$$J = \frac{n(A \cap B)}{n(A)+n(B)-n(A \cap B)} = \frac{n(A \cap B)}{n(A \cup B)}, \qquad (4.1)$$

де A – множина ключових слів, заданих автором;

B – множина ключових слів, знайдених програмно;

$n(A)$ – кількість елементів множини ключових слів, заданих автором;

$n(B)$ – кількість елементів множини ключових слів, знайдених програмно;

$n(A \cap B)$ – кількість елементів множини-результату перетину двох множин;

$n(A \cup B)$ – кількість елементів множини-результату об'єднання двох множин.

Абсолютна точність, у свою чергу, визначається, як відношення кількості правильно знайдених програмою ключових слів до кількості еталонних ключових слів (заданих автором):

$$a = \frac{n(A \cap B)}{n(A)} \tag{4.2}$$

Застосування формальних критеріїв для повноти і точності дозволить більш об’єктивно оцінити релевантність отриманих результатів пошуку ключових слів на основі запропонованої інформаційної технології.

Розглянемо приклад обчислення кількісних характеристик. Для цього було обрано текст «A Workingman's Poet» [154], де відомі ключові слова, що задані автором: american, literature, books, chicago, poetry, publishing, twentieth century, united states. За результатами експерименту маємо перші десять кандидатів в ключові слова, знайдені розробленим методом: sandburg, poem, write, poet, poetry, book, life, lincoln, learn, speak.

Отже, кількість ключових слів дорівнює 10, а кількість правильно знайдених програмою ключових слів – 2, тоді абсолютну точність обчислюємо як $\frac{2}{10} = 0,2$.

Кількість елементів у множині ключових слів, заданих автором – 10, а в множині знайдених програмно теж дорівнює 10. Тоді повнота за Жаккаром складає $\frac{2}{(10+10)-2} = 0,1111111.$

Відносно малі значення повноти і точності пошуку ключових слів для розглянутого прикладу пояснюються досить малим обсягом тексту з [154]. Для проведення більш масштабного експерименту було використано текст статті з 1460 слів «A new pattern for historical geography: working with enthusiast communities and public history» [155], а результати власної розробки порівнювалися з результатами аналогічних програм пошуку ключових слів [156].

Ключові слова, задані автором: Participation, Public history, Enthusiast communities, Museums, Heritage [156].

Результати знаходження ключових слів для власної розробки і аналогів наведено в таблиці 4.11. Поряд з ключовими словами вказана позиція, на якій знаходиться ключове слово [156].

Таблиця 4.11 – Результати пошуку ключових слів

Слова задані автором		власна розробка		rise-top		advego		seotool	
1	Participation	-	work	-	historical	-	historical	-	historical
2	Public	5	community	4	enthusiast	4	enthusiast	4	enthusiast
3	history	-	geography	5	communities	-	for	5	communities
4	Enthusiast	1	participation	1	participation	5	community	1	participation
5	communities	4	enthusiast	-	geography	-	this	-	work
6	Museums	-	geographer	-	work	6	museum	-	geography
7	Heritage	6	museum	-	research	-	geography	-	new

Результат повноти і точності отриманих ключових слів наведено в таблицях 4.12 та 4.13 відповідно.

Таблиця 4.12 – Результати повноти отриманих ключових слів

Назва	власна розробка	rise-top	advego	seotool
Повнота (Jaccard)	0,4	0,2727272727	0,2727272727	0,2727272727

Таблиця 4.13 – Результати точності отриманих ключових слів

Назва	власна розробка	rise-top	advego	seotool
Точність (Absolute)	0,5714285714	0,4285714286	0,4285714286	0,4285714286

На рисунку 4.12 наведені гістограми повноти за Жаккаром і абсолютної точності для власної розробки і аналогів [156].

Повнота і точність знаходження ключових слів повинні бути якомога більшими. Як видно з гістограм, власна розробка має кращі кількісні характеристики, порівняно з аналогами – 40% та 57%. Тобто, одночасно підвищено на 12,7 % повноту та на 14,3 % точність отриманих ключових слів [156].

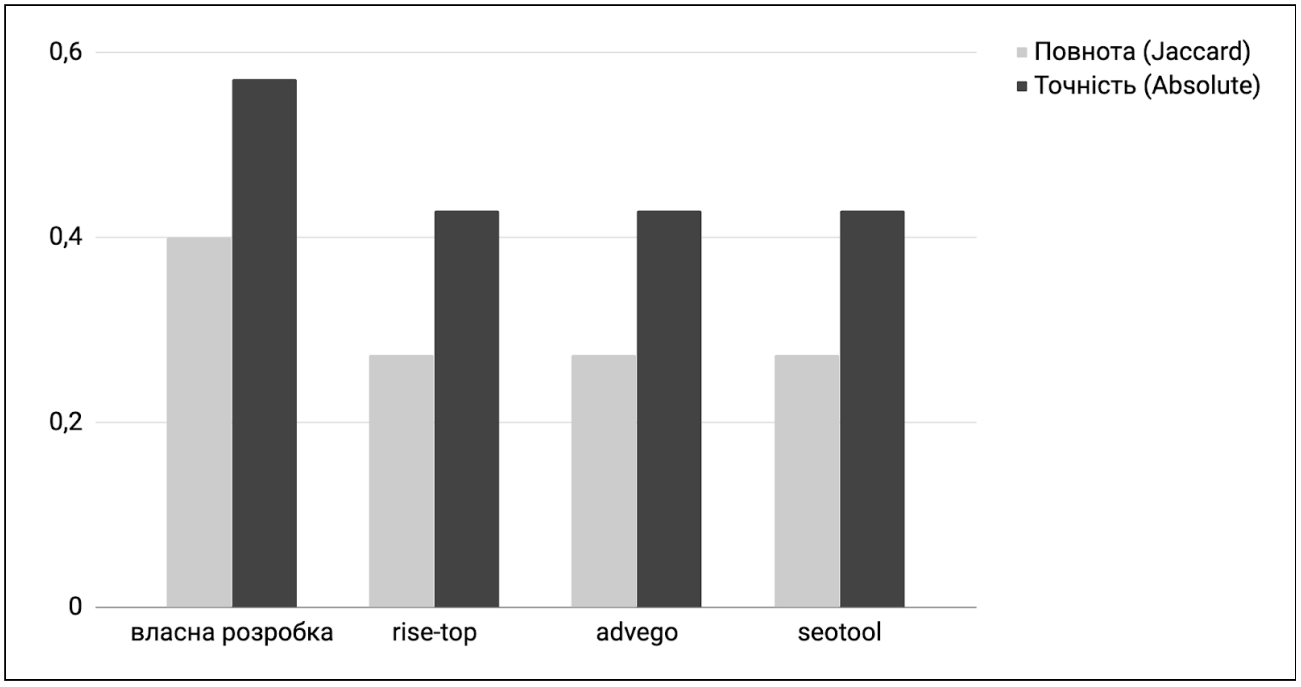


Рисунок 4.12 – Гістограма повноти за Жаккаром і абсолютної точності

Для тексту [127] з прикладу в п.4.3 результати пошуку ключових слів наведено в таблиці 4.1. Розраховані для них за тією ж методикою значення повноти і точності у порівнянні з аналогами представлено в таблиці 4.14.

Таблиця 4.14 – Результати повноти і точності отриманих ключових слів

Назва	власна розробка	rise-top	advego	seotool
Повнота (Jaccard)	0,375	0,2941176471	0,2941176471	0,2941176471
Точність (Absolute)	0,5454545455	0,4545454545	0,4545454545	0,4545454545

Отримані дані показують, що власна розробка має кращі кількісні характеристики, порівняно з аналогами – 38% та 55%. Тобто, і для цього прикладу запропонована інформаційна технологія одночасно збільшує на 8,1 % повноту та на 9,1 % точність пошуку ключових слів від програм-аналогів [156].

Для проведення великого масштабного експерименту обрано базу фільмів з kaggle [157]. Набір даних складається з фільмів, випущених до липня 2017 року включно. Він містить 45 000 записів, з яких вибрано близько 30 000, оскільки у решти записів або немає заданих ключових слів або такі не були розпізнані парсером.

З файлу movies_metadata.csv використовувалися поля:

- id: унікальний ідентифікатор фільму;
- overview: опис фільму;
- genres: список категорій, до яких відноситься фільм.

З файлу keywords.csv використовувалися поля:

- id: унікальний ідентифікатор фільму;
- keywords: еталонні ключові слова, задані автором.

Для даної бази фільмів запропонованою інформаційною технологією було знайдено ключові слова для 42.3% текстів (рисунок 4.13) з описом фільму у записах. Іншими словами, отримана розробка для більше ніж двох п'ятих

коротких англомовних анотацій фільмів дозволила знайти одне і більше ключових слів, заданих автором.

Для текстів з анотаціями, де не вдалося знайти ключові слова, які можна вважати еталонними (задані автором), негативні результати можна пояснити такими причинами:

- еталонним є тільки одне ключове слово,
- присутні іншомовні слова в тексті або ключових словах,
- еталонні ключові слова жодного разу не зустрічаються в тексті,
- текст анотації надкороткий (складається лише з кількох слів).

Отже, порівняно низький відсоток знайдених ключових слів пояснюється специфікою експериментальних даних.

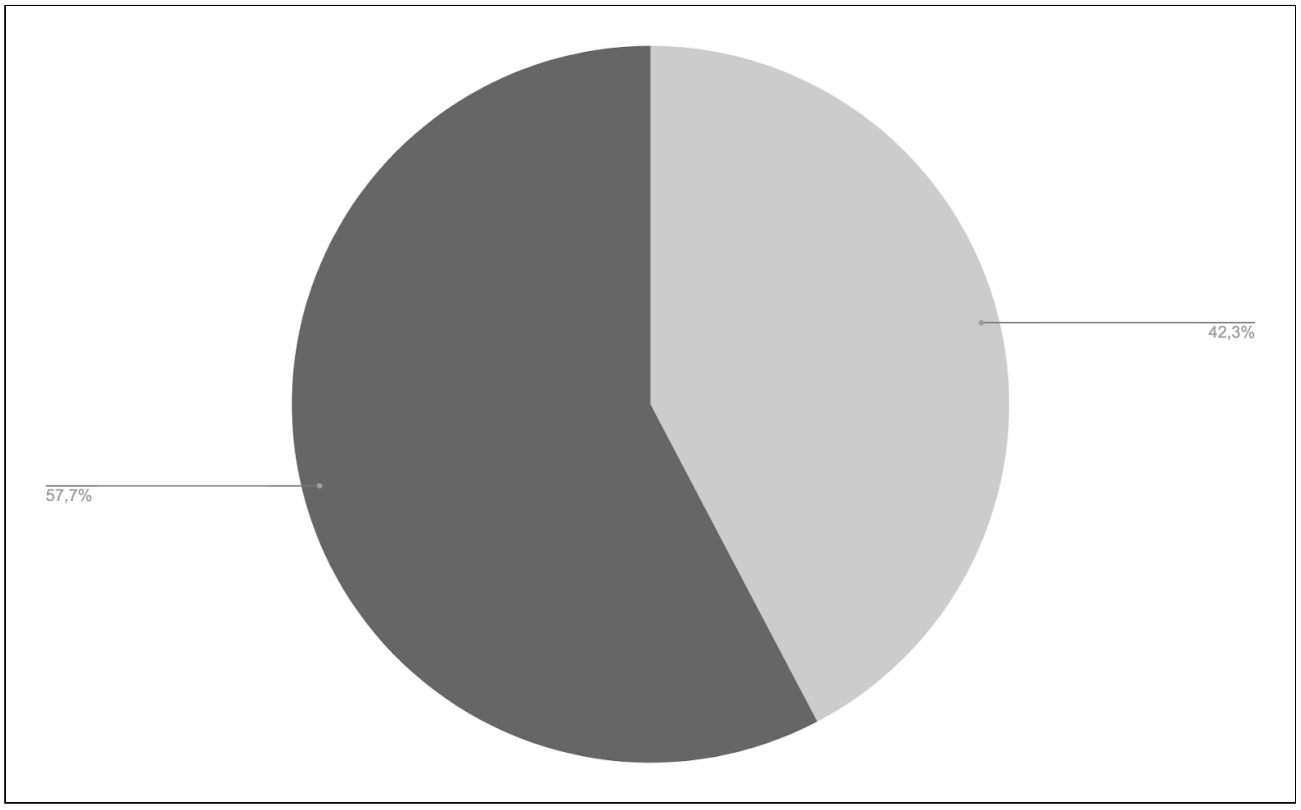


Рисунок 4.13 – Діаграма кількості фільмів, для яких було знайдено ключові слова, задані автором

Для даного експерименту вважаємо ключові слова, задані автором (еталонні), множиною. Не розглядаємо ключові слова як список тому, що маємо загальний опис фільму, в якому кожне ключове слово рівноцінне. Критерії для списку можна розглядати, наприклад, для наукових статей, де автор сам визначає послідовність ключових слів на основі важливості певних термінів з огляду на зміст цієї публікації.

На рисунку 4.14 зображено фрагмент вихідного файлу з результатами визначення кількісних характеристик для кожного опису фільму.

textId	etalonKeywords	commonElements	absolute	jaccard
314580	5	0	0	0
105503	3	0	0	0
346592	1	1	1	1
441728	17	2	0,117647	0,0625
95015	6	1	0,166667	0,090909
228221	2	0	0	0
374142	4	0	0	0
336380	6	0	0	0
270886	2	0	0	0
297721	1	0	0	0
121945	1	0	0	0
388910	4	0	0	0
291540	5	1	0,2	0,111111
96912	3	1	0,333333	0,2
101852	2	0	0	0
188509	2	0	0	0

Рисунок 4.14 – Фрагмент вихідного файлу результатів визначення кількісних характеристик анотацій до фільмів

Вихідний файл містить записи (рядки) з результатами для кожного окремого фільму з такими полями:

- textId: унікальний ідентифікатор фільму;
- etalonKeywords: кількість заданих автором ключових слів;
- commonElements: кількість правильно знайдених ключових слів розробленим методом;
- jaccard: обчислене значення повноти за Жаккаром – має значення від нуля до одиниці;

- absolute: обчислене значення точності – має значення від 0 до 1.

З еталонною множиною порівнюється така ж кількість ключових слів, знайдених розробленою технологією. При цьому, якщо автором задано 1 ключове слово, а розробленою технологією знайдено, наприклад, 37 слів, то для визначення кількісних характеристик, береться тільки перше слово. Особливістю списку ключових слів розробленої технології є те, що на першому місці знаходиться слово, що найбільше підходить, а на останньому таке, що підходить найменше. Тому для 10 еталонних слів і 25 слів, знайдених розробленим методом, буде братися для визначення кількісних характеристик тільки перші 10 слів – таким чином для розробленої технології забезпечується конвертація списку слів у множину.

Якщо відсортувати результати пошуку ключових слів за зростанням точності та повноти, то отримаємо графік наведений на рисунку 4.15, де по осі x розташований порядковий номер тексту.

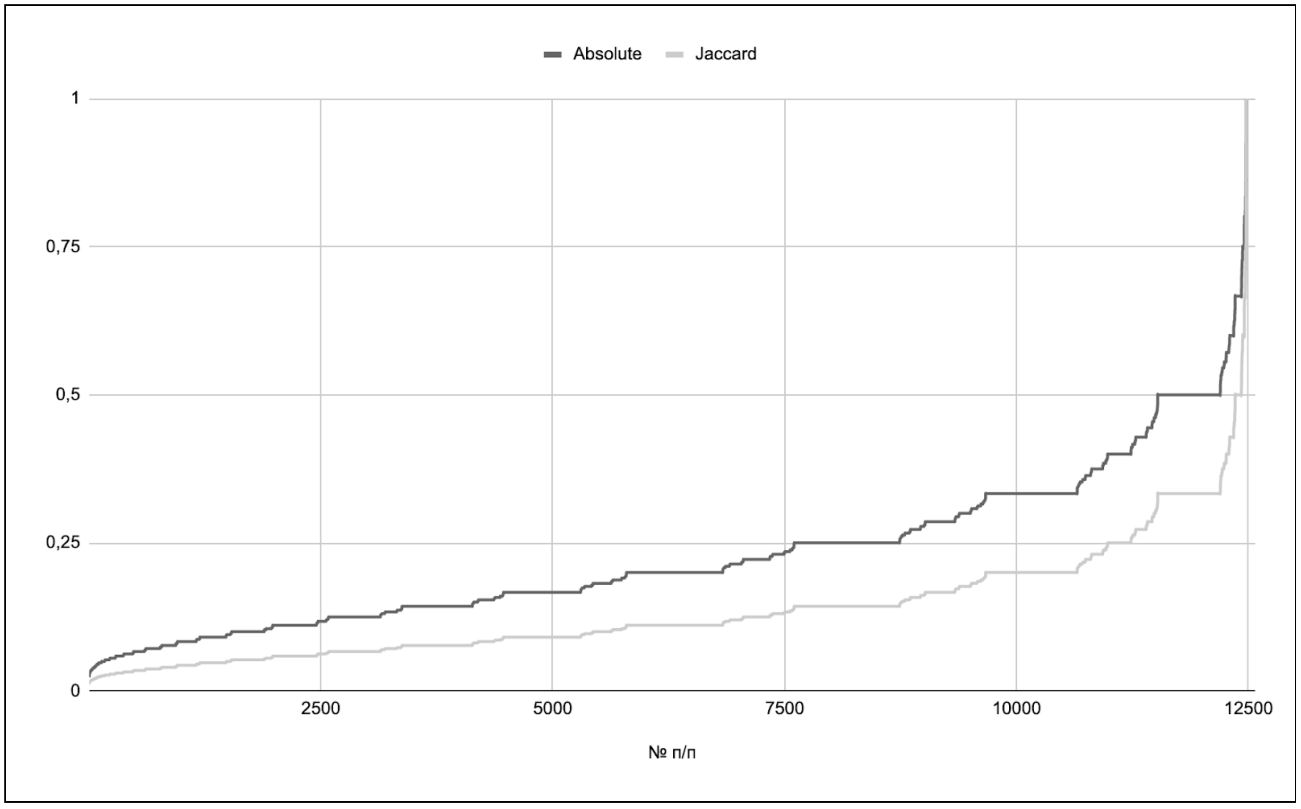


Рисунок 4.15 – Графік отриманих результатів пошуку ключових слів у анотаціях до фільмів за зростанням точності та повноти

Гістограми абсолютної точності та повноти за Жаккаром, наведені відповідно на рисунках 4.15 і 4.17. По осі x розташовані діапазони значень точності чи повноти, а по осі y – кількість текстів з таким значенням точності чи повноти.

Як видно з графіка і гістограм, точність та повнота знаходження ключових слів представлені у всіх діапазонах від нуля до одиниці.

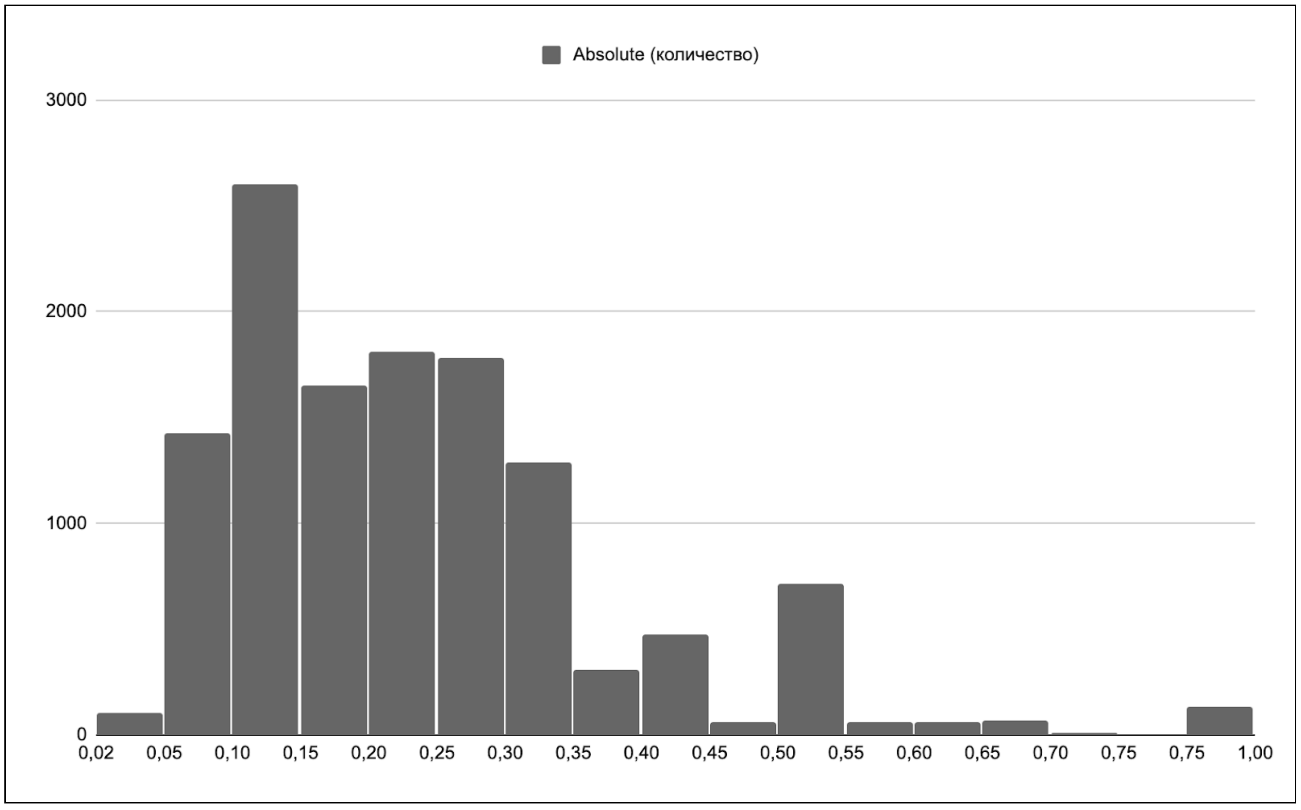


Рисунок 4.16 – Гістограми абсолютної точності для фільмів

Найбільше результатів отримано з точністю від 0.2 до 0.55, що складає 51.2% від усіх позитивних результатів, а також від 0.05 до 0.2 – 45%. За повнотою найбільше таких результатів, 55%, маємо від 0.02 до 0.12 і 41.1% в діапазоні від 0.12 до 0.37.

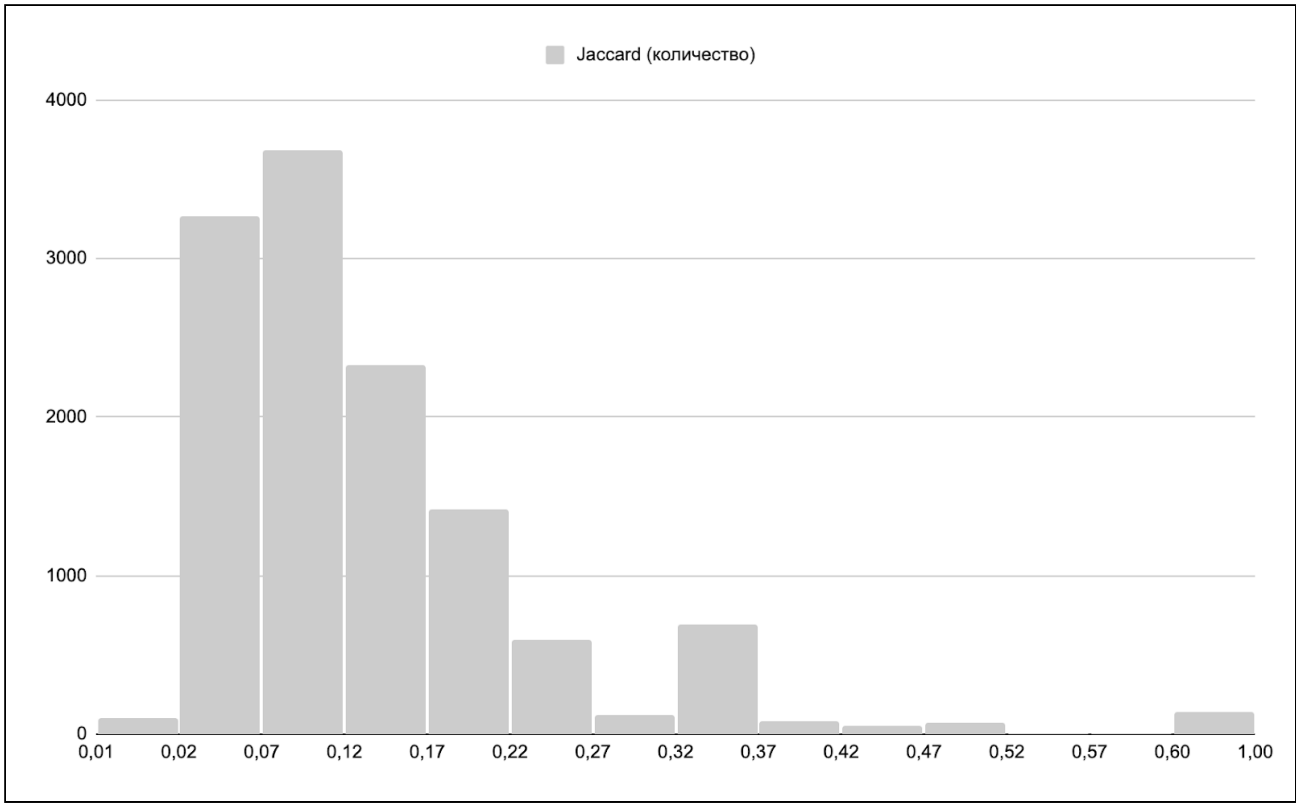


Рисунок 4.17 – Гістограми повноти за Жаккаром для фільмів

Окремої уваги потребує факт “викиду” на діаграмах для одиниці. Він відповідає випадкам, коли розроблена технологія правильно знайшла єдине еталонне ключове слово. Це свідчить про перспективність застосування технології саме для таких ситуацій, коли потрібно знайти найбільш значуще слово з короткого тексту.

Середнє значення повноти за Жаккаром і абсолютної точності (рисунок 4.18) для знайдених розробленою технологією ключових слів, відповідно складає 23.3% і 14.2%.

Набір даних складається з фільмів, що відносяться до двадцяти категорій (рисунок 4.19), причому один фільм може відноситися до декількох категорій.

Категорії, що мають найбільшу кількість фільмів, це Drama, Comedy, Thriller, Romance, Action. Результати обчислення середнього значення точності та повноти для п’яти найбільших категорій наведено в таблиці 4.15.

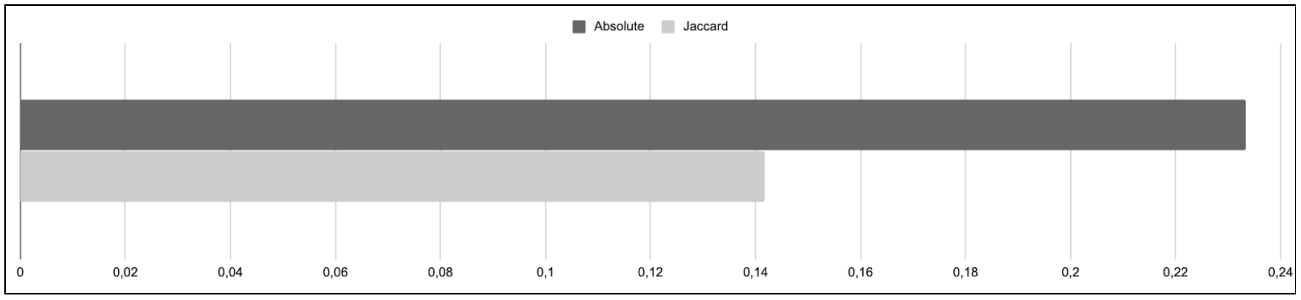


Рисунок 4.18 – Середнє значення повноти за Жаккаром і абсолютної точності для фільмів

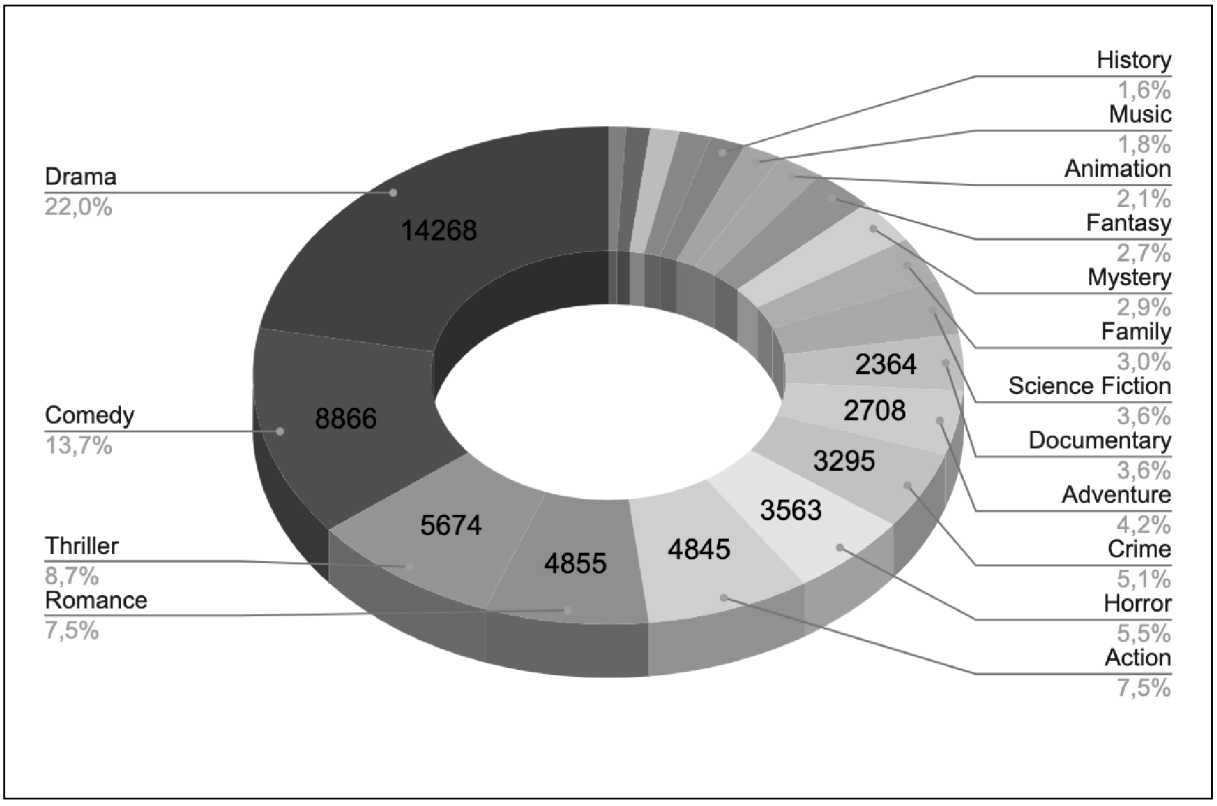


Рисунок 4.19 – Категорії бази фільмів

Таблиця 4.15 – Результати середнього значення точності та повноти для найбільших категорій

Назва	Action	Romance	Thriller	Comedy	Drama
Absolute	0,22142	0,22766	0,220114	0,231997	0,227988
Jaccard	0,133538	0,138171	0,131441	0,14134	0,137378

На рисунку 4.20 наведені гістограми точності та повноти для найбільших категорій.

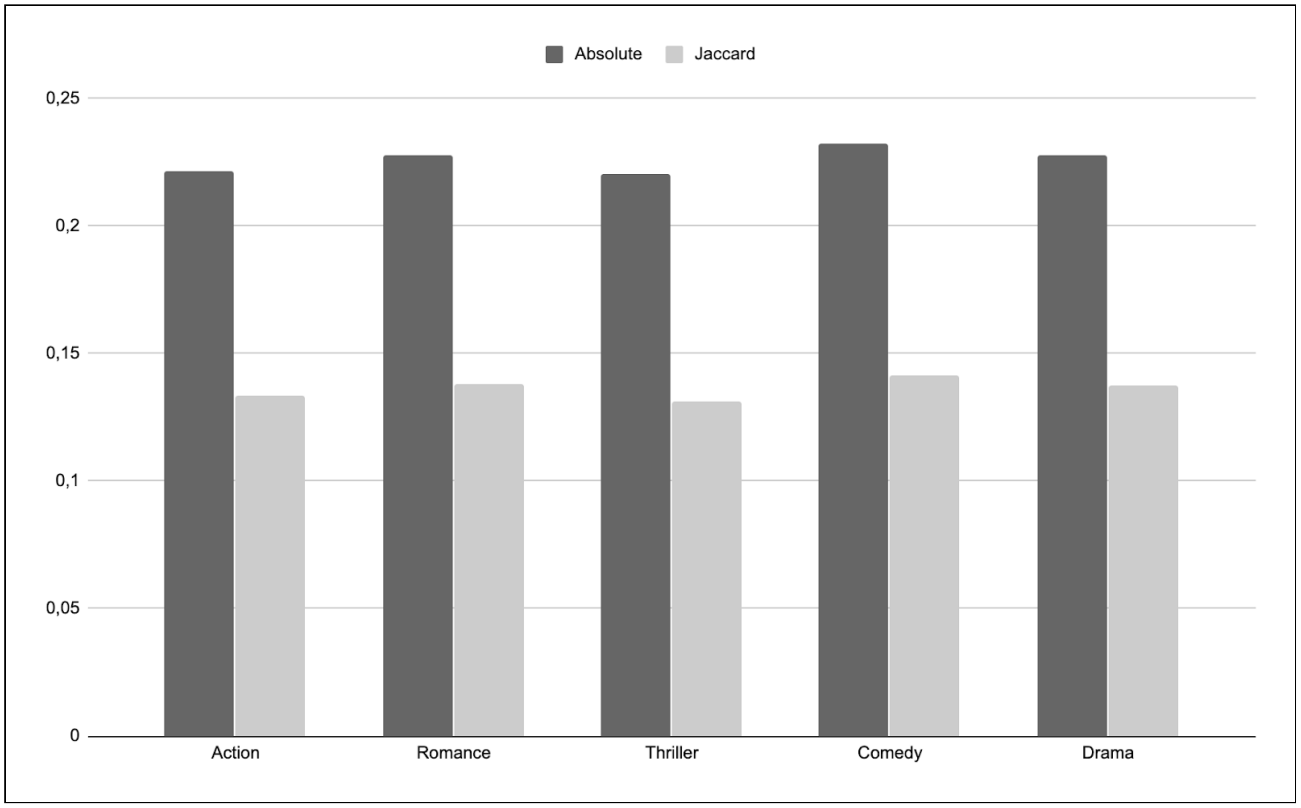


Рисунок 4.20 – Гістограми точності та повноти
для найбільших категорій фільмів

Як видно з таблиці 4.15 і гістограм 4.5.9, у середньому значення точності та повноти для п’яти категорій фільмів приблизно однакові, що свідчить про певну лексичну незалежність запропонованої інформаційної технології.

4.6 Аналіз отриманих результатів

Вибір лінгвістичного пакету DKPro Core для імплементації розробленої інформаційної технології обумовлено тим, що він є більшим за збір компонентів аналізу природно-мовного контенту, які взаємодіють між собою. Пакет було побудовано з метою підвищення продуктивності дослідників, що працюють з

автоматичним аналізом текстової інформації. Результати проведеного дослідження підтвердили правильність вибору DKPro Core, який дійсно забезпечує можливість зосередитися на реальних наукових задачах, а не на розробці програмного коду.

Проведені експериментальні дослідження продемонстрували переваги запропонованої інформаційної технології та перспективність її застосування. Найперше, що потрібно зазначити – підтверджено справедливість гіпотези та відповідної інформаційної оцінки з п.2.2 щодо лінійного зростання частоти значущих слів тексту, з яких обираються ключові, за рахунок врахування синтаксичних зв'язків між словоформами у реченнях англомовного тексту. Зокрема, у експерименті для текстів з офіційними англомовними перекладами Послань президента РФ до Федеральних зборів 2013 [148] та 2014 [148] років власна розробка порівнювалася з частотним статистичним методом.

Результати порівняння отриманих ключових слів для [148] свідчать, що при знаходження однакової кількості (10) значущих ключових слів власна розробка додає додатково 2 стоп-слова, а частотний словник – 11. Аналогічне порівняння для [149] демонструє схожу пропорцію надлишковості стоп-слів при пошуку 10 значущих ключових слів – власна розробка додає 4 стоп-слова, а частотний словник – 20. Це означає, що запропонований підхід забезпечує збільшення питомої ваги значущих ключових слів у межах від 0,33-0,48 до 0,71-0,83 (від 148% до 251%), при цьому приблизно на 80% стабільно відфільтровуються стоп-слова у кожному експерименті.

Порівняльний аналіз складу значущих ключових слів для цих двох текстів, отриманих відомим та запропонованим підходами, додатково продемонстрував досить показові результати, які певним чином виходять за межі формального лексичного аналізу. Наприкінці 2013 року президент РФ говорив: we, I, work, need, system. А рівно через рік, у 2014, він уже казав: I, Russia, people, Russian, work – застосування формальних засобів інформаційної

технології недвозначно свідчить про суттєву зміну у публічній політичній риториці сусідньої країни.

За результатами пошуку ключових слів для короткої анотації з 141 слова [127], видно, що власна розробка знаходить найбільше слів, заданих автором – 6 слів з 11. При цьому програми-аналоги знаходять для цього тексту по 5 слів. Отже, запропонована інформаційна технологія одночасно збільшує на 8,1% повноту за метрикою Жаккара (до 38%) та на 9,1% абсолютну точність (до 55%) пошуку ключових слів у порівнянні з програмами-аналогами.

Аналогічний експеримент було проведено для тексту статті, що має на порядок більше слів – 1460 [155]. Власна розробка стабільно демонструє кращі кількісні характеристики повноти і точності знаходження ключових слів, порівняно з аналогами – 40% та 57%. Для цього експерименту також одночасно підвищено на 12,7% повноту та на 14,3% точність отриманих ключових слів.

За рахунок використання удосконаленого методу зменшення впливу вербального шуму на пошук ключових слів вдалося суттєво зменшити або повністю виключити шумові слова при пошуку ключових слів англомовного тексту на основі інструментальних засобів пакету DKPro Core. Так, у розглянутому в п.4.3 тексту, що складається з двох англомовних речень, вдалося зменшити кількість кандидатів в ключові слова з 23 до 17 (на 35,3%), а також повністю видалити шумові слова (на 100%).

У більш масштабній апробації двох запропонованих методів у п.4.3 проведено експеримент з текстом [153], що складається з 1588 слів. Послідовне використання удосконаленого методу до зменшення шуму після основного методу пошуку ключових слів також дозволило покращити загальні результати – знайти три слова з заданих автором (participatory, historical, geography), тоді як без зменшення шуму було знайдено два слова, заданих автором (participatory, geography). Цим самим було збільшено на 7,5% повноту за метрикою Жаккара (до 20%) та на 11,1% абсолютну точність (до 33,3%) пошуку ключових слів.

Окрім того, проведено заміну 4-х шумових слів (100%) з 9-ти першого списку на 4 значущих слова-кандидата у ключові слова.

Масштабні експериментальні дослідження було проведено для оцінки перспективного напрямку впровадження запропонованої інформаційної технології, що полягає у пошуку ключових слів для дуже коротких англомовних текстів. Експериментальну базу склали близько 30 000 відібраних коротких анотацій до фільмів, що супроводжувалися заданими ключовими словами та були розбиті на категорії за жанрами.

Потрібно звернути увагу на явну “незручність” експериментального лексичного матеріалу, наприклад, у багатьох випадках еталонним є тільки одне ключове слово або еталонні ключові слова жодного разу не зустрічаються в тексті, текст анотації складається лише з кількох слів тощо. Не зважаючи на такі складнощі, запропонована інформаційна технологія знаходить одне і більше ключових слів, заданих автором для 42.3% опрацьованих анотацій. При цьому 51.2% усіх позитивних результатів отримано з немалою точністю від 0.2 до 0.55, а 41.1% позитивних результатів отримано з повнотою в діапазоні від 0.12 до 0.37.

Середнє значення повноти за Жаккаром і абсолютної точності для знайдених розробленою технологією ключових слів, відповідно складає 23.3% і 14.2%. Важливим фактом є те, що розроблена технологія у багатьох випадках правильно знайшла єдине еталонне ключове слово. Це свідчить про перспективність застосування технології у тих ситуаціях, коли потрібно знайти найбільш значуще слово з короткого тексту.

У середньому значення точності та повноти результатів пошуку ключових слів для анотацій за п’ятьма основними категоріями фільмів приблизно однакові, що дає підстави стверджувати про лексичну незалежність запропонованої інформаційної технології для англомовних текстів.

До обмежень у застосуванні запропонованої інформаційної технології можна віднести швидкодію її практичної реалізації засобами DKPro Core, зокрема, відносно задовгим для онлайн режиму є час створення багаторівневої розмітки тексту. Але це, в свою чергу, може бути виправлено за рахунок використання більш потужного апаратного забезпечення або платформ хмарних обчислень, що дозволяють мати у своєму розпорядженні віртуальний кластер комп'ютерів. Потрібного ступеню розпаралелювання процесів не важко досягти технологічно, оскільки додатки пошуку ключових слів і зменшення вербального шуму написані на Java та можуть бути легко розгорнуті на таких платформах.

Перспективою подальших досліджень пошуку ключових слів є проведення більш масштабних експериментів для текстів різних категорій з метою визначення додаткових шляхів підвищення релевантності запропонованої інформаційної технології. Доцільно також використання аналогічного функціоналу нових лінгвістичних пакетів, що підтримують більше мов, в тому числі і українську.

Впровадження запропонованої інформаційної технології не потребує додаткових витрат для компанії. Аналогами та перспективними об'єктами впровадження розробленої інформаційної технології можуть бути сайти SEO оптимізації з можливістю пошуку ключових слів, а також автоматизовані системи підтримки англомовного контент-аналізу.

4.7 Висновки до розділу

Для розробки програмного забезпечення було обрано мову програмування Java та лінгвістичний пакет DKPro Core – набір програмних компонентів для обробки природної мови, заснований на Apache UIMA framework. Наразі DKPro Core об'єднує 94 моделі на 15 природних мовах.

Запропоновано структуру інформаційної технології пошуку ключових слів, яка забезпечує послідовне застосування основного та додаткового методів,

обгрунтованих у розділі 3. Відповідно до отриманої структури розроблено програмне забезпечення у вигляді незалежних модулів DKPro Core.

Проведено серію експериментів для визначення кількісних характеристик релевантності отриманих результатів пошуку ключових слів. Аналіз результатів експерименту показав суттєві переваги запропонованої інформаційної технології за кількісними характеристиками порівняно з аналогами. Так, у порівнянні з частотним словником власна розробка забезпечує збільшення питомої ваги значущих ключових слів у межах від 0,33-0,48 до 0,71-0,83 (від 148% до 251%), при цьому приблизно на 80% стабільно відфільтровуються стоп-слова у кожному експерименті. У порівнянні з програмами-аналогами запропонована інформаційна технологія одночасно збільшує у межах від 8,1% до 12,7% повноту за метрикою Жаккара та від 9,1% до 14,3% абсолютну точність пошуку ключових слів для англомовних текстів обсягом 140-1400 слів.

Послідовне використання удосконаленого методу до зменшення шуму після основного методу пошуку ключових слів також дозволило покращити релевантність отриманих результатів. Зокрема, для тексту з приблизно 1600 слів було збільшено на 7,5% повноту за метрикою Жаккара (до 20%) та на 11,1% абсолютну точність (до 33,3%) пошуку ключових слів. Ще однією суттєвою перевагою в порівнянні з аналогами є те, що запропонована інформаційна технологія пошуку ключових слів дозволяє повністю (до 100%) виключити шумові слова.

Масштабний есперимент на основі 30 000 відібраних коротких анотацій до фільмів з відомими (заданими авторами) ключовими словами також продемонстрував перспективні результати. На “незручному” для такої задачі матеріалі запропонована інформаційна технологія знайшла одне і більше ключових слів, заданих автором для 42.3% опрацьованих анотацій. При цьому 51.2% усіх позитивних результатів отримано з немалою точністю від 20% до 55%, а 41.1% позитивних результатів отримано з повнотою в діапазоні від 12%

до 37%. Середнє значення повноти за Жаккаром і абсолютної точності для знайдених розробленою технологією ключових слів, відповідно складає 23.3% і 14.2%.

Отже, запропонований метод пошуку ключових слів дає змогу підвищити чисельні характеристики якості пошуку ключових слів, а саме повноту за метрикою Жаккара і абсолютну точність. Удосконалений метод зменшення впливу вербального шуму на пошук ключових слів дозволив підвищити якість отриманих ключових слів у порівнянні з основним методом.

Набула подальшого розвитку інформаційна технологія пошуку ключових слів, яка, на відміну від існуючих, враховує додаткову інформацію процесів парсингу речень у межах послідовного застосування двох запропонованих методів, що дозволило уточнити чисельні оцінки змістовних параметрів тексту та підвищити якість пошуку його ключових слів. Визначено обмеження та перспективні напрями впровадження розробленої інформаційної технології.

ВИСНОВКИ

Внаслідок проведеного дослідження було вирішене актуальне наукове завдання підвищення якості пошуку ключових слів у англомовному тексті шляхом розробки інформаційної технології пошуку ключових слів на основі парсингу англомовних текстів. Мета і задачі дослідження формувалися на гіпотезі про те, що підвищення якості процесу та результатів пошуку ключових слів для природно-мовного тексту вимагає залучення додаткової інформації про цей текст, причому універсального, а не специфічного характеру.

В роботі запропоновано підхід до пошуку ключових слів, що базується на використанні додаткової інформації універсального характеру про складні залежності між членами англомовного речення. За результатами математичного моделювання формалізовано задачу пошуку ключових слів тексту як параметричну ідентифікацію функції згортки вербальної інформації за критерієм максимума інформації у зв'язках, які поєднують n обраних ключових слів тексту T між собою та з усіма m' значущими словами цього тексту.

У межах запропонованої математичної моделі обгрунтовано обов'язкове врахування 38 значущих типів зв'язків в процесі пошуку ключових слів та виключення з процесу аналізу тексту 7-ми неінформативних типів зв'язків, а також 21 тегу, якими позначаються неінформативні частин мови. Такий вибір може вважатися головним обмеженням моделі, що пропонується для англомовних текстів.

Удосконалено модель пошуку ключових слів, яка, на відміну від існуючих, побудована на основі інформаційної оцінки результатів парсингу тексту та враховує результати аналізу зв’язків між лексичними одиницями тексту, що дозволило формалізувати критерій якості процесу пошуку ключових слів.

Уперше розроблено метод пошуку ключових слів, який, на відміну від існуючих, базується на знаходженні синтаксичних зв’язків між словоформами у реченнях англомовного тексту за допомогою технологічних можливостей парсингу сучасних лінгвістичних пакетів. Запропонований метод дає змогу підвищити чисельні характеристики якості пошуку ключових слів, а саме повноту (за метрикою Жаккара) і точність.

Запропоновано використовувати комбінований підходу для зменшення колізії при знаходженні ключових слів, зокрема спочатку перевіряти ключові слова з однаковою частотою на зв'язність. На другому етапі, якщо у блоці потенційних ще залишилися ключові слова з однаковою частотою, вибираються спочатку іменники, потім дієслова, а потім інші частини мови. Зменшення кількості шумових слів пропонується досягти за допомогою комбінації підходів: заміна займенників на відповідні до них іменники; вилучення словосполучень із типами зв’язків, які не несуть суттєвого смислового навантаження; вилучення слів, які відносяться до неінформативних частин мови; вилучення слів, які відносяться до списку стоп-слів.

Удосконалено метод зменшення впливу вербального шуму на пошук ключових слів, який, на відміну від існуючих, побудовано на основі стенфордської класифікації зв'язків між лексичними одиницями речення, що дозволило підвищити якість отриманих ключових слів у порівнянні з основним методом.

Для розробки програмного забезпечення було обрано мову програмування Java та лінгвістичний пакет DKPro Core – набір програмних компонентів для обробки природної мови, заснований на Apache UIMA framework. Розроблені структура інформаційної технології та відповідне програмне забезпечення.

Експериментально підтверджено теоретичні оцінки формальних меж лінійного збільшення кількості інформації для значущих слів тексту, з яких обираються ключові, внаслідок проведення парсингу та врахування результатів

аналізу зв’язків між лексичними одиницями тексту. Досягнуто збільшення питомої ваги значущих ключових слів у межах від 0,33-0,48 до 0,71-0,83 (від 148% до 251%) у порівнянні з частотним словником.

Експериментальна апробація запропонованої інформаційної технології показала її переваги за кількісними характеристиками. Так, у порівнянні з 3-ма програмами-аналогами запропонована інформаційна технологія одночасно збільшує у межах від 8,1% до 12,7% повноту за метрикою Жаккара та від 9,1% до 14,3% абсолютну точність пошуку ключових слів для англомовних текстів обсягом 140-1400 слів. Послідовне використання удосконаленого методу до зменшення шуму після основного методу пошуку ключових слів дозволило для тексту з приблизно 1600 слів підвищити на 7,5% повноту за метрикою Жаккара (до 20%) та на 11,1% абсолютну точність (до 33,3%) пошуку ключових слів, при цьому повністю відфільтровано шумові слова.

Масштабний експеримент на основі 30 000 відібраних коротких анотацій до фільмів з відомими (заданими авторами) ключовими словами також продемонстрував перспективні результати. На максимально “незручному” для такої задачі природно-мовному матеріалі запропонована інформаційна технологія знайшла одне і більше ключових слів, заданих автором для 42.3% опрацьованих анотацій. Середнє значення повноти точності для знайдених розробленою технологією ключових слів відповідно складає 23.3% і 14.2%.

Висока актуальність отриманої розробки для задач комп'ютерної лінгвістики пояснюється тим, що запропоновані моделі, методи і алгоритми пошуку ключових слів базуються на знаходженні синтаксичних зв’язків між словоформами у реченнях англомовного тексту, що дає змогу підвищити чисельні характеристики якості отриманих ключових слів. Визначено часткові обмеження розробленої інформаційної технології та перспективні напрями її впровадження, зокрема сайти SEO оптимізації з можливістю пошуку ключових слів, а також автоматизовані системи підтримки англомовного контент-аналізу.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

[1] Ю.С. Ершов, "Выделение ключевых слов в русскоязычных текстах", *Молодежный научно-технический вестник. - М.: ФГБОУ ВПО "МГТУ им. Н.Э. Баумана"* № ФС77-51038, с. 70-79, 2014.

[2] C. Zhang, H. Wang, Y. Liu, D. Wu, Y. Liao, and B. Wang, "Automatic keyword extraction from documents using conditional random fields", *Journal of Computational Information Systems №4,* pp. 1169-1180, 2008.

[3] R. Feldman, and J. Sanger, "Task-Oriented approaches", *The text mining handbook: advanced approaches in analyzing unstructured data, Cambridge: Cambridge Univ. Pr.*, pp. 58-62, 2013.

[4] J. Mijic, B. Bašic, and J. Šnajder, "Robust keyphrase extraction for a large-scale Croatian news production system", *FASSBL7*, pp. 59-66, 2010.

[5] G. Palshikar, "Keyword extraction from a single document using centrality measures", *Lecture Notes in Computer Science Pattern Recognition and Machine Intelligence,* pp. 503-510, 2007.

[6] R. Mihalcea, and P. Tarau, "Textrank: Bringing order into text", *Proceedings of the 2004 conference on empirical methods in natural language processing,* pp. 404-411, 2004.

[7] S. Lahiri, S. Choudhury, and C. Caragea, "Keyword and keyphrase extraction using centrality measures on collocation networks", arXiv preprint arXiv:1401.6571, 2014.

[8] F. Boudin "Comparison of Centrality Measures for Graph-Based Keyphrase Extraction", *Proceedings of the Sixth International Joint Conference on Natural Language Processing (IJCNLP),* pp. 834-838, 2013.

[9] К. Н. Даркулова, и Г. Ергешова, "Необходимость выделения ключевых слов для свёртывания текста", [9] *Лингвистический анализ научного текста. VI Международная студенческая электронная научная конференция,*

Южно-Казахстанский государственный университет им. Мухтара Ауэзова Шымкент, с. 30-35, 2014.

[10] O. Bisikalo, A. Lisovenko, O. Jahumovuch, S. Trachenko, and M. Pradivliannyi, "System of computational linguistic on base of the figurative text comprehension", *Proceedings of the 2016 IEEE 1st International Conference on Data Stream Mining and Processing (DSMP 2016),* pp. 69-74, 2016. DOI: 10.1109/DSMP.2016.7583510.

[11] A. Hulth, "Improved automatic keyword extraction given more linguistic knowledge", *Proceedings of the 2003 conference on Empirical methods in natural language processing*, pp. 216-223, 2003. doi:10.3115/1119355.1119383

[12] Y. HaCohen-Kerner, "Automatic extraction of keywords from abstracts", *International Conference on Knowledge-Based and Intelligent Information and Engineering Systems,* pp. 843-849, 2003.

[13] N. Pudota, A. Dattolo, A. Baruzzo, and C. Tasso, "A New Domain Independent Keyphrase Extraction System", *Communications in Computer and Information Science Digital Libraries*, pp. 67-78, 2010. doi: 10.1007/978-3-642-15850-6_8

[14] В. Шорошев и Е. Маевский, "Методические основы эффективного поиска информации в сети интернет", *Правове, нормативне та метрологічне забезпечення системи захисту інформації в Україні*, № 4, с. 81-88, 2002.

[15] А.М. Андреев, Д.В.Березкин, В.В. Сюзев, и В.И. Шабанов, "Модели и методы автоматической классификации текстовых документов", *Вестн. МГТУ. Сер. Приборостроение,* МГТУ, №3, с. 64-94, 2003.

[16] T. Joachims, "Probabilistic Analysis of the Rocchio Algorithm with TFIDF for Text Categorization", *Carnegie-mellon Univ Pittsburgh PA Dept of computer science,* No. CMU-CS-96-118, pp. 143-151, 1996.

[17] N. Fuhr, N. Govert, M. Lalmas, and F. Sebastiani, "Categorisation tool: Final prototype. Deliverable 4.3, Project LE4-8303 «EUROSEARCH»", *Commission of the European Communities*, 30 p. 1999.

[18] L. Larkey, and W. Croft, "Combining classifiers in text categorization. In Proceedings of SIGIR 96", *19th ACM International Conference on Research and Developmentin Information Retrieval*, pp. 289-297, 1996.

[19] T. Joachims, "A probabilistic analysis of the rocchio algorithm with TFIDF for text categorization", *In Proc. of the ICML'97*, pp. 289-297, 1997.

[20] M. Litvak, M. Last, H. Aizenman, I. Gobits, and A. Kandel, "DegExt - A language-independent graph-based keyphrase extractor", *Advances in intelligent web mastering - 3*, pp. 121-130, 2011. doi:10.1007/978-3-642-18029-3_13

[21] W. Abilhoa, and L. Castro, "A keyword extraction method from twitter messages represented as graphs", *Applied Mathematics and Computation, №240*, pp. 308-325, 2014. doi:10.1016/j.amc.2014.04.090

[22] Z. Zhou, X. Zou, X. Lv, and J. Hu, "Research on Weighted Complex Network Based Keywords Extraction", *Lecture Notes in Computer Science Chinese Lexical Semantics, №8229*, pp. 442-452, 2013. doi:10.1007/978-3-642-45185-0_47

[23] X. Wan, and J. Xiao, "Single Document Keyphrase Extraction Using Neighborhood Knowledge", *Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence,* pp. 855-860, 2008.

[24] Е.Г. Абрамов, "Подбор ключевых слов для научной статьи", *Научная периодика: проблемы и решения, № 1(2)б*, pp. 35-40, 2011.

[25] Как составить список ключевых слов?, [Электронный ресурс]. Режим доступа: http://blog.creativeconomy.ru/2009/04/02/kak-sostavit-spisok-klyuchevyx-slov/.

[26] J. Borge-Holthoefer, and A. Arenas, "Semantic Networks: Structure and Dynamics", *Entropy, Vol. 12, № 5*, pp. 1264-1302, 2010. doi: 10.3390/e12051264

[27] A. Masucci, and G. Rodgers, "Differences Between Normal And Shuffled Texts: Structural Properties Of Weighted Networks", *Advances in Complex Systems, Vol. 12, № 01*, pp. 113-129, 2009. doi: 10.1142/s0219525909002039

[28] И.Е. Воронина, А.А. Кретов, и О.С. Титова, "Программные средства выявления семантического поля слов", *Вестн. Воронеж. гос. ун-та. Серия Системный анализ и информационные технологии, № 2,* с. 111-122, 2008.

[29] В.Т. Титов, *Частная квантитативная лексикология романских языков: Монография,* Воронеж: Изд-во Воронеж. гос. ун-та, 552 с. 2004.

[30] А.А. Кретов, "Метод формального выделения тематически нейтральной лексики (на примере старославянских текстов)", *Вестн. Воронеж. гос. ун-та. Серия Системный анализ и информационные технологии, № 1*, с. 81-90, 2007.

[31] Функциональный подход к выделению ключевых слов: методика и реализация, [Электронный ресурс]. Режим доступа: http://www.vestnik.vsu.ru/pdf/analiz/2009/01/2009-01-10.pdf

[32] И.Е. Воронина, *Компьютерное моделирование лингвистических объектов: монография,* Воронеж: Издательско- полиграфический центр ВГУ, 177 с. 2007.

[33] Е.В. Крештель, А.А. Кретов, и И.Е. Воронина, "Алгоритмы выделения ключевых слов в текстах на естественных языках", *Информатика: проблемы, методология, технологии : матер. 3-й регион. науч.-метод. конфер.* Воронеж. Изд-во ВГУ, с. 35, 2001.

[34] Программные средства выявления семантического поля слов, [Электронный ресурс]. Режим доступа: http://www.vestnik.vsu.ru/pdf/analiz/2008/02/2008_02_19.pdf

[35] D. Bourigault, "Surface Grammatical Analysis for the Extraction of Terminological Noun Phrases", *Proceedings of the 14th conference on Computational linguistics-Volume 3. Association for Computational Linguistics.* Nantes, France, pp. 977-981, 1992.

[36] П.И. Браславский, и Е.А. Соколов, "Автоматическое извлечение терминологии с использованием поисковых машин Интернета", *Компьютерная лингвистика и интеллектуальные технологии: Труды Международной конференции Диалог*. М.: РГГУ, с. 89-94, 2007.

[37] M. Baroni, "BootCaT: Bootstrapping Corpora and Terms from the Web", *Proceedings of LREC. Lisbon: ELDA,* pp. 1313-1316, 2004.

[38] K. Frantzi, S. Ananiadou, and H. Mima, "Automatic recognition of multi-word terms:. the C-value/NC-value method", *International Journal on Digital Libraries, Vol. 3*, pp. 115-130, 2000.

[39] Б.В. Добров, Н.В. Лукашевич, и С.В. Сыромятников, "Формирование базы терминологических словосочетаний по текстам предметной области", *Электронные библиотеки: перспективные методы и технологии, электронные коллекции: Труды пятой Всероссийской научной конференции,* с. 201-210, 2003.

[40] П.И. Браславский, и Е.А. Соколов, "Сравнение четырех методов автоматического извлечения двухсловных терминов из текста", *Компьютерная лингвистика и интеллектуальные технологии: Труды Междунар. конф. Диалог'2006*. - М.: РГГУ, с. 88-94, 2006.

[41] С.Д. Шелов, "Терминоведение: семь вопросов и семь ответов по семантике термина", *Информационные процессы и системы, №2,* с. 1-12, 2001.

[42] Синтаксический анализ. Проект АОТ, [Электронный ресурс]. Режим доступа: http://www.aot.ru/docs/synan.html

[43] П.И. Браславский, и Е.А. Соколов, "Сравнение пяти методов извлечения терминов произвольной длины", *Компьютерная лингвистика и интеллектуальные технологии: По материалам ежегодной Международной конференции «Диалог»,* с. 67-74, 2008.

[44] О.В. Пескова, "Автоматическое формирование рубрикатора полнотекстовых документо", *Электронные библиотеки: перспективные методы и технологии, электронные коллекции,* с. 139-148, 2008.

[45] О.В. Бісікало, та О.В. Яхимович, "Автоматизоване визначення лексичних технологій з тезаурусу технічного спрямування", *Оптико-електронні інформаційно-енергетичні технології*, № 1 (31), с. 26-38, 2016. ISSN 1681-7893.

[46] O. Medelyan, and I. Witten, "Thesaurus based automatic keyphrase indexing", *Proceedings of the 6th ACM/IEEE-CS joint conference on Digital libraries,* pp. 296-297, 2006. doi:10.1145/1141753.1141819

[47] J. Bezdek, and N. Pal, "Some New Indexes of Cluster Validity", *IEEE Transactions On Systems, Man And Cybernetics, Vol. 28, no. 3,* pp. 301-315, 1998.

[48] M. Halkidi, V. Batistakis, and M. Vazirgiannis, "On Clustering Validation", *Journal of Intelligent Information Systems, Vol. 17,* pp. 107-145, 2001.

[49] В.Б. Барахнин, и Д.А. Ткачев, "Кластеризация текстовых документов на основе составных ключевых термов", *Вестник НГУ. Серия: Информационные технологии, № 8(2),* с. 5-14, 2010.

[50] А.М. Федотов, и В.Б. Барахнин, "К вопросу о поиске документов «по аналогии»", *Вестн. Новосиб. гос. ун-та. Серия: Информационные технологии, Т. 7, № 4*, с. 3-14, 2009.

[51] В.Б. Барахнин, В.А. Нехаева, и А.М. Федотов, "О задании меры сходства для кластеризации текстовых документов", *Вестн. Новосиб. гос. ун-та. Серия: Информационные технологии, Т. 6, № 1,* с. 3-9, 2008.

[52] E. Martin, "Microblogging - more than fun?", *Proceedings of IADIS Mobile Learning Conference. - Inmaculada Arnedillo Sánchez and Pedro Isaías ed., Portugal,* pp. 155-159, 2008.

[53] D. Herman, M. Janh, and M. Ryan, "The Routledge Encyclopedia of Narrative Theory", *London, Routledge,* 997 p, 2005.

[54] M. Böhringer, "Really Social Syndication: A Conceptual View on Microblogging", *Sprouts: Working Papers on Information Systems, № 9(31)*, pp. 147-157, 2009.

[55] D. Karger, and D. Quan, "What would it mean to blog on the semantic web?", *Web Semantics: Science, Services and Agents on the World Wide Web. - Selected Papers from the International Semantic Web Conference, Hiroshima, Japan, №3,* pp. 143-147, 2004.

[56] P. Turney, "Learning to extract keyphrases from text", *Technical report, National Research Council, Institute for Informational Technology,* pp. 143-147, 1999.

[57] P. Chen, and S. Lin, "Automatic keyword prediction using Google similarity distance", *Expert Systems with Applications, №37,* pp. 1928-1938, 2010. doi:10.1016/j.eswa.2009.07.016

[58] F. Sebastiani, "Machine learning in automated text categorization", *ACM computing surveys (CSUR), №34,* pp. 1-47, 2002.

[59] Y. Hacohen, Z. Gross, and A. Masa, "Automatic Extraction and Learning of Keyphrases from Scientific Articles", *Computational Linguistics and Intelligent Text Processing Lecture Notes in Computer Science,* pp. 657-669, 2005. doi:10.1007/978-3-540-30586-6_74

[60] M. Krapivin, A. Autayeu, M. Marchese, E. Blanzieri, and N. Segata, "Keyphrases Extraction from Scientific Documents: Improving Machine Learning Approaches with Natural Language Processing", *International Conference on Asian Digital Libraries Lecture Notes in Computer Science, №6102,* pp. 102-111, 2010. doi:10.1007/978-3-642-13654-2_12

[61] M. Bekavac, and J. Šnajder, "Gpkex: Genetically programmed keyphrase extraction from croatian texts", *Proceedings of the 4th Biennial International Workshop on Balto-Slavic Natural Language Processing,* pp. 43-47, 2013.

[62] R. Ahel, B. Dalbelo Bašic, and J. Šnajder, "Automatic keyphrase extraction from Croatian newspaper articles", *The Future of Information Sciences, Digital Resources and Knowledge Sharing,* pp. 207-218, 2009.

[63] D. Turdakov, "Word sense disambiguation methods", *Programming and Computer Software, Vol. 36, № 6*, pp. 309-326, 2010.

[64] D. Turdakov, and S. Kuznetsov, "Automatic word sense disambiguation based on document networks", *Programming and Computer Software, Vol. 36, № 1,* pp. 11-18, 2010.

[65] D. Lizorkin, P. Velikhov, M. Grinev, and D. Turdakov, "Accuracy estimate and optimization techniques for SimRank computation", *The International Journal on Very Large Data Bases archive, Vol. 19, № 1,* pp. 15-19, 2010.

[66] D. Lizorkin, P. Velikhov, M. Grinev, and D. Turdakov, "Accuracy Estimate and Optimization Techniques for SimRank Computation", *Proceedings of the VLDB Endowment, Vol. 1, № 1,* pp. 12-17, 2008.

[67] M. Grinev, D. Lizorkin, and D. Turdakov, "Effective Extraction of Thematically Grouped Key Terms From Text", *Proc. of the AAAI 2009 Spring Symposium on Social Semantic Web,* pp. 39-44, 2009.

[68] D. Turdakov, and D. Lizorkin, "HMM Expanded to Multiple Interleaved Chains as a Model for Word Sense Disambiguation", *The 23rd Pacific Asia Conference on Language, Information and Computations,* pp. 549-559, 2009.

[69] M. Grineva, M. Grinev, and D. Lizorkin, "Extracting Key Terms From Noisy and Multitheme Documents", *WWW2009: 18th International World Wide Web Conference*, pp. 521-533, 2009.

[70] A. Guo, and Y. Tao, "Research and Improvement of Feature Words Weight Based on TFIDF Algorithm", *IEEE Information Technology, Networking, Electronic and Automation Control Conference (May 2016)*, 2016. doi:10.1109/itnec.2016.7560393

[71] M. Grineva, M. Grinev, A. Boldakov, L. Novak, A. Syssoev, and D. Lizorkin, "Sifting Micro-blogging Stream for Events of User Interest", *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pp. 327-333, 2009.

[72] J. Reed, Y. Jiao, T. Potok, B. Klump, M. Elmore, and A. Hurson, "TF-ICF: A New Term Weighting Scheme for Clustering Dynamic Data Streams", *Proc. Machine Learning and Applications,* pp. 258-263, 2006.

[73] R. Mihalcea, and A. Csomai, "Wikify!: linking documents to encyclopedic knowledge", *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management,* pp. 233-242, 2007.

[74] G. Salton, "The SMART Retrieval System - experiments in automatic", *Prentice-Hall, Inc., Englewood Cliffs*, pp. 111-121, 1971.

[75] Z. Dejin, and M. Rosson, "How and why people Twitter: the role that micro blogging plays in informal communication at work", *Proceedings of the ACM 2009 international conference on Supporting group work*, pp. 31-40, 2009.

[76] P. McFedries, "All A-Twitter", IEEE Spectrum, № 84, pp. 14-18, 2007.

[77] A. Java, X. Song, T. Finin, and B. Tseng, "Why we twitter: understanding microblogging usage and communities", *Proc. WebKDD/SNA-KDD '07. ACM Press,* pp. 47-50, 2007.

[78] B. Krishnamurthy, P. Gill, and M. Arlitt, "A few chirps about twitter", *Proc. WOSP '08. - ACM Press,* pp. 33-36, 2008.

[79] C. Honeycutt, and S. Herring, "Beyond microblogging: Conversation and collaboration via Twitter", *Proc. HICSS '09. - IEEE Press,* pp. 6-9, 2009.

[80] M. Naaman, J. Boase, and C. Lai, "Is it really about me? Message content in social awareness streams", *Proc. CSCW 2010,* pp. 58-63, 2010.

[81] B. Huberman, D. Romero, and F. Wu, "Social networks that matter: Twitter under the microscope", *First Monday, Vol. 14, №1,* pp. 14-21, 2008.

[82] А.В. Коршунов, "Извлечение ключевых терминов из сообщений микроблогов с помощью Википедии", *Труды Института системного программирования РАН, №20,* с. 102-115, 2011.

[83] A. Özgür, J. Hur, and Y. He, "The Interaction Network Ontology-Supported Modeling and Mining of Complex Interactions Represented with Multiple Keywords in Biomedical Literature.", *BioData Mining 9, no. 1 (December 2016)*, doi:10.1186/s13040-016-0118-0

[84] W. Wong, W. Liu, and M. Bennamoun, "Ontology Learning from Text", *ACM Computing Surveys 44, no. 4 (August 1, 2012),* pp. 1-36, 2012. doi:10.1145/2333112.2333115

[85] D. Korobkin, S. Fomenkov, and S. Kolesnikov, "Method of ontology-based extraction of physical effect description", *Vestnik Komp'iuternykh i Informatsionnykh Tekhnologii (2015),* pp. 28-35, 2015. doi:10.14489/vkit.2015.02.pp.028-035

[86] Л.Л. Волкова, "Приложения теории тесного мира в компьютерной лингвистике", *3-я Международная научно-практическая конференция «Модель подготовки специалистов новой формации, адаптированных к инновационному развитию отраслей»: сборник трудов. - Душанбе,* с. 172-155, 2012.

[87] Е.И. Большакова, Э.С. Клышинский, Д.В. Ландэ, А.А. Носков, О.В. Пескова, и Е.В. Ягунова, *Автоматическая обработка текстов на естественном языке и компьютерная лингвистика: учеб. пособие*, МИЭМ, 272 с. 2011.

[88] Бесплатный онлайн-генератор ключевых слов с текста [Электронный ресурс]. Режим доступа: http://seotool.by/analiz/seo/keywordstext.php

[89] Генератор ключевых слов с текста., [Электронный ресурс]. Режим доступа: http://www.rise-top.com/keywordstext.php

[90] R. Mihalcea, and A. Csomai, "Wikify!: linking documents to encyclopedic knowledge", *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management. Lisbon,* pp. 233-242, 2007. doi: http://doi.org/10.1145/1321440.1321475

[91] J. Wu, and A. Agogino, "Automating keyphrase extraction with multi-objective genetic algorithms", *37th Annual Hawaii International Conference on System Sciences, 2004. Proceedings of the IEEE*, 2004.

[92] Y. Zhang, E. Milios, and N. Zincir-Heywood, "A comparison of keyword-and keyterm-based methods for automatic web site summarization", *AAAI04 Workshop on Adaptive Text Extraction and Mining,* pp. 15-20, 2004.

[93] С.В. Кулешов, А.А. Зайцева, и В.С. Марков, "Ассоциативно-онтологический подход к обработке текстов на естественном языке", *Интеллектуальные технологии на транспорте, № 4,* с. 40-45, 2015.

[94] О.В. Яхимович, "Формалізація задачі визначення ключових слів тексту", *XLVIII науково-технічній конференції підрозділів ВНТУ: факультет комп'ютерних систем та автоматики*, Вінниця, 2019, с. 1136-1138.

[95] A set of Unicode character values, [Online]. Available: http://www.unicode.org/reports/tr44/#General_Category_Values

[96] Universal Dependencies (UD): Introduction, [Online]. Available: http://universaldependencies.org/introduction.html

[97] S. Sonawane, and P. Kulkarni, "Graph based Representation and Analysis of Text Document: A Survey of Techniques", *International Journal of Computer Applications, Vol. 96, № 19,* pp. 1-8, 2014. doi: 10.5120/16899-6972

[98] Universal Dependency Relations, [Online]. Available: http://universaldependencies.org/u/dep/

[99] C. Manning, and M. de Marneffe, Stanford typed dependencies manual, [Online]. Available: https://nlp.stanford.edu/software/dependencies_manual.pdf

[100] Stanford dependency hierarchy, [Online]. Available: https://nlp-ml.io/jg/software/pac/standep.html

[101] О.В. Бісікало, та О.В. Яхимович, "Знаходження ключових слів англомовного тексту за допомогою інструментальних засобів пакету DKPro Core", *Інформаційні технології та комп'ютерна інженерія*, № 2(34), с. 10-14, 2015. ISSN 1999-9941.

[102] О.В. Бісікало, "Концептуальна модель системи образного аналізу і синтезу природно-мовних конструкцій", *Математичні машини і системи, № 2,* с. 184-187, 2013.

[103] О.В. Бісікало, *Формальні методи образного аналізу та синтезу природно-мовних конструкцій : монографія,* Вінниця : ВНТУ, 316 с. 2013.

[104] С. Турчиняк, "Критерії відбору лексичного матеріалу та процес засвоєння лексичних одиниць", *V Міжнародна науково-практична інтернет-конференція «Проблеми та перспективи розвитку науки на початку третього тисячоліття у країнах СНД».* - Переяслав-Хмельницький, 17 - 19 листопада 2012, с. 211-213 2012.

[105] Слово як лексична одиниця, [Електронний ресурс]. Режим доступу: http://school-world.com.ua/lib/slovo-yak-leksichna-odinicya/

[106] Determiner, [Online]. Available: http://universaldependencies.org/u/dep/det.html

[107] O. Bisikalo, and A. Yahimovich. *Keyword search based on lexical relationships in the text*. Beau Bassin-Rose Hill, Mauritius: Lap Lambert Academic Publishing, 2019. ISBN 978-620-0-00314-0.

[108] Welo, E. "Null Anaphora", Encyclopedia of Ancient Greek Language and Linguistics (2013), [Online]. Available: https://referenceworks.brillonline.com/entries/encyclopedia-of-ancient-greek-language-and-linguistics/*COM_00000254

[109] Fixed multiword expression, [Online]. Available: http://universaldependencies.org/u/dep/fixed.html

[110] Punctuation, [Online]. Available: http://universaldependencies.org/u/dep/punct.html

[111] Root, [Online]. Available: http://universaldependencies.org/u/dep/root.html

[112] O. Bisikalo, and A. Yahimovich. *Lexical relationships-based keywords selection in english texts*. Vinnytsya, Ukraine: VNTU, 2020.

[113] A. Taylor, M. Marcus, and B. Santorini, The Penn Treebank: An Overview", [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.9.8216&rep=rep1&type=pdf

[114] Penn Treebank II Constituent Tags: Word level, [Online]. Available: http://www.surdeanu.info/mihai/teaching/ista555-fall13/readings/PennTreebankConstituents.html#Word

[115] Alphabetical list of part-of-speech tags used in the Penn Treebank Project, [Online]. Available: https://www.ling.upenn.edu/courses/Fall_2003/ling001/penn_treebank_pos.html

[116] О.В. Бісікало, та О.В. Яхимович, "Метод визначення ключових слів англомовного тексту на основі DKPro Core", *Технологический аудит и резервы производства: Информационные технологии*, Том 1, № 2(21), с. 26-30, 2015. ISSN 2226-3780.

[117] Интегрированные пакеты, [Электронный ресурс]. Режим доступа: https://nlpub.ru/%D0%9E%D0%B1%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B0_%D1%82%D0%B5%D0%BA%D1%81%D1%82%D0%B0#.D0.98.D0.BD.D1.82.D0.B5.D0.B3.D1.80.D0.B8.D1.80.D0.BE.D0.B2.D0.B0.D0.BD.D0.BD.D1.8B.D0.B5_.D0.BF.D0.B0.D0.BA.D0.B5.D1.82.D1.8B

[118] Natural Language Toolkit, [Online]. Available: http://www.nltk.org/

[119] О.В. Бісікало, О.В. Яхимович, А.І. Лісовенко, та Траченко С.С., "Моделювання процесів побудови парадигматичних зв’язків між словоформами на основі вимірювання текстової інформації", *Вимірювання, контроль та діагностика в технічних системах (ВКДТС-2015)*, Вінниця, 2015, с. 119-121.

[120] A collection of software components for natural language processing (NLP) based on the Apache UIMA framework, [Online]. Available: https://dkpro.github.io/dkpro-core/

[121] Stanford NLP, [Online]. Available: https://nlp.stanford.edu/software/index.shtml

[122] SharpNLP - open source natural language processing tools, [Online]. Available: https://archive.codeplex.com/?p=sharpnlp

[123] MeTA: ModErn Text Analysis, [Online]. Available: https://meta-toolkit.org/

[124] Apache OpenNLP, [Online]. Available: https://nlpub.ru/Apache_OpenNLP

[125] О.В. Яхимович, "Визначення ключових слів з тексту повідомлень мікроблогів", *XLV науково-технічній конференції підрозділів ВНТУ: факультет комп'ютерних систем та автоматики*, Вінниця, 2016, с. 1119-1121.

[126] О.В. Яхимович, "Колізія при знаходженні ключових слів", *XLVI науково-технічній конференції підрозділів ВНТУ: факультет комп'ютерних систем та автоматики*, Вінниця, 2017, с. 1312-1314.

[127] L. Burgareli, "Variability management in software product lines using adaptive object and reflection. Journal of Aerospace Technology and Management, V. 1, № 2.", *Journal of Aerospace Technology and Management, V. 1, № 2. (2009, Jul.-Dec.)* [Online]. Available: http://www.jatm.com.br/papers/vol1_n2/JATMv1n2_thesis_abstracts.pdf

[128] N. Astrakhantsev, "Automatic Term Acquisition from Domain-Specific Text Collection by Using Wikipedia.", *Proceedings of the Institute for System Programming of RAS 26, no. 4,* pp. 7-20, 2014. doi:10.15514/ispras-2014-26(4)-1

[129] Л.А. Гращенко, "О модельном стоп-словаре", *Известия Академии наук Республики Таджикистан. Отделение физико-математических, химических, геологических и технических наук, № 1 (150),* с. 40-46, 2013.

[130] О.В. Яхимович, "Зменшення вербального шуму при визначенні ключових слів", *XLVII науково-технічній конференції підрозділів ВНТУ: факультет комп'ютерних систем та автоматики*, Вінниця, 2018, с. 1463-1465.

[131] Natural Language Processing: Integration of Automatic and Manual Analysis, [Online]. Available: http://tuprints.ulb.tu-darmstadt.de/4151/1/rec-thesis-final.pdf

[132] О.В. Бісікало, Р.Н. Квєтний, С.Г. Кривогубченко, Л.Є Азарова, та О.В. Яхимович, "Синтез інтегрованої бази знань природно-мовного контенту", ВНТУ, Вінниця, Д/б № 0114U003462, 2015.

[133] О.В.Бісікало, Р.Н. Квєтний, С.Г. Кривогубченко, А.І. Лісовенко, та О.В. Яхимович, "Розв’язання семантико-залежних задач обробки природно-мовних об’єктів на основі бази знань", ВНТУ, Вінниця, Д/б № 0114U003462, 2016.

[134] О.В. Бісікало, О.В. Яхимович, А.І. Лісовенко, та Траченко С.С., "Підтримка діалогу з навчальним контентом", *Адаптивні технології управління навчанням: матеріали першої міжнародної конференції*, Одеса, 2015, с. 97-100.

[135] О. В. Бісікало, А. І. Лісовенко, О. В. Яхимович, та В. В. Шолота, "Спосіб автоматичного пошуку ключових слів з використанням технології DKPro Core", *МПК G06F 17/21, G06F 17/27, G06F 17/28. № и 2019 00016*, Черв. 25, 2019.

[136] Р. Розенталь, "Почему Java, а не что-то другое?", [Электронный ресурс]. Режим доступа: http://www.fainaidea.com/jeto-interesno-znat/pochemu-java-a-ne-chto-to-drugoe-130156.html

[137] Почему Java так популярна?, [Электронный ресурс]. Режим доступа: https://vertex-academy.com/tutorials/ru/pochemu-yazyk-java-tak-populyaren/

[138] 12 причин длительного доминирования Java, [Электронный ресурс]. Режим доступа: https://habr.com/post/201612/

[139] Cloud Natural Language, [Online]. Available: https://cloud.google.com/natural-language/

[140] Natural Language Framework, [Online]. Available: https://developer.apple.com/documentation/naturallanguage

[141] T. Casino, "Apple's Natural Language Processing (NLP) API", [Online]. Available: https://willowtreeapps.com/ideas/apples-natural-language-processing-nlp-api

[142] О.В. Яхимович, "Застосування інструментальних засобів пакету DKPRO CORE для визначення ключових слів англомовного тексту", *XLIV науково-технічній конференції підрозділів ВНТУ: факультет комп'ютерних систем та автоматики*, Вінниця, 2015, с. 7.

[143] Л.О. Терещенко, та І.І. Матієнко-Зубенко, "Інформаційні системи і технології обліку:", Навч. Посібник. К.: КНЕУ, 2003.

[144] С.Д. Штовба, О.В. Штовба, О.В. Яхимович та М.В. Петричко, "Вплив синтаксичних зв’язків у реченнях на якість ідентифікації токсичних коментарів в соціальній мережі", *Наукові праці ВНТУ*, № 4, с. 1-8, 2019. DOI: https://doi.org/10.31649/2307-5376-2019-4-35-42.

[145] Darmstadt Knowledge Processing Repository Based on UIMA, [Online]. Available: https://www.ukp.tu-darmstadt.de/fileadmin/user_upload/Group_UKP/publikationen/2007/gldv-uima-ukp.pdf

[146] О.В. Яхимович, "Визначення ключових слів англомовного тексту з використанням технології DKPRO CORE", *Молодь в технічних науках: дослідження, проблеми, перспективи (МТН-2015) : Матеріали міжнародної Інтернет-конференції*, Вінниця, 2015, с. 72-74. ISBN 978-966-924-027-9.

[147] О.В. Бісікало, А.І. Лісовенко, О.В. Яхимович, та С.С. Траченко, "Визначення змістовних ознак тексту на основі аналізу зв’язків між лексичними одиницями", *Вісник НТУ «ХПІ». Серія "Механіко-технологічні системи та комплекси"*, № 21 (1130), с. 83-89, 2015. ISSN 2411-2798.

[148] Address by President of the Russian Federation, [Online]. Available: http://eng.kremlin.ru/transcripts/6402

[149] Address by President of the Russian Federation, [Online]. Available: http://eng.kremlin.ru/news/6889

[150] English grammatical relations, [Online]. Available: http://universaldependencies.org/en/dep/

[151] K. Bougé, "Lists of stop words", [Online]. Available: https://sites.goo-gle.com/site/kevinbouge/stopwords-lists

[152] О.В. Бісікало, О.В. Яхимович, та Я.В. Яхимович, "Розробка методу фільтрації вербального шуму в процесі пошуку ключових слів англомовного

тексту", *Технологический аудит и резервы производства: Информационные технологии*, № 6(44), с. 33–41, 2018. ISSN 2226-3780.

[153] L. Cameron, "Participation, archival activism and learning to learn", [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0305748814001030

[154] D. Heitman, "Workingman's Poet", *Humanities №34*, 2013 p. 28, 2013.

[155] A new pattern for historical geography: working with enthusiast communities and public history, [Online]. Available: http://ac.els-cdn.com/S0305748814001029/1-s2.0-S0305748814001029-main.pdf?_tid=d45ec9e6-ba7b-11e4-b562-00000aab0f01&acdnat=1424600353_4-8bb4ef54fffbc3b800698d175c3c052

[156] O.V. Bisikalo, W. Wójcik, O.V. Yahimovich, and S. Smailova, "Method of determining of keywords in English texts based on the DKPro Core", *Proceedings of SPIE 10031, Photonics Applications in Astronomy, Communications, Industry, and High-Energy Physics Experiments 2016*, 100314T, 2016. DOI:10.1117/12.2249225.

[157] The Movies Dataset, [Online]. Available: https://www.kaggle.com/rounakbanik/the-movies-dataset

ДОДАТКИ

Додаток А

Список опублікованих праць за темою дисертації

[1] O.V. Bisikalo, W. Wójcik, O.V. Yahimovich, and S. Smailova, "Method of determining of keywords in English texts based on the DKPro Core", *Proceedings of SPIE 10031, Photonics Applications in Astronomy, Communications, Industry, and High-Energy Physics Experiments 2016*, 100314T, 2016. DOI:10.1117/12.2249225.

[2] O. Bisikalo, A. Lisovenko, O. Jahumovuch, S. Trachenko, and M. Pradivliannyi, "System of computational linguistic on base of the figurative text comprehension", *Proceedings of the 2016 IEEE 1st International Conference on Data Stream Mining and Processing (DSMP 2016)*, pp. 69-74, 2016. DOI: 10.1109/DSMP.2016.7583510.

[3] О.В. Бісікало, та О.В. Яхимович, "Метод визначення ключових слів англомовного тексту на основі DKPro Core", *Технологический аудит и резервы производства: Информационные технологии*, Том 1, № 2(21), с. 26-30, 2015. ISSN 2226-3780.

[4] О.В. Бісікало, А.І. Лісовенко, О.В. Яхимович, та С.С. Траченко, "Визначення змістовних ознак тексту на основі аналізу зв’язків між лексичними одиницями", *Вісник НТУ «ХПІ». Серія "Механіко-технологічні системи та комплекси"*, № 21 (1130), с. 83-89, 2015. ISSN 2411-2798.

[5] О.В. Бісікало, та О.В. Яхимович, "Знаходження ключових слів англомовного тексту за допомогою інструментальних засобів пакету DKPro Core", *Інформаційні технології та комп’ютерна інженерія*, № 2(34), с. 10-14, 2015. ISSN 1999-9941.

[6] О.В. Бісікало, та О.В. Яхимович, "Автоматизоване визначення лексичних технологій з тезаурусу технічного спрямування",

Оптико-електронні інформаційно-енергетичні технології, № 1 (31), с. 26-38, 2016. ISSN 1681-7893.

[7] О.В. Бісікало, О.В. Яхимович, та Я.В. Яхимович, "Розробка методу фільтрації вербального шуму в процесі пошуку ключових слів англомовного тексту", *Технологический аудит и резервы производства: Информационные технологии*, № 6(44), с. 33–41, 2018. ISSN 2226-3780.

[8] С.Д. Штовба, О.В. Штовба, О.В. Яхимович, та М.В. Петричко, "Вплив синтаксичних зв’язків у реченнях на якість ідентифікації токсичних коментарів в соціальній мережі", *Наукові праці ВНТУ*, № 4, с. 1-8, 2019. DOI: https://doi.org/10.31649/2307-5376-2019-4-35-42.

[9] О.В. Яхимович, "Визначення ключових слів англомовного тексту з використанням технології DKPRO CORE", *Молодь в технічних науках: дослідження, проблеми, перспективи (МТН-2015) : Матеріали міжнародної Інтернет-конференції*, Вінниця, 2015, с. 72-74. ISBN 978-966-924-027-9.

[10] О.В. Бісікало, О.В. Яхимович, А.І. Лісовенко, та Траченко С.С., "Підтримка діалогу з навчальним контентом", *Адаптивні технології управління навчанням: матеріали першої міжнародної конференції*, Одеса, 2015, с. 97-100.

[11] О.В. Бісікало, О.В. Яхимович, А.І. Лісовенко, та Траченко С.С., "Моделювання процесів побудови парадигматичних зв’язків між словоформами на основі вимірювання текстової інформації", *Вимірювання, контроль та діагностика в технічних системах (ВКДТС-2015)*, Вінниця, 2015, с. 119-121.

[12] О.В. Яхимович, "Застосування інструментальних засобів пакету DKPRO CORE для визначення ключових слів англомовного тексту", *XLIV науково-технічній конференції підрозділів ВНТУ: факультет комп'ютерних систем та автоматики*, Вінниця, 2015, с. 7.

[13] О.В. Яхимович, "Визначення ключових слів з тексту повідомлень мікроблогів", *XLV науково-технічній конференції підрозділів ВНТУ: факультет комп'ютерних систем та автоматики*, Вінниця, 2016, с. 1119-1121.

[14] О.В. Яхимович, "Колізія при знаходженні ключових слів", *XLVI науково-технічній конференції підрозділів ВНТУ: факультет комп'ютерних систем та автоматики*, Вінниця, 2017, с. 1312-1314.

[15] О.В. Яхимович, "Зменшення вербального шуму при визначенні ключових слів", *XLVII науково-технічній конференції підрозділів ВНТУ: факультет комп'ютерних систем та автоматики*, Вінниця, 2018, с. 1463-1465.

[16] О.В. Яхимович, "Формалізація задачі визначення ключових слів тексту", *XLVIII науково-технічній конференції підрозділів ВНТУ: факультет комп'ютерних систем та автоматики*, Вінниця, 2019, с. 1136-1138.

[17] О.В. Бісікало, Р.Н. Квєтний, С.Г. Кривогубченко, Л.Є Азарова, та О.В. Яхимович, "Синтез інтегрованої бази знань природно-мовного контенту", ВНТУ, Вінниця, Д/б № 0114U003462, 2015.

[18] О.В.Бісікало, Р.Н. Квєтний, С.Г. Кривогубченко, А.І. Лісовенко, та О.В. Яхимович, "Розв’язання семантико-залежних задач обробки природно-мовних об’єктів на основі бази знань", ВНТУ, Вінниця, Д/б № 0114U003462, 2016.

[19] О. В. Бісікало, А. І. Лісовенко, О. В. Яхимович, та В. В. Шолота, "Спосіб автоматичного пошуку ключових слів з використанням технології DKPro Core", *МПК G06F 17/21, G06F 17/27, G06F 17/28. № и 2019 00016*, Черв. 25, 2019.

[20] O. Bisikalo, and A. Yahimovich. *Keyword search based on lexical relationships in the text*. Beau Bassin-Rose Hill, Mauritius: Lap Lambert Academic Publishing, 2019. ISBN 978-620-0-00314-0.

[21] O. Bisikalo, and A. Yahimovich. *Lexical relationships-based keywords selection in english texts*. Vinnytsia, Ukraine: VNTU, 2020.

Додаток Б
Практичні поради при складанні списку ключових слів

Не бійтеся використовувати занадто велику кількість ключових слів. 3-4 слова - цього буде явно недостатньо для того, щоб відобразити характер окремої статті. Краще використовувати 10-15 слів залежно від обсягу тексту. Кожне додаткове ключове слово – це можливість залучити читачів до статті.

Використовуйте базові терміни разом з більш складними. Разом з ключовим словом бухгалтерський облік основних засобів краще додати також і базові слова, повний набір буде таким: бухгалтерський облік основних засобів, бухгалтерський облік, основні засоби.

Не бійтеся використовувати повтори і синоніми. Знову ж таки, різні читачі можуть шукати одне і те ж, але з різних пошукових запитів. Максимальна кількість синонімів дозволить більшій кількості читачів знайти нашу статтю. Але в той же час не забувайте про наступний принцип.

Не використовуйте занадто складні слова. Словосполучення, в яких використовується більше трьох слів найчастіше можна розбити на кілька ключових слів. Наприклад, обробка та аналіз даних, взаємозв'язок (кореляція) ризиків - краще вказати такі ключові слова: обробка даних, аналіз даних, взаємозв'язок ризиків, кореляція ризиків.

По можливості не використовуйте слова в лапках. Навіть якщо використаний в статті термін може виявитися спірним, абстрактним або просто несерйозною, краще вказати його без лапок. Це також відноситься до назв організацій. Однією з причин буде хоча б те, що використовувані різними редакторами лапки можуть бути різними. У підсумку однакові ключові слова виявляться відірваними одне від іншого. Наступні слова матимуть малу ефективність.

Програма «Чиста вода», ВАТ «Укрпошта», краще використовувати їх у такому варіанті: програма Чиста вода, Укрпошта.

Не використовуйте слова з комами. Це пов'язано з тим, що в бібліотечних та пошукових системах ключові слова розділяються саме комами. Якщо використовувати термін слово з комою, то, швидше за все, осмислений термін буде розбитий на кілька безглуздих і неефективних в плані пошуку частин.

Замість фактори, що визначають якість, краще використовувати слова чинники якості, визначення якості.

Використовуйте слова в основній формі. Основною формою слова є однина, називний відмінок. По можливості слід застосовувати слова саме в цій формі: мета інвестицій, інвестиційно-привабліва галузь.

У той же час деякі слова в основній формі якраз не використовуються, в цьому випадку вони повинні вживатися в формі, що найбільш часто зустрічається. Наприклад: інвестиції, види інвестицій.

Кожне ключове слово – це самостійний елемент. Ключові слова повинні мати власне значення. Невірним варіантом буде людський капітал, його оцінка, краще використовувати набір: людський капітал, оцінка людського капіталу.

Також не забувайте про необхідність правильно оформити список ключових слів, інакше зусилля по підбору списку можуть бути зведені нанівець через помилки в обробці даних.

Не вказуйте перше слово у списку ключових слів з великої літери. При обробці таке слово може бути враховано як самостійне. Наприклад, в Науковій електронній бібліотеці ключові слова «інвестиції» і «Інвестиції» будуть різними.

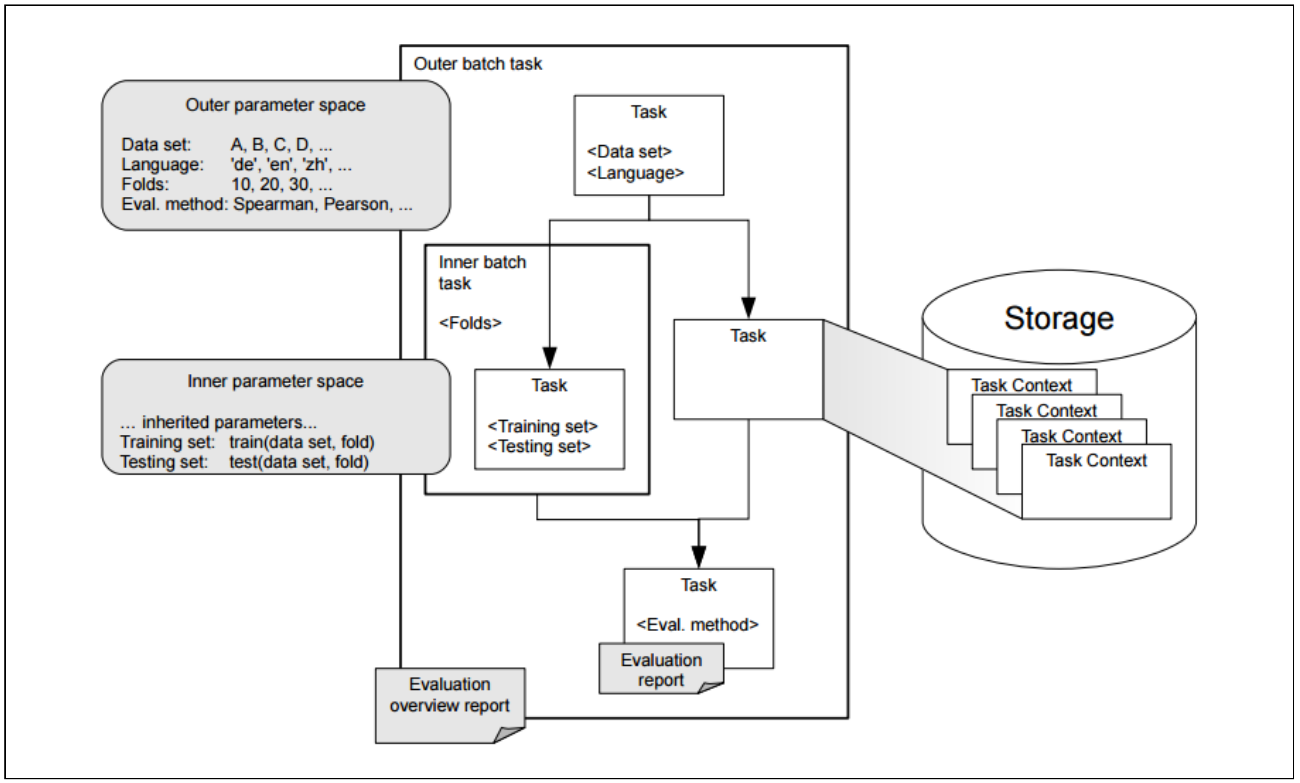
Розділяйте слова комами. Більшість систем обробки текстів сприймають саме кому як роздільник між ключовими словами.

Використання ком при оформленні списку допоможе заощадити час при розмітці статті перед внесенням до бази даних.

Не ставте крапку в кінці списку ключових слів. Крапка випадково може бути додана до останнього ключового слова, через що воно втратить пошукову ефективність [29].

Додаток В

Схематичне зображення робочого процесу в DKPro Lab



Додаток Г

Впровадження результатів роботи

ЗАТВЕРДЖУЮ

Директор
НВП «СПІЛЬНА СПРАВА», ТОВ
_____ А. О. Чикалова
«10» серпня 2020 р.

АКТ

впровадження результатів кандидатської дисертаційної роботи
Яхимовича Олександра Вікторовича

Комісія у складі керівника відділу обробки даних к.т.н. Білоконя Сергія Анатолійовича та провідного спеціаліста к.т.н. Сторчака Володимира Григоровича цим актом свідчить про впровадження на підприємстві результатів, отриманих у кандидатській дисертаційній роботі Яхимовича О.В., а саме, інформаційної технології пошуку ключових слів на основі аналізу зв'язків між лексичними одиницями тексту.

Під час апробації у 2019 р. науково-технічної продукції на НВП «СПІЛЬНА СПРАВА», ТОВ було проведено експериментальні дослідження:

1. Запропонована інформаційна технологія пошуку ключових слів на основі аналізу зв'язків між лексичними одиницями тексту використовує інструментальні засоби лінгвістичного пакет DKPro Core для аналізу вхідної текстової інформації та визначення зв'язків між лексичними одиницями.

2. Розроблене програмне забезпечення можна запускати в будь-якому середовищі за допомогою JVM (Java Virtual Machine), тому було підтверджено зручну перевагу розробки в тому, що вона працює на різних операційних системах.

3. Проведено експеримент з використанням розробленого програмного забезпечення для англомовного тексту «A new pattern for historical geography: working with enthusiast communities and public history» ([Electronic resource]. – Available at: \www/URL: http://ac.els-cdn.com/S0305748814001029/1-s2.0-S0305748814001029-main.pdf?_tid=d45ec9e6-ba7b-11e4-b562-

00000aab0f01&acdnat=1424600353_4-8bb4ef54fffbc3b800698d175c3c052). Зв'язки між лексичними одиницями речень вхідного тексту побудовано на основі стенфордської класифікації зв'язків.

4. Еталонні ключові слова для визначення релевантність результатів експерименту задані автором вхідного тексту.

5. Зменшення кількості вербального шуму досягнуто за допомогою підходів: заміна займенників на відповідні до них іменники; вилучення словосполучень із типами зв'язків, які не несуть суттєвого смислового навантаження; вилучення слів, які відносяться до неінформативних частин мови; вилучення слів, які відносяться до списку стоп-слів. Використовуючи дані підходи вдалося зменшити кількість шуму при визначення ключових слів для англомовних текстів.

6. Розроблене програмне забезпечення для реалізації запропонованої інформаційної технології підвищує релевантність отриманих ключових слів порівняно з аналогами. Зокрема, за повнотою на 13% на основі метрики Жаккара та за точністю на 14,3% на основі абсолютної метрики.

Розроблена інформаційна технологія пошуку ключових слів на основі аналізу зв'язків між лексичними одиницями тексту заплановано застосовувати у проекті «Розробка системи управління контекстною рекламою» для визначення ключових слів з опису продукту і на їх основі створення відповідної контекстної реклами. Практична цінність отриманих наукових результатів не викликає сумнівів.

Члени комісії:

керівник відділу обробки даних, к.т.н. С.А. Білоконь

провідний спеціаліст, к.т.н. В.Г. Сторчак

Додаток Д

Граф зв'язків між словами