МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний аерокосмічний університет ім. М. Є. Жуковського
«Харківський авіаційний інститут»

Факультет програмної інженерії та бізнесу

Кафедра інженерії програмного забезпечення

Пояснювальна записка до дипломного проєкту

магістра
(освітній ступінь)

на тему «Аналіз ефективності використання методів штучного інтелекту для задач пошуку інформації у інтернет-середовищі»

ХАІ.603.667п2.121.168218.20О

Виконав: студент 6 курсу групи № 667п2
Спеціальність 121 – Інженерія програмного забезпечення
(код та найменування)
Освітня програма Хмарні обчислення та Інтернет речей
(найменування)
Шевченко А.В.
(прізвище й ініціали студента)
Керівник Пудовкіна Л.Ф.
(прізвище та ініціали)
Рецензент Мартовицький В.О.
(прізвище та ініціали)

Харків – 2020

Міністерство світи і науки України
Національний аерокосмічний університет ім. М. Є. Жуковського
«Харківський авіаційний інститут»

Факультет програмної інженерії та бізнесу
(повне найменування)
Кафедра інженерії програмного забезпечення
(повне найменування)
Рівень вищої освіти другий (магістерський)

Спеціальність 121 – інженерія програмного забезпечення
(код та найменування)
Освітня програма хмарні обчислення та Інтернет речей
(найменування)

ЗАТВЕРДЖУЮ
Завідувач кафедри
_____ _____І. Б. Туркін_____
(підпис) (ініціали та прізвище)
“_____” _____ 2020 року

З А В Д А Н Н Я
НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ) СТУДЕНТУ

Шевченко Андрію Володимировичу
(прізвище, ім’я, по батькові)
1. Тема дипломного проекту *Аналіз ефективності використання методів штучного інтелекту для задач пошуку інформації у інтернет-середовищі*
керівник дипломного проекту *Пудовкіна Лариса Федорівна, к.т.н., доцент*
(прізвище, ім’я, по батькові, науковий ступінь, вчене звання)
затверджені наказом вищого навчального закладу від “__” ___2020 року № ___
2. Термін подання студентом роботи _____
3. Вихідні дані до роботи: *методи штучного інтелекту*
4. Зміст пояснювальної записки (перелік питань, які потрібно розробити)
провести огляд та аналіз процесу інформаційного пошуку і розробити його структурну модель; провести аналіз застосовності математичних методів штучних інтелектуальних агентів для реалізації інформаційного пошуку; розробити прототип пошукової системи; провести тестування розробленого прототипу

5. Перелік графічного матеріалу
РПЗ – стор. 81, рисунків – 48 шт., таблиць – 2 шт., презентація – 15 слайдів

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1	Пудовкіна Л.Ф., доц. каф. 603		
2	Пудовкіна Л.Ф., доц. каф. 603		
3	Пудовкіна Л.Ф., доц. каф. 603		

8. Нормоконтроль _____ __В.А. Постернакова__ «____» _____2020 р.
(підпис) (ініціали та прізвище)

7. Дата видачі завдання _____

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проекту	Строк виконання етапів проекту	Примітка
1	Отримання і затвердження теми диплому	03.09.2019	
2	Аналіз предметної області	04.09.2019	
3	Постановка задачі	20.11.2019	
4	Проведення теоретичних досліджень	22.11.2019	
5	Розробка прототипу ПЗ	02.09.2020	
6	Підготовка пояснювальної записки	22.10.2020	
7	Оформлення пояснювальної записки до дипломного проекту	10.11.2020	
8	Передзахист дипломного проекту	27.11.2020	
9	Захист дипломного проекту	24.12.2020	

Студент _____ Шевченко А.В.
____
(підпис) (прізвище та ініціали)

Керівник роботи _____ Пудовкіна Л.Ф.
(підпис) (прізвище та ініціали)

РЕФЕРАТ

Пояснювальна записка до дипломного проєкту містить 81 стор., 48 рис., 26 джерел.

Об'єкт дослідження - процес пошуку інформації у інтернет-середовищі.

Предмет дослідження - математичні методи штучних інтелектуальних агентів в задачах пошуку інформації у інтернет-середовищі.

Метою роботи є підвищення релевантності результатів автоматичного інформаційного пошуку шляхом застосування мультиагентного підходу.

Для досягнення поставленої мети необхідно вирішити наступні завдання: провести огляд та аналіз процесу інформаційного пошуку і розробити його структурну модель; провести аналіз застосовності математичних методів штучних інтелектуальних агентів для реалізації інформаційного пошуку; розробити прототип пошукової системи; провести тестування розробленого прототипу.

Наукова новизна. Удосконалено метод автоматичного інформаційного пошуку, який на відміну від існуючих використовує алгоритми нечіткого логічного висновку для ранжування результатів пошуку, що дає змогу підвищіти їх релевантність.

Практична значимість отриманих результатів. В результаті проведених досліджень запропонована структура системи реалізована у вигляді мультиагентної системи багатомовного пошуку на базі програмних платформ JADE.

АПРОКСИМАЦІЯ, АСОЦІАТИВНІСТЬ, БАГАТОМОВНИЙ ІНФОРМАЦІЙНИЙ ПОШУК, ШТУЧНІ АГЕНТИ

ABSTRACT

Explanatory note to the master's thesis 81 pp., 48 fig., 26 sources.

The object of study - the process of finding information in the Internet environment.

The subject of research - mathematical methods of artificial intelligent agents in the search for information in the Internet environment.

The aim of the work is to increase the relevance of the results of automatic information retrieval by applying a multi-agent approach.

To achieve this goal it is necessary to solve the following tasks: to review and analyze the process of information retrieval and develop its structural model; to analyze the applicability of mathematical methods of artificial intelligent agents for the implementation of information retrieval; develop a prototype search engine; to test the developed prototype.

Scientific novelty. The method of automatic information retrieval has been improved, which, unlike the existing ones, uses fuzzy inference algorithms to rank search results, which allows to increase their relevance.

The practical significance of the results obtained. As a result of the research, the proposed system structure is implemented in the form of a multi-agent multilingual search system based on JADE software platforms.

.

APPROXIMATION, ASSOCIATIVENESS, MULTILINGUAL INFORMATION SEARCH, ARTIFICIAL AGENTS

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

ДТП – дорожньо-транспортне порушення.

ПЗ – програмне забезпечення.

ПП – програмний продукт.

ВСТУП

Обсяг спеціальної інформації, що відноситься до різних галузей науки й техніки, в Інтернеті постійно росте. Використання цієї інформації неможливо без ефективного інструмента пошуку у всім доступному обсязі даних. Такий інструмент повинен шукати дані, що цікавлять користувача-фахівця не тільки в спеціально структурованих але й у неструктурованих документах на всіх відомих користувачеві мовах, тобто здійснювати багатомовний інформаційний пошук, що орієнтований на представників наукового співтовариства, що знають іноземні мови, а також іноземних студентів, що навчаються у різних напрямках. Основним критерієм якості для багатомовних інформаційних пошукових систем у силу особливості їх застосування для пошуку наукової або навчальної інформації кількома мовами є не стільки швидкодія, як для звичайних пошукових систем, скільки висока релевантність перших отриманих результатів. Під релевантністю при цьому розуміється семантична відповідність пошукового запиту та знайденого документа.

Провідні універсальні пошукові системи, такі як Google, Bing і ін. забезпечують високу швидкодію та повноту пошуку мовою запиту, але для одержання результатів на заданих мовах вимагають зміни регіону пошуку й, отже, введення окремого запиту для пошуку на кожній мові, а також не допускають об’єднання і якісного ранжування отриманих результатів.

Нечисленні існуючі спеціалізовані системи багатомовного інформаційного пошуку мають істотні недоліки. Так, наприклад система, пропонована Chandra Mohan, Sadanandam, Raju Korra (англійський – французький – німецький – гінді, 2013), має значний час пошуку та не допускає ранжування результатів, а в системі, запропонованої Leyla Zhuhadar, Olfa Nasraoui, Robert Wyatt, Elizabeth Romero (англійський – іспанський, 2010) використовуються складні методи ранжування знайдених документів, застосування яких додатково збільшує і так істотний час відгуку системи.

Також не існує систем багатомовних пошукових систем, орієнтованих на арабську і російську мови. Отже, проблема створення методики багатомовного інформаційного пошуку (БІП) з ранжуванням отриманих результатів по ступеню релевантності залишається актуальною.

Разом із цим практика останніх років показала, що при створенні розподілених інтелектуальних систем доцільно використовувати мультиагентну технологію. Це пов’язане з тим, що мультиагентні системи мають високу гнучкість, гарну масштабованість і підвищеною надійністю. Зазначені властивості виявилися вирішальними при виборі мультиагентної реалізації системи БІП. При цьому ефективна реалізація мультиагентної системи допускає обґрунтований вибір її архітектури, а тому виконання відповідних досліджень.

Об'єкт дослідження - процес пошуку інформації у інтернет-середовищі.

Предмет дослідження - математичні методи штучних інтелектуальних агентів в задачах пошуку інформації у інтернет-середовищі.

Метою роботи є підвищення релевантності результатів автоматичного інформаційного пошуку шляхом застосування мультиагентного підходу.

Для досягнення поставленої мети необхідно вирішити наступні завдання: провести огляд та аналіз процесу інформаційного пошуку і розробити його структурну модель; провести аналіз застосовності математичних методів штучних інтелектуальних агентів для реалізації інформаційного пошуку; розробити прототип пошукової системи; провести тестування розробленого прототипу.

Наукова новизна. Удосконалено метод автоматичного інформаційного пошуку, який на відміну від існуючих використовує алгоритми нечіткого логічного висновку для ранжування результатів пошуку, що дає змогу підвищіти їх релевантність.

Практична значимість отриманих результатів. В результаті проведених досліджень запропонована структура системи реалізована у вигляді

мультиагентної системи багатомовного пошуку на базі програмних платформ JADE.

.

1 АНАЛІЗ ПРОЦЕСУ ІНФОРМАЦІЙНОГО ПОШУКУ І ПОСТАНОВКА ЗАВДАННЯ

1.1 Аналіз процесу багатомовного інформаційного пошуку

У результаті розвитку Інтернету й інших глобальних мереж люди, яким необхідна інформація, часто перевантажені значним обсягом певної доступної інформації, а пошук корисної інформації вимагає значних зусиль. Системи інформаційного пошуку (ІП) розроблені, щоб допомогти людині витягти корисну або его інформацію, що цікавить, з різних зборів документів. Системи ІП і системи пошуку документів не є останніми інноваціями. Вони існують із часів перших бібліотек у формі бібліотечних каталогів. З тих пір системи інформаційного пошуку швидко змінилися через ріст обсягів інформації в текстовому виді, доступної в цифровому й паперовому виді. Це значне збільшення доступної інформації призвело до необхідності розробки автоматизованої системи інформаційного пошуку.

Автоматизовані системи ІП були розроблені, щоб допомогти організовувати величезні обсяги наукової літератури, які розвивалися з 1940 р. Багато університетів, корпорації й публічні бібліотеки зараз використовують системи ІП для надання доступу до книг, журналів та інших документів. Комерційні системи ІП пропонують бази даних, що містять мільйони документів у різноманітті предметних областей. Словники й енциклопедії баз даних зараз широко доступні на ПК.

ІП виявився корисним у таких розрізнених областях, як автоматизація бізнесу й розробка програмного забезпечення. Будь-яка дисципліна, яка опирається на документи, у своїй роботі може потенційно використовувати переваги ІП. Система ІП зіставляє користувацькі запити – формальні вираження інформаційних потреб – і документи, що зберігаються в базі даних. Документ є об’єктом даних, зазвичай текстових, хоча він може також містити інші типи

даних, такі як фотографії, графи і т.д. Часто самі документи не зберігаються безпосередньо в системі ІП, але представлені в ній як ідентифікатори.

Термін «ІП» – пошук неструктурованих записів, таких як записи, що зазвичай являють собою текст у вільній формі природньою мовою. Зрозуміло, що інші типи даних теж можуть бути не структурованими, наприклад фотознімки, аудіо, відео й т.п. Проте дослідження в області ІП були присвячені зазвичай пошуку тексту природньою мовою, акцент був зроблений на важливість і величезні обсяги текстових даних, що перебувають у мережі Інтернет і приватних архівах.

Деякі моменти з термінології необхідно уточнити. Записи, у яких здійснюється ІП, часто називають «документами». ІП документів здійснюється в організованих (відносно статично) сховищах, що часто називаються «колекцією» (Слово «архів» теж використовується. Також використовується слово «корпус». Термін «цифрова бібліотека» стає найпоширенішим. Але загальний термін «колекція» все ще часто використовується в науковій літературі). Однак слід розуміти, що термін ІП не ставиться тільки до статичних колекцій. Колекція може бути потоком повідомлень, наприклад повідомлень електронної пошти, факсів, розсилань новин, що протікають через Інтернет або яку-небудь приватну мережу [1], [2].

Система ІП документів зазвичай складається із трьох основних підсистем: відображення документа, відображення користувацьких вимог (запитів) і алгоритмів, використовуваних для визначення відповідності користувацьких вимог (запитів) відображенням документів. Найпростіша архітектура показана на рис. 1.1.

Колекція документів складається з великої кількості документів, що містять інформацію про різні предметні області та теми, що представляють інтерес. Зміст документа перетвориться в відображення документа (вручну або автоматично). Відображення документів зроблено таким чином, щоб зіставлення їх з пошуковими запитами було легким. Інше припущення про відображення документів про те, що таке відображення повинно коректно

відображати задум автора. Головним завданням у відображенні є те, як вибрати відповідні терміни указника. Звичайне відображення здійснюється шляхом витягу ключових слів, які вважаються ідентифікаторами контенту, і упорядкуванням їх у заданому форматі.

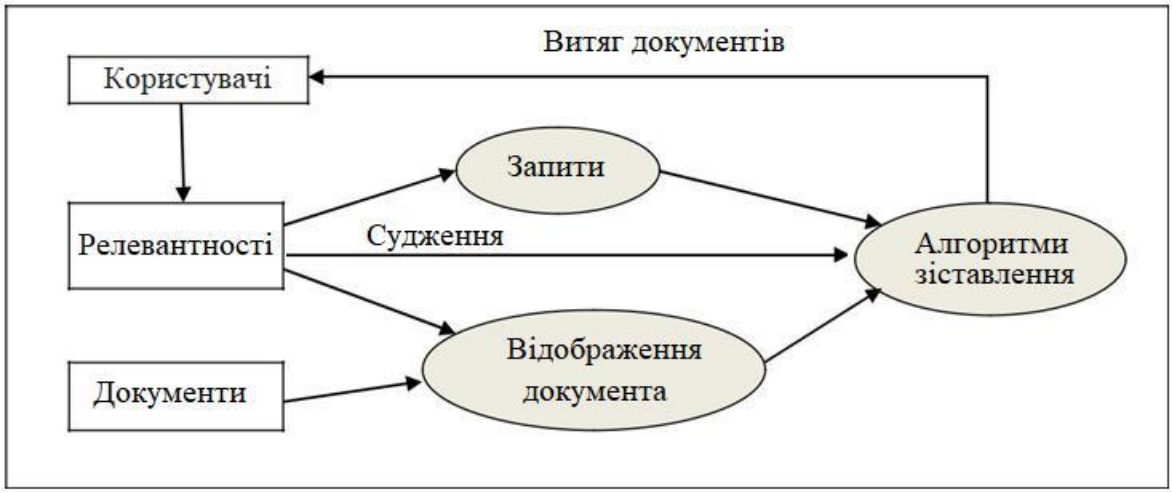


Рисунок 1.1 – Базова архітектура інформаційно-пошукових систем [1]

Запити перетворять інформаційну потребу користувача у форму, яка коректно відображає інформаційну вимогу користувача і підходить для здійснення пошуку на відповідність. Форматування запиту залежить від основної моделі ІП, яка використовується в системі.

Користувач привласнює рейтинг отриманим документам, як релевантним або нерелевантним відповідно до його інформаційної потреби.

Основна проблема, що стоїть перед будь-якою системою ІП, полягає в тому, як знаходити тільки документи, релевантні інформаційним потребам користувача, при цьому не витягаючи нерелевантні.

Зворотний зв’язок по оцінці користувачем релевантності зазвичай використовується системою (див. рис. 1.1) для поліпшення описів документів або запитів з очікуванням того, що загальна продуктивність системи покращиться після введення такого зворотного зв’язку [1], [3].

Алгоритми зіставлення використовують відображення документа й запиту для пошуку документів, відібраних системою як релевантні. Однак документи,

які система повертає, необов’язково повинні бути релевантними з користувацької точки зору. Двома основними факторами, які впливають на невідповідність між набором документів, відібраних системою, і тими, передбачуваними користувачем як релевантні конкретному запиту, є неоднозначність природньої мови й можливий обмежений набір знань користувачів в області запиту. Проблема двозначності (неоднозначності) природньої мови пояснюється тим, що концепція може бути виражена декількома способами. Наприклад, розглянемо слово windows (вікна). Користувач використовує це слово для пошуку документів в операційній системі Windows або документів, що пояснюють, як класифікувати різні типи архітектури, розглядаючи форми вікон [3]. Формулювання методів для подолання проблеми двозначності природніх мов є головною метою досліджень інформаційного пошуку.

Крім традиційного інформаційного пошуку (ІП) або пошуку на одній мові, де документи й запити написані на тій же мові, в області дослідження ІП розглядаються ще два типи ІП: кросмовний ІП у багатомовному середовищі й багатомовний ІП (БІП).

1.2 Аналіз методів багатомовного інформаційного пошуку

В той же час як ІП був активною областю досліджень протягом багатьох десятиліть, більша частина його історії дуже сильно схилялася у бік англійської мови, як мови, обраної для дослідницьких цілей і оцінки. Якими б вони не були протягом цих років, багато наукових робіт, що були присвячені майже винятковій роботі з англійською мовою, як мовою ІП, втратили свою актуальність. Інтернет вже більше не є одномовним, і неангломовний контент значно збільшується. З 2005 року дві третини всіх користувачів мережі Інтернет не виявилися носіями англійської мови [4].

В дійсності тільки одна п’ята частина користувачів мережі Інтернет є носієм англійської мови. Природа мережі Інтернет не знає мовних меж. Люди з різних країн, що говорять на різних мовах, використовують мережу Інтернет. Це однозначно мотивує розвиток і поліпшення багатомовних методів ІП. Люди часто можуть бути зацікавлені в релевантній інформації на різних мовах, яка отримана за допомогою одного процесу пошуку з використанням багатомовних методів. Це також дозволяє користувачам виражати потреби в інформації на своїх рідних мовах, тоді як результати пошуку можуть бути на інших [5].

Багатомовний ІП вимагає гарного розуміння питань, що стосуються ІП на одній мові. Для малих європейських мов, таких як голландська і фінська, витрати на розробку й підтримку мовної інфраструктури відносно високі. Але положення мов, для яких розроблена невелика кількість обчислювальних інструментів, буде ускладнюватися в зростаючому глобальному суспільстві через культурні й економічні причини [4].

Кросмовний ІП є завданням пошуку документів, релевантних запиту, на деякій мові (мові запиту) у наборі документів на деякій іншій мові (мові набору). Кросмовний ІП є підобластью ІП інформації, що стосується пошуку, записаною мовою, відмінною від мови користувацького запиту. Наприклад, користувач може створити свій запит англійською мовою, а одержати релевантні документи на французькій. Для цього більшість систем кросмовного ІП використовують технології перекладу [5].

Система багатомовного ІП (БІП) допомагає користувачам скласти запит на одній мові та здійснювати пошук документів на більш ніж одній мові [6]. Система БІП застосовується там, де набір даних складається з документів на різних мовах і користувачі системи ІП можуть читати на деяких мовах, на яких написані документи. У більшості випадків люди в дійсності мають базові навички читання й розуміння на деякій іншій мові, на відміну від їхньої рідної мови, на якій вони зазвичай пишуть пошукові запити. Далі, якщо користувачі не розуміють мову знайденого документа, можуть бути використані системи машинного перекладу (МП) для одержання тексту рідною мовою користувача.

Багатомовні системи, про яких є відомості в періодиці. В [8] використовується метод перекладу запиту для пошуку документів кількома мовами з технікою розширення для перекладу по фразах. Вони також використовували розвідувачі, що застосовують модель векторного простору для зіставлення термінів запиту із проіндексованими документами, де використовувалося рівняння скорінгу. Скорінговий алгоритм заснований на документах, що представлені у вигляді векторів. Кожне відображення вектора-терміну пов'язане з кожним полем документа, для якого користувач робить запит на англійській або іспанській мові.

Система MULINEX [9] – повністю реалізований багатомовний розвідувач і навігаційна система для Всесвітньої павутини. Система дозволяє користувачам шукати й переглядати мультимовні збори документів з використанням їх рідної мови, щоб формулювати, розширювати й уточнювати запити, переглядати набір результатів і читати знайдені документи. Цей мультимовний функціонал отриманий завдяки використанню перекладу запитів зі словником, категоризації документів кількома мовами та автоматичному перекладу анотацій і документів. Система встановлена в складі онлайн служб двох компаній –
Інтернет контент і сервіс провайдерах. Спочатку запит перекладається з рідної мови користувача на мову документів, що проглядаються, за допомогою двомовного словника. При цьому первісний запит розділяється на слова з використанням морфологічного аналізатора. Потім кожне слово автоматично перекладається на цільову мову з використанням словника, який машина здатна прочитати. У системі прийнятий словниково-орієнтований підхід, що поєднує в собі переклад фраз, аналіз спільної появи перед перекладом і після розширення запиту. Підхід був оцінений доменними експертами й результати представлені таким чином, що досягається переклад речень. Досягнутий рівень в 74.6% поліпшення точності у порівнянні з простим перекладом кожного слова.

В деяких системах мультимовний словник заснований на перекладі кожного слова запиту, а збори на англійській, французькій, німецькій та гінді обробляються системами ІП і БІП. Для перекладу запитів був використаний

перекладач Google. При цьому англійська мова розглядалася як мова джерела, а французька, німецька і гінді – як цільові мови.

У [10] авторами представлений метод на основі GHSOM для виявлення відповідностей між різними мовами й застосуванням цього методу для завдання БІП. Експерименти показали, що цей метод надав багатообіцяючий підхід для рішення завдання БІП.

Автором [11] запропонована нова Веб-пошукова система, яка автоматично класифікує зібрані документи, здійснює пошук інформації кількома мовами (наприклад, на японській або англійській). Це досягається обробкою за допомогою формалізованого опису предметної області – онтології. Вони створили багатомовну онтологію для застосування як вказівник словника. Для конкретних предметних областей онтологія керує відносинами ключових слів і їх ваг, згідно з якими класифікуються документи.

Перераховані вище системи дозволяють здійснювати автоматичний переклад тексту на іншу мову й виконати пошук документів на цих мовах. Порівняльні дані по перерахованих вище системах, представлено в таблиці 1. Більшість із них використовує переклад запитів методом «слово-слово». Це приводить до великої кількості помилок перекладу, у свою чергу нерелевантних документів, що приводять до включення в результати. Також практично у всіх системах відсутнє ранжування знайдених документів. Єдина система, що використовує ранжування, застосовує для виконання класифікацію на базі онтології.

Крім того всі ці системи все ще мають високий рівень помилок, які, наприклад, виникають внаслідок неоднозначності термінів і складності граматики. З неоднозначністю термінів стикаються й одномовні системи ІП, але для систем БІП помилки неоднозначності термінів на стадії перекладу потенційно [5] можуть збільшитися. Ці проблеми, можливо, не будуть вирішені в найближчому майбутньому. Це мотивує розробку багатомовних методів ІП, які не залежать від систем машинного перекладу або, як мінімум, здатні компенсувати помилки таких систем.

Таблиця 1.1 – Аналіз існуючих систем БІП

Автори	Мови	Особливості реалізації	Недоліки
Jialun Qin, Yilu Zhou, Michael Chau, Hsinch un Chen, (2016)	Англійсько-китайська	Переклад запиту «слово – слово».	Помилки перекладу запиту. Додаткові витрати часу на обробку нерелевантних посилань. Відсутність ранжування документів.
Leyla Zhuhadar, Olfa Nasraoui, Robert Wyatt, Elizabeth Romero (2015)	Англійсько-іспанська	Переклад запиту по словнику з обмеженим набором наукових тем. Складна оцінка релевантності по моделі векторного простору.	Велики часові витрати на підрахунок оцінки релевантності та видалення нерелевантних документів.
Chandra Mohan, Sadanandam, Raju Korra (2013)	Англійсько-французько-німецько-гінді	Переклад запиту «слово – слово». Збір текстів на чотирьох мовах.	Помилки перекладу запиту. Відсутність ранжування документів.

Вузьким місцем у розвитку багатомовних підходів в ІП є мовні ресурси, які є посередниками між мовами. Прикладами таких ресурсів, які часто використовуються в справжніх багатомовних системах ІП, є білінгвістичні словники, такі як Інтерлінгва Wordnet і Eurowordnet 2. Подібні ресурси зазвичай написані окремими авторами й покривають тільки обмежений набір тем [11].

Проблема БІП, по суті, полягає в машинному перекладі тексту дуже маленького обсягу (запиту). Є такі підходи до цієї проблеми. Перший – переклад зі словником з використанням багатомовних словників, які здатна прочитати машина, а другий – автоматичний витяг можливих еквівалентних

перекладів за допомогою паралельного статистичного аналізу або зіставлення зборів документів. Існує серйозне питання для систем БІП: яким чином користувачі зможуть оцінити релевантність знайдених документів, представлених кількома мовами, і яким чином буде здійснюватися вибір найбільш релевантних документів для перекладу машиною або людиною [12], [11]. Більшість систем БІП використовують деякий тип перекладу. У той час як існують методи без перекладу, такі як методи, засновані на онотологічному відображенні документів і запитів. Багатомовна онтологія відображення документів/запитів використана в [12], [7]. Інтеграція перекладу запиту й документа з одномовним ІП для поліпшення точності пошуку представлена в [11], [7] і виконує кластеризацію для підвищення ефективності перегляду веб-сторінок.

В кросмовному БІП потреба в інформації та відповідний запит користувача можуть бути сформульовані на мовах, відмінних від тих, на яких написані документи. Релевантність є принципово незалежною від мови характеристикою документа. Більшість підходів БІП або безпосередньо засновані на одномовному ІП, або використовують як мінімум одну стандартну модель ІП. БІП можна розглядати як проблему об’єднання результатів роботи систем ІП на різних мовах. При цьому необхідна первинна обробка запитів мовою, але в цілому БІП заснований на тих же індексних структурах і покладається на подобу документів і моделі пошуку, що відомі для одномовного ІП [5], [7].

Система ІП працює в такий спосіб: користувач пише запит на деякій мові, система шукає документи, пов’язані із запитом і надає результати на тій ж мові, на якій було сформульовано запит.

Індексування документів і запитів є дуже важливими завданнями в будь-якій системі ІП. Однак якісне індексування документів і запитів також вважається найбільш складним завданням.

Завдання індексування вважається складною в реалізації, тому що неоднозначність природних мов представляє невизначеність у процесі аналізу

текстів для індексування документів і запитів. Салтон [11] вважає, що процес індексування не потрібен, якщо набори документів невеликі. У невеликих наборах метод повнотекстового сканування буде більш ефективним при пошуку документів у колекції, ніж використання функції співставлення для документа й індексу запитів.

Сьогоднішні бази даних документів великі через обсяги інформації, що доступна в цифровій формі, і цей обсяг буде збільшуватися із часом. Методи повнотекстового сканування непрактичні для таких баз даних в умові можливостей існуючих комп'ютерних технологій. Інакше кажучи, індексування документів повинне проводитися незалежно від проблеми й неоднозначності текстового аналізу. Як наслідок, зменшення неоднозначності стає частиною проблемної області завдання індексування документів і запитів у системах ІП.

1.3 Аналіз методів індексування документів і запитів

Три головні фактори властиві проблемі неоднозначності при індексуванні документів і запитів. По-перше, є проблема, що виникла внаслідок різноманітності способів, якими може бути виражена концепція [11]. Одне слово може мати кілька тлумачень у різному контексті. Частково це питання мови. Друга проблема може виникнути через специфікацію запиту. Іноді користувач не надає достатніх відомостей або специфікацій у запиті. Це приводить до невизначеного запиту. Запит по специфікації може також виникнути, коли запит сам по собі неповний. Наприклад, користувач може побажати інформацію не тільки про спосіб виробництва, але й про конструктивні аспекти товару. Проте, малоймовірно, що система буде знаходити документи, що містять особливості конструкції електронних товарів, тому що дизайн і виробництво не можуть бути узагальнені у «способі виробництва».

Різниця неточних і незавершених запитів полягає в тому, що у випадку неточного запиту користувач може не усвідомлювати наявність неточності у запиті, у той час як у випадку незавершеного запиту користувач не зміг достатньо конкретизувати запит.

Неточність або незавершеність специфікації менш очевидні, ніж проблема мінливості. Проте, це також важливо. І запит по специфікації, і проблема мінливості виникають із невідання користувача.

Третя проблема полягає в зменшенні дескриптора документа. Цю проблему ніколи не вдасться повністю уникнути – автор документа завжди залишає багато недоговореного по темі, що не є завжди шкідливо. Може здаватись, що формування компактних описів змісту документа збільшують невизначеність, але це може збільшити й ефективність відповідності, й ефективність класифікації документів [3].

ІП, таким чином, може накласти суперечливі вимоги до дескрипторів контексту: так вони повинні бути узагальнені, але при цьому досить точні. Задоволення цих вимог стає основною метою мови індексування – мови, необхідної для здійснення процесу індексування [3], [11].

Процес індексації в ранніх системах ІП проводився вручну людьми – експертами у відповідній предметній області. На сьогоднішній день ручне індексування все ще вважається кращим ніж автоматичне індексування через його здатність справлятися з невизначеністю. Однак ручне індексування страждає від високих операційних витрат і є практично неможливим для виконання в сьогоднішніх базах документів внаслідок їх розмірів. Автоматична індексація стала активною областю досліджень ІП. Для виконання автоматичної індексації необхідно визначити мову індексування.

Мова індексування складається зі словника термінів і методів побудови відображень.

Словник мови індексування може бути отриманий з тексту описуваного документа або може бути виведений незалежно від тексту. Використання елементів словника, отриманого безпосередньо з тексту, називається підходом

природньої мови [13]. Інший підхід, який використовує терміни словника незалежно від тексту, відомий як метод керованого словника.

Існує багато методів відображення конструкцій у системах ІП [14]. Проте, усі вони мають загальну мету індексації – створювати документи й запити. Для досягнення цієї мети методи побудови індексних конструкцій повинні здійснити наступні кроки:

усунення загальних термінів для документа або запиту, які є поганими дискримінаторами – системи зазвичай мають список загальних термінів, які отримуються зі списку стоп-слів;

розподіл документа й запиту на індивідуальні

терміни; видалення суфіксів і префіксів з термінів;

присвоєння ваг термінам для ідентифікації значущих термінів колекції.

1.4 Аналіз критеріїв оцінки систем інформаційного пошуку

Якість сучасних пошукових систем оцінюється більшою кількістю критеріїв, серед яких основними вважаються [15]:

час обробки запиту;

пертинентність;

релевантність;

точність; повнота;

випадання;

Час обробки пошукового запиту – це час, що минув з моменту введення користувачем запиту до моменту видачі результату пошуковою системою. Час обробки запиту сучасних пошукових систем, таких як Google, Yandex, Bing і т.п., при пошуку сотень тисяч документів обчислюється частками секунди, тобто дуже мало.

Пертинентність – метод оцінки якості пошуку, який визначає, наскільки добре результат пошуку задовольняє інформаційну потребу користувача, що виражена тим або іншим ступенем повноти і точності в пошуковому запиті. Цей метод визначається суб’єктивним сприйняттям користувача.

Релевантність – метод оцінки наскільки гарно список результатів відповідає запиту, тобто семантична відповідність пошукового запиту та знайденого документа. Це більш вузьке поняття, ніж пертинентність, оскільки документ може бути релевантний, але не пертинентний, тобто задовольняти запиту, але не потребу користувача.

Точність – відношення числа знайдених релевантних документів до знайдених документів.

Повнота – відношення числа знайдених релевантних документів до релевантних документів у базі.

Випадання – ймовірність знаходження нерелевантного документа – відношення числа знайдених нерелевантних документів до загального числа нерелевантних документів у базі.

F-міра – характеризує точність і повноту одним значенням. Для оцінки якості ранжування використовуються критерії [19]:

точність перших N документів (precision(n));

R-точність (R-precision).

Останні критерії різняться тим, що для першого обчислюється відношення релевантних документів серед перших N знайдених, до числа N, для другого – ураховується, що кількість знайдених документів може бути менше N, відповідно в якості рішення береться відношення числа релевантних документів серед N документів, де N кількість знайдених релевантних документів. У такий спосіб останній критерій використовується тоді, коли треба врахувати істотну різницю в кількості знайдених по запиту документів.

В атестаційній роботі передбачається аналізування якості ранжування документів для запитів, в яких існує велика кількість посилань, тому в якості основного критерію обраний критерій точність перших N документів.

1.5 Аналіз застосовності мультиагентної організації для реалізації БІП

Однією з дуже серйозних проблем БІП, також як і ІП взагалі, є його висока трудомісткість. Веб надає величезну кількість коштовної інформації, але дуже важкий і затратний за часом пошук тисяч веб-сторінок, що відносяться до предметної області, фільтрація релевантних, аналіз цієї інформації й інтеграції їх у базу знань.

Ця проблема може бути частково вирішена використанням мультиагентної реалізації пошукової системи. Широкий спектр можливостей мультиагентної системи (МАС) дозволяє користувачеві використовувати функціональні, методичні, алгоритмічні й процедурні запити для дослідження й обробки даних, і в тому числі реалізовувати розпаралелювання обробки для скорочення часу на пошук документів і їх ранжування.

Під агентом розуміється комп'ютерна система, розгорнута в деякому середовищі, здатна автономно функціонувати в цьому середовищі в порядку, що задовольняє цілям розробки [19].

Мультиагентна система – це система, що містить набір агентів, які взаємодіють із комунікаційними протоколами й здатні працювати в їхньому середовищі. Різні агенти мають різні сфери впливу, таким чином, маючи контроль (або як мінімум вплив) над різними частинами середовища, ці сфери впливу можуть у деяких випадках перекриватися; те, що вони збігаються, може привести до залежностей між агентами [19], [20]. Мультиагентна система є системою, що містить два й більше агентів, один з яких, як мінімум, є автономним. Агент є програмним модулем, спроможним до взаємодії з іншими агентами й здатним незалежно виконувати завдання. Взаємодія між агентами може здійснюватися за допомогою мови комунікації агентів [21], [12]. МАС застосовувалися в різних областях, включаючи обробку текстів на природніх мовах [13].

МАС складається з декількох агентів з різними функціями, де проблемно-орієнтований модуль рішення є агентом у системі. Агент у реальній системі може бути розроблений з використанням різних інструментів і методів рішення різних завдань. При рішенні складного завдання останнє розділяється на декілька простих підзадач, програми рішення яких взаємодіють між собою, у такий спосіб спрощується рішення складних завдань, а кожна самостійна програма стає Агентом [14].

Можна виділити кілька типів агентів [14].

Реактивний агент часто характеризується як «неінтелектуальний». Це дуже простий компонент системи, який сприймає середовище й здатний в ньому функціонувати. Його здатність відповідає тільки режиму дії стимулу, який можна розглядати як форму комунікації.

Когнітивний агент є агентом інтелектуальним, в основному символічним відображенням, що характеризується знанням ментальних понять. Він має часткове знання про середовище в явних цілях і здатний планувати свою поведінку, запам’ятовувати свої попередні дії, спілкуватися шляхом відправлення повідомлень, вести переговори і т.д.

Інтенсіональний агент або ВБН (Віра, Бажання й Намір) є інтелектуальним агентом, який застосовує модель людського інтелекту та бачення світу, використовуючи ментальні концепції, такі як знання, вірування, наміри, бажання, вибір і зобов’язання. Його поведінка може передбачати присудження переконань, бажань і намірів.

Раціональний агент є агентом, який діє таким чином, щоб одержати найбільш успішне виконання поставлених завдань. Для цього необхідно мати спосіб оцінки продуктивності, якщо можлива мета пов’язана із приватним завданням, яке повинен виконати агент.

Адаптивний агент є агентом, який адаптується до будь-яких змін середовища. Це дуже інтелектуальний агент, тому що здатний змінювати свою мету й базу знань відповідно до змін.

Комунікативний агент є агентом, який передає інформацію всьому, що його оточує. Ця інформація може бути отримана з його власних знань або передана іншими агентами.

З точки зору реалізації МАС агент є процедурою, яка може імітувати поведінку людини та відносини з деяким інтелектом, автоматично працювати й надавати відповідні сервіси [24].

Програмний агент є одиницею програмного забезпечення, яка працює безупинно й незалежно в умовах певних обставин, взаємодіючи з агентами та процесами [15].

Інтелектуальні мультиагентні системи мають значний потенціал використання для різних цілей і областей дослідження, у тому числі й для побудови систем, що реалізують БІП.

ІП є однією зі сфер, у якій успішно застосована мультиагентна організація. Інформаційний агент – агент, що має доступ до одного або декількох джерел інформації, які здатні зберігати й обробляти інформацію, отриману із цих джерел, для того щоб відповісти на запити користувачів або інших інформаційних агентів [4]. Джерела інформації можуть бути різних видів, включаючи веб-сервіси, веб-сайти й традиційні бази даних. ІП, без сумніву, складна область, і агентоорієнтовані обчислення є перспективним підходом для розробки додатків у складних областях. Однак, незважаючи на велику кількість досліджень протягом багатьох років, ряд завдань все ще потрібно розв’язати, щоб зробити агентоорієнтовані обчислення широко прийнятою парадигмою розробки програмного забезпечення й перетворити абстракції агентоорієнтованого програмного забезпечення в прикладні інструменти для подолання складностей у сучасних прикладних областях.

В літературі запропоноване декілька централізованих агентських архітектур для виконання завдань інформаційного пошуку. Серед інших можна назвати Newt [16], Letizia [17], Webwatcher [18], Softbots [19], Webwatcher and Softbots.

Newt [16] була створена як співтовариство інтерфейсних агентів фільтрації інформації, які одержують налаштування користувача та діють від його імені. Для фільтрації інформації агенти використовують алгоритм фільтрації по ключовим словам, у той час як прийняті адаптивні методи є зворотним зв’язком по релевантності з генетичними алгоритмами.

Letizia [17] є розумним інтерфейсним користувацьким агентом, що допомагають користувачеві переглядати веб-сторінки. Пошук інформації приводить до кооперативної роботи користувача й програмного агента: обоє переглядають той же самий простір пошуку зв’язаних веб-документів з метою знайти, що цікавить користувача.

Webwatcher [18] є пошуковим агентом, який переходить по гіперпосиланнях згідно з інтересами користувача, повертаючи список відзначених посилань. На відміну від системопомічників, при перегляді веб-сторінок або пошуку інформації Softbots [19] бере на себе функції користувача й динамічно синтезує необхідну послідовність команд мережі Інтернет відповідною мовою.

Незважаючи на те, що централізований підхід може мати деякі переваги в завданнях інформаційного пошуку, він може зіштовхнутися з декількома проблемами, зокрема, як масштабувати архітектуру для великої кількості користувачів, як надати високу доступність у випадку постійного попиту задіяних служб, а також як надати високу довіру у випадку конфіденційної інформації, такої як персональні дані. У цьому зв’язку в літературі були запропоновані підходящі мультиагентні системи для рішення завдань інформаційного пошуку. До таких систем відносяться CEMAS , агенти ІП і кооперативна мультиагентна система для інформаційного пошуку у вебі, запропоновану в [5]. В CEMAS (Concept Mulit-Agent System) основна ідея полягає в тому, щоб мати спеціалізованих агентів для кожного ключового завдання:

обмін поняттями й зв’язками;

відображення користувача;

пошук нових релевантних документів, що збігаються з існуючими поняттями;

координація агентів.

ІП агенти реалізують мультиагентну модель на основі XML для ІП. Відповідна платформа складена з агентів трьох типів: (i) керуючі агенти для витягу семантики інформації й здійснення актуальних завдань агентів і координаторів, (ii) інтерфейсні агенти, розроблені для взаємодії з користувачами, і (iii) пошукові агенти для пошуку інформації у Вебі. Нарешті в [12] основна ідея полягає в прийнятті розумних агентів, які імітують поведінку користувачів, що шукають інформацію. У цих цілях агенти можуть створювати профілі користувача для того, щоб передбачити й досягати його/її бажані цілі.

1.6 Постановка задач дослідження

Аналіз існуючих систем БІП, методів і засобів його реалізації, методів і засобів реалізації одномовного ІП виявив наступні проблеми, властиві моделям і методам, використовуваним при побудові пошукових систем:

Неточний переклад термінів внаслідок їхньої неоднозначності (контекстної залежності) і складності граматики природньої мови при використанні систем машинного перекладу, що приводить до знаходження й обробці нерелевантних документів поряд з релевантними.

Більша обчислювальна складність одержання оцінок релевантності результатів, що приводить до неприпустимо великого часу ранжування для великої кількості знайдених документів. Проведений аналіз показує, що мультиагентний підхід ефективний у тому випадку, якщо структура виконуваних операцій і взаємодія між агентами добре пророблені. Досягнення цієї мети припускає розробку структурної моделі БІП.

У такий спосіб для створення ефективної системи БІП необхідно:

Поліпшити якість перекладу ключових словосполучень.

Виконати аналіз процесу БІП і розробити його структурну модель для наступної декомпозиції й реалізації пошукової системи з використанням мультиагентної технології, що забезпечує гарне розпаралелення й масштабування інформаційних систем.

Розробити метод оцінювання релевантності документів, що забезпечує точність результатів на рівні існуючих методів, але має більш низьку обчислювальну складність.

Розробити програмну модель запропонованих моделей і алгоритмів.

2 ОПИС ПРОВЕДЕНИХ ТЕОРЕТИЧНИХ ДОСЛІДЖЕНЬ

2.1 Аналіз моделей оцінки релевантності інформаційного пошуку

Алгоритми зіставлення є основними в системах ІП. Як тільки відображення документів і запитів побудовані, ці відображення використовуються алгоритмом зіставлення для досягнення трьох завдань:

пошуку й визначення елементів, що відносяться до запиту користувача;

ідентифікації зв'язаних документів, що й відрізняються, у колекції;

прогнозування релевантності документів на основі запиту користувача.

Завдяки використанню індексних термінів в певній області та змісті багато алгоритмів зіставлення були запропоновані дослідниками в області ІП протягом багатьох років. У цьому розділі буде розглянуто чотири різні моделі зіставлення, а саме: логічні, засновані на векторному просторі, ймовірнісні та моделі мереж логічного висновку.

Логічна модель ІП базується на класичній теорії множин. Документи в цій моделі представлені як множини термінів, що знаходяться в них (використовуються не всі слова), а запити представлено логічними виразами. Ключові слова в запиті можуть бути пов'язані з допомогою булевих операторів И, АБО й НЕ [11]. Кожний термін може мати один або два логічні стани, він може або бути присутнім (логічна 1), або відсутнім (логічний 0) [11].

Релевантність документа запиту користувача визначається обчисленням логічного значення запиту, як 1 або 0. Значення 1 надається кожному терміну запиту, який існує в множині термінів, що представляє документ, і 0 для кожного терміну, який не існує у відображенні документа [11].

Логічна модель вважається найпростішим алгоритмом зіставлення в ІП. Взаємозв'язки або подібності між окремими документами при цьому не використовуються так само, як і взаємозв'язки між термінами в запиті. У

системах, які використовують Логічна модель, запит користувача представлений тільки як комбінації термінів, які містять передбачуваний релевантний документ. Наприклад, комусь будуть потрібні документи, що містять два терміни (конструкція й виробництво) або три терміни (дешевий, електронний і гарний). Запит Q може бути сформульований у такий спосіб:

```
Q = (конструкція AND виробництво) OR (дешевий AND
електроніка AND гарний).
```

Простота реалізації є головною перевагою логічної моделі. Відповідність документів запиту обчислюється винятково на основі бінарного рішення: чи існують терміни запиту в документі. У результаті документи, витягнуті логічною моделлю, зважуються по користувацьких запитах. Таким чином, перший витягнутий документ не обов'язково буде найбільш релевантним [17].

При використанні цієї системи швидко стало очевидно, що логічні системи мають ряд недоліків, наприклад відсутність внутрішнього поняття ієрархії документа, а для користувача важко сформувати гарний пошуковий запит (гарний пошуковий запит залежить від здатностей і знань користувача).

Хоча логічні системи зазвичай повертають відповідні документи в деякому порядку, наприклад, упорядкованими по даті або по іншій особливості документа, ранжування по релевантності в такій системі не виконується. Функція пошуку в логічній моделі ІП розглядає документ тільки як релевантний або нерелевантний [20].

З аналізу джерел було зроблено висновок, що логічні системи менш ефективні, ніж системи з пошуком по релевантності, однак багато користувачів як і раніше використовують логічні системи, тому що вони відчувають, що можуть більшою мірою контролювати процес пошуку. Тим не менш, більшість користувачів щодня очікують від пошукової системи, щоб результати пошуку були ранжовані за степеню релевантності. Системи пошуку по релевантності ранжують документи за оцінкою корисності документа для запиту користувача. Більшість таких систем надають числовий бал кожному документу й ранжують документи по цьому балу [11], [31].

Вирішенням проблем логічної моделі стало метою досліджень в області ІП. Це дослідження зосереджене на побудові моделі пошуку, яка має можливість зважувати релевантність документів запиту.

2.2 Застосування методів векторних моделей

У векторної моделі текст представлений вектором слів [12]. Відображення набору документів як векторів у загальному векторному просторі, відоме як векторна модель, має фундаментальне значення при реалізації ІП.

Воно використовується при ранжуванні оцінки відповідності документів запиту, при класифікації й кластеризації документів. Спочатку були запропоновані основні ідеї скоринга у векторному просторі; ключовим кроком у цьому напрямку є визначення видів запитів як векторів у тому ж векторному просторі, що й колекція документів [11]. Вектор документа характеризує відносну важливість термінів у документі. Виявлення термінів – окреме завдання, але терміни, як правило, є словами й фразами. Якщо в якості термінів (ключових слів) обрані слова, то кожне слово в словнику стає самостійним виміром у векторному просторі дуже великої розмірності. Будь-який текст може бути представлений вектором у цьому просторі великої розмірності. Якщо термін належить тексту, то він одержує ненульове значення у векторі тексту разом з відповідною розмірністю. Оскільки будь-який текст містить обмежений набір термінів (словник може складатися з мільйонів виражень), більшість термінів дуже рідкісна. Більшість систем на основі векторної моделі працюють у позитивному квадранті векторного простору, тобто ніякому терміну не призначене негативне значення [11], [12].

Для того щоб призначити числовий бал відповідності документа запиту, модель вимірює подобу між вектором запиту (тому що запит також є просто текстом і може бути перетворений у вектор) і вектором документа. Повна

подібність між двома векторами не властива моделі. Як правило, у якості міри розбіжності між векторами використовується кут між двома векторами, а косинус кута використовується як числова міра подібності (оскільки косинус має гарну властивість, це 1:0 для однакових векторів і 0:0 для ортогональних векторів). У якості альтернативи, як міри подібності між двома векторами часто використовується внутрішній (або скалярний) добуток. Якщо всі вектори мають довжину, рівну одиниці, то косинус кута між двома векторами є таким же, як і їх скалярний добуток.

Множина документів у колекції можна розглядати як набір векторів у векторному просторі, в якому є одна координата для кожного терміну. Це відображення ігнорує відносний порядок термінів у документі. Завдання полягає в тому, щоб кількісно оцінити подібність між двома документами в цьому векторному просторі.

Першою спробою є розгляд величини відмінності векторів між двома векторами документів. Цей метод має недоліки: два документи з дуже схожим змістом можуть мати значну різницю векторів просто тому, що один набагато довше іншого. Таким чином, відносні розподіли термінів можуть бути однаковими в обох документах, але абсолютний термін може повторюватися набагато частіше [11], [13].

Модель векторного простору пропонує життєздатний компроміс обробки в ІП. Саме цей спосіб розрахунку подібності документів та запиту буде використаний для визначення подібності між користувачами та документами, для здійснення процесу фільтрації (витягу) у поточній роботі, оскільки вона поєднує в собі ясність і простоту, а також пропонує ефективний метод відображення документів.

Метод також дозволяє послідовне використання ваг термінів у відображеннях запитів для трьох моделей генерації профілів: явної, непрямої й гібридної.

2.3 Імовірнісні моделі

Сімейство моделей інформаційного пошуку (ІП) засноване на тому, що документи повинні бути ранжовані по зменшенню ймовірності релевантності їх запиту. Це часто називають принципом ранжування по ймовірності [11]. Оскільки дійсні ймовірності недоступні системі ІП, імовірнісні моделі оцінюють ймовірність релевантності документів для запиту. Ця оцінка є ключовим елементом імовірнісної моделі, і це те, чому більшість імовірнісних моделей відрізняються друг від друга.

Первісну ідею ймовірнісного ІП запропонували Марус і Кунанс в 1960 р. Після було запропоновано багато ймовірнісних моделей з різними способами оцінки ймовірності.

Припускаючи незалежність для термінів (як правило, слів), можливо використовувати Байєсівську оцінку для знаходження ймовірності наявності/відсутності терміну в релевантних/нерелевантних документах, яка використовує ймовірність наявності терміну в релевантному документі для всіх термінів, близьких до запиту й документу, і ймовірність відсутності у релевантних документах для всіх термінів, що присутні у запиті й позначене відсутнім у документі. Якщо позначене через, тобто формула ранжування скорочується й приймає вид. Різні припущення для оцінки приводять до різних функцій ранжування.

Імовірнісна модель пошуку заснована на припущеннях, наприклад, що 50% документів, що містять термін, релевантний цьому терміну, однак не всі припущення відповідають реальності. Загальне число релевантних документів необхідно вгадати, і p – константа, що не завжди вірно [13]. Для досягнення точних результатів імовірнісна модель пошуку вимагає того, щоб терміни були незалежні. Обчислення ваги нехтує частотністю терміну і його розташуванням у документах, але відображення документа в імовірнісній моделі дуже просте. У порівнянні з нею модель векторного простору поєднує ясність і гнучкість, її

основна алгебраїчна модель достатньо визначена і добре розуміється, а документи представлені більш докладно.

2.4 Моделі логічного висновку

У цієї моделі пошук документів моделюється як процес отримання висновку з мережі логічного висновку [14]. Більшість операцій, реалізованих у системах ІП, може бути реалізовано згідно з цією моделлю. У найпростішій реалізації цієї моделі кожному терміну зіставляється певна оцінка значимості, а число кредитів, отримане від декількох термінів, що зустрічаються в документі, накопичується, даючи запиту обчислити еквівалентне числове значення значимості для документа. З точки зору експлуатації сила обробки терміну для документа може бути розглянута як вага терміну в документі, а ранжування документів у найпростішій формі згідно з цією моделлю стає схожим з ранжуванням документів по векторній моделі та ймовірнісним моделям, описаним вище.

Модель мережі логічного висновку здатна здійснювати ранжування на основі багатьох даних джерел наявності шляхом перебору цих сутностей. У моделі використовується мережа Баєса для моделювання документів і їх змісту, а також запиту. Мережа логічного висновку складається із двох підмереж: мережі документів, сгенерованої під час індексації й потім статистичної обробки під час пошуку, і мережі запитів, сгенерованої з тексту запиту під час пошуку.

Мережа документів являє собою колекцію (набір) документів і складається з вузлів для кожного документа (називається вузлами документа) і вузлів для кожного поняття (вузли понять документа). Вузли документів являють собою елементи, що витягають усередині мережі, тобто, ті предмети, які ми прагнемо бачити в результаті ранжування. Звичайний зв’язок між вузлом

документа й вузлом поняття показує, що вміст документа представлений поняттям. Кожний зв'язок складається з умовної ймовірності або ваги, що показує силу взаємозв'язку. Оцінка вузла здійснюється з використанням значень батьківських вузлів і умовних ймовірностей.

Мережа запитів являє собою поданий запит і складається зі структури вузлів, що представляють необхідні поняття (вузлів понять запиту), та операторів (вузлів операторів запиту), з'єднаних у формі перевернеої деревоподібної структури. Мережа запитів побудована з використанням кінцевого листа (вузол I на рис. 2.1), який представляє інформацію, необхідну користувачеві. Структура дозволяє використання статистичних операторів та статистичних апроксимацій логічних операторів, які наведено в таблиці 1.2 (як у реалізації запитів INQUERY [15].

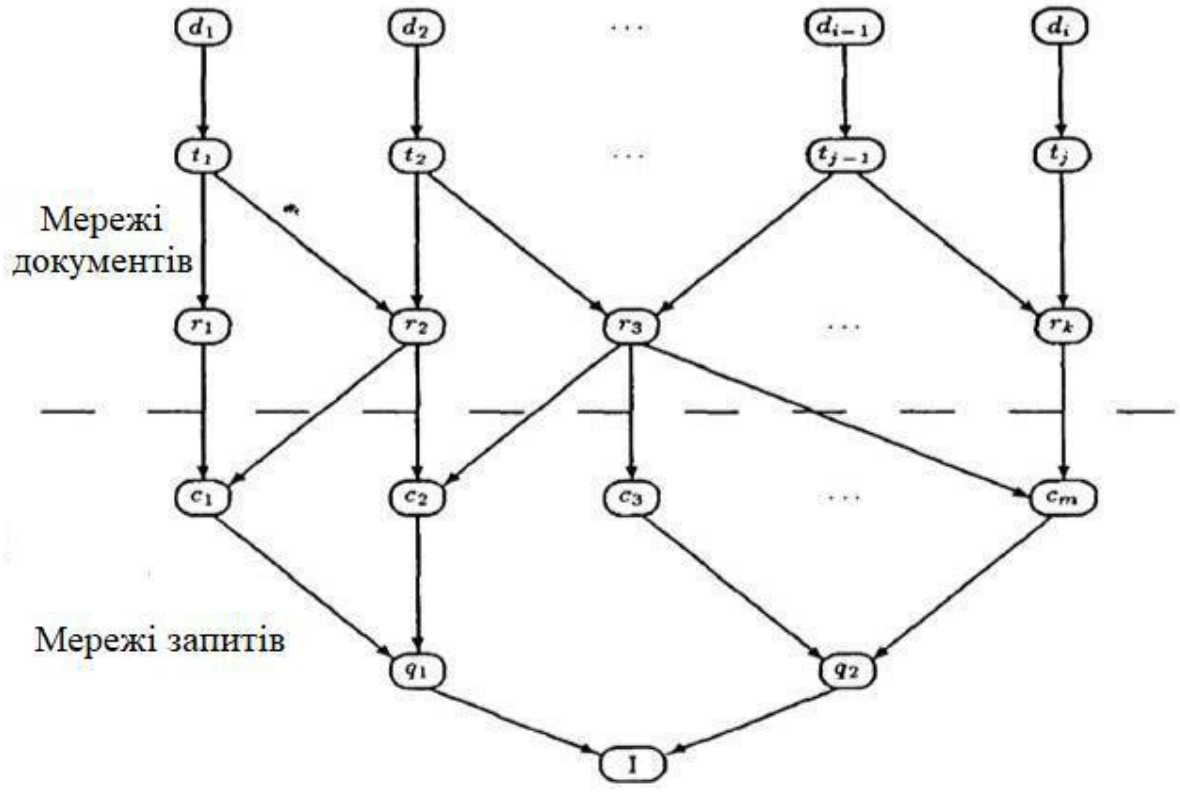


Рисунок 2.1 – Приклад простої моделі мережі логічного висновку [15]

Дві інших дії реалізують здійснення запиту: процес приєднання, у якому мережа запитів приєднується до мережі даних для формування закінченої мережі, що вважається завершеним, якщо поняття в обох мережах однакові, і процес оцінки, у якому повна мережа логічного висновку оцінюється для кожного вузла документа, щоб сформувати ймовірність релевантності запиту. Оцінка ініціюється шляхом присвоєння виходу одного вузла документа значення 1, а всім іншим вузлам документа 0. Це робиться, у свою чергу, для кожного вузла документа і мережа оцінюється (див. [16], [17]). Імовірність релевантності документа береться з останнього вузла 1 і використовується для здійснення ранжування.

Таблиця 2.1 – Оператори, підтримувані моделлю мережі логічного висновку

And	Кон’юнкція термінів
Or	Диз’юнкція термінів
Not	Заперечення терміну (вхідний умовивід)
Sum	Підсумування вхідних умовиводів
Wsum	Зважена сума вхідних умовиводів
Max	Максимум вхідних умовиводів

Аналіз моделей оцінки відповідності документів запиту показав, що векторна модель має дуже просту математичну форму. Імовірнісна модель має сильну математичну основу, а моделі мереж висновку подолали проблеми імовірнісних оцінок, заснованих на Баєсовських мережах [18].

В поточній роботі буде запропонована модель, що долає проблеми оцінки й обмеження моделей оцінки релевантності, розглянутих вище. Ця модель поєднує векторний метод і систему нечіткого логічного висновку.

2.5 Структура документа

Для розробки нових алгоритмів збільшення релевантності документів – результатів, отриманих автоматизованою системою ІП у Веб – необхідно виконати аналіз операцій одно- та багатомовних інформаційних пошуків.

Частота терміну t_f у документі може показувати важливість цього терміну в документі. Інакше кажучи, частота терміну може бути використана для узагальнення вмісту документа. Однак використання тільки однієї частоти документа недостатньо, тому що вона не може бути використана для ефективного виділення [13] документів у зборах.

Розглянуто наступний випадок: слово «комп'ютер» може мати дуже велику частоту в документах, що стосуються теорії ЕОМ і її додаткам. Майже кожний документ має високу частоту використання терміну «комп'ютер», тому що областями збори є теорія ЕОМ і додатка. Цей випадок показує, що слово «комп'ютер» не дозволяє виконати відбір документів. Чим більша кількість документів, де зустрічається деякий термін, тим менша важливість цього терміну при пошуку. Таким чином, гарне відображення документа повинно бути здатне узагальнювати й виділяти документи одночасно.

Інвертована частота документа *idf* може бути використана для зважування терміну як дискримінатор. Комбінація *tf* і *idf* зазвичай використовується в рівнянні (3.1).

$$w_{i,j} = tf_{i,j} \text{ x } idf_j \ , \tag{3.1}$$

Вага називається *tf-idf*, що розшифровується як частота терміну – інвертована частота документа, і вага *tf-idf* часто використовується в ІП і аналізі текстів. Ця вага є статистичним методом, що використовується для оцінки,

наскільки важливе слово в документі, колекції або зборах. Важливість збільшується пропорційно числу появ слова в документі, але вона скомпенсована частотою слова в зборах. Варіації схем зважування по *tf-idf* часто використовуються пошуковими системами в якості центрального інструмента скорінга й ранжування по релевантності документів конкретному запиту користувача.

Вага терміну *t* у документі *d* згідно *tf-idf* визначена рівнянням:

$$w(t,d) = (tf - idf)_{i,j} = tf_{i,j} \times idf_j \quad , \tag{3.2}$$

тобто. вага *tf-idf* є добутком двох статистик, частоти терміну й інвертованої (зворотної) частоти документа [14]. Різні способи визначення точних значень обох статистик представлено в табл. 3.1, 3.2.

Таблиця 3.1 – Варіанти зважування частоти терміну в документі

Схема зважування	Вага TF
Двійкова	{0,1}
Частотна	$f_{t,d}$
Логарифмічна нормалізація	$1 + \log f_{t,d}$
Подвійна нормалізація 0.5	$0.5 + 0.5\frac{f_{t,d}}{\max f_{t,d}}$
Подвійна нормалізація K	$K + (1-K)\frac{f_{t,d}}{\max f_{t,d}}$

Таблиця 3.2 – Варіанти зважування зворотної частоти документів

Схема зважування	Вага IDF
Унарна	1
Інвертована частотна	$\log \frac{N}{n_t}$
Інвертована частотна зглажена	$\log(1+\frac{N}{n_t})$
Інвертована частотна	$\log(1+\frac{\max_t n_t}{n_t})$
Імовірнісна інвертована частотна	$\log \frac{N-n_t}{n_t}$

3.2 Алгоритм нормування ваг термінів і функції ранжування

Системи автоматизованого інформаційного пошуку працюють із документами змінної довжини в зборах текстів. Для корекції розбіжностей довжин документів використовується коефіцієнт нормування. Якщо не використовувати коефіцієнт нормування, ті короткі витягнуті документи не можуть бути визнані релевантними. Нормування використовується для прийнятного пошуку документів усіх довжин [15], [16], [11] і для компенсації переваги, яку мають довгі документи щодо коротких у порядку пошуку. До таких переваг відносяться:

велика кількість використовуваних термінів;

часте використання тих самих термінів.

Нормована частота терміну *t* у документі *d* показана в (3.3) як відношення частоти кожного терміну в документі до максимальної частоти терміну в цьому документі.

$$f_{dt} = \frac{f_{req}(t,d)}{max(f_{req}(t,d))} \quad . \tag{3.3}$$

Одна з найпростіших функцій ранжування обчислюється шляхом підсумовування оцінок *tf-idf* для кожного терміну запиту:

$$score(q,d) = \sum_{t \in q} w(t,d) \quad . \tag{3.4}$$

Функція ранжування використовує *tf-idf* для опису документа в моделі векторного простору. Як зазначено вище, ця модель заснована на інтерпретації документів і запитів, як векторів у багатомірному просторі документів [14], [31], [17]. Косинусний метод характеризує кут між вектором запиту й вектором документа в *m*-мірному просторі документів. Подібність вектора документа й запиту в цьому випадку рівняється косинусу кута між ними:

$$\cos(\vec{q},\vec{d_i}) = sim(\vec{q},\vec{d_i}) = \frac{\vec{q}}{|\vec{q}|} \cdot \frac{\vec{d_i}}{|\vec{d_i}|} = \frac{\sum_{j=1}^{m} w_{ij} \times w_{qj}}{\sqrt{\sum_{j=1}^{m} w_{ij}^2} \times \sqrt{\sum_{j=1}^{m} w_{qj}^2}} \ , \tag{3.5}$$

де q –вектор запиту,

di –вектор документа i,

$w_{i,j}$ – вага терміну j у документі i, w_{qj} – вага терміну j у запиті q.

Якщо усі вектори нормалізовані, то косинус обчислюється як:

$$sim(q,d_i) = \cos\theta = \sum_{j=1}^{m} w_{ij} \times w_{qj} \tag{3.6}$$

Рівняння використані для оцінки релевантності знайдених текстових документів.

На рис. 3.1 показаний приклад відображення моделі векторного простору для системи із двох термінів. Кожна вісь у просторі відповідає терміну. Положення кожного вектора-документа в просторі визначене магнітудою (вагою) термінів у цьому векторі. Обчислення міри подібності між вектором-документом і вектором-запитом здійснюється як функція магнітуд схожих термінів у відповідних векторах, яка може бути використана для ідентифікації релевантних документів. Найпростіша схема обчислення подібності полягає в припущенні того, що документ, що містить більшість термінів запиту, буде найбільш релевантним. Подібність між $D1$ і $D2$ буде вимірятися кутом α. Подібність між документами $D1$ і запитом Q виміряється кутом θ.

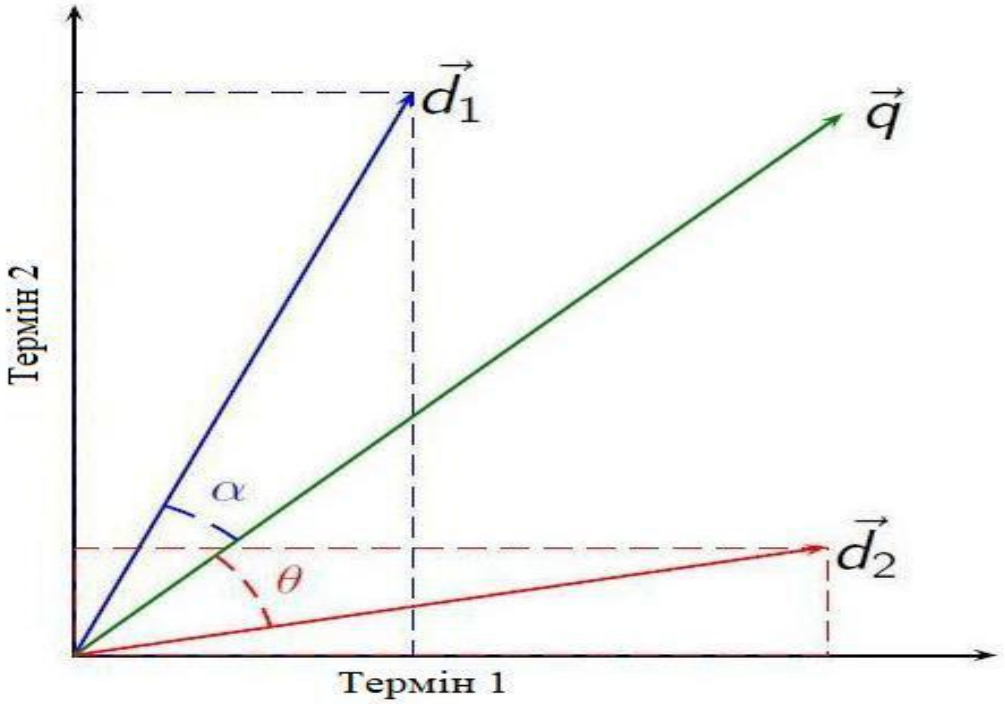


Рисунок 3.1 – Двовимірний векторний простір

3.3 Розробка мультиагентної архітектури системи БІП

Пошукова система Веб – це програмне забезпечення, розроблювальне для пошуку інформації у Всесвітній павутині. Результати пошуку зазвичай представляються у вигляді рядка результатів, що часто посилаються на сторінки результатів пошукової системи.

В поточній роботі розглядається мультиагентна реалізація пошукової системи. Спочатку розглянуто одномовний пошук. МАС, призначена для здійснення одномовного інформаційного пошуку, повинна виконувати чотири узагальнені операції: 1 – введення ключового слова, 2 – пошук в Інтернеті по ключовому слову; 3 – витяг необхідної інформації з Веб-джерел і аналіз добутих текстів; 4 – ранжування результатів і збереження вихідних даних у базі даних. Запропонована мультіагентна система складається із чотирьох шарів, кожен з

яких виконує одну із зазначених вище узагальнених операцій інформаційного пошуку (рис. 3.2).

Рисунок 3.2 – Функціональна схема МАС для одномовного пошуку

Рівень інтерфейсів – на цьому рівні інтерфейс користувача дозволяє взаємодіяти із системою через графічні текстоорієнтовані інтерфейси шляхом введення ключового слова.

Пошуковий рівень**.** На цьому рівні пошуковий агент посилає ключове слово в пошукову машину Google, яка повертає посилання, збираючи URL доступних веб-сайтів в Інтернеті, і зберігає їх у базі даних.

Рівень аналізу текстів. На даному рівні агент автоматично витягає тексти з Url-посилання, зазвичай з великої кількості різних неструктурованих текстових ресурсів і виявляє в них корисну інформацію, використовуючи лексемізацію (видалення розділових знаків, спеціальних символів і заміну відступів і інших нетекстових символів одним пробілом), фільтрацію стопових слів (видалення нейтральних слів, які не характеризують документ) і лематизацію (виявлення загального кореня слів). Також на цьому етапі розраховуються ваги термінів у знайдених документах, а також виконується їхня нормалізація.

Рівень ранжування результатів – на останньому рівні виконується розрахунок релевантності текстів і ранжування документів [18]. Рівні пошуку й аналізу реалізовані агентами, які використовують інструмент для інформаційного пошуку з відкритим вихідним кодом Rapidminer.

Кожен агент може підключатися друг до друга, а також взаємодіяти один з одним при рішенні завдання автоматизованого пошуку документів і витягу ваг кожного терміну після лексемізації, видалення стопових слів, лематизації з використанням стандартної програмної Java-платформи (JADE) і додатка Rapidminer. Вхідний запит у такій системі представлений для однієї мови. Також можливо повторення запиту з використанням будь-якої іншої мови, але це займе багато часу, і модель системи буде негнучкою. Багатомовна система дозволить розв’язати цю проблему.

Реалізація автоматичного багатомовного пошуку крім операцій одномовного пошуку допускає виконання перекладу й обробки текстів на різних мовах. Усі ці операції доцільно виконувати різними агентами, як показано на рис. 3.3.

Користувач вводить ключове слово запиту за допомогою інтерфейсу на кожній з трьох мов. Агент 1 вводить запит. Агент 2 перекладає запит на інші мови. Агент 3 шукає Url-посилання в Інтернеті. Агент 4, Агент 5 і Агент 6 здійснюють пошук інформації із цих посилань і витягають необхідну інформацію, а також виконують аналіз текстів і розрахунок ваг термінів для кожного документа. Агент 7 виконує ранжування результатів пошуку.

При декомпозиції розроблювальної мультиагентної системи враховувалися два принципи:

спеціалізація агента для виконання однієї (узагальненої) операції;

зменшення обсягів переданої в повідомленні інформації.

Запропонована архітектура тримовної многоагентної системи пошуку може бути розширена для довільної кількості мов. Для додавання ще однієї мови слід додати словники для перекладу термінів на цю мову й агента, що вміє виконувати аналіз текстів на цій мові.

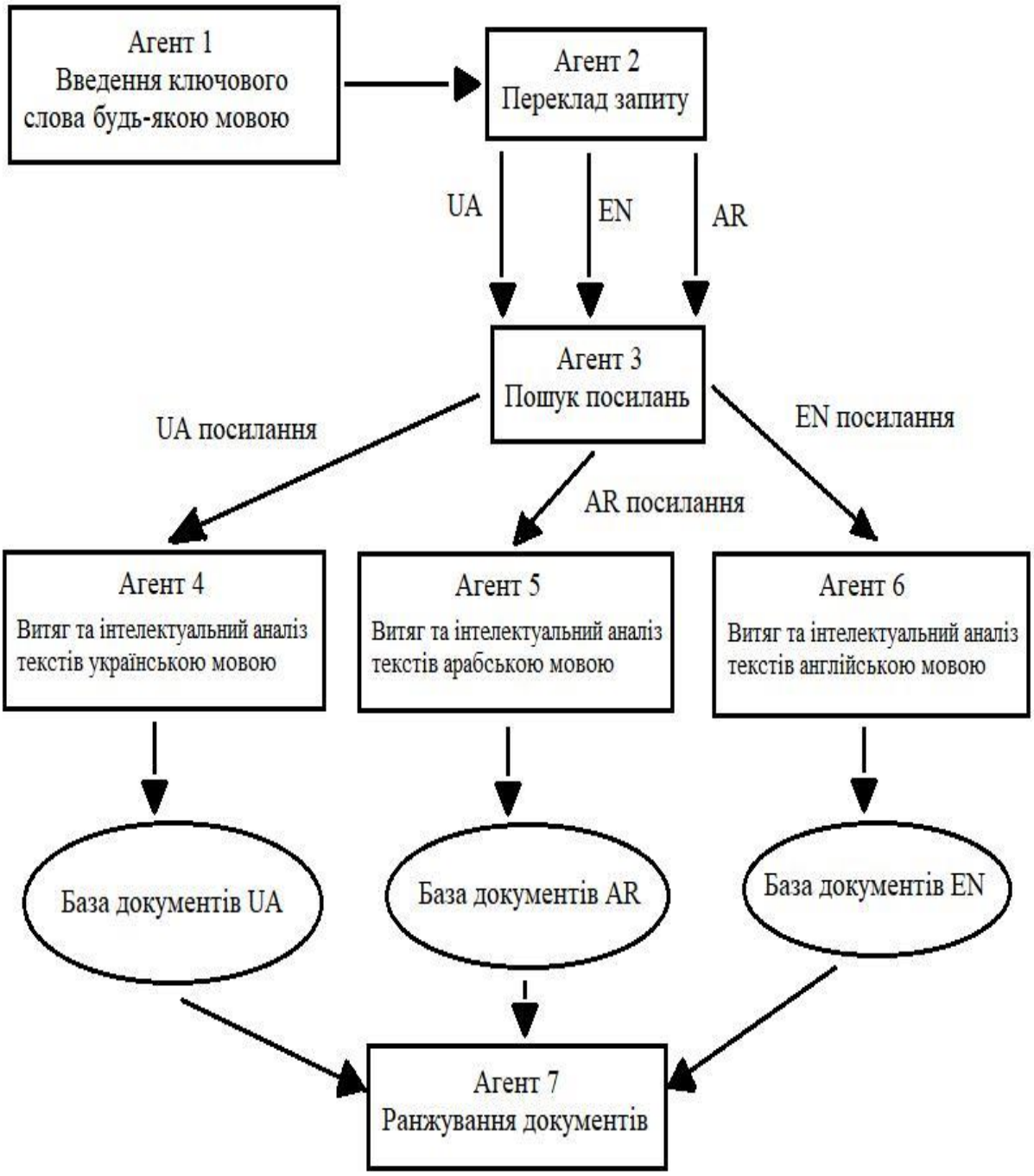


Рисунок 3.3 – Архітектура мультиагентної системи БІП

Розробимо модель процесу пошуку, що враховує декомпозицію процесу по тим же принципам.

2.6 Розробка моделі інформаційного пошуку для однієї мови

Щоб показати деталізацію перерахованих узагальнених операцій, модель, що представляє процес пошуку, повинна дозволяти відображати відносини між окремими об’єктами, між окремим об’єктом і групою об’єктів і між групами об’єктів. У якості такої моделі в роботі використовується метаграф – математична модель, заснована на графі, в якому кожна позначена вершина є множиною інших, що включають одну або більше вершин. Він зберігає всі властивості графів. Метаграфи використовуються в багатьох додатках в сфері систем обробки інформації, підтримки прийняття рішень, моделей керування, систем керування, заснованих на правилах, у яких одна робота складається з багатьох завдань обробки інформації, виконуваних людьми або машинами. Метаграфи в таких ситуаціях можуть надати корисну й всеосяжну функцію для моделювання шляхом розширення функціональних можливостей, запропонованих традиційними графовими структурами, тобто диграфами та гіперграфами. Метаграф дозволяє різним компонентам процесу бути представленим як графічно, так і аналітично.

Метаграф $S = <X, E>$ розглядається як графічне зображення, що складається із двох множин X і E. Тут X є множиною, що породжує, a E – множиною ребер, визначених множиною, що породжує множину, що породжує, X метаграфа S, тобто набір елементів $X = \{x_1\}, l= 1,2,3,\ldots\ldots,L$ представляє змінні й з’являється в ребрах метаграфа.

Ребром метаграфа є пара e = <Ve, We> E (де E є набір ребер), який складається із внутрішньої вершини Ve $\subseteq X$ і зовнішньої вершини We$\subseteq$X. Простий шлях $h(x, y)$ від елемента x до елемента y є послідовністю ребер $<e_1, e_2, \ldots, ek>$, такий що x – внутрішня вершина (e_1), y – зовнішня вершина (ek) і для $ek, k=1,2,\ldots,K-1$. При цьому $x \neq y$. Загальним входом x у шляху (позначеному як загальний вхід(x)) є набір усіх інших елементів внутрішньої вершини дуг, які шляхи не перебувають у зовнішній вершині інших дуг шляху, і загальний вихід

y (позначений як загальний вихід(y)) є набором всіх елементів зовнішніх вершин, відмінних від *y*. Довжиною простого шляху є число дуг у шляху. Модель метаграфа у такий спосіб дозволяє показати два рівні операцій інформаційного пошуку – елементарні й узагальнені, але не дозволяє при цьому врахувати невизначеність використовуваних оцінок.

Відображення невизначеної відповідності або невідповідності документа множини результатів можливо в нечітких графових моделях. Так, для нечіткого графа нечітка множина вершин визначена функцією μ: $X \rightarrow [0, 1]$, а нечітка множина ребер функцією ρ: $\rightarrow [0, 1]$.Таким чином, нечіткий граф може бути описаний множинами вершин і ребер, а також двома функціями μ і ρ. Для зручності позначення множини *X* і *E* опускають, і тому в літературі використовується позначення *G*= (*X*, *E*) або G = (μ, ρ), де вершини й ребра мають значення приналежності.

Був запропонований ряд нечітких графів для відображення невизначених відносин між нечіткими елементами. Однак існування нечітких графів не дозволяє ефективно моделювати прямі відносини між множинами нечітких елементів. Широко використовувані нечіткі графи мають наступні обмеження, розповсюджувані й на традиційні графи:

ненаправлений нечіткий граф може відображати відносини між існуючими двома змінними, він не може відобразити напрямок цих відносин;

відносини входів і виходів між парами елементів можуть бути описані нечітким спрямованим графом. Однак він не може відобразити відносини, де є більш ніж одна змінна на вході й/або на виході;

нечіткий гіперграф описує будь-яке нечітке відношення як множину нечітких елементів, але не може відрізняти вхідні змінні від вихідних;

− використовуючи дуги в комбінації з ребрами, нечіткими ТА/АБО, за допомогою графа можна спробувати відобразити відносини, навіть якщо є більш ніж одна вхідна й вихідна змінні. При описі відносин між множинами змінних виникає занадто багато сполучених ребер для нечітких ТА/АБО, що важко розрізняти на графі.

Для всіх графів відсутні методи алгебраїчного аналізу для маніпулювання відносинами між множинами елементів. Це мотивує розробку нечіткого метаграфа. Основна відмінність між нечіткими метаграфами й традиційними структурами теорії графів полягає в тому, що нечіткий метаграф описує прямі відносини між множинами елементів замість окремих елементів. Кожне ребро в нечіткому метаграфі є впорядкованою парою множин елементів. А ребро не є ні впорядкованою парою елементів у нечіткому спрямованому графові, ні неупорядкованою множиною елементів у нечіткому гіперграфі.

Нечіткий метаграф являє собою узагальнення концепції метаграфа, який був запропонований [18]. Нечіткий метаграф може надати рішення в умовах складних обставин, при яких інші структури графів знайти дуже складно. Таким чином, користувач за допомогою цієї ефективної системи прийняття рішень зможе зробити швидкі ефективні рішення для подолання проблеми [19]. Нечіткий метаграф є концепцією фазифікації чіткого метаграфа за допомогою нечіткої множини.

Нечітка множина, що породжує, є множино. вузлів усіх елементів нечіткого метаграфа. Розглянемо кінцеву множину $X=\{x_1, x_2, x_3 \ldots x_l\}$. Нечіткий

метаграф у розглянутих роботах представлений як у трійка якій – нечітка множина X, і нечітка границя множини $\tilde{E}=\{\tilde{e}_k,\ k=1,\ 2,\ 3 \ldots\ K\}$. . Також, як і у звичайному метаграфі кожен компонент описано упорядкованою парою $< V_k,\ W_k>$. У парі k підмножина внутрішніх вершин. k і k – підмножина зовнішніх вершин.

На рис. 3.4 показано нечіткий метаграф, чия гранична множина складається з

$$\tilde{e}_1 = <\{\tilde{\tilde{X}}_1, \tilde{\tilde{X}}_2\},\ \{\tilde{\tilde{X}}_3\}> \text{ и } \tilde{e}_2 = <\{\tilde{\tilde{X}}_3, \tilde{\tilde{X}}_4\},\ \{\tilde{\tilde{X}}_5, \tilde{\tilde{X}}_6\}>.$$

Аналіз застосовності різних видів структурних моделей, дозволяє побудувати модель одномовного пошуку у вигляді нечіткого орієнтованого метарафа.

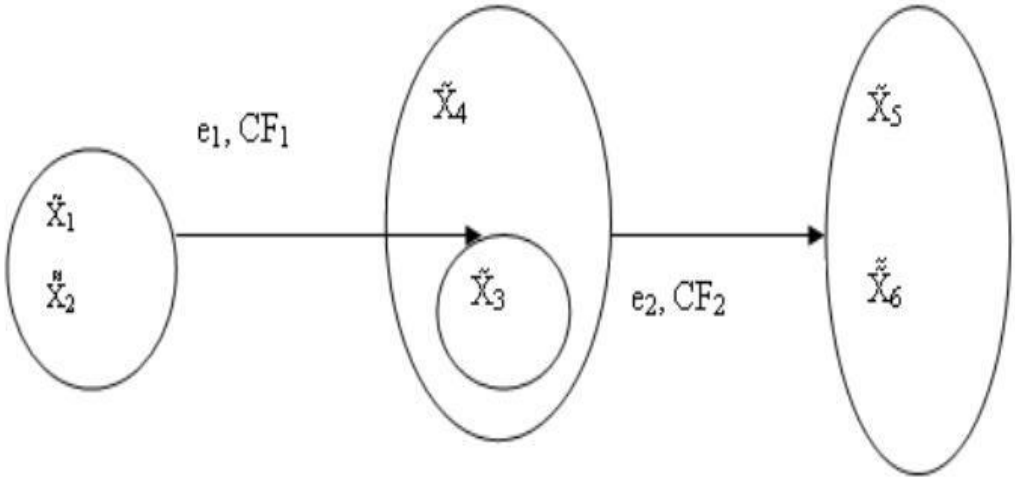


Рисунок 3.4 – Нечіткий метаграф [18]

В даній роботі під орієнтованим метаграфом розуміється четвірка:

$$\vec{M} = (X, V, U, E),$$

де X – множина вершин,

U – множина дуг,

u X x X,

V – множина метавершин $v \in X$ або $V \in 2^X$,

2^X – множина усіх підмножин множини X,

E – множина метадуг $e \in V \times V \cup V \times X \cup X \times V$.

Таким чином, кожна метавершина являє собою множину, що включає одну або більше вершин множини X, а метадуги з'єднують метавершини v між собою або метавершини v з вершинами x або вершини x з метавершинами v.

У метаграфі – моделі одномовного інформаційного пошуку: X – уточнена множина операції процесу пошуку; V – множина узагальнених операцій, виконуваних агентами; U – множина зв'язків між базовими операціями; E – множина повідомлень, переданих між агентами. Для відображення в моделі неоднозначності з погляду релевантності результатів виконуваних операцій зважимо вершини $x \in X$. У якості ваг будемо використовувати нечіткі оцінки *виду* $\mu: X \rightarrow [0, 1]$. *Зміст цієї ваги для кожної операції свій. Так, для операції*

введення запиту x_1 вага $\mu_1 \leftrightarrow \xi_1$ відображає вплив обраних користувачем для запиту термінів і їх порядку в запиті на множину одержуваних при пошуку посилань, для операції пошуку неоднозначність пов'язану з різними способами індексування документів, а для операції ранжування – багатозначність оцінки релевантності знайдених документів по вагах термінів у запиті й документах.

Остаточно маємо модель процесу інформаційного пошуку у вигляді нечіткого метаграфа:

$$S = (<X, M>, V, U, E),$$

де X – множина вершин метаграфа, відповідних до множини операцій, які здійснюються в процесі пошуку, обробки й ранжування документів;

V – множина метавершин, відповідних до підмножин операцій, виконуваних агентами;

U – множина дуг – множина зв'язків між операціями (передача керування й даних);

E – множина метадуг – множина повідомлень, переданих між агентами.

Метаграф одномовного пошуку показано на рис 3.5.

Для нечіткого метаграфа мультиагентної системи одномовного інформаційного пошуку:

$X = \{x_i, i = 1,10\}$ – основні операції пошуку ;

$\nu_1 = \{x_1\}$ – відповідає інтерфейсному агенту;

$\nu_2 = \{x_2, x_3, x_4\}$ – відповідає пошуковому агентові;

$\nu_3 = \{ x_5, x_6, x_7, x_8, x_9\}$ - відповідає агентові витягу документів і попередньої обробки текстів;

$\nu_4 = \{x_{10}\}$ – відповідає агенту оцінки релевантності й ранжування результатів пошуку;

$U = \{u_j, j = 1,9\}$, де $u_1 = (x_1, x_2)$, $u_2 = (x_2, x_3)$, $u_3 = (x_3, x_4)$, $u_4 = (x_4, x_5)$, $u_5 = (x_5, x_6)$, $u_6 = (x_6, x_7)$, $u_7 = (x_7, x_8)$, $u_8 = (x_8, x_9)$, $u_9 = (x_9, x_{10})$

$E = \{e_1, e_2, e_3\}$, де

$e_1 = (\nu_1, \nu_2)$ - відповідає передачі запиту для виконання пошуку,

$e_2 = (\nu_2, \nu_3)$ - відповідає передачі посилань для витягу текстів і їх обробки,

$e_3 = (v_3, v_4)$ - відповідає передачі посилань і оцінок ваг для оцінки релевантності й ранжування.

Таблиця 3.3 – Умовні позначки елементарних операцій пошуку

Вершина	Позначення	Операція, що моделюється
x_1	UI	Уведення ключового словосполучення
x_2	GIS	Передача ключової комбінації Google
x_3	CURL	Одержання посилань на документи
x_4	SURL	Збереження посилань
x_5	RD	Витяг документів за посиланням
x_6	DT	Лексемізація тексту документа
x_7	DF	Видалення нейтральних слів
x_8	DS	Виділення основ слів у тексті документа
x_9	TW	Обчислення ваг термінів у документі
x_{10}	OE	Оцінка релевантності й ранжування документів

Отриманий нечіткий метаграф одномовної мультиагентної системи інформаційного пошуку показано на рисунку 3.5

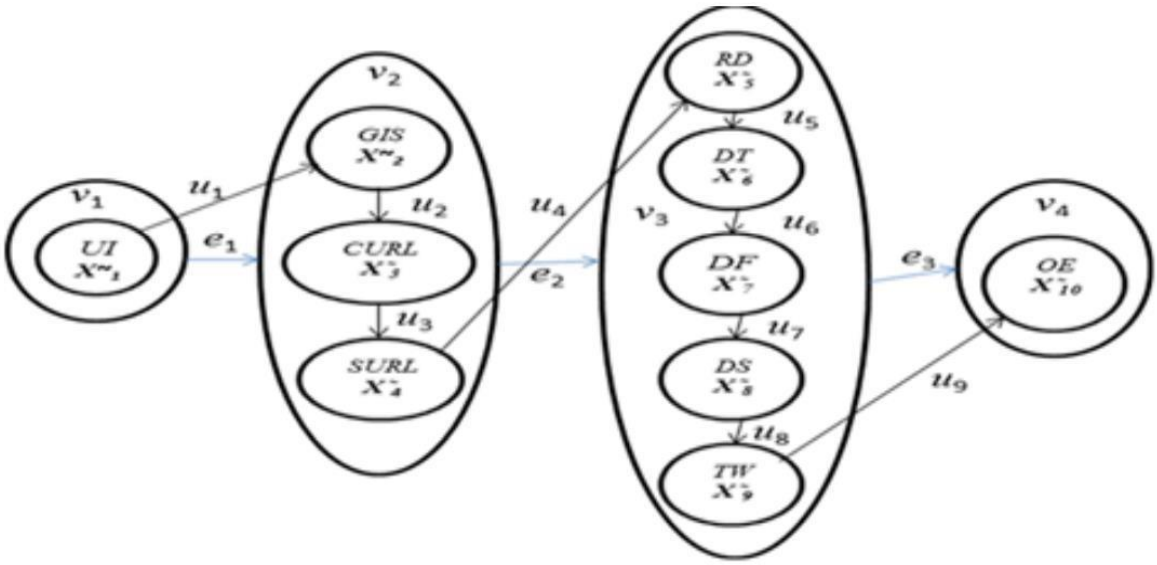


Рисунок 3.5 – Модель процесу інформаційного пошуку для однієї мови у вигляді нечіткого метаграфа

Мультиагентная система, що побудована по зазначеній моделі, включає систему ухвалення рішення про ступінь відповідності документа запиту.

2.7 Розробка моделі багатомовного інформаційного пошуку

Модель автоматичного БІП, реалізованого на мультиагентній платформі будуємться за аналогією з одномовним пошуком. Нечіткий метаграф, що представляє собою модель процесу інформаційного пошуку на трьох мовах, описується так само, як і модель одномовного пошуку:

$$S = (<X, M>, V, U, E),$$

де $X = \{x_i, i = 1,23\}$ – множина вершин, відповідних до множини базових операцій пошуку на трьох мовах;

M – множина функцій що належить вершинам метаграфа;

$V = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7\}$ – множина метавершин, відповідних до підмножин операцій, виконуваних агентами:

інтерфейсним агентом – $v_1 = \{ x_1\}$;

агентом-перекладачем – $v_2 = \{ x_2, x_3, x_4\}$;

пошуковим агентом – $v_3 = \{ x_5, x_6, x_7\}$;

агентом витягу та обробки текстів англійською мовою – $v_4 = \{x_8, x_9, x_{10}, x_{11}, x_{12}\}$;

агентом витягу та обробки текстів російською мовою – $v_5 = \{x_{13}, x_{14}, x_{15}, x_{16}, x_{17}\}$;

агентом витягу та обробки текстів арабською мовою – $v_6 = \{x_{18}, x_{19}, x_{20}, x_{21}, x_{22}\}$;

агентом оцінки релевантності й ранжування результатів пошуку – $v_7 = \{ x_{23}\}$;

$U = \{u_j, j =1,24\}$ – множина дуг-зв’язків між операціями

$u_1 = (x_1, x_2)$, $u_2 = (x_1, x_3)$, $u_3 = (x_1, x_4)$, $u_4 = (x_2, x_5)$, $u_5 = (x_3, x_6)$, $u_6 = (x_4, x_7)$, $u_7 = (x_5, x_8)$, $u_8 = (x_8, x_9)$, $u_9 = (x_9, x_{10})$, $u_{10} = (x_{10}, x_{11})$, $u_{11} = (x_{11}, x_{12})$, $u_{12} = (x_6, x_{13})$, $u_{13} = (x_{13}, x_{14})$, $u_{14} = (x_{14}, x_{15})$, $u_{15} = (x_{15}, x_{16})$, $u_{16} = (x_{16}, x_{17})$, $u_{17} = (x_7, x_{18})$, $u_{18} = (x_{18}, x_{19})$, $u_{19} = (x_{19}, x_{20})$, $u_{20} = (x_{20}, x_{21})$, $u_{21} = (x_{21}, x_{22})$, $u_{22} = (x_{12}, x_{23})$, $u_{23} = (x_{17}, x_{23})$, $u_{24} = (x_{22}, x_{23})$

E = {e_1, e_1, e_1, e_1, e_1, e_1, e_1, e_8} – множина метадуг – множина повідомлень, переданих між агентами:

$e_1 = (v_1, v_2)$, $e_2 = (v_2, v_3)$, $e_3 = (v_3, v_4)$, $e_4 = (v_3, v_5)$, $e_5 = (v_3, v_6)$, $e_6 = (v_4, v_7)$, $e_7 = (v_5, v_7)$, $e_8 = (v_6, v_7)$,

Модель процесу БІП у вигляді нечіткого метаграфа показано на рис. 3.6. При цьому RU – операції, що відносяться до російської мови, AR – до арабської мови й EN – до англійської мови. Щоб не перевантажувати рисунок, множина дуг *U* на ньому не показані.

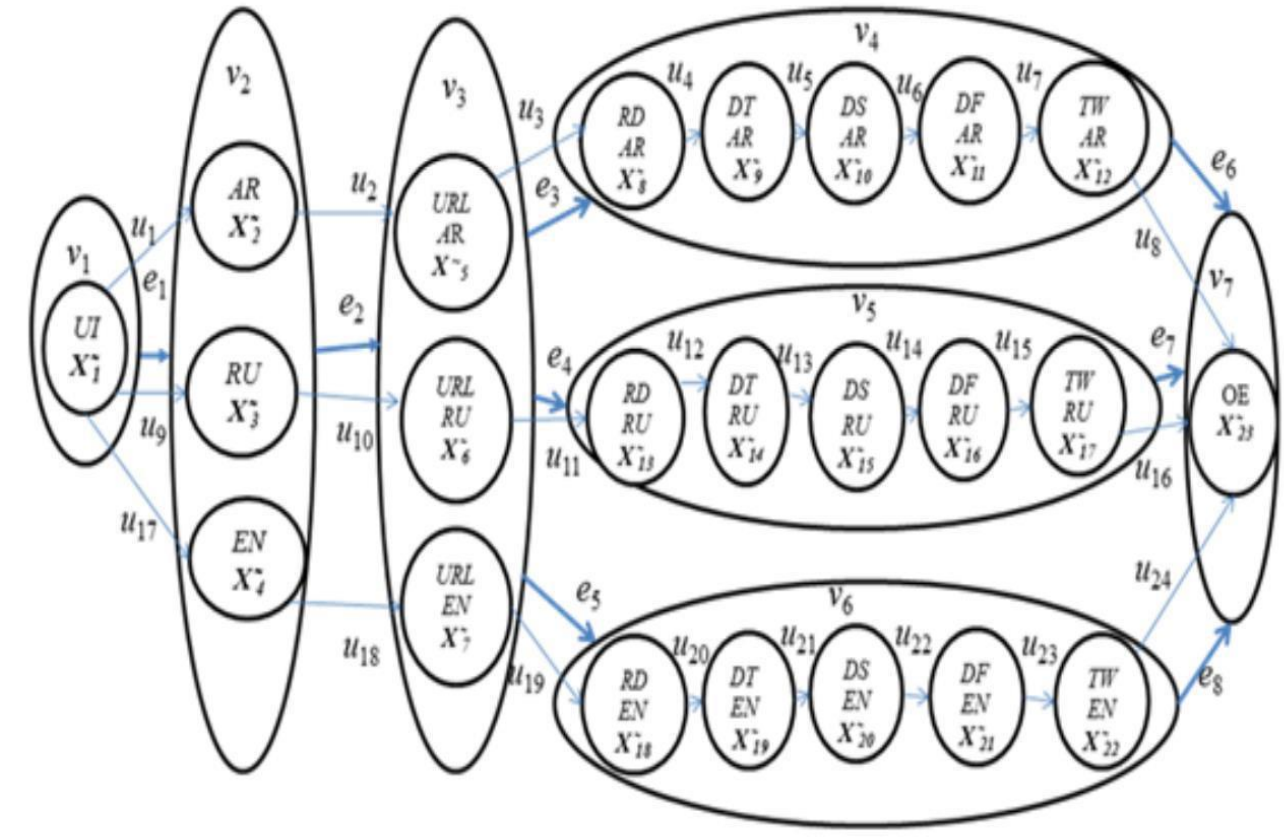


Рисунок 3.6 – Структура БІП для трьох мов (AR – арабська мова, RU – російська мова, EN – англійська мова)

Таблиця 3.4 – Умовні позначення основних операцій пошуку для БІП

Елемент множини	Позначення	Операція, що моделюється
x_1	UI	Введення запиту на одній з мов
x_2, x_3, x_4	AR, RU, EN	Переклад запиту на інші мови
x_5, x_6, x_7	URL AR, RU, EN	Пошук посилань у системі Google
x_8, x_{13}, x_{18}	RD AR, RU, EN	Витяг документів за посиланням
x_9, x_{14}, x_{19}	DT AR, RU, EN	Лексемізація тексту документа
x_{10}, x_{15}, x_{20}	DF AR, RU, EN	Видалення нейтральних слів
x_{11}, x_{16}, x_{21}	DS AR, RU, EN	Виділення основ слів у тексті документа
x_{12}, x_{17}, x_{22}	TW AR, RU, EN	Обчислення ваг термінів у документі
x_{23}	OE	Оцінка релевантності й ранжування документів

Аналогічно можуть бути побудовані моделі БІП, розраховані на більшу кількість мов.

Аналіз існуючих оцінок релевантності документів показав, що в якості вихідних даних для оцінки релевантності документів доцільно використовувати частоту появи терміну в документі, зважену зворотною частотою використання термінів у документах вибірки та нормовану з урахуванням довжини тексту.

3 ОПИС ФУНКЦІОНУВАННЯ РОЗРОБЛЕНОГО ПЗ

3.1 Реалізація алгоритму оцінювання релевантності документів

Вихідними даними для обчислення оцінки релевантності документа служать нормовані ваги термінів у запиті $w(tj,q)$ і документі $w(tj,d)$. Відповідно їх кількість визначається кількістю слів у запиті. Далі виконуються наступні операції.

на основі нечіткого метаграфа виконується налаштування нечітких правил для СНЛВ;

здійснюється фазифікація, при цьому чіткі вхідні дані $w(t,q)$ і $w(t,d)$ перетворюються в лінгвістичні змінні, на основі функцій приналежності, що зберігаються в нечіткій базі знань, у процесі фазифікації функції приналежності, певні для вхідних змінних, застосовуються до їхніх реальних значень, так що для кожного правила може бути визначений умовний ступінь істинності. Нечіткі вираження антецедентів можуть приймати значення приналежності між 0 і 1 ;

здійснюється нечіткий висновок, у процесі якого використання введення нечітких правил перетворить нечіткий внесок у нечіткий висновок. Після оцінки результатів кожного правила ці результати повинні бути об’єднані, щоб одержати кінцевий результат (вихід агрегації);

здійснюється дефазифікація – при цьому використовується метод центроїда. (Нечіткий висновок типу Сугено не має дефазифікації, вихід Сугено є лінійним або постійним, тому при застосуванні цієї СНЛВ дефазифікація не потрібна).

Отримана експертна оцінка релевантності документа далі використовується для ранжування результатів БІП. На рис. 4.1 показаний алгоритм оцінки релевантності документів з використанням СНЛВ.

Для оцінки застосовності систем Мамдані й Сугено для визначення ступеню релевантності знайдених документів були виконані експериментальні дослідження.

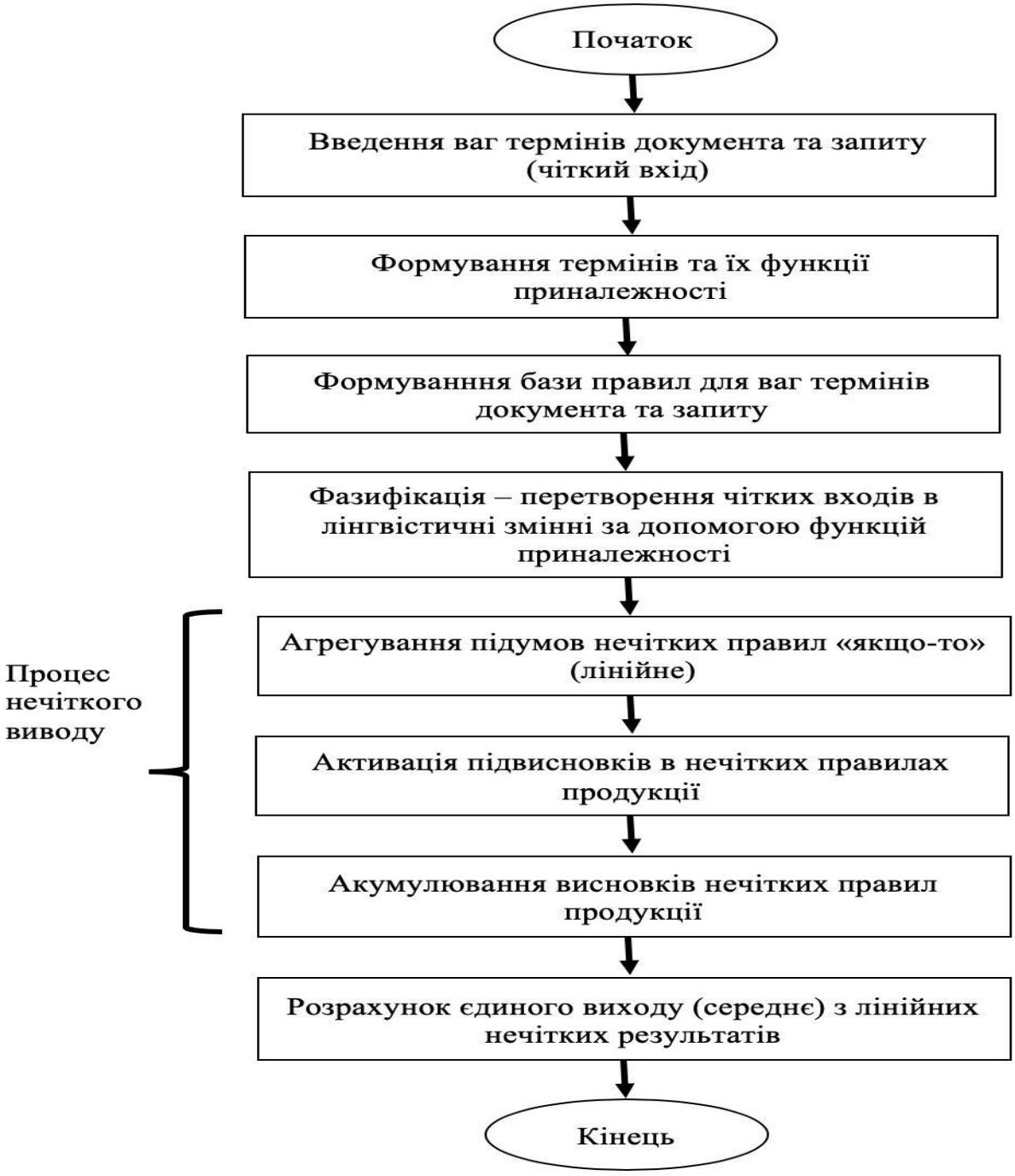


Рисунок 4.1 – Алгоритм експертної оцінки релевантності за допомогою нечіткого логічного висновку

3.2 Експериментальне оцінювання застосовності систем нечіткого висновку

Експеримент проводиться для оцінки ранжування релевантних документів при використанні ключових слів запиту «системи керування автоматичні енергії».

Для оцінки релевантності документів у мультиагентній системі БІП на базі СНЛВ Мамдані й Сугено, з використанням косинусної подоби для лінгвістичних змінних ваги *j*-го терміну в запиті та у документі були використані наступні правила, засновані на коді (див. Додаток А):

Кількість входів СНЛВ Мамдані залежить від кількості термінів у запиті, відповідно для запиту із чотирьох термінів на вході системи 8 змінних: чотири ($w(t_1,q)$, $w(t_2,q)$, $w(t_3,q)$, $w(t_4,q)$, – для ваг термінів документа й чотири $w(t_1,di)$ і $w(t_2,di)$,$w(t_3, di)$, $w(t_4,di)$ – для ваг термінів запиту), як показано на рис. 4.2.

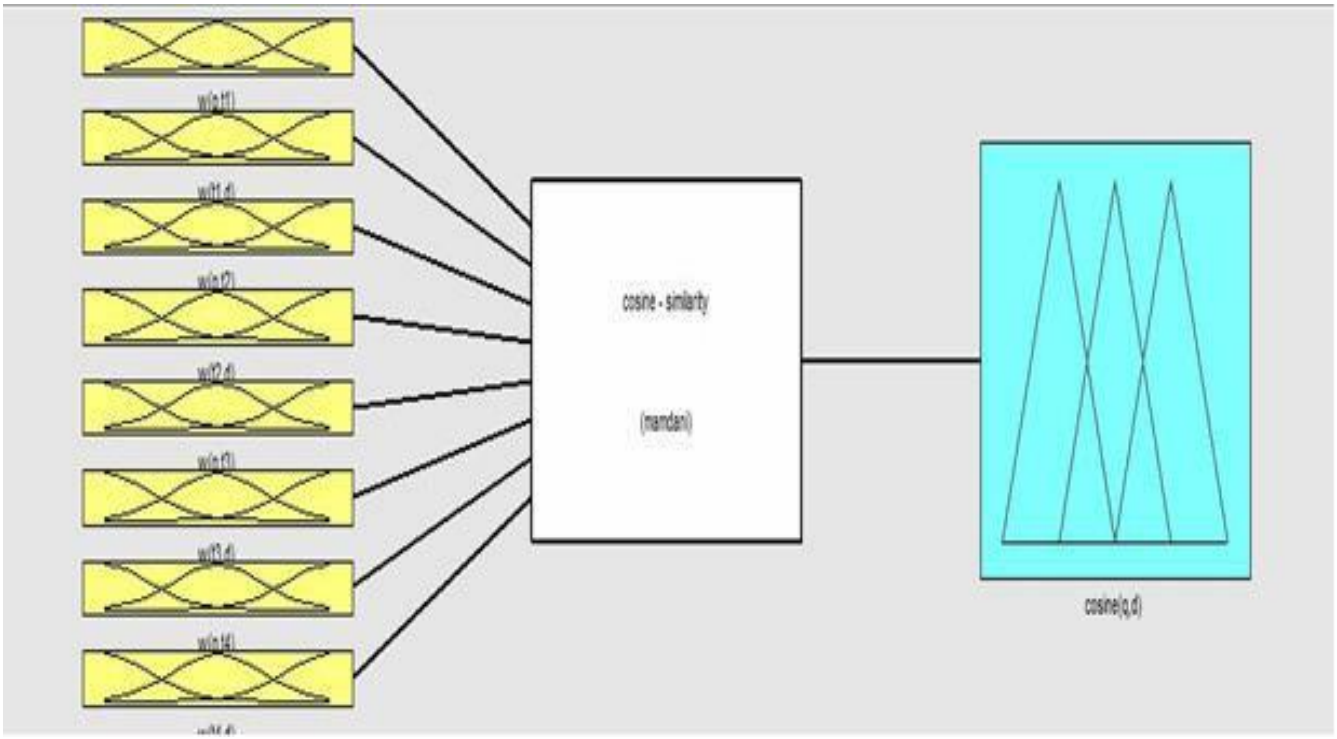


Рисунок 4.2 – Система нечіткого логічного висновку Мамдані

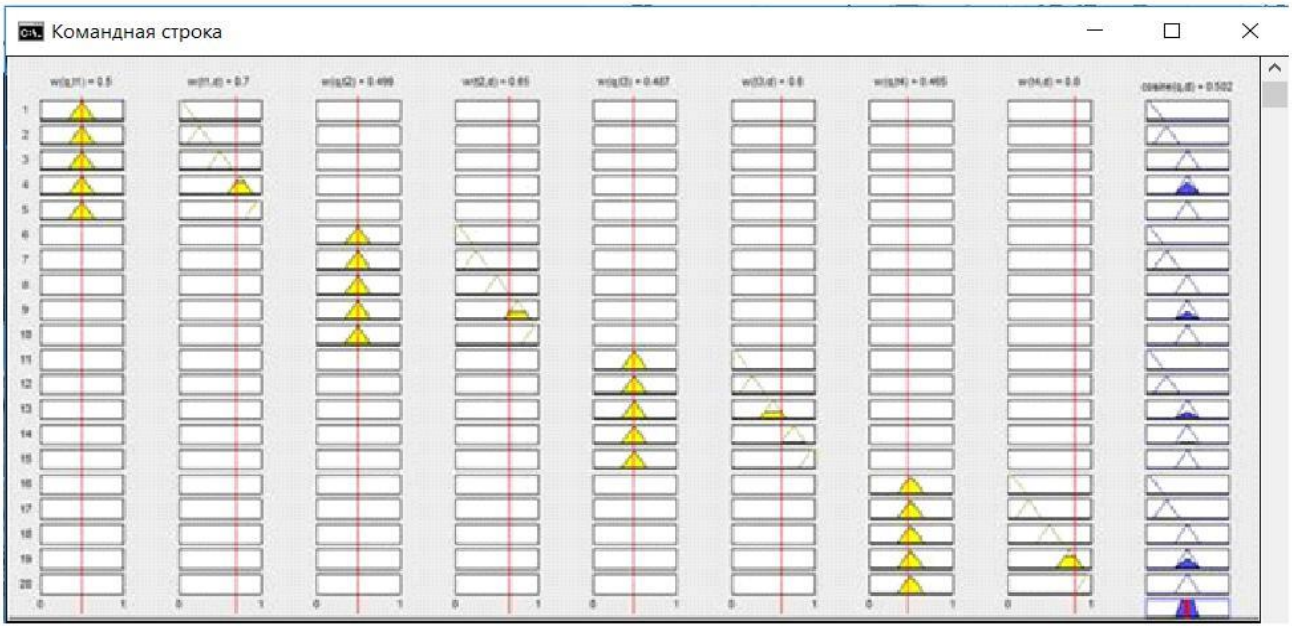
Система має один вихід – бал документа (оцінку релевантності). Кожна з обраних вхідних і вихідних змінних описана множино. з п’яти лінгвістичних значень (дуже низький, низький, середній, високий і дуже високий), визначених трикутною функцією приналежності, у такий спосіб, що дозволяє здійснювати процедуру фазифікації для перетворення обмірюваних числових значень в одне з нечітких значень.

Функції приналежності використовуються для фазифікації й дефазифікації, для розмітки нечітких вхідних значень і лінгвістичних термінів та навпаки. Функція приналежності також застосовується для кількісної характеристики лінгвістичного терма.

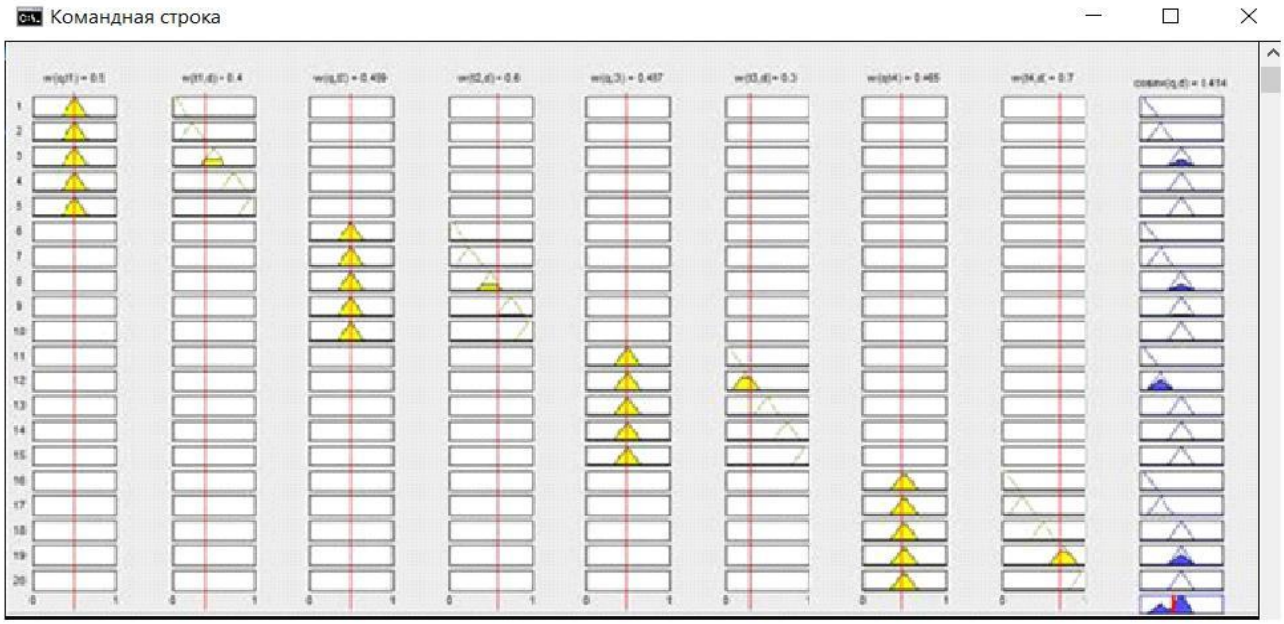
В експерименті вхід обробляє НСЛВ типу Мамдани, використовуючи трикутну функцію приналежності й правила, описані вище. Вхід для процесу дефазифікації полягає в агрегації виходу нечіткої множини (суми після застосування правил), і одержання для вихідної множини результату (зважене середнє, як показано на рис. 4.3), Документ на рисунку 3.5 *a*, мав найбільший бал (значення центроїда – 0.502), документ на рисунку 3.5 *b*, мав найменший бал (зважене середнє – 0.414).

Початкові кроки й налаштування СНЛВ Сугено аналогічні налаштуванням СНЛВ типу Мамдани. У цьому випадку на вході також вісім значень (чотири ваги термінів документа й чотири ваги термінів запиту), як показано на рис. 4.4, і генерує один вихід, який показує міру подоби.

Кожна з обраних вхідних змінних описана множиною з п’яти нечітких лінгвістичних значень, визначених трикутною функцією приналежності, як для СНЛВ типу Мамдані. На відміну від діапазону вихідних значень СНЛВ Мамдані, діапазон виходів СНЛВ Сугено – значення, які перебувають між 0 і 1.



a



b

Рисунок 4.3 – Результати експерименту з використанням СНЛВ Мамдані:

a – оцінка для найбільш відповідного документа;

b – оцінка для найменш відповідного документа

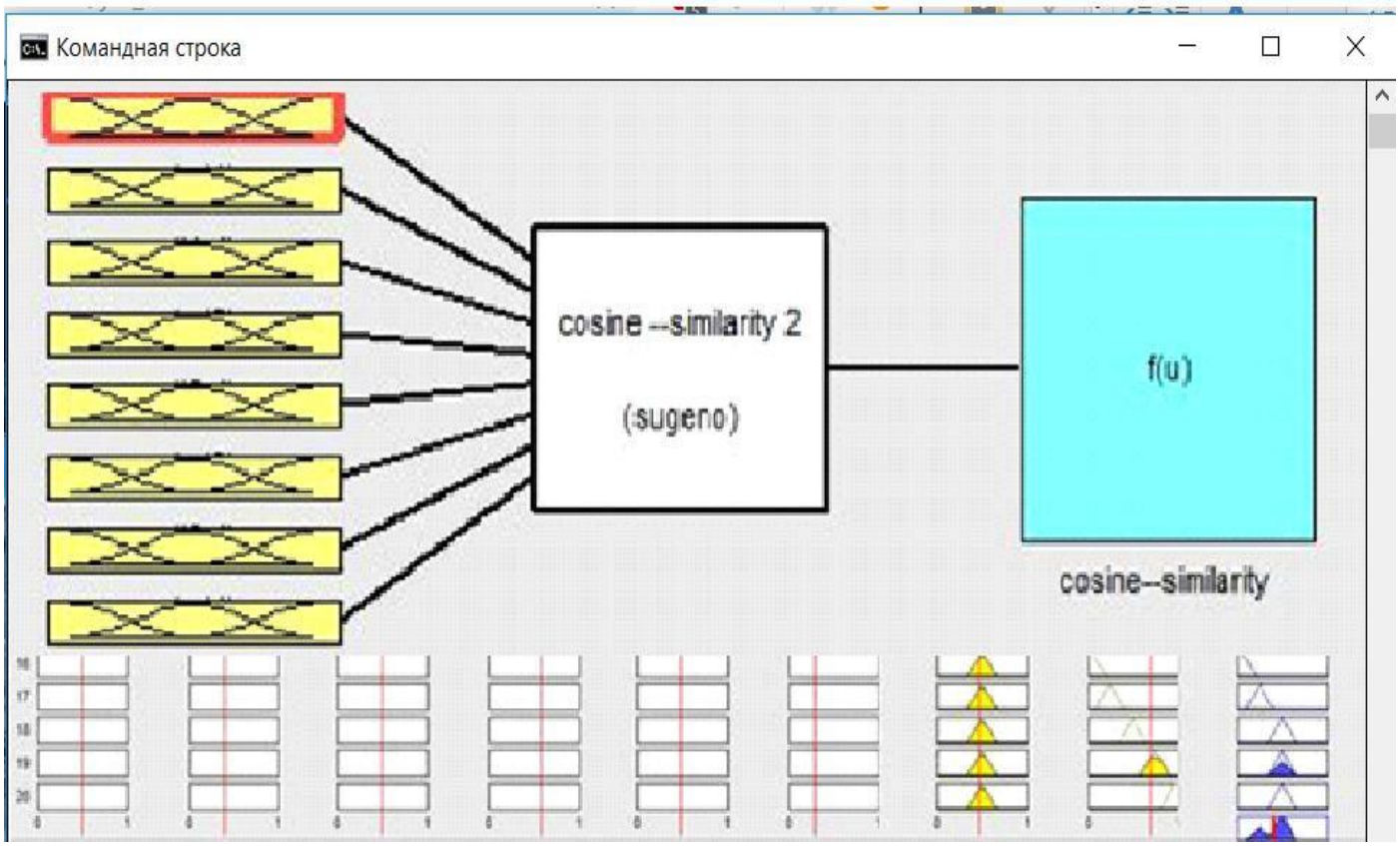


Рисунок 4.4 – Система нечіткого логічного висновку Сугено

База правил для СНЛВ типу Сугено така ж, як і для СНЛВ типу Мамдани. В експерименті вхідні дані оброблені НСЛВ типу Сугено з використанням правил. Множина виходів була представлена одним числом (зваженим середнім), як показано на рисунках 4.5, *а* й 4.5, *б*. Документ на рисунку 4.5,*а* мав найбільше значення (зважене середнє – 0.8), документ на рис. 4.5,*б* мав найменший бал (зважене середнє – 0.7).

Результати експериментів показали, що, незважаючи на відмінність одержуваних балів, для релевантних документів СНЛВ Мамдані й Сугено працюють аналогічно, тобто порядки ранжування документів по оцінках, отриманих за допомогою обох систем, практично однакові. Отже, для оцінки бала ранжування документа можна використовувати обидва типи СНЛВ.

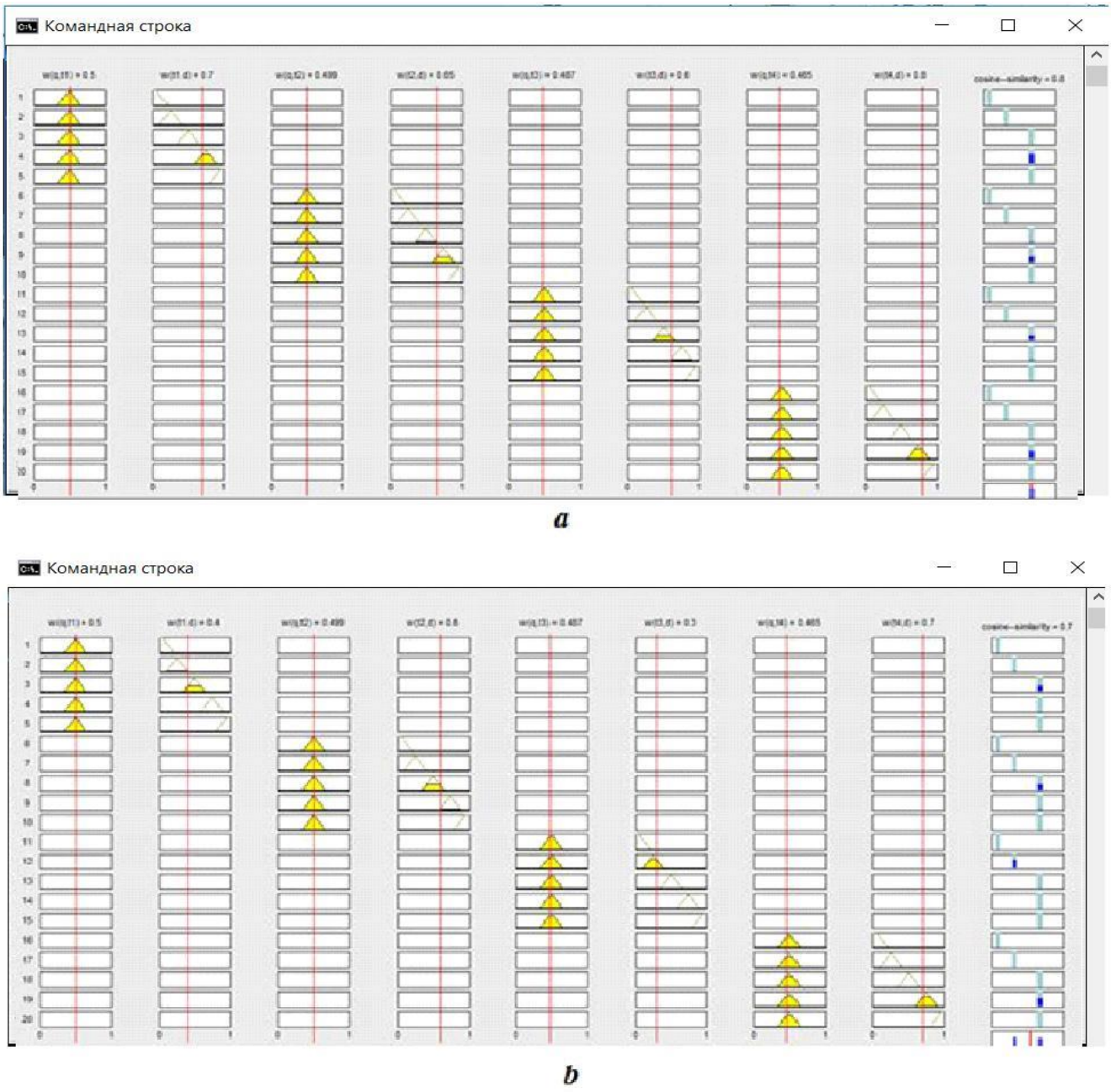


Рисунок 4.5 – Результати експерименту з використанням НСЛВ Сугено *a* – оцінка для найбільш відповідного документа; *b* – оцінка для найменш відповідного документа

Однак СНЛВ Сугено дозволяє одержати оцінку в 50-60 разів швидше, отже, використання системи нечіткого висновку Сугено є оптимальним. У такий спосіб обидві СНЛВ і Мамдані, і Сугено можуть бути використані для оцінки релевантності документів.

Аналогічні результати були отримані і для БІП.

3.3 Розробка ПЗ багатомовного інформаційного пошуку за допомогою мультиагентної системи

Методика виконання БІП включає наступні операції.

Користувач вводить ключове слово на будь-якій мові в користувацькому інтерфейсі.

Виконується переклад даного запиту на інші мови. Якщо введені ключові слова не знайдені в словнику, користувач ще раз вводить ключові слова.

З використанням пошукової машини Google здійснюється пошук по ключових словах у мережі Інтернет. Google повертає посилання, збираючи URL-посилання на доступних йому веб-сайтах. Якщо немає доступних URL-посилань, користувач ще раз вводить ключове слово для одержання URL-посилань, що задовольняють його запиту.

Виконується витяг документів за знайденими посиланнями.

Виконується аналіз текстів на всіх використовуваних мовах з використанням алгоритмів лематизації, видалення стоп-слів і лексемізації.

Обчислюється нормалізовані зважені ваги термінів для запиту і документів.

Розраховуються експертні оцінки релевантності для кожного документа.

Виконується ранжування результатів пошуку за релевантністю знайдених документів.

Дана послідовність представлена у вигляді алгоритму наведена на рис. 4.6.

Для реалізації агентів були використані програмні платформи JADE і Rapid Miner. На даний момент час ведуться роботи в напрямку стандартизації технологій агентів FIPA (Foundation for Intelligent Physical Agents) [13], і реалізації середовищ розробки для побудови мультиагентних систем. FIPA є некомерційною асоціацією компаній і організацій, що займається специфікаціями загальних агентських технологій.

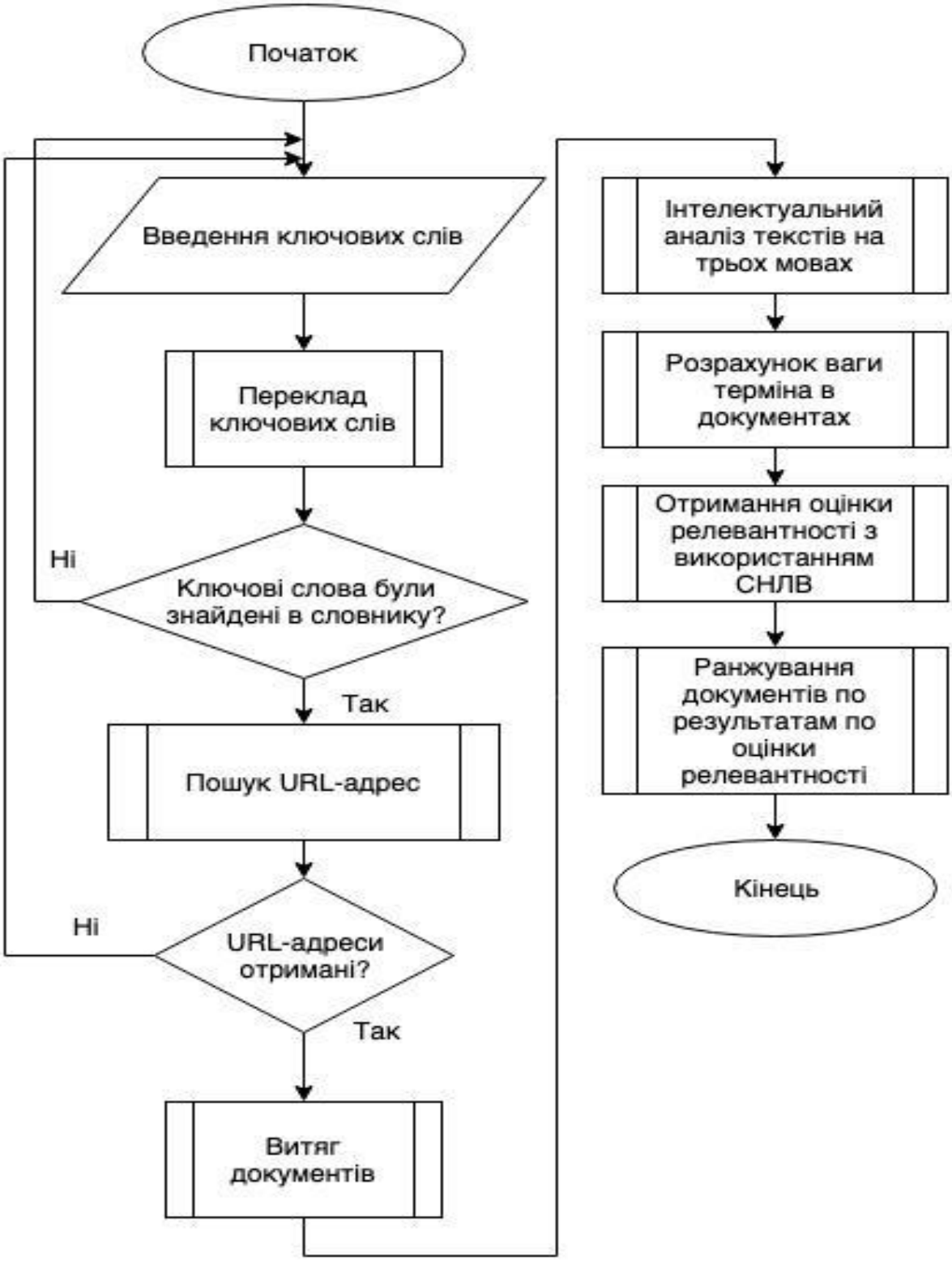


Рисунок 4.6 – Алгоритм багатомовного інформаційного пошуку

Робота зі стандартизації, що здійсньюється FIPA, є напрямком, що дозволяють легко організувати взаємозамінність між агентними системами, тому що FIPA не використовує мову комунікації агентів і вказує ключових агентів, необхідних для керування системою.

Онтологія необхідна для взаємодії між системами й визначає транспортний рівень протоколів.

JADE (Java Agent Development Framework) є програмним забезпеченням для розробки агентських додатків у відповідності зі специфікаціями FIPA для взаємозамінного інтелектуального МАС. FIPA дотримується двох основних правил. Перше полягає в тому, що час, витрачений на досягнення консенсусу й завершення роботи над стандартом не повинен бути тривалим і не повинен гальмувати прогрес, а, навпаки, прискорювати його до того, як почнеться виробництво. Друге полягає в тому, що зовнішня поведінка компонентів системи повинна бути зазначена без зазначень деталей реалізації й внутрішніх архітектур агентів.

Зокрема, внутрішня архітектура JADE є пропрієтарною, навіть якщо вона відповідає інтерфейсам, стандартизованим FIPA. По-перше, було описано еталонну модель агентної платформи, як показано, на рис. 4.7. Зазвичай вони визначають ролі деяких ключових агентів, необхідних для керування платформою, і вказують мову керування контентом і онтологію.

Три ключові ролі були введені в агентську платформу. Система керування агентом є агентом, який здійснює контроль доступу й використання платформи. Він відповідає за аутентифікацію резидентних агентів і керування реєстраціями. Канал комунікації агентів є агентом, який надає шлях для базових контактів між агентами усередині й поза платформою.

Рисунок 4.7 – FIPA еталонна модель агентської платформи

Це базовий метод комунікації, який надає надійну, упорядковану й точну службу розкладу повідомлень. Він повинен підтримувати протокол IIOP (Internet Inter-Orb Protocol) для взаємозамінності між різними агентським платформами. Менеджер каталогів є агентом, що надає сервіс довідників агентам платформи. Помітимо, що відсутні обмеження для реальної технології, використаної при реалізації платформи: платформа на основі електронної пошти, CORBA **(**Common Object Request Broker Architecture), багатопоточні Java-додатки можуть відповідати стандартам FIPA. Стандарт регламентує також мову комунікації агентів ACL (Agent Communication Language). Комунікація між агентами заснована на передачі повідомлень, де агенти здійснюють комунікацію шляхом формулювання й відправлення індивідуальних повідомлень один одному. ACL FIPA стандартизує мову повідомлень шляхом установки кодування, семантики й прагматики повідомлень.

JADE (Java Agent Development Framework) є програмним фреймворком для спрощення розробки додатків агентів у відповідності зі стандартами FIPA про взаємозамінність інтелектуальних мультиагентних систем. Метою JADE є спрощення розробки при гарантуванні відповідності за допомогою всеосяжного

набору системних служб і агентів. FIPA – сумісна агентська платформа, яка включає систему керування агентами, менеджер каталогів і канал комунікації агентів. Усі ці три агенти автоматично активуються під час запуску розподіленої агентської платформи. Агентська платформа може бути розділена на декілька хостів (передбачається, що між ними немає міжмережевого екрана – фаєрвола). Передбачається наявність тільки одного Java-додатка й, таким чином, однієї віртуальної машини, що виконується на кожному хості.

3.4 Архітектура ПЗ

Агенти реалізовані як один Java-потік, і для ефективної комунікації між агентами на хості використовуються Java-події. Паралельні завдання також можуть бути запущені на одному агенті, і JADE планує ці завдання більш ефективним чином, ніж це робить Java віртуальна машина для потоків.

Графічний інтерфейс користувача для керування декількома агентами й агентними платформами з одного агента – можна здійснювати моніторинг і протоколювання активності кожної платформи.

Агентська платформа JADE сумісна зі стандартами FIPA97 і включає всі необхідні агенти для керування платформою. Усі комунікації між агентами здійснюються за допомогою передачі повідомлень, у яких FIPAACL є мовою повідомлень.

Повнофункціональна агентська платформа (АП) потім збирається з декількох агентів-контейнерів, як показано на рисунку 4.8.

Спеціальний контейнер виконує функцію фронтенду, запускає й виконує керуючі агенти, представляючи всю платформу зовнішньому світу.

Дозволено розподіл агентів у комп’ютерній мережі при умові збереження RMI зв’язків між її хостами. Був розроблений спеціальний легкий контейнер для запуску агентів у Веб-браузері

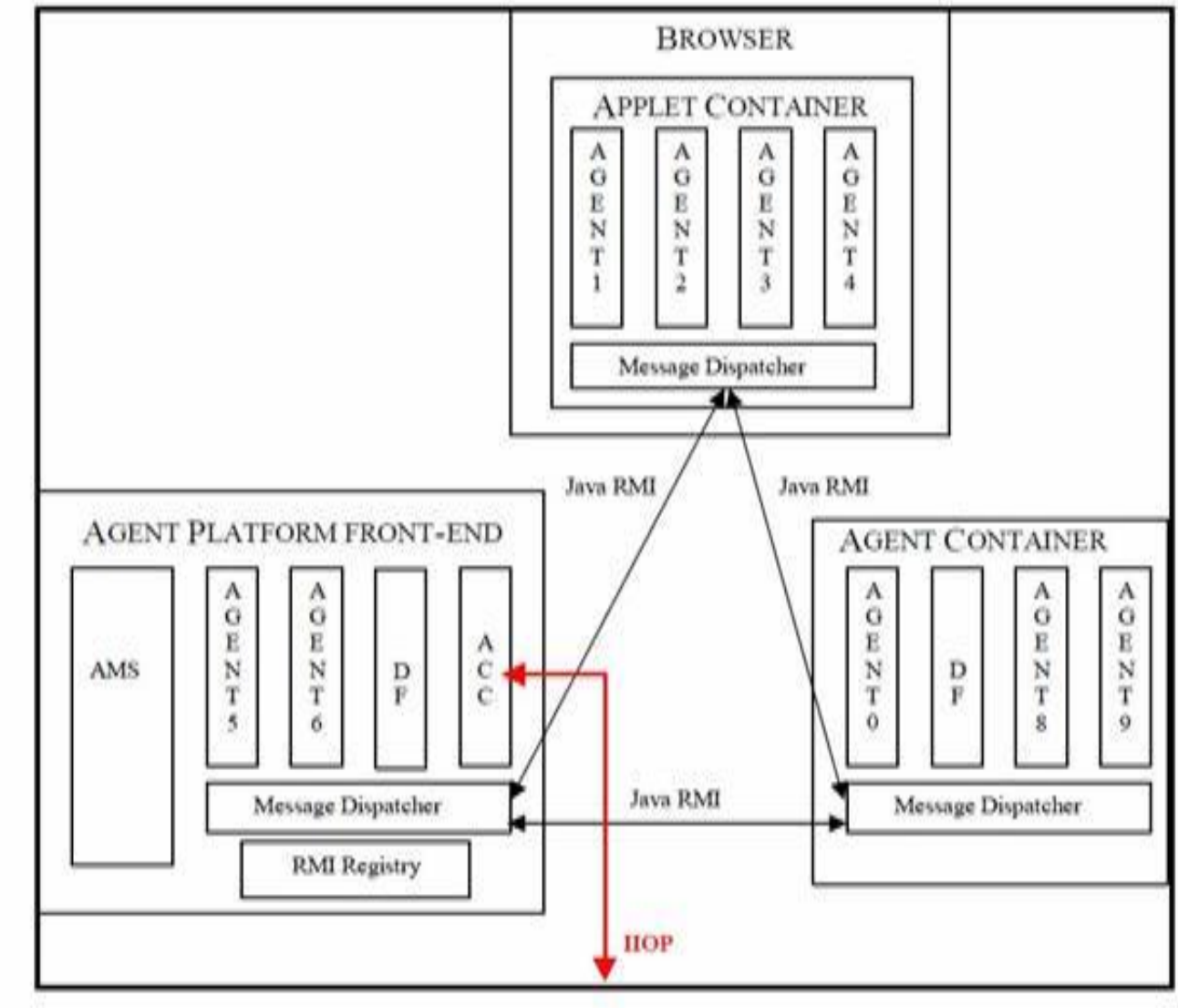


Рисунок 4.8 – Архітектура програмного забезпечення (далі ПО) агентської платформи JADE

.

Подальший розвиток мультиагентної системи відбувається як мультиагентна система пошуку документів згідно із запитом користувача тільки на одній мові (Англійський). У цьому випадку запропонована система використовує JADE платформу як програмне забезпечення для розробки агентів, реалізованих на мові Java. На рис. 4.9. показана JADE платформа, у якій користувач може використовувати графічний інтерфейс менеджера вилучених агентів як користувацький інтерфейс для написання й виконання запитів.

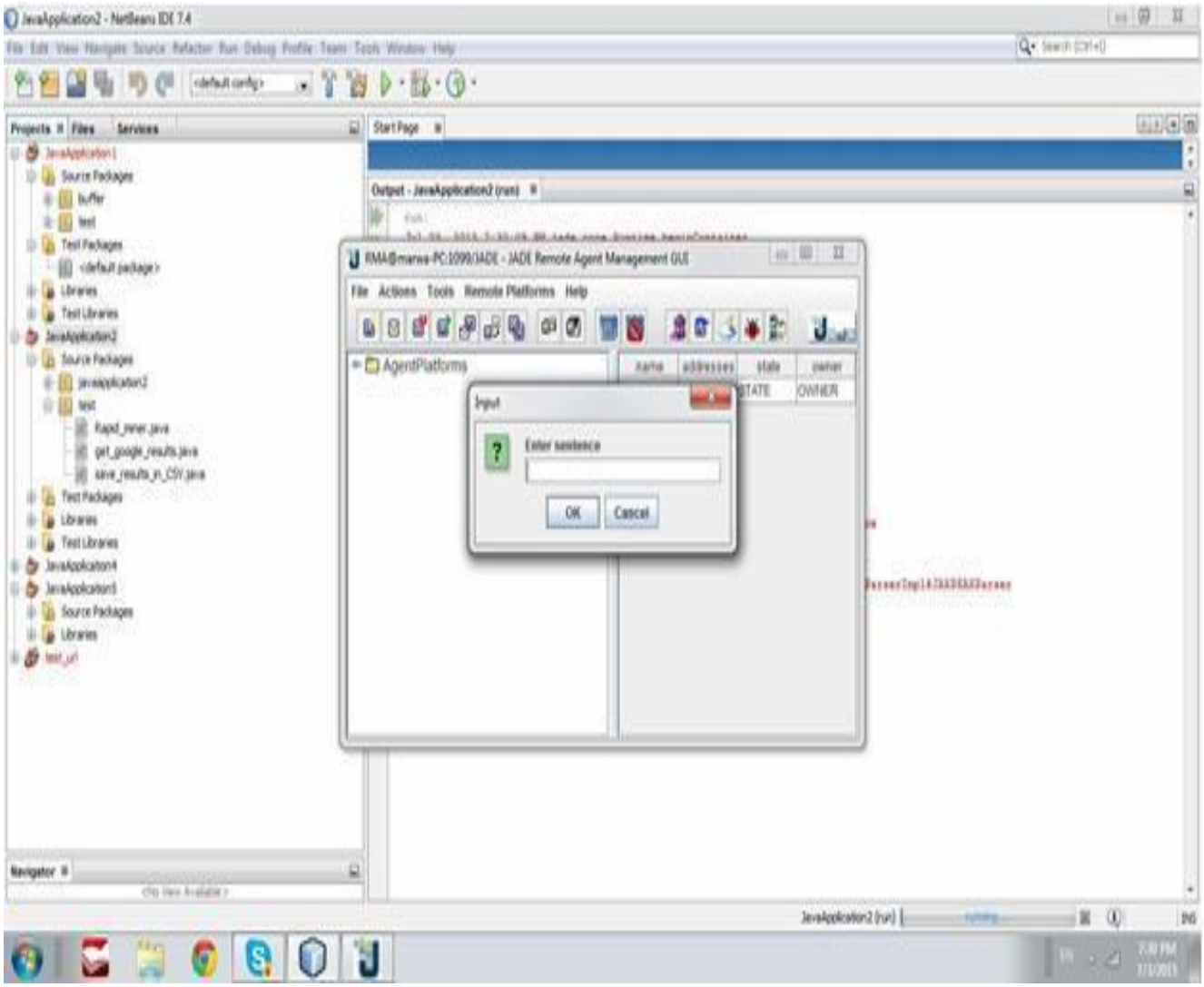


Рисунок 4.9 – Платформа JADE з менеджером вилучених агентів в якості користувацького інтерфейсу

На ри.с 4.10 показане введені користувачем слова запиту(«robotics» «engineering» «distance» «learning»).

На платформі реалізовано запропонований нами мультиагент, а агенти представлені вгорі праворуч, на рис. 4.11.

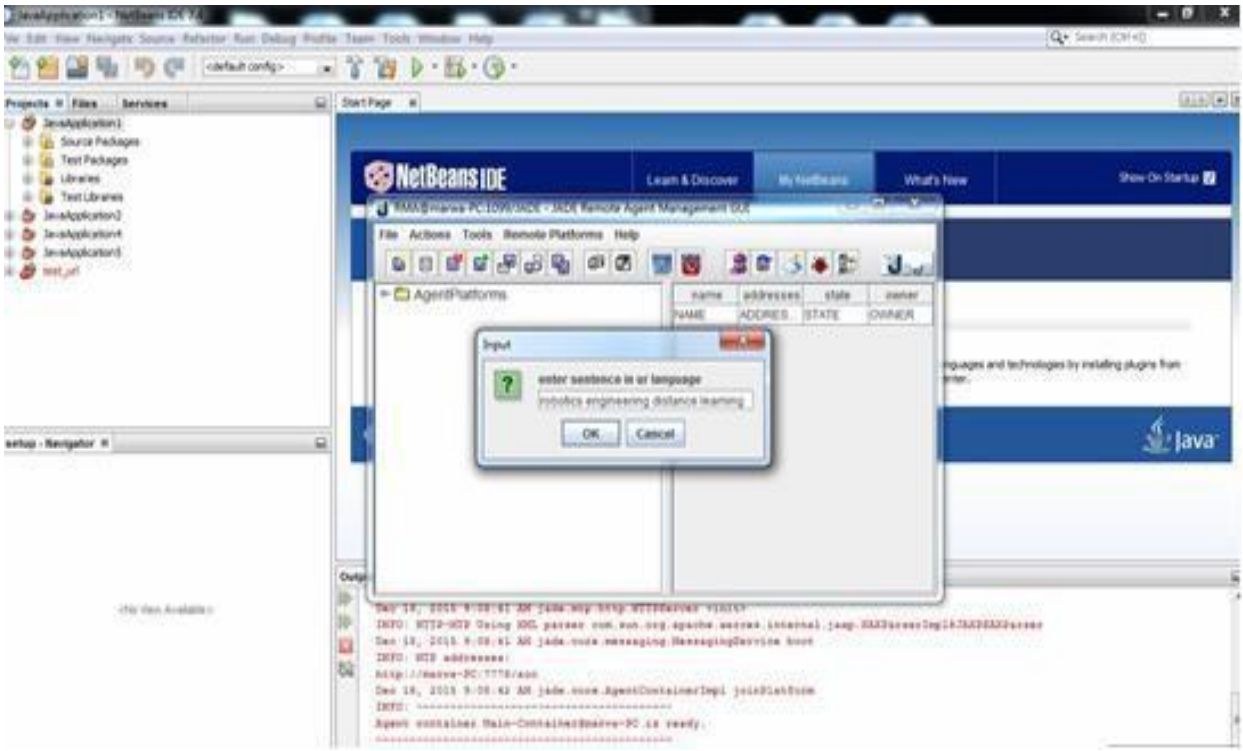


Рисунок 4.10 – Платформа JADE з менеджером вилучених агентів після введення користувачем слова запиту («robotics» «engineering» «distance» «learning»).

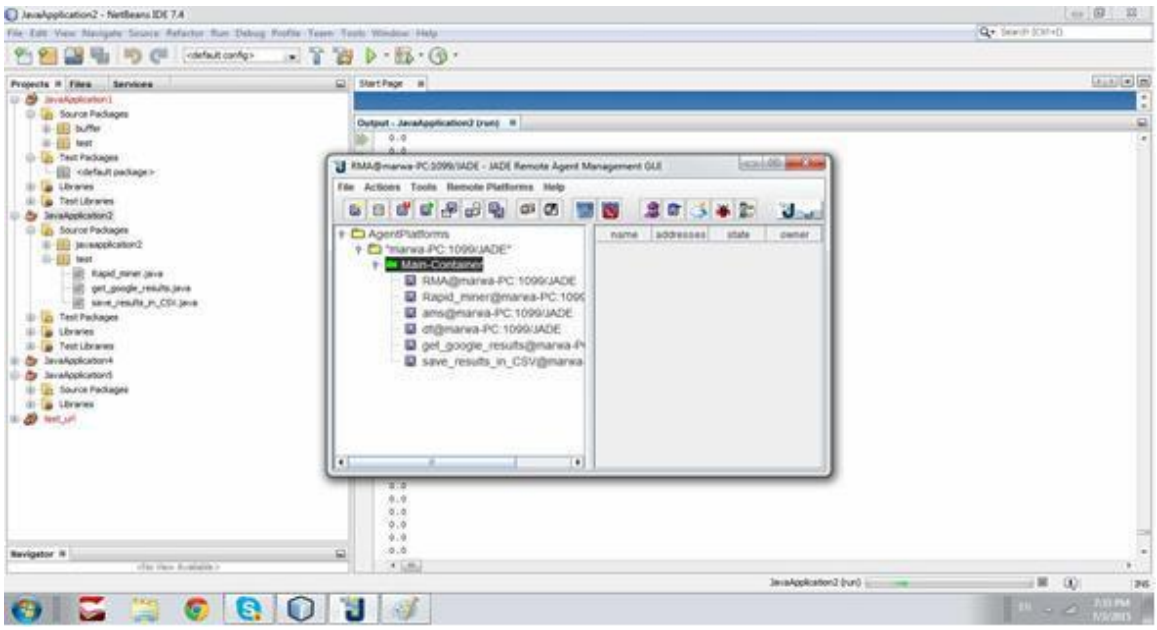


Рисунок 4.11 – Платформа JADE із запропонованим нами мультиагентним контейнером

Розроблена мультиагентна система використовує платформу JADE, яка змодельована як UML-асоціації між класами,. Вона складається з п’яти класів.

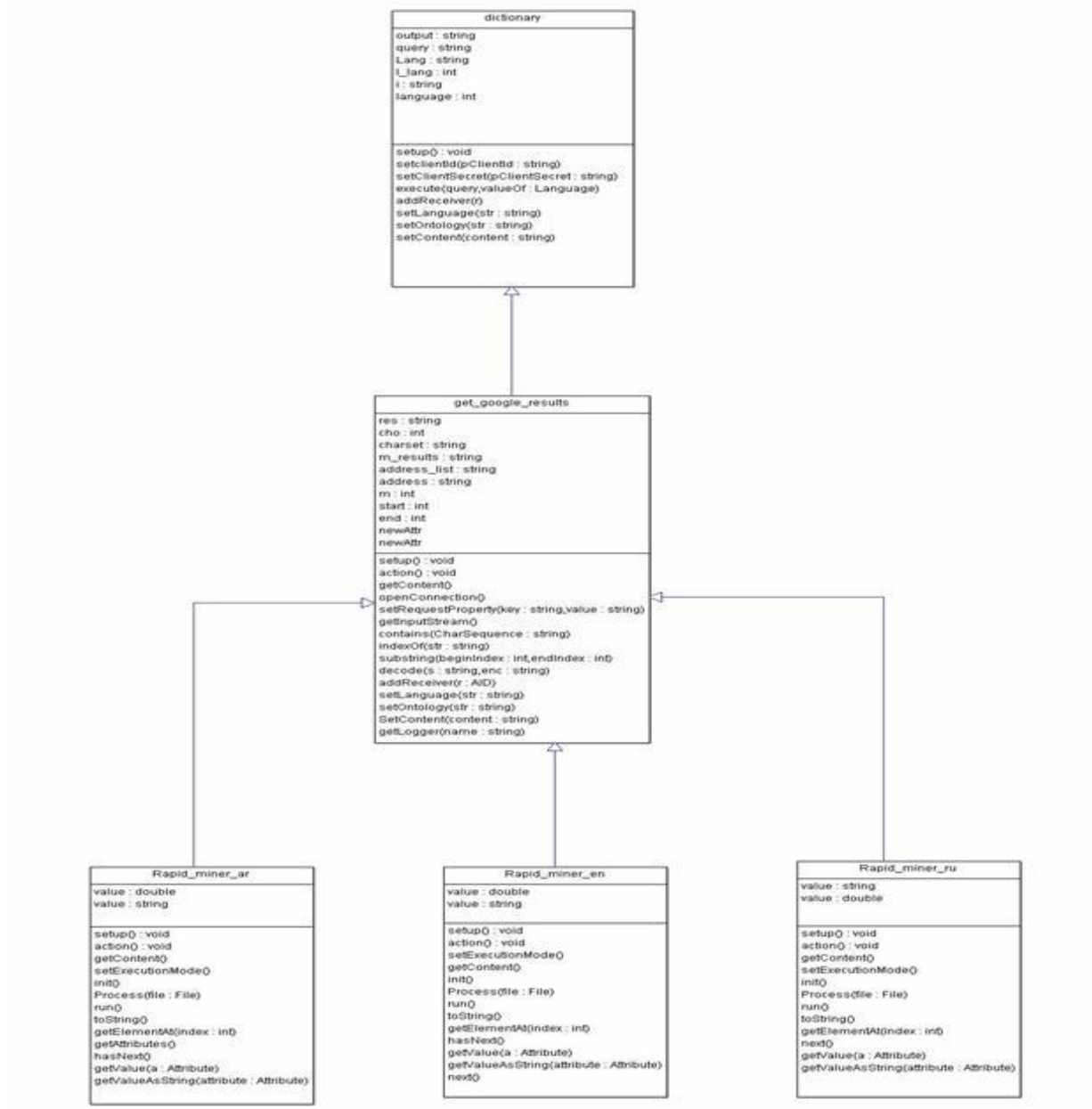


Рисунок 4.12 – Діаграма класів запропонованої системи

На рисунку 4.13 представлена діаграма послідовностей, що ілюструє виконання завдань користувачами і системою

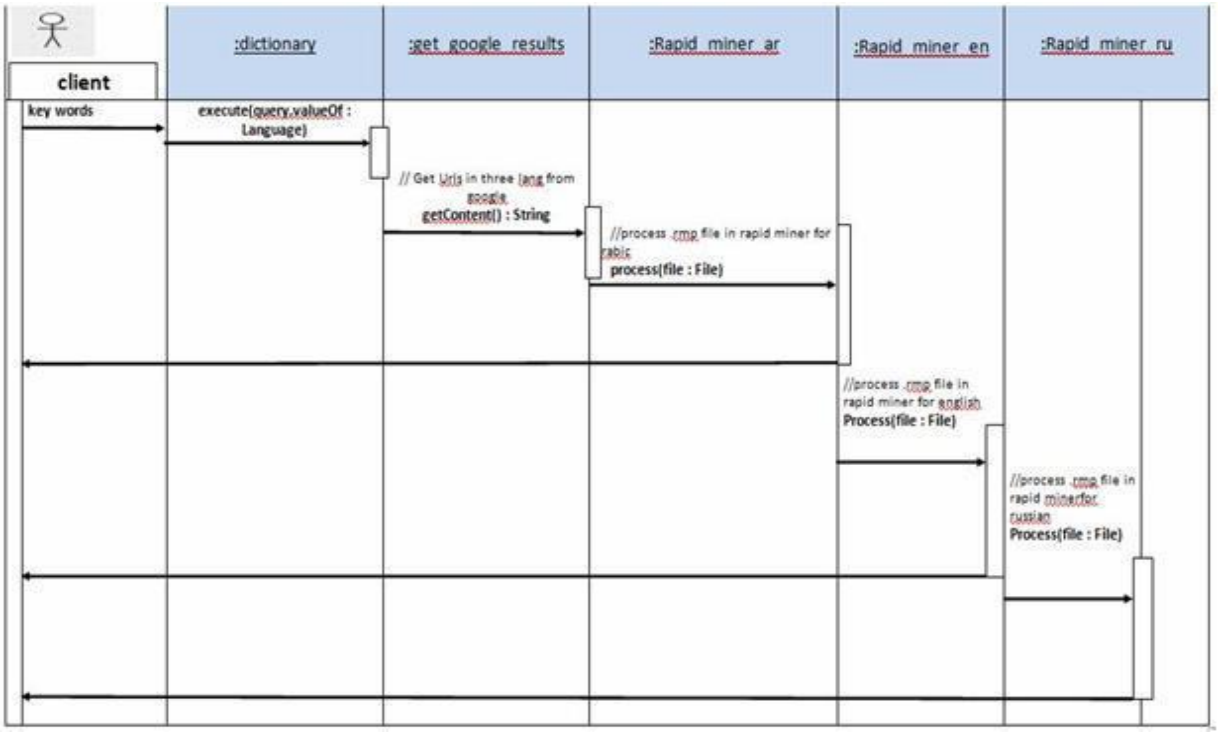


Рисунок 4.13 – Діаграма послідовностей запропонованої системи

Архітектура ПЗ заснована на співіснуванні декількох віртуальних машин (ВМ) Java, а комунікація здійснюється за допомогою Java RMI (Remote Method Invocation) між різними ВМ і сигналізації подій у кожній ВМ. Кожна ВМ являє собою найпростіший контейнер агентів, який надає повне середовище для запуску й виконання агентів, що дозволяє декільком агентам одночасно запускатися на одному і тому ж самому хості. В цілому – ця архітектура дозволяє запускатися декільком ВМ на одному хості; однак цього робити не рекомендується через збільшення витрат ресурсів і відсутності яких-небудь переваг. Кожний агент-контейнер є багатопотоковим середовищем виконання, що складається з одного потоку для кожного агента й системи потоків, породжених RMI (runtime system), для диспетчеризації повідомлень.

3.5 Реалізація експериментальної мультиагентної системи для багатомовного інформаційного пошуку

Проведено застосування мультиагентної системі пошуку документів за запитом користувача на трьох мовах: англійською, російською й арабською. Запропонована система також використовує платформу JADE як програмне забезпечення для розробки агентів, реалізованих на Java. На рис. 5.1 показана платформа JADE, у якій користувач може використовувати графічний інтерфейс менеджера вилучених агентів в якості користувацького інтерфейсу для формування та здійснення запиту.

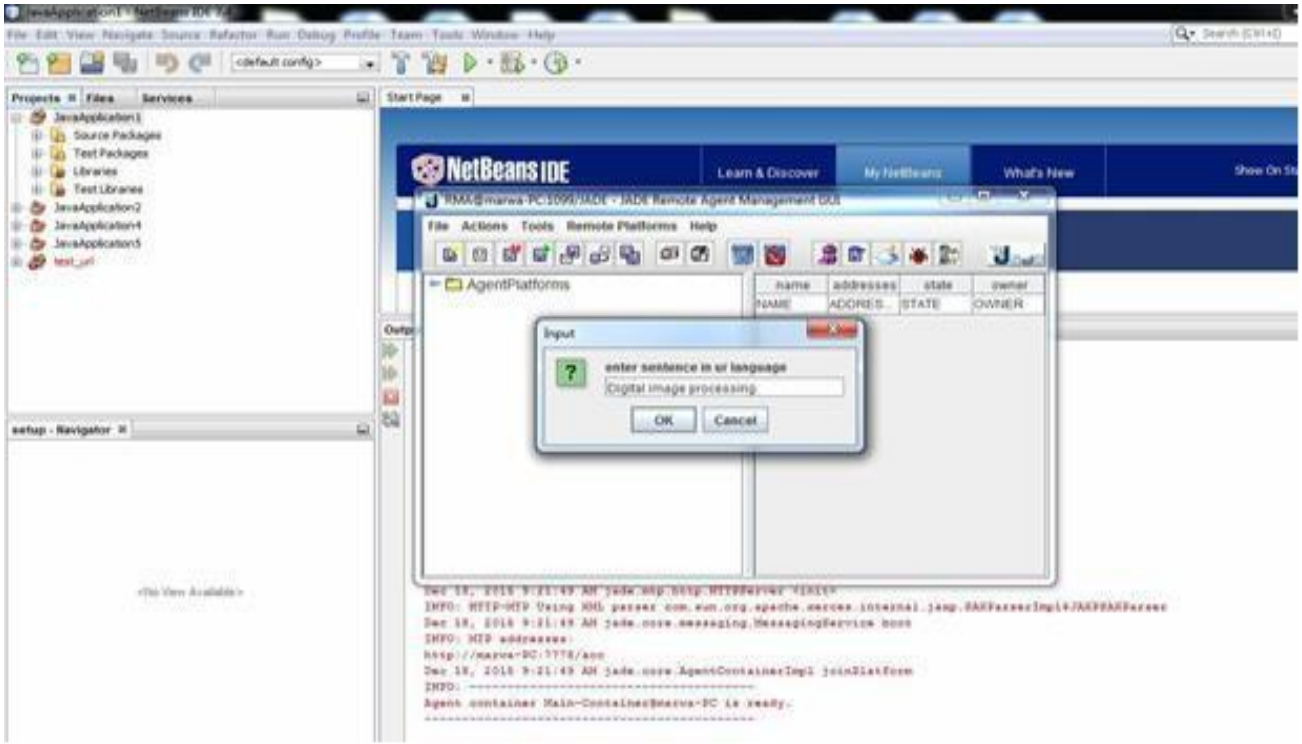


Рисунок 5.1 – Платформа JADE з менеджером запропонованих агентів в якості користувальницького інтерфейсу

Запропонований нами мультиагент реалізований на JADE платформі. Агенти показані вгорі, на рис. 5.2. Вихід агента-перекладача, коли користувач вводить слова запиту «Digital image processing», представлено у центрі сторінки, на рис. 5.3.

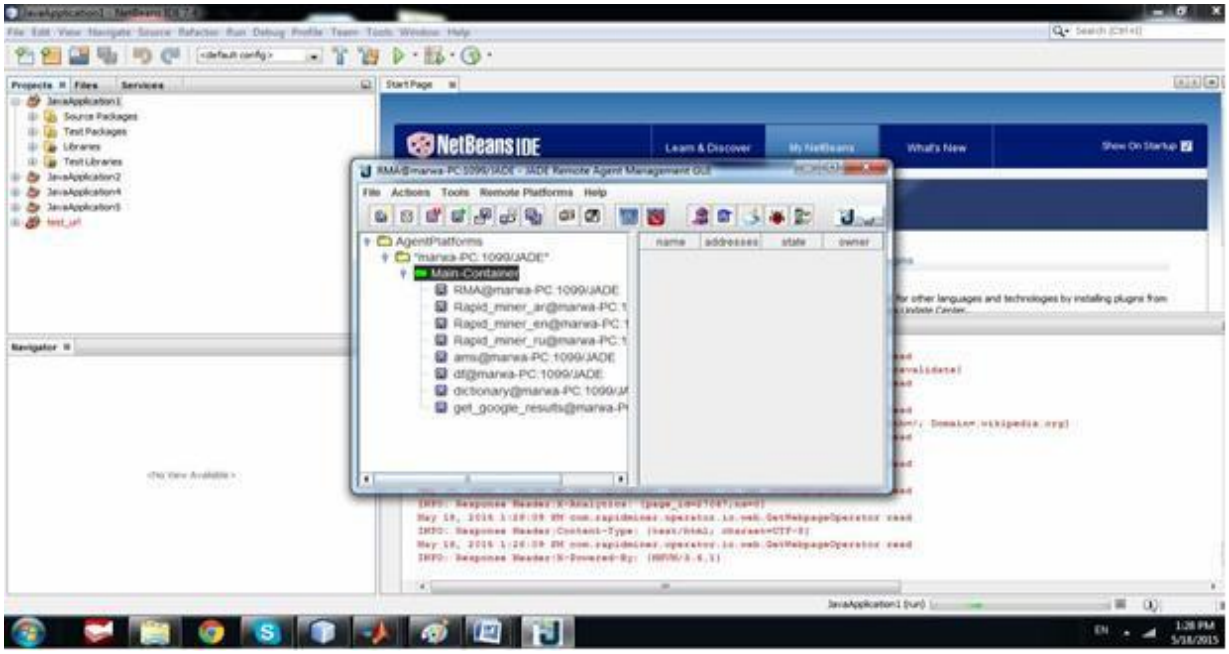


Рисунок 5.2 – Платформа JADE з мультиагентами

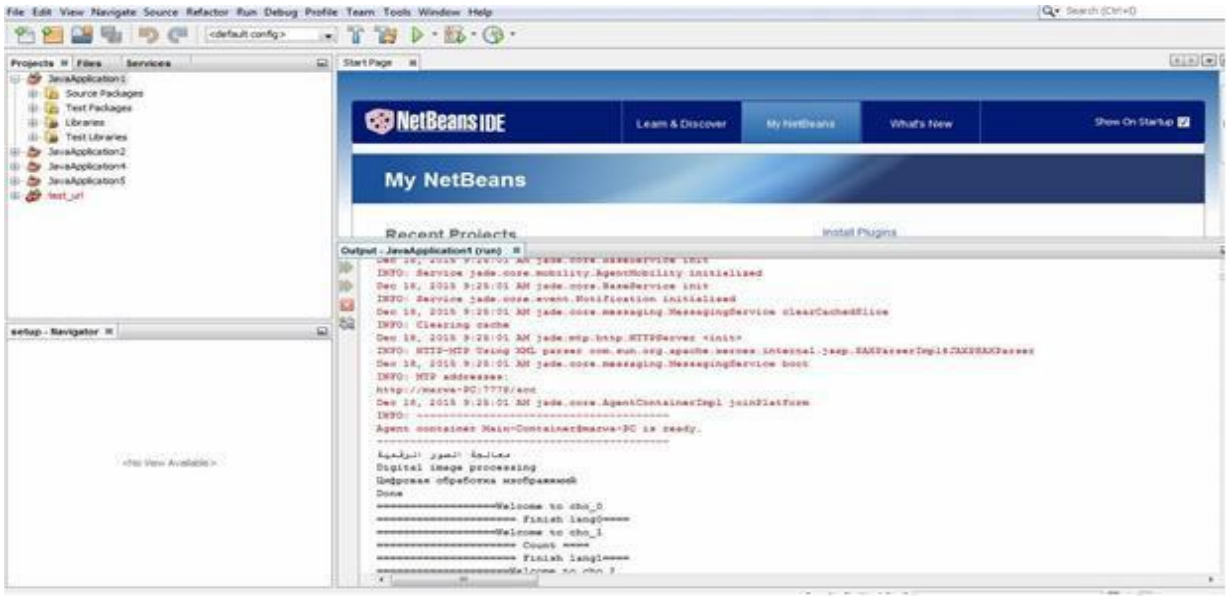


Рисунок 5.3 – Платформа JADE з виводов агента-перекладача для користувацького запиту «Digital image processing»

5.2 Ранжування результатів БІП з використанням СНЛВ Сугено

Проведено оцінки релевантності результатів БІП, виконаних з використанням СНЛВ Сугено. В якості входів використовувалися значення ваг термінів $w(t_1,q)$, $w(t_2,q)$, $w(t_3,q)$, $w(t_4,q)$, $w(t_1,di)$ і $w(t_2,di)$,$w(t_3, di)$, $w(t_4,di)$ для документів і запитів англійською та російською мовами. Також використовувалися правила й функції приналежності ЛЗ «Вага терміну». На рис. 5.4 показане використання нечітких лінгвістичних правил для представлення ваг термінів у вигляді косинусних балів для ранжування документів за запитом «робототехніка інженерного дистанційного навчання». На рис. 5.4,*а* показане найменше значення балу (середнє зважене 0,685), а на рис. 5.4,*б* – найбільше значення балу (середнє зважене 0.8).

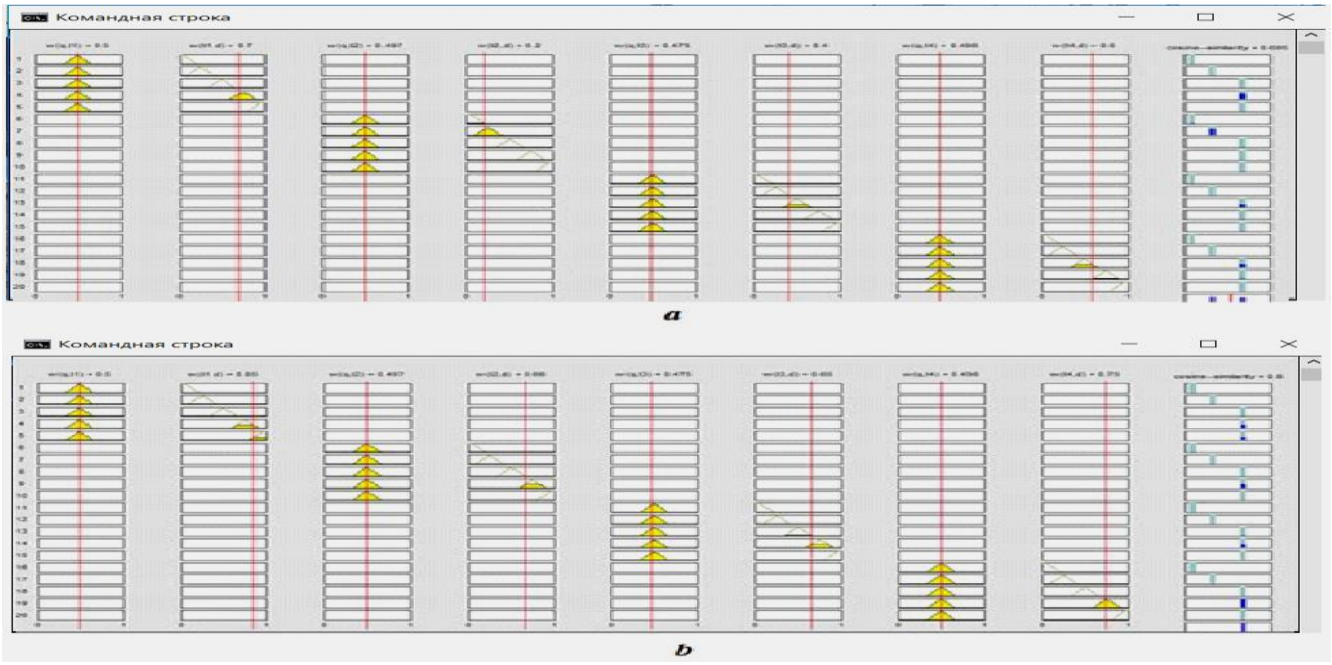


Рисунок 5.4 – Результати оцінки релевантності документів за допомогою системи Сугено для російської мови

3.6 Оцінка якості запропонованої мультиагентної системи

У відповідності з розробленою архітектурою система включає сім агентів: інтерфейсного агента, агента-перекладача, пошукового агента, трьох агентів обробки текстів на різних мовах для одержання ваг термінів і агента оцінки релевантності й ранжування. Пошуковий агент формує необхідні дані для передачі запиту і його параметрів системі Google для пошуку англійською, російською і зберігає результати пошуку для подальшої обробки. Тому основні критерії якості експериментальної пошукової системи, такі як точність, повнота, випадання й ін. визначаються пошуковою системою Google.

Приклади порівняння точності результатів, отриманих для запитів з різною кількістю термінів, представлені: для запитів англійською мовою – у таблицях 5.1 і 5.2;

Таблиця 5.1 – Точність результатів перших N, отриманих до ранжування, для англійської мови

Пошуковий запит	Загальне число сайтів	Кількість обраних сайтів	Кількість релевантних посилань	Кількість нерелевантних посилань	Точність ранжування
Digital image processing	75300000	100	91	9	0.91
Zewail university in egypt	244000	100	88	12	0.88
Robotics engineering distance learning	149000	100	73	17	0.73
Computer engineering technology	162000000	100	89	11	0.89

Таблиця 5.2 – Точність перших N результатів, отриманих після ранжування, для англійської мови

Пошуковий запит	Загальне число сайтів	Кількість обраних сайтів	Кількість релевантних посилань	Кількість нерелевантних посилань	Точність ранжування
Digital image processing	75300000	100	98	2	0.98
Zewail university in egypt	244000	100	94	6	0.94
Robotics engineering distance learning	149000	100	85	15	0.85
Computer engineering technology	162000000	100	96	4	0.96

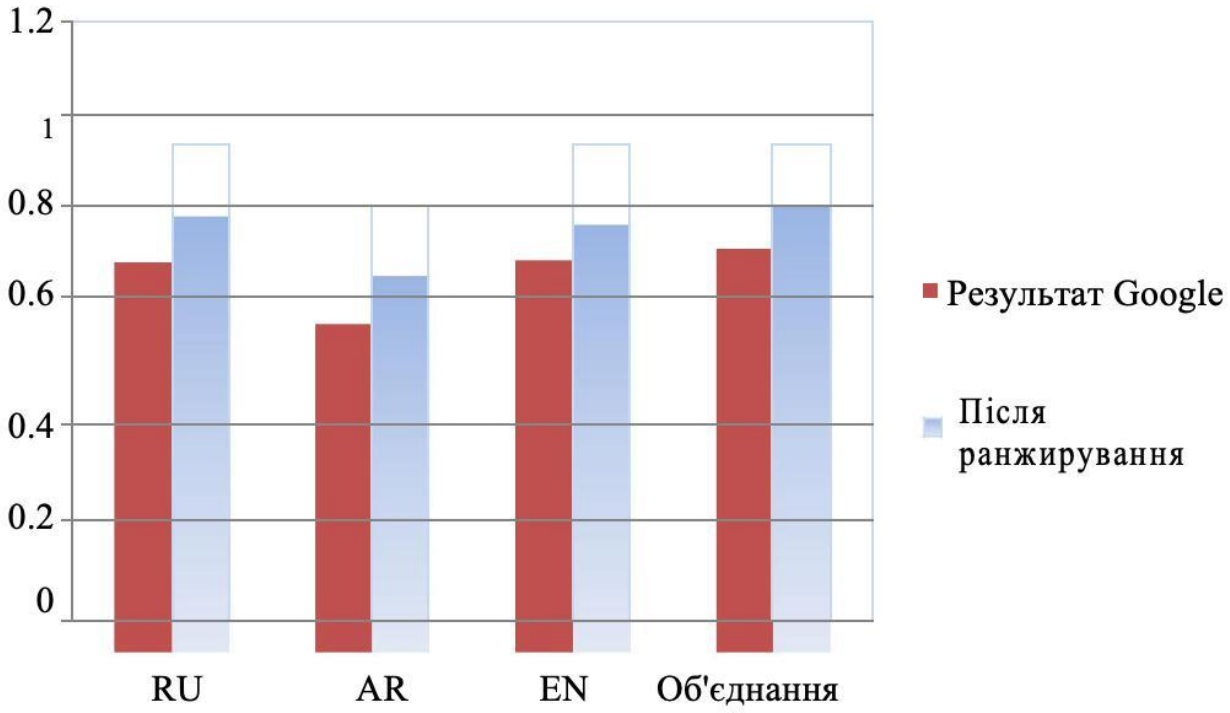


Рисунок 5.5 – Діаграма оцінки точності перших результатів БІП до й після ранжування документів за ступенем релевантності

Підсумовуючи, для запитів, кількість знайдених релевантних посилань за якими перевищує кілька тисяч, ранжування збільшує точність пошуку для перших ста документів на 9±4%.

Запропонована структура системи реалізована у вигляді мультиагентної системи багатомовного пошуку на базі програмних платформ JADE

Експериментально отримано, що при кількості отриманих посилань порядку десятків і сотень тисяч реалізована система збільшує точність результатів пошуку для перших ста документів на 9±4 %.

ВИСНОВКИ

В області прискорення й підвищення якості автоматичного багатомовного інформаційного пошуку застосовується мультиагентний підхід. Дисертаційна робота містить нове рішення актуального наукового завдання – завдання підвищення якості багатомовного інформаційного пошуку.

В роботі отримані наступні результати:

Виконаний аналіз робіт в області кросмовного БІП.

Сформульовані вимоги до якості багатомовного пошуку та системи інформаційного пошуку для декількох мов. Запропонована формальне представлення, додаткового до традиційно використовуваних, критерію якості БІП на основі поняття релевантності перших результатів пошуку.

На основі проведеного аналізу запропонована модель одно- і багатомовного ІП у вигляді нечіткого метаграфа, яка враховує неоднозначність результатів окремих операцій пошуку.

Уведено лінгвістичну змінну «Вага терміну», що дозволяє формалізувати лінгвістичні оцінки його важливості при оцінці релевантності документа. Запропоновано використовувати для ранжування результатів пошуку СНЛВ Мамдани або Сугено. У результаті експериментального дослідження показано, що для одержання оцінок релевантності знайдених документів слід використовувати систему нечіткого логічного висновку Сугено.

Розроблено методику БІП, що включає попередню обробку текстів для уточнення частоти появи термінів у тексті й експертну оцінку релевантності знайдених документів, а також наступне ранжування отриманих результатів.

Розроблено архітектуру мультиагентної системи БІП, визначені функції агентів і формат передачі повідомлень між ними.

Розроблена мультиагентна система багатомовного пошуку для трьох мов: російської, англійської й арабської, яка доведена до рівня дослідницького прототипу. На даний момент вона проходить досвідчену експлуатацію.

Експериментально отримано, що при кількості отриманих посилань порядку десятків і сотень тисяч, реалізована система збільшує точність результатів пошуку для перших ста документів на 9±4 %.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Sagayam R., Srinivasan S., Roshni S. A Survey of Text Mining: Retrieval Extraction and Indexing Techniques. // International Journal of Computer & Electronics Research (IJCER) . – 2012. – Vol. 2. – №5– P. 1443-1444.

2. Hollink V. Monolingual Document Retrieval for European Languages. / V. Hollink, J. Kamps, C. Monz, M .de Rijke M // International Journal of Informatiom Retrieval. – 2016. – Vol. 6. – P. 33-52.

3. Salton G. A Simple Blueprint for Automatic Boolean Query processing. // The Journal of Information Processing and Management. – 1988. – Vol 24. – № 3. – P. 269-280.

4. Fuhr N. Two Models of Retrieval with Probabilistic Indexing. // In Proc. of the 9th Annual Conference on Research and Development in Information Retrieval. – New York . – 1986. – P. 249-257.

5. Lewis D., Sparck-Jones K. Natural Language Processing for Information Retrieval // The Journal of Communication the ACM. – 1996. – Vol 39. – № 1. – P. 92-101. 11. Milstead J. Subject Access Systems : Orlando Academic Press. – 2009.

6. Sharma M., Patel R. A Survey on Information Retrieval Models,Techniques And Applications. // International Journal of Emerging Technology and Advanced Engineering. – 2013. – Vol 3. – № 11. – P. 542-545.

7. Singhal A. Modern information retrieval: a brief overview.// Bulletin of the IEEE Computer Society Technical Committee on Data Engineering. – 2001. – Vol 24. – P. 35-43.

8. Salton G, Wong A, and Yang .C. A Vector Space Model for Automatic Indexing. // The Journal of Communincations the ACM. – 2015. – Vol 18. – № 11. – P. 613-620.

9. Khankasikam K. A comparison of information retrieval models applied to Thai digital library. // In Proc. Of The 2nd International Conference of Computer and Automation Engineering (ICCAE) .— 2018. – Vol 1. – P.335 – 338.

10. Dhavachelvan P., Pothulasujatha. A Review on the Cross and Multilingual. International Journal of Web and Semantic Technology (Ijwest) . –

11. Korra R. Performance evaluation of Multilingual Information Retrieval (MLIR) system over Information Retrieval (IR). // Proceedings of the International Conference system Recent Trends in Information Technology (ICRTIT) . – 2017. —

12. Sorg P. Exploiting Social Semantics for Multilingual: Phd dissertation: Karlsruher institut fur technologie. – 2011.

13. Cross-language information retrieval. [Електронний ресурс]. URL: https://en.wikipedia.org/wiki/Cross-language_information_retrieval (дата звернення: 27.05.2019)..

14. Sujatha P. A Review on Performance Evaluation Measures of Multi Lingual Information Retrieval Systems.// International Journal of Advanced Research in Computer Science and Software Engineering. – 2012. – №8. – Vol. 2. – P.440- 446

15. Sujatha P., Dhavachelvan P. Precision at Korramultilingual Information Retrieval. // International Journal of Computer Applications. – 2011. – Vol 24. – №9. – P. 40-43.

16. Zhuhadar L., Nasraoui O., Wyatt R ., Romero E. Multi-language Ontology-Based Search Engine. // In Proc. of The Third International Conference on Advances in Computer-Human Interactions. – Netherlands. – 2010. – P. 13-18 .

17. Capstick J., Diagne K. MULINEX: Multilingual Web Search and Navigation. // In Proc. of Natural Language Processing and Industrial Applications. –

18. Maeda A., Sadat F. Query Term Disambiguation for Web Cross-Language Information Retrieval using a Search Engine. // In Proc. of the Fifth Int'l Workshop on Info. Retrieval with Asian Languages. – China. – 2016. – P. 173-179.

19. Jialun Q., Zhou Y., Hsinchun C. Multilingual Web retrieval: An experiment in English-Chinese business intelligence. // Journal of the American

Society for Information Science and Technology (JASIST). – 2016. – Vol. 5. – P. 671-683.

20. Sethuramalingam S., Vasudeva V. IIIT Hyderabad's CLIR experiments for FIRE-2008// In Proc. of The working notes of First Workshop of Forum for Information Retrieval Evaluation (FIRE) . – 2008. – Kolkata.

21. Круглов В. Порівняння алгоритмів Мамдани й Сугено в завданні апроксимації функції. // Нейрокомп'ютери: розробка, застосування. – 2013. – № 5. – С. 70–82

22. Іванова Г.С., Андрєєв А.М., Шоуман М.А. Пошук і Ранжирування документів з використанням мультиагентной системи. //Фундаментальні дослідження. – 2015. – № 10 . – частина 3. – С.489–494. (дата зверненя: 17.05.2019).

23. Aref M. A multi-agent system for natural language understanding. // In Proc. of International Conference on Integration of Knowledge Intensive Multi-Agent Systems (KIMAS '03) . – USA. – 2003. – P. 36-40.

24. Websearchengine.Wikipedia.TheFreeEncylopedia. [Електронний ресурс].URL: http://en.wikipedia.org/wiki/Web_search_engine. (дата зверненя: 27.04.2019).

25. Sheth B., Maes P. Evolving agents for personalized information filtering. // In Proc. of IEEE Conference on Artificial Intelligence for Applications (CAIA-93) . – 1993. – P. 345–352

26. Lieberman H. Letizia: An agent that assists web browsing. // In Proc. of the Fourteenth International Joint Conference on Artificial Intelligence (IJCAI-95) . – Canada. 2015. – P. – 924–929.