

## Додаток А

## ЛІСТИНГ

```
package evolution2;
import java.awt.*;
import java.awt.event.*;
import java.util.*;
/**
 *
 * @author
 */
public class Settings extends Frame {
    private static ArrayList<Checkbox> checkboxeseff = new
ArrayList(11);
    private static ArrayList<Checkbox> checkboxesrec = new
ArrayList(12);

    Textfield wight = new Textfield();
    Label widthl = new Label ("Висота поля:");
    Textfield height = new Textfield();
    Label heightl = new Label ("Ширина поля:");
    Textfield foodchance = new Textfield();
    Label foodchancel = new Label ("Імовірність ресурсу:");
    Textfield numagents = new Textfield();
    Label numagentsl = new Label ("Кількість агентів, що
заселяються:");

    Textfield delay = new Textfield();
    Label delayl = new Label ("Затримка, мс:");
    Panel panel = new Panel();
    Button sub = new Button("Установити");
    GridLayout manager;

    Textfield reprflow = new Textfield();
    Label reprflowl = new Label ("Витрати на розмноження:");
```

```

Textfield reprfailflow = new Textfield();
Label reprfailflowl = new Label ("Витрати при невдалім
розмноженні:");
Textfield moveflow = new Textfield();
Label moveflowl = new Label ("Витрати на переміщення:");
Textfield movefailflow = new Textfield();
Label movefailflowl = new Label ("Витрати при невдалому
переміщення:");
Textfield rotateflow = new Textfield();
Label rotateflowl = new Label ("Витрати на поворот:");

Textfield waitflow = new Textfield();
Label waitflowl = new Label ("Витрати при відпочинку:");
Textfield eatfailflow = new Textfield();
Label eatfailflowl = new Label ("Витрати при невдалім
харчуванні:");
Textfield eatreplenish = new Textfield();
Label eatreplenishl = new Label ("Ефективність споживання
ресурсу:");
Textfield reprenergylim = new Textfield();
Label reprenergyliml = new Label ("Обмеження на розмноження по
ресурсу:");
Textfield scionenergypart = new Textfield();
Label scionenergypartl = new Label ("Частина ресурсу
нащадка:");

Textfield donateflow = new Textfield();
Label donateflowl = new Label (" ресурс, Що Віддається:");
Textfield donatefailflow = new Textfield();
Label donatefailflowl = new Label ("Витрати при невдалій
віддачі:");
Textfield snatchreplenish = new Textfield();
Label snatchreplenishl = new Label (" ресурс, Що
Відбирається:");
Textfield snatchfailflow = new Textfield();

```

```
Label snatchfailflow1 = new Label ("Витрати при невдалому  
вдборі:");  
  
Textfield energy = new Textfield();  
Label energy1 = new Label ("Внутрішній ресурс:");  
Textfield direction = new Textfield();  
Label direction1 = new Label ("Напрямок:");  
  
Textfield ampl = new Textfield();  
Label ampl1 = new Label ("Амплітуда мутацій:");  
Textfield startampl = new Textfield();  
Label startampl1 = new Label ("Амплітуда мутацій:");  
  
{  
    checkboxeseff.add(new Checkbox("Переміщення"));  
    checkboxeseff.add(new Checkbox("Поворот праворуч"));  
    checkboxeseff.add(new Checkbox("Поворот ліворуч"));  
    checkboxeseff.add(new Checkbox("Поглинання ресурсу "));  
    checkboxeseff.add(new Checkbox("Розмноження"));  
    checkboxeseff.add(new Checkbox("Очікування"));  
    checkboxeseff.add(new Checkbox("Віддати ресурс"));  
    checkboxeseff.add(new Checkbox("Відібрати ресурс"));  
    checkboxeseff.add(new Checkbox("Нащадок спереду"));  
    checkboxeseff.add(new Checkbox("Нащадок праворуч"));  
    checkboxeseff.add(new Checkbox("Нащадок ліворуч"));  
    checkboxeseff.add(new Checkbox("Нащадок позаду"));  
  
    checkboxesrec.add(new Checkbox("Внутрішня енергія"));  
    checkboxesrec.add(new Checkbox("Ресурс у гнізді"));  
    checkboxesrec.add(new Checkbox("Ресурс спереду"));  
    checkboxesrec.add(new Checkbox("Ресурс праворуч"));  
    checkboxesrec.add(new Checkbox("Ресурс ліворуч"));  
    checkboxesrec.add(new Checkbox("Агент спереду"));  
    checkboxesrec.add(new Checkbox("Агент праворуч"));  
    checkboxesrec.add(new Checkbox("Агент ліворуч"));  
    checkboxesrec.add(new Checkbox("Енергія мети"));
```

```

checkboxesrec.add(new Checkbox("Споріднення мети"));
checkboxesrec.add(new Checkbox("Споріднення сусідів"));
}

Label receptors = new Label("Рецептори");
Label effectors = new Label("Ефектори");

Checkbox newform = new Checkbox("Створювати новий вид при
заселенні");

Settings (int mode) {
    switch (mode) {
        case 1 : totalsetmodeone(); break;
        case 2 : totalsetmodetwo(); break;
        case 3 : totalsetmodethree(); break;
    }
}

Settings (Agent agent, int mode) {
    if (mode == 2)
        agentsetmodetwo(agent);
    else
        agentsetmodeone(agent);
}

Settings (Agent agent, int i, int o, Outsinsapses outsins) {
    double[][] sin = agent.getsin();
    Textfield text = new Textfield();
    Label label = new Label ("Вага зв'язку " + i + " " + o);
    text.setText(sin[i][o] + "");
    sub.addActionListener((ActionEvent event) -> {
        agent.setsin(new Double(text.getText()), i, o);
        outsins.sin.repaint();
        this.dispose();
    });
    panel.setLayout(manager = new GridLayout(3, 1));
}

```

```

panel.add(label);
panel.add(text);
panel.add(sub);
this.add(panel);
this.setsize(100, 200);
this.setvisible(true);
}

private void totalsetmodeone() {
    panel.setLayout(manager = new GridLayout(23, 2));
    panel.add(width1);
    panel.add(wight);
    wight.setText(Field.getwight() + "");
    panel.add(height1);
    panel.add(height);
    height.setText(Field.getlenght()+ "");
    panel.add(foodchancel);
    panel.add(foodchance);
    foodchance.setText(Field.getfoodchance()+ "");
    panel.add(numagents1);
    panel.add(numagents);
    numagents.setText(Field.getnumagents()+ "");
    panel.add(delay1);
    panel.add(delay);
    delay.setText(Field.getdelay()+ "");

    panel.add(amp11);
    panel.add(amp1);
    amp1.setText(Lifeform.getamp1()+ "");
    panel.add(energy1);
    panel.add(energy);
    energy.setText(Lifeform.getenergy()+ "");

    panel.add(reprflow1);
    panel.add(reprflow);
    reprflow.setText(Lifeform.getreprflow()+ "");
}

```

```

panel.add(reprfailflowl);
panel.add(reprfailflow);
reprfailflow.setText(Lifeform.getreprfailflow()+ "");
panel.add(moveflowl);
panel.add(moveflow);
moveflow.setText(Lifeform.getmoveflow()+ "");
panel.add(movefailflowl);
panel.add(movefailflow);
movefailflow.setText(Lifeform.getmovefailflow()+ "");
panel.add(rotateflowl);
panel.add(rotateflow);
rotateflow.setText(Lifeform.getrotateflow()+ "");

newform.setState(Lifeform.getnewform());
this.add(panel);
this.setSize(550, 500);
this.setVisible(true);
sub.addActionListener((ActionEvent event) -> {
    if (!wight.getText().equals(""))
        Field.setup("wight", wight.getText());
    if (!height.getText().equals(""))
        Field.setup("length", height.getText());
    if (!foodchance.getText().equals(""))
        Field.setfoodchance(new
Double(foodchance.getText()));
    if (!numagents.getText().equals(""))

Field.setnumagents(Integer.parseInt(numagents.getText()));
    if (!delay.getText().equals(""))
        Field.setdelay(Integer.parseInt(delay.getText()));

    if (!energy.getText().equals(""))
        Lifeform.setenergy(new Double(energy.getText()));
    if (!reprflow.getText().equals(""))

Lifeform.setreprflow(Integer.parseInt(reprflow.getText()));

```

```

        if (!reprfailflow.getText().equals(""))

Lifeform.setreprfailflow(Integer.parseInt(reprfailflow.getText()));
        if (!moveflow.getText().equals(""))

Lifeform.setmoveflow(Integer.parseInt(moveflow.getText()));

Lifeform.setsnatchfailflow(Integer.parseInt(snatchfailflow.getText(
)));

        if (!startampl.getText().equals(""))
            Lifeform.setstartampl(new
Double(startampl.getText()));
            Lifeform.setnewform(newform.getState());

        this.dispose();
    });
}

private void totalsetmodetwo() {
    for (int i = 0; i < 11; i++)
        if (Lifeform.getinavail()[i] == 1)
            checkboxesrec.get(i).setState(true);
    for (int i = 0; i < 12; i++)
        if (Lifeform.getoutavail()[i] == 1)
            checkboxeseff.get(i).setState(true);
    panel.setLayout(manager = new GridLayout(13, 2));
    panel.add(effectors);
    panel.add(receptors);
    for (int i = 0; i < 11; i++) {
        panel.add(checkboxeseff.get(i));
        panel.add(checkboxesrec.get(i));

    }
    panel.add(checkboxeseff.get(11));
    panel.add(sub);
    this.add(panel);
}

```

```
this.setsize(320, 400);
this.setvisible(true);

sub.addactionlistener((Actionevent event) -> {
    for (int i = 0; i < 11; i++) {
        if (checkboxesrec.get(i).getState())
            Lifeform.getinavail()[i] = 1;
        else
            Lifeform.getinavail()[i] = 0;
    }
    for (int i = 0; i < 12; i++) {
        if (checkboxeseff.get(i).getState())
            Lifeform.getoutavail()[i] = 1;
        else
            Lifeform.getoutavail()[i] = 0;
    }
    this.dispose();
});
}
}
```



Додаток Б  
Слайди презентації

**Міністерство освіти і науки України**  
**Харківський національний університет радіоелектроніки**

**Атестаційна робота магістра**

**Дослідження алгоритмів пошуку інформації на основі  
штучних агентів**

**Керівник:**  
проф. Шубін І.Ю.

**Виконав:**  
Струков О.М.,  
ст. гр. ПЗмзд-17-1

Харків, 2019

1 **1**

## **Актуальність**

- Ефективний інструмент пошуку у всім доступному обсязі даних 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 на технічних напрямках .*

2

## Мета роботи

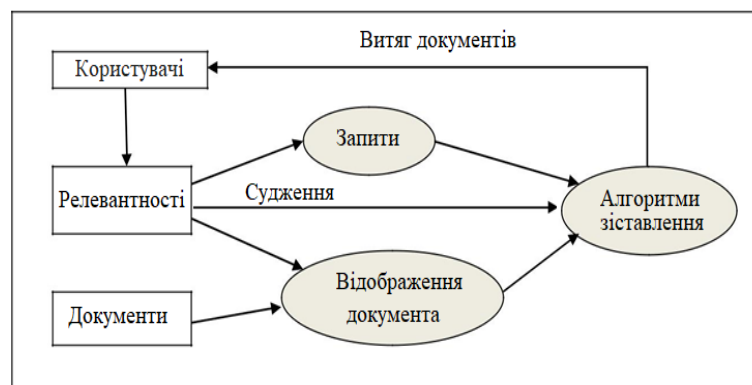
- підвищення релевантності перших результатів автоматичного багатомовного інформаційного пошуку в рамках застосування мультиагентного підходу.

Для досягнення поставленої мети необхідно було розв'язати **наступні завдання:**

- аналіз процесу БІП документів і виявлення операцій, які повинні бути виконані в процесі пошуку на заданих мовах для одержання результатів, ранжованих по ступеню релевантності запиту;
- розробка архітектури і реалізації мультиагентної системи БІП;
- розробка структурної моделі БІП, що враховує мультиагентну реалізацію пошукової системи;
- Розробка програмних процедур оцінки релевантності знайденого документа по якісно заданим вагам термінів у запиті та документі.
- виконання експериментальних досліджень і оцінка їх результатів

3

## Базова архітектура інформаційно-пошукових систем



4

## Індексування

- Наразі ручне індексування все ще вважається кращим ніж автоматичне через його здатність справлятися з невизначеністю.
- Однак ручне індексування страждає від високих операційних витрат і є практично неможливим для виконання в сьгоднішніх 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.

5

## Оцінки якості пошукових систем

- Якість сучасних пошукових систем оцінюється більшою кількістю критеріїв, серед яких основними вважаються:
  - час обробки запиту;
  - пертинентність;
  - релевантність;
  - точність;
  - повнота;
  - випадання.

6

## Постановка задач дослідження

- 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:

- 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 алгоритм оцінювання релевантності документів, що забезпечує точність результатів на рівні існуючих методів, але має більш низьку обчислювальну складність.
- Розробити програмну модель запропонованих моделей і алгоритмів.

7

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

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

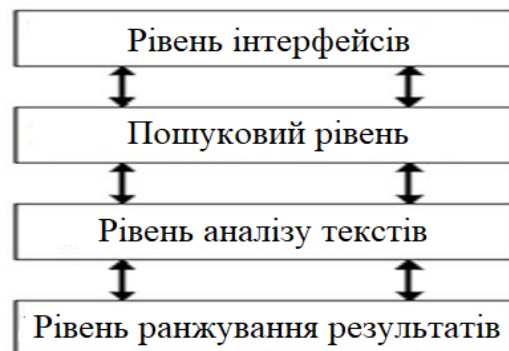
8

## Варіанти зважування зворотної частоти документів

| Схема зважування                 | Варіант 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 | $\log \frac{N - n_t}{n_t}$         |

9

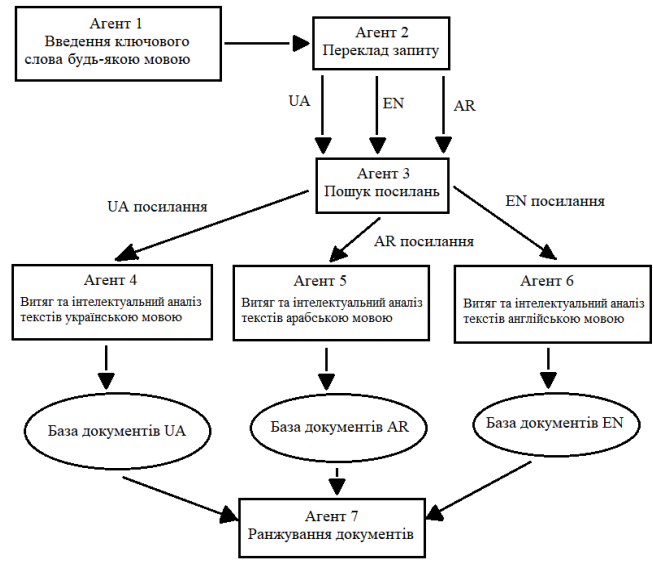
## Розробка мультиагентної архітектури системи БІП



Функціональна схема МАС для одномовного пошуку

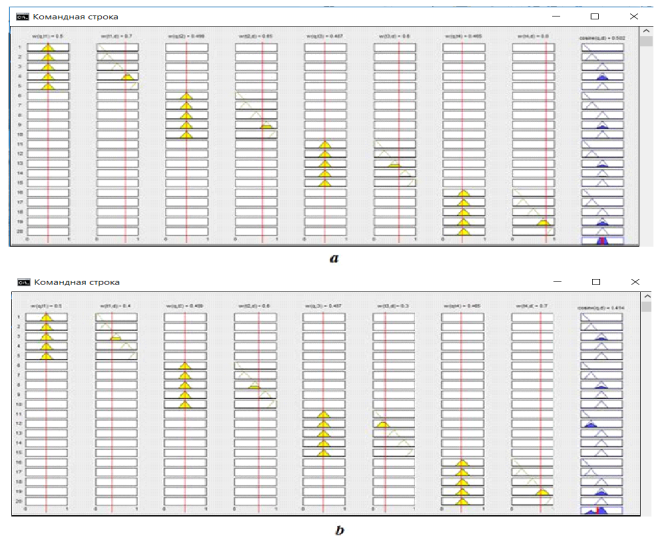
10

## Архітектура мультиагентної системи БІП



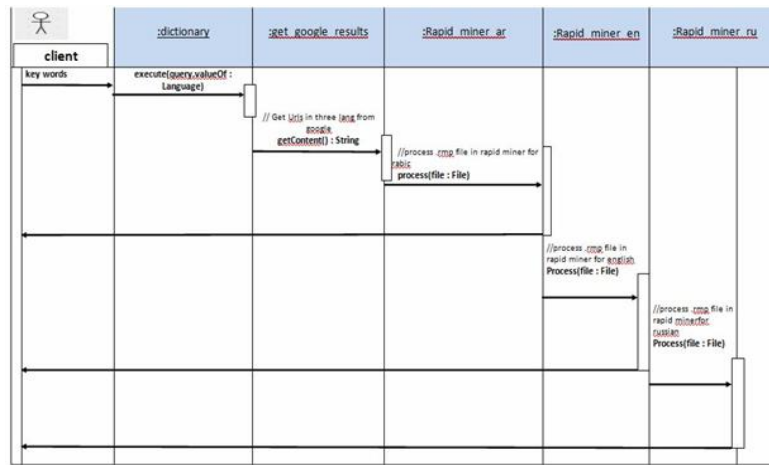
11

## Алгоритм експертної оцінки релевантності за допомогою нечіткого логічного висновку



12

## Діаграма послідовностей запропонованої системи



13

## Висновки

В результаті роботи

- розроблена багатоагентна архітектура системи БП, визначені функції агентів і взаємозв'язки між ними, що дозволяє одержати більш ефективну реалізацію системи БП;
- введена лінгвістична змінна «Вага терміну», що дозволяє формалізувати вербальну оцінку релевантності документів і використовувати її для наступного ранжування результатів пошуку;
- запропоноване ранжування результатів пошуку у випадку однієї або декількох мов з використанням алгоритмів нечіткого логічного висновку типів Мамдані й Сугено, що збільшило ймовірність знаходження необхідних документів серед перших знайдених у порівнянні з існуючими пошуковими системами.
- створенно мультиагентну систему інформаційного пошуку для російської і англійської мов.

14

14



ДОДАТОК В  
Апробація роботи



7<sup>th</sup> International Scientific Conference

**Science progress in European countries:  
new concepts and modern solutions**

Hosted by the ORT Publishing and

The Center for Scientific Research "Solution"

Conference papers

May 31, 2019

Stuttgart, Germany

---

---

7<sup>th</sup> International Scientific Conference

*"Science progress in European countries: new concepts and modern solutions"*: Papers of the 7th International Scientific Conference.  
May 31, 2019, Stuttgart, Germany. 436 p.

Edited by **Ludwig Siebenberg**

Technical Editor: **Peter Meyer**

ISBN 978-3-944375-22-9

Published and printed in Germany by ORT Publishing (Germany) in  
association with the Center For Scientific Research "Solution" (Ukraine)  
May 31, 2019.

**ORT Publishing**

Schwieberdinger Str. 59

70435 Stuttgart, Germany

All rights reserved

© ORT Publishing

© All authors of the current issue

ISBN 978-3-944375-22-9