

Київський національний університет імені Тараса Шевченка  
Факультет інформаційних технологій  
Кафедра програмних систем і технологій

УДК 004.42

*На правах рукопису*

## **ВИПУСКНА КВАЛІФІКАЦІЙНА БАКАЛАВРСЬКА РОБОТА**

Тема: “Методи програмного виявлення інформаційних вкидань  
на сайтах новин та соціальних мережах”

Спеціальність - 121 “Інженерія програмного забезпечення”

### **ПОЯСНЮВАЛЬНА ЗАПИСКА (позначення)**

Студент

ІПЗ-42

Іван КАРПЕЦЬ

(шифр групи)(підпис) (розшифровка підпису) (дата)

Науковий керівник

к.ф.-м.н., доц. Сергій ПОЛЯКОВ

(посада) (підпис) (розшифровка підпису) (дата)

Консультант

з питань нормоконтролю - фахівець

Тамара ЧАПОВСЬКА

(підпис) (розшифровка підпису) (дата)

Допускається до захисту

з питань нормоконтролю

Завідувач кафедри

д.т.н., доц. Олексій БИЧКОВ

(посада) (підпис) (розшифровка підпису) (дата)

Київ - 2021

Київський національний університет імені Тараса Шевченка  
 Факультет інформаційних технологій  
 Кафедра програмних систем і технологій  
 Спеціальність 121 “Інженерія програмного забезпечення”

ЗАТВЕРДЖУЮ:  
 Завідувач кафедри програмних систем і  
 технологій \_\_\_\_\_ (Олексій БИЧКОВ)  
 „\_\_\_” \_\_\_\_\_ 20\_\_ р.

### ЗАВДАННЯ

## НА ВИПУСКНУ КВАЛІФІКАЦІЙНУ БАКАЛАВРСЬКУ РОБОТУ СТУДЕНТУ

Карпецю Івану Івановичу

(прізвище, ім'я, по батькові)

1. Тема випускної кваліфікаційної бакалаврської роботи “Методи програмного виявлення інформаційних вкидань на сайтах новин та соціальних мережах”  
керівник проекту (роботи) Поляков Сергій Анатолійович к.ф.-м.н., доцент  
затверджена наказом вищого навчального закладу від „11” листопада 2020р. № 6
2. Строк здачі студентом закінченої роботи \_\_\_\_\_
3. Вихідні дані до роботи Звіти та наукові дослідження методів виявлення інформаційних вкидань на сайтах новин та соціальних мережах
4. Зміст пояснювальної записки (перелік питань, що їх належить розробити)
  1. Визначення проблеми інформаційних вкидань
  2. Аналіз методів виявлення інформаційних вкидань
  3. Аналіз методів програмного виявлення інформаційних вкидань
  4. Розробка алгоритмічної моделі та програмної реалізації методів виявлення інформаційних вкидань на сайтах новин та соціальних мережах
5. Перелік графічного матеріалу (з точним забезпеченням обов'язкових креслень)
  1. Косинус між векторами(рис. 3.1, ст. 27)
  2. Графічний алгоритм (рис. 3.2, ст. 28)
  3. Список слів пов'язаних зі словом "Швеція" за допомогою Word2vec, у порядку близькості(рис. 3.3, ст. 29)

4. CBOW модель з одним словом в контексті(рис. 3.4, ст. 30)

5. Графічний алгоритм(рис. 3.5, ст. 31)

6. Skip-Gram модель(рис. 3.6, ст. 32)

7. Метод опорних векторів(рис. 3.7, ст. 35)

8. Приклад виділяючої площини(рис. 3.8, ст. 36)

## 6. Консультанти з роботи із зазначенням розділів роботи, що їх стосуються

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв
1	Сергій ПОЛЯКОВ	23.01.2021	01.02.2021
2	Сергій ПОЛЯКОВ	01.02.2021	18.02.2021
3	Сергій ПОЛЯКОВ	18.02.2021	28.04.2021
4	Сергій ПОЛЯКОВ	28.04.2021	25.05.2021

7. Дата видачі завдання 17.03.2021

Керівник Сергій ПОЛЯКОВ  
(підпис) (розшифровка підпису)

Завдання прийняв до виконання Іван КАРПЕЦЬ  
(підпис) (розшифровка підпису)

## КАЛЕНДАРНИЙ ПЛАН

Номер і назва етапів бакалаврської роботи	Термін виконання етапів роботи	Примітка
1. Вибір і вивчення літератури	23.01.2021	виконано
2. Аналіз алгоритмів машинного навчання для обробки даних	18.02.2021	виконано
3. Розробка алгоритмічної моделі	11.03.2021	виконано
4. Аналіз алгоритму машинного навчання	15.04.2021	виконано
5. Розробка алгоритму для визначення інформаційних вкидань	28.04.2021	виконано
6. Затвердження пояснювальної записки роботи завідувачем кафедри	01.06.2021	виконано

Студент – бакалавр Іван КАРПЕЦЬ  
(підпис) (розшифровка підпису)

Керівник роботи Сергій ПОЛЯКОВ  
(підпис) (розшифровка підпису)

## АНОТАЦІЯ

**Випускна кваліфікаційна бакалаврська робота:** 60 с., 8 рис., 8 табл., 3 додат., 18 джерела.

**Тема:** Методи програмного виявлення інформаційних вкидань на сайтах новин та соціальних мережах.

**Об'єкт дослідження:** дані публікацій на сайтах новин та соціальних мережах.

**Мета роботи:** аналіз та розробка програмних методів виявлення інформаційних вкидань на сайтах новин та соціальних мережах.

**Предмет дослідження:** алгоритми машинного навчання аналізу публікацій на сайтах новин та соціальних мережах.

### **Результати дослідження:**

У цій роботі було досліджено проблему підроблених новин та реалізовану програмне забезпечення для їх виявлення. На етапі характеристики введено основні поняття та принципи фальшивих новин як у традиційних, так і в соціальних медіа. На фазі виявлення розглянуто наявні підходи та реалізовано програмне забезпечення для виявлення інформаційних вкидань. Також описано набори даних, показники оцінки та перспективні напрямки досліджень фальшивих новин.

### **Висновок**

Широке поширення підроблених новин може мати надзвичайно негативний вплив на індивіди та суспільство. Тому виявлення фальшивих новин в інтернеті останнім часом стало популярним дослідженням, яке привертає величезну увагу.

ІНФОРМАЦІЙНІ ВКИДАННЯ, ФАЛЬШИВІ НОВИНИ, МАШИННЕ НАВЧАННЯ,  
АЛГОРИТМ.

## АННОТАЦИЯ

**Выпускная квалификационная бакалаврская работа:** 60 с., 8 рис., 8 табл., 3 доп., 18 источника.

**Тема:** Методы программного выявления информационных вбросов на сайтах новостей и социальных сетях.

**Объект исследования:** данные публикаций на сайтах новостей и социальных сетях.

**Цель работы:** разработка метода выявления информационных вбросов.

**Предмет исследования:** методы выявления информационных вбросов.

**Результаты исследования:**

В этой работе были исследована проблема поддельных новостей и реализовано программное обеспечение для их обнаружения. На этапе характеристики введено основные понятия и принципы фальшивых новостей как в традиционных, так и в социальных медиа. На фазе выявления рассмотрены имеющиеся подходы и реализовано программное обеспечение для выявления информационных вбросов. Также описаны наборы данных, показатели оценки и перспективные направления исследований фальшивых новостей.

**Выводы**

Широкое распространение поддельных новостей может иметь чрезвычайно негативное влияние на индивиды и общество. Поэтому выявление фальшивых новостей в интернете в последнее время стало популярным исследованием, которое привлекает огромное внимание.

ИНФОРМАЦИОННЫЙ ВБРОС, ФЕЙКОВЫЕ НОВОСТИ, МАШИННОЕ ОБУЧЕНИЕ, АЛГОРИТМ.

## ANNOTATION

**Final qualifying bachelor's thesis:** 60 pages, 8 figures, 8 tables, 3 appendices, 18 sources.

**Topic:** Software methods of fake news detection on news websites and social networks.

**Object of research:** publications on news sites and social networks.

**Purpose:** develop a method for detecting fake news.

**Subject of research:** methods of detecting fake news.

**Results of the research:**

This paper investigated the problem of fake news and implemented software to detect them. At the stage of characterization, the basic concepts and principles of fake news are introduced in both traditional and social media. In the detection phase, the existing approaches are considered and software for detecting information injections is implemented. Data sets, evaluation indicators and promising areas of fake news research are also described.

**Conclusion**

Widespread fake news can have extremely negative effects on individuals and society. Therefore, the detection of fake news on the Internet has recently become a popular study, which attracts a lot of attention.

FAKE NEWS, MACHINE LEARNING, ALGORITHM.

## ЗМІСТ

	Стр.
ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ .....	8
ВСТУП.....	9
<b>РОЗДІЛ 1</b>	
<b>ХАРАКТЕРИСТИКА ФЕЙКОВИХ НОВИН</b>	
1.1. Підроблені новини в традиційних ЗМІ .....	14
1.2. Соціальні основи екосистеми підроблених новин .....	15
1.3. Підроблені новини в соціальних медіа .....	17
1.4. Висновки до розділу .....	19
<b>РОЗДІЛ 2</b>	
<b>ВИЗНАЧЕННЯ ФЕЙКОВИХ НОВИН</b>	
2.1. Визначення проблеми .....	20
2.2. Оцінка ефективності методів виявлення фейкових новин .....	21
2.2.1 Набори даних.....	21
2.2.2 Метрики оцінки .....	22
2.3. Виявлення спаму та ботів .....	23
2.4. Висновки до розділу .....	24
<b>РОЗДІЛ 3</b>	
<b>ВИБІР ТЕХНОЛОГІЙ ТА АЛГОРИТМІВ ВИРІШЕННЯ ПРОБЛЕМИ ПРОГРАМИ</b>	
3.1. Векторна модель.....	25
3.2. Заходи подібності. Косинусна схожість .....	29
3.3. Модель Word2vec. Методи CBOW та Skip-Gram.....	30
3.4. Бібліотека Tkinter .....	34
3.5. Класифікатори .....	34
3.6. Рядки .....	36

3.7. Висновки до розділу .....	39
--------------------------------	----

## **РОЗДІЛ 4**

### **РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ**

4.1. Файлова система програми .....	41
-------------------------------------	----

4.2 Архітектурний шаблон програми .....	42
---	----

4.3. Висновки до розділу .....	45
--------------------------------	----

ВИСНОВКИ .....	46
----------------	----

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ .....	47
----------------------------------	----

ДОДАТКИ .....	49
---------------	----

**ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ,  
СКОРОЧЕНЬ І ТЕРМІНІВ**

МН - машинне навчання

ОС - операційна система

ПЗ - програмне забезпечення

MVC - model/view/controller

ООП - об'єктно орієнтоване програмування

TDD(test-driven development) - керування тестами розробка

## ВСТУП

### Актуальність роботи

Наукова публікація - це тривалий процес, що включає визначення теми дослідження, або концепції дослідження, проектування досліджень, збір даних, аналіз даних і результатів, публікацію досліджень. Усі процеси потрібно виконувати правильно, дотримуючись міжнародної етики, щоб науково-дослідна публікація збагатила науку, технологію та мистецтво. Ролі редакторів та відповідних інструментів для фільтрації статей, які збираються публікуватись, стають дуже важливими для того, щоб представлені документи були науково та етично прийнятними.

Широко встановлено, що етичні проступки в науковій публікації конкретно полягають у формуванні даних, багаторазовому поданні, плагіаті та неправильному використанні авторства.

Оскільки ми багато часу проводимо в соціальних мережах, то все більше людей прагнуть шукати новини в інтернеті, а не використовувати традиційні методи. Причини цього наступні: зручність і доступність в порівнянні з традиційними джерелами інформації, такими як газети, телепередачі; можливість легше ділитися, коментувати і обговорювати новини з друзями та іншими читачами в соціальних мережах. Як результат, сайти новин в інтернеті стали головним джерелом новин для суспільства. Незважаючи на це, якість новин в інтернеті є гіршою ніж традиційних джерел. Дешевше та легше розповсюджувати новини онлайн, тому з'являються фейкові новини для політичних та комерційних цілей. Широке поширення фальшивих новин може мати серйозний негативний вплив на людей та суспільство. По-перше, підроблені новини можуть порушити баланс автентичності екосистеми новин. По-друге, фальшиві новини навмисно переконують споживачів приймати помилкові переконання. Підробленими новинами зазвичай маніпулюють пропагандисти для передачі політичного повідомлення та впливу. По-третє, фейкові новини змінюють спосіб інтерпретації та реагування на реальні новини у людей. Наприклад, деякі новини були створені, щоб

викликати недовіру, перешкоджаючи визначення людьми, що є правдою, а що - ні. Щоб пом'якшити негативні наслідки, спричинені фальшивими новинами, слід розробляти та досліджувати методи виявлення фейкових новин в інтернеті. Виявлення фальшивих новин у соціальних медіа створює кілька нових дослідницьких проблем. Хоча самі фальшиві новини є не новою проблемою - нації використовують ЗМІ для здійснення пропаганди століттями - зростання веб-породжених новин у соціальних медіа робить фальшиві новини більш впливовими, які кидають виклики традиційним журналістським нормам. Існує кілька характеристик цієї проблеми, які роблять її однозначно складною для автоматизованого виявлення.

По-перше, фальшиві новини навмисно написані для введення в оману читачів, що робить їх нетривіальними для виявлення просто на основі контенту. Зміст фейкових новин є досить різноманітним за тематикою, стилями та медіа-платформами. Наприклад, фальшиві новини можуть наводити правдиві докази в невірному контексті. Для вдосконалення виявлення слід застосовувати іншу допоміжну інформацію, наприклад, знання про соціальну діяльність користувачів.

По-друге, використання цієї допоміжної інформації насправді призводить до іншого критичного завдання: якість самих даних. Підроблені новини, як правило, пов'язані з подіями, які, можливо, не були належним чином перевірені наявними базами знань через відсутність підтверджуючих доказів.

Ефективні методи диференціації надійних користувачів, вилучення корисних публікацій в інтернеті - це відкрита область дослідження, що потребує подальших досліджень.

У даній курсовій роботі розглянуто методи виявлення інформаційних вкидань. Було досліджено, що:

- Підроблені новини у соціальних мережах існують вже багато років; однак, немає погодженого визначення терміну "фейкові новини".
- Інтернет виявився потужним джерелом підроблених новин. Існують певні шаблони, які можна використовувати для виявлення підроблених новин. Огляд

існуючих методів виявлення підроблених новин може забезпечити базове розуміння методів виявлення інформаційних вкидань.

- Виявлення підроблених новин перебуває на початковій стадії свого розвитку, все ще існує багато складних питань, які потребують подальшого дослідження. Необхідно обговорити потенційні напрямки досліджень, які можуть покращити виявлення підроблених новин.

### **Мета і задачі дослідження**

**Метою** бакалаврської роботи є аналіз та розробка програмних методів визначення інформаційних вкидань на сайтах новин та соціальних мережах.

Розроблена система повинна аналізувати набір даних та визначати вірогідність того, що це фейкові дані.

Досягнення мети включало розв'язання таких **задач**:

- 1) огляд існуючих методів визначення інформаційних вкидань;
- 2) вибір ефективного алгоритму для визначення інформаційних вкидань та обґрунтування доцільності його використання;
- 3) програмна реалізація алгоритму.

**Об'єктом дослідження** є дані публікацій на сайтах новин та соціальних мережах.

**Предметом дослідження** є алгоритми машинного навчання для виявлення фейкових новин.

### **Методи дослідження**

Дані дослідження структуровані в формат зручний для їх аналізу. Інформаційні вкидання визначаються за допомогою алгоритмів машинного навчання.

### **Наукова новизна отриманих результатів**

Досліджено можливості аналізу публікацій на сайтах новин та соціальних мережах, застосовуючи алгебраїчні та програмні методи аналізу. Виконана програмна реалізація алгоритму для виявлення інформаційних вкидань.

### **Практичне значення одержаних результатів**

Одержана реалізація методів аналізу тексту та алгоритмів машинного навчання для класифікації фальшивих новин дозволяє аналізувати сайти новин та стрічки соціальних мереж.

### **Особистий внесок студента**

Розроблена програмна реалізація методів обробки тексту та алгоритми машинного навчання, що визначають інформаційні вкидання з набору даних, одержаних з сайтів новин та соціальних мереж.

### **Структура та обсяг роботи**

Робота викладена на 60 сторінках друкованого тексту, яка складається із вступу, чотирьох розділів та висновків до кожного розділу, загального висновку, списку використаних джерел (18 найменування). Робота містить 8 таблиць, 8 рисунки та 3 додатки, обсягом 12 стор.

## РОЗДІЛ 1

### ХАРАКТЕРИСТИКА ФЕЙКОВИХ НОВИН

Фейкові новини існують майже з того часу, коли був винайдений перший друкарський верстат в 1439 році. Однак немає конкретного визначення терміну "фейкові новини".

Вузьке визначення фальшивих новин - це статті новин, які навмисно допускають недостовірну інформацію і можуть ввести читачів у оману.

По-перше, фальшиві новини включають неправдиву інформацію, яку можна перевірити. По-друге, фейкові новини створюються з нечесним наміром для того, щоб ввести в оману споживачів. Більш широкі визначення фальшивих новин зосереджуються на достовірності або намірі вмісту новин. Вважають, що сатиричні новини є фальшивими, оскільки вміст неправдивий, хоча сатира часто орієнтована на розваги і розкриває власну оманливість споживачам.

У даній курсовій роботі я використовую вузьке визначення фейкових новин. Є три причини вибору цього поняття. По-перше, основні наміри підроблених новин мають теоретичне та практичне значення, що дає змогу глибше зрозуміти та проаналізувати цю тему. По-друге, будь-які прийоми для перевірки істинності, що стосується вузької концепції підроблених новин, також можна застосовувати до більш широкого визначення. По-третє, це визначення здатне усунути неоднозначності між фальшивими новинами та схожими визначеннями.

Наступні поняття не є підробленими новинами згідно визначення:

- сатиричні новини - це новини, які не мають наміру вводити в оману споживачів, тому навряд будуть сприйняті неправильно;
- чутки, які не походять від подій;
- теорії змов, які важко перевірити на істинність;
- дезінформація, яка створюється ненавмисно;
- містифікації, мотивовані лише розвагою або націлені на афери.

## 1.1. Підроблені новини в традиційних ЗМІ

Інформаційні вкидання це не нова проблема. Екологія фальшивих новин змінюється. Спочатку це були газети, радіо/телебачення. Потім з'явився інтернет, де інформаційні вкидання широко розповсюджуються і зараз.

У даному підрозділі буде розглянуто декілька психологічних та суспільствознавчих основ, які описують вплив підроблених новин на індивіда.

Люди не дуже добре розрізняють справжні та підроблені новини. Існує кілька психологічних та когнітивних теорій, які можуть пояснити це явище та впливову силу фальшивих новин.

Традиційні фейкові новини в основному націлені на використання індивідуальних вразливостей споживачів. Є два основні фактори, які роблять споживачів природно вразливими до підроблених новин:

- Наївний реалізм: споживачі схильні вважати, що їхнє сприйняття реальності є єдиним точним, і тоді всі, хто не згодні з ними, є не обізнаними;
- Підтверджувальне упередження: споживачі вважають за краще отримувати інформацію, яка підтверджує їх існуючі погляди.

Через ці когнітивні упередження, притаманні людській природі, фальшиві новини часто можна сприймати як реальні. Більше того, коли формується неправильне сприйняття, його дуже важко змінити. Дослідження психології людини показують, що виправлення недостовірної інформації шляхом подання правдивої може навіть погіршити ситуацію.

## 1.2. Соціальні основи екосистеми підроблених новин

Враховуючи всю екосистему споживання новин, я можу також описати деякі соціальні динаміки, які сприяють розповсюдженню фальшивих новин.

Теорія перспектив описує прийняття рішень як процес, за допомогою якого люди роблять вибір, виходячи із перемог і поразок порівняно з їхнім теперішнім станом. Як описано теорією соціальної ідентичності і теорією нормативного впливу, прагнення до соціального прийняття є важливим для особистості людини та її самооцінки. Завдяки цьому користувачі можуть вибрати "соціально безпечний" варіант при споживанні та розповсюдженні новин. Користувачі дотримуються норм, що встановлені в громаді, навіть якщо новини, які вони поширюють, є фальшивими новинами.

Для пояснення фальшивих новин я припускаю, що в інформації є два види ключових гравців: видавець та споживач. Процес публікації новин моделюється як відображення  $s$  в  $a$  із ефектом спотвореності  $b$ ,  $s \xrightarrow{b} a$ , де  $b = [-1,0,1]$ .

Користь для видавництва:

- короткострокової корисності: стимул до отримання максимального прибутку, що співвідноситься із кількістю досягнутих споживачів;
- довгострокова корисність: їх репутація з точки зору достовірності новин.

Користь для споживачів:

- інформаційна корисність: отримання правдивої інформації (зазвичай потрібні додаткові інвестиційні витрати);
- психологічна користь: отримання новин, що задовольняють припущенням стосовно чогось.

Видавець та споживач намагаються максимізувати користь під час процесу споживання новин. Ми можемо прийняти той факт, що фейкові новини з'являються коли з точки зору видавництва домінує короткотривала користь, а з точки зору споживача домінує психологічна користь.

### **1.3. Підроблені новини в соціальних медіа**

#### **Зловмисні облікові записи**

Багато користувачів соціальних медіа є правомірними, також є група користувачів, що є зловмисниками, а в деяких випадках навіть не справжніми людьми. Через те, що створення аккаунту в соціальних медіа не потребує багато зусиль, це заохочує створення шкідливих облікових записів, наприклад соціальних ботів, користувачів кіборгів та тролів. Соціальний бот посилається на аккаунт в соціальних медіа, який контролюється комп'ютерним алгоритмом для автоматичного створення контенту та взаємодії з іншими людьми, ботами. Соціальні боти можуть стати шкідливими, розробленими спеціально з метою заподіяння шкоди, наприклад, маніпулювання та розповсюдження підроблених новин у соціальних мережах.

#### **Тролінг**

Троль – це людина, яка навмисно намагається викликати конфлікт, ворожість або суперечки в соціальних мережах Інтернету. Платформи, що орієнтовані на тролів, можуть включати розділи коментарів на YouTube, форуми або чати.

Тролі часто використовують запальні повідомлення, щоб викликати емоційні реакції людей, порушуючи громадянську дискусію. Тролінг відбувається там, де люди можуть вільно ділитися своїми думками.

Тролінг впливає на поведінку, настрої людей та на контекст онлайн-дискусій, що дозволяє легко поширювати підроблені новини серед "нормальних" онлайн-спільнот. Мета тролінгу - спровокувати людей, в результаті чого виникають внутрішні негативні емоції, такі як гнів і страх, сумніви, недовіра та нераціональна поведінка.

## **Кіборги**

Нарешті, кіборги - це користувачі, що можуть поширювати підроблені новини, поєднуючи автоматизовані дії з людським вкладом. Зазвичай облікові записи кіборгів реєструються людиною як “камуфляж” і встановлюють автоматизовані програми для здійснення діяльності в соціальних медіа. Ці зловмисні акаунти стають потужним джерелом для розповсюдження підроблених новин в соціальних медіа.

## **Ехо-ефект камери**

Соціальні медіа надають нову парадигму створення та споживання інформації для користувачів. Процес пошуку та споживання інформації змінюється від опосередкованої форми до безпосереднього способу. Споживачі вибірково піддаються певним новинам через спосіб стрічки новин, що з'явилася на їх домашній сторінці в соціальних мережах, посилюючи психологічні виклики щодо розвіяння фальшивих новин. Наприклад, користувачі Facebook завжди стежать за однодумцями і, таким чином, отримують новини, які просувають їх підписники. Тому користувачі в соціальних мережах, як правило, утворюють групи, що містять однодумців, де вони потім поляризують свою думку, що призводить до камерного ефекту. Ефект камери полегшує процес, завдяки якому люди споживають і вірять фальшивим новинам через наступні психологічні фактори:

- соціальна довіра, означає, що люди будуть вірити інформації, якщо хтось інший вірить, особливо коли недостатньо інформації і доступу до правдивого джерела;
- евристичність, означає, що споживачі можуть віддавати перевагу інформації, яку вони чують часто, навіть якщо це фальшиві новини.

В результаті ефект ехо-камери створює сегментовані, однорідні спільноти з дуже обмеженою інформаційною екосистемою. Дослідження показують, що однорідні спільноти стають основною рушійною силою розповсюдження недостовірної інформації, що ще більше посилює поляризацію.

#### **1.4. Висновки до розділу**

Інформаційні вкидання існують досить давно, з того моменту, коли з'явилися джерела поширення інформації. Фейкові новини - це будь-які джерела інформації, що навмисно допускають недостовірну інформацію і можуть ввести читачів у оману.

Наступні поняття не є підробленими новинами згідно визначення: сатиричні новини, чутки, теорії змов, дезінформація, містифікації. Під час розповсюдження інформації є два види ключових гравців: видавець та споживач. Є два основні фактори, які роблять споживачів природно вразливими до підроблених новин: наївний реалізм, підтверджувальне упередження. Є різні джерела поширення інформаційних вкидань з своїм рівнем автоматизації: зловмисні облікові записи, тролі, кіборги, ехо-ефект камери.

## РОЗДІЛ 2

### ВИЗНАЧЕННЯ ФЕЙКОВИХ НОВИН

У попередньому розділі було визначено концептуальну характеристику традиційних фейкових новин. На основі цієї характеристики далі визначено проблему фейкових новин та запропоновані підходи їх виявлення.

#### 2.1. Визначення проблеми

У цьому підрозділі представлено математичні подробиці формулювання виявлення підроблених новин у соціальних медіа. Зокрема, визначено ключові компоненти підроблених новин, а потім представлено формальне визначення виявлення фейкових новин.

Основні позначення:

- Стаття новин складається з двох основних компонентів: видавець і зміст. Видавець  $\vec{p}_a$  включає в себе набір функцій профілю для опису автора, таких як ім'я, домен, вік та інші атрибути. Зміст  $\vec{c}_a$  складається з набору атрибутів які представляють статтю та містять заголовок, текст, зображення тощо.
- Ми також визначаємо взаємодії з новинами як набір кортежів  $\mathcal{E} = \{e_{it}\}$ , щоб представити процес поширення новин з плином часу серед  $n$  користувачів  $\mathcal{U} = \{u_1, u_2, \dots, u_n\}$  та їх відповідні повідомлення  $\mathcal{P} = \{p_1, p_2, \dots, p_n\}$  у соціальних медіа щодо статті новин  $a$ . Кожен  $e_{it} = \{u_i, p_i, t\}$  означає, що користувач  $u_i$  розповсюджує статтю новин  $a$ , використовуючи  $p_i$  за час  $t$ . Зауважте, що якщо стаття ще не була задіяна, то  $t = \text{false}$  та  $u_i$  представляє видавництво.

## Визначення 2 (Виявлення інформаційних вкидань)

Дано залучення в соціальних новинах  $\mathcal{E}$  серед користувачів  $n$  в статтях  $a$ . Завдання виявлення підроблених новин - передбачити, чи буде стаття “ $a$ ” фейком чи ні, тобто  $\mathcal{F} : \mathcal{E} \rightarrow \{0, 1\}$  такий як,

$$\mathcal{F}(a) = \begin{cases} 1, & \text{якщо це фейкові новини} \\ 0, & \text{в іншому випадку} \end{cases} \quad (2.1)$$

де  $\mathcal{F}$  - функція передбачення.

Виявлення підроблених новин це проблема бінарної класифікації з наступної причини: інформація, що публікується видавцем, може бути спотворена або ж ні.

Фреймворк для виявлення фейкових новин включає дві фази: функції вилучення та побудова моделі. Вилучення являє собою представлення змісту новин та пов'язаних даних у формальній математичній структурі та побудова моделі машинного навчання, щоб краще диференціювати підроблені новини від реальних.

## 2.2. Оцінка ефективності методів виявлення фейкових новин

В даному розділі розглянуто як оцінити результати роботи алгоритмів виявлення підроблених новин, доступні набори даних та метрики оцінки методів виявлення фейкових новин.

### 2.2.1 Набори даних

Інтернет-новини можна збирати з різних джерел: сторінки інформаційних агентств, пошукові системи та веб-сайти. Однак ручне визначення правдивості новин є складним завданням, як правило вимагає аналітиків з доменною експертизою, які виконують ретельний аналіз. Як правило, дані новин можуть бути зібрані такими

способами: журналістами-експертами, веб-сайти швидкої перевірки, детектори. Однак, немає узгоджених наборів даних для виявлення підроблених новин.

Загальнодоступні набори даних наведені нижче:

- BuzzFeedNews: Цей набір даних містить повний зразок новин, опублікованих у Facebook дев'ятьма агенціями протягом тижня перед американськими виборами 2016 року.  
Кожен пост та пов'язана стаття перевірялися п'ятьма журналістами BuzzFeed.
- LIAR: Цей набір даних зібраний за допомогою веб-сайту швидкої перевірки PolitiFact. Він включає 12 836 постів, які відібрані із випусків новин, телебачення або радіо інтерв'ю. Рівень правдивості оцінювався за допомогою міток: фальшиві, ледь правдиві, напівправдиві, здебільшого правдиві і правдиві.
- BS Detector: Цей набір даних зібраний за допомогою браузерного розширення, що називається "BS Detector", розроблений для перевірки правдивості новин. Він здійснює пошук всіх ненадійних посилань на джерела. Рівень правдивості оцінюється самою програмою, а не людиною.
- CREDBANK: Цей набір даних включає приблизно 60 мільйонів твітів, які охоплюють 96 днів, починаючи з жовтня 2015 року.

### 2.2.2 Метрики оцінки

Для оцінки ефективності алгоритмів виявлення фальшивих новин використовувались різні показники оцінки. У цьому підрозділі описано найбільш широко використовувані метрики для виявлення фейкових новин. Більшість існуючих підходів рахують проблему фейкових новин як проблему класифікації, яка передбачає, чи є фальшива стаття:

- Істинно позитивний (TP): коли фейкові новини нотуються як фальшиві новини;
- Істинно негативний (TN): коли справжні новини нотуються як правдиві новини;
- Хибний негатив (FN): коли справжні новини нотуються як фальшиві новини;
- Хибний позитивний (FP): коли фейкові новини нотуються як вірні новини.

Сформулювавши проблему класифікації можна визначити наступні показники,

$$\begin{aligned}
 Precision &= \frac{|TP|}{|TP| + |FP|} \\
 Recall &= \frac{|TP|}{|TP| + |FN|} \\
 F1 &= 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \\
 Accuracy &= \frac{|TP| + |TN|}{|TP| + |TN| + |FP| + |FN|}
 \end{aligned}
 \tag{2.2.2}$$

Ці метрики широко використовуються в машинному навчанні. Точність вимірює схожість між передбачуваними фейковими новинами та реальними підробленими новинами. Чим більші показники Precision, Recall, F1 та Accuracy тим краща продуктивність.

### 2.3. Виявлення спаму та ботів

Виявлення спамерів у соціальних мережах має на меті знайти зловмисних користувачів, які займаються розповсюдженням реклами, поширенням порнографії, розповсюдженням вірусів та фішингом, останнім часом привернула широку увагу. Існуючі підходи для виявлення спамерів в основному покладається на вилучення деталей з інформації про користувачів та про соціальні мережі. Крім того, збільшилася кількість фейкової інформації, що розповсюджують боти, оскільки пости не перевіряються на правдивість. Головна проблема ботів полягає в тому, що вони можуть створювати помилкове враження, що інформація є дуже популярною та схваленою багатьма людьми, що дає можливість ефекту камери для поширення фальшивих новин.

## 2.4. Висновки до розділу

Стаття новин складається з двох основних компонентів: видавець і зміст. Видавець включає в себе: ім'я, домен, вік та інші атрибути. Зміст складається з набору атрибутів, які представляють статтю та містять заголовок, текст, зображення тощо. З іншої сторони виявлення підроблених новин це проблема бінарної класифікації з наступної причини: інформація, що публікується видавцем, може бути спотворена або ж ні. Інтернет-новини можна отримати з різних джерел: сторінки інформаційних агентств, пошукові системи та веб-сайти. Загальнодоступні набори даних: BuzzFeedNews, LIAR, BS Detector, CREDBANK. Найбільш широко використовувані метрики для виявлення фейкових новин: істинно-позитивний(TP), істинно-негативний(TN), хибний-негативний(FN), хибний-позитивний(FP). Виявлення спамерів – це процес виявлення зловмисних користувачів, які займаються розповсюдженням реклами, поширенням порнографії, розповсюдженням вірусів та фішингом.

## РОЗДІЛ 3

### ВИБІР ТЕХНОЛОГІЙ ТА АЛГОРИТМІВ ВИРІШЕННЯ ПРОБЛЕМИ ПРОГРАМИ

#### 3.1. Векторна модель

Векторна космічна модель - це алгебраїчна модель, що містить два кроки: на першому кроці представляємо текстові документи у векторах, а на другому кроці перетворюємо їх на числа, щоб можна було застосовувати будь-які методи виведення тексту, такі як пошук інформації, вилучення інформації, фільтрація інформації тощо.

##### *Презентація двох документів в векторній моделі*

- Документ 1: Полюбила чорнобрива козака дівчина.
- Документ 2: Полюбила світлоброва поляка дівчина.
- Запит: козака.

##### *Представлення документів у вигляді векторів*

Цей крок включає розбиття кожного документу на слова, застосовуючи етапи попередньої обробки, такі як видалення пустих символів, пунктуацій, спеціальних символів тощо. Після попередньої обробки, документи представляються як вектори слів.

Нижче наведено зразкове подання векторів документів.

- Документ 1: (Полюбила, чорнобрива, козака, дівчина)
- Документ 2: (Полюбила, світлоброва, поляка, дівчина)
- Запит: (козака)

Наступним кроком є подання створених вище векторів у числовий формат, відомий як матриця термінового документа.

Термінальна матриця документа - це спосіб подання векторів документів у форматі матриці, в якому кожен рядок представляє елемент з вектора, а стовпці

представляють вектори документів. Якщо елемент присутній у документі, то відповідне значення комірки містить 1, якщо елемента немає в документі, то значення комірки містить 0.

Після створення матриці термінового документа обчислюються ваги елементів матриці для всіх документів.

Метод, відомий як tf-idf дає більш високу вагу словам, які зустрічаються частіше в документі, але рідше трапляється у всіх інших документах:

$$\text{Tf-idf} = \text{tf} \times \text{idf} \quad (2.1)$$

Де tf = частота елемента;

idf = обернена частота документа;

idf =  $\log(N / df)$ , де df - частотна кількість документів;

Таблиця 3.1

Загальна кількість документів

Загальна кількість документів
2

Таблиця 3.2

Термін матриця документа

Матриця термінів документів			
слова\документи	Документ1	Документ2	Запит
полюбила	1	1	0
чорнобрива	1	0	0
світлоброва	0	1	0
козака	1	0	1
поляка	0	1	0
дівчина	1	1	0

Таблиця 3.3

## Обернена частота документа

Обчислення <u>Idf</u>	
Частота документів(df)	<u>Idf</u> – $\log(N/df)$
2	0
1	0.30103
1	0.30103
1	0.30103
1	0.30103
2	0

Примітка: idf обчислюється за допомогою логарифма зворотної частки між кількістю документів та частотою документа.

Таблиця 3.4

## Tf-idf обчислення

Матриця термінів документів з <u>tf-idf</u>			
слова\документи	Документ1	Документ2	Запит
полюбила	0	0	0
чорнобрива	0.30103	0	0
світлоброва	0	0.30103	0
козака	0.30103	0	0.30103
поляка	0	0.30103	0
дівчина	0	0	0

Існує багато інших варіантів того, як можна обчислюємо термін-частоту (tf) та зворотню частоту документа (idf). Нижче зображено інші варіанти tf та idf.

Таблиця 3.5

## Варіації частотної частоти

Variants of term frequency (TF) weight

weighting scheme	TF weight
binary	0, 1
raw count	$f_{t,d}$
term frequency	$f_{t,d} / \sum_{t' \in d} f_{t',d}$
log normalization	$1 + \log(f_{t,d})$
double normalization 0.5	$0.5 + 0.5 \cdot \frac{f_{t,d}}{\max_{\{t' \in d\}} f_{t',d}}$
double normalization K	$K + (1 - K) \frac{f_{t,d}}{\max_{\{t' \in d\}} f_{t',d}}$

Таблиця 3.6

## Зворотні зміни частоти документа

Variants of inverse document frequency (IDF) weight

weighting scheme	IDF weight ( $n_t =  \{d \in D : t \in d\} $ )
unary	1
inverse document frequency	$\log \frac{N}{n_t} = -\log \frac{n_t}{N}$
inverse document frequency smooth	$\log \left( 1 + \frac{N}{n_t} \right)$
inverse document frequency max	$\log \left( \frac{\max_{\{t' \in d\}} n_{t'}}{1 + n_t} \right)$
probabilistic inverse document frequency	$\log \frac{N - n_t}{n_t}$

### 3.2. Заходи подібності. Косинусна схожість

Косинус подібності - це коефіцієнт подібності двох не нульових векторів у предгільбертовому просторі, який обчислюється як косинус кута між ними. Обчисливши косинус кута між кожним вектором документу та вектором запиту, можна отримати його схожість. Це оцінка напрямку, а не величини: два вектори з однаковим напрямком мають косинус подібності 1, а два вектори, які утворюють кут  $90^\circ$  один відносно одного, мають подібність 0, а два діаметрально направлені вектори мають подібність -1, незалежно від їх довжини. Косинус подібності часто використовують в позитивному просторі, для якого результат обмежений проміжком  $[0,1]$ . Щоб знайти відповідний документ до терміну запиту, обчислюється показник подібності між кожним вектором документа та вектором запитом, застосувавши подібність косинуса. Нарешті, документи, що мають великі коефіцієнти подібності, вважатимуться схожими до запиту. Кожен вектор документу представляє точку в векторному просторі. У наведеному нижче прикладі запит, документ 1 і документ 2 представляють три точки у векторному просторі. Тепер ми можемо порівняти запит з кожним документом, обчисливши косинус між ними.

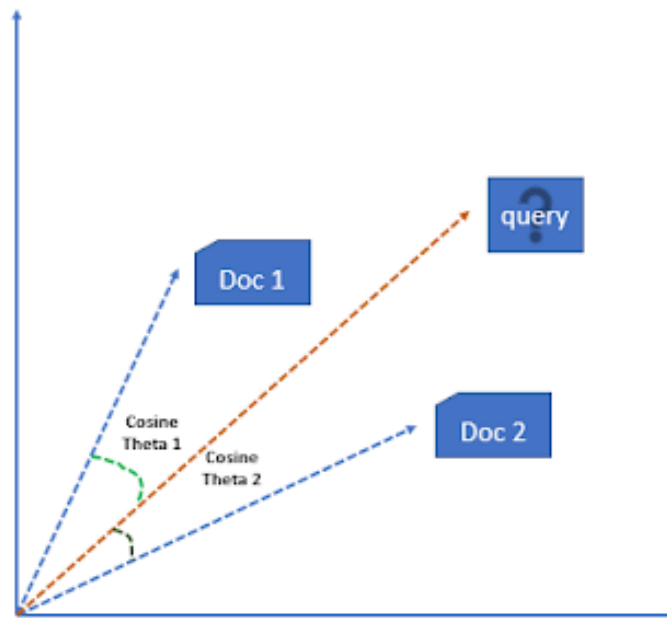


Рис. 3.1. Косинус між векторами

Крім косинусної подібності є інші варіанти визначення подібності векторів:

- Жакардова відстань;
- Розходження Кульбака-Лейблера;
- Евклідова відстань.

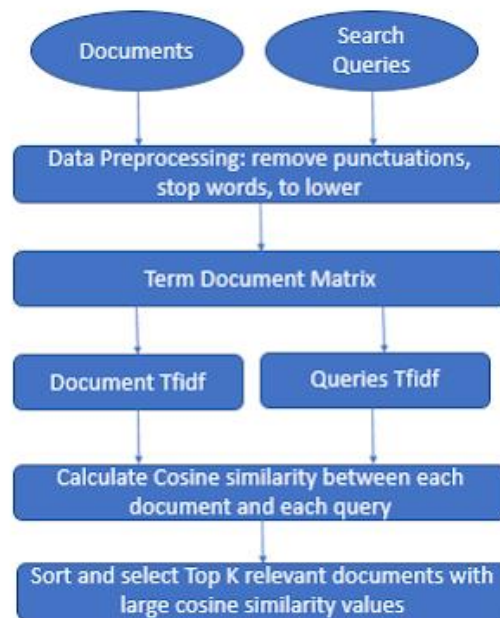


Рис. 3.2. Графічний алгоритм

### 3.3. Модель Word2vec. Методи CBOW та Skip-Gram

Word2vec - це набір моделей, що використовуються для представлення розподілених подань слів будь-якої множини  $S$ . Word2Vec (W2V) - це алгоритм, який приймає множину слів як вхідні дані та виводить векторне представлення для кожного слова.

Word2vec - це двошарова нейронна мережа, яка обробляє текст «векторизацією» слів. Його вхід - це текстовий корпус, а вихід - це набір векторів. Хоча Word2vec не є глибокою нейронною мережею, він перетворює текст у числову форму, яку можуть зрозуміти глибокі нейронні мережі.

Word2vec модель може бути застосована до генів, коду, лайків, музики, де можна побачити шаблони.

Мета Word2vec - це згрупувати вектори подібних слів разом у векторному просторі. Враховуючи достатню кількість даних, Word2vec може робити дуже точні здогадки про значення слова. Ці здогадки можна використовувати для встановлення слів-синонімів (наприклад, "хороший" - "добрий", що "комп'ютер" - "ноутбук"), або кластеризувати документи та класифікувати їх за темою. Ці кластери можуть стати основою пошуку, аналізу у таких різноманітних галузях, як наукові дослідження, юридичні відкриття, електронна комерція та управління відносинами з клієнтами.

Вихідні дані нейронної мережі – це словник, в якому слово – це ключ, а значення - це вектор, який можна обробляти алгоритмами машинного навчання для виявлення зв'язків між словами. Визначаючи косинус подібності, немає подібності, якщо кут дорівнюватиме 90, а повна подібність дорівнюватиме 0.

Word	Cosine distance
norway	0.760124
denmark	0.715460
finland	0.620022
switzerland	0.588132
belgium	0.585835
netherlands	0.574631
iceland	0.562368
estonia	0.547621
slovenia	0.531408

Рис. 3.3. Список слів пов'язаних зі словом "Швеція" за допомогою Word2vec, у порядку близькості

Алгоритм Word2Vec має два варіанта реалізації: Skip Gram та Common Bag Of Words (CBOW).

Метод CBOW приймає на вхід контекст кожного і намагається передбачити слово, що відповідає контексту.

CBOW намагається передбачити цільове слово, аналізуючи контекст оточуючих слів. Розглянемо речення: “It is a pleasant day”. Модель перетворює це речення на пари слів у формі - (контекстне слово, цільове слово). Користувач має встановити розмір вікна. Якщо вікно для контекстного слова дорівнює 2, тоді пари слів будуть виглядати так: ([it, a], is), ([it, pleasant], a), ([a, day], pleasant). За допомогою цих пар слів модель намагається передбачити цільове слово, що розглядається як контекстне слово.

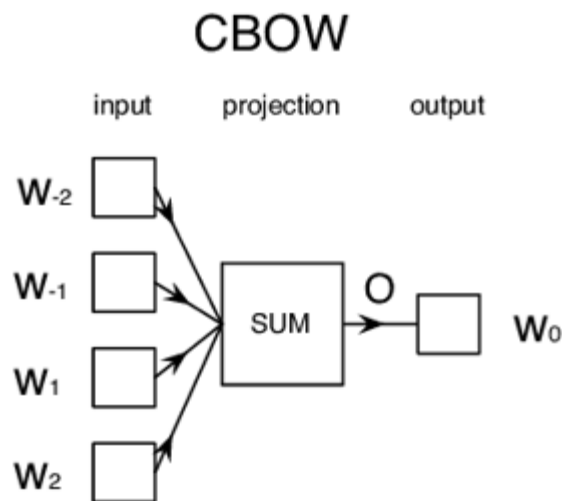


Рис. 3.4. CBOW модель

Існує також інша модель, за допомогою якої можна зробити те ж саме - Skip-Gram модель.

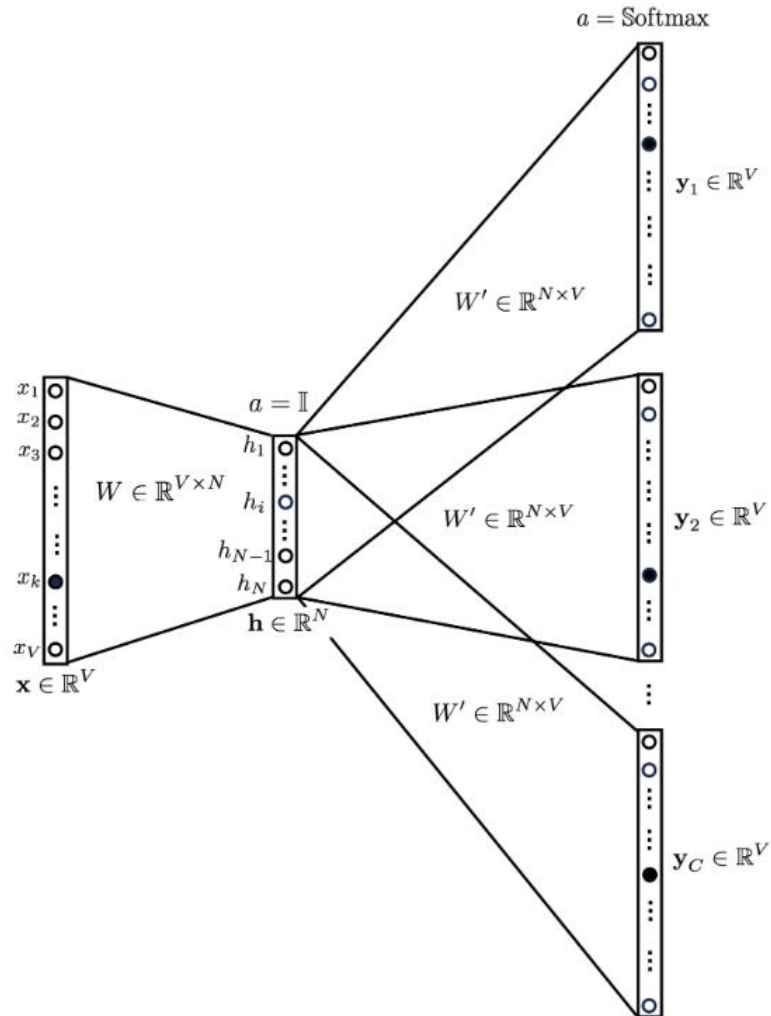


Рис. 3.6. Skip-Gram модель

Skip-gram використовується для визначення контекстного слова для будь-якого цільового слова. Skip-gram модель – це протилежність моделі CBOW. Дана модель отримує на вхід цільове слова, контекстні слова – вихідні дані моделі. Контекстних слів може бути декілька, що ускладнює цю проблему.

У обох моделях є свої переваги та недоліки. Skip Gram добре працює з невеликою кількістю даних та краще відображає рідкісні слова.

З іншого боку, CBOW швидше та краще відображає слова, що трапляються частіше.

### 3.4. Бібліотека Tkinter

Tkinter - це багатоплатформна графічна бібліотека інтерфейсів на основі засобів Tk для мови програмування Python. Бібліотека Tkinter не реалізує власний інтерфейс до бібліотеки Tk, а забезпечує конвертування звернень Python в звернення Tcl - мови, яка тісно інтегрована з Tk. Таким чином Tkinter є обгорткою для Tcl/Tk. В графічний інтерфейс користувача входять всі вікна, кнопки, текстові поля для вводу, скроллери, списки, флажки та ін. Через них користувач взаємодіє та керує програмою.

### 3.5. Класифікатори

Наївний баєсів класифікатор – це набір алгоритмів класифікації, що базуються на теоремі Баєса для визначення ймовірності приналежності спостереження (елемента вибірки) до одного з класів при припущенні (наївному) незалежності змінних.

За допомогою Байєсів класифікатора можна визначити ймовірність приналежності спостереження до класу.

Логістична регресія - це алгоритм класифікації. Він використовується для прогнозування двійкового результату на основі набору незалежних змінних.

Бінарний результат - це коли існує лише два можливих сценарії - або подія трапляється, або ні. Незалежні змінні - це ті змінні або фактори, які впливають на результат.

Нехай є деяка випадкова величина  $Y$ , що може набувати лише двох значень, які, як правило, позначаються цифрами 0 і 1. Нехай ця величина залежить від деякої множини змінних  $x = (1, x_1, \dots, x_n)^T$ . Залежність  $Y$  від  $x_1, \dots, x_n$  можна визначити ввівши додаткову змінну  $u^*$ , де

$$y^* = \theta^T x = \theta_0 + \theta_1 x_1 + \dots + \theta_n x_n + \varepsilon$$

$$Y = \begin{cases} 0, & y^* \leq 0 \\ 1, & y^* > 0 \end{cases} \quad (3.1)$$

Логістична регресія - це тип аналізу, який слід використовувати під час роботи з двійковими даними. Наприклад, двійковими змінними можна вважати - "так" або "ні".

Random forest – це метод машинного навчання для класифікації, регресії та інших задач, що базуються на деревах прийняття рішень під час тренування моделі і визначають моду для класів або усереднений прогноз побудованих дерев.

Метод опорних векторів(SVM) - це машинний алгоритм, котрий навчається на прикладах та використовується для класифікації об'єктів. SVM може також розрізняти аварійний режим роботи електромеханічної системи та класифікувати його за наявності попередніх досліджень, можливих за технологічними вимогами режимів роботи. Даний підхід розкриває великі можливості для побудови адаптивних систем автоматичного керування. SVM базується на математичній сутності – алгоритм максимізації деякої математичної функції відносно наявного набору даних. SVM має чотири ключові поняття: відділяюча гіперплощина, гіперплощина максимальної межі, м'яка межа, функція ядра. Відділяюча гіперплощина є математичною сутністю, що відділяє між собою класи об'єктів з однаковими ознаками.

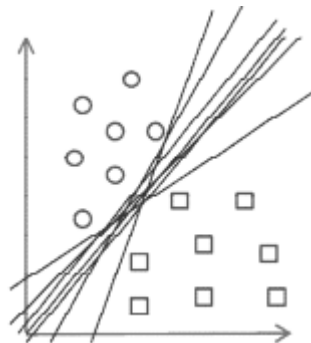


Рис. 3.7. Метод опорних векторів

Ця процедура може бути екстрапольована математично до вимірів, значно вищих за третій. Термін, що визначає лінію, котра відділяє елементи різних класів - багатовимірна гіперплощина. Спосіб, яким можна провести відділяючу гіперплощину за методом SVM не є унікальним. Завжди існує багато різних можливостей розташування гіперплощини.

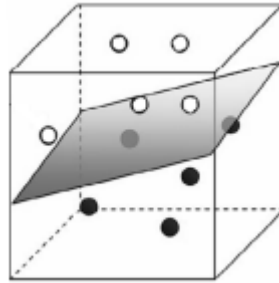


Рис. 3.8. Приклад відділяючої площини

Стохастичний градієнтний спуск - це ітеративний метод оптимізації градієнтного спуску за допомогою стохастичного наближення. Використовується для прискорення пошуку цільової функції за допомогою обмеженого за розміром тренувального набору, який вибирається випадкового при кожній ітерації.

### 3.6. Рядки

У комп'ютерному програмуванні рядок традиційно є послідовністю символів, або буквальною константою, або змінною. Рядок, як правило, розглядається як тип даних і часто реалізується як структура даних масиву байтів (або символів), що зберігає послідовність елементів (як правило, символів), використовуючи кодування символів. Рядок також може позначати більш загальні масиви чи інші послідовності (або список) типів та структур даних.

Залежно від мови програмування та використовуваного типу даних, змінна, оголошена рядком, може спричинити статичне розподілення пам'яті на заздалегідь

задану максимальну довжину, або використовувати динамічний розподіл, щоб вона могла вміщувати змінну кількість елементів. Коли рядок відображається буквально у вихідному коді, він зазначається як буквенний рядок або анонімний рядок. У формальних мовах, які використовуються в математичній логіці та теоретичній інформатиці, рядок - це кінцева послідовність символів, яка вибирається з набору, який називається алфавітом.

### **Індексація**

Рядки - це впорядковані послідовності символічних даних. Індексація дозволяє отримати доступ до окремих символів у рядку безпосередньо за допомогою числового значення. Індексація рядків відбувається на основі нуля: перший символ у рядку має індекс 0, наступний - 1 тощо.

### **Конкатенація**

У формальній теорії мови та комп'ютерному програмуванні конкатенація рядків - це операція з'єднання рядків символів від кінця до кінця. Наприклад, з'єднання "капу" і "ста" - це "капуста".

### **Регістр**

Однією з найпоширеніших речей, яку роблять розробники програмного забезпечення, є «нормалізація» тексту для цілей порівняння. Розробники можуть порівнювати рядки в залежності від регістру. Unicode визначає властивості верхнього, нижнього та титульного регістру для символів, а також спеціальні випадки, які впливають на використання тексту певною мовою.

Багато розробників вважають, що порівняння з урахуванням регістру досягається шляхом зіставлення обох рядків у порівнянні з верхнім чи нижнім регістром, а потім порівнянням отриманих байтів.

## **Split та Join**

Операції Join та Split виконуються над рядками для з'єднання або розділення рядка. Розділити рядок може бути часом дуже корисно, особливо коли вам потрібні лише певні частини рядка. Простий, але ефективний приклад - це розділення ім'я та прізвища людини.

## **Порівняння**

Порівняння рядків - це процес аналізу та виведення відмінностей чи подібностей між двома рядками.

### 3.7. Висновки до розділу

Вибір методів для виконання задачі знаходження інформаційних вкидань дуже важливий так як похибка в обчисленні неприпустима. Векторна модель - це модель, що містить два кроки: на першому кроці представляємо текстові документи у векторах, а на другому кроці перетворюємо їх на числа, щоб можна було застосовувати будь-які методи виведення тексту, такі як пошук інформації, вилучення інформації, фільтрація інформації тощо.

Наступним кроком є подання створених векторів у числовий формат, відомий як матриця термінового документа. Існує багато інших варіантів того, як можна обчислюємо термін-частоту (tf) та зворотню частоту документа (idf).

Косинус подібності - це коефіцієнт подібності двох не нульових векторів у гільбертовому просторі, який обчислюється як косинус кута між ними. Обчисливши косинус кута між кожним вектором документа та вектором запиту, можна отримати його схожість. Щоб знайти відповідний документ до терміну запиту, обчислюється показник подібності між кожним вектором документа та вектором запитом, застосувавши подібність косинуса. Нарешті, документи, що мають великі коефіцієнти подібності, вважатимуться схожими до запиту.

Word2vec - це моделі, що використовуються для представлення розподілених подань слів будь-якої множини  $S$ . Алгоритм приймає множину слів як вхідні дані та виводить векторне представлення для кожного слова. Він має два варіанта реалізації: Skip Gram та Common Bag Of Words (CBOW).

Метод CBOW приймає на вхід контекст кожного і намагається передбачити слово, що відповідає контексту. Існує також інша модель, за допомогою якої можна зробити те ж саме - Skip-Gram модель.

Після того, коли дані згруповані та готові до подальшої обробки, виконується ряд задач за допомогою класифікаторів: наївний баєсів класифікатор, логістична регресія, бінарна класифікація, random forest(випадковий ліс),метод опорних векторів(SVM), стохастичний градієнтний спуск.

## РОЗДІЛ 4

### РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Після огляду існуючих методів визначення фейкових новин було прийнято рішення вирішено використати методи обробки природних мов, алгоритми машинного навчання та бібліотеки мови програмування Python.

Джерелом даних, що використовується для цього проекту, є набір даних LIAR, який містить 3 файли з форматом .tsv для тестування, навчання та перевірки. Нижче наведено опис файлів даних, що використовуються для цього проекту.

#### 4.1. Файлова система програми

##### **DataPrep.py**

Цей файл містить усі функції, що необхідні для обробки вхідних документів та текстів. Спочатку виконується зчитування файлів даних, потім виконується попередня обробка: перевірка якості даних, відсутність значення тощо.

##### **FeatureSelection.py**

У цьому файлі використовуються методи видалення особливостей за допомогою бібліотек мови програмування Python. Для визначення особливостей використовували такі методи, як “bag-of-words”, “n-gram” та tf-idf для зважування слів у контексті.

##### **Classifier.py**

В цьому файлі знаходяться всі класифікатори для виявлення інформаційних вкидань. Функції подаються в різних класифікаторах: наївний баєсів, логістичної регресії, опорних векторів, стохастичного градієнту, випадковий ліс. Кожен із видалених творів використовувався у всіх класифікаторах. Після виконання роботи всіх класифікаторів, дві найкращі моделі були обрані в якості кандидатів для класифікації підроблених новин.

### **Prediction.py**

Найкращим класифікатором стала логістична регресія, яка збережена під ім'ям `final_model.sav`. Цей класифікатор використовується файлом `prediction.py` для визначення фейкових новин.

### **Main\_window.py**

Цей файл відповідає за відображення графічного інтерфейсу користувача. Він “спілкується” з `prediction.py` за допомогою подій. Для відображення графічного інтерфейсу була використана бібліотека Tkinter.

## **4.2 Архітектурний шаблон програми**

MVC(Model-view-controller) - це спосіб організації коду програми, який передбачає виділення шарів, що виконують певні задачі. Один шар відповідає за дані додатка, інший відповідає за графічний інтерфейс, а третій контролює роботу програми.

### **Компоненти MVC:**

- Модель - це компонент, що відповідає за дані, а також визначає структуру та розрахунки програми.
- Вид - це компонент, що відповідає за взаємодію з користувачем. Тобто, код компонента визначає зовнішній вигляд додатку і способи його використання.
- Контролер - це компонент, що відповідає за зв'язок між моделлю та виглядом. Код компонента визначає як програма реагує на дії користувача.

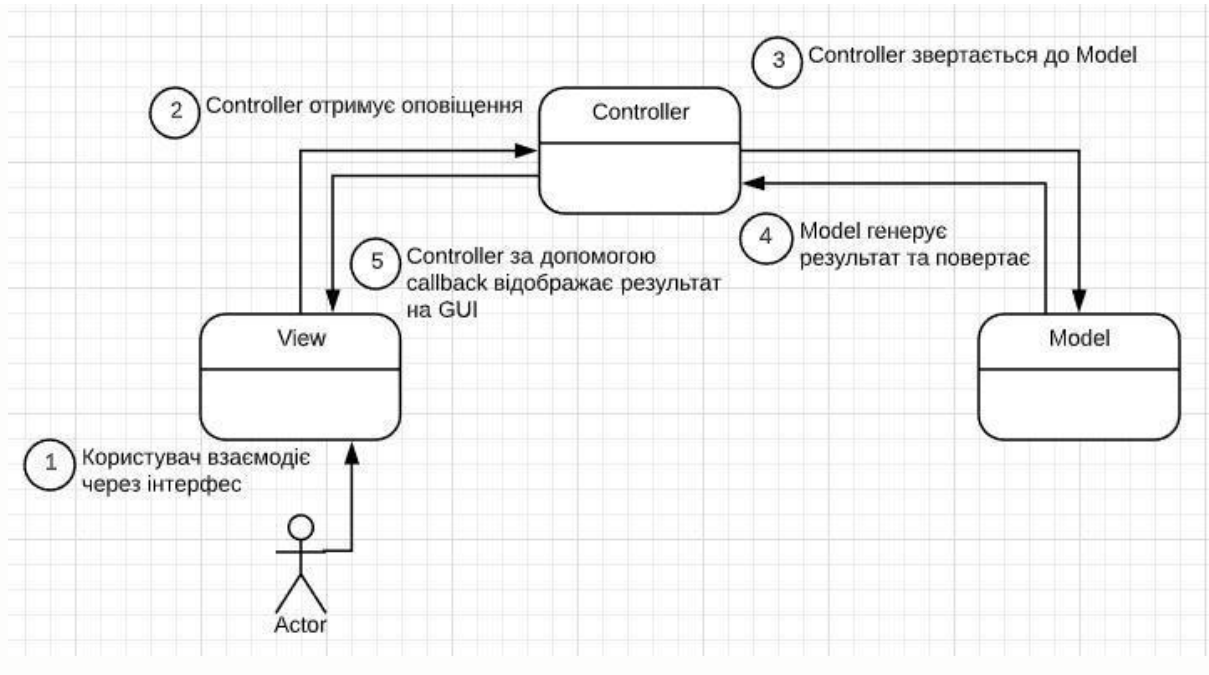


Рис. 4.1 MVC діаграма розробленої програми

### Використання делегатів з подіями

Події - це дії користувача, такі як натискання клавіш, переміщення миші сповіщення, створені системою тощо. Програми повинні реагувати на події, коли вони відбуваються. Події використовуються для міжпроцесорного спілкування.

Події оголошуються в класі та асоціюються з обробниками подій, використовуючи делегати в межах одного класу чи якогось іншого класу. Клас, що містить подію, використовується для публікації події і називається клас видавця. Деякий інший клас, який приймає цю подію, називається абонентським класом. Події використовують модель видавця-підписника.

Видавець - це об'єкт, який містить визначення події та делегата. Ініціалізація делегата подій також визначається в цьому об'єкті. Об'єкт класу видавця викликає подію та повідомляє про це іншим об'єктам.

Абонент - це об'єкт, який приймає подію та забезпечує обробку подій. Делегат класу видавця викликає метод (обробник подій) класу підписника.

Події використовуються для взаємодії між View та Controller. При натисканні на клавішу “Calculate” виконується метод `button_calculate_click()`, який в свою чергу відправляє повідомлення всім підписникам на подію. В результаті цього у підписників виконується метод `on_calculate_button()`. Той в свою чергу взаємодіє з Model та визиває `callback` метод, що відображає результат на екрані (`callback` - це виконуваний код, який передається як аргумент іншому коду).

### 4.3. Висновки до розділу

В результаті вивчення методів обробки та аналізу тексту, для виявлення інформаційних вкидань, було розроблено систему на основі мови програмування Python. Реалізовано найефективніші методи обробки та аналізу тексту, алгоритми машинного навчання, класифікації для поставленої задачі.

Джерелом даних, що використовується для цього проекту, є набір даних LIAR. Liar - набір даних зібраний за допомогою веб-сайту швидкої перевірки PolitiFact. Він включає 12 836 постів, які відібрані із випусків новин, телебачення або радіо інтерв'ю. Файлова система програми: DataPrep.py, FeatureSelection.py, Classifier.py, Prediction.py, Main\_window.py. Система була розроблена на основі архітектурного шаблону MVC(model-view-controller). Це спосіб організації коду програми, який передбачає виділення шарів, що виконують певні задачі. Один шар відповідає за дані додатка, інший відповідає за графічний інтерфейс, а третій контролює роботу програми.

Взаємодія між об'єктами система побудована на основі делегатів з подіями. Це дії користувача, такі як натискання клавіш, переміщення миші сповіщення, створені системою тощо.

## ВИСНОВКИ

Останнім часом через бурхливий розвиток соціальних мереж інформаційні вкидання з'являються та швидко розповсюджуються по світу в комерційних та політичних цілях. Користувачі соціальних мереж можуть легко заразитися цими підробленими новинами. Важливою метою для удосконалення достовірності інформації в соціальних мережах є своєчасне їх виявлення.

Соціальні медіа для споживання новин - це двосічний меч. З одного боку, низька вартість, простий доступ та швидке розповсюдження інформації спонукають людей шукати та споживати новини із соціальних медіа. З іншого боку, це дозволяє широке розповсюдження "фальшивих новин", тобто низькоякісних новин, із навмисно недостовірною інформацією. Широке поширення підроблених новин може мати надзвичайно негативний вплив на індивіди та суспільство. Тому виявлення фальшивих новин в інтернеті останнім часом стало популярним дослідженням, яке привертає величезну увагу.

Зі зростанням популярності соціальних медіа все більше і більше людей споживають новини із соціальних медіа замість традиційних засобів інформації. Однак соціальні медіа також використовуються для поширення фейкових новин, що має сильний негативний вплив на окремих користувачів та в загальному суспільство. У цій роботі було досліджено проблему підроблених новин та реалізовану програмне забезпечення для їх виявлення. На етапі характеристики введено основні поняття та принципи фальшивих новин як у традиційних, так і в соціальних медіа. На фазі виявлення розглянуто наявні підходи та реалізовано програмне забезпечення для виявлення інформаційних вкидань. Також описано набори даних, показники оцінки та перспективні напрямки досліджень фальшивих новин.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Sadia Afroz, Michael Brennan, and Rachel Greenstadt. Detecting hoaxes, frauds, and deception in writing style online. In ISSP'12.
2. Hunt Allcott and Matthew Gentzkow. Social media and fake news in the 2016 election. Technical report, National Bureau of Economic Research, 2017.
3. Solomon E Asch and H Guetzkow. Effects of group pressure upon the modification and distortion of judgments. *Groups, leadership, and men*, pages 222–236.
4. Yimin Chen, Niall J Conroy, and Victoria L Rubin. Misleading online content: Recognizing clickbait as false news. In *Proceedings of the 2015 ACM on Workshop on Multimodal Deception Detection*, pages 15–19. ACM, 2015.
5. Xia Hu, Jiliang Tang, Huiji Gao, and Huan Liu. Social spammer detection with sentiment information. In *ICDM'14*.
6. Arkaitz Zubiaga, Ahmet Aker, Kalina Bontcheva, Maria Liakata, and Rob Procter. Detection and resolution of rumours in social media: A survey. arXiv preprint arXiv:1704.00656, 2017.
7. Suhang Wang, Jiliang Tang, Charu Aggarwal, and Huan Liu. Linked document embedding for classification.
8. Eugenio Tacchini, Gabriele Ballarin, Marco L Della Vedova, Stefano Moret, and Luca de Alfaro. Some like it hoax: Automated fake news detection in social networks. arXiv preprint arXiv:1704.07506, 2017.
9. Jiliang Tang, Yi Chang, and Huan Liu. Mining social media with social theories: a survey. *ACM SIGKDD Explorations Newsletter*, 15(2):20–29, 2014.
10. Amos Tversky and Daniel Kahneman. Advances in prospect theory: Cumulative representation of uncertainty. *Journal of Risk and uncertainty*, 5(4):297–323, 1992.
11. ДСТУ 3008-2015: Державний стандарт України «Документація. Звіти у сфері науки і техніки. Структура і правила оформлення». - Київ:ДП «УкрНПНЦ», 2016. – 25 с.

12. Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.
13. Kyumin Lee, James Caverlee, and Steve Webb. Uncovering social spammers: social honeypots+ machine learning. In SIGIR’10.
14. Amr Magdy and Nayer Wanas. Web-based statistical fact checking of textual documents. In Proceedings of the 2nd international workshop on Search and mining user-generated contents, pages 103–110. ACM, 2010.
15. Filippo Menczer. The spread of misinformation in social media. In WWW’16.
16. Kai Shu, Suhang Wang, Jiliang Tang, Reza Zafarani, and Huan Liu. User identity linkage across online social networks: A review. *ACM SIGKDD Explorations Newsletter*, 18(2):5–17, 2017.
17. Jiliang Tang, Yi Chang, and Huan Liu. Mining social media with social theories: a survey. *ACM SIGKDD Explorations Newsletter*, 15(2):20–29, 2014.
18. Suhang Wang, Jiliang Tang, Charu Aggarwal, and Huan Liu. Linked document embedding for classification. In CIKM’16.

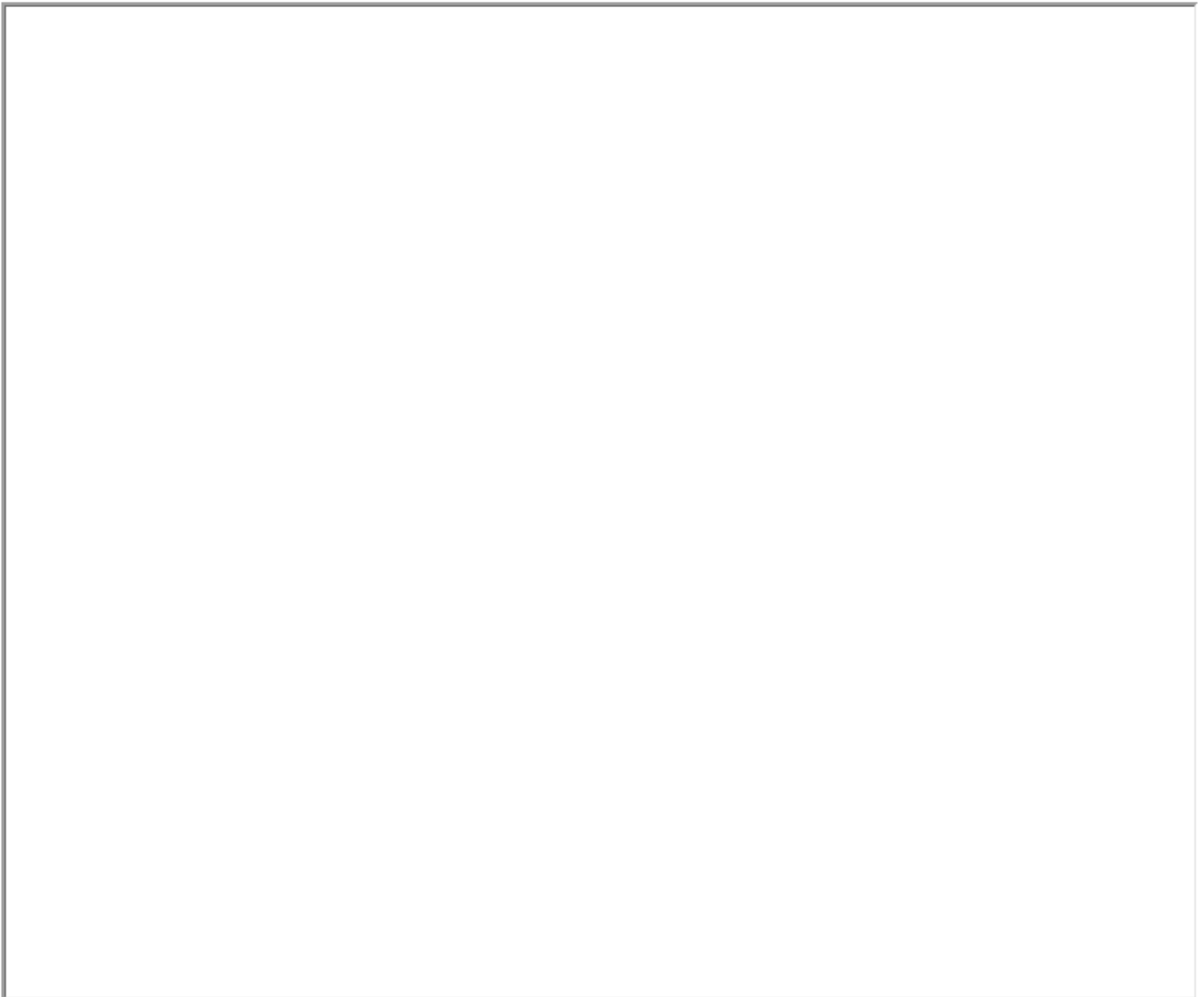
## ДОДАТКИ

### Додаток А – Інтерфейс програми

🔍 Детектор фейкових новин



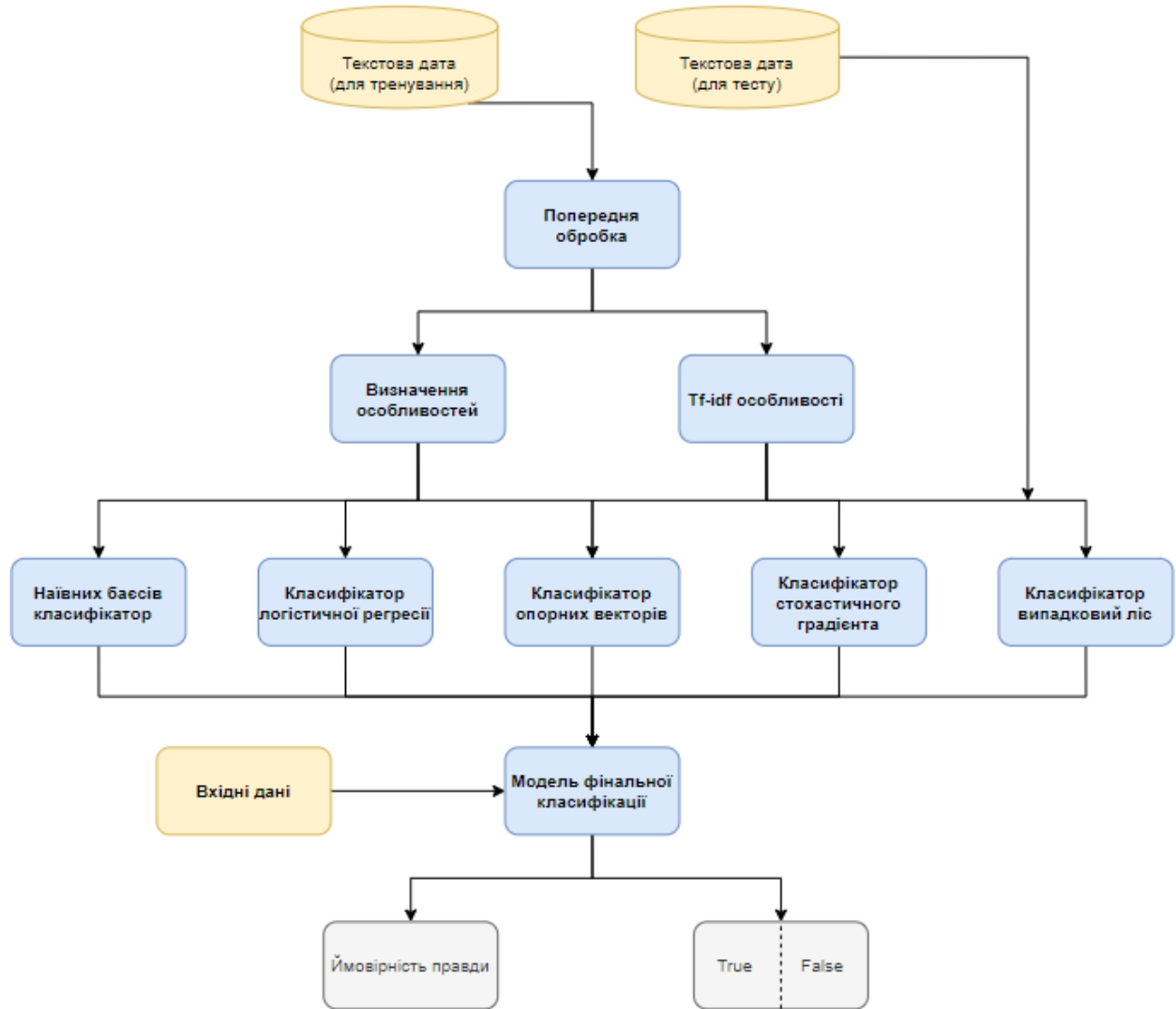
## ПЕРЕВІР НА ФЕЙКОВІСТЬ



**Фейкові новини:**  
**Коефіцієнт:**

**Результат**

## Додаток Б - Діаграма процесів



**Додаток В – Software Architecture Document**

**Software Architecture Document (SAD)**

*“Software methods of fake news detection on news websites and social networks”*

CONTENT OWNER: Ivan Karpets

18.05.2021

## Revision History

<b>Version</b>	<b>Description of Versions / Changes</b>	<b>Responsible Party</b>	<b>Date</b>
1.0	Initial version	Ivan Karpets	08/01/21
1.1	Add introduction section	Ivan Karpets	15/01/21
1.2	Add architectural representation	Ivan Karpets	23/01/21
1.3	Add architectural goals and constraints	Ivan Karpets	15/02/21
1.4	Add use-case view	Ivan Karpets	27/02/21
1.5	Add modules view	Ivan Karpets	13/03/21
1.6	Add process view	Ivan Karpets	17/04/21

# **Software Architecture Document**

## **1. Introduction**

This document gives an overview and explains the fake news detection software architecture.

In the document the architecture design goals, the use cases of the system, the architectural styles, and the selected components are described. The document provides a comprehensive overview of the fake news detection software system. It used as a communication medium between the software architect and other team members.

### **1.1. Purpose**

The Software Architecture Document (SAD) gives a extensive architectural overview of the Software methods of fake news detection on news websites and social networks. It presents a number of different architectural views to see the different aspects of the system.

### **1.2. Scope**

The scope of this document is to present the design of the system that was developed.

This document describes the different views of the software design that are considered to be architecturally considered. These elements and behaviors are fundamental for guiding the structure of the system and understanding the system.

### 1.3. Definitions, Acronyms, and Abbreviations

- **AI** – Artificial intelligence
- **ML** - Machine learning
- **MVC** – Model/View/Controller
- **UML** – Unified Modeling Language
- **User** - This is any user who uses the app

### 1.4. References

1. <https://slite.com/learn/how-to-write-project-proposal>
2. <https://smart-it.io/blog/writing-a-project-management-plan-for-software-development/>
3. <https://www.mitre.org/publications/systems-engineering-guide/se-lifecycle-building-blocks/concept-development/concept-of-operations>

### 1.5. Overview

Section 2: Describes the usage of each view

Section 3: Describes the design goals and restrictions

Section 4: Describes use-cases

Section 5: Modules view

Section 6: Process view

## 2. Architectural Representation

The views used to document the fake news detection software architecture.

are:

### Use Case view

**Audience**: actual end-users of the system.

**Area**: describes the users and use-cases for the system, this view presents the needs of the user and is elaborated further at the design level to describe discrete flows and constraints in more detail. Described the scenario, central functionality of the system

**Related Artifacts** : Use-Case Model, Use-Case documents

### Modules view

**Audience**: Developers, architects.

**Area**: describes the modules implemented of the system, their responsibilities, relations.

**Related Artifacts**: Modules Model

### Process view

**Audience**: Data specialists, analysts.

**Area**: describes the process flow of the developed system. Also shows how processes are dependent.

**Related Artifacts**: Process Model

### 3. Architectural Goals and Constraints

There are important requirements and constraints the architecture:

1. The system is meant as a proof of concept for a more complete project prediction system to be built in the future. Therefore one of the primary stakeholders in this document and the system as a whole are future architects and designers, not necessarily users as is normally the case. As a result, one goal of this document is to be useful to future architects and designers.
2. The software was developed using Python programming language. Python is a general-purpose high-level programming language. The language is focused on improving developer productivity and code readability. Python supports several programming paradigms: structured, object-oriented, functional, imperative, and aspect-oriented. The language has dynamic typing, automatic memory management, full introspection, an exception handling mechanism, support for multithreaded computations, and convenient high-level data structures. Python program code is organized into functions and classes, which can be combined into modules, which in turn can be bundled. Python is commonly used as interpreted but can be compiled to Java bytecode and MSIL (within the .NET framework).
3. The system should use some third-party libraries but not greater than 30%.
4. The primary goal of the system architecture is to the amount of code coupling. The system was designed such as each module should serve a single responsibility. This is performed using isolation of components that can be changed from the rest of the system.

## 4. Use-Case View

The main goal of the section is to give additional information surrounding the usage of the system and the communication between its components. Section 4.1 shows an actor and gives a description in the overall use context of the system. In section 4.2, the most common use-cases are showed using UML use-case and sequence diagrams to clarify the interactions between components.

### 4.1. Actors

#### User

The user will use all functionality of the developed software. User does not have any restrictions and can interact with the software using graphical user interface.

### 4.2. Use-Case Realizations

User inputs the data to compare.

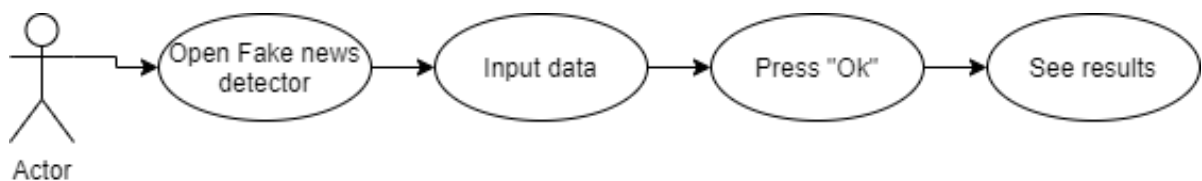


Figure D.4.1. Use-case diagram

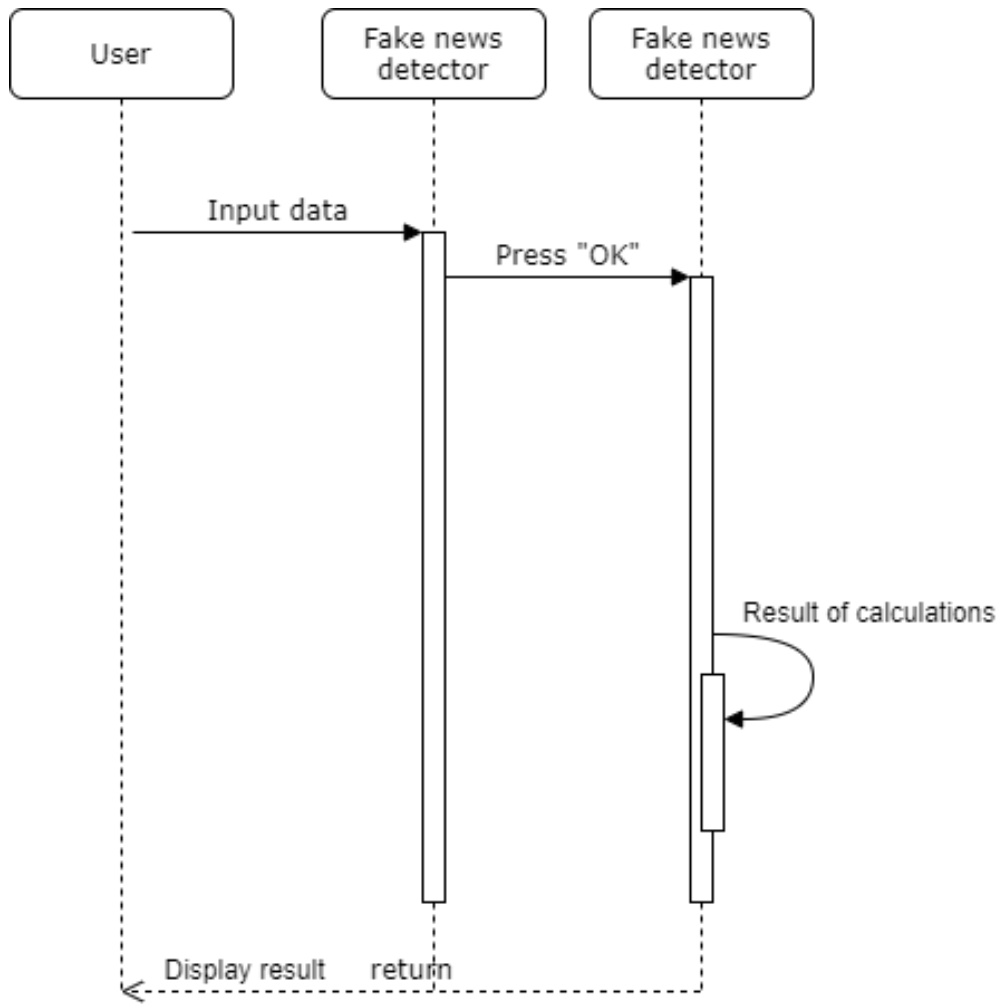


Figure D.4.2. Sequence Diagram

## 5. Modules View

In the section described the system modules and relations between them, interfaces used to communicate.

Module	Definition
DataPrep	The module has all the pre processing methods that process all input documents and texts. First of all we read the train, test and validation data files then do pre processing like tokenizing, stemming etc. There are some data analysis is performed like response variable distribution and data quality validations like on null or missing values etc.
FeatureSelection	The module does feature extraction and selection methods from sci-kit learn python libraries. Feature selection is performed using methods like simple bag-of-words and n-grams and then term frequency like tf-tdf weighting. Also used word2vec and POS tagging to extract the features.
Classifier	The module contains the classifiers for finding the fake news. The extracted features goes into different classifiers: Naive-bayes, Logistic Regression, Linear SVM, Stochastic gradient descent and Random forest classifiers from sklearn. Each feature was used in all of the classifiers. Once fitting the model, it compares the f1 score and checked the confusion matrix. After passing all the classifiers, the 2 models with the best performance were selected as candidate models for the classification of fake news. Then tuned the parameters by implementing the GridSearchCV methods for these candidate models and selected the most efficient parameters for these classifiers. As a result, the chosen model was used to detect fake news with the probability of being true. In addition to this, extracted the top 50 features from term frequency tf-idf vectorizer to see which words are the most and important in each of the classes. Precision-Recall and Learning Curves also used to see how the training and test suite performs as we increase the amount of data in the classifiers.
Prediction	It takes a news article as input from the user, after which the model is used to finalize the classification, which is displayed to the user along with the probability of being true.

## 6. Process View

This diagram illustrates the process flow of the system. Each element represents a single process. Relationship cardinality is denoted with UML multiplicity notation. This diagram illustrates how processes are dependent.

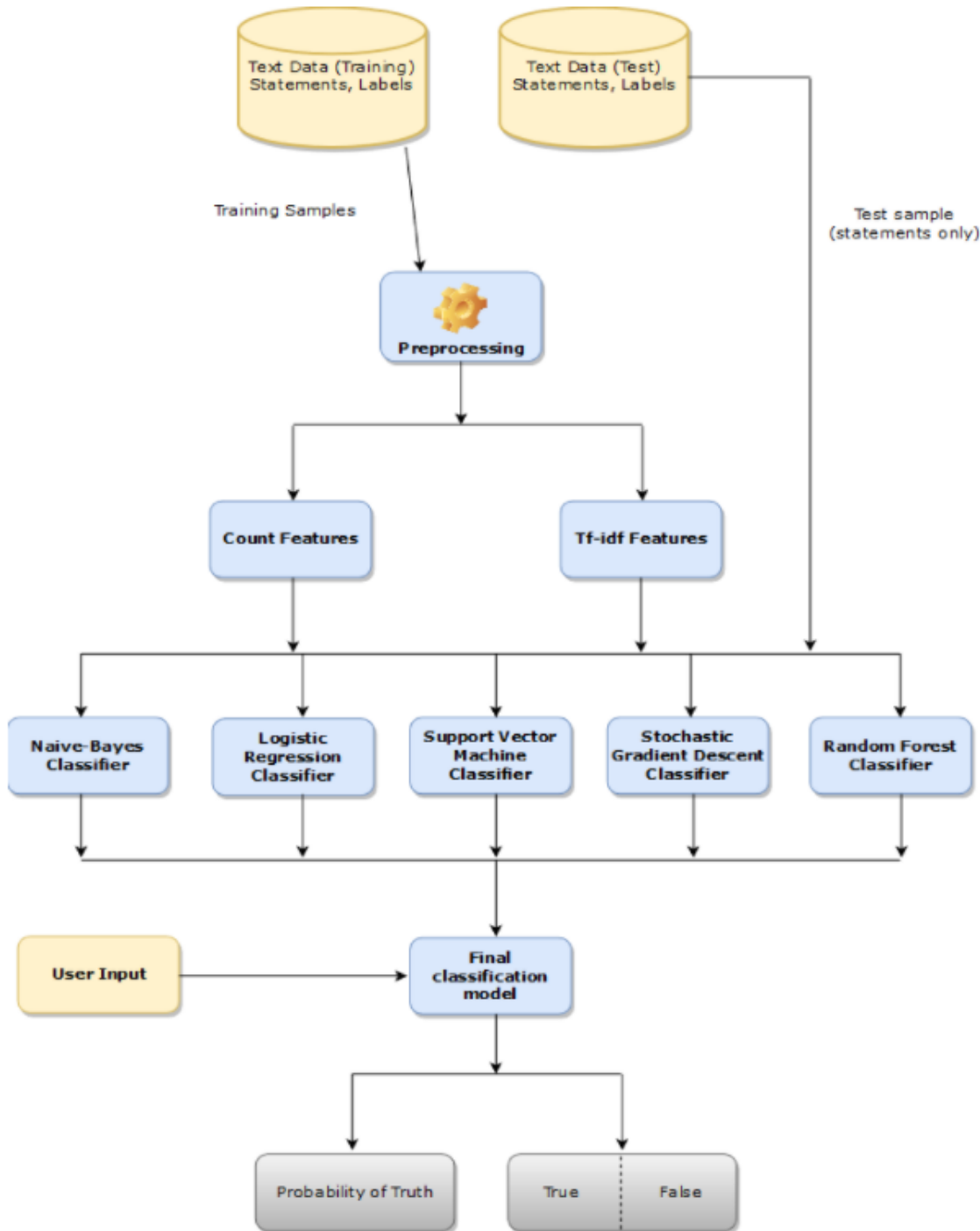


Figure D.8.2. Process Flow Diagram