ДОДАТОК А

Програмний код

```
public protocol TextClassifier { // Протокол текстового класифікатору
    func predictedLabel(for text: String) -> String?

    var name: String { get }
}

public protocol Dataset { // Протокол набору даних
    var items: [DatasetItem] { get }
    var labels: Set<String> { get }
    func items(for label: String) -> [DatasetItem]
}

public protocol Preprocessor { // Протокол препроцесору текста

    func preprocess(text: String) -> [String: Int]

    func preprocessedText(for text: String) -> String
}

public struct DatasetItem: Codable { // Єлемент, що повертається від Dataset
    public var id: Int
    public var text: String
    public var label: String
    public var predictedLabel: String?

    public init(id: Int, text: String, label: String, predictedLabel: String?
= nil) {
        self.id = id
        self.text = text
        self.label = label
        self.predictedLabel = predictedLabel
    }
}

public class TrivialPreprocessor: Preprocessor { // Простий обробник тексту

    public init() { }

    public func words(of text: String) -> [String] {
        let words = text
            .components(separatedBy: .whitespacesAndNewlines)
            .map { $0.components(separatedBy:
CharacterSet.punctuationCharacters).joined() }
            .filter { !$0.isEmpty }
```

```
        return words
    }

    public func preprocess(text: String) -> [String: Int] {
        let features = words(of: text).reduce(into: [String: Int]()) {
result, word in
            result[word, default: 0] += 1
        }
        return features
    }

    public func preprocessedText(for text: String) -> String {
        return words(of: text).joined(separator: " ")
    }
}

public class AdvancedPreprocessor: Preprocessor { // Продвинутий обробник
тексту

    public init() {}

    public func words(of text: String) -> [String] {
        var preprocessedText = text

        let types: NSTextCheckingResult.CheckingType = [.phoneNumber, .link,
.date]
        if let detector = try? NSDataDetector(types: types.rawValue) {
            preprocessedText = detector.stringByReplacingMatches(
                in: text,
                options: [],
                range: NSRange(location: 0, length: preprocessedText.count),
                withTemplate: " "
            )
        }

        if let numberSequenceRegexp = try? NSRegularExpression(pattern:
"\\d+") {
            preprocessedText = numberSequenceRegexp.stringByReplacingMatches(
                in: preprocessedText,
                options: [],
                range: NSRange(location: 0, length: preprocessedText.count),
                withTemplate: ""
            )
        }

        let words = preprocessedText
            .components(separatedBy: .whitespacesAndNewlines)
            .map { $0.components(separatedBy:
CharacterSet.punctuationCharacters).joined() }
            .filter { !$0.isEmpty }
        return words
    }
```

```
    public func preprocess(text: String) -> [String : Int] {
        let features = words(of: text).reduce(into: [String: Int]()) {
result, word in
            result[word, default: 0] += 1
        }
        return features
    }

    public func preprocessedText(for text: String) -> String {
        return words(of: text).joined(separator: " ")
    }
}
```

ДОДАТОК Б

Слайди презентації

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

АТЕСТАЦІЙНА РОБОТА МАГІСТРА

Дослідження методів розпізнання спаму у вхідних sms-повідомленнях для iOS

Виконав:
ст. гр. ІПЗм-18-1
Єрьоменко М. О.

Науковий керівник:
проф. д.т.н. Єрохін А.Л.

Мета роботи

- Дослідження методів класифікації SMS-спаму
- Розробка мобільного додатку для фільтрації SMS-спаму на iOS пристроях

Аналіз предметної галузі

Спам це

- Shoulder of Pork and HAM
- Масова розсилка
- Без дозволу користувача
- Небажана інформація
- Складає 15% від загальної кількості повідомлень

За способами розповсюдження

- Електрона пошта
- Соціальні мережі
- SMS

3

Аналіз предметної галузі | Набір даних

Існуючі набори

- Відсутність української
  - Іноземні: англійська, французька, китайська та інші
- Невідповідний контент
  - Моделям потрібні «реальні» дані

Власний набір

- 20 000+ повідомлень
  - 63% спаму
  - 37% не спаму
- Повідомлення, що отримують користувачі в Україні

4

Аналіз зібраних даних

Види sms спаму

- Реклама
- Фішинг - виманювання персональних даних
- «Нігерійські листи» - виманювання грошових коштів

Зміст SMS повідомлень в Україні

Інформаційні повідомлення	38%
Реклама	55%
Фішинг - виманювання персональних даних	3%
«Нігерійські листи» - виманювання різними способами грошових коштів	4%

5

Постановка задачі

- Проаналізувати моделі методів класифікації спаму для sms-повідомлень
- Сформувати набір sms-повідомлень українською мовою, що будуть використані при тренуванні машинних моделей
- Розробити машині моделі
- Розробити iOS додаток та інтегрувати машині моделі, що дозволить:
  - переглядати список sms-повідомлень
  - переглядати результати класифікації кожного метода
- Оцінити точність методів класифікації та порівняти результати

6

Аналіз методів | Наївний байєсів класифікатор

- В основі лежить теорема Байеса
- Припускає незалежність
  - Змісту елементів
  - Порядку елементів
- Мета: знайти найбільш ймовірний клас

$$P(c|d) = \frac{P(d|c)P(c)}{P(d)}$$

$P(c|d)$ – ймовірність того, що документ d потрапить в клас c

$P(d|c)$ – ймовірність зустріти документ d серед всіх документів класу c

$P(c)$ – ймовірність зустріти документ класу c в корпусі документів

$P(d)$ – ймовірність документа наявності d в корпусі документів

7

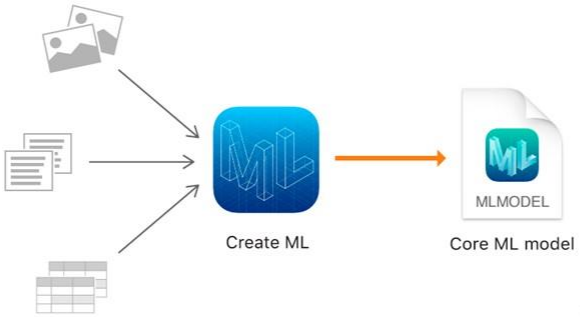
Аналіз методів |ЛСА - латентно-семантичний аналіз

- Навчання без вчителя
- Припускає
  - близькі за значенням слова будуть зустрічатися в аналогічних фрагментах тексту
- Дозволяє сформувати класи

8

Аналіз методів | CoreML моделі

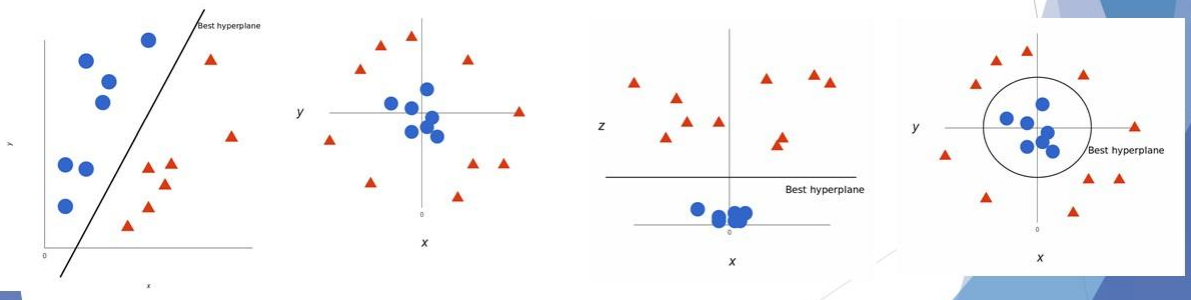
- метод максимальної ентропії (MaxEnt)
  - знаходить залежності між елементами в рамках одного повідомлення
- метод умовного випадкового поля (CRF)
  - знаходить залежності між елементами в рамках усього набору



10

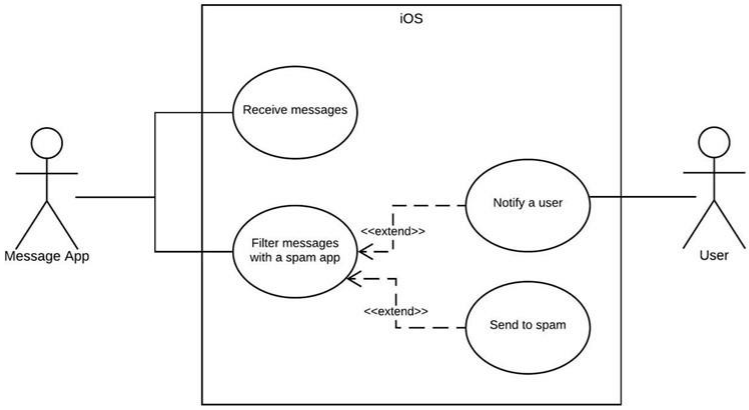
Аналіз методів | SVM – метод опорних векторів

- Основна мета:
  - побудувати гіперплощину у просторі високої або нескінченної вимірності



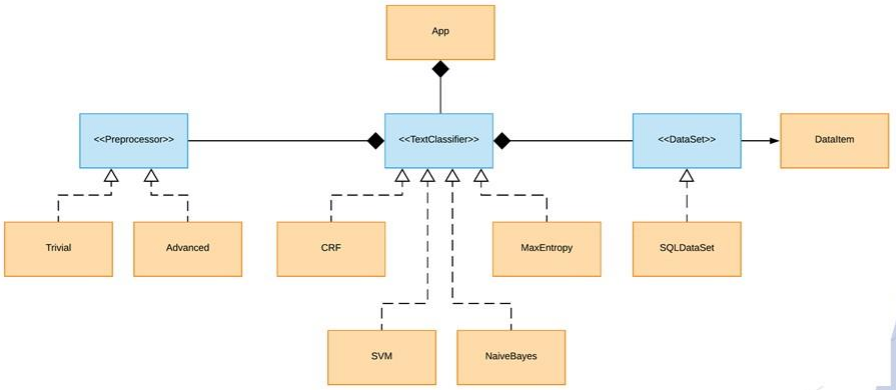
9

Діаграма прецедентів



11

Діаграма класів



12

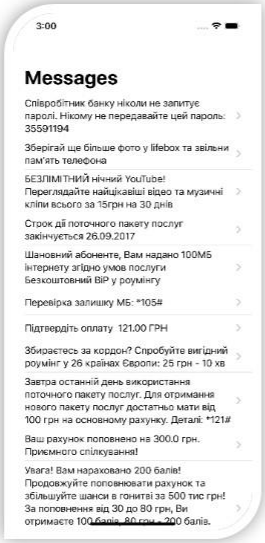
Стек технологій

- iOS 13
- Swift
- SwiftUI
- CocoaPods
- CoreML

- Python
- scikit-learn

13

Інтерфейс iOS додатку



14

Порівняння за точністю

Класифікатор	Точність
Наївний байєсівський класифікатор	89%
SVM	93%
CoreML + (MaxEntropy)	94%
CoreML + (CRF)	96%

15

Висновки

- Існує потреба у засобах фільтрації рекламних спам повідомлень
- Окрім рекламного контенту спам може призвести до збитків
- Сформована класифікація спаму на основні українських sms
- Досліджено 5 методів класифікації спаму
- Проведене порівняння розроблених класифікаторів за точністю розпізнання
- Розроблено додаток та інтегровані машинні моделі для фільтрації вхідних sms-повідомлень на платформі iOS

16

Пропозиції для подальшого дослідження

- Дослідити класифікатори з іншими мовами: англійська, російська та транслітерація
- Реалізувати функціонал накопичування спам повідомлень за певний період
- Окрім тексту, аналізувати номер телефону з якого надходить sms
- Додати автоматичну заявку про видалення номера абонента з розсилок

ДОДАТОК В

Апробація результатів роботи

АНАЛІЗ ІСНУЮЧИХ ПРОГРАМНИХ РІШЕНЬ ДЛЯ ВИЯВЛЕННЯ СПАМУ

Єрьоменко М. О.

Харківський національний університет радіоелектроніки

м. Харків

Більшість з існуючих програм для боротьби зі спамом фільтрують повідомлення, що приходять в поштову скриньку. Це зручно з двох причин. По-перше, за допомогою цих програм можна не перекачувати з сервера непотрібні листи. По-друге, вони дозволяють організувати сортування решти кореспонденції.

Антиспам додатки для мобільних пристроїв можна розділити на категорії декількома способами [1, 2]:

За реалізацією:

- антиспам функціонал, реалізований в комплексному антивірусному засобі (як правило, це додатки з багатим функціоналом);
- окремий додаток антиспам, з функціями блокування вхідних дзвінків / SMS;
- антиспам функціонал, реалізований в складі різних месенджерів.

За параметрами , що блокуються:

- за номером телефону;
- за номером телефону і ключовими словами;
- за номером телефону і хешу текстового повідомлення.

Блокування за номером телефону - напевно найпростіший спосіб блокувати спам, однак багато рішень не вміють блокувати спам, який прийшов ні з цифрового номера, а з текстового.

Блокування по тексту повідомлення - цей варіант більш кращий, тому що однотипний спам може надходити з різних номерів.

За наявністю і розташуванням глобального чорного списку (номерів, ключових слів) для аналізу контенту:

- бази немає (наповнюється користувачем / зберігається на телефоні / НЕ завантажується на сервер);
- база в телефоні;
- база в телефоні і на віддаленому сервері;

В якості власного персоналізованого механізму захисту вбудованих комунікаційних додатків на платформі iOS від спаму, буде реалізований модуль, який розробники комунікаційних додатків зможуть використовувати для фільтрації небажаних повідомлень.

Список літератури

1. Борьба со спамом: история и методы [Електроний ресурс] // URL: https://mipt.ru/dmcp/student/diff_articles/no_spam.php (дата звернення: 22.02.20)

2. СМС спам - как бороться с рекламными SMS и звонками // Адвокатское Бюро Шмелёва [Електроний ресурс] // URL: http://www.advocatshmelev.narod.ru/sms-spam.html — (дата звернення 15.02.2020).

експлуатації на уповільнених швидкостях. *Наукові Вісті Далівського Університету*, *18*. https://doi.org/10.33216/2222-3428-2020-18-9

[10] Лапкина, И. А., Малаксиано, Н. А., & Главатских, В. И. (2019). Многокритериальный подход к обоснованию выбора проекта приобретения и эксплуатации судна-балкера. *Збірник Наукових Праць ДУІТ. Серія "Транспортні Системи і Технології," 33*(2), 99–110. https://doi.org/10.32703/2617-9040-2019-33-2-10

DOI 10.36074/15.05.2020.v2.20

АНАЛІЗ ІСНУЮЧИХ ПРОГРАМНИХ РІШЕНЬ ДЛЯ ВИЯВЛЕННЯ СПАМУ

Єрьоменко Максим Олександрович
здобувач магістерського ступеня факультету комп'ютерних наук
Харківський національний університет радіоелектроніки

НАУКОВИЙ КЕРІВНИК:

Єрохін А.Л.
канд. тех. наук, декан факультету комп'ютерних наук
Харківський національний університет радіоелектроніки

УКРАЇНА

В якості власного персоналізованого механізму захисту вбудованих комунікаційних додатків на платформі iOS від спаму, буде реалізований модуль, який розробники комунікаційних додатків зможуть використовувати для фільтрації небажаних повідомлень.

На сьогоднішній день частка sms-спаму складає близько 15% від загальної кількості спам-повідомлень та все ще залишається одним з головних та дешевих каналів для масових розсилок. Окрім небажаного рекламного контенту спам може призвести до зараження пристрою, втрати персональних даних та матеріальних збитків.

Хоча деякі країни поступово роблять законодавчі обмеження для масових розсилок, загальна кількість повідомлень залишається на зазначеному рівні, що робить актуальним дослідження та розробку сучасних засобів фільтрації спаму.

Більшість з існуючих програм для боротьби зі спамом фільтрують повідомлення, що приходять в поштову скриньку. Це зручно з двох причин. По-перше, за допомогою цих програм можна не перекачувати з сервера непотрібні листи. По-друге, вони дозволяють організувати сортування решти кореспонденції.

Антиспам додатки для мобільних пристроїв можна розділити на категорії декількома способами [1, 2]:

За реалізацією:

– антиспам функціонал, реалізований в комплексному антивірусному засобі (як правило, це додатки з багатим функціоналом);

– окремий додаток антиспам, з функціями блокування вхідних дзвінків / SMS;

– антиспам функціонал, реалізований в складі різних месенджерів.

За параметрами , що блокуються:

– за номером телефону;

– за номером телефону і ключовими словами;

– за номером телефону і хешу текстового повідомлення.

Блокування за номером телефону - напевно найпростіший спосіб блокувати спам, однак багато рішень не вміють блокувати спам, який прийшов ні з цифрового номера, а з текстового.

Блокування по тексту повідомлення - цей варіант більш кращий, тому що однотипний спам може надходити з різних номерів.

За наявністю і розташуванням глобального чорного списку (номерів, ключових слів) для аналізу контенту:

– бази немає (наповнюється користувачем / зберігається на телефоні / НЕ завантажується на сервер);

– база в телефоні;

– база в телефоні і на віддаленому сервері;

Висновки.

В роботі проведено аналіз предметної області та сформована класифікація спаму. Досліджені методи класифікації як наївний байєсівський класифікатор, SVM, метод максимальної ентропії та метод умовного випадкового поля. Створена база sms-повідомлень українською мовою. Розроблені машинні моделі за допомогою Python та інструменту scikit-learn. Проведене порівняння розроблених класифікаторів за точністю розпізнання. Розроблено додаток та інтегровані машинні моделі для фільтрації вхідних sms-повідомлень на платформі iOS.

Усі методи показують досить високу точність розпізнання: від мінімальних 89% для байєсівського класифікатора до максимальних 96% за допомогою методу умовного випадкового поля. Метод SVM та наївний байєсівський класифікатор доцільно використовувати, коли обсяг тренувального набору даних невеликий (до 500 повідомлень). Натомість, методи CRF та MaxEntropy дозволяють отримати точність розпізнання вище, але потребують більшого обсягу тренувальних даних (від 20000 повідомлень).

Список використаних джерел:

[1] Клименко, А. П. Борьба со спамом: история и методы. МФТИ. Вилучено з https://mipt.ru/dmcp/student/diff_articles/no_spam.php (дата звернення: 22.02.2020).

[2] СМС спам – как бороться с рекламными SMS и звонками. Адвокатское Бюро Шмелёва. Вилучено з http://www.advocatshmelev.narod.ru/sms-spam.html (дата звернення: 15.02.2020).