

Для виконання задачі атрибуції тексту і експертні, і формальні методи намагаються зробити одне і те саме: визначити авторський стиль та порівняти його з вже відомим. Під “авторським стилем” зазвичай розуміють останні три рівні дослідження тексту [3].

Розроблювальна система повинна поєднувати різні формальні методи атрибуції тексту, та повинна бути протестована на текстах різних розмірів, жанрів та стилів. Таке тестування дасть можливість бути впевненим в валідності системи для її подальшого використання або імплементації в вже існуючі.

Одним з основних методів, що будуть використані в розроблювальній системі є так звані інформаційні портрети тексту [2]. Ідея інформаційних портретів полягає у тому, що “авторський стиль” на високому рівні представлений поєднанням слів. Схильність автора використовувати ті чи інші комбінації слів найкраще відображає його стиль і як результат саме комбінації слів слід розглядати для визначення авторства тексту. Проте комбінацій слів може бути велика кількість, один текст може мати від 1000 до 10000 унікальних слів, для визначення частоти використання пар слів необхідно будет побудувати матрицю 10000 на 10000 та проводити її повний аналіз, що є довгим та місцезаратним процесом, натомість інформаційний портрет є матрицею взаємної інформації між комбінаціями літер [4]. Такий підхід скорочує набір даних, що пришвидшує їх накопичення та порівняння. Слід також зазначити, що комбінаціями літер можуть слугувати не тільки сусідні пари літер, а і їх тріади, або пари літер через одну. Найкраще себе може показати система, що проводить побудову і порівняння всіх трьох типів портретів.

Проте може виникнути питання достовірності та доцільності такого підходу. Для тестування цього методу було відібрано 8 авторів XIX-XX століть, для кожного з них було обрано одинадцять творів, десять з яких використовувалися для побудови загального портрету автора, а останній для порівняння з усіма портретами авторів. Як результат з масиву портретів авторів, той, що мав найбільший коефіцієнт кореляції та найменше середнє квадратичне відхилення між матрицею взаємної інформації, обирався як портрет, що належить оригінальному автору тексту. Таке тестування показало, що цей метод дає точність 88%.

Список використаних джерел

1. Lomakina L.S., Rodionov V.B., Surkova A.S. Hierarchical Clustering of Text Documents // Automation and Remote Control. 2014. V. 72. № 9. P. 345–351.
2. Ломакина Л.С., Мордвинов А.В., Суркова А.С. Построение и исследование модели текста для его классификации по предметным категориям // Системы управления и информационные технологии. 2011. № 1(43). С. 16–20.
3. Surkova A.S., Domnin A.A., Bulatov I.V., Tsarev A.A. Neural networks and decision trees algorithms – the base of automated text classification and clustering // Am. J. Control Systems and Information Technology. Science Book Publishing House, LLC. 2013. № 2. P. 33–35. 137
4. Smith R.E., Jiang M.K. MILCS: a mutual information learning classifier system // Proceedings of the Genetic and Evolutionary Computation Conference (GECCO '07). 2007. P. 2945–2952.

УДК 330.88(045)

СТВОРЕННЯ СИСТЕМИ КОМУНІКАЦІЙ МІЖ АДМІНІСТРАЦІЄЮ УНІВЕРСИТЕТУ ТА ЗДОБУВАЧАМИ ВИЩОЇ ОСВІТИ

Іскрижицький А. М., студ. гр. ПІ-161,

Іскрижицька О. К., студ. гр. ПІ-161

Науковий керівник: Трунова О.В., к.пед.н., доцент

Національний університет «Чернігівська політехніка»

Зміни у системі освіти, зміна поколінь, пріоритетів молодого покоління призводить до зміни взаємодії між адміністрацією університету та здобувачами вищої освіти (ЗВО), традиційна схема спілкування між студентами та адміністрацією стає менш ефективною. Люди звикають до взаємодії через месенджери, мобільні додатки, уникаючи при цьому

великих, застарілих та перевантажених інформацією сайтів. Метою даної роботи є покращення взаємодії між адміністрацією та ЗВО. Завданнями роботи є полегшення комунікації, доступу до інформації, зробити зручним процес отримання необхідної інформації.

Вдалим рішенням даної проблеми з точки зору ЗВО є мобільний додаток або псевдо мобільний додаток, як наприклад Telegram, оскільки даний месенджер постійно розвивається, є безкоштовним, доступним з будь-якого користувацького пристрою, безпечним та має зручний функціонал.

Засоби взаємодії людей з інформаційним ресурсом через системи Telegram та його API [1] дозволяють:

- підписатися на повідомлення від адміністраторів Telegram-боту;
- отримати необхідну інформацію;
- використовувати стандартний UX дизайн платформи Telegram;
- мати доступ до боту з будь-якого користувацького пристрою (мобільний додаток на платформі Android, IOS або ПК з операційною системою Windows, Mac OS, Ubuntu та інші шляхом активних додатків або веб- додатків.

З ініціативи директора ННІ електронних та інформаційних технологій Національного університету «Чернігівська політехніка» Іванця С.А. було прийнято рішення створити систему, яка надавала б ЗВО та адміністрації університету такі можливості для покращення взаємодії.

Система забезпечує для здобувача вищої освіти такі можливості (див. рис. 1):

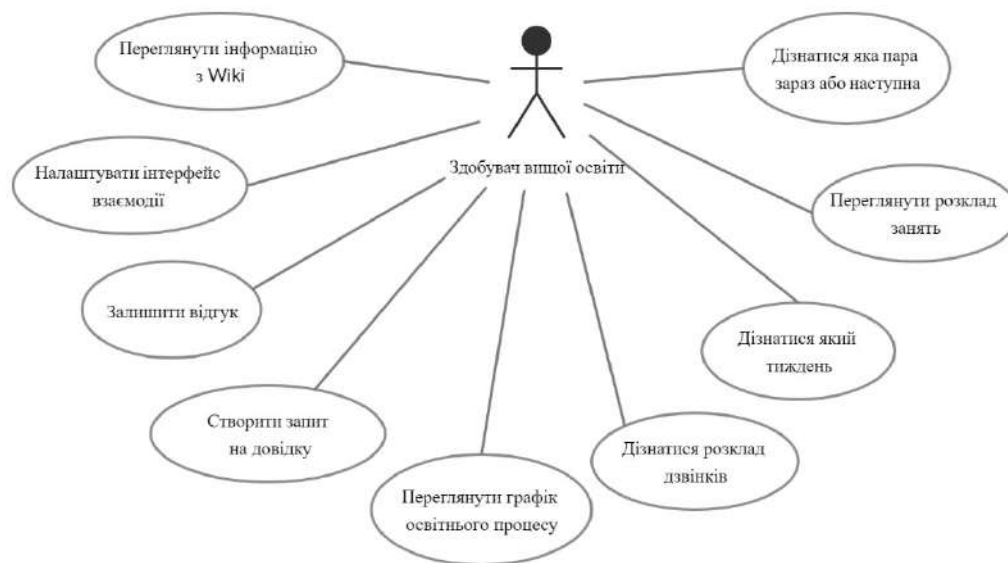


Рисунок 1 – Use Case діаграма для актора здобувач вищої освіти

Для того щоб надавати таку інформацію, переглядати відгуки та мати можливість сповіщати ЗВО, підписаних на Telegram-бот, було прийнято рішення розробити веб-додаток для адміністрування.

Проаналізувавши існуючі технології, були вибрані: мова програмування TypeScript [2], платформу NodeJS [3], база даних MongoDB [4], фреймворк для веб-додатку ReactJS [5]. Оскільки вони дають можливість реалізувати описаний функціонал в короткий термін, завдяки використанню великої кількості готових рішень та менеджера пакетів npm.

Найбільш зручною методологією розробки для команди з двох людей є Kanban першого рівня.

Мікро-сервісний підхід є найбільш зручним для розробки блоків системи. При використанні мікро-сервісного підходу було прийнято рішення розділити серверну частину системи на дві складові:

- API для взаємодії з веб додатком, яке має надавати набір функцій, що дозволяють читати, записувати в базу даних або відправляти повідомлення до Telegram-боту та бути надійно захищене від зловмисників;

- npm пакет node-telegram бот API для взаємодії з Telegram, що дозволяє встановити websocket з'єднання, слухати повідомлення, що були відправлені Telegram-боту, а також сповіщення користувачам, що підписані на Telegram-бот. Функціонал даної бібліотеки не структурує користувацький код, тому поверх даної бібліотеки була написана обгортка, що надає можливість використовувати патерн middleware.

Третім компонентом системи є веб додаток, що використовує такі технології: мову програмування TypeScript, фреймворк ReactJS, Material UI [6] – бібліотека компонентів React заснована на Google Material Design [7], Redux [8] для управління станом додатку.

Telegram-бот доступний для ЗВО з вересня 2019 року. Проте задовольнити всі потреби користувача такою малою командою досить важко. Також з плином часу будуть з'являтися нові кейси, що потребуватимуть більше функціонала системи комунікації. Тому дана система буде у вільному доступі і кожен бажаючий зможе покращувати її. Такий підхід дозволить не тільки покращити взаємодію між Telegram ЗВО та адміністрацією університету, але й усіма бажаючим брати участь у розробці та впровадженні своїх ідей в працюючий проект. Можливе додавання таких функцій, як перегляд розкладу заліків та екзаменів, можливість дізнатися який зараз екзамен або який наступний, сповіщення про події, заходи та конференції.

Список використаних джерел

- 1.Telegram APIs. [Електронний ресурс]. – Режим доступу: <https://core.telegram.org/> (дата звернення: 12.03.20). – Назва з екрана.
- 2.TypeScript language. [Електронний ресурс]. – Режим доступу: <https://www.typescriptlang.org/> (дата звернення: 12.03.20). – Назва з екрана.
- 3.Node JS. [Електронний ресурс]. – Режим доступу: <https://nodejs.org/uk/>(дата звернення: 12.03.20). – Назва з екрана.
- 4.MongoDB. [Електронний ресурс]. – Режим доступу: <https://www.mongodb.com/>(дата звернення: 12.03.20). – Назва з екрана.
- 5.ReactJS. [Електронний ресурс]. – Режим доступу: <https://reactjs.org/>(дата звернення: 12.03.20). – Назва з екрана.
- 6.Material UI. [Електронний ресурс]. – Режим доступу: <https://material-ui.com/>(дата звернення: 12.03.20). – Назва з екрана.
- 7.Google Material Design. [Електронний ресурс]. – Режим доступу: <https://material.io/design/>(дата звернення: 12.03.20). – Назва з екрана.
- 8.Redux. [Електронний ресурс]. – Режим доступу: <https://redux.js.org/>(дата звернення: 12.03.20). – Назва з екрана.

УДК 004.93'1

УДОСКОНАЛЕНИЙ АЛГОРИТМ ПОШУКУ ОБЛИЧЧЯ ТА ОЧЕЙ НА БАЗІ КАСКАДНОГО КЛАСИФІКАТОРА ХААРА

Бугай Ю.В., студ. гр. ПІ-161; **Войтенко В.П.**, к.т.н., доцент
Національний університет «Чернігівська політехніка»

Розпізнавання облич – це один із найважливіших напрямів розпізнавання образів, яке, зазвичай, має за мету віднесення вихідних даних до певного класу за допомогою виділення істотних ознак, що характеризують ці дані, із загальної маси даних [1]. Перспективи розвитку методів розпізнавання облич досить значні [2, 3]. Так, сьогодні їхнє використання можна знайти у різних сферах – від безпеки до комерції.

Наразі відомі алгоритми розпізнавання облич мають ряд недоліків, а саме: