

программного продукту. Практическая часть разработки – рабочий стол – приложение, разработанный для платформы Windows.

*Ключевые слова:* Python, десктоп-приложение, программный продукт, Матрица Эйзенхауэра.

*Abstract.* The article is devoted to the development of a software product to facilitate planning your own affairs in terms of importance and urgency. The paper describes the advantages and disadvantages of this software product. The practical part of development is the desktop – an application developed for the Windows platform.

*Key words:* Python, desktop application, software product, Eisenhower Matrix.

## СПИСОК ЛІТЕРАТУРИ

1. Swaroop C.H. A byte of Python
2. Функціональність PyCharm. URL: <https://www.jetbrains.com/ru-ru/pycharm/features/>
3. Що таке Tkinter. 2018. URL: <https://younglinux.info/tkinter/tkinter.php>
4. Навчання Python Gui. 2019. URL: <https://pythonru.com/uroki/obuchenie-python-gui-uroki-po-tkinter>
5. Уроки для вивчення Tkinter. 2019. URL: <https://python-scripts.com/tkinter-introduction>
6. Введення в Tkinter. 2011. URL: <https://habr.com/ru/post/133337/>
7. Реалізація MVC паттерна. 2018. URL: <https://habr.com/ru/post/150267/>
8. Що таке MVC: початкові концепції. 2019. URL: [https://skillbox.ru/media/code/chto\\_takoe\\_mvc\\_bazovye\\_kontseptsii\\_i\\_primer\\_prilozheniya/](https://skillbox.ru/media/code/chto_takoe_mvc_bazovye_kontseptsii_i_primer_prilozheniya/)
9. Як користуватись Microsoft Access. 2013. URL: <https://ru.wikihow.com/%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D1%82%D1%8C%D1%81%D1%8F-Microsoft-Access>
10. Microsoft Access. 2010. URL:
11. URL: <https://sites.google.com/site/karnasiy1course/home/microsoft-access>

УДК 621.391.63/316.5

## РОЗРОБКА СЕРВІСУ РОЗПІЗНАННЯ МОВЛЕННЯ VAMBUSA

*І. О. Ставицький, А. Р. Нескородєва*

*Анотація.* Співробітниками міжкафедральної лабораторії з машинного навчання і інтелектуального аналізу даних був розроблений тестовий варіант сервісу Vambusa. Згаданий сервіс дозволить користувачу без втрати якості сприйняття обробляти значні обсяги аудіо та відео інформації. Сервіс Vambusa призначений для транскрибування мовлення у текст та генерації стислого переказу отриманого тексту (summary). Описані особливості архітектури мікросервісів застосунку, проведено порівняння із існуючими аналогами.

*Ключові слова:* обробка природної мови, розпізнавання мовлення, мікросервіс.

Обробка природної мови (NLP – Natural-language processing) є одним з найскладніших і разом з тим найперспективніших напрямів сучасного розвитку задач машинного навчання та аналізу даних (популярніше кажучи – задач штучного інтелекту, AI – Artificial intelligence). Згідно із сучасними визначеннями [1], обробку природної мови можна визначити як програмні та комп'ютерні додатки, створені для застосування обчислювальних методів для аналізу та синтезу природної мови та мовлення. Головна мета таких додатків полягає у реалізації обробки мови, подібно тому, як це робить людина, для аналізу в широкому сенсі текстів, сформованих за допомогою комп'ютерної техніки.

На сучасному етапі розвитку технологій NLP, серед головних завдань обробки природної мови можна виділити:

- *видобування даних* (data mining);
- *генерування природної мови* (NLG – natural-language generation);
- *інформаційний пошук* (information retrieval);
- *машинний переклад* (machine translation);
- *розпізнавання мовлення* (speech recognition), або як технічний термін *мовлення-у-текст* (STT – speech to text);
- *синтез мовлення*;
- тощо.

Розпізнавання мови – це міждисциплінарний синтез інформатики та комп’ютерної лінгвістики; метою розпізнавання мови є розробка методології та відповідних технологій, що дозволяють розпізнавати та транслювати природню (розмовну) мову в текст засобами обчислювальної техніки.

Представлена робота присвячена демонстрації результатів роботи авторів з розробки сервісу для транскрибування мовлення у текст та генерує стислий переказ отриманого тексту, або summary. Робота виконана в рамках дослідницької програми міжкафедральної лабораторії з машинного навчання і інтелектуального аналізу даних факультету інформаційних і прикладних технологій Донецького національного університету імені Василя Стуса.

Згідно із аналізом статистики [2], лише на відеохостингу Youtube кожен хвилину викладається 500 годин відео. Зрозуміло, щоб в живу опрацювати лише цей, викладений за одну хвилину, відеоматеріал, людині необхідно 21 добу неперервно дивитись завантажений відеоконтент. Тому користувачі часто використовують функцію пришвидшення або в ручну промотують великі фрагменти відео. Таким чином, користувачі очевидно жертвують якістю задля суттєвого збільшення кількості отриманої інформації. Вочевидь, такий спосіб обробки відеоматеріалів є прийнятно ефективним далеко не для будь-якого типу контенту. Актуальною є задача розробки онлайн сервісу, що, по-перше, транскрибував би мовлення у відеофайлі у текст, та по-друге (і це значно складніша операція), міг би бути спроможним сконструювати з транскрипції тексту його стислий переказ (summary). З цією метою авторами статті була розроблена тестова версія сервісу Bambusa. Нижче на рисунку 1 представлений тестовий інтерфейс сервісу.

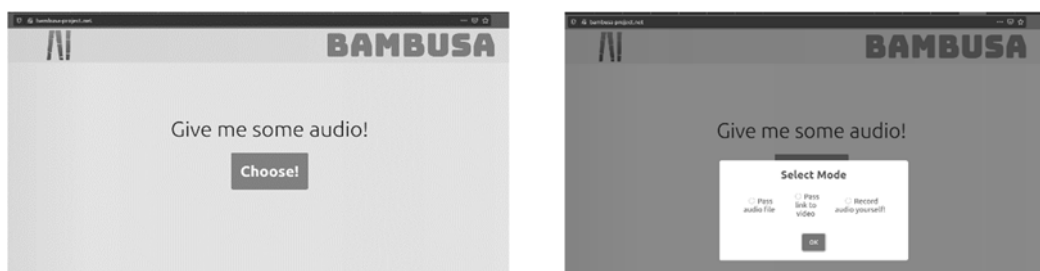


Рис 1. Інтерфейс сервісу Bambusa

В перспективі Bambusa дозволить користувачу без втрати якості обробити значно більший об’єм інформації. Час обробки залежить від довжини завантаженого файлу у відео/аудіо форматі.

Зважаючи на комплексність виконуваної задачі і необхідності поєднати одразу декілька підпрограм, а саме розпізнавання мовлення, корекцію слів та орфографії, та переказ тексту, було вирішено обрати архітектуру мікросервісів (рисунк 2).

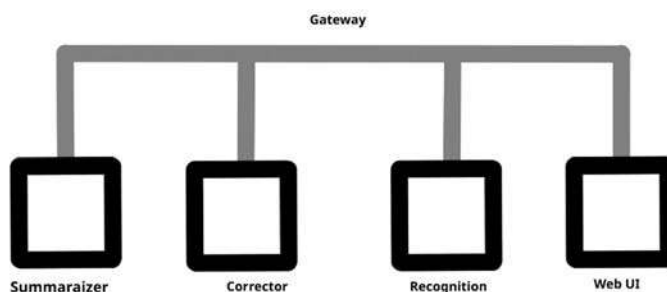


Рис 2. Архітектура мікросервісів

В загальному викладі, то архітектурний стиль мікросервісів – це підхід, коли застосунок конструється як набір невеликих сервісів, кожен з яких працює у власному процесі і спілкується з іншими сервісами, використовуючи прості механізми, як правило HTTP. Ці

сервіси побудовані навколо відповідних бізнес-потреб і розгортаються незалежно з використанням повністю автоматизованого середовища. Як стандарт, існує абсолютний мінімум централізованого управління цими сервісами. При цьому, самі по собі ці сервіси можуть бути написані на різних мовах і використовувати різні технології зберігання даних [3].

В мету забезпечення незалежності використовується технологія Docker для виключення конфліктів бібліотек, а також задля можливості змінювати модулі не ризикуючи зламати весь проект.

Розроблені мікросервіси:

- Recognizer – мікросервіс, головним завданням якого є транскрибування. Приймаючи на вхід аудіо запис чи аудіо у реальному часі він перетворює аудіо у текст та передає у Corrector (оскільки модель може видавати досить “сирий” текст, що потребує корекції). Для роботи сервісу використовується модель Silero. Модель підтримує 3 мови, а саме: англійську, німецьку, іспанську.

- Corrector – ключовий та самий критичний сервіс. Метою його роботи є корекція орфографії та пунктуації. Сам сервіс складається із 2 моделей: Punctuator для перевірки пунктуації та SpellChecker для перевірки орфографії. Критичність сервісу полягає у тому, що саме від якості виправленого тексту залежить якість подальшого переказу тексту.

- Summarizer – задача сервісу полягає у тому, щоб сформулювати для користувача текст, що складатиметься із заданої кількості речень. У роботі сервісу задіяна модель Spacy.

- Web – сервіс являє собою чистий UI – інтерфейс для взаємодії із користувачем. Інтерфейс дає змогу завантажити файл чи почати записувати аудіо безпосередньо на сторінці.

Для того, щоб сервіси могли взаємодіяти один з одним, автори використали gateway, що працює наступним чином: POST-запитом відсилається аудіо-файл/ посилання на відео на gateway, після попадання готовий файл відсилається до Recognizer (якщо потрібно завантажити відео – завантажується і теж відправляється). З gateway отримується текст та відсилається на Recognizer, з нього Corrector, а з нього на Summarizer для створення короткого змісту, після чого готовий текст надсилається користувачу.

Під час розробки сервісу, автори стикнулися із наступними проблемами.

У Recognizer виникли проблеми з перевіркою на тип файлу, що його завантажує користувач. Vambusa підтримує на даний момент 12 типів аудіо/відео файлів, а саме: aac, ac3, aiff, flac, m4a, mp3, mp4, ogg, opus, ts, wma, wav. Також зараз сервіс має обмеження на розмір файлу в 100Мб. Для роботи моделі розпізнавання необхідно мати файл типу wav. Тому спочатку необхідна перевірка на розширення файлу перед його конвертацією саме у Recognizer. Перевірка на тип файлу користувача відбувається у вебчастині, і якщо формат не підтримується, файл не відправляється у Recognizer.

У Corrector найбільш слабе місце – це модель, яка розставляє знаки пунктуації. Після обробки тексту, що складається лише з літер у нижньому регістрі, модель видає текст зі знаками пунктуації, але розставляю їх відносно рідко. Тобто текст складається з досить великих за обсягом речень. Це не заважає роботі моделі, яка виправляє помилки. SpellChecker перевіряє кожне слово окремо, незалежно від контексту. Текст з великих речень, що їх в подальшому отримає Summarizer, заводить моделі зробити якісний переказ тексту.

Модель у Summarizer шукає найбільш змістовне та популярне речення у тексті, а точніше необхідну кількість речень. Цей змінний параметр був встановлений на рівні 3. Після обробки тексту у Corrector моделлю Punctuator отримаємо 3–4 речення. В такому випадку після обробки тексту моделлю Spacy буде отримано 75 % або 100 % початкового тексту. Тобто наразі Vambusa погано працює для переказу малого за обсягом тексту.

Розроблений сервіс був порівняний з існуючими аналогами, наприклад, з сервісом транскрипції Otter.ai. Він існує вже 5 років та має доволі потужний функціонал: розпізнавання спікерів (працює, на думку авторів, досить сумнівно), створення хмари найпопулярніших слів, редагування та спільний доступ до транскрипції, інтеграція з Zoom та Google Meet Recordings,

розширення для Google Chrome, можливість робити нотатки під час зібрання у Zoom. Сервіс є комерційним, має декілька тарифів з помісячною оплатою, але користувач все одно має обмеження по хвилини для транскрипції. Є сервіси, що вимагають плату погодинно.

В якості прикладу за допомогою сервісу Bambusa був опрацьований відеоролик з мережі: «How much sleep do you really need? | Sleeping with Science, a TED series» [4]. Нижче наведене отриманий стисла анотація представленої в ролику текстової інформації (рисунок 3).

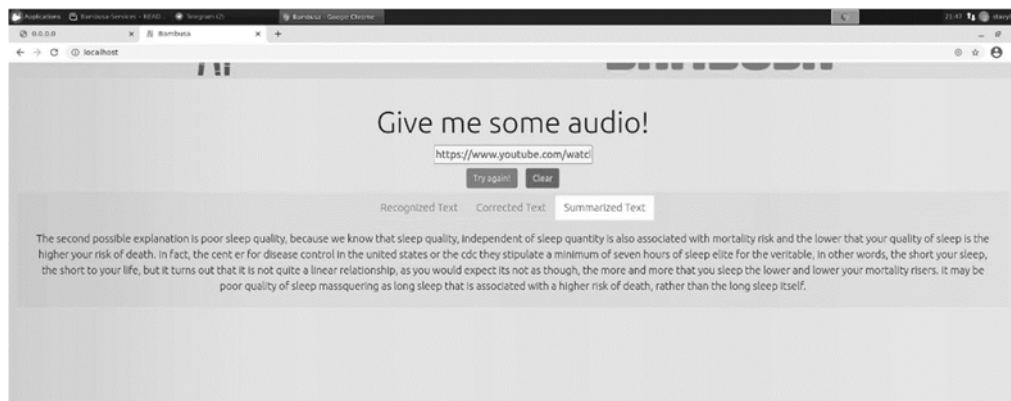


Рис. 3. Результати роботи сервісу Bambusa

Власне сама анотація: «The second possible explanation is poor sleep quality, because we know that sleep quality, independent of sleep quantity is also associated with mortality risk and the lower that your quality of sleep is the higher your risk of death. In fact, the center for disease control in the united states or the cdc they stipulate a minimum of seven hours of sleep elite for the veritable, in other words, the short your sleep, the short to your life, but it turns out that it is not quite a linear relationship, as you would expect its not as though, the more and more that you sleep the lower and lower your mortality risers. It may be poor quality of sleep massquering as long sleep that is associated with a higher risk of death, rather than the long sleep itself». Результат переказу основної текстової інформації у ролику можна вважати задовільним.

В подальшому до функціоналу сервісу Bambusa планується додати розпізнавання мовлення у реальному часі, тайм-коди у транскрипції, керування розміром переказу, редагування тексту транскрипції та переказу, інтеграція з іншими популярними відеосервісами, такими як Zoom, Microsoft Teams тощо.

*Аннотация.* Сотрудниками межкафедральной лаборатории машинного обучения и интеллектуального анализа данных был разработан тестовый вариант сервиса Bambusa. Указанный сервис позволяет пользователю обрабатывать большие объемы аудио и видео информации без потери качества восприятия. Сервис Bambusa предназначен для транскрибирования речи в текст и генерации краткого пересказа полученного текста. Описаны особенности архитектуры микросервисов приложения, приведены характеристики схожих коммерческих сервисов.

*Ключевые слова:* обработка естественного языка, распознавание речи, микросервис.

*Abstract.* A test version of the Bambusa service was developed by the staff of the interdepartmental laboratory for machine learning and data mining. This service allows the user to process large volumes of audio and video information without losing the quality of perception. The Bambusa service is designed to transcribe speech into text and generate a short retelling of the received text. The features of the microservices architecture of the application are described, the characteristics of similar commercial services are given.

*Keywords:* natural-language processing, speech recognition, microservice.

## СПИСОК ЛІТЕРАТУРИ

1. Zehra T., Umut A. Natural language processing applications in library and information science / Online Information Review. 2019. DOI: 10.1108/oir-07-2018-0217
2. URL: <https://rezart.agency/blog/youtube-stats-2020/> (дата звертання 30.03.2021).
3. Zmerzlyi I. Мікросервісна архітектура. URL: <https://medium.com/@IvanZmerzlyi/microservices-architecture-461687045b3d> (дата звертання 30.03.2021).
4. URL: <https://www.youtube.com/watch?v=knJWF4km3y0> (дата звертання 30.03.2021).