Міністерство освіти і науки України
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «КИЄВО-МОГИЛЯНСЬКА АКАДЕМІЯ»
Кафедра мультимедійних систем факультету інформатики

NLP: Automating the Creation of News Digests

Текстова частина до курсової роботи
за спеціальністю «Інженерія програмного забезпечення» 121

Керівник курсової роботи
старший викладач Смиш О.Р.
_____
(підпис)
«____» _____ 2024 р.

Виконав студент
Орлов С.О.
«____» _____ 2024 р.

Київ 2024

Міністерство освіти і науки України

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «КИЄВО-МОГИЛЯНСЬКА АКАДЕМІЯ»

Кафедра мультимедійних систем факультету інформатики

ЗАТВЕРДЖУЮ

Зав.кафедри мультимедійних систем,

проф., д.ф.-м.н.

_____О. П. Жежерун

(підпис)

«____»_____2023 р.

ІНДИВІДУАЛЬНЕ ЗАВДАННЯ

на курсову роботу

студенту Орлову С.О. факультету інформатики 3 курсу

ТЕМА: **NLP: Automating the Creation of News Digests**

Зміст ТЧ до курсової роботи:

Індивідуальне завдання

Introduction

Analysis of available solutions

Analysis of Telegram channels with the news digests

Analysis of models for summarization

Development of the bot

Conclusions

References

Appendixes

Дата видачі «___» _____ 2023 р.

Керівник _____

(підпис)

Завдання отримав _____

(підпис)

ТЕМА: **NLP: Automating the Creation of News Digests**

Календарний план виконання роботи:

№ п/п	**Назва етапу курсової роботи**	**Термін виконання етапу**	**Примітка**
1	Аналіз та узгодження теми роботи	15.07.2023 — 01.10.2023	
2	Огляд технічної документації за темою роботи	01.10.2023 — 06.05.2024	
3	Аналіз проблематики та наявних рішень	01.11.2023 — 20.04.2024	
4	Розробка і тестування	12.11.2023 — 16.05.2024	
5	Захист роботи на конференції	17.05.2024	
6	Захист роботи	23.05.2024 — 24.05.04	

Студент Орлов С.О.

Керівник Смиш О.Р.

«_____» 2023 р.

ЗМІСТ

АНОТАЦІЯ

Ця робота демонструє етапи розробки системи для автоматизації створення новинних дайджестів у вигляді чатбота для Telegram. Описано аналіз новинних дайджестів, створюваних українськомовними Telegram-каналів, оцінено мовні моделі на точність сумаризації, описано процес розробки Telegram-чатбота та обґрунтовано застосування програмних засобів та технологій у роботі.

Ключові слова: natural language processing, chatbot, summarization, language models, Telegram, mass media.

INTRODUCTION

In a day, an average person consumes around 74 GB of data, according to an article by Sabine Heim and Andreas Keil [1]. That amount is equivalent to the one a bright-minded individual would have faced in their lifetime 500 years ago. That is one example of a shift in lifestyle caused by free access to information the Internet provides humanity with.

The world nowadays is a dynamic place where each moment plenty of events are simultaneously happening at a rapid speed. Some of those events have a direct influence on the lives of regular citizens, like passing a new bill, information about corrupt government, or messages warning about catastrophes. That is one of the reasons why people are required to follow a lot of news sources so as not to miss something crucial. It results in overexposure to media, which on a surface level does not seem to cause much damage, but has a negative impact on humanity's mental state [2]. This problem could be solved if news digests were prepared regularly. That is one way to brief the readers on the most prominent news without consuming too much of their time and energy. However, preparing a news digest requires extra effort from journalists and may take several hours to complete.

Telegram has become extremely common not only as a messenger but as a mass media itself. According to the survey, 72% of Ukrainian respondents cited Telegram channels as the main news source in 2023 among social media, and that number keeps growing throughout the years [3]. No product designed to work with the Ukrainian language that could offer all the necessary functionality to create a news digest conveniently was introduced to the market, despite its popularity and Ukrainian society’s dependence on the news since the beginning of the full-scale russian invasion.

This coursework’s goal is to fill the gap in the currently available resources designed to automate the creation of digests based on the content of Ukrainian-language Telegram news channels.

The object of the coursework is the processing of Ukrainian-language news text data.

The subject of the coursework is text analysis using various natural language processing instruments and methods and their usage to create a chatbot for generating news digests.

To achieve the set goal, the following tasks were completed:

researched available tools and methods of Ukrainian natural language processing;

investigated the creation of digests by Ukrainian-language Telegram news channels;

analyzed available language models with a view to determining the most optimal one in terms of accuracy and reliability for summarization;

developed a chatbot for automatic regular summarization of Ukrainian news Telegram channels.

The scientific novelty of the coursework is:

- for the first time, the dataset of news headlines included in the digests of Ukrainian-language news Telegram channels along with the full texts of the posts was collected;
- for the first time, digests of Ukrainian-language news Telegram channels were analyzed to identify the common structural elements;
- for the first time, the accuracy of summarizing the posts' content of Ukrainian-language Telegram channels by language models was evaluated based on the collected dataset;
- for the first time, the software for creating news digests in Ukrainian was developed in the form of a Telegram chatbot using natural language processing techniques.

The results of the work were presented at the international scientific conference «Scientific Horizons of the 21st Century: Multidisciplinary Research».

The coursework includes an introduction, four sections, general conclusions, and a reference list with 52 titles. The coursework spans 38 pages, 24 of which are of main text, 3 appendices across 3 pages, and 3 figures.

1. ANALYSIS OF AVAILABLE SOLUTIONS

1.1. Definition of NLP

Natural language processing, abbreviated as NLP, is a theoretically driven field of computational techniques for processing and representing naturally occurring texts at one or multiple levels of linguistic analysis. It is used to accurately imitate human-like language comprehension with a variety of ways of applications [4]. Its origins trace back to the 1940s, after World War II. At the time, people started to realize how important and useful the ability to translate into a different language can be, which was the reason they made the first attempts to create a machine that could function in this way. From 1957 to 1970, researchers divided into two groups concerning NLP: stochastic and symbolic. Stochastic researchers were focusing on statistical and probabilistic approaches to NLP, such as developing optical character recognition and pattern recognition between texts. Symbolic, or rule-based, researchers primarily concentrated on formal languages and generating syntax; this group involved plenty of scientists who called this branch the beginning of artificial intelligence [5].

In modern times, natural language processing is still evolving, but it already has lots of applications not only in the business world but in the daily lives of most people, going far beyond programmed translators. For instance, smartphones are equipped with a virtual helper capable of listening to human speech, interpreting it, giving a meaningful and thoughtful response, or even performing described actions. What historically needed to be moderated by human beings now can rely on a program, like detecting spam or filtering dangerous content on social media [6].

1.2. NLP techniques

1.2.1. Text normalization

Text normalization is a fundamental technique when it comes to natural language processing since the program is not able to understand the conversation's semantics in a human-like way, which is why text must be normalized first [7]. Normalization is

transforming the text to reduce it to the standard form that is more convenient to process. The most frequently performed kinds of normalization are tokenization, stemming, and lemmatization.

Tokenization is a process of dividing large pieces of text into smaller chunks that are easier to work with. Most often it is performed to separate the text into sentences, words, or characters [8]. Stemming is the task of reducing a word to its base (stem) by discarding variable or auxiliary parts, such as suffixes or endings. The results of stemming are sometimes very similar to determining the word's root, but due to the differences in the methods, the processed word may look different from the morphological root of the word [9]. Its main drawback lies in the inability to understand the meaning of the word, therefore having similar stems in unrelated words. In particular, the words «місто» (translate: «city») and «міст» (translate: «bridge») have identical stems despite having a huge difference in meaning.

Lemmatization is also used to retrieve the base form of the word. Unlike stemming, it usually turns to proper morphological analysis, the use of vocabulary, and other information to return the correct result [10]. Lemmatization is a better approach since it is less error-prone when the form of the word is completely different from the root.

1.2.2. TF-IDF

TF-IDF is a natural language processing technique used to estimate how important different words are in a sentence. It is applied in various ways, including text classification and to help a machine learning model read words [7].

TF (term frequency) is the weight of a term that is calculated as the proportion of times the word occurs in the text to the total number of words.

DF (document frequency) is responsible for measuring the importance of a document in a corpus. To calculate it, the number of documents with the word occurring is divided by the amount of documents. The next step is to take the logarithm of that number to make calculations easier for the computer and add 1 to ensure it under no circumstances can be 0 since DF stands in the place of the divisor in the next formula.

Some tokens called stop words have little to no importance, and words mentioned in the text the least carry the most meaning. However, all words are considered equally important while computing TF, which is not always true. For this reason, the term's IDF (inverse document frequency) is calculated by dividing the total number of words by the document frequency of the term.

The final TF-IDF result equals the term frequency multiplied by its IDF, which measures how meaningful that word is in the corpus's context [11].

1.2.3. Dependency parsing

Dependency parsing is an essential step to determine the relationships between different words. The first step is to tokenize the sentence and find its grammatical center, which in most cases results in being the sentence's predicate. In case of its absence, another member of the predicative center is indicated as the root. After defining the grammatical center, the dependent tokens are searched and their relations are marked. The dependency tree is built by recursive analysis of all chains of dependencies that are present in the sentence [12]. Because any word cannot depend on more than one and one word is guaranteed to be independent, it is possible to build a dependency tree for every sentence.

1.2. Analysis of NLP instruments

Natural language processing is a broad umbrella term to describe various tasks including but not limited to separating a string with human language text into sentences and words, restoring the word's base form, and recognizing its part of speech and morphological features. The work can be performed with the help of many NLP instruments, such as UDPipe, Stanza, and spaCy.

UDPipe [13] is an NLP tool for tokenization, tagging, lemmatization, and dependency parsing of CoNLL-U files. It is maintained by the Institute of Formal and Applied Linguistics and can provide models for almost every Universal Dependencies treebank, including Ukrainian. The accuracy of the latest version of the model is

estimated to be 99.81% for tokenization, 97.47% for lemmatization, and 90.37% for dependency parsing [14].

Designed by The Stanford NLP Group, Stanza [15] is another instrument for natural language processing. It is a Python package with instruments for natural language processing in more than 70 languages. That is achieved by applying the principles of the Universal Dependencies framework. Its accuracy when working with Ukrainian-language texts is indicated on the official website and is 99.79% for tokenization, 96.72% for lemmatization, and 88.60% for dependency parsing [16].

Spacy (stylized as «spaCy») [17] is an open-source Python library published by the company Explosion. As stated on the website, it is developed specifically for the production use of advanced natural language processing solutions. To work with the texts written in Ukrainian, the uk_core_news pipelines family is used. The calculated precision of the largest is equal to 100% for tokenization and 97% for dependency parsing [18]. However, it is not capable of lemmatizing text, which is needed during the ROUGE estimation performed later.

Based on the results of the analysis, UDPipe was chosen for further work.

2. ANALYSIS OF TELEGRAM CHANNELS WITH NEWS DIGESTS

Ukrainian news Telegram channels were investigated in search of those implementing summarization of their content on a regular basis. Their peculiarities were studied and described to find common features that have to be implemented in the minimally viable product. Among those Ukrainian Telegram news channels using regular summarization of posts, the following were found:

- СЛУХ [19];
- Факти ICTV [20];
- КМДА – офіційний канал [21];
- Ukraine NOW [22];
- hromadske [23];
- Агент Києва [24];
- Суспільне Кропивницький [25];
- СУСПІЛЬНЕ КУЛЬТУРА [26].

2.1. Digests by SLUKH

СЛУХ (transliterated as «SLUKH») is an independent Ukrainian media about Ukrainian and worldwide music, whose mission is to introduce modern Ukrainian music and culture to the world, as it is mentioned on their website [27]. Their main resource is their website with long reads toggling the topics of performing arts of national and worldwide value.

The Telegram channel is used to inform the audience about the release of a new article, as well as to mention small news posts not worthy of extensive material. The predominant content here is excerpts from larger texts with a link to a full article. Their news selections are published every week.

The channel was created on March 28, 2018, but it was not until March 2020 that they started making weekly summaries. Out of every channel analyzed, their digests cover the most news — more than 15 on average. It is notable that with every article mentioned in the summary, the main word in all of their sentences is formatted as a link,

sending the user to the full message or the original article on their website. They also use contextual emojis at the beginning of a headline to indicate what the news is about. Each of their news posts has a headline, which in most cases is similar to the sentence used in a summary to reference the original news. They follow a strict template since all of their recent summaries start with the phrase «Що було #наслуху минулого тижня:» (translate: «What was #oneveryoneslips last week»). Despite the fact this phrase has faced multiple changes throughout its existence, the same hashtag was present in all their past overviews. The mentions of the news are separated with a blank line.

2.2. Digests by Факти ICTV

«Our goal: to form the most complete picture of life in Ukraine and the world. / This is an internet portal about people and for people. We reflect what Ukraine and the world live by.», — is what is said by representatives of Факти ICTV (transliterated as «Fakty ICTV») about their media on their website [28]. Fakty ICTV is a Telegram channel with news for a mass audience, which, as the name suggests, is produced by the Ukrainian TV channel ICTV. The Telegram channel was created on October 18, 2016. They have altered their template several times, making it difficult to track when exactly they started posting regular daily recaps. An example of the earliest digest found is dated June 15, 2017.

At the moment of writing, this channel publishes summaries three times a day. Of all the channels that were investigated, this is the only one to do it with that frequency. The first two overviews show the main totals for a certain moment of the day. One such is published around noon, the other one — around 18:00. The third shows the most important news throughout the day. It is usually published between 21:30 and 22:00.

Just like SLUKH, they follow a guideline to make all of their summaries more similar to each other. In particular, they start all their summaries with the phrase, including a hashtag #новини (translate: «#news»). The date is placed in the header if it is a general summary of news occurring on a certain day. If the summary is based on some point of the day, the introductory phrase also includes the creation time. Unlike the previous media, they have a footer with links to the other platforms where they can

be found. The links to the full articles on their website are also present. Furthermore, they use emojis as bullet points before every news, however, it does not become different depending on the context, as it is a white rectangle. Their recent summaries contain 4 to 8 news excerpts.

2.3. Digests by КМДА – офіційний канал

КМДА – офіційний канал (translate: «KMSA – official channel») is another Ukrainian Telegram news channel. It is stated in its bio that the channel belongs to and is managed by the Kyiv Municipal State Administration's representatives. It was created on March 30, 2020, but it was only in April 2022 that they started posting daily news overviews. However, that August the idea of daily digests was abandoned in favor of doing it once a week. Apart from the frequency of digests, no drastic changes were made. Each summary starts with the phrase «Київ: найважливіше за тиждень🔵🟡» (translate: «Kyiv: most important during the week🔵🟡») followed by the most important news occurring that week. The main words in the headlines are linked to the full article on the website or the full Telegram post. This is the only investigated channel that does not limit news text to a headline but provides more context with a few sentences. It is restricted to use more than 4096 characters in a Telegram post, which impacts the quantity of news they can highlight with that approach. Typically, their summaries feature 4 to 6 news. In a similar way to SLUKH, they put a matching emoji before the headline of each news and separate descriptions of different events with a blank line. The Ukrainian text ends with the words «Детальніше на kyivcity.gov.ua.» (translate: «More details on kyivcity.gov.ua»), where the link forwards the user to their website. After the Ukrainian part of the digest is done, it is translated into English and formatted in the same style. This is the only channel studied to provide English translations of their digests.

2.4. Digests by Ukraine NOW

Ukraine NOW originally started as an international marketing campaign of the Ukrainian government whose goal was to shape the brand of Ukraine in the world,

attract investments to the country, and improve the tourism potential [29]. Their Telegram channel was created on March 5, 2020, and it mostly covered topics of COVID-19 and its influence on Ukraine. Since the start of the full-scale invasion Ukraine NOW has transformed into one of many non-profit initiatives that arose as a response to russian aggression [30], and so did their channel. Its main coverage is news about Ukraine, its people, and the war realities.

Digests were published by Ukraine NOW ever since its creation, however, on August 9, 2022, they posted the first daily overview with a template that resembled the one they were still using at the time of writing. It features a headline that starts with the day («Головні новини за…»; translate: «Main news for…»), followed by the day of the summary. The emoji of the black square is used as a bullet point before each headline in a digest. Every digest is closed with the emoji of a check mark and a call to follow the channel with a hyperlink. The links to the full posts are also present as the main word of each headline. No matter how many posts were published during the day, only the most significant material is included in that day's digest, which is why the amount of news in the digest varies.

2.5. Digests by hromadske

Hromadske (stylized in lowercase) is a not-for-profit independent organization established by journalists in 2013 that aims to independently and impartially convey important messages to society through all available means [31]. Their Telegram channel was created on June 23, 2016, and since the end of 2018, it has been regularly posting daily news digests.

Hromadske's digests start with the most significant news as a headline. Divided with a dash, the template phrase «важливе за день:» (translate: «significant of the day») is placed next to it. Each news is shortened to its respective headline and listed on the digest, starting with an orange diamond emoji functioning as a bullet point marker and preceding a blank line. Each overview is finished with the organization's branding and a hyperlink to the channel attached to the word «підписатися» (translate:

«follow»). The amount of news featured varies from 4 to 6. Each news headline has a link to the full article attached to the main word in the sentence.

2.6. Digests by Агент Києва

Агент Києва (translate: «Agent of Kyiv» is a private channel owned by an anonymous author. Its content consists of news about the war in Ukraine and Kyiv's realities. The first news digest was posted on the same day the channel began its activity — February 11, 2024. Since then, the recap was posted no less than 2 hours before the end of every day. The author confirmed that all digests are prepared manually with the news being selected subjectively. Each digest features 3 to 6 headlines with the links to the original posts. The informative part is situated between the phrase «Головне за день:» (translate: «The main thing of the day») and a closing link to the channel with the text «Агент Києва | Підписатись» (translate: «Agent of Kyiv | Subscribe»). Similarly to SLUKH, every post that is published to the channel contains a headline, which is used in the digest while referencing it.

2.7. Digests by Суспільне Кропивницький

Public Broadcasting Company of Ukraine, referred to as Suspilne, is a social and political internet publication belonging to the Ukrainian society and financed by citizens' taxes [32]. Суспільне Кропивницький (translate: «Suspilne Kropyvnytskyi») is one of the many regional Telegram news channels in its ownership, which was created in January 2020. It posted its first news summary in 2020, but since May 2022 it became a daily routine to make a news digest with the most important events of the day.

Like Ukraine NOW, the quantity of news featured on its digests varies depending on how many events occurring that day could be considered essential to highlight. Every digest has a simple structure: starting with the date followed by the phrase «Новини дня» (translate: «News of the day»). Unlike most channels, it does not have a standard ending that would be used in the post. However, each headline on the digest has a similar emoji as a bullet point and a link to the full article on the website.

2.8. Digests by Суспільне Культура

Суспільне Культура (stylized in uppercase; translate: «Suspilne Culture») is another product of PBC. The channel is exclusively focusing on the culture of Ukraine and the world. The Telegram channel was created in July 2022, and since the beginning, they have been posting the news digests every week. The number of featured news varies but is close to 10.

Their digests start with mentions of the most important news of the week, followed by the headline «— головні новини тижня від Суспільне Культура:» (translate: «main news of the week from Suspilne Culture»). All news is separated with a blank line. The blue diamond emoji is used as a bullet point for most news. Sometimes Suspilne Culture features more than one news on the same topic in the digest. In this case, the name of the topic is written instead of the full news, and then the news is published with a separate bullet point in the form of a black square. As in previous channels, the main word of each news item is highlighted with a hyperlink to the full article. The digest is ended by the standard phrase «Щоб не пропустити головні новини культури, підписуйтеся на Telegram-канал Суспільне Культура.» (translate: «Follow Suspilne Culture's Telegram channel not to miss the culture's main news.»), containing a hyperlink to the channel.

2.9. Conclusion of a section

As a result of the analysis, it was found that all examined Ukrainian news Telegram channels insert links to full articles in the headline of the news they are abbreviating, with 88% percent of the channels surveyed highlighting the main word in the headline by doing so. Each channel has its template of bullet points, a title, and sometimes an ending. The number of separator lines varies between channels but is the same among all posts of one channel. It was established that each channel also has a certain frequency with which it posts news, 50% do it once a week, 38% every day, and 12% — several times a day.

3. ANALYSIS OF MODELS FOR SUMMARIZATION

3.1. Definition of summarization

Summarizing is a natural language processing task of formulating the main idea of the text in order to make it shorter by conveying the essence. Two existing kinds of approaches to summarization are extractive and abstractive [33].

Extractive summarization is achieved by working with the source text's material. All the terms are investigated to find those carrying the most meaning, then they are used like building blocks to generate the final result. This approach ensures that the words in the summary are the same as those present in the original text.

In contrast to it, abstractive summarization does not rely on the specific words in the original text. It uses a model to generate a paraphrase of the key aspects, which is why the words may be different as a result. However, this way of creating a summary feels closer to how an actual human being does it, thus feeling more natural.

3.2. ROUGE Analysis

ROUGE (Recall-Oriented Understudy for Gisting Evaluation) is a family of metrics commonly used in natural language processing and text summarization tasks to evaluate the quality of machine-generated summaries called predictions by comparing them to man-made reference summaries. It calculates the summary's precision and recall rate based on the overlapping parts between the reference and the prediction. It was selected because it provides an objective way to evaluate the quality of generated summaries and is language-agnostic as long as the text can be normalized [34].

The ROGUE metrics used are ROUGE-1, ROUGE-2, and ROUGE-L.

ROUGE-1 counts a percentage of overlapping unigrams, or single-word tokens, in the reference and the model's prediction. The recall rate is the number of unigrams that are present in both the summary and the reference divided by the total number of unigrams in the reference summary. This number represents how informative the summary is.

The precision is calculated to know how wordy the summary is by dividing the number of overlapping unigrams by the total number of unigrams in the prediction. ROUGE-2 operates in the same way except for comparing bigrams, meaning couples of consecutive tokens. By doing this, the evaluation considers the order of the words, not only their presence in the sentence. ROUGE-L's recall is a fraction of the number of words in the longest common sequence of tokens between the summaries and the number of words in the model's generated summary. Similarly, the length of the longest common sequence is divided by the number of words in the reference summary.

The F1 score is a resulting measure that combines recall and precision. It is calculated as their harmonic mean: $F1_{ROUGE} = 2 * \frac{recall * precision}{recall + precision}$.

ROUGE is one of the best ways to evaluate extractive summarization since all the words from the original text are present in the generated version, thus having a major chance of words overlapping. However, it seems that it does not do justice to abstractive summarization, because a prediction with a semantically accurate summary that replaces some words with their synonyms is only punished for the additional hard work. For that reason, another metric had to be applied.

3.3. BERTScore

BERTScore is another metric used for evaluating the accuracy of machine-generated summaries. Unlike ROUGE, it is considerate of the word's meanings, therefore not only the appearance is compared but also the semantic differences [35]. It is based on the BERT (Bidirectional Encoder Representations from Transformers) model, a state-of-the-art pre-trained language representation model developed by Google.

To evaluate the accuracy of summarization, BERTScore relies on the human-made summary as well. The difference is that after tokenizing the input, the contextual embedding for each word is generated. Once every word's embedding is ready, each word of the candidate is paired with each word of the reference to compute precision,

and each word of the reference is paired with each word of the candidate to compute recall. The computation is pairwise cosine similarity.

However, rare words are more reliable indicators of a good summary than common words, which is why the importance weighing can be performed by calculating the IDFs, the process was described alongside TF-IDF.

With x being the reference tokens and $\hat{x}$ — the candidate tokens:

$$R_{BERT} = \frac{\sum_{x_i \in x} idf(x_i)\, max_{\hat{x}_j \in \hat{x}}\, x_i^\top \hat{x}_j}{\sum_{x_i \in x} idf(x_i)}$$

$$P_{BERT} = \frac{\sum_{\hat{x}_j \in \hat{x}} idf(\hat{x}_j)\, max_{x_i \in x}\, x_i^\top \hat{x}_j}{\sum_{\hat{x}_j \in \hat{x}} idf(\hat{x}_j)}.$$

Similarly to ROUGE, BERTScore F1 is calculated as the harmonic mean of precision and recall:

$$F1_{BERT} = 2 * \frac{recall * precision}{recall + precision}.$$

Overall, BERTScore considers the contextual representations learned by BERT to provide an effective measure of the quality of the machine-generated text, taking into account semantic aspects of the language. It has been shown to correlate well with human judgment in various text-generation tasks and is widely used in natural language processing research and evaluation.

3.4. Summarization accuracy evaluation

To evaluate the accuracy of summarization performed by models, a dataset of 100 posts that were published to the Ukrainian news Telegram channels was collected. Apart from the posts' full texts, it included the headlines that were used to reference the respective posts in news digests. The content of multiple news channels was used, including «СЛУХ», «Агент Києва», «КМДА — офіційний канал», and «Ukraine NOW», to ensure the diversity of news topics because the final product has to work best with a wide range of news categories.

Since both the text of the news and the result of its reduction done with human supervision are available for every post, it is possible to apply reference-based ROUGE and BERTScore metrics to evaluate the work of the models.

The evaluated models included Llama 3 by Meta [36], Copilot by Microsoft [37], ChatGPT 3.5 by OpenAI [38], Gemini 1.0 Pro by Google [39], SGaleshchuk's t5-large-ua-news [40], ukr-models' uk-summarizer [41] and BUET CSE NLP Group's mT5_multilingual_XLSum [42].

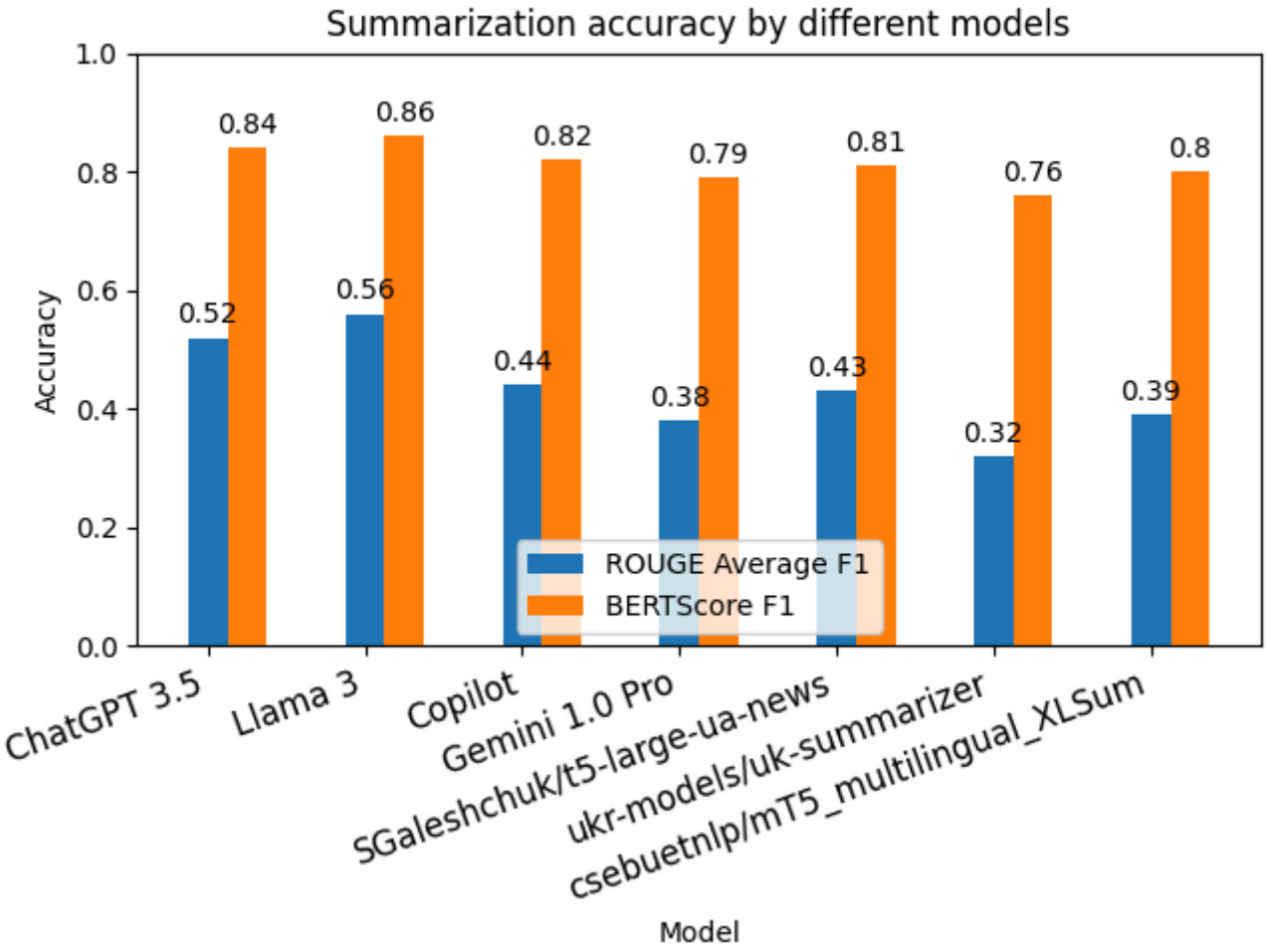


Figure 3.1. Results of summarization accuracy evaluation

3.5. Conclusion of the section

As a result of the evaluation, it was determined that the best results were shown by Llama 3 in both ROUGE and BERTScore. This means that the model did the best job of conveying the meaning of the news story's full text by summarizing it in the same way a

human would, and did so with minimal paraphrasing, making it consistent with human-produced headlines.

Hence, Llama 3 is selected for use as a model for summarization based on the results.

4. DEVELOPMENT OF THE BOT

The structure of a typical news digest was determined earlier, but a few more steps have to be taken to prepare such summaries. First, the news published over a certain time must be examined to find the most relevant. After that, a catchy headline has to be assigned to each of the texts. This requires content creators to carefully read the news stories and find a headline that would be simple enough for the majority of users to understand, and short enough to pass all platforms' restrictions. It should also be engaging enough so that a reader would want to proceed to read the whole article. The process can be time-consuming when done manually, however, this task can be streamlined by analyzing the content and generating catchy headlines automatically with the use of natural language processing, saving time and ensuring consistency in quality and style. In this section, the stages of replicating such functionality are described to provide an argumentation on the made decisions.

4.1. Creating a bot token

Before developing a Telegram bot, the bot token must be generated. To do this, BotFather [43] is used. It is an official Telegram bot that manages bots, apps, and games created by the users of Telegram. To create a bot on Telegram, the /newbot command is sent. Once a bot token has been received, it is time to set up the future bot.

4.2. Connecting to the bot

The program is connected to the bot with the Python Telegram Bot library [44]. It is a common solution for developing a Telegram bot because of its speed, asynchronous operations support, regular improvements, bug fixes, and customization opportunities. In particular, it supports both polling and webhooks, which can lead to easy scalability.

To start the bot, the following code is used:

main.py

```
1 from telegram.ext import Application
2 from config import Config
3 app = Application.builder().token(Config.TELEGRAM_BOT_TOKEN).build()
```

Figure 4.1. Bot connection code

4.3. Accessing messages

For privacy purposes, Telegram prohibits bots from accessing user messages both in private conversations and in public channels. Another library — Telethon [45] — was used to solve this problem. It can connect to Telegram not only via the Bot API but also with the TDLib API [46]. This is another kind of Telegram API provided to developers to implement unofficial Telegram applications or apps that require access to user chats. On the Telegram website, the request is made to generate a TDLib API and hash. To work with it, a Telethon client was added to the code.

main.py

```
1 from telethon import TelegramClient
2
3 client = TelegramClient('anon', Config.TELEGRAM_CORE_API_ID,
  Config.TELEGRAM_CORE_API_HASH)
```

Figure 4.2. Client connection code

The messages are accessed with the method client.get_messages, which takes a chat ID, and many optional arguments, such as the limit and the offset date. Thanks to this, the bot can access messages in user's chats.

4.4. Handlers

To define a handler, the asynchronous functions that take the update and the context as arguments have to be implemented, and then passed to the handler's constructor. Updates are objects with detailed information about the messages or callbacks that triggered the handler. The information includes the sender's name,

username, ID, the content of the message, whether it is original or was forwarded from a different channel, and other statistics. The context contains temporary data about a bot, a user, or a chat that triggered the handler. It is user-specific and unless changed remains the same for every update from the same user.

4.4.1. Conversation handler

Traditionally, the Python Telegram Bot library uses command or message handlers to process input. However, if the user's actions are different at some stages when working with the bot, the conversation handler has to be used. For that, the different states are defined as integers, and each state is assigned its own set of handlers. It greatly expands the capabilities of the bot, since the commands do not necessarily have to be supported in parallel, but are used to build a certain chain of interactions.

4.4.2. Command handlers

Command handlers are responsible for processing the commands sent by a user. Some of those commands are: /start, /help, /add_service, and /manage_services. They are supposed to be achievable at any moment. Other commands include /cancel and /empty. They are only used in the states of user input when the user wants to leave a field empty or cancel the operation.

4.4.3. Message handlers

The message handlers are responsible for processing the forms of user input outside of commands. In this bot, they are used to handle the user input while adding the channel to the bot, renaming the channel in the database, and setting the digest template consisting of the headline, the ending, and custom bullet points, since they cannot be chosen from a multiple choice list.

4.4.4. Command handlers

When Telegram bots first were popularized, keyboard buttons were the primary tool to give users a choice of options to pick from. However, since Bot API 2.0 [47] was introduced, most of the interaction in the bots has happened via inline keyboard buttons. Another type of handler used is a callback query handler. Unlike regular keyboard

buttons that appear instead of the input field and send the predefined phrases on tap, inline keyboard buttons are attached to a certain message. They interact by sending callbacks to the bot. The callback information is defined during the creation of every inline keyboard button.

4.5. Database

Now that the bot is able to read information about its users and process the messages, it needs a data structure to store information about the added channels. For this purpose, an SQLite [48] database was chosen. This is the database designed to be lightweight and fast, and it achieves it by eliminating the server or configuration files. Also, the database is natively supported in Python with the module *sqlite3* [49], which helps prevent SQL injections with the use of prepared statements [50].

The main entities in the database are users and channels. The users table contains information about the users that started the bot, including their names and IDs. The channels table contains all the digests' settings for each channel added by the channel's owner. The columns include the ID of the digest's settings, the user ID, the channel ID, the channel's name, the date of making the last digest, the date of making a new digest, the frequency of the digest's creation, the least positivity of the news featured in a summary. Information about the digest's template (headline, ending, dividing lines between news, bullet points, limits for the number of news) is also included.

4.6. Scheduler

To generate news digests automatically, a scheduling system is implemented to handle the timely execution of the process. One of the most frequently used scheduling systems is Advanced Python Scheduler, reduced to APScheduler [51]. It was chosen because of the ability to schedule tasks to a certain date and the support of asynchronous work. To store the information about future digests, the respective jobs are added to the scheduler with the ID of the digest. When the user changes the frequency of generating digests or launches the digest early, the date is moved. This is an efficient decision that helps the bot track time without requiring regular database monitoring.

4.7. Creation of a digest

When the generation of a digest is started, first a bot captures the messages that were posted during the necessary period. If the number of published messages is smaller than what the user set, the message is sent to the bot that not enough messages were published to make an overview. Since news digests are made with the intention to relieve readers of the need to read long materials in search of the main news, they are optional when there is not much news to begin with. On collecting enough material, it is filtered by its positivity gathered from the audience's reactions and sorted by its relevance. The task is to highlight the most significant events of the time frame. The importance of an event can be evaluated by calculating how it resonated with the audience of the media. Multiple methods are used depending on the given information and the context. ERR (Engagement Rate based on Reach) takes into account the total amount of engagements divided by the total amount of people that have seen the post. ERI (Engagement Rate based on Impression) and ERV (Engagement Rate by Views) are two metrics, that measure the audiences' feedback when they see the post [52].

In Telegram there are several ways the post can be interacted with: users can leave reactions or forward the post to others, if the respective options were activated in the channel's settings, or post replies, if a chat for comments was attached to the channel. Presuming that if the user does not engage with a message when they see it, it is not particularly interesting to them, it is possible to rank the news by their relevance if that statistic is available during the analysis. Since every message's view counter is only increased the first time the post is seen by any account during the day, it seems reasonable to sort the news based on their ERR, dividing the total amount of post's engagements by its reach.

When the news is sorted, the text is shortened to the headline with the Llama 3 model, which was evaluated to be the most human-like when dealing with a task of summarization. The next step is determining the main word of each headline, which is done by dependency parsing and taking the root. Then, according to the given template, the text of the digest is formed and sent to the user.

4.8. Conclusions of the section

The Telegram chatbot for summarizing the content of Telegram news channels was developed. The functionality supported by the bot includes:

- adding new channels to the bot;
- removing channels from the bot;
- receiving digests automatically;
- toggling the digest generation;
- renaming the channel;
- setting the frequency of automatic summarization;
- setting news' minimum positivity;
- setting the limit on the quantity of news in the digest;
- setting the digest's template;
- generating digests based on the content of a certain Telegram channel.

CONCLUSIONS

The goal was achieved by addressing the gap in the available resources for automating the creation of digests based on the content of Ukrainian-language Telegram news channels.

The tasks were accomplished due to the following:

- a dataset of news headlines included in the digests of Ukrainian-language news Telegram channels, along with the full texts of the posts, was collected.
- digests of Ukrainian-language news Telegram channels were analyzed to pinpoint common structural elements.
- the accuracy of summarizing the posts' content of Ukrainian-language Telegram channels by language models was evaluated based on the collected dataset
- a chatbot for automatic regular summarization of Ukrainian news Telegram channels' content was developed.

The results of this work were approved by the committee of the international scientific conference «Scientific Horizons of the 21st Century: Multidisciplinary Research».

The developed Telegram chatbot optimizes the process of generating news digests based on the content of Ukrainian-language news channels by applying natural language processing techniques. The accompanying research lays the groundwork for future advancements and applications in this area, as similar functionality could be implemented on different platforms using the principles described in this coursework.

REFERENCES

1. Heim S., Keil A. Too much information, too little time: how the brain separates important from unimportant things in our fast-paced media world. Frontiers for Young Minds. URL: https://kids.frontiersin.org/articles/10.3389/frym.2017.00023#:~:text=Scientists%20have%20measured%20the%20amount,billboards,%20and%20many%20other%20gadgets (date of access: 05.04.2024).
2. Charlotte H. Media overload is hurting our mental health. Here are ways to manage headline stress. https://www.apa.org. URL: https://www.apa.org/monitor/2022/11/strain-media-overload (date of access: 05.04.2024).
3. Поліковська Ю. За рік кількість українців, які отримують новини з телеграму, зросла ще на 12%, – дослідження. ms.detector.media. URL: https://ms.detector.media/sotsmerezhi/post/33364/2023-11-01-za-rik-kilkist-ukraintsiv-yaki-otrymuyut-novyny-z-telegramu-zrosla-shche-na-12-doslidzhennya/ (дата звернення: 28.04.2024).
4. Liddy E. *Natural Language Processing*. Encyclopedia of Library and Information Science. 2nd ed. NY, 2001. P. 3. URL: https://surface.syr.edu/cgi/viewcontent.cgi?article=1043&context=istpub (date of access: 01.05.2024).
5. Roberts E. *Natural language processing. Overview - History.* Computer Science. URL: https://cs.stanford.edu/people/eroberts/courses/soco/projects/2004-05/nlp/overview_history.html (date of access: 01.05.2024).
6. IBM. *What Is Natural Language Processing?* URL: https://www.ibm.com/topics/natural-language-processing (date of access: 08.05.2024).
7. Gillis A. S., Lutkevich B., Burns E. *What is natural language processing?*. TechTarget. URL:

https://www.techtarget.com/searchenterpriseai/definition/natural-language-processing-NLP (date of access: 01.05.2024).

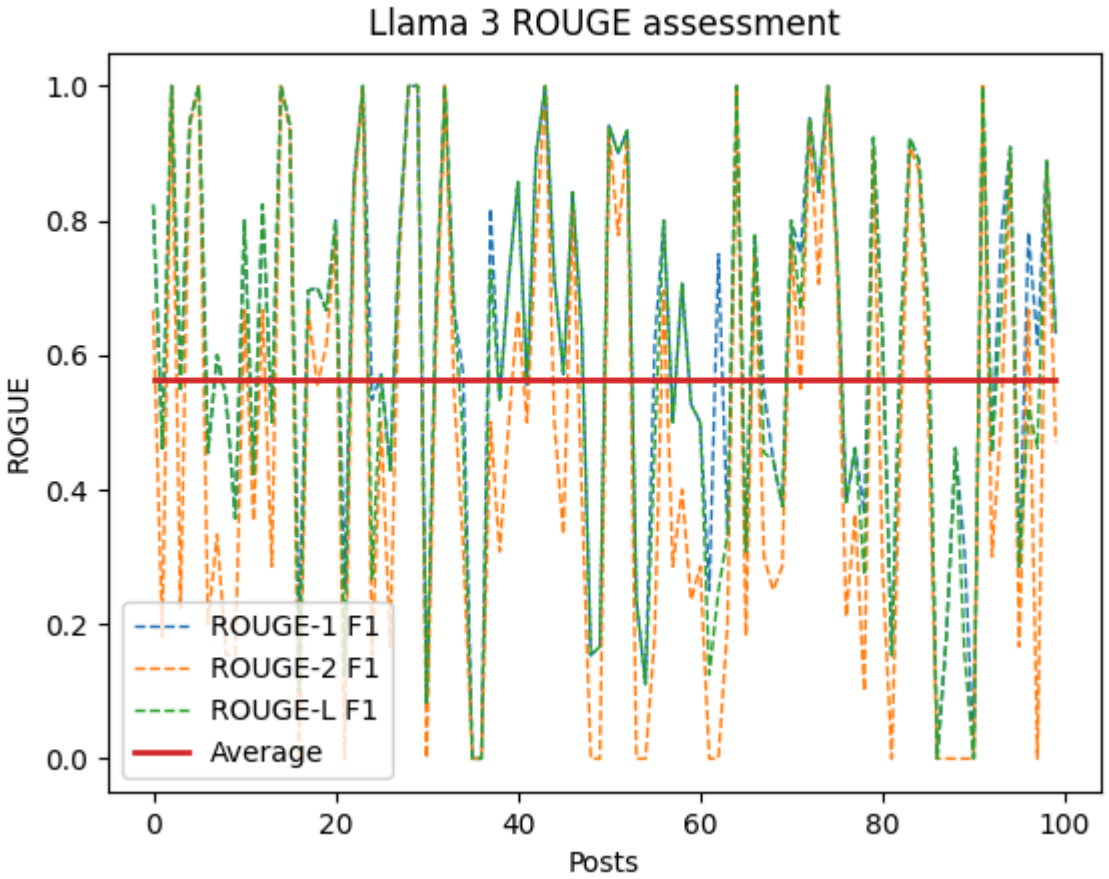
8. Глибовець А.М., Точицький В.В. Алгоритм токенізації та стемінгу для текстів українською мовою, УДК 004.912:811.161.2.
9. Grefenstette G., Tapanainen P. What is a word, what is a sentence? Problems of tokenization. 1994. URL: https://www.dfki.de/~neumann/qa-course/grefenstette94what.pdf (date of access: 08.05.2024).
10. Jurafsky D., Martin J. H. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition (3rd draft)*. 2023. 418 p. URL: https://web.stanford.edu/~jurafsky/slp3/ed3book.pdf (date of access: 26.04.2024).
11. Hamdaoui Y. TF-IDF: An Introduction. *Built In.* URL: https://builtin.com/articles/tf-idf (date of access: 14.05.2024).
12. Лангенбах М. Автоматичний синтаксичний аналіз речення за принципами граматики залежностей. *Науковий вісник Східноєвропейського національного університету імені Лесі Українки. Філологічні науки. Мовознавство.* 2015. № 3. С. 249–254. URL: http://nbuv.gov.ua/UJRN/Nvvnuflm_2015_3_48 (дата звернення: 30.04.2024).
13. Straka M. UDPipe. Version 2.0. URL: http://ufal.mff.cuni.cz/udpipe.
14. UDPipe 2 Models. ÚFAL. URL: https://ufal.mff.cuni.cz/udpipe/2/models#universal_dependencies_212_models (date of access: 08.05.2024).
15. Stanza: A Python Natural Language Processing Toolkit for Many Human Languages / P. Qi et al. *Association for Computational Linguistics (ACL) System Demonstrations.* 2020. URL: https://nlp.stanford.edu/pubs/qi2020stanza.pdf (date of access: 19.04.2024).
16. Model Performance. Stanza. URL: https://stanfordnlp.github.io/stanza/performance.html (date of access: 11.05.2024).

17. Explosion. spaCy. 2016. URL: https://spacy.io/.
18. Ukrainian. *spaCy Models Documentation.* URL: https://spacy.io/models/uk (date of access: 11.05.2024).
19. СЛУХ. *Telegram*. URL: https://t.me/slukh.
20. Факти ICTV. *Telegram*. URL: https://t.me/FAKTY_ICTV.
21. КМДА — офіційний канал. *Telegram*. URL:https://t.me/KyivCityOfficial.
22. Ukraine NOW. Telegram. URL: https://t.me/UkraineNow.
23. hromadske. Telegram. URL: https://t.me/hromadske_ua.
24. Агент Києва. *Telegram*. URL: https://t.me/+L3jYnY12z_o1OGEy (date of access: 14.05.2024).
25. Суспільне Кропивницький. Telegram. URL: https://t.me/suspilnekropyvnytskyi.
26. СУСПІЛЬНЕ КУЛЬТУРА. *Telegram*. URL: https://t.me/suspilne_culture.
27. СЛУХ. About Slukh. URL: https://slukh.media/en/slukh-about/ (date of access: 17.05.2024).
28. Про Факти. *Факти ICTV*. URL: https://fakty.com.ua/ua/about_site/ (дата звернення: 07.05.2024).
29. Ukraine Now. Бренд України у світі. *banda.agency.* URL: https://banda.agency/ukrainenow/ (дата звернення: 11.05.2024).
30. Help Ukraine. *UkraineNow*. URL: https://www.ukrainenow.org/ (date of access: 11.05.2024).
31. Hromadske. About us. hromadske.ua. URL: https://hromadske.ua/en/about (date of access: 03.05.2024).
32. Суспільне | Новини. Головна. URL: https://suspilne.media/ (дата звернення: 01.05.2024).
33. What is NLP Text Summarization: Benefits & Use Cases. *Accern.* URL: https://www.accern.com/resources/what-is-nlp-text-summarization-benefits-use-cases (date of access: 12.05.2024).

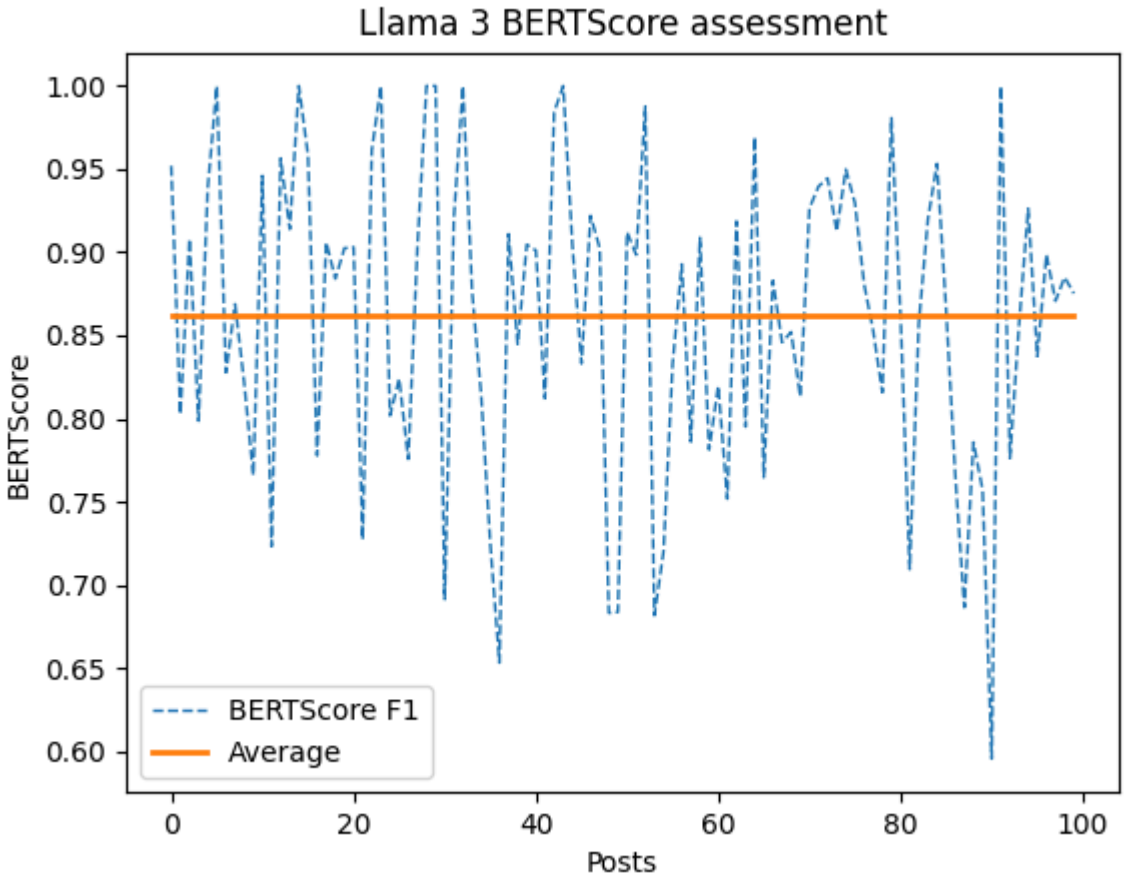
34. Chin-Yew Lin. 2004. ROUGE: A Package for Automatic Evaluation of Summaries. In Text Summarization Branches Out, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.
35. BERTScore: Evaluating Text Generation with BERT / T. Zhang et al. 2020. URL: https://doi.org/10.48550/arXiv.1904.09675.
36. Meta. Llama. Version 3. 2024. URL: https://llama.meta.com/llama3
37. Microsoft. Copilot. 2024. URL: https://copilot.microsoft.com
38. OpenAI. ChatGPT. Version 3.5. 2024. URL: https://chat.openai.com
39. Google. (2024). Gemini. Version 1.0 Pro. 2024. URL: https://gemini.google.com
40. Galeshchuk S. t5-large-ua-news. 2022. URL: https://huggingface.co/SGaleshchuk/t5-large-ua-news.
41. ukr-models. uk-summarizer. 2023. URL: https://huggingface.co/ukr-models/uk-summarizer.
42. BUET CSE NLP Group. mT5_multilingual_XLSum. 2021. URL: https://huggingface.co/ukr-models/uk-summarizer.
43. Telegram Messenger Inc. BotFather. 2015. URL: https://t.me/BotFather.
44. Python Telegram Bot. Version 21.1.1. 2024. URL: https://python-telegram-bot.org/.
45. Telethon. Version 1.35.1. URL: https://docs.telethon.dev/en/stable/.
46. Telegram APIs. *Telegram*. URL: https://core.telegram.org/ (date of access: 13.02.2024).
47. Introducing Bot API 2.0. *Telegram APIs*. URL: https://core.telegram.org/bots/2-0-intro (date of access: 03.04.2024).
48. SQLite. Version 3.45.3. 2024. URL: https://www.sqlite.org/ (date of access: 26.04.2024).
49. Häring G. sqlite3. Version 3.12. URL: https://docs.python.org/3/library/sqlite3.html.

50. SQL Injection Prevention. *OWASP Cheat Sheet Series*. URL: https://cheatsheetseries.owasp.org/cheatsheets/SQL_Injection_Prevention_Cheat_Sheet.html (date of access: 28.03.2024).
51. Grönholm A. APScheduler (Advanced Python Scheduler). Version 3.10.4. 2024. URL: https://apscheduler.readthedocs.io/en/3.x/ (date of access: 17.04.2024).
52. Dewi N. F., Riyadi D., Santoso R. K. Social Media as a Platform for Information: Study in the Marketing Department at Hospital X. *Proceedings of the 6th International Conference on Vocational Education Applied Science and Technology (ICVEAST 2023)*. 2023. P. 187–188. URL: https://doi.org/10.2991/978-2-38476-132-6_17 (date of access: 02.05.2024).

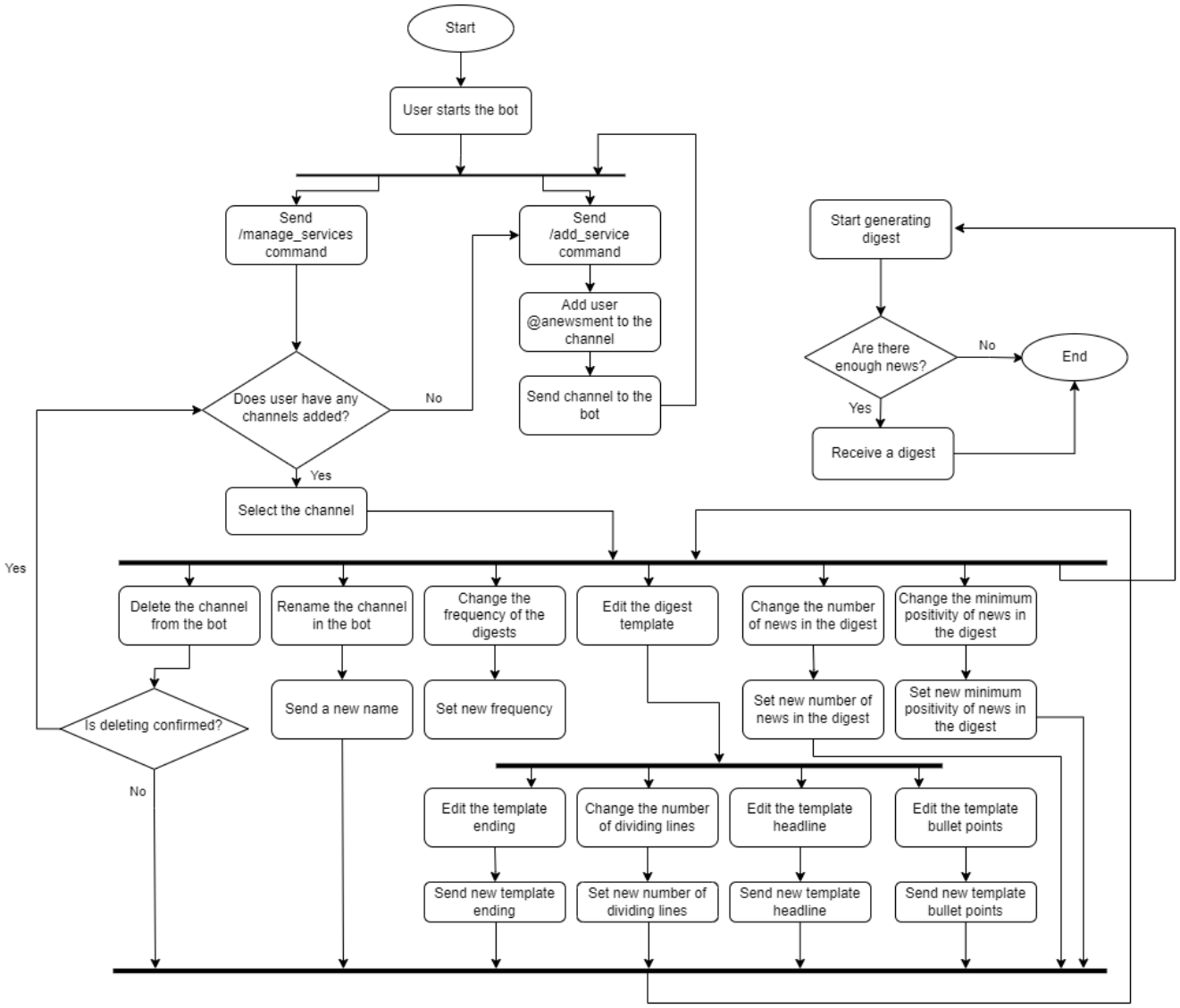
APPENDICES



Appendix A. Result of Llama 3's ROUGE assessment



Appendix B. Result of Llama 3's BERTScore assessment



Appendix C. User's workflow from starting the bot to launching the digest's creation