OSTIS Technology Integration with Third-party NLP Services

Alena Cherkas, Aliaksandr Kupo
Francisk Skorina
Gomel State University
Gomel, Belarus
Email: alenacherkas.iss@gmail.com, kupo@gsu.by

***Abstract*—This article is devoted to the integration of OSTIS Technology with third-party services based on neural networks. As an example of integration, the task of keeping minutes of meetings is considered, the solution of which is based on transcription and summarization. Transcription is carried out using the Whisper service, summarization — retext.ai.**

***Keywords*—OSTIS Technology integration; neural networks; transcription task; summarization task; Natural Language Processing**

I. Introduction

After the appearance of ChatGPT, its analogues and other services based on neural networks in the general availability, a large number of opportunities appeared in the professional activities of representatives of different professions, scientific activities, and personal life. In particular, the scope of application of OSTIS technology has expanded [1].

Integration of OSTIS Technology with third-party services based on neural networks allows you to create universal components adapted to reuse. Thus, for example, when using ChatGPT, it becomes possible not only to get an answer to a given question, but also to save it by formalization, that is, writing on SC-code [2], thereby replenishing the global knowledge base. After that, the system using this knowledge base will be able to process the information received to solve other tasks [3].

Thus, when integrated with ChatGPT or its analogues, the Nika dialog platform, operating on the basis of OSTIS, can access the advantages of dialog systems representing a neural network, while maintaining the advantages of OSTIS.

ChatGPT can also generate insufficiently correct responses. If the user does not have enough knowledge in the field of the question being asked, then it is almost impossible to distinguish truth from lies in the neural network response. This reduces the quality of work with the service and its capabilities. The integration of ChatGPT with OSTIS allows you to identify incorrect information in the responses of the service.

The task of integrating ChatGPT with OSTIS, namely the Nika dialog system, has already been solved. However, OSTIS Technology can be integrated with services of completely different orientation [1], working on the basis of neural networks. Thus, it is possible to optimize their work, expand functionality, and reduce the number of shortcomings [4].

II. Overview of existing approaches

The possibility of integration is demonstrated by the example of to creating a meeting protocol. To do this, the tasks of transcription and summarization will be solved. To convert an audio recording into a text format, there are many services based on neural networks. One of the available services is Whisper from OpenAI.

Whisper is an Automatic Speech Recognition (ASR) system trained on more than 650 thousand hours of multilingual and multitasking controlled data collected from the Internet. The developers of the service note that using such a large and diverse data set leads to increased resistance to accents, background noise and technical language. In addition, Whisper allows you to perform transcription in several languages, as well as translation from these languages into English.

Whisper is built on the transformer architecture, stacking encoder blocks and decoder blocks with the attention mechanism propagating information between both. It will take the audio recording, split it into 30-second chunks and process them one by one. For each 30-second recording, it will encode the audio using the encoder section and save the position of each word said, and leverage this encoded information to find what was said using the decoder.

The decoder will predict what user call tokens from all this information, which are basically each words being said. Then, it will repeat this process for the next word using all the same information as well as the predicted previous word, helping it guess the next one that would make more sense.

The overall architecture is a classic encoder-decoder that is similar to GPT-3 and other language models. OpenAI open-sourced their code and everything instead of an API, so everyone can use Whisper as a pre-trained foundation architecture to build upon and create more powerful models for his necessities.

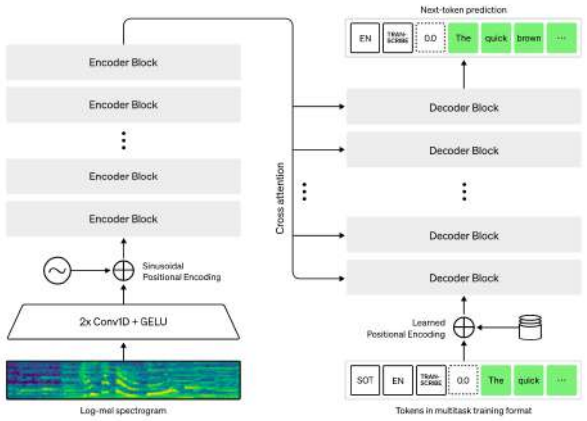

Figure 1. Whisper architecture [5]

Transcription of audio into text takes place in several stages. The audio signal that is recorded on the device may contain noises, echoes, pauses and other acoustic features that make speech recognition difficult. Therefore, before transcription, the audio signal passes through various filters and noise reduction algorithms that allow you to get a cleaner sound.

After preprocessing the audio signal, speech recognition algorithms are superimposed on it. Usually these algorithms are based on neural networks that are trained to recognize speech in different conditions. These networks are used to recognize phonemes, words and phrases in audio.

After the neural network recognizes speech in audio, it matches the received phrases with the text. Usually, matching algorithms are used for this, which find the most suitable variants of recognized phrases for each section of audio.

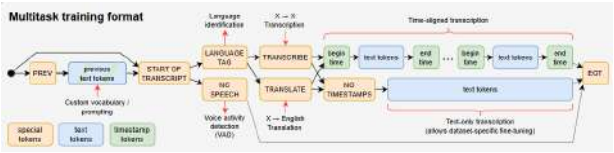

Figure 2. Whisper multitasking format [6]

Neural networks cannot always correctly recognize speech in audio, so Whisper uses various error-handling algorithms that allow you to correct recognition errors and improve the quality of transcription. The end result is a textual representation of audio that can be used for various tasks, such as content search and analysis, automatic transcriptions, and subtitle creation.

The amount of pre-training speech recognition data for a given language is very predictive of zero-shot performance on that language in Fleurs.

For small models, performance on English speech

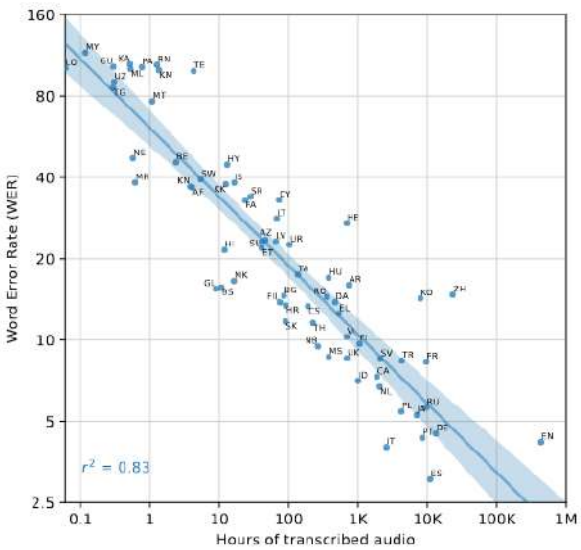

Figure 3. Correlation of pre-training supervision amount with downstream speech recognition performance [6]

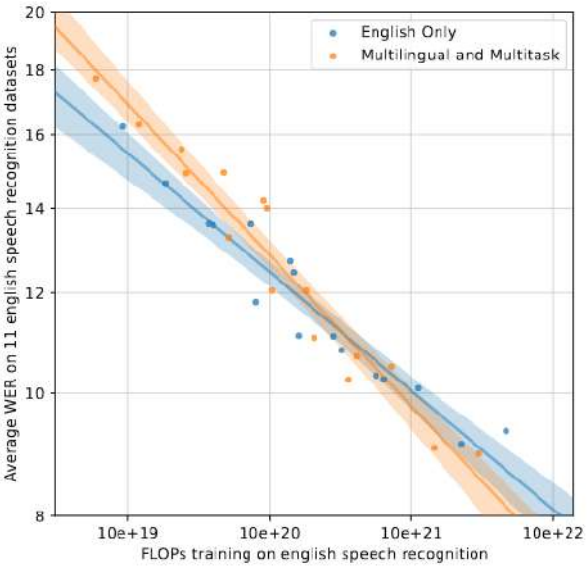

Figure 4. Multitask and multilingual transfer improves with scale [6]

recogni- tion degrades when trained jointly in a multitask and multilingual setup. However, multilingual and multitask models benefit more from scale and eventually outperform models trained on English data only. At the 4-th picture 95 percents bootstrap estimate confidence intervals are shown [6].

Another problem is creating an output from the existing text version of the meeting. This will help to create a meeting protocol.

Text Summarization is a natural language processing (NLP) task that involves condensing a lengthy text document into a shorter, more compact version while still

retaining the most important information and meaning. The goal is to produce a summary that accurately represents the content of the original text in a concise form.

During training, the neural network learns to identify important phrases, sentences, and ideas in the full-length text, and generate a shorter summary that accurately captures the main points.

There are different approaches to text summarization, including extractive methods that identify and extract important sentences or phrases from the text, and abstractive methods that generate new text based on the content of the original text.

One approach for text summarization is to use an encoder-decoder architecture with an attention mechanism. The encoder takes in the full-length text and encodes it into a fixed-length vector representation. The decoder then generates the summary based on this encoded representation. The attention mechanism allows the decoder to focus on the most important parts of the encoded representation while generating the summary.

A major limitation of this network is its incapacity to extract significant contextual connections from extended semantic sentences. When a lengthy text contains contextual relationships within its substrings, the basic seq2seq model cannot recognize those contexts. As a result, the model's performance is somewhat compromised, leading to reduced accuracy.

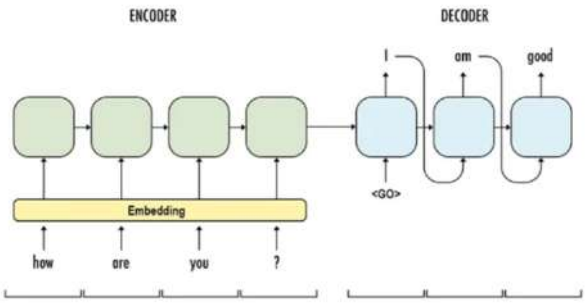

Figure 5. Encoder-decoder model [7]

The encoder network extracts or obtain features from input data and products internal state vector or context vectors, which summarizes the input sequence. In STM networks, these vectors are referred to as hidden state and cell state vectors. The decoder network interprets the context vector produced by the encoder and generates the output sequence. The final call of the encoder produces the context vector that serves as the input to the first call of the decoder network. The decoder then use this context vector along with the initial states to start generating the output sequence, and the outputs are considered for future predictions.

Attention is a type of enhancement to the current sequence-to-sequence network that addresses its limitations. It derives its name from its ability to highlight important parts of a sequence. By considering only a few relevant items in the input sequence, the output sequence becomes conditional and is guided by weighted constraints. These constraints are the contexts that receive attention, which are subsequently used for training and predicting the desired results.

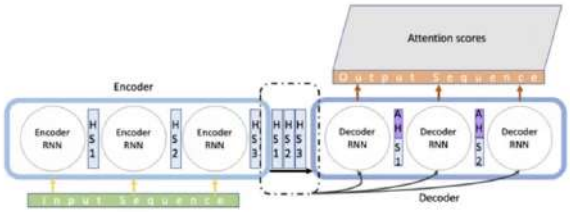

Figure 6. Attention architecture [7]

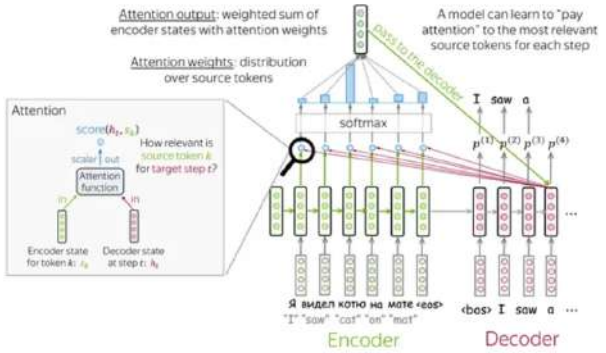

Figure 7. Attention model [7]

The attention-based sequence-to-sequence model requires considerable computational power, but its performance is superior to that of the traditional sequence-to-sequence model. Additionally, the model can demonstrate how attention is focused on the input sequence when predicting the output sequence. This is helpful in comprehending and identifying the specific input-output pairs and the extent to which the model considers them.

The model aims to create a context vector that is tailored for each output time step by selectively filtering the input sequence [7].

Another approach is to use reinforcement learning, where the neural network is trained to generate summaries that maximize a reward function based on the quality of the summary. This approach can be useful when there is not enough labeled data available for supervised learning [8].

The key to neural network text summarization is in finding a balance between generating a summary that is short and concise while still accurately conveying the main ideas of the full-length text.

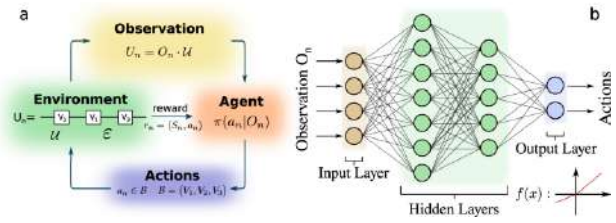

Figure 8. Rainforest learning model [8]

It doesn't matter which service to choose for summarization. But one of the most convenient services — ReText.AI. ReText.AI's summarization feature can help shorten text content while preserving its meaning. It can create shorter descriptions or summaries, simplify complex information for easier comprehension, and adapt the text to social networks, saving authors time and increasing productivity.

III. Proposed approach

The purpose of the work is to expand the functionality, increase the possibilities, ways of using the meeting protocol, find the best solution and optimize the process of finding errors and inaccuracies of the result.

One of the options for achieving the goals mentioned above is integration with OSTIS Technology. This way the meeting is presented in the knowledge base, its specification necessarily includes information about the participants, the start and end time of the meeting, the topic of the meeting, as well as an audio recording of the meeting is stored in the form of an ostis-system file.

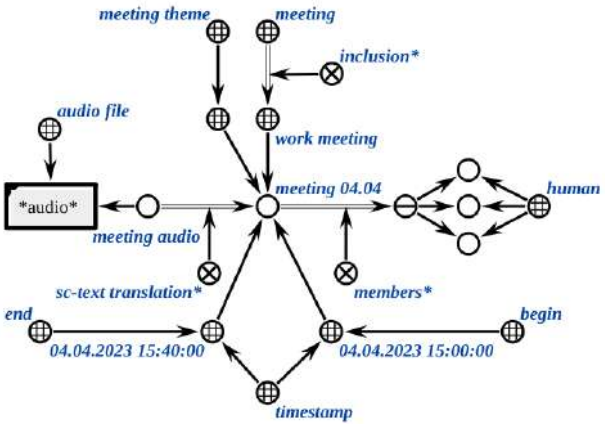

Figure 9. Text format of the meeting recording in SCg

To solve the problem under consideration, the OSTIS-system problem solver applied a strategy for decomposing the problem into subtasks. Further, problem solver analyzed that for summarization it is necessary to obtain a natural-language representation of the meeting in a text format and then use a summarization agent for it. Therefore, the speech-to-text translation agent is called.

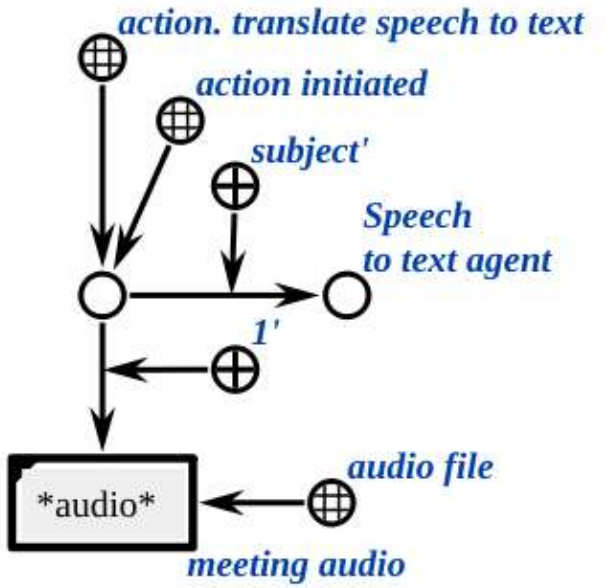

Figure 10. Speech-to-text translation agent input

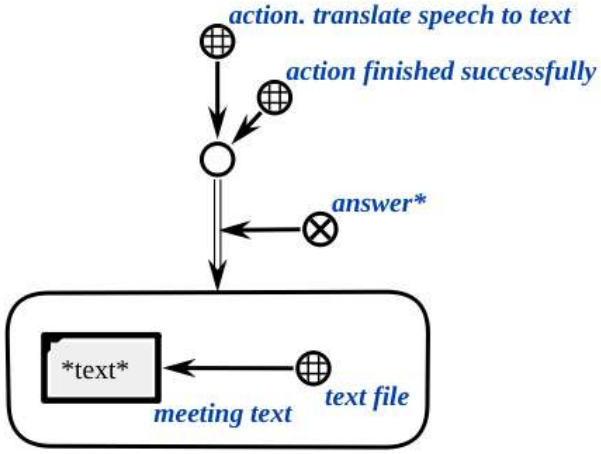

Figure 11. Speech-to-text translation agent output

The audio recording of the meeting is transcribed into a text format using the Whisper service. Next, a structure is formed in the knowledge base that connects the same meeting sign with its different external representations, that is, audio and text. Thus, from the audio format at the input, a text format at the output is obtained.

Thus, the task of transcription is solved. Further, to obtain the minutes of meetings, it is necessary to solve the summarization problem. To do this, the summarization agent is called. Summarization agent is the code that accesses the service and immerses its response into the knowledge base of the OSTIS system. The input for the service is also taken from the knowledge base, that is, the input is the result of solving the task of transcribing using the Whisper service and translating the audio recording

of the meeting into the text format.

The text received from the audio recording is fed to the input of the summarization agent. And the result of the agent's work is a text reduced to the specified number of words. Which, at the same time, retains its meaning.

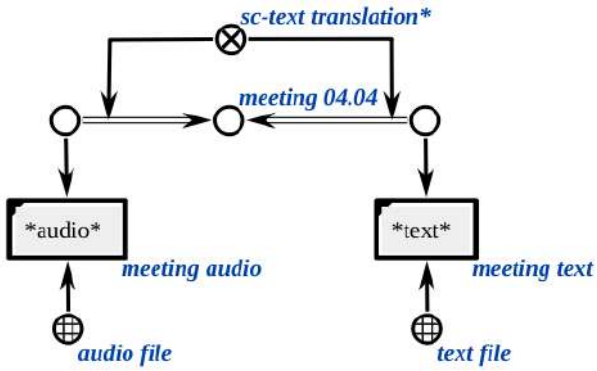

Figure 12. Speech-to-text translation agent

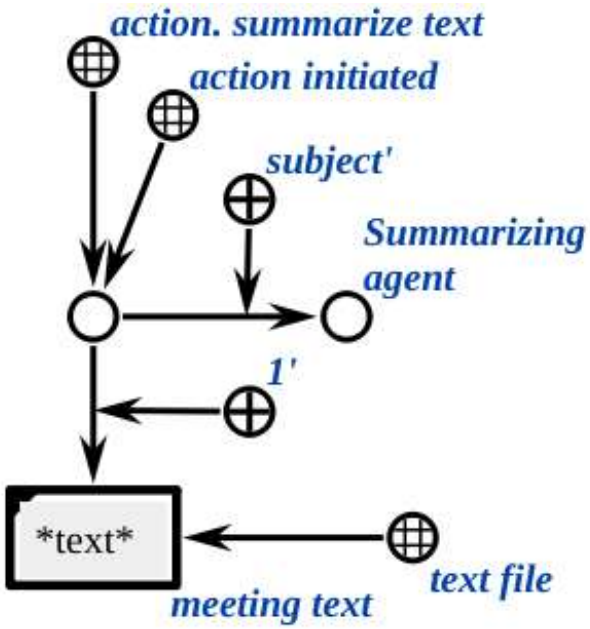

Figure 13. Summarization agent input

In fact, the above actions occur with knowledge base constructs, and text and sound are done only by NLP agents. NLP agents are natural language processing agents, computer programs that are designed to understand and respond to natural language input. These agents use a combination of techniques from computer science, linguistics, and artificial intelligence to analyze and interpret human language. NLP agents can be used in a wide range of applications, such as chatbots, virtual assistants, language translators, speech recognition systems, and sentiment analysis tools. These agents can help automate tasks, provide personalized recommendations, and improve the

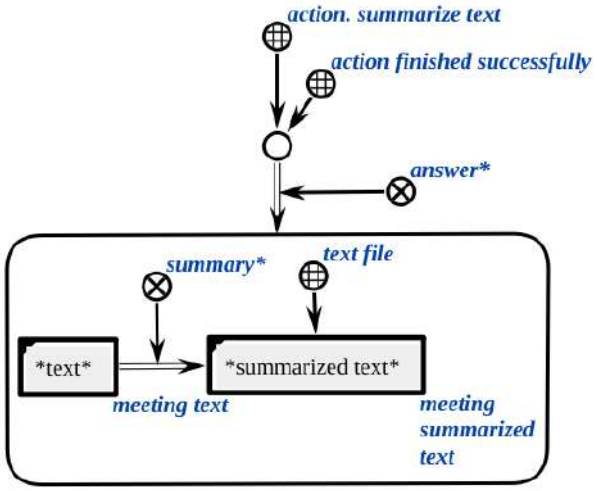

Figure 14. Summarization agent output

overall user experience by allowing humans to interact with technology in a more natural way.

NLP agents typically involve a combination of several components, including speech recognition, natural language understanding, natural language generation, and dialogue management. These components work together to enable the agent to understand and respond to human language input in a meaningful way.

Everything that is developed using OSTIS technology should be presented, first of all, in the form of knowledge base constructions, that is, described in SC-code. Including the meaning of the text and sound. And only then, in order to display the already processed information to the user, a natural language interface is used, that is, NLP. The text-to-speech translation agent is a classic NLP task, as is text-to-speech translation.

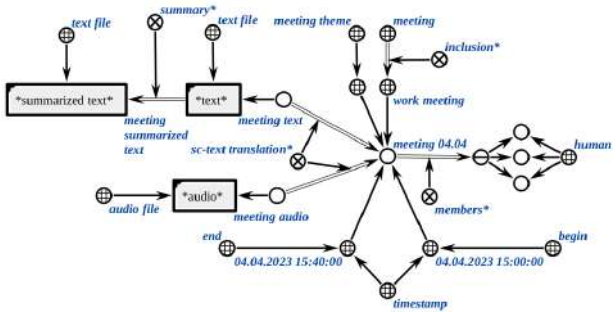

Figure 15. SCg-result of making a call protocol

OSTIS technology in the creation of a meeting protocol allows user to integrate various problem-solving models, achieve compatibility, effectively use knowledge bases to solve transcription and summarization tasks. OSTIS makes it possible not only to summarize the results of the meeting, but also to highlight the meaning of what was said during the meeting, process the context, analyze

the content, identify problem areas and contradictions, and offer solutions to them [9].

This will bring the use of the developed meeting protocol to a new level, because the data will be stored in a single OSTIS memory [10]. Thus, any OSTIS application will have access to them, will be able to use and explain. Such a system will be easy to integrate with other neural networks, expand the functionality by using and reusing existing developments.

IV. Conclusion

The paper proposes one of the solutions for keeping minutes of meetings based on solving the problems of transcription and summarization. The article describes the existing approaches to solving such problems, and also offers a solution for integrating existing approaches with OSTIS Technology, the advantages of such a solution are marked.

The results obtained will allow not only to get a brief conclusion from the meeting, but also to create an intelligent system based on the knowledge base. The ready-made system allows to highlight the meaning of what was said during the meeting, process the content, analyze the content, identify problem areas and contradictions, and offer solutions to them.

After the project is implemented, OSTIS allows you to reuse the received solutions in other tasks and projects. And also, integrate the system with other applications. As a result, all applications developed on the basis of OSTIS technology have access to a common knowledge base.

References

[1] V. Golenkov, N. Guliakina, and D. Shunkevich, *Otkrytaya tekhnologiya ontologicheskogo proektirovaniya, proizvodstva i ekspluatatsii semanticheski sovmestimykh gibridnykh intellektual'nykh komp'yuternykh sistem [Open technology of ontological design, production and operation of semantically compatible hybrid intelligent computer systems]*. Minsk: Bestprint [Bestprint], 2021, (In Russ.).

[2] (2021, Jun) Ostis applications. [Online]. Available: https://github.com/ostis-ai/sc-web

[3] O. Yara, A. Brazheyev, L. Golovko, L. Golovko, and V. Bashkatova, "Legal regulation of the use of artificial intelligence: Problems and development prospects," *European Journal of Sustainable Development*, vol. 10, p. 281, 02 2021.

[4] (2022, September) IMS.ostis Metasystem. [Online]. Available: https://ims.ostis.net

[5] (2015-2023) Whisper. [Online]. Available: https://openai.com/research/whisper

[6] (2015-2023) Robust speech recognition via large-scale weak supervision. [Online]. Available: https://cdn.openai.com/papers/whisper.pdf

[7] (2021, February) Understanding encoders decoders with attention based mechanism. [Online]. Available: https://medium.com/data-science-community-srm/understanding-encoders-decoders-with-attention-based-mechanism-c1eb7164c581

[8] (2021, August) Quantum compiling by deep reinforcement learning. [Online]. Available: https://www.nature.com/articles/s42005-021-00684-3

[9] (2021, Jun) Sc-machine. [Online]. Available: https://github.com/ostis-ai/sc-machine

[10] (2021, Jun) OSTIS. [Online]. Available: https://github.com/ostis-ai

Интеграция Технологии OSTIS со сторонними NLP сервисами

Черкас Е. О., Купо А. Н.

Данная статья посвящена интеграции Технологии OSTIS со сторонними сервисами, работающими на базе нейронных сетей. В качестве примера интеграции рассматривается задача ведения протокола совещаний, решение которой основано на транскрибировании и суммаризации. Транскрибация осуществляется с помощью сервиса Whisper, суммаризация — retext.ai.

Received 13.03.2023