References

1. Jiang T, Hou Y, Yang J. Literature Review on the Development of Visualization Studies (2012–2022). Engineering Proceedings. 2023; 38(1):89. https://doi.org/10.3390/engproc2023038089 .
2. Noptanit Chotisarn, Supaporn Lonapalawong, Tianye Zhang, Wei Chen A Systematic Literature Review of Modern Software Visualization : Journal of Visualization URL: https://ar5iv.labs.arxiv.org/html/2003.00643

PRACTICE IMPLEMENTATION OF NEURAL NETWORK MODEL BART-LARGE-CNN FOR TEXT ANNOTATION

Mazurets Oleksandr
Ph.D in Engineering Science, Associate Professor
exe.chong@gmail.com
Molchanova Maryna
Teacher
m.o.molchanova@gmail.com
Klimenko Valeriia
Teacher
ler.klimenko.8@gmail.com
Prosvitliuk Mykhailo
Bachelor student
prosvitlyuk@ukr.net
Computer Science Department
Khmelnytskyi National University, Ukraine

The task of annotating and abstracting textual information has been carried out since the emergence of the field of natural language processing. These are important tasks in the context of research in natural language processing and artificial intelligence. Natural language processing is one of the disciplines of information technology and deals with the analysis and synthesis of textual information that people use for communication [1].

Annotation is the process of analyzing and structuring information to create a short description or summary of a text document, such as a book, article, or scientific publication. The main goal of annotation is to provide a brief description of the document's content, revealing its main idea, key aspects, and structure, but not disclosing all the information [2].

Today, there are several approaches to automatic annotation that can be divided into two main groups: methods for compiling excerpts (extractive algorithms) and generating summaries (generative algorithms) [3].

Extractive algorithms generate an annotation using text fragments of the input document. They select the text blocks with the highest lexical and statistical significance and combine them into a composite annotation. This approach is easy to

implement, does not require large computing resources, but may not provide sufficient quality due to the lack of semantic analysis of the text.

Generating algorithms analyze the input document to find information that is used to generate the annotation text. They are able to take into account semantic relations in the text, avoid duplication of information between the main text and the annotation, and ensure the completeness of the annotation. This approach requires more computing resources, but can provide higher quality annotation.

The aim of the work is to simplify the creation of annotations of artistic works by automating annotation using machine learning tools. To achieve this goal, it is necessary to create an appropriate software implementation in the form of a windowed application designed for automated annotation of artistic works based on user text.

The method of annotating works of fiction using machine learning is designed to automate the creation of annotations, which in turn simplifies the process of describing the content, key themes, characters, and other aspects of a work. The method works by converting input data in the form of text to annotate a work of art, a trained machine learning model, and desired annotation parameters into output data in the form of an annotation and a numerical assessment of the quality of the annotation.

The first step of the method is to load the trained machine learning model, which is the input to the method. The model in the basic format is in English.

The next step is the automated translation of the text for annotation into English, after which intermediate data is generated in the form of a work of fiction in English.

The next step is to use the trained machine learning model to annotate the text with the specified parameters. The parameters are the desired annotation sizes (minimum and maximum). Another feature of this step is the algorithm for splitting the text into parts, since the trained model has a limit of 1024 tokens.

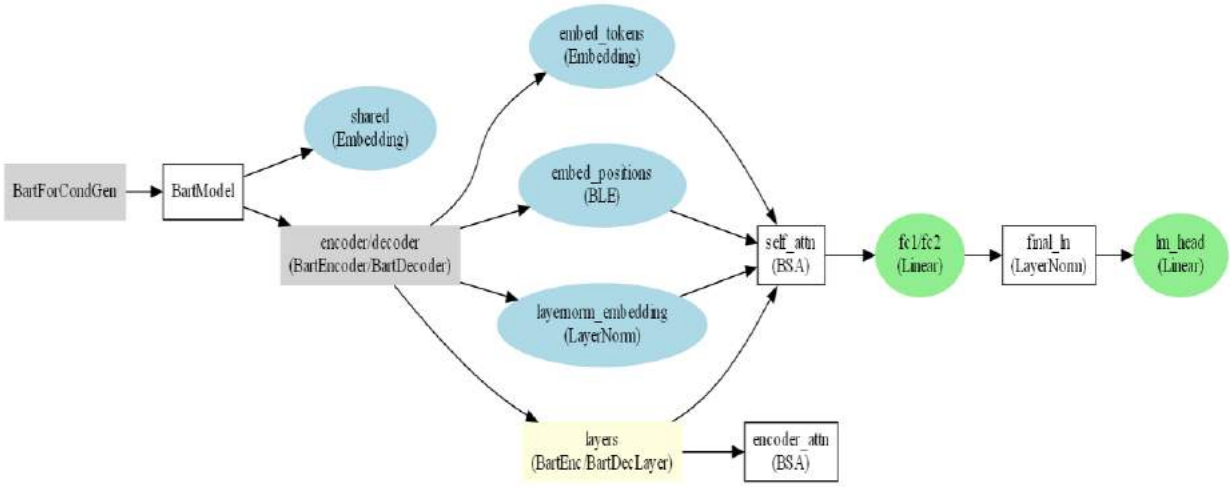

Figure 1. Architecture of the neural network model of machine learning “BART-Large-CNN” [4].

Therefore, we take the entire text and divide it into sentence fragments of no more than 1024 (up to the point). Depending on the number of parts obtained, each part is subject to restrictions on the size of the annotation. The resulting annotation is a complex one, consisting of parts presented in English.

The next step is to translate the annotation from English into Ukrainian. The translation is performed in the same way as from Ukrainian to English.

Figure 2. Detailed architecture of the neural network machine learning model «BART-Large-CNN» for text annotation.

The last step is to evaluate the annotation according to the parameters of the Flash readability index, Flash Kinkaid class level, error rate, and ROUGE metrics.

The output of the proposed method is the obtained annotation, which is close to the specified parameters, as well as numerical estimates of the annotation quality.

The input to the method of annotating works of art is a trained machine learning model. Therefore, we will use the neural network machine learning model «Bart-large-cnn» [4], the schematic architecture of which is shown in Figure 1.

BART is a transducer-encoder (seq2seq) model with a bidirectional (BERT-like) encoder and an autoregressive (GPT-like) decoder. The BART is pre-trained by distorting the text with an arbitrary noise function and training the model to reconstruct the original text [4].

This neural network model is particularly effective when configured for text generation, but also works well for comprehension tasks (e.g., text classification, answering questions). This machine learning model was trained on CNN Daily Mail, a large collection of text-summary pairs. The main components of machine learning model layers. The shared (Embedding) layer is used to encode and decode the input text and represent words in the vector space.

The encoder (BartEncoder) layer accepts the input text and performs its encoding. It includes layers for processing the input text and generating its internal representation.

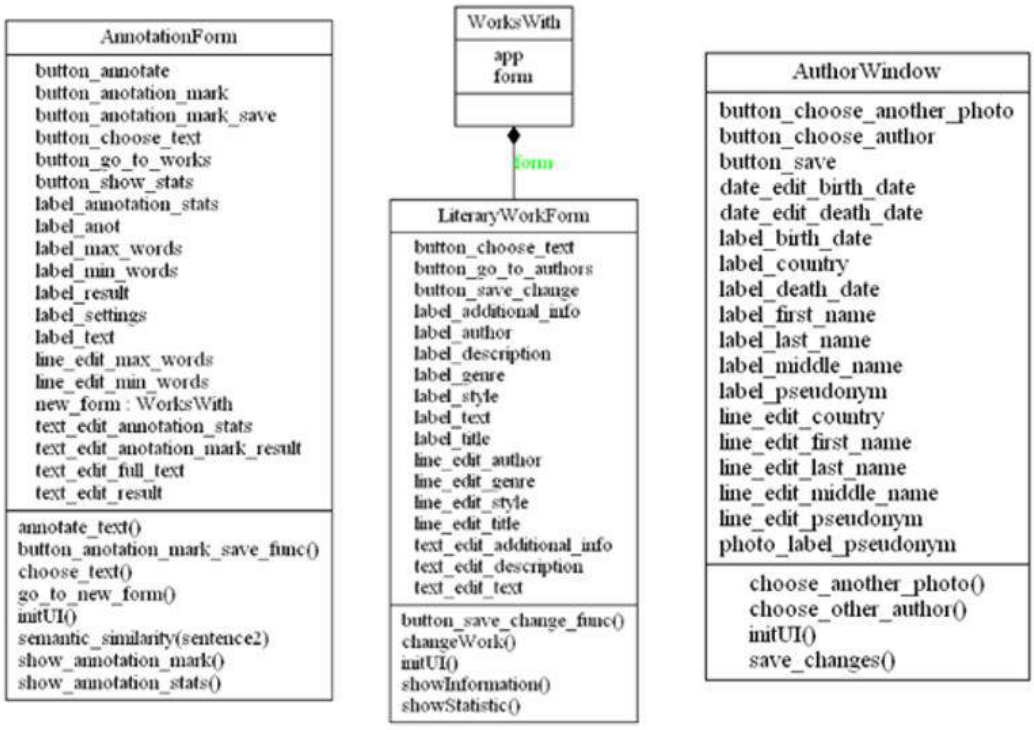

Figure 3. Class diagram of the software system for network machine learning model «BART-Large-CNN» for text annotation.

A layer of the decoder type (BartDecoder) uses the internal representation of the input text received from the encoder to generate the output text. It includes layers for generating text taking into account the context and the previous part of the generated text.

The layer of type lm_head (Linear) is the last layer of the model, which is used to predict the next token in the text. The size of the output space of this layer corresponds to the number of unique tokens in the input text.

The full architecture of the used neural network is shown in Figure 2. Each of the encoders and decoders has 12 layers, which include attention mechanisms, linear transformations, and normalization. Attention mechanisms allow the model to focus on important parts of the input text during encoding and decoding. Each layer also uses the GELU activation function for nonlinearity. The embedding size and internal dimension of the text representations in this model is 1024.

The following set of tools will be used to develop the software: PyCharm programming environment, Python programming language, SQLite DBMS, and SQL query language. The structure of the program components of the system in the form of a class diagram is shown in Figure 3.

The «AnnotationForm» class is the main class of the software application that implements the functionality of 2 subsystems: the subsystem for intelligent generation of artistic annotation and the subsystem for assessing quality of annotations of artistic works. Result of annotation generation is shown in Figure 4.

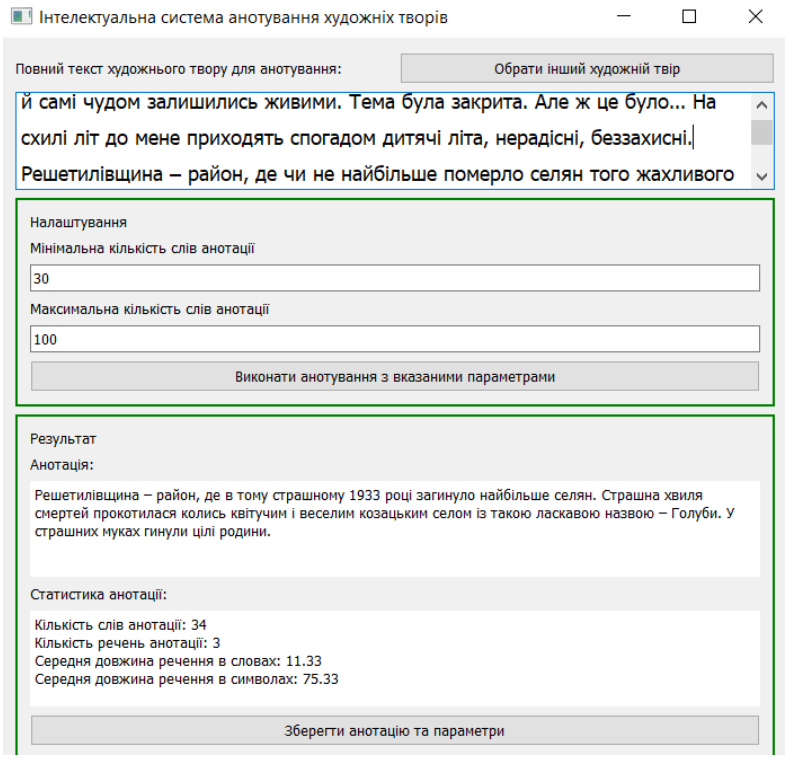

Figure 4. Example of text annotation generation by neural network model «BART-Large-CNN».

The initUI() method initializes the user interface, adds all elements to the layout, and sets their properties. The annotate_text() method annotates text using the BART model. First, it translates the input text into English, creates an annotation, and then translates the result back to Ukrainian.

The show_annotation_stats() method calculates and displays statistics for the generated annotation, including the number of words and sentences, and the average sentence length. The choose_text() method allows you to select another text for annotation. The show_annotation_mark() method evaluates the quality of the

annotation using the Flash-Kinkaid readability indices and the number of grammatical errors using language_tool_python. It also calculates the ROUGE metric to evaluate the correspondence of the annotation to the source text. The semantic_similarity() method calculates the semantic similarity between two sentences using WordNet.

In turn, the classes «AuthorWindow» and «LiteraryWorkForm» implement the functionality of the subsystem «Digital Library of Works of Fiction». Their functionality is designed to work with authors and works of art.

The initUI(self) method initializes the user interface, creating all the necessary widgets and layouts. It sets the window title, adds buttons, text fields, labels, and date selection fields. Connects buttons to the appropriate methods to handle click events. The choose_another_author(self) method handles the click of the Choose Another Author button. The choose_another_photo(self) method handles the click of the «Choose another photo» button. The save_changes(self) method handles the click of «Save changes» button. It collects data from input fields (nickname, surname, first name, patronymic, country, date of birth, date of death) and writes them to database.

Thus, the scheme of the method of annotating works of art was described, which works by converting input data in the form of text for annotating a work of art, a trained machine learning model, and desired annotation parameters into output data in the form of an annotation and a numerical evaluation of the quality of the annotation and is intended for automated annotation creation. We also present the neural network architecture of the machine learning model, which is the input to the proposed method of annotating works of art. This neural network model belongs to the «transformers» type and is currently one of the most powerful text generation models. A practical implementation of the method of annotating works of art has been created and the main purposes of the software components of the intelligent system for annotating works of art have been described.

References

1. Rules and methods of annotation, examples. URL: https://presa.com.ua/navchannia/anotuvannya-tse-shcho-take-pravila-ta-metodiki-prikladi.html
2. Overview of automatic text annotation methods. URL: https://core.ac.uk/reader/52160900
3. Zalutska O., Molchanova M., Sobko O., Mazurets O., Pasichnyk O., Barmak O., Krak I. Method for Sentiment Analysis of Ukrainian-Language Reviews in E-Commerce Using RoBERTa Neural Network. CEUR Workshop Proceedings, 2023, vol. 3387, pp. 344-356.
4. BART Model Architecture. URL: https://medium.com/@nadirapovey/bart-model-architecture-8ac1cea0e877