RESEARCHES ON DEVELOPING AN ALTERNATIVE METHOD FOR MULTI DOCUMENT SUMMARIZATION IN TURKISH

1. RESEARCHERS AND INSTITUTION INFORMATION

Emre SENGİR

Izmir Institute of Technology,

Electronics and Communications Dep.,

Student.

emresengir@std.iyte.edu.tr

Buket ERŞAHİN

Izmir Institute of Technology,

Computer Engineering Dep.,

Instructor.

buketoksuzoglu@iyte.edu.tr

ABSTRACT

This study is about researching and proposing alternative and simplifying methods on Multi-Document Summarization in Turkish. We intended to learn, research and test new methods in the given time of the summer internship of the student. During the time we had, we firstly study on the field of machine learning with courses and articles about given topic, after gathering fundamental knowledge then we started to search for a field that is fit and usable by the intends of study. We gathered the necessary information on a dataset, determined the main steps to obtain our goal and started to implement it in as best as we can. Meanwhile we tested different operations in different steps to increase efficiency. During our researches we learned many things that is contributing to the field and us, but unable to perfectly functionalize it in the given time, so this paper shares the knowledge all we gather during the researches.

2. ABOUT STUDY

Matter of Study: This research paper is about a current natural language processing application, Multi-Document Summarization in title, which is one of the applications of machine learning. In detail we tried to develop an MDS model for Turkish language in order to contribute the lack of up-to-date information about this topic.

Scope and Purpose of Research: This study is planned to develop a fully functional model in the given time as much as possible. In order to develop this model we benefited the previous researches and tools that they had used, but as we pointed out, we also tried to implement a model that updates the previous studies in an improving and simplified manner, for example chromadb is one of the tools that we have been used for database management with intend to manipulate contents and simplify some operations that will be explained in further parts.

RESEARCHES ON DEVELOPING AN ALTERNATIVE METHOD FOR MULTI DOCUMENT SUMMARIZATION IN TURKISH

3. INTRODUCTION

As many formerly written researches declares, main reason of this research is to propose, or guide, to a model that can efficiently performs text summarization, specifically in Turkish. We do not want to mutter about the growing written information in modern world and subjective need for summarization of those informations to reach out the main idea in short explanations. Instead we want to directly express our purposes and solution ideas on research.

First thing to say about research of ours is that we wish to update and contribute to literature on Turkish text summarization which lacks on the field with having mostly old written, few and can be improved by the developments with new machine learning methods. For this intend, in order to enlarge the area of study we decided to work on both two basis methods that is extractive and abstractive. In a nutshell, these two terms can be explained as such; as the word tells extractive method extracts the important words, terms and likely the sentences in general by scoring them with respect to some word based mathematical functions that is applied to the text, and abstractive method is the innovator model, that in a similar term to us humans paraphrases the given text with respect to the learning model, mostly neural networks.

Before diving into the details of research, we thought that some knowledge about the Turkish language and it's effect on developing a method in comparison to the similar researches on languages like English is necessary. A research we encountered during our learning phase was about examining effect of an important characteristic of Turkish language on Multi-Document Summarization which is that Turkish language is an agglutinative language. It tells that a root word having the possibility to have many meanings by taking simple and meaningless suffixes and this can continue up to the numbers like three times suffixed, this results in word corpuses drew out from the texts reaching huge amount of numbers. This problem needs a solution but before the solution it also creates a dilemma that also needs to be pointed out. The dilemma is that if we take this number of words directly to our corpus it will be result in computational costs during processing data and difficulties in applying learning models, in the other hand if we take words as root only, then this will result in losing the meaning that has passed from the text, also making harder to understand the summarized text that is created out of these simplified words. Researches concluded that prefixing the words in a certain number of letters is the most efficient way but in our research we decided to use stemming and taking the roots of words [1], this decision has been made due to the timing issues of the research.

By stemming roots of words out of text creates our corpus, for this corpus we determined four main steps where first step is the source of corpus that is gathering the dataset. Following steps are thought in order to draw related documents out of the dataset to classify and generate. Before defining these steps, dataset we gather has the content obtained from Turkish Airlines frequently asked questions [6]. To gather a dataset web-scrapping is an useful tool but some safety protocols may blockade it like we encountered on THY website so we forced to draw it manually. Manually drawing dataset resulted in a relatively smaller one but according to previous studies' dataset sizes we created a three column and one hundred forty nine row dataset that is fit for purpose.

Columns are labelled as question, topic and answer. After gathering the dataset and processing necessary text; stemming, lowering, cleaning stop-words and tokenizing etc., we determine following three steps which are the first, embedding with chromadb, querying and obtaining related questions to the input, first step also does the work of classifying, instead developing a classifier model with machine learning, we benefited chromadb's attributes that is very simple to classify related texts (questions). Second step is extracting the related questions' answers and third step is to train a recurrent neural network model to generate a summarized answer to the user inputted prompt in abstractive manner.

Our roadmap with introduction of research overall as explained as above. Rest of the paper flows as given, fourth part details the related works that is used on or influenced our study, fifth part explains the experimental setup, tools we used during the research, sixth part on evaluation and discussion, and at last we concluded the paper.

4. RELATED WORKS

There are many studies for text summarization in general and a few for Turkish application. One of them is the one we referred in previous section that analyzes the effect of word's structure to the performance of sum [1]. Although this referred research uses hybrid model, it only shares information on word structure's effect, so for a detailed research we looked up for other extractive and abstractive researches. In an extractive model used research, researchers develop a structure that is similar to a neural network in fundamental components terms but there was no learning just the neuron (scoring functions) and weights that is adjusted to increase performance on determining the related sentences [2]. Some of the functions that is widely used in extractive approaches is term frequency and inverse-document frequency which are the functions we used in our extraction step. There are more complex and more useful functions for sentence scoring, like sentence position and centrality but because of timing issues we are only be able to apply tf-idf for our extractive step.

For abstractive summarization we are unable to find a fully completed work with recurrent neural networks in Turkish. So in order to gain an insight we examined studies on other languages. One of the content describes an RNN model, even though the model does not fit properly to our desired one, that uses sequential learning and uses book of Platon's Republic as source text. Writer takes the book as text and does the necessary text processing, after that he determines some properties to divide the text in many sentences that creates sequences which moves one word at a time to the end of the book sentence by sentence (x+1)[1]. By taking the last word of these sentences' (+1) and after the necessary padding, encoding and categorization writer sets the neural network, encoded sentences as input (x) and last words of sentences that is categorized as outputs (+1) resulting in a sequential learning model that operates with some parameters, seed text or the question as determining parameter [3].

[1] :x stands for the number of words in the sentence and +1 is the last word of the sentence total in x+1

5. EXPERIMENTAL SETUP

Dataset: Due to the short time interval of the research, in contrast to the dataset content that is used to train abstractive deep learning models in [1] and [2] we are unable to generate human summaries in order to create a supervised learning model for training main texts with human generated summaries to compare the results. Instead we came up with the simple implementation of sequential learning referred as [3], as we explained in related works part.

So by cutting the needed data, we gathered our dataset forming from three columns; topic, question, answer. We used THY FAQ as one and the main source of the dataset and did some manipulation in order to create some variance in the dataset that may effect the efficiency by using ChatGPT. Since the issues with web-scraping on THY website we manually draw the data to an .csv file for easy reading while processing it.

Stemming: As we pointed out previously, by the nature of Turkish language study [1] gives us an insight on the stemming policies to determine the most fit one in many. In order to stem the words while text processing we benefited from the python library for Turkish language processing VNLP [4] that has many useful features for future improvements. For our study we manipulated stemming function simply and obtained root versions of the words that is processed sentence by sentence to a list of words of the sentence, resulting in 2D array that formed up from list of sentences consists the stemmed words, meanwhile function handled lowering and removing stop-words in Turkish.

Classification: From the question column of our dataset we processed question texts and embedded them by using chromadb library. While working on this embedding and classification part we also used topic column of the dataset to assign 'metadatas' on questions which resulted in increasing efficiency on determining the related questions for the given query. In detail, during the embedding, we used sentence transformer model of [5], for proper embedding in Turkish, this is also an efficiency increasing factor meanwhile classification.

We performed classification by using sentence transformer model to embed the query, with attribute of chromadb we easily find n related questions by taking their distances to the query. Another parameter of the embedded questions besides 'metadatas' is the 'ids', first we assigned these ids by numbering them one by one during embedding then after classification we collect the ids for finding the indices of the related questions to draw the answers out of answers column to gather it into one text for extracting purposes, Thus we created the multi document file causally under one string.

Extraction: As the first step of forming the summarization for the given query, with the text that consist n related questions' answers to this query we performed extraction operation to the text by following three main steps after applying stemming and removal of stop-words to this text. As we said, in many sentence scoring functions we decided to apply tf-idf scoring that works for word based scoring, so first two steps is constructing scoring table of words with respect to term frequency and inverse document frequency on sentences of text. With those look-up tables (matrices) we scored the sentences and took the averages of the sentences as threshold which divides the number of sentences in half and gathering the result sum in another single string for feeding RNN.

Abstraction: Unfortunately we couldn't obtain an observable result on seeding query text to RNN because of some issues that will be shared in the discussion part. Slightly, the operation that RNN performs is as we explained in the related work part and in detail at the book [3], by creating sequences of size n similar to the sentences but differs word by word, we train the RNN to guess the which next word could come in the sequence with respect to the seed text, and by creating a simple closed loop (or recursive function we may call) we can continuously seed the function and generate a sequence of words that is fit to the seed text and the continuation words.

6. EVALUATION AND DISCUSSION

Except abstraction we performed all the operations almost satisfactorily but in a decreasing efficiency trend because of issues encountered between steps. Starting from the dataset part, configuration of dataset might be different as in the study [2], as if we had the necessary time and help on generating human summarization or etc., having limiting factors during the research like that cut us improving previous model and performing our work.

Effect of stemming, simplification and cleaning of text data on performance is confirmed in study [1] and with the insight we gained our approaches during categorization and classification with factors like 'metadatas' while working with chromadb, showed us that as we apply these deterministic factors on text data we obtain greater relatively logical relate on query in perspective of a human. In the short time we worked with chromadb usages we learned are very impressive and can be used to improve and develop current and future models on MDS or any type of natural language processing projects. We performed classification on questions since we thought that complexity level of an answer text is greater than a question, but different approaches may presents assertive behaviors in the future studies. In addition, we also examined something that could be useful for future studies, which is embedding a huge dictionary of words. We said that during classification chromadb uses distances of sentences to find related ones, but during this operation tool only uses words that comes with the embedded data which is relatively small in comparison to a huge size of dictionary that could be embedded previously to close the gaps between words in terms of distance, also probable to result in closing the gap between sentences.

Main reason in using a hybrid model for our research is to widen up the area of study parallel to the learning goals. Although we are unable to functionalize and conclude our study completely we reached satisfying results during the testing of steps separately and learned a lot. But one of the thing that needs to be pointed out that our answers content of dataset is probably asks no need for extraction method to shorten the text to create a trainee text for RNN model. In fact we observed a great negative effect during extraction. It is that between the answers, there are common sentences that forwards users to a source on the related topic and they use similar phrases that increases the frequency of terms resulting in repeating sentences in extracted sum, draws the meaning of sum to nowhere.

In separate analysis of steps, abstractive approach we implemented is likely to be most basis sequential learning model that uses large amount of unorganized data, in comparison to the study [2], that operates on word based learning. It might be useful for small studies like ours but if one asks for a greater consistency in result, this is not the model to look for.

7. CONCLUSION

In conclusion, our research had the primary intend to gain insights about the topic and develop a model to update and simplify the previous studies. In the end we analyzed effects of text processing steps on querying and classifying text data, explained our attempts on using chromadb for Turkish language processing model. Argued on the models' requirements in order to be fit for evaluating dataset and produce a healthy result that can be processed between steps. We intend to continue to work on and improve the model we developed so far, meanwhile contributing and benefiting the studies that has been presented so far about the field.

RESEARCHES ON DEVELOPING AN ALTERNATIVE METHOD FOR MULTI DOCUMENT SUMMARIZATION IN TURKISH

REFERENCES:

[1]: Nuzumlalı, Muhammed & Ozgur, Arzucan. (2014). Analyzing Stemming Approaches for Turkish Multi-Document Summarization. 10.3115/v1/D14-1077.

[2]: Ertam, F., & Aydin, G. (2021). Abstractive text summarization using deep learning with a new Turkish summarization benchmark dataset. *Concurrency and Computation: Practice and Experience, 34*.

[3]: Brownlee, Jason, 2017, "Chapter 20: Project: Develop a Neural Language Model for Text Generation", Deep Learning for Natural Language Processing: Language Modelling, Sarah Martin, pg. 226-245.

[4]: https://vnlp.readthedocs.io/en/latest/

[5]: https://huggingface.co/emrecan/bert-base-turkish-cased-mean-nli-stsb-tr#citing--authors

[6]: https://www.turkishairlines.com/tr-int/bilgi-edin/index.html