RESEARCH ARTICLE | SEPTEMBER 05 2023

Text summarization and translation on multimodal data

Y. Krishna Bhargavi; P. Srinivas; V. Vamshi Krishna; P. Suvarna Rao; N. Upendhar

AIP Conference Proceedings 2754, 110005 (2023)
https://doi.org/10.1063/5.0161374

Text Summarization and Translation on Multimodal Data

Y. Krishna Bhargavi[1, a)], P. Srinivas[1, b)], V. Vamshi Krishna[1, c)],
P. Suvarna Rao[1, d)], N. Upendhar[1, e)]

[1]*Department of CSE, GRIET, Hyderabad, Telangana, India.*

Corresponding author: [a)] *kittu.bhargavi@gmail.com*
[b)] *porandlasrinivas704@gmail.com*
[c)] *vanagantivamshi123@gmail.com*
[d)] *suvarnarao66@gmail.com*
[e)] *upendharnemmani@gmail.com*

Abstract. In recent times, there is an extensive growth of multi model data holders like images, pdfs and documents holding a large amount of data due to rapid increase in usage of internet, social media and other information providing technologies. Manually extracting relevant information and discarding redundant information from the large sets of data is a prolonged work and enormous work. The automation of multi-modal data summarization is an important and necessary task. The rapid advancement in the deep learning techniques as neural networks shows the strength of Deep learning in text summarization. This paper proposes a neural-based abstractive multi-modal data summarization based on pre-trained modal approach in Transfer Learning. Existing models like TFRSP, RNN unable to get accurate results and consumes more GPU resources. This method first extracts the text from the documents, images and web-URL and encodes the extracted text from document, image and web-URLs with positional encoding and abstract the summary probability of sentence with unified Text-to-Text Transformer approach in transfer learning. This experiment is done by fine tuning the trained model on Colossal Clean Crawled Corpus(C4) dataset. Experiments show that our method outperforms the novel neural network architecture for language understanding methods for summarization and then translates to the local languages which can be saved as future references.

Keywords: Summarization, Abstractive Summarization, Transfer Learning, Neural Networks, Multimodal Data Summarization, Document Summarization, Text Extraction, ROUGE.

INTRODUCTION

With the increase in the extensive growth of multi-modal data, such as images and documents such as news, blogs on the internet, the need for text summarization [1] is becoming a predominant method for people to easily obtain information. The text summarization techniques are categorized into Abstractive text summarization [11] and Extractive text summarization [8],[9],[10]. The authentic text summarization [2] is different from multi-modal data summarization [3]. Most of the studies lighten text summarization which generates the summaries of document textual data. Image summarization [5] is another research direction of summarization that outputs the summaries of input image sets. Multimodal summarization outputs the multimodal summaries [3] of multimodal data. This paper summarizes the documents with text, Images by extracting the text from the images [5], and web URLs by getting the web page textual data from those particular websites [4]. To build an abstractive multi-modal [6] summary from documents, images and web URLs one should have a note to these problems.

1. How the abstractive summarization works? Does it improve the text summary? How do you score the output?
2. How to extract the text from the images?
3. How to extract the text from the URLs?

This paper proposes a Transfer learning based abstractive text summarization [7] to solve the above questions and integrate them. The basic working style of this model is as follows; initially the input type is categorized based on the text modal as documents, Images, web URLs [4]. Then the extraction of text is done that follows the abstractive summarization using weighted averages by self-attention methods [12] through the trained model data over the news summary data. We had used a base T5 Transfer learning model by training a news

International Conference on Materials, Processing & Characterization
AIP Conf. Proc. 2754, 110005-1–110005-7; https://doi.org/10.1063/5.0161374
Published by AIP Publishing. 978-0-7354-4634-2/$30.00

summary dataset as the training input and the manually created text summary as target output for textual images, text documents and URLs. This paper proposes a neural-based abstractive multi-modal data summarization based on a pre-trained model approach, which calculates the abstractive summary probability of the sentences. The results have shown that our model outperforms the novel neural network architecture for language understanding methods.

RELATED WORK

Ekaterina Zolotareva [13] proposed an “Abstractive text summarization using transfer learning”, in his work the text problem has been explored using Sequence-to-Sequence recurrent neural networks and Transfer learning with a unified Text-to-Text Transformer approaches. In this experiment, a T5 is used, which corresponds to the Text-to-Text Transfer Transformer. This transformer-based sequence to sequence model trains the architecture in a unified objective manner, the key difference in this experiment corresponding to other architectures is the “mask” used by various attention mechanisms of the model.

Manisha and Manoj Gorai [14] proposed a Layout and Text extraction method from document image using Neural Networks. This experiment provides an effective method for recognition of text in the document image with respect to the layout provided. In this experiment, The layout structure forms the basis for uprooting the information from the document image. This study of the layout requires the basic understanding of the positioning of the subparts in the document image. The initial phase of the task begins with the processing of the document image by binarizing the image in-order to eliminate the noise and coloured components. This process is later followed by segmentation, the task of cropping the image into smaller modules and then recursively processing each of the images. Further the sentences in the document image could be segregated into words and later into alphabets by using the logic of spaces between the words. This proposed method is based out of the concept of Convolutional Neural Network and Python is used as the primary coding environment as it has multiple inbuilt libraries to support for image processing and implementation of CNN. Based on the dataset of alphanumerical alphabets fed to the model, CNN algorithm classifies the recognised alphabets according to the uprooted features. In the end, Letters identified using the CNN algorithm are converted back to words and subsequently back to sentences. To conclude, the resulted output from this stage is then stored in a structured table format, which is finally converted to excel sheets. This Proposed model would give governing results when used for improving the functionality of the automation tool in a paperless environment.

Zepeng, Bin Xue [15] proposed an “Abstractive summarization model with a feature-enhanced seq2seq structure”. In this experiment, they raised the traditional seq2seq structure limitation. The traditional seq2seq has minimal ability to capture global features and long-term features caused due to range of attention mechanism and structure using RNN which resulted in a lack of information in the generated summary. In this experiment, they proposed a feature enhanced seq2seq structure for abstractive summarization which overcomes the limitations of the traditional Seq2Seq model. The proposed model utilizes the non-local network and the memory network to reform the encoder and decoder in the traditional seq2seq structure. This experiment improved 6% for R-L metrics for the dataset they had used resulting in the improvement of model performance with a higher quality summary.

Tamir Hassan and Robert Baumgartner [16] proposed “Intelligent Text Extraction from PDF Documents”. In this experiment, they used segmentation or layout analysis to understand the document. They break down the pages recursively into blocks. These blocks are termed as the atomic smallest logical entity in the document structure. This process is repeated for all the regions until no potential division with 100% white space is found. The major advantage of this approach is its hierarchical structure obtained which can be easily represented as the logical structure of the page. This has been achieved by combining levels where the division or cut is made previously. By combining both the bottom-up and top-down approaches the results can be further improved.

Ram kumar P [17] proposed “Text summarization using text frequency ranking prediction”. In this experiment, they combined both supervised and unsupervised algorithms. They used Term Frequency – Inverse Document Frequency – Text Rank (TF-IDF-TR) as a supervised learning algorithm and Seq2Seq (Sequence to Sequence) mode as a supervised learning algorithm. They combined both supervised and unsupervised learning algorithms to achieve the advantages of both abstractive summarization and extractive summarization. Text Frequency Ranking Sentence Prediction (TFRSP) algorithm proposed in this experiment deals with both abstractive and extractive summarization. Seq2Seq model used for abstractive summarization is a supervised learning algorithm that involves both training and testing. These experimental results are compared with existing methods' ROUGE values and result in an increase in the accuracy of the summary.

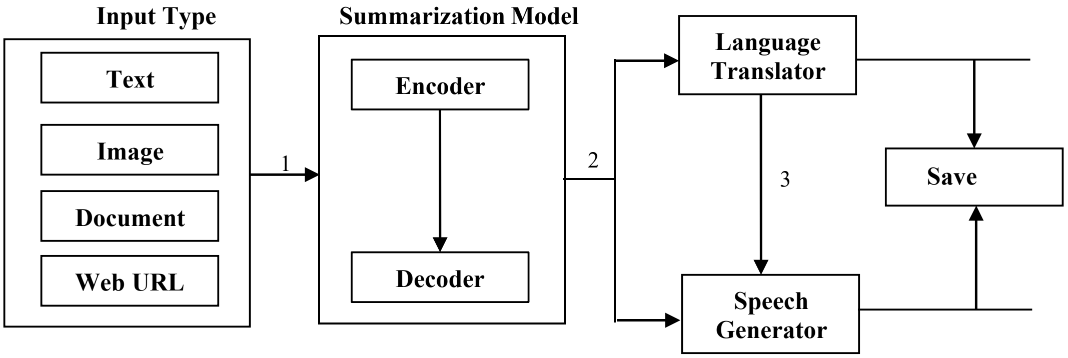
Thivaharan S [18] proposed “A Survey on Python Libraries used for Social Media Content Scraping”. This paper provides information about three web scraping libraries namely LXml, RegEx, and Beautiful Soup. This paper contained a detailed analysis of these libraries with the evaluation metrics such as their minimal gap in classification, minimum response time, and processing cycle consumption. These three libraries are subjected to a dataset containing 15,206 words for analysis. The dataset is flexed from worst case to best case complex patterns through average case complex patterns to ensure the durability of these scrapping libraries in all circumstances. The time required for each case is measured through a python time-lapse method named time Clock(). This experiment compared LXml, RegEx, and Beautiful Soup library's results through the mentioned metrics and RegEx resulted in better results in web scrapping and pattern matching.

PROPOSED APPROACH

This experiment uses Text-to-Text Transfer Transformer (T5) a Transfer Learning model which is a transformer-based sequence to sequence model that trains the architecture in a unified objective manner and models every problem in a text-to-text format which means the input would be in the text format and the output would also be in the text format. Unlike the other methods, In T5, the input sequence can be passed in parallel by considering, that the current word has dependency over the previous words in hidden state. Word embeddings are obtained one word at a time with the transformer encoder, on the other hand there is no concept of time stamp for input that passes all the words simultaneously, determine the word embedding simultaneously and perform various task using T5 such as summarization, translation which helps in doing discriminative tasks such as classifications. The idea of transfer learning is, to pre-train the model with enormous high-quality data on some self-supervised objective and then followed by a fine-tuning procedure of the same mode on certain downstream tasks. A key differentiator for other architectures compared to T5 is "mask". T5 uses it in various attention mechanisms. Remember that Transformer's self-attention operation takes a sequence as input and outputs a new sequence [14]. Each item in the output sequence is created by calculating the weighted average of the items in the input sequence. The weighted average is calculated as (1)

$$wt = \frac{frequency\ \text{of}\ the\ term}{Total\ no.\ \text{of}\ terms \in the\ document} \qquad (1)$$

Here in this experiment our model is pretrained on Colossal Clean Crawled Corpus(C4) dataset. The given figure (Fig – 1) represents the system architecture.



1 Extracted Text
2 Summarized Text
3 Translated Text

FIGURE 1. System Architecture

The system is developed to receive multi-modal data (Image, Document, web-URL) as input and extract the text fromthe multi-modal data. The extracted text is passed to Summarization model to generate summary of the text. The summary of the text is translated and generate speech to various languages and able to save the translatedtext and speech.

Dataset Selection and Pre-processing

The dataset used for the experiment is the News summary dataset obtained from *Kaggle*[1]. The dataset is made by collecting news articles from different news publishers like Guardian, Hindu, and Indian times from February to August 2017. The dataset contains 4515 samples of articles that includes the names of the labels as Author name, Headlines, URL of Article, Short text, Complete Article. The selected data set is undergone through the pre-processing process which is performed by following tasks. Firstly, removing null values using *dropna()*[2] function. Then, the Tokenizing process is considered as a key part. In this process, the strings are split into sub-word token strings and the sub-word tokens are converted to ids. The T5TokenizerFast is used for preparing the inputs for this model. T5TokenizerFast was implemented to allow significant speed-up while performing batched tokenization and additional methods to map between the original string and token spaces.

T5 Model Hyperparameters

The Hyper-parameters are tuned based on considering the available computational resources and power at hand. Manual Configuration is used to select the hyper-parameters. The dataset is divided into training and testing data using the *train_test_split*[3] method from *sklearn.model_selection* module. This process is used to divide the data into train and test data. Training data contains 80% and testing data contains 20% samples of the whole dataset

- Global seed = 42 (default)
- Learning Rate = 0.0001
- Optimizer = AdamW
- TRAIN_BATCH_SIZE = 8
- VALID_BATCH_SIZE = 8
- TRAIN_EPOCHS = 5
- Text_max_token_len = 512
- Summary_max_token_len = 128

Text Extraction from Images

In this experiment text from the given image is extracted [5],[19],[21] using *tesseract*[4] an open-source optical character recognition (OCR) engine. It recognizes the text from images, the text can be computer generator or handwritten. It extracts the text from images in two-pass process. In first pass tesseract tries to extract each word form the image. All the satisfactory words are used as training data to train an adaptive classifier. This classifier tries to recognize text more accurately. The classifier had learned some knowledge for contributing near the top of the page. The words which are not well recognized in the first pass are recognized again in the second pass using adaptive classifier. In final phase checks and resolves the alternative hypothesis to locate the x-height for small cap text and fuzzy spaces.

Web Extraction

Information extraction from the internet is called web scraping. In this experiment, the text is extracted from the websites using *beautiful*[5] *Soup* an open-source python library [18],[20],[22]. Following steps are made to extract text from the websites. In the first step collect the repositories and mark the data components. In the second step, relevant information from each DOM component is extracted. In the third step, associated links and location, specific information from the repository is stored. In the fourth, step the repository description is documented for further pattern matching. The final step encodes the scrapped digital content.

[1] https://www.kaggle.com/sunnysai12345/news-summary

[2] https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.dropna.html

[3] https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.train_test_split.html

[4] https://pypi.org/project/pytesseract/

[5] https://pypi.org/project/beautifulsoup4/

[6] https://pypi.org/project/googletrans/

[7] https://pypi.org/project/gTTS/

[8] https://pypi.org/project/rouge/

Text Translation

In this experiment, Text Translation is done using *googletrans*[6] python library [23]. It is an open-source library for implementing Google translates API. It is a statistical machine translation (SMT). Google Translate (GT) translate the whole sentence at a time, rather than just word by word. GT uses the commonality found inbetween many languages for translation. GT translate the given text to English and then translate it to a targetlanguage. GT is based on Google Neural Machine Translation, a large artificial neural network. GT follows example-based machine translation and learns from millions of examples.

Speech Generation

In this experiment, Speech Synthesis is done by the using *gTTS*[7] python library [24],[25]. Text-to-speech synthesis is a method for generating spoken language from the descriptive written text. Speech synthesis is done in two steps. In the first step input text is processed through Natural Language Processing (NLP) methods to perform Inference engine, Linguistic formalism and Logical inferences and generate Narrow Phonetic Transcriptions and phones Prosody. This processed text is passed to the second step where mathematical model algorithms and computation are done in digital signal processing and generates the speech.

Evaluation Metrics

Evaluating the quality of system generated summaries is an avoidable task for the development of summarization models. Manual evaluation of summaries is a costly and burdensome task. So, automatic evaluation of summarizing is efficient. In this experiment, the summarization model is evaluated by using ROUGE[8] (Recall-Oriented Understudy for Gisting Evaluation). The scores are generated by the computed number of words overlapping between the system-generated summary and reference summary. There are different types of ROUGE namely ROUGE-L, ROUGE-S, ROUGE-W, ROUGE-N. In this experiment, ROUGE-L and ROUGE-N (ROUGE-1, ROUGE-2) are measured.

A. ROUGE-N:

It is measured by computing n-grams overlapping between system generated and reference summaries, where n-grams refer to the number of consecutive tokens. ROUGE-1 is for unigram, ROUGE-2 is for bigramoverlapping tokens.

B. ROUGE-L:

It was measured by computing the Longest common subsequence (LCS) matching between reference and system-generated summaries.

RESULTS

In this experiment T5 (Text-to-Text Transfer Transformer) method is trained for the summarization problem using news summary dataset. In this experiment the summarization model performs well due to extreme care taken for hyper parameters tuning. The below table (Table -1) show the achieved and existing [13] ROUGE values.

TABLE 1. ROUGE scores between existing and achieved results

	ROUGE-1			**ROUGE-2**			**ROUGE-L**		
	Recall	**Precision**	**F1**	**Recall**	**Precision**	**F1**	**Recall**	**Precision**	**F1**
Achieved Results	0.494	0.502	0.496	0.283	0.293	0.287	0.458	0.466	0.460
Existi ng Results	0.480	0.467	0.473	0.269	0.261	0.265	0.389	0.338	0.361

The below graph (Fig -2) plots the ROUGE scores between achieved results and existing results.

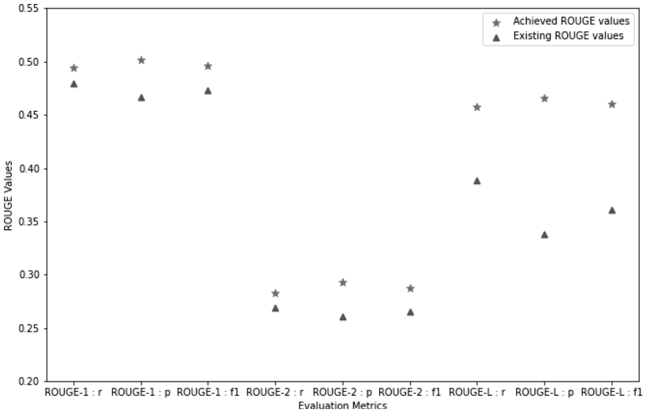


FIGURE 2. Comparing ROUGE values between achieved and exiting results

In this experiment, multi-modal data summaries are generated by extracting text data from different data holders like Images, documents, web URLs. Different types of methods are used to extract data from different types of input. The data from Images and web-URLs are extracted using optical character recognition and web-scraping respectively. The summary text is translated, and speech synthesised using python translation and speech synthesis libraries.

CONCLUSION AND FUTURE SCOPE

In this paper, we had proposed a Text-to-Text Transfer based model for the abstractive multimodal text summarization that uses the information of the three modalities namely text, image, web URLs. It uses the tri-model attention layer to utilise and decode all the modalities. We explored the role by adding the text extraction process from the web URLs and text transformation with a downloader into the local language shows a remarkable achievement with the effectiveness of our approach by fine-tuning the hyperparameters, but still, there are some areas which we would like to consider in further research. This model shows an abstractive technique that doesn’t work on images in the web URL it only extracts text from the URL. We would be working on that to extract better results. We would also be working in the direction of sentence compression from the abstracted text in our summaries.

REFERENCES

1. Rahul, S. Adhikari and Monika, "NLP based Machine Learning Approaches for Text Summarization," 2020 *Fourth International Conference on Computing Methodologies and Communication (ICCMC), (*2020), pp. 535-538.
2. Awasthi, K. Gupta, P. S. Bhogal, S. S. Anand and P. K. Soni, "Natural Language Processing (NLP) based Text Summarization - A Survey," 2021 6th *International Conference on Inventive Computation Technologies (ICICT),* (2021), pp. 1310-1317.
3. J. Zhu, Y. Zhou, J. Zhang, H. Li, C. Zong, and C. Li, “Multimodal Summarization with Guidance of Multimodal Reference”, *AAAI*, vol. 34, no. 05, (Apr. 2020), pp. 9749-9756.
4. Maududie, W. E. Y. Retnani and M. A. Rohim, "An Approach of Web Scraping on News Website based on Regular Expression*," 2018 2nd East Indonesia Conference on Computer and Information Technology (EIConCIT), (2018)*, pp. 203-207.
5. R. Malik and SeongAh Chin, "Extraction of text in images," *Proceedings 1999 International Conference on Information Intelligence and Systems (Cat. No.PR00446)*, (1999), pp. 534-537.
6. Khullar, Aman & Arora, Udit. (2020). MAST: Multimodal Abstractive Summarization with Trimodal Hierarchical Attention. 60-69. 10.18653/v1/2020.nlpbt-1.7.
7. C. Subakan, M. Ravanelli, S. Cornell, M. Bronzi and J. Zhong, "Attention Is All You Need In Speech Separation," ICASSP 2021 - 2021 *IEEE International Conference on Acoustics,* Speech and Signal Processing (ICASSP), (2021), pp. 21-25.
8. J. Chen and H. Zhuge, "Extractive Text-Image Summarization Using Multi-Modal RNN," 2018 14th *International Conference on Semantics, Knowledge, and Grids (SKG),* (2018), pp. 245-248.
9. Gupta, Vishal & Lehal, Gurpreet. (2010). A Survey of Text Summarization Extractive Techniques. *Journal of Emerging Technologies in Web Intelligence. 2. 10.4304/jetwi.2.3.258-268.*

10. N. S. Shirwandkar and S. Kulkarni, "Extractive Text Summarization Using Deep Learning," 2018 Fourth *International Conference on Computing Communication Control and Automation (ICCUBEA),* (2018), pp. 1-5.
11. H. T. Le and T. M. Le, "An approach to abstractive text summarization," 2013 International Conference on Soft Computing and Pattern Recognition (SoCPaR), (2013), pp. 371-376.
12. Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS'17). Curran Associates Inc., Red Hook, NY, USA, 6000–6010.
13. Zolotareva, Ekaterina & Misikir Tashu, Tsegaye & Horvath, Tomas. (2020). Abstractive Text Summarization using Transfer Learning.
14. M. Gorai and M. J. Nene, "Layout and Text Extraction from Document Images using Neural Networks," *2020 5th International Conference on Communication and Electronics Systems (ICCES),* (2020), pp. 1107-1112.
15. Z. Hao, J. Ji, T. Xie and B. Xue, "Abstractive Summarization Model with a Feature-Enhanced Seq2Seq Structure," *2020 5th Asia-Pacific Conference on Intelligent Robot Systems (ACIRS),* 2020, pp. 163-167.
16. T. Hassan and R. Baumgartner, "Intelligent Text Extraction from PDF Documents," *International Conference on Computational Intelligence for Modelling, Control and Automation and International Conference on Intelligent Agents, Web Technologies, and Internet Commerce (CIMCA-IAWTIC'06),* (2005), pp. 2-6.
17. M. S M, R. M P, A. R E and E. S. G SR, "Text Summarization Using Text Frequency Ranking Sentence Prediction," *2020 4th International Conference on Computer, Communication and Signal Processing (ICCCSP), (2020), pp. 1-5.*
18. S. Thivaharan., G. Srivatsun. and S. Sarathambekai., "A Survey on Python Libraries Used for Social Media Content Scraping," *2020 International Conference on Smart Electronics and Communication (ICOSEC), (2020), pp. 361-366.*
19. R. Smith, "An Overview of the Tesseract OCR Engine," *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007),* (2007), pp. 629-633.
20. ChunmeiZheng et al, A Study of Web Information Extraction Technology Based on Beautiful Soup, Journal of computers, Volume 10, Number 6, (November 2015).
21. Stephan Richter, lxml - XML and HTML with Python, https://lxml.de/
22. V. Singrodia, A. Mitra and S. Paul, "A Review on Web Scrapping and its Applications," *2019 International Conference on Computer Communication and Informatics (ICCCI)*, (2019), pp. 1-6.
23. Ghasemi, Hadis & Hashemian, Mahmood. (2016). A Comparative Study of Google Translate Translations: An Error Analysis of English-to-Persian and Persian-to-English Translations. English Language Teaching. 9. 13. 10.5539/elt. v9n3p13.
24. Nwakanma, Ifeanyi & Oluigbo, Ikenna & Izunna, Okpala. (2014). Text – To – Speech Synthesis (TTS). 2. 154-163.
25. Dong, Li & Yang, Nan & Wang, Wenhui & Wei, Furu & Liu, Xiaodong & Wang, Yu & Gao, Jianfeng & Zhou, Ming & Hon, Hsiao-Wuen. (2019). Unified Language Model Pre-training for Natural Language Understanding and Generation.