CS230

Abstractive Summarization on COVID-19 Publications

Zhengzhi Lou
szlou@stanford.edu

Ju Zhang
juzhang@stanford.edu

Abstract

The COVID-19 pandemic has made a drastic impact on the whole world and sparked intensive scientific research into this disease, which generates tremendous amounts of literature. In this project, we explored how to apply three deep-learning-based abstractive summarization models, namely BART, T5, and PropheNet, to generate summaries on COVID-19 academic literature corpus. We successfully summarized the abstract of each article and evaluate the metric from each fine-tuned model. Finally, we used our fine-tuned model to generate summarization across papers that share a similar topic. We conclude that though promising, the current abstractive summarization methods are still far from perfect to summarize complex subject matters in a meaningful way.

1 Introduction

Since the beginning of 2020, the spread of COVID-19 has caused a global pandemic and deprived lives of thousands of people. Under the current situation, tremendous scientific efforts have been made to fight the disease. Consequently, numerous research literature is published daily to provide valuable insights into the disease. However, the sheer size of the corpus also poses a challenge for researchers and medical professionals who want to quickly find information and understand updates within a particular research branch, since searching the related topics and reading papers one by one is pretty time-consuming. To help resolve this problem, we propose a deep-learning-based pipeline to perform abstractive text summarization on COVID-19 related scientific publications. In particular, this project aims to build a pipeline that can 1) cluster a large volume of papers based on their content, and 2) generate a concise and high-level summary on each cluster of corpus.

2 Related work

We used a two-stage approach to solve this problem. First, we clustered the articles into distinctive topics. Then, we generated a natural language summary for each cluster.

Document clustering has been intensively studied and it can be achieved by a variety of methods. Previous studies include using traditional machine learning algorithms, such as K-means clustering based on tokens [1] as well as topic modeling [2]. More recently, with the help of deep-learning-based models such as BERT, sentence or word encoding can be generated first then fed into algorithms including k-means clustering [3].

On abstractive summarization, the current state-of-the-art approach stems from the application of attentional recurrent neural network (RNN) on machine translation [4]. Several papers introduce this transplant: Nallapati et al. tailored the model by capturing hierarchical document structure and modeling rare words [5], while Chopra et al. assigned scores to each sentence for the RNN to focus

CS230: Deep Learning, Spring 2020, Stanford University, CA. (LateX template borrowed from NIPS 2017.)

their attention on [6]. Later, improvements have been proposed [7] [8]. We have decided to adopt this approach and push it further. We want to incorporate recent advancements in word representation such as BERT [9] [10], BART [11], and ProphetNet [12].

3 Dataset and Preprocessing

We utilize dataset “COVID-19 Open Research Dataset Challenge (CORD-19)" from Kaggle [13]. It contains more than 130,000 scholarly articles, including over 60,000 with full text, about COVID-19, SARS-CoV-2, and related coronaviruses. Each paper has been split into different chunks including authors, abstracts, body text, and references, saved as a JSON file under different keys.

The dataset seems be to aggregated from multiple sources and thus is in a relatively messy state. Some of the articles have missing values in title, abstract, or authors. We have removed papers that are: 1) without title or abstracts 2) with unreasonable title length (shorter 5 characters or abstract longer than 30 words) 3) with unreasonable abstract length (shorter than 20 words or larger than 500 words). Also, we noticed that not all papers are in English. We detected some entries in Italian and removed them from the overall training data as well. The final dataset we have contains 40,430 literature entries.

Next, we conducted document clustering based on the abstracts. 40,430 papers are divided into 11 clusters. The distribution of clusters is shown in Figure 1. For the fine-tuning purpose, we divide the selected paper into 60:20:20 train:dev:test subsets. Model performance was compared only based on the test data subset.

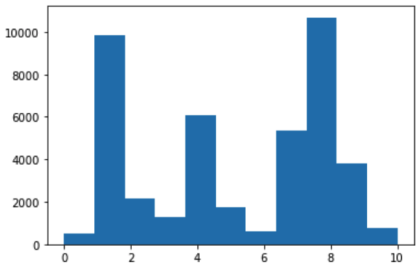

Figure 1: Papers are divided in to 11 topics based on their highest probability of belonging to each topic

4 Methods

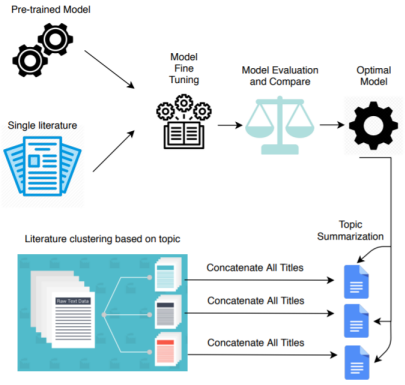

Figure 2: Overview of the project workflow

The workflow of the project is demonstrated in Figure. 2. We began from pre-trained models publicly available and then fine-tuned the models with the training dataset, then we compared the fine-tuned

models based on ROUGE. The best model was selected and used to summarize the topic of papers. Titles of all papers in one topic were concatenated to represent the corpus of the topic.

4.1 Abstractive Summarization on Single Paper Level

When using deep learning-based models to generate a title from abstract for each individual paper, we have adopted three Transformer [14] models: BART, T5, and ProphetNet.

4.1.1 BART

BART model is a denoising autoencoder for pretraining sequence-to-sequence models, which could be thought of as a generalized BERT model [11]. It is implemented as a sequence-to-sequence model with a bidirectional encoder over corrupted text and a left-to-right autoregressive decoder. When properly fine-tuned, BART is effective for text generation and text comprehension tasks. We sourced and adopted the open-source NLP framework developed by HuggingFace, Inc. [15].

4.1.2 T5

The T5 (Text-To-Text Transfer Transformer) model [16] is a unified framework that converts every language problem into a text-to-text format. It handles a wide variety of tasks, such as translation, classification, QA, summarization, by treating all tasks uniformly as taking some input text and outputting some text where the task type is embedded as descriptors in the input. This approach enables a single model to perform a wide variety of supervised tasks such as translation, classification, QA, summarization, and even regression. We sourced and adopted the open-source NLP framework developed by HuggingFace, Inc [15].

4.1.3 ProphetNet

Extending the idea of predicting the next word in a sentence, ProphetNet tries to predict future n-grams[12]. This model mitigates the shortcoming of previous models focusing on local correlations instead of long-term dependencies, especially when greedy decoding is used in place of beam search.

ProphetNet is different from Transformer with two major changes. On the decoder side, it tries to predict n future tokens simutaneously:

$$p(y_t|y_{<t}, x), ..., p(y_{t+n-1}|y_{<t}, x) = \textbf{Decoder}(y_{<t}, H_{\text{enc}}) \tag{1}$$

where H_{enc} and the representation output from encoder. The second change is the n-stream self-attention module. The module contains a main stream self-attention module for the hidden state, and n predicting stream self-attention for the n-gram outputs.

In this project, we adopted source code by authors of the original paper with further changes.

4.1.4 Model Fine-Tuning

We fine-tuned each of the Transformer models with our COVID-19 corpus. Specifically, we use abstracts as inputs, with titles of corresponding articles as outputs. When then evaluated and compared three models using Rouge score [17].

4.2 Clustering Latent Dirichlet Allocation (LDA)

Latent Dirichlet allocation (LDA) is a generative statistical model that allows sets of observations to be explained by unobserved groups on why some parts of the data are similar [18]. For example, if observations are words collected into documents, it posits that each document is a mixture of a small number of topics and that each word's presence is attributable to one of the document's topics.

4.3 Abstractive Summarization on Cluster Level

We concatenate the titles of each cluster, up to the maximum length corresponding to the limit of the Transformer models we deployed and analyzed the result manually in the next section.

5 Experiments/Results/Discussion

5.1 Abstractive Summarization on Individual Papers

We used BART, T5, and ProphetNet to conduct abstractive summarization on each individual article. Model are fine-tuned on our COVID-19 literature dataset before being evaluated. The cross-comparison between ROUGE scores of different model results shown in Table 1.

	ProphetNet			BART			T5		
Score Type	low	mid	high	low	mid	high	low	mid	high
ROUGE-1-R	**62.65**	**64.06**	**65.29**	10.77	10.9	11.04	13.79	13.95	14.11
ROUGE-1-P	27.54	28.38	29.18	30.46	30.79	31.15	**56.21**	**56.67**	**57.11**
ROUGE-1-F	**37.14**	**38.12**	**39.01**	15.26	15.42	15.61	21.32	21.54	21.75
ROUGE-2-R	**40.09**	**41.48**	**42.98**	2.64	2.72	2.79	5.09	5.2	5.33
ROUGE-2-P	17.32	18.08	18.89	8.07	8.3	8.54	**21.46**	**21.88**	**22.29**
ROUGE-2-F	**23.43**	**24.37**	**25.31**	3.81	3.92	4.02	7.88	8.05	8.22
ROUGE-L-R	**51.56**	**52.78**	**54.06**	8.78	8.89	9	9.86	9.99	10.12
ROUGE-L-P	22.22	22.97	23.70	25.44	25.74	26.06	**40.75**	**41.17**	**41.58**
ROUGE-L-F	**30.13**	**31.00**	**31.82**	12.52	12.66	12.8	15.25	15.43	15.61

Table 1: ROUGE scores from three models. ROUGE1: Rouge score based on unigram, ROUGE2: ROGUE score based on bigrams, ROUGE-L: ROUGE score based on longest sequence, R:recall, P:precision, F:f-1

We found that the fine-tuned ProphetNet is strong in ROUGE recall score and f-1 measures, whereas the fine-tuned T5 model has a relatively good ROUGE precision score. The trend is consistent regardless of the length of the n-grams we are focusing on.

An example of abstractive summarization on a single abstract is shown below:

> Original Title: **rapid identification of malaria vaccine candidates based on a-helical coiled coil protein motif**
>
> Generated Summarization: **a-helical coiled coil domains of proteins predicted to be present in the parasite erythrocytic stage are expected to mimic structurally "native" epitopes . the 95 chemically synthesized peptides were all specifically recognized by human immune sera and by immunization of mice . these antibodies did not show significant cross reactions.**

5.2 LDA Classification Summarization

Our ultimate goal is to summarize each topic of scientific literature. To achieve that, clustering papers is a necessary step. Using the LDA method, we associate all literature with scores for 11 topics, computed based on the unigrams and bi-grams. The number of 11 was chosen based on the best coherence score. After that, we assign the topic of the highest probability to the paper. Figure 3 shows the topic modeling.

5.3 Abstractive Summarization on topics

The last step is to summarize across all articles in a given topic. We concatenate all paper titles in a given topic, up to the maximum token limit (512) imposed by our model, and generate the summarization using the fine-tuned ProphetNet model. An example of the result is shown below

> **maternal ly - derived antibodies enhance the im mun ogenic ity of infectious disease viral vaccines in non hum an primate s real - time reverse transcription pc r ass ay for detection of por cine toro virus type 2 infection in dia rr hei c dogs**

Even though the summarization contains phrases such as "vaccines in non-human primates", it is still hard to infer any meaningful insights from it.

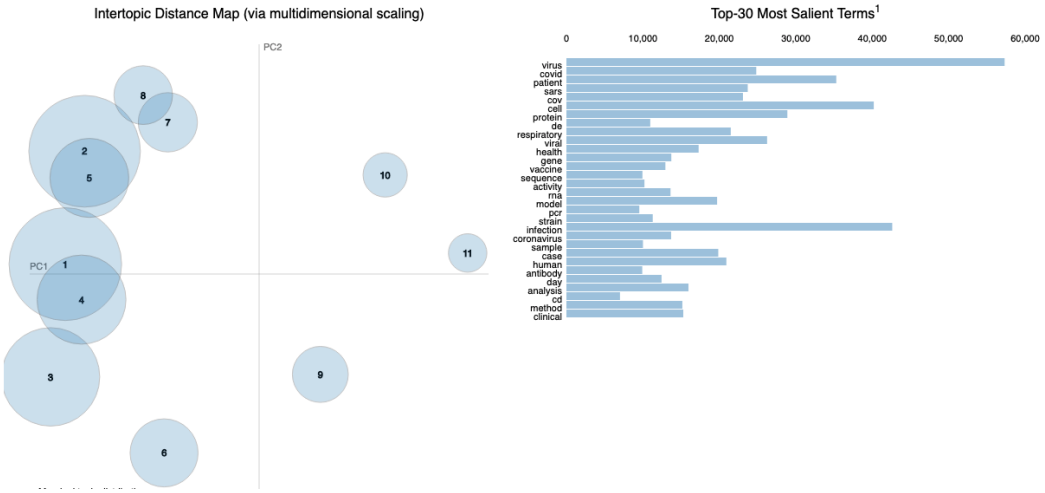

Figure 3: LDA Topic Modeling. Left: 11 topics are generated using LDA; Right: top 30 grams that is most salient

6 Conclusion /Future Work

Our project goal of generating human-readable summarization from a scientific paper or a cluster of is a difficult task. We succeed to find a use case and implement abstractive summarization on COVID-19 literature, however, the generated result is less ideal. Besides intrinsic issues with transformer models themselves, In this project e are limited by 1) computing resource 2) Lack of data

The original plan is to summarize using the full text of papers. However, we find that most of the mainstream transformer models limited the maximum sequence to 512/1024. Even if that can be circumvented, it is extremely hard to fine-tune the models due to the GPU RAM limitation and the size of pre-trained transformer models.

All pre-trained models we are able to find were not based on the corpus of biomedical matters. It is necessary to apply transfer learning and fine-tune the model on our COVID-19 datasets. However, in our case we use title is ofter not the optimal or even bad representation of the content of scientific papers. If given unlimited time and human resources, the optimal way is to annotate each paper and generate ground truth. In this project, we do not have sufficient time to do that.

There exist other deep learning-based abstractive summarization, for example, the BERT-based summary model, bertsum [19]. We do not have time to explore those. It would be interesting to compare across all the mainstream summarization transformer models and test their performance on scientific paper summarization.

7 Contributions

Zhengzhi Lou and Ju Zhang contributed equally to this project. The team members split the workload across different tasks, including data processing, model implementation, model training, and prediction.

References

[1] Rajender Nath Anjali Vashist. Document clustering using improved k-means algorithm. *International Journal of Science and Research*, 2319-7064, 2013.

[2] Nils Newman Arho Suominen Chyi-Kwei Yau, Alan Porter. Exploring the limits of transfer learning with a unified text-to-text transformer. *Scientometrics*, 100, pages767–786(2014), 2014.

[3] Y. Li, J. Cai, and J. Wang. A text document clustering method based on weighted bert model. In *2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*, volume 1, pages 1426–1430, 2020.

[4] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

[5] Ramesh Nallapati, Bowen Zhou, Caglar Gulcehre, Bing Xiang, et al. Abstractive text summarization using sequence-to-sequence rnns and beyond. *arXiv preprint arXiv:1602.06023*, 2016.

[6] Sumit Chopra, Michael Auli, and Alexander M Rush. Abstractive sentence summarization with attentive recurrent neural networks. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 93–98, 2016.

[7] Romain Paulus, Caiming Xiong, and Richard Socher. A deep reinforced model for abstractive summarization. *arXiv preprint arXiv:1705.04304*, 2017.

[8] Abigail See, Peter J Liu, and Christopher D Manning. Get to the point: Summarization with pointer-generator networks. *arXiv preprint arXiv:1704.04368*, 2017.

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[10] Yang Liu and Mirella Lapata. Text summarization with pretrained encoders. *arXiv preprint arXiv:1908.08345*, 2019.

[11] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*, 2019.

[12] Yu Yan, Weizhen Qi, Yeyun Gong, Dayiheng Liu, Nan Duan, Jiusheng Chen, Ruofei Zhang, and Ming Zhou. Prophetnet: Predicting future n-gram for sequence-to-sequence pre-training. *arXiv preprint arXiv:2001.04063*, 2020.

[13] COVID-19 Open Research Dataset (CORD-19). 2020. Version 2020-MM-DD. Retrieved from https://pages.semanticscholar.org/coronavirus-research. Accessed 2020-MM-DD. doi:10.5281/zenodo.3715505.

[14] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[15] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, R'emi Louf, Morgan Funtowicz, and Jamie Brew. Huggingface's transformers: State-of-the-art natural language processing. *ArXiv*, abs/1910.03771, 2019.

[16] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv e-prints*, 2019.

[17] Chin-Yew Lin. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain, July 2004. Association for Computational Linguistics.

[18] Andrew Y. Ng David M. Blei and Michael I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research 3 (2003) 9*, 2003.

[19] Yang Liu. Fine-tune bert for extractive summarization. *arXiv preprint arXiv:1903.10318*, 2019.