Application and analysis of text summarization for biomedical domain content

Karen Ouyang
Biomedical Informatics
Stanford University
kjouyang@stanford.edu

Abstract

Biomedical information in the form of scientific articles and electronic medical records is increasing at an alarmingly fast pace. The output of publications in the biomedical domain is estimated to double every 5-10 years, currently with more than 3000 new articles published per day. There is clear utility in having systems that can automatically handle various natural language processing (NLP) tasks, such as text summarization. Summarization is the task of distilling longer text to a shorter version that retains the key information from the original text. The objective of this project is to apply NLP machine learning models for text summarization that perform well on general language text summarization datasets and further modify/adapt for biomedical domain specific text summarization. I evaluate and compare the performance on general language (CNN-DailyMail) versus biomedical-specific (BioASQ) datasets, and analyze results to leverage general language models for biomedical domain-specific application. The results demonstrate that models trained on general language achieve comparable results on a biomedical test set, outperforming the general language test set in one model.

1 Introduction

Summarization is the task of automatically condensing a piece of text to a shorter version while retaining key information content and overall meaning. This is an important topic for NLP research because of the increasing demand for various information access applications. There are two main paradigms to summarization: extractive and abstractive. Extractive approaches form summaries by extracting and concatenating the most important spans from the source text, while abstractive methods generate candidate summaries that contain novel words and phrases not featured in the source text, usually requiring additional rewriting operations. An extractive method has the benefit of maintaining reasonable levels of grammaticality and accuracy. Conversely, the ability to generalize, paraphrase, and introduce additional knowledge are key features in an abstractive framework.

While there have been significant strides in improving language tasks for general language, addressing domain-specific contexts still remains challenging. In the scientific biomedical field, the output of publications is estimated to double every 5-10 years, currently with more than 3000 new articles published per day. However, in contrast to general language tasks, it is challenging to generate a biomedical domain-specific dataset comparable in scale. In this project, I aim to leverage general language trained models for transfer learning of biomedical domain-specific application.

2 Related Work

Neural encoder-decoder models are widely used in text summarization applications. These models use recurrent neural networks (RNN) to encode an input sentence into a fixed vector, and create a new output sequence from that vector using another RNN. Word embeddings are used to convert language tokens to vectors that are used as inputs for these networks. Attention mechanisms allow the model to look back at parts of the encoded input sequence while the output is generated. Nallapati et al., 2016 developed a neural abstractive model [1] and subsequently extractive approach using hierarchical RNNs to select sentences [2]. This follows the creation of one of the first large-scale abstractive summarization baseline for longer text - the CNN-DailyMail dataset.

The language representation model BERT (Bidirectional Encoder Representations from Transformers) is based on work in pre-training contextual representations [3]. The key is that BERT pre-trains deep bidirectional representations using only a plain text corpus (Wikipedia), by jointly conditioning on both left and right context in all layers. As such, the pre-trained BERT representations can be fine-tuned with just one additional output layer to create models for a wide range of tasks without having to substantially modify model architecture for each specific task.

3 Approach

The approach for this project is to implement and analyze abstractive and extractive text summarization machine learning models for general language as well as biomedical domain-specific text.

PGEN-abstractive

For abstractive text summarization, we used a sequence-to-sequence model that utilizes recurrent neural networks (RNNs) for biomedical text summarization (Figure 1). The model has an encoder and decoder component. The encoder RNN reads in the source content word by word, producing a sequence of encoder hidden states. The decoder uses this information and begins to output words to form a summary. At each step, the decoder receives as input the previous word of the summary and uses it to update the decoder hidden state. I modified and fine-tuned the model from See et al. (2017) that uses a hybrid pointer-generator network to copy words from the source text via pointing, aiding accurate reproduction of information, while retaining the ability to produce novel words through the generator [4]. These features are especially important for biomedical content because there are substantial numbers of biomedical terms/words not present in a general language vocabulary.

BERT-extractive

For extractive text summarization, we incorporated pre-trained contextual embeddings (PCE) (Figure 1). The objective of PCE is to generate word embeddings that depend on the context in which the word appears in the text opposed to traditional word embeddings such as Word2Vec where each word in the vocabulary is mapped to a fixed vector, regardless of context. PCEs are built by pretraining weights on a large-scale language modeling dataset, then loading the pretrained weights into a model. I used the language representation model BERT which pre-trains deep bidirectional representations using only a plain text corpus, by jointly conditioning on both left and right context in all layers. As such, the pre-trained BERT representations can be fine-tuned with an additional output layer to create models for a wide range of tasks. I further adapted and fine-tuned a model described in Liu et al. 2019 which adopts BERT pre-trained on a large dataset with a powerful architecture for learning complex features to further boost the performance of extractive summarization [5]. Input sequence and embeddings of BERT were modified to enable summary extraction and summarization-specific layers were stacked on top of the BERT outputs, to capture source text-level features for extracting summaries.

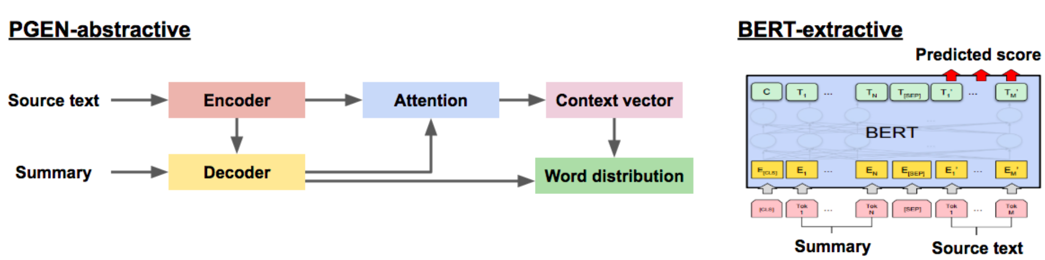

Figure 1: Schematic of abstractive and extractive text summarization models

4 Experiments

4.1 Data

General language dataset

The CNN/Daily Mail dataset contains online news articles paired with multi-sentence summaries. Dataset downloaded from https://cs.nyu.edu/ kcho/DMQA/.

- Training: 287,226 pairs | Validation: 13,368 pairs | Testing: 11,490 pairs

Biomedical domain dataset:

BioASQ contains biomedical texts paired along with gold standard (reference) summaries. Dataset obtained from http://bioasq.org/.

- Training: 815 pairs | Validation: 193 pairs | Testing: 665 pairs

CONTEXT: a cheeky monkey was captured on camera snatching a banana from a female tourist before slapping her gopro when she got too close. filmed in the thai town of kanchanaburi, the monkey approaches the woman, who holds a banana, with its outstretched hands. grabbing it in both hands, the monkey takes a small bite before pulling it from its skin, which he leaves with the lady. (...) getting right up into the camera's lens the monkey appears to sniff it while diverting its eyes, as if hoping that it is food. the cheeky monkey snatches the banana from the woman's hand and begins scoffing it down, realising that it is out of luck, it returns to its original position and continues tucking into the banana. (...) the footage was captured by maja and diano, a pair who describe themselves on their youtube channel as a young married couple with an impulse to explore, film and edit great travel moments. (...)
SUMMARY (Ground Truth): the woman holds out a banana, which the monkey quickly snatches. monkey then approaches the camera and sniffs it to see if it is food. woman gets too close to protective monkey and it slaps her gopro. the footage was captured by a couple in thai town of kanchanaburi

Figure 2: Example of context summary pair from CNN-DailyMail

4.2 Evaluation Method

ROUGE is an automatic evaluation method based on the similarity of n-grams. ROUGE-n scores are calculated as a ratio of the maximum number of n-grams co-occurring between a candidate and reference summary over the number of n-grams in the reference summary. ROUGE is a recall value that measures how many n-grams from the reference summaries appeared in the candidate summaries. The ROUGE-L measure employs the concept of longest common subsequence (LCS) between the two sequences of text. The thought is that the longer the LCS between two summary sentences, the more similar they are.

4.3 Experimental Details

Data Pre-processing

CNN/Daily Mail dataset was downloaded from https://cs.nyu.edu/ kcho/DMQA/ and further preprocessed to format for input into model. The biomedical domain-specific text summarization dataset was obtained from http://bioasq.org/. The BioASQ dataset consists of several different types of language task challenges. For specific text summarization task, I wrote custom python scripts to collect biomedical publication abstract, relevant paragraph content, and summarization text. I also wrote additional functions to format the data as needed for training, validating, and testing the model.

Model training

Experiments were carried out on a Google compute engine with machine type: n1-standard-16 (16 vCPUs, 60 GB memory) and 2 x NVIDIA Tesla T4 GPUs.

To generate models for text summarization, I trained BERT-extractive model (parameters: learning rate 2e-3, dropout 0.1, batch-size 3000, train-steps 50000, attention-heads 8, bert-base-uncased version) and PGEN-abstractive model (parameters: train-steps 160000, hidden-dim 256, emb-dim 128, learning-rate 0.15, batch-size 16, vocab-size 50000), respectively, on the CNN/DailyMail and BioASQ joint training set and evaluated on the validation set to determine the checkpoint model with the best performance. Performance of the trained models were then evaluated on the test sets using the ROUGE metrics.

Analysis

The training and validation sets from CNN-DailyMail and BioASQ were combined to jointly train the two models. Evaluation of model performance was then performed on the separate test sets. I compared the ROUGE scores for the CNN-DailyMail and BioASQ test sets across the two models and analyzed the distribution of source text and summary length. I then visualized attention to determine which words and phrases were highlighted in creating the summary.

4.4 Results and Analysis

The models were evaluated on the CNN-DailyMail versus BioASQ test sets with the ROUGE metric (Table 1). For the CNN-DailyMail test set, the BERT-extractive model achieved higher ROUGE scores compared to the PGEN-abstractive model. This is expected and is routinely reported as extractive systems tend to achieve higher ROUGE scores than abstractive. The same trend is also observed with the BioASQ test set. Specifically, there is a greater performance difference between the models for BioASQ compared to CNN-DailyMail test set (average 13.95 vs 6.49).

When comparing the performance between CNN-DailyMail and BioASQ test sets, we find that the models achieve comparable results. BioASQ outperforms CNN-DailyMail on the BERT-extractive model while CNN-DailyMail edges out BioASQ on the PGEN-abstractive model.

Table 1: Comparison of CNN-DailyMail versus BioASQ results with ROUGE

Model	Test dataset	ROUGE-1	ROUGE-2	ROUGE-L
BERT-extractive	CNN-DailyMail	43.16	20.22	39.56
	BioASQ	45.85	32.20	39.93
PGEN-abstractive	CNN-DailyMail	35.39	15.11	32.97
	BioASQ	32.85	17.74	25.54

To ensure that the models were properly trained, we observe the training and validation loss during model training (Figure 3). In both models, the training/validation losses were similar and converged as the training steps increased. This supports that both models were appropriately and sufficiently trained.

Figure 3: Model training and validation loss

We further analyzed the distribution of word counts for the source text and summaries across CNN-DailyMail and BioASQ datasets (Figure 4). The results show that source texts for CNN-DailyMail are on average longer than those for BioASQ. While the summary lengths are slightly longer in CNN-DailyMail compared to BioASQ, the distribution is tighter for CNN-DailyMail and much more variable and spread-out for BioASQ. We reason that the higher ROUGE scores for BioASQ on the BERT-extractive model is most likely due to the overall shorter source text and summary lengths.

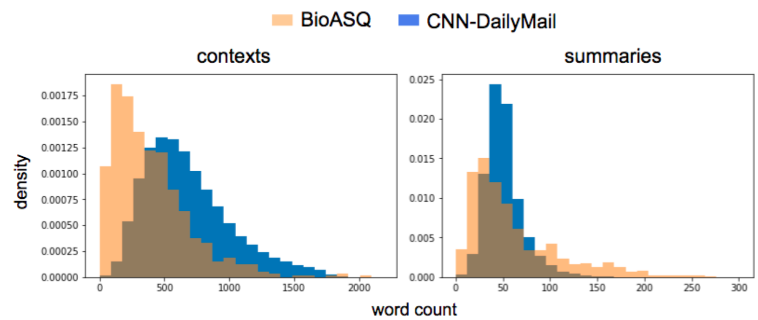

Figure 4: Word count distributions for contexts and summaries

Finally, we observe an example of summary outputs from both models on a BioASQ source text sample where attention is visualized (Figure 5). The word of interest is denoted in the orange square and the attention distribution projected in yellow on the source text. The likelihood a word is generated illustrated by the intensity of the green highlight. The probability of generating the word of interest is listed (prob = 0.249).

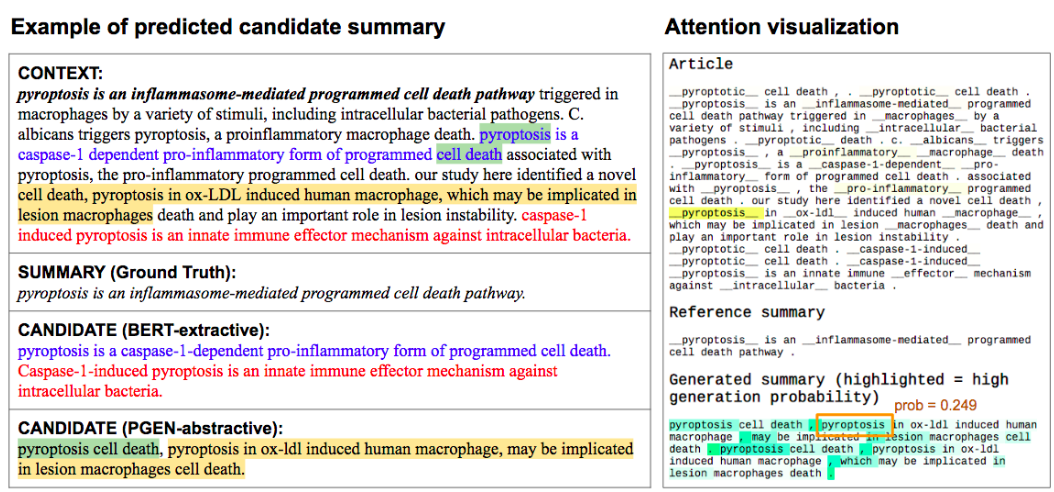

Figure 5: Example of predicted candidate summary and attention visualization

5 Conclusions and Future Work

Our results demonstrate that models overwhelmingly trained on CNN-DailyMail achieve comparable summarization results on BioASQ. Higher ROUGE scores for the BioASQ dataset in the extractive model are likely due to differences in distribution of source text and summary lengths. Slightly lower ROUGE scores in the abstractive model may represent insufficient biomedical text training.

I would like to autogenerate larger scale biomedical literature text summarization datasets based on increased utility of "highlights" in research manuscript cover letters. Furthermore, synthesizing and summarizing information across multiple sources is a core task that scientists perform daily. For future work, I aim to develop a model that can accommodate multiple input source texts.

Thanks to the CS229 course staff for their help and Google Cloud Platform for providing compute resources.

6 References

[1] Ramesh Nallapati, Bowen Zhou, et al. Abstractive text summarization using sequence-to-sequence rnns and beyond. arXiv preprint arXiv:1602.06023, 2016.

[2] Ramesh Nallapati, Feifei Zhai, and Bowen Zhou. Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. Proceedings of the 31st AAAI conference, 2017.

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

[4] Abigail See, Peter J. Liu, and Christopher D. Manning. Get to the point: Summarization with pointer-generator networks. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 1073–1083, July 2017.

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