Summarization of Scientific Paper using Unified Model

SUMIT AGARWAL suagarwal@cs.stonybrook.edu
UTKARSH GARG ugarg@cs.stonybrook.edu
VIRAJ KAMAT vikamat@cs.stonybrook.edu

1 Abstract

Given existing approaches to summarize text such as data from newspaper sets, we wish to develop over these approaches on a relatively new domain, i.e Scientific research-papers,in our case. We aim to produce a more efficient and effective model with a novel loss function that can utilize segments of scientific papers such as title, abstract/synopsis and the paper content to produces a summary that captures the gist of the paper effectively. We also wish to evaluate our results,against the established baselines.

2 Task Definition

One concern of our current time are the vast number of scientific papers published everyday, it would be humongous for any individual to scan through all these papers published and to keep pace with them. One approach to solve this issue is to apply Natural language processing to summarize the scientific papers effectively such that one can glance through the content quickly and understand the overall content of the paper.

One of the major challenges of summarizing text is that existing models in attempting to summarize content tends to get narrowly focused on one part of the content say a paragraph, and miss out on important information present in the other parts. This is important in terms of scientific paper summarizing as it needs to spread its learning on all aspects of the paper and produce the required summary. This is one of the reasons we call our approach a Unified approach.

To facilitate the summarizing of text in scientific papers we can look at two key segments of a paper

1. The title of the paper, which can be learnt from the abstract/synopsis of the scientific paper.

2. The abstract/synopsis of the paper which can be learnt from the over all content of the scientific paper.

Thus we wish to use the existing segments of the scientific paper to aid us in the construction of the model.

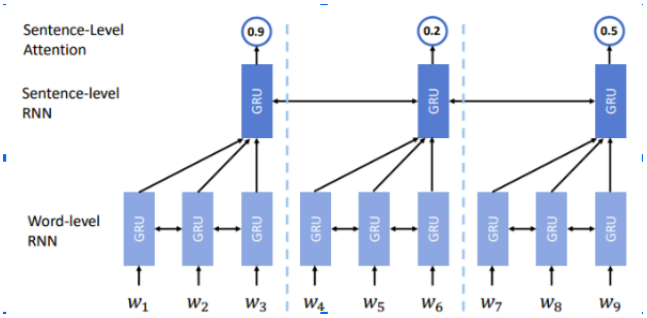

Figure 1: Extractor Model in use!

2.1 Existing Approaches

1. **Extractive Summarization :** This method selects the important sentences from the input text to generate the summary of a text. It picks the most important yet not coherent and concise sentences. Most extractive approaches rely on Reccurrent neural network models to achieve this task and tend to select only the first few sentences when learning since in most articles the gist of the content is present in the first few sentences.

   We can think of extractive summarization as a pen highlighter which only highlights the important sentences of a paragraph. We can already see why this method is not effective as it only narrowly focuses on a few sentences, rather than the entire content of the paper.

2. **Abstractive Summarization :** This method is an ongoing research in the Natural language processing and requires complex neural models to achieve its objective. Rather than focusing on a few sentences ,Abstractive summarization focuses on words in the sentence, which are then converted into new words using OOV (out of vocabulary) corpus to generate new phrases and sentences from scratch. It also relies on RNN's and since instead, involves a complex process understanding the language, the context and generating new sentences, it requires large sets of data to train the model.

3 Motivation

As mentioned previously, the need for summarization of text comes from the fact that a lot of content is available on a single topic, and people wish to get a glance of this text with ease. While most text summaries are built on newspapers, and existing models have shown some capabilities in doing so, we wish to apply these summarization techniques with modification such that they can be used on scientific papers.

Why ?

- Scientific papers get published at an astonishingly high rate, and research members in specific fields, in order to be up to date with the progress in their field, need to read these as they keep getting published. This gets hard as more and more papers get published every week. The aim of our model is thus to use a Unified model for text summarization in scientific papers such that the general gist of the scientific papers can be understood at a glance without compromising the quality of the content presented as is evident in the Extractive approach of summarization.

One such place where we assume our approach will be most useful is in scientific literature, where doctors and people in medical research can be up to date with latest improvements in medical research by glancing through various articles published in the field they themselves are researching into.

4 Proposed Approach

The extractive approach is an RNN that works by reading the entire document and memorizing the document. Since the aim is to summarize by picking those sentences in the document which are relevant it does so by once more traversing through each of the sentence memorized and classifying whether or not it belongs to the summary.

We can broadly divide our extractive approach into two types : -

Classify – The classifier is an RNN based sequence classification model that assigns each sentence a 0/1 binary value based on whether it belongs to the summary or not. It does that by scanning the entire document at once.

Select – In this case it acts as generative model that sequentially generates the indices of the sentences as they should belong to the summary i.e it assigns a rank to the sentences.

4.1 Unified Model

In our approach we combine the sentence level attention of the extractive approach and the word level attention of the abstractive approach so as to modulate the word-level attention, such that words in less attended sentences are less likely to be generated.Basically, the model uses extractive approach to shortlist important sentences with high recall to further facilitate the abstractor.In this way, we are overcoming the shortcomings of abtractive technique, by amalgamating benefits of extractive technique.

4.2 Procedure:

Select top K sentences from a document, using Extractive Model, and run Abstractive Model on these selected sentences while dynamically generating summary words. Moreover, we will use sentence-level attention to calculate dynamic word-level attention.

Models in use:

- For Extractive model [EM] - Bidirectional GRU
- For Abstractive model [AM] - Encoder (Bidirectional LSTM) and Decoder (Unidirectional LSTM)

Input to the model

- The input of both extractor and abstracter is a sequence of words w $= [w_1, w_2, \ldots, w_m, \ldots]$, where m is the word index.

- The sequence of words also forms a sequence of sentences s $= [s_1, s_2, \ldots, s_n, \ldots]$, where n is the sentence index.

- The m^{th} word is mapped into the $n(m)^{th}$ sentence, where n$(\cdot)$ is the mapping function.

- The output of the extractor is the sentence level attention $\beta = [\beta_1, \beta_2, \ldots, \beta_n, \ldots]$, where n is the probability of the n^{th} sentence been extracted into the summary. On the other hand, our attention-based abstractor computes word-level attention $\boldsymbol{\alpha}^t = [\alpha_1^t, \alpha_2^t, \ldots, \alpha_m^t, \ldots]$, dynamically while generating the t^{th} word in the summary. The output of the abstracter is the summary text $\mathbf{y} = \left[y^1, y^2, \ldots, y^t, \ldots\right]$, where y^t is t^{th} word in the summary.

The **updated word attention** is given as follows:

$$\hat{\alpha}_m^t = \frac{\alpha_m^t \times \beta_{n(m)}}{\sum_m \alpha_m^t \times \beta_{n(m)}}$$

where, β_n is the sentence level attention, $\hat{\alpha}_m^t$ is the modulated word level attention

Figure 2: Our unified model combines the word-level and sentence-level attentions. Inconsistency occurs when word attention is high but sentence attention is low (see red arrow).

5 Dataset

Our initial model was trained on summarization of news articles. Our dataset consisted of news stories collected from the CNN and DailyMail. The following is the link of the Github page which hosted the dataset: https://github.com/abisee/cnn-dailymail. It involves downloading and tokenizing the dataset to be fed into our model to get baseline results on the performance of the summarization of the model.

Data-driven summarization of scientific articles involved generating the title of a paper from its abstract (title-gen) or abstract from its full body (body-gen). title-gen was constructed from the MEDLINE dataset, whereas body-gen from the PubMed Open Access Subset. Some statistics on the datasets:

title-gen	*Abstract*	*Title*
Token count	245 ± 54	15 ± 4
Sentence count	14 ± 4	1
Sent. token count	26 ± 14	-
Overlap	$73\% \pm 18\%$	
Repeat	$44\% \pm 11\%$	-
Size (tr/val/test)	$5'000'000/6844/6935$	

Figure 2: Title-gen dataset

abstract-gen	*Body*	*Abstract*
Token count	4600 ± 1987	254 ± 54
Sentence count	172 ± 78	10 ± 3
Sent. token count	26 ± 17	26 ± 14
Overlap	$68\% \pm 10\%$	
Repeat	$74\% \pm 7\%$	$44\% \pm 11\%$
Size (tr/val/test)	$893'835/10'916/10'812$	

Figure 3: Abstract-gen dataset

With the above two datasets in, we then went on to our training process.

6 Implementation

In order to generate our dataset for training our model, we developed our own custom tokenizer and data preprocessor that took sentences from the input. Our input basically consisted of the following :

- Title of the paper
- Abstract or synopsis
- Main body or content of the paper

In order to tokenize our input we made use of the Stanford Core NLP tokenizer to help us create word tokens. We then created training, test and validation sets based on our tokenizer.

For preprocessing our data, we scanned through many scientific papers publicly available and from that took the Titles, Abstracts and main contents of the paper. With the help of this we were able to create training and validation sets as follows:

- When summarizing the Abstract or the synopsis of the data we compared the output against the Title, hence we extracted the input as the Abstract/Synopsis and the expected output as the Title for training.
- When summarizing the main content of the paper, we compared it against the Abstract/Synopsis of the paper, i.e we extracted the main content of the paper as the input and the Abstract/Synopsis as its expected summary.

The following are the list of hyperparameters that we explored during our training:

- max_grad_norm: 2.0,
- max_select_sent: 20,
- max_sent_len: 50,
- max_train_iter: 10000,
- min_dec_steps: 35,
- min_select_sent: 5,

- mode: evalall,
- model: end2end,
- model_max_to_keep: 5,
- rand_unif_init_mag: 0.02,
- select_method: prob,
- selector_loss_wt: 5.0,
- single_pass: True,
- start_eval_rouge: 30000,
- thres: 0.4,
- trunc_norm_init_std: 0.0001,
- vocab_path: data/finished_files/vocab,
- vocab_size: 50000

Below is our custom implementation described in detail:

6.1 Creating Raw Dataset:

- Took scietific paper dataset from https://github.com/ninikolov/data-driven-summarization
- Structure of above data-set: - 1 file for abstracts of all scientific papers, 1 file for titles of all corresponding scientific papers.
- Created 1 file per scientific paper, where each file has abstract and title as highlight of the paper.
- Kept the same ratio for Train, Validate and Test data-set as mentioned in the above github link

6.2 Creating Tokenized Binary formatted data-set [Serialize data-set using tf-Example]

- Created own $custom\_make\_data.py$ to convert above created raw data-set into tokenized binary format.
- Firstly, created placeholder for tokenized data-set, where for each scientific paper there will be one tokenized file and also created a mapping from input file location to output file location. (mapping.txt)
- Now passed this mapping.txt alongwith above created raw data-set to Stanford NLP PTBTokenizer to create tokenized file for each input.
- Now, iterate through each tokenized file and performed some processing like converting to lowercase, mark end of line by explicitly adding period, etc as mentioned in the baseline model and serilaize it using tf.Example to convert it into binary format.
- Lastly, for each category (train, validate, test) , divided the dataset into chunks of 1000 tokenized binary files.

7 Evaluation

We are using **Rouge**(Recall-Oriented Understudy for Gisting Evaluation) which is a recall based metric used for summarization. We can use **Overlap** to then verify if genuinely new content was generated instead of simply copying text. **Repeat** can be used to check the tendency of how much the model has a tendency to repeat words.

Here's a brief overview on ROUGE:
ROUGE is essentially based off of a set of metrics,used for evaluating automatic summarization of texts as well as machine translation. It works by comparing an automatically produced summary or translation against a set of reference summaries.

Recall for ROUGE can be evaluated as follows:

$$\frac{number\_of\_overlapping\_words}{total\_words\_in\_reference\_summary}$$

There're different versions of ROUGE evaluation:

ROUGE-N, ROUGE-S and ROUGE-L can be thought of as the granularity of texts being compared between the system summaries and reference summaries. For example, ROUGE-1 refers to overlap of unigrams between the system summary and reference summary. ROUGE-2 refers to the overlap of bigrams between the system and reference summaries.

For our evaluation, we have stuck to ROUGE-1 in order to proof-check our model,against the baseline.

Here's the analysis of our Results:
(The first 3 evaluations are as per the available models already tested on summarization tasks.)

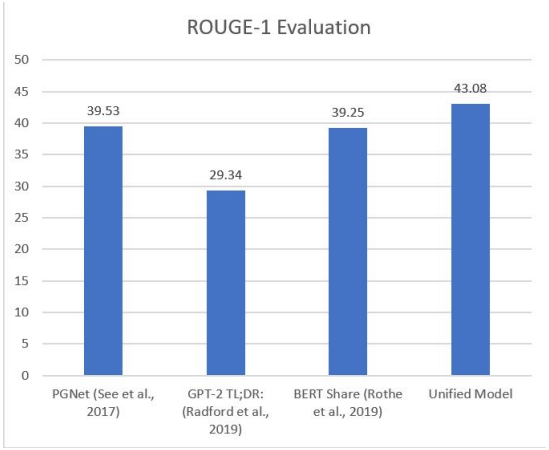

Figure 4: Evaluation of Rouge -1 Scores across different models.

8 Analysis:

We have extracted scientific papers from Pubmed.com taken from github repository. The trained model on CNN/Daily mail dataset was then hypertuned and run on these papers achieving an accuracy of 43.08% . Below is the sample example from our model:

- - **Output from Extractor:** As in our model, extractor behaves like a selector, basically it selects all the relevant sentences which contain most of the important information. In our sample example, the extractor model selected "Diabetic neuropathy is a common complication of diabetes that may be disabling and even contribute to mortality" because the probability for this sentence is quite high. Similarly, other output sentences also have reasonable high sentence attention / probability.
- **Output from Abstractor:** Abstractor behaves like a rewriter in our model, which means it will try to output selected words in a sentence which convey the main context. In our sample example model has re-written the sentence to make some meaning out of it. For example: "Trial definitively established an important role of improved metabolic control in the primary prevention of clinical neuropathy". Here rewriter has used its own vocabulary like Trial definitively to make the output sentence more meaningful.
- **Output from Unified:** This is a combination of both selector and rewriter, which means model takes the sentence level attention as input and combined it with word level attention. For example: "None of these syndromes is pathognomonic for diabetes, and they may occur idiopathically or in association with other disorders in nondiabetic persons." Here model choose this sentence because its sentence level attention is quite high and it has re-written "in association with other disorders" to make the context more clear.

Here is an example document from Medical domain , which we have summarized using the 3 Approaches: Extractive, Abstractive and Unified.

Original document which was summarized

Original :

Diabetic neuropathy is a common complication of diabetes that may be disabling and even contribute to mortality. Diabetic peripheral neuropathy encompasses a group of clinical and subclinical syndromes, each characterized by diffuse or focal damage to peripheral somatic or autonomic nerve fibers. None of these syndromes is pathognomonic for diabetes, and they may occur idiopathically or in association with other disorders in nondiabetic persons. Distal symmetric sensorimotor polyneuropathy is the most common form of peripheral neuropathy and is the leading cause of lower limb amputation. The characteristic slowing of sensory and motor nerve conduction velocities and advancing distal symmetric sensorimotor deficits are ascribed to an underlying insidious, chronically progressive, length-dependent, distal axonopathy of the dying-back type primarily, but not exclusively, affecting sensory nerve fibers. The cumulative prevalence of clinical diabetic neuropathy parallels the degree and duration of antecedent hyperglycemia, and the Diabetes Control and Complications Trial definitively established an important role of improved metabolic control in the primary prevention of clinical neuropathy. Improved blood glucose control substantially reduces the risk of developing diabetic polyneuropathy in type 1 diabetes mellitus, thereby strongly implicating hyperglycemia as the important causative factor in this degenerative disease process. Studies in experimental animal models reveal several glucose-related metabolic mechanisms that could initiate neurochemical, neurotrophic, and/or neurovascular defects culminating in a peripheral sensorimotor and autonomic neuropathy. Other than improved blood glucose control, therapy for diabetic neuropathy remains palliative and supportive, although this is expected to change radically as new insights into the pathogenetic mechanisms of diabetic neuropathy give rise to specific new mechanism-based therapies.(...)

Abstractive and Extractive approaches towards summarization

Abstractive :

neuropathy is a common complication of diabetes. None of these syndromes is pathognomonic for diabetes. Complications Trial definitively established an important role of improved metabolic control in the primary prevention of clinical neuropathy. Improved blood glucose control substantially reduces the risk of developing diabetic polyneuropathy in type 1 diabetes mellitus. neurovascular defects culminating in a peripheral sensorimotor .

Extractive :

Diabetic neuropathy is a common complication of diabetes that may be disabling and even contribute to mortality. None of these syndromes is pathognomonic for diabetes, and they may occur idiopathically or in association with other disorders in nondiabetic persons

Complications Trial definitively established an important role of improved metabolic control in the primary prevention of clinical neuropathy. Improved blood glucose control substantially reduces the risk of developing diabetic polyneuropathy in type 1 diabetes mellitus. Studies in experimental animal models reveal several glucose-related metabolic mechanisms that could initiate neurochemical, neurotrophic, and/or neurovascular defects culminating in a peripheral sensorimotor and autonomic neuropathy

Our Unified approach towards summarization

Unified :

Diabetic neuropathy is a common complication of diabetes that may be disabling and could even lead to death. Diabetic peripheral neuropathy encompasses a group of clinical and subclinical syndromes, each characterized by diffuse or focal damage to peripheral somatic or autonomic nerve fibers. None of these are symptoms for diabetes, and they may occur idiopathically or comorbid with other disorders in nondiabetic persons. Distal symmetric sensorimotor polyneuropathy is the most common form of peripheral neuropathy and is the primary cause of lower limb amputation. The characteristic slowing of sensory and motor nerve conduction velocities and advancing distal symmetric sensorimotor deficiencies are attributed to an underlying insidious, chronically progressive, length-dependent, distal axonopathy of the dying-back type primarily, but not exclusively, affecting sensory nerve fibers. The combined presence of clinical diabetic neuropathy parallels the degree and duration of antecedent hyperglycemia, improved metabolic control in the primary prevention of clinical neuropathy. Improved blood glucose control significantly reduces the risk of developing diabetic polyneuropathy in type 1 diabetes mellitus, thereby strongly implicating hyperglycemia as the important causative factor in this degenerative disease process..

9 Code:

The code repository for this project can be found in the following Github link, with clear instructions posted on README, containing information on setup and list of major software package requirements:

https://github.com/utk1801/Summarization-of-Scientific-Paper-using-Unified-Model

10 Conclusion:

We went through an in-depth research on Unified Modeling approach for Document Summarization. The base paper was very well explained and detailed on their approach. We developed on this baseline model by tuning the hyperparameters , and applied it on scientific papers, taken from PubMed, to discover the unified model being capable of correctly summarizing the paper with an accuracy of 43.08.

Key Learnings:

Our observation is that the model does really well in creating summaries from scientific papers belonging to medical domain which is the data it is trained on. However if presented with publications of other fields that is not medical, it still creates good summaries but those summaries were more extractive in nature.

Future Scope:

- We can try to increase the training dataset size and build the model. The generalization capability of a deep learning model enhances with an increase in the training dataset size.
- Use the beam search strategy for decoding the test sequence instead of using the greedy approach (argmax).
- Evaluate the performance of your model based on the BLEU score.
- Implement pointer-generator networks and coverage mechanisms.

References

[1] Unified Model for text summarization,
https://www.aclweb.org/anthology/P18-1013.pdf

[2] Data-Driven Summarization of scientific papers (Dataset),
https://github.com/ninikolov/data-driven-summarizationl

[3] ROUGE ACL Paper link,
https://www.aclweb.org/anthology/W04-1013.pdf

[4] PubMed dataset link,
https://drive.google.com/drive/folders/17sPutnazCN2MI-7v88KTQ1lndX1-UBGv

[5] CNN/DailyMail Dataset for news stories,
https://github.com/abisee/cnn-dailymail