FAKULTÄT FÜR
INFORMATIK

SEA: Summary Evaluation of Academic Publications with Unsupervised Methods

Team Project

Submission Date : 3^{rd} *May* 2021

Supervisor: Prof. Dr.-Ing. Ernesto William De Luca

Advisor: Sabine Wehnert

Authors:

Sanket Joshi (223751)

Shivani Jadhav (223856)

Abstract

If the summary of a scientific paper can be obtained, then this summary can be used to understand what the paper is all about instead of going through it manually. To validate the correctness of the summarizer and in the efforts to improve it, a ground truth (in our case reference summaries) is of utmost importance. Ground truth is useful to evaluate the generated summaries that might contain paraphrased sentences on different abstraction levels and hence, datasets with only one correct ground truth may only paint a fraction of the whole picture. The lack of robust evaluation methods which takes into consideration this issue, challenges the reliability of current summarization techniques. In this report, in the urge to find an alternative for supervised summary evaluation, we propose two alternative unsupervised methods. First, we propose the "Relative Clustering Comparison Score", consisting of three individual scores for cluster evaluation (Adjusted Rand Index (RCCScore_RI), Mutual Information (RCCScore_MI), and Completeness (RCCScore_CO)). Second, a further method assesses the ranking the sentences we obtain based on cosine similarity, and it is called the ""Relative Minimum Edit Distance"(RMDScore). Further to unveil the most important part in the research paper, each of the research papers from the *AIPubSumm* dataset is summarized using Textrank first by only considering the introduction of each paper and then by considering the entire paper. Initial results from our experiments depict an interesting pattern indicating that authors tend to write the most informative parts in the introduction section. This may be attributed to the fact that often authors tend to summarize the whole work in the introduction section itself.

Contents

Chapter 1

Introduction

Text summarization is a research field with a primary goal of condensing a long document and summarizing it within given bounds in such a way that all the important elements and the semantics from the original document are preserved. There exists two types of text summarization techniques, extractive text summarization [1] and abstractive text summarization. Abstractive text summarization [2] is a technique where a new text is generated which may paraphrase the original text. Extractive text summarization on the other hand selects the most important sentence(s) from the document to use it or them as the summary.

In our work, we focus on extractive text summarization of scientific publications which may have some important information clustered in some parts (e.g, the introduction) due to their common structure. The dataset that we use was offered by ScienceDirect[1] so that we could make use of the author-generated paper highlights it offers. This dataset was proposed by E.Collins et al. [3] and every paper in this dataset has a title, abstract, author-written highlights and author-defined keywords. These author-generated highlights provide a gist of that author's paper and often exhibit a high overlap with a sentence from the original paper content. So, we have also considered these author-generated highlights as ground truth.

We illustrate the anatomy of the publications within those datasets in Figure 1.1. All papers contain an Abstract, an Introduction, Highlights, and further sections.

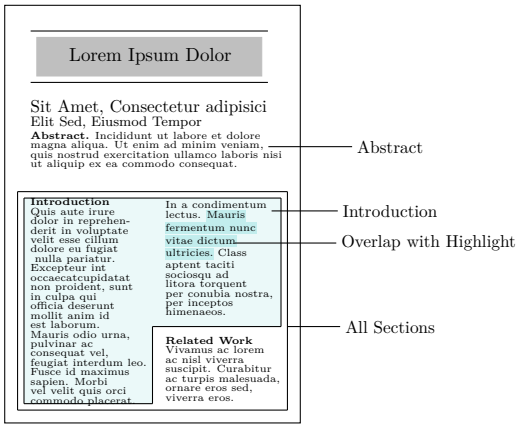

Figure 1.1: Anatomy of the publications in the ScienceDirect datasets.

[1] https://www.sciencedirect.com/

One of the major challenges to obtain good machine-generated summaries is a lack of a robust evaluation approach. The approaches that are widely used for evaluating the generated summaries require a reference summary, such as the aforementioned highlights. However, there are a lot of documents whose reference summary is not available or not the only valid option. There can be many good candidates for a generated summary, so that there is a need to evaluate the generated summaries in a more generalised manner. Hence, for evaluation, along with the existing evaluation approaches for summarizers, we are evaluating scoring techniques which do not require the reference summaries, i.e, completely unsupervised.

Research Questions

1. Can summaries be evaluated in an unsupervised way without any requirement of reference summaries?

2. Is the most informative part of a research paper its introduction section?

This chapter is followed by the chapter, Related work where we not only describe the existing work but also show it's relation with our contribution. A chapter is then dedicated to the key Concept's description and Implementation details. This chapter is followed by an in depth explanation of the approaches, which is then followed by the Evaluation chapter. Each chapter is subdivided with respect to RQ1 and RQ2. At last, in the conclusion we summarize our findings.

Chapter 2

Related work

A basic sentence scoring technique using Term Frequency - Inverse Document Frequency (TF-IDF) after applying a standard pipeline for preprocessing on the text was proposed by Vodolazova et al. [4]. They mention that certain features should be given more weight, such as upper-cased words or font style changes (e.g., bold, italic). Given our corpus of research papers, the impact of the words in the title and of keywords can also be amplified. In this work, we employ this sentence-level TF-IDF approach.

In 2004, Mihalcea and Tarau [5] proposed an approach to extract sentences for the summary using the TextRank approach. In their work, TextRank was used to perform firstly keyword extraction and then sentence extraction which can be then used for text summarization.

A graph based sentence scoring method name LexRank was introduced by Erkan, G. and Radev, D. R. in 2011 [6]. The idea behind technique was to use eigen vector centrality and finding most important sentence when the sentences are represented in a graph format.

A fair amount of work has also been done on the extractive text summarization of scientific papers. E.Collins et al. [3] have introduced a dataset of scientific papers with 10,000 publications in the training set and 150 papers in the test set, which they named *CSPubSumm*. Moreover, they have built a deep learning architecture, SAFNet, which applies a Long Short-Term Memory Network (LSTM) as a sentence encoder in addition to an abstract vector and common features (e.g., the title score measuring the overlap between the title and the reference sentence). While such task-specific deep learning models generally require ground truth for training, we investigate in this work whether we could still evaluate their performance using unsupervised metrics based on similarity. We use *CSPubSumm* dataset to run our experiments.

A regression-based approach with n-gram overlap features was introduced by L. Cagliero and M. La Quatra [7].They also worked on text summarization of scientific articles. They compare their approach with different existing supervised and unsupervised approaches of summary generation on an existing dataset*CSPubSumm* by E.Collins et al. [3] and also on their own datsets, among which we find *AIPubSumm* for our experiments.

Several different evaluation metrics for extractive text summarization have been presented and exploited during the years. ROUGE is the most commonly used metric to compare the generated summaries with the reference summaries, taking into consideration the exact words [8]. There are variants of ROUGE, namely ROUGE-n, ROUGE-l, ROUGE-w, and

ROUGE-s. Word Mover's Distance, based on the idea that distances between the word vectors are to some extent semantically meaningful was proposed by Kusner et al. [9]. Recently, the scoring metric BERTScore was proposed [10]; it exploits the contextualised word embeddings and is based on computing cosine similarity.

In 2003, Ahmad et al. introduced the approach of evaluating the summaries by automatically categorizing them together with the full text into clusters and plotting them on a 2Dmap [11]. The idea is that if the summaries and the full text are assigned to the same position on the map, then the summary can be considered as a good representation of the whole text. One of our proposed evaluation techniques, "Relative Clustering Comparison Score", is inspired from this categorization approach. However, unlike our adaptation, Ahmad et al. exploit the labels to get the final score and check whether the generated summaries also predict the same labels as their respective content. They trained the Self-Organizing Map(SOM) using the keywords which represented the articles that were automatically derived from the dataset which acted like the labels and then tested on the remaining articles. The goodness was evaluated by checking whether the summaries (that can comprise of presence or absence of the salient terms) are assigned to the same position as the full text then the generated summaries are good. Minimum edit distance (or word error rate) [12] has been used to evaluate the generated summaries either by calculating the edit distance at a word level (TER) [13], stem level (ITER) [14] or even at the character level (CHARACTER) [15], (EED) [16], requiring a reference summary. Our second proposed evaluation method, "Relative Ranking Comparison Score", is based on the minimum edit distance, but without any need of a reference summary: we get the minimum edit distance by comparing the rankings of the content of a document and rankings of the generated summary of that document.

Chapter 3

Concept and Implementation

3.1 Concept

After setting our scope in relation to the related work, we continue with fundamental concepts. We also describe some concepts which are a part of our proposed approach for evaluation. We first describe the overall workflow, followed by various extractive text summarization techniques and evaluation metrics. We then describe the proposed unsupervised evaluation approach.

Workflow

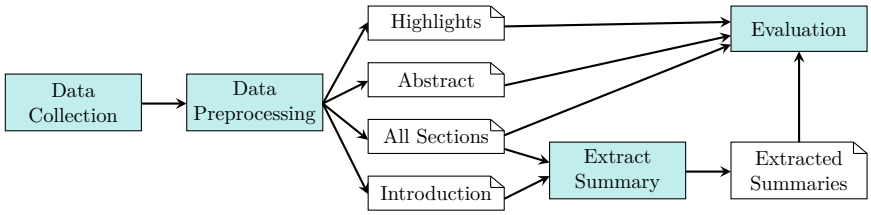

Figure 3.1: Extractive text summarization workflow

Figure 3.1 depicts the general workflow leading to the unsupervised evaluation without using the highlights, which first starts with data collection (.xml files which are parsed using **ElemenTree**) and data preprocessing with feature selection. Then the entire document is converted into various sections according to the anatomy. All Sections and in addition also only the Introduction of the paper are passed to the extraction phase. Then the extracted summary along with the abstract and highlights are passed on to the evaluation phase.

Extractive text summarisation techniques

1. Sentence level TF-IDF

   Our first approach for computing the sentence score is using TF-IDF on the sentence level. Each sentence is considered as one document and the rest of the paper as a whole corpus. The TF-IDF scores for all words in a sentence are summed up and normalized

according to the length of the sentence. Jones [17] has used a keyword-based score as an additional feature for scoring the sentence with the idea that if a sentence has more keywords then that sentence is comparatively more important than other sentences. We also adapted this idea and used the keywords from the keywords section. If the word is a keyword then we assign a higher weight. The idea of using keywords can also be extended to using the words from the title of the paper.

2. TextRank

   The second approach is TextRank [5]. The concept of Textrank is adapted from the concept of the Google PageRank algorithm which was used to rank the webpages. The idea behind TextRank is that it assumes the rank of a sentence in a document is dependent on the importance of a sentence suggested by other sentences in terms of links. As TextRank is a graph-based approach, each sentence in the document is considered as one node, and the sentences of the graph and the weights are calculated using the similarity between the sentences and a weighted edge is formed between the sentences. We apply the PageRank algorithm on the weighted graph and is run until it converges. The scores for each node are obtained and as each node corresponds to one sentence, we get a score for each sentence.

3. LexRank

   LexRank [6] is another approach based on Eigenvector centrality in graphs. The idea behind computing Eigenvector Degree Centrality is that every link to and from a node is considered as a vote that determines the overall value of the node. A sentence that has a connection to a high-scoring Eigenvector centrality score will contribute or in other words, is more important than a sentence that has a connection to a low-scoring Eigenvector centrality score. Basically, it calculates the impact each node has on the graph. We use LexRank to find the most central sentences that can be included in the summary.

4. Ensemble method

   For generating the summaries, we came up with one more approach that combines the score for each sentence given by the TF-IDF method and the score given by the TextRank method. In the ensemble model we used TextRank which is strengthened using a pretrained embedding. We add both the scores and we get a cumulative score for every sentence, which is afterwards normalized.

One of the most important aspects of research is the evaluation. After getting the summaries of the documents, we need to check how accurate the obtained summaries are. Considering our documents are research papers, we can use the abstract part of the research paper as a summary in itself. However, we are not going to consider abstract as the reference summary for evaluation. Also, as mentioned previously we have highlights which can have significant overlaps with sentences in the research paper. Hence, we generate the summaries using the approaches mentioned above, we will evaluate the generated summaries by using highlights as the reference summaries. The various evaluation approaches are given below:

Extractive Summary Evaluation

1. ROUGE

   For extractive text summarisation, ROUGE [18] is most relevant. n-gram recall between the generated summary and target summary is ROUGE-n. If n=1, it basically counts the number of common words in both generated and target summaries and calculates the recall based on that. There are various variants of ROUGE namely,ROUGE-n, ROUGE-l, ROUGE-w, and ROUGE-s. In our case we only calculate the ROUGE-l generated summary using the Highlight sentences. However, ROUGE being an n-gram based approach does not take into account the semantic dependencies between the words. Hence, we cannot totally rely on this bag of words based approach.

2. Word Mover's Distance

   We have used Word Mover's Distance to calculate the distance between the words from the generated summary and the ground truth which in our case we consider as the highlights section. We follow the same concept by using Word Mover's Distance that distances between the word vectors are to some extent semantically meaningful [9]. We compare the scores when the summary is generated only from the Introduction with highlights as the ground truth and when the summary the summary is generated from the all sections of the paper with highlights as the ground truth. We have used the pre-trained Google News Model embeddings for getting the vector representation for both the generated summary and the ground truth.

3. BERTScore

   The BERTScore [10] applies recent advancements in distributional semantics to evaluate machine-generated text. Using contextual word embeddings from a BERT model, each token of a reference sentence is encoded and compared to each token of a candidate sentence. For each token of the reference sentence, we compute the IDF-score and multiply it by the highest cosine similarity with a token from the candidate sentence. This is divided by the product of all IDF-scores. As a result, we obtain a metric with the same range of -1 to 1 as the cosine similarity, which can be optionally rescaled for the interval between 0 and 1.

Proposed Unsupervised Evaluation Approach

One of the challenges with the summarization task is that there can be various candidate summaries for the same original content. Hence, we need something which would not just evaluate the extracted summary with respect to a given set of finite reference summary sentences. We are proposing two methods called "Relative Clustering Comparison Score" and "Relative Ranking comparison Score" which are based on the hypothesis:

If the syntactic and semantic similarity relations among extracted summaries are preserved in the same way as the syntactic and semantic similarity relations among the documents, it can be stated that the extracted summaries are capturing the gist of the documents in a proper way.

1. Clustering-based Evaluation

   Aligning to the proposed hypothesis, this approach is based on the assumption that if the candidate summaries form the same clusters as the summaries drawn from all sections (excluding the abstract and author highlights) then the extracted summaries are good."

   Figure 3.2 describes the clustering-based evaluation. First we need to generate the summary for all the research papers. We vectorized extracted summary and all sections and then performed clustering. This is done to evaluate the extracted summaries. We pass them pairwise so that we get the optimal number of clusters for the entire pair.

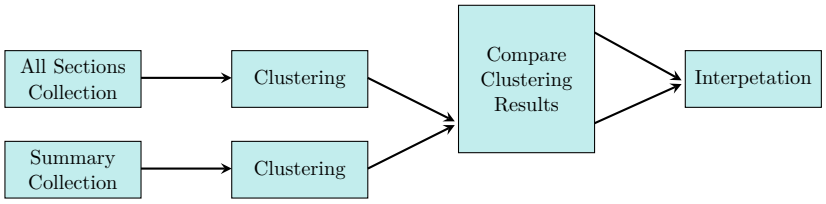

Figure 3.2: Summary evaluation process via clustering assessment

2. Relative Minimum Edit Distance

   Figure 3.3 describes this approach which is based on the assumption that if the similarity ranking obtained for a candidate summary of a document is the same as the similarity ranking obtained by considering all sections of the document, the extracted summary is considered to be good.

   For a document from all sections, a ranking is given to all other documents based on the cosine similarity. Then we obtain the ranking for the vectorized extracted summary. The rankings are then compared using the minimum edit distance. An average minimum edit distance score is obtained. This score is converted it to a similarity score:

   $$\text{WMDScore} = 1 - AvgMinimumEditDistance$$

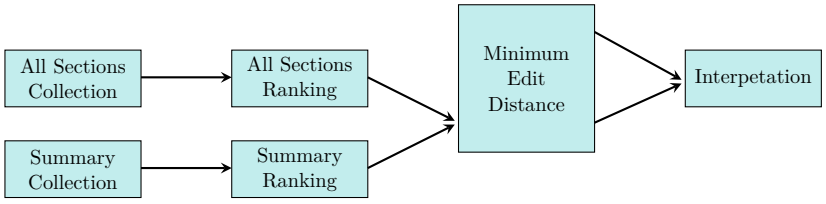

Figure 3.3: Summary evaluation process via ranking assessment

Proving the hypothesis

To prove the hypothesis we used only the *AIPubSumm dataset*. The extracted summaries are evaluated using the proposed approaches "Relative Minimum Edit Distance" and the

"Relative Clustering Comparison Score". The obtained scores are compared with the metrics that use a supervised setting, ROUGE-l and BERTScore.

3.2 Implementation

Vectorisation

Before clustering or ranking, the text needs to be converted to a numerical representation. We use TF-IDF, Pretrained word embeddings, Pretrained word embeddings with fine tuning as the vectorisation techniques.

K-means

The kmeans algortithm and PCA followed by kmeans(PCA-kmeans). We first need to set the number of clusters. To choose this K we will need to use the elbow plot and then apply kmeans with number of clusters as K on the abstract part and assign labels to the documents based on the cluster number they are appearing in. We will then do the same thing for all sections part, highlight part and extracted summary part. This is to first prove our hypothesis and after it is proven then we can use it to evaluate the extracted summaries. We can then plot the cluster number along with the document ids and try to visualise if the hypothesis holds true. However, we just could not prove something based on mere visualisation and we wanted a number which will tells us whether the results are acceptable or no. So, we set the cluster labels for the all section as the ground truth and use Rand Index, Mutual Information and Completeness. These measures are described in the following block.

Cluster comparison

"The Rand Index computes a similarity measure between two clusterings by considering all pairs of samples and counting pairs that are assigned in the same or different clusters in the predicted and true clusterings." Mutual Information score measures the information between the two clusterings. Completeness score measures how many samples in a given cluster also belong to the same class. The clustering can be said complete if all the samples that belong to same class also are the elements of the same cluster.

Now that we have described all the concepts and implementations of this project, we move towards the actual implementation for answering the two research questions 1.

Implementation for RQ1

After generating the summaries, we proceed with proving the hypothesis. We first evaluate all the generated summaries using supervised evaluation methods taking into account the generated summaries as the candidate summaries and author-written highlights as the reference summaries. We calculate the BERTScore and Word mover's distance in the similar way. Further, we perform clustering and calculate all the cluster comparison and relative minimum edit distance scores from the unsupervised evaluation methods by varying the number of sentences to be extracted as summary sentences. We then try to look for a pattern that

correlates the scores from the proposed unsupervised evaluation metrics with the existing supervised evaluation metrics.

Implementation for RQ2

Figure 3.4 describes the entire workflow to solve RQ2. It consists of 3 pipelines. In pipeline 1, summaries are extracted using the research papers using TextRank [5] using only the introduction (IntroSumm) and using all sections (AllSecSumm). In pipeline 2, evaluation of the two summaries is done with the author-highlights as reference summaries using ROUGE-l and BERTScore. In pipeline3, all sections excluding abstract and highlights are vectorised using TF-IDF vectoriser and used as features to cluster the papers using K-means to form the first set of clusters. Second set of clusters for the papers are obtained using the TF-IDF vectors of AllSecSumm and Third set using the TF-IDF vectors of IntroSumm. RI AllSec is the Rand Index score obtained by comparing first set and second set of clusters and RI Intro is the Rand Index score that is calculated using first set and third set of clusters. We have used *AIPubSumm* [7] for our experiments.

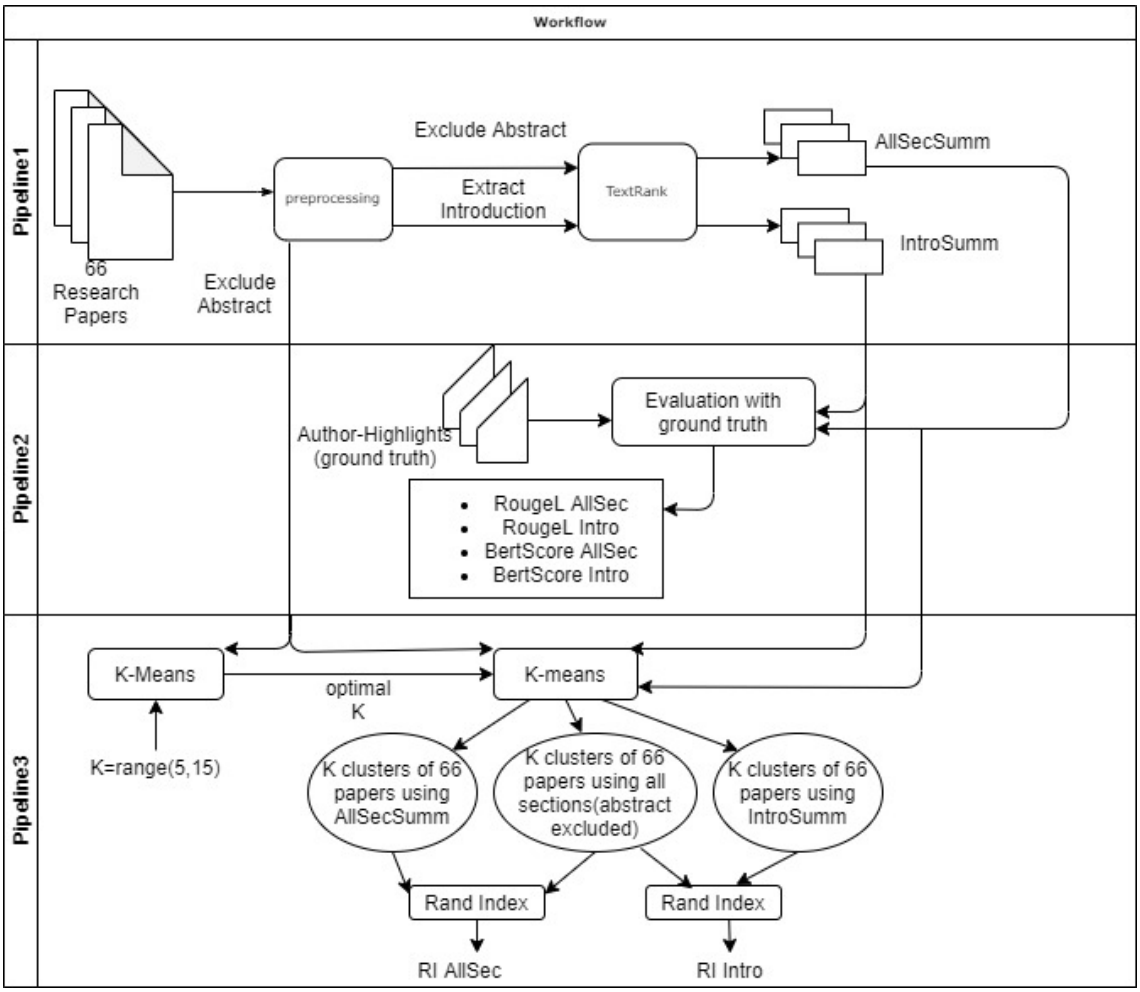

Figure 3.4: RQ2 workflow

Chapter 4

Evaluation

4.1 Experimental Setup

We discuss the experimental setup in three parts: data collection and preprocessing, summaries extraction, and generated summaries evaluation.

4.1.1 Data collection and preprocessing

The experiments were run on two datasets which consist of scientific papers. In addition to all the usual sections of a research paper, along with the abstract, both the datasets have highlights which consist of important sentences of that paper according to its author. After analysing the datasets, it can be concluded that the highlights are not always exactly extracted from the paper but can also be paraphrased by the author. By performing crawling on the ScienceDirect website, as done by Collins et al. [3] and Cagliero et al. [7], the dataset is created by separating out Highlights, Abstract, Introduction and one with all sections of a paper (excluding Abstract and Highlights) for all the papers. For AIPubSumm, one paper was not considered as it has been redirected. So the experiment was performed with 65 papers from AIPubSumm and 150 papers from CSPubSumm. All the text was preprocessed by first doing tokenization, followed by stopwords removal and lemmatization. The domain-knowledge based experiments were done using a subset of AIPubSumm dataset. As we are dealing with extractive text summarization, similar to the authors of these paper, highlights sentences can be considered as the ground truth.

4.1.2 Summaries extraction

The summaries are generated first from all the sections of the paper and then only from the introduction part of the paper using unsupervised approaches. The number of sentences for the generated summaries are varied from 1 to 10 for evaluation purpose. The extractive text summarization techniques that are used are LexRank, TextRank, Sentence-level TF-IDF, TextRank with Glove Embeddings and an Ensemble of Sentence-level TF-IDF and TextRank.

4.2 Results and Discussions

4.2.1 RQ1

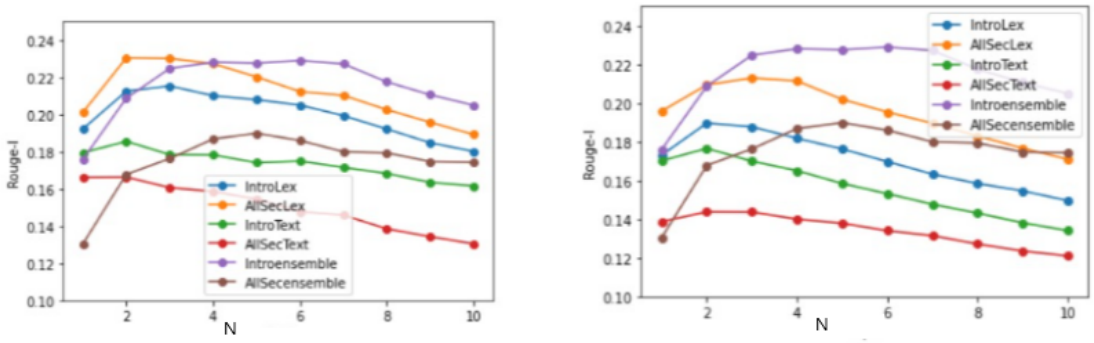

Figure 4.1: (a) AIPubSumm (b) CSPubSumm

The graphs in Figure 4.1 show that the ensemble method which we are proposing gives the highest ROUGE score when the number of sentences are greater than 4. After doing the data profiling, we found that there are total 263 highlights from the 65 papers for *AIPubSumm* and 616 highlights from 150 papers for *CSPubSumm*. The average number of sentences present in the highlights section is 4.04 and 4.1 repectively. Therefore, a good ROUGE score when N=4 can be considered significant. Hence, we use Ensemble method of summarization using only introduction section to prove our proposed hypothesis.

The following tables 4.1 and 4.2 try to correlate all the metrics that are used to evaluate the generated summaries. The scores using the evaluation metrics, ROUGE-l, BERTScore, WMDScore, Relative Minimum Edit Distance Score(RMDScore), Relative clustering comparison score with Rand Index (RCCScore_RI), Relative clustering comparison score with Mutual Information (RCCScore_MI) and Relative clustering comparison score with Completeness (RCCScore_CO) of generated summaries with varying number of sentences (4,8,10) in the generated summaries are given.

	4	8	10
ROUGE-l	0.212	0.183	0.171
BertScore	0.835	0.823	0.819
WMDScore	0.715	0.716	0.717
RMDScore	0.00685	0.00686	0.0069
RCCScore_RI	0.125	0.169	0.276
RCCScore_MI	0.228	0.311	0.383
RCCScore_CO	0.436	0.487	0.569

Table 4.1: CSPubSumm scores

	4	8	10
ROUGE-l	0.227	0.202	0.189
BertScore	0.843	0.836	0.832
WMDScore	0.713	0.711	0.700
RMDScore	0.0161	0.0164	0.0165
RCCScore_RI	0.167	0.197	0.335
RCCScore_MI	0.264	0.286	0.352
RCCScore_CO	0.446	0.468	0.547

Table 4.2: AIPubSumm scores

In the above tables we have compared the scores we get from both supervised and unsupervised evaluation metrics. We can observe a pattern that we get the best ROUGE-l score when we extract 4 sentences because the average number os sentences present in highlights are 4 and as we increase the number of sentences, the ROUGE score decreases. The pattern correlates with the BERTScore. The RCC scores for RI, MI and CO correlate with the WMD Score and RMD Score because we see a pattern of increasing scores as we increase the number of sentences. This gives rise to a question whether we can trust the supervised metric to validate our work because the three supervised metrics that we used ROUGE, BERT and WMD do not go hand in hand.

Other Findings

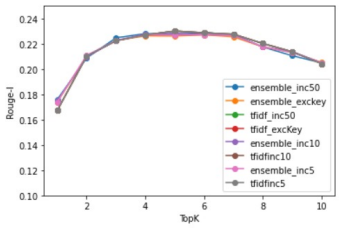

Figure 4.2: ROUGE scores after including keywords

We have addressed one of our research questions that was related to check the impact on ROUGE score after the keywords were included by analyzing the above graph. In the above graph, we have compared the ROUGE scores we get from different techniques on the AIPubSumm dataset. We also check whether including or excluding the keywords made any significant contribution to the generated summaries. For this, we just extract the keywords from the keywords section and if the keyword is present in the sentence then we give it a higher weight. We observed that the inclusion of the keywords did affect the scores, but not to the extent that we could see a major change. After investigating the reason behind this we got to know that half of the files from our dataset had less than 5 keywords in the introduction section. We have plotted a histogram of the keyword count and we can see clearly see the reason behind keywords not contributing to the score. Inclusion of keywords from the papers did not have any significant change in the overall performance of summary generation technique due to a few obvious reasons like keywords may be not be present in their exact form

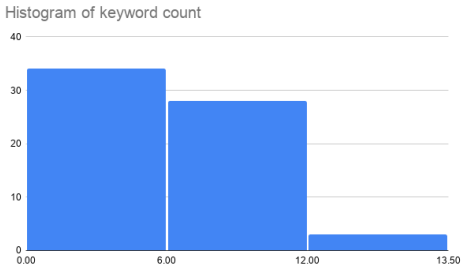

Figure 4.3: Introduction

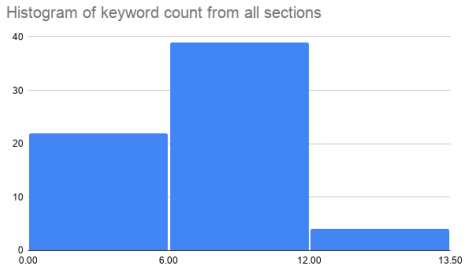

Figure 4.4: All sections

in the content of the paper, a particular keyword can be referred to by using it's synonym or by a pronoun. We can improve this by adding words in the title, Bold and Italic words along with their synonyms to this keyword list.

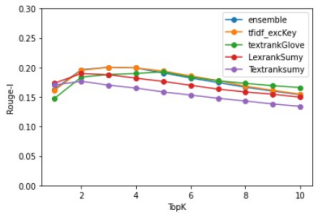

Figure 4.5: CSPubSumm Intro

The above graph also shows the ROUGE-l scores that we got by summarizing the sentences using TextRank from the **sumy** package and TextRank that we use from the **networkx** package and apply Glove embeddings to the sentence vectors. We can clearly see that after using the embeddings we get a better ROUGE score

4.2.2 RQ2

We only considered *AIPubSumm* dataset to conduct experiments to answer RQ2.

Table 4.3: Evaluation without Ground Truth (higher values are better)

Metric \N	**4**	**8**	**10**
RI-Intro	**0.2123**	**0.3383**	**0.3834**
RI-AllSec	0.1195	0.3348	0.2609

Table 4.4: Evaluation with Ground Truth (higher values are better)

Metric \N	**4**	**8**	**10**
ROUGEL-Intro	**0.1704**	**0.1683**	**0.1615**
ROUGEL-AllSec	0.1544	0.1385	0.1304
BERTScore-Intro	**0.8368**	**0.8334**	**0.8315**
BERTScore-AllSec	0.8245	0.8206	0.8185

To demonstrate the merits of generating summaries using the Introduction part, we compare the results for different values of number of sentences N (4,8,10) that are extracted as summaries. Based on the scores shown in Table 4.3, the Rand Index score of IntroSumm clearly outperforms the Rand Index score of AllSecSumm. The Evaluation techniques using reference summaries as shown in Table 4.4 also give the same result.

Initial empirical evidence indicate an interesting trend whereby the quality of the generated summaries is better when we only use the introduction section with much less data compared to the entire paper. This raises a question - "do authors tend to write the most informative parts in the introduction section?". The findings may be due to the data itself. To explore it further we did the following analysis which is shown in the Figure 4.6

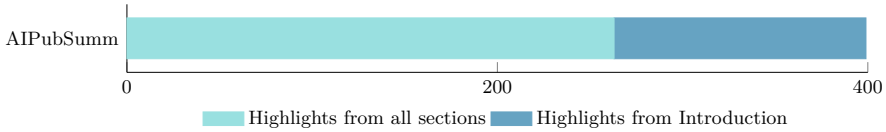

Figure 4.6: Origin of the Highlights from both ScienceDirect datasets.

We find that only $1/3^{rd}$ amount of the highlights is generated from the introduction part of the research papers in *AIPubSumm* and hence cannot be considered as one of the reasons for getting good scores when only introduction part of the paper is considered. We argue that such findings may be useful since we can achieve better performance by using less data.

Chapter 5

Conclusion and Future work

In this work, we proposed the unsupervised evaluation approaches and we tried to validate those approaches with the current supervised evaluation approaches, but we observe that more hyperparameter tuning would yield better results. We can also explore clustering methods other than Kmeans and then proceed with further analysis.

Further, the hypothesis can be proven by using the abstract of the papers. Similar to what was done with the extracted summaries, the abstract of the papers can evaluated using the proposed approaches, "Relative Minimum Edit Distance" and the "Relative Clustering Comparison Score". The score obtained should be the minimum that should be achieved by the generated summaries for them to be considered good, assuming that abstract of a paper is supposed to be the best summary. It is also worth noting that findings obtained from RQ2 may be useful since we can achieve better performance by using less data.

Also, often datasets with only one correct ground truth may only paint a fraction of the whole picture. This may be of primal importance for the community for research that attempt to learn supervised models exploiting ground truth. How true is the ground truth? Is there a single version of truth or there is room for equally convincing alternatives? In our case also, there can be multiple candidate summaries and relying on just one ground truth may not capture the overall performance. To tackle this, a user study can be conducted where the domain experts mark the summaries as relevant or irrelevant and we can compare the scores with the ROUGE-l F1 scores from the papers.

Bibliography

[1] V. Gupta and G. S. Lehal, "A survey of text summarization extractive techniques," *Journal of emerging technologies in web intelligence*, vol. 2, no. 3, pp. 258–268, 2010.

[2] R. Nallapati and B. Xiang, "Abstractive Text Summarization using Sequence-to-sequence RNNs and Beyond Cicero dos Santos," pp. 280–290, 2016.

[3] E. Collins *et al.*, "A supervised approach to extractive summarisation of scientific papers," *arXiv preprint arXiv:1706.03946*, 2017.

[4] T. Vodolazova *et al.*, "The role of statistical and semantic features in single-document extractive summarization," 2013.

[5] R. Mihalcea and P. Tarau, "Textrank: Bringing order into text," in *Proceedings of the 2004 conference on empirical methods in natural language processing*, pp. 404–411, 2004.

[6] G. Erkan and D. R. Radev, "Lexrank: Graph-based lexical centrality as salience in text summarization," *Journal of artificial intelligence research*, vol. 22, pp. 457–479, 2004.

[7] L. Cagliero and M. La Quatra, "Extracting highlights of scientific articles: A supervised summarization approach," *Expert Systems with Applications*, vol. 160, p. 113659, 2020.

[8] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," in *Text summarization branches out*, pp. 74–81, 2004.

[9] M. Kusner *et al.*, "From word embeddings to document distances," *Proceedings of the 32nd International Conference on Machine Learning (ICML 2015)*, pp. 957–966, 01 2015.

[10] T. Zhang *et al.*, "Bertscore: Evaluating text generation with bert," *arXiv preprint arXiv:1904.09675*, 2019.

[11] K. Ahmad *et al.*, "Summary evaluation and text categorization," in *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pp. 443–444, 2003.

[12] V. I. Levenshtein, "Binary codes capable of correcting deletions, insertions, and reversals," in *Soviet physics doklady*, vol. 10, pp. 707–710, 1966.

[13] M. Snover *et al.*, "A study of translation edit rate with targeted human annotation," in *Proceedings of association for machine translation in the Americas*, vol. 200, Cambridge, MA, 2006.

[14] J. Panja and S. K. Naskar, "Iter: Improving translation edit rate through optimizable edit costs," in *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, pp. 746–750, 2018.

[15] W. Wang *et al.*, "Character: Translation edit rate on character level," in *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers*, pp. 505–510, 2016.

[16] P. Stanchev *et al.*, "Eed: Extended edit distance measure for machine translation," in *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pp. 514–520, 2019.

[17] K. S. Jones, "Automatic summarising: The state of the art," *Information Processing & Management*, vol. 43, no. 6, pp. 1449–1481, 2007.

[18] C.-y. Lin and M. Rey, "ROUGE : A Package for Automatic Evaluation of Summaries," 2001.