Query-focused Extractive Summarization using pre-trained models

Fatemeh Rahimi and Juan Antonio Ramirez-Orta

Dalhousie University

March 5, 2020

Abstract

In the process of writing a research paper, researchers often spend a lot of time organizing and summarizing previous work related to the research. To help with this problem, the proposal of this project is to use Query-focused Extractive Summarization algorithms to produce relevant highlights in related research. For this project, the problem of insufficient labeled data was solved by using pre-trained models such as BERT and BioBERT to produce accurate representations of words. To measure the validity of the approaches, they were applied to the BioASQ dataset of medical articles and obtained results consistent with each other using Cosine similarity and Euclidean distance, each with several pre-trained models. One of the challenges of this project was that producing the embeddings of a lot of sentences with a pre-trained model is a very time-consuming task, so a scalable tool was developed to efficiently compute token embeddings for a variety of pre-trained models.

1 Introduction

Writing the related work part of a research paper is time consuming since researchers have to read whole papers to find the parts that are related to their research. A system that automatically highlights the most related parts to their research would save a lot of time and energy for the scientists.

A solution to this problem can be addressed using summarization techniques. Automatic Summarization is the task of finding the most salient information in a document or set of documents. There are several divisions of Automatic Summarization, one of which is Extractive Summarization. Extractive Summarization works by identifying and retrieving the most important sentences in a document, which is similar to highlighting the most important parts while reading it.

One related variation of Extractive Summarization aligned with this problem statement is Query-focused Summarization. A Query-focused Summarization

System produces an automatic summary of the documents which is specifically related to a specific content requested by a user (which is called the query). One of the most used techniques to select the most relevant sentences is ranking algorithms. The purpose of such is to select the top-ranked sentences to include in the returned summary using a similarity metric such as Cosine similarity or Euclidean distance.

Pre-trained models have proven to successfully achieve state-of-the-art results in different Natural Language Processing tasks. One of the most groudbreaking pre-trained models was BERT (Devlin et al., 2019), from the Google AI Language team. The use of different variations of BERT, such as BERT-Base or BioBERT (Lee et al., 2019) is a proper way to find meaningful representations for the sentences within documents. These representations are often referred to as embeddings, which are useful when using ranking algorithms.

Inspired by the scoring techniques, a variety of experiments on the BioASQ dataset were performed to validate the proposed ranking approaches to help researchers when writing a literature review for their researches. BioASQ is a dataset consisting of PubMed scientific articles that reflects the nature of the proposed problem.

Producing the embeddings from pre-trained models can be quite time consuming due to the size of the models. To speed up the process, a scalable tool was developed to parallelly and efficiently produce and compute the token embeddings from the pre-trained models.

2 Related Work

Research on Multi-document summarization, which is the task of summarizing a set of documents, has been brought to the research spotlight in recent years, focusing in Extractive Summarization. Lecture Summarizer (Miller, 2019) is an example of such approaches, which uses BERT and K-means clustering to identify the closest sentences to the document centroid to include them in the summary.

Both supervised and unsupervised ways of extracting a query-focused summarization have been explored by researchers.

As an example of the supervised approaches, BERTSUM proposed a binary classification approach by adding an additional layer on top of BERT (Liu, 2019). Another supervised study on query-focused summarizaiton was established by applying automatic annotation techniques (Chali and Hasan, 2012).

In the unsupervised approaches, recent methods tried approaching this problem with Ranking and Clustering algorithms. The Ranking Algorithms or Scoring techniques are used to assign a score to sentences of how much they are related to the given query. A comparative study was accomplished on 15 different Scoring techniques in the literature (Ferreira et al., 2013). Ranking SVM addressed query-focused summarization by combining different features of a sentence (Shen and Li, 2011). Some papers involved using Reinforcement Learning, such as the one in (Narayan et al., 2018) and BanditSum (Dong et al., 2019).

3 Problem Definition and Methodology

3.1 Problem definition

Organizing the references of a research is difficult because researchers have to read whole papers to find the sentences related to their research. This means that their capacity to manage their references is limited by their reading speed, availability of the reading resources and their ability to find the documents with the right content. Given the explosion of scientific literature with the advent of Internet, this task is an ever-increasingly difficult endeavor crucial for the timely usage of the existing information.

To guide the scientists in speeding up this process, this project proposes a system that uses Query-focused Extractive Summarization algorithms to highlight or extract the most relevant sentences in researcher's selected related papers, given that this paper is related to the researcher's problem that they want to solve. The problem definition can be identified as a query to this system while the related papers are the documents to summarize.

Using this approach requires a good dataset of scientific papers and their references with the most relevant sentences highlighted, a way of obtaining sentence embeddings for all the sentences present in the dataset, a method to select the most relevant sentences in the summary, and an objective metric to automatically evaluate the sentences highlighted by the algorithm.

3.2 Methodology

3.2.1 Query-focused Extractive Summarization

Finding an summary which is related specifically to a question or query is called Query-focuses Extractive Summarization. Query-focused Extractive Summarization can both be addressed with supervised or unsupervised learning approaches.

In the supervised setting, a corpus of documents and their extracted summaries is needed. In this corpus, all the sentences are labeled as present or not in the golden summary and this label is the target variable to train a binary classifier, which can then be used to select the sentences that will comprise the summary of new documents.

On the other hand, there is a number of unsupervised techniques useful for the task: ranking, clustering and graph-based algorithms can all be used to identify the most important sentences of the documents, which can then be filtered to form a summary related to the user query.

3.2.2 Ranking algorithms for Query-focused Extractive Summarization

The hypothesis for this approach is that when a sentence of the document is related to the query, there is a good similarity metric between the embedding of the sentence and the embedding of the query. Examples of this metrics are

Cosine Similarity, Dot Product and Euclidean Distance. After computing the similarity metric between each sentence and the query, the sentences that are more aligned with the query in terms of content are found by sorting the values of this metric.

3.2.3 Pre-trained models

Pre-trained language models that have already learned good representations using the context of a language can be used in different Natural Language Processing applications, such as Question Answering, Sentiment Analysis, and in this case, Summarization. The representations that these models provide are vectors of numbers which are called embeddings. A breakthrough in language pre-trained models was BERT(Devlin et al., 2019) which produced token embeddings for the tokens in a given sentence. In this project, the pre-trained models used were BERT-Tiny, BERT-Base and BioBERT.

These three models are variations of the same foundational architecture. BERT-Tiny and BERT-Base are general domain language models, while BioBERT is a domain-specific language presentation model which has been trained on large-scale biomedical corpora. Some details of these models are shown in the Table 1. There are different architectures of BioBERT available in the literature (Lee et al., 2019), throughout this project only the BioBERT-Base architecture was employed.

Model	Layers	Hidden Units
BERT-Tiny	2	128
BERT-Base	12	768
BioBERT	12	768

Table 1: Architectural details of the different pre-trained models used in this project. All the layers have same number of hidden units while they are trained on different corpora. BERT-Tiny and BERT-Base are trained on Wikipedia and book corpus, while BioBERT is trained on PubMed articles.

3.2.4 Embedding of a sentence

There are different ways of to represent a sentence. The most common approach is to calculate the average or summation of all the embeddings of the tokens within a sentence. A more simple, ad-hoc approach is to use the [CLS] token produced by BERT, present at the beginning of each sentence before feeding it into the pre-trained model. Sentence-BERT(Reimers and Gurevych, 2019) is another variation of BERT which is introduces different variations of pooling layers on top of the architecture to produce better sentence embeddings. In this work, because of time constraints, it was decided to use the embedding of the [CLS] token as the representative for the sentence.

3.2.5 Evaluation of the summary produced

To evaluate the quality of the summaries produced by the different approaches, a metric called ROUGE was used, which stands for Recall Oriented Understudy for Gisting Evaluation and was introduced in (Lin, 2004). ROUGE has several flavors, but what all of them do is to compare the set of n-grams of the proposed summary with the set of n-grams of the "golden summary" (with same n as for the proposed summary). After comparing these two sets, a Precision-like and Recall-like metrics are computed. A better suited variation for summarization is ROUGE-L, which measures the longest common subsequence of tokens between the intended summary and the reference summary.

3.2.6 Overall process

The system is composed of the following three phases: embeddings production, summary production and summary evaluation.

In the first phase, the objective is to produce accurate mathematical representations of all the sentences present both in the query and the documents. In order to do this, the system relies on pre-trained language models to produce context-dependent embeddings.

In the second phase, the goal is to use the embeddings produced in the first phase to create a summary of the documents. To create such a summary, the proposed system uses ranking and clustering methods. Several strategies and hyperparameters for both approaches can be easily tried in this phase.

In the last phase, the aim is to decide if the summary produced in the second phase is close enough to the golden summary provided in the dataset. To achieve this, several variations of the ROUGE metric can be used.

A diagram of the overall process is displayed in Figure 1.

4 Experiment Design

4.1 The BioASQ dataset

BioASQ is a challenge on large-scale biomedical semantic indexing and question answering which happens on a yearly basis (BioASQ, 2019). This challenge consists of a couple of sub challenges each year. The dataset of this challenge consists of 3,243 questions along with the documents related to them and a golden extractive summary, provided in a JSON file indexed by the questions. The query of the summary is given as the value of the "body" field and the documents to be processed are given as the values of the "documents list" field. The value of documents is a list of url links to PubMed scientific articles (PubMed, 2019). For this challenge only the title and the abstract of these papers were considered as the input documents for the summarization process. The golden Extractive Summary for this specific query is provided in the "snippets" field, which is a list of the documents and the offset of the beginning and end of the

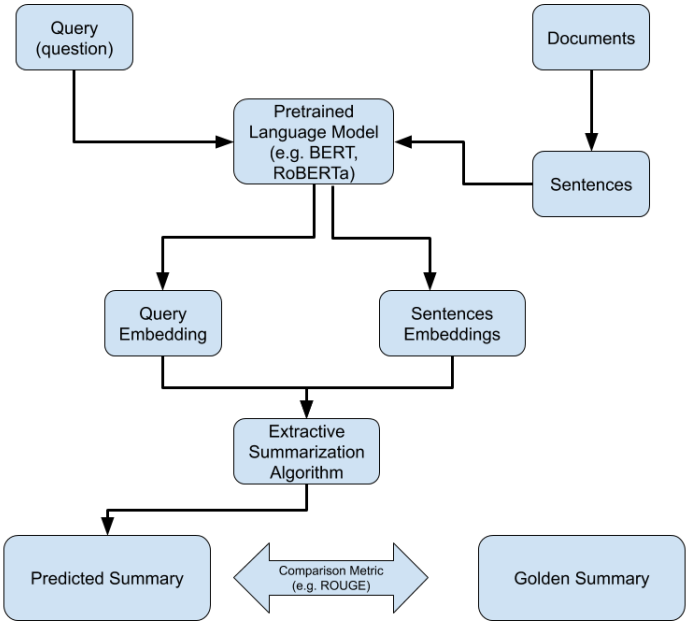

Figure 1: The overall functioning of the system. In the first phase (upper boxes of the diagram), the embeddings for both the sentences and the query are produced. In the second phase (middle boxes of the diagram) a Query-focused Extractive Summary is produced. In the third phase (lower boxes of the diagram), the summary produced by the process is compared against the correct summary.

text snippet. The inputs to the system for finding the summaries are the abstracts of the papers that are given in the "documents" field, while the output is a list of snippets that specifies which document and the offset pairs of where this extractive summary was found.

After learning that the whole dataset was too huge to do full experiments with it, it was chosen a random sample of 100 questions and the documents relevant to these questions were retrieved. In the end, the sampled dataset was composed of 1,010 documents.

A sample of the structure of the BioASQ dataset can be found in Figure 2.

4.1.1 On the technical difficulties of producing the embeddings for a lot of sentences

The first attempt was to load all the text in memory and use the source code for BERT and their package to produce the embeddings. After several attempts, it was found that even with good computational resources (like Compute Canada)

```
{"questions":[
   {
       "body":"Is Rheumatoid Arthritis more common in men or women?",
       "id":"5118dd1305c10fae750000010",
       "documents": [
           "http://www.ncbi.nlm.nih.gov/pubmed/12723987"
           , ...
       ],
       "snippets":[
           {
               "document": "http://www.ncbi.nlm.nih.gov/pubmed/22853635",
               "text": "The expression and clinical course of RA are
                  Influenced by gender. In developed countries the
                  prevalence of RA is 0,5 to 1.0%, with a male:female
                  ratio of 1:3.",
               "offsetInBeginSection": 559,
               "offsetInEndSection": 718,
               "beginSection": "sections.0"
               "endSection": "sections.0"
           }, ...
       ]
   }, ...
]}
```


Figure 2: Sample of the BioASQ dataset, which includes the question as the "body" field, and the correct summary of question in the "snippets" field.

they were taking a lot of time.

It is for these reasons that it was decided to use the scripts from the BERT team. The paper (Devlin et al., 2019) is accompanied with a GitHub repository that includes some wrappers to do some useful tasks surrounding BERT. One of these tasks is to take a plain text file and for every sentence in the file, produce the embeddings for all the tokens, including the [CLS] token. After performing several tests with this script for producing embeddings, it was discovered that it was still taking a lot of time, so a wrapper was created that using the multiprocessing module of Python, gets rid of the Global Interpreter Lock and for every processor, creates a process that runs the embeddings script for a batch of sentences. This approach proved to be very effective, as it was able to produce the embeddings for all the sentences present in the whole dataset for tiny-BERT, base-BERT and BioBERT in less than two hours for each of these pre-trained models.

4.2 Ranking Algorithms

This project experimented with different scoring methods to calculate a score according to the given query. The embedding of a sentence within the documents (s) and the query(q), that were obtained using different pre-trained models, were used with Cosine Similarity (Eq.1), Dot Product(Eq.2) and Euclidean Distance Eq. 3 for ranking purposes. These three approaches have been proved to be appropriate metrics when working with comparing embeddings of sentences.

$$cos\_sim(q,s) = \frac{\vec{q}\cdot\vec{s}}{|\vec{q}|\cdot|\vec{s}|} = \frac{\sum_{i=1}^{m} w_{i,q}w_{i,d}}{\sqrt{\sum_{i=1}^{m} w_{i,q}^2}\cdot\sqrt{\sum_{i=1}^{m} w_{i,s}^2}} \tag{1}$$

$$dot\_prod(q,s) = \vec{q}\cdot\vec{s} = \sum_{i=1}^{m} w_{i,q}w_{i,s} \tag{2}$$

$$euclidean\_dist(q,s) = \sqrt{\sum_{i=1}^{n} |w_{i,q} - w_{i,s}|^2} \tag{3}$$

4.2.1 Euclidean distance

After learning that Euclidean Distance was the approach that performed the worst in the previous experiment, this project decided to focus in this metric and try to improve its performance.

With this metric, the idea is to take advantage of the natural division of the text into documents and questions to create a preliminary summary that could then be further summarized comparing it with the query. The hypothesis of this new approach is that by selecting the most relevant sentences of each structure (document or question), the Recall of the Query-focused summarization technique can be improved. A new set of experiments was performed, using the following two similar approaches: pre-summarizing by question and pre-summarizing by document. For each of these two approaches, general summaries independent of the query were produced to measure the impact of the query-focused summarization.

The first step is to summarize the structure by computing its centroid as the mean of the embeddings of the sentences it contains and then select the n closest sentences to the structure centroid. This step has several hyper parameters to tune: the number of sentences to take from each document (n), the metric used to compare embeddings (e.g. euclidean distance, cosine similarity) the way of of producing the embedding of each sentence (using the [CLS] token or other methods), the pre-trained model used to produce the embeddings (e.g. BERT-Tiny, BERT-Base, BioBERT) and the way of producing the structure centroid (e.g. mean, median).

The second step is to create a final summary that is more related to the query. It is possible to simply take the structure summary produced in the previous step (approach that was retained to measure the impact of the second step), but a more sensitive approach is to rank the sentences of the structure summary and only keep the m closest sentences to the embedding of the query. This second step also has several hyper parameters: the number of sentences to keep (m), the way of producing the embedding of the query and the metric used to compare the embedding of the query and the structure summary.

5 Evaluation

The two ranking approaches, the first one being to compare three different scoring functions, and the second one focusing exclusively on Euclidean Distance were implemented and evaluated. Rouge-1, Rouge-2 and Rouge-L were the three evaluation metrics used for comparing the computed summary by the proposed system against the gold standard summary.

5.1 Results of Ranking

The hypothesis for this approach is that the sentences that are more similar to the query should have a higher score with Cosine Similarity and Dot Product, while with Euclidean Distance, the more similar the sentences, the less value it would result in. After sorting the scores of these sentences, a certain number of best sentences need to be selected to be part of the summary.

This project experimented with three variations to find the best number of sentences to include in the summary, each of them with BERT-Tiny, BERT-Base and BioBERT.

In the first variation (Variation 1), only as many sentences as the golden summary were included. In the second variation (Variation 2), a range of 5 to 30 sentences were feed into the system. Lastly, the third variation (Variation 3) extracted the top n% of sentences from the documents related to the given query, with n varying from 10 to 60.

The Table 2 displays the results of these three variations using the BioBERT model with the sample of 100 Questions as an example of which of these variations performed better. As displayed in Fig.3, including 26 sentences achieves the highest F-measure. In Variation 3, it was found out that including 60% of the sentences have the highest F-measure. As it's depicted in Fig.4 Variation 3 was not useful and meaningful as much as the other variations.

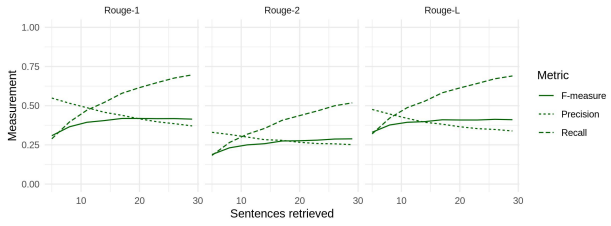

Figure 3: These plots are showing the experiment of attempt 2, which was including the range of 5 to 30 top ranked sentences in the summary. It is shown that selecting 26 sentences resulted in the highest f-measure for RougeL.

It was noticed that Variation 2 almost always resulted in better F-measure. As is shown in Table. 3, BioBERT achieved the highest results compared to the other two pre-trained models most of the time. It is also depicted that Cosine similarity is achieved better results than the other two scoring techniques.

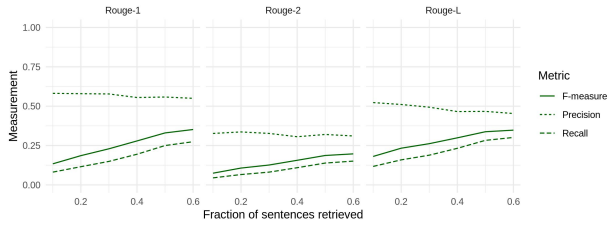

Figure 4: Comparison of performance for including the ranges of 10% to 60% of the top ranked sentences. Including top 60% ranked sentences resulted in the best F-measurement for Rouge-L.

Attempts	Rouge1			Rouge2			RougeL		
	F	P	R	F	P	R	F	P	R
Variation 1	0.43	0.53	0.38	0.25	0.3	0.22	0.39	0.43	0.39
Variation 2	0.42	0.39	0.68	0.29	0.26	0.50	0.41	0.35	0.67
Variation 3	0.35	0.55	0.27	0.20	0.31	0.15	0.35	0.45	0.30

Table 2: Experimental results of using BioBERT with the three variations. The values displayed are the maximum values that were achieved for these variations.

Algorithms	Models	Rouge 1			Rouge 2			Rouge L		
		F	P	R	F	P	R	F	P	R
Cosine Sim.	BERT-Tiny	41	41	61	27	26	43	39	36	60
	BERT-Base	41	39	66	28	26	48	41	35	65
	BioBERT	**42**	**39**	**68**	**29**	**26**	**50**	**41**	**35**	**67**
Dot product	BERT-Tiny	41	39	64	27	25	46	40	35	63
	BERT-Base	31	38	66	28	25	48	40	34	65
	BioBERT	41	37	66	27	24	47	40	34	66
Euclidean Dist.	BERT-Tiny	36	30	68	23	19	48	36	27	67
	BERT-Base	39	51	33	22	28	19	36	41	34
	BioBERT	43	53	39	25	31	22	40	44	39

Table 3: Experimental results for the first approach. All the scoring algorithms were tried using the different pre-trained model embeddings that were obtained and it has shown that Cosine Similarity and BioBERT achieved the best F-measure compared to other options. The listed results for each model is the best metrics achieved while trying the three variations.

5.2 Ranking with euclidean distance

In this approach, the following variations of this method were used: Variation A.1 (producing the preliminary summary by question), Variation A.2 (producing the preliminary summary by question, related to the query), Variation B.1 (producing the preliminary summary by document) and Variation B.2 (produc-

ing the preliminary summary by document, related to the query). Each of these variations was repeated with three different pre-trained models: BERT-Tiny, BERT-Base and BioBERT.

In variation A.1, the number of sentences to retain from each question was tested from 2 to 60. In variation A.2, the number of sentences to retain from each question was tested from 2 to 20 and the number of sentences to keep in the final summary from 2 to 50. In variation A.3, the number of sentences to keep from each document was tested from 2 to 20. In variation A.4, the number of sentences to keep from each document was tested from 2 to 10 and the number of sentences to keep in the final summary from 2 to 50. The plots showing the complete results from all the experiments can be found in Appendix A.

As displayed in Table 4, overall the most successful variation was A.1, sometimes followed by A.2. These conclusion appears to be somewhat independent of the pre-trained model used and the flavor of ROUGE, although the difference between all the variations is very small in terms of F-measure.

Model	Approach	Rouge 1			Rouge 2			Rouge L		
		F	P	R	F	P	R	F	P	R
BERT-Tiny	A.1	40	36	67	28	21	60	**40**	**32**	**68**
	A.2	40	41	59	26	25	42	39	35	59
	B.1	40	38	56	27	20	68	38	32	61
	B.2	**41**	**42**	**54**	**28**	**22**	**57**	40	32	66
BERT-Base	A.1	**40**	**39**	**59**	**27**	**20**	**62**	**39**	**33**	**61**
	A.2	**40**	**39**	**59**	25	24	40	38	34	57
	B.1	38	36	55	26	19	71	37	29	68
	B.2	40	39	57	27	21	55	39	33	61
BioBERT	A.1	**41**	**41**	**59**	**29**	**23**	**58**	**40**	**31**	**74**
	A.2	**41**	**41**	**59**	27	25	43	40	37	58
	B.1	40	42	48	27	19	69	38	31	63
	B.2	41	39	62	28	22	60	40	33	67

Table 4: Experimental results for the second approach. F-measure (F), Precision (P) and Recall (R) for the best-performing ranking algorithms using Euclidean distance. Every metric value was rounded and multiplied by 100 to increase readability. The bolded cells are the best-performing for each model within each flavor of ROUGE. The variations taken were the following: Variation A.1 (producing a preliminary summary by question), Variation A.2 (producing a preliminary summary by question, related to the query), Variation B.1 (producing a preliminary summary by document) and Variation B.2 (producing a preliminary summary by document, related to the query).

It is important to note that the variations did impact the metrics in a significant manner, where the variations using the query generally have lower precision but higher recall than the variation that are not using it.

This is aligned with the original hypothesis that using the query would im-

prove the Recall of the system, but could also be due to the fact that given the dataset, it was already known from the beginning that the documents were related to the query, so further filtering them takes away important information that otherwise would have made it into the final summary.

6 Conclusion

This project experimented with different pre-trained models and ranking algorithms to find query-focused summarization of multiple documents. After a comparative study on different scoring techniques and pre-trained language models, Cosine Similarity with BioBERT got the best result. The results showed that using BioBERT was almost always better than the other models, which concludes that BioBERT sentence embeddings were more meaningful compared to other two models. An explanation for this fact is that the BioASQ dataset consists of bio-medical scientific papers, where the vocabulary is considerably different from general domain text. One of the challenges of this project was the time consuming part of extracting the embeddings. To address this issue a tool to efficiently compute embeddings for a variety of pre-trained models were developed which is also scalable.

7 Future work

The developed system will be adapted to help scientists to speed up their exploration in related research work. These selected papers (documents) and the problem definition that researchers want to address (query) will be used as the inputs of this system. The returned summary of this system would be the highlighted scientific papers that assist researchers to write their literature review in a shorter time span.

There are number of ways in which this system can be improved. One of such is graph-based approaches, where the idea is to create a graph of sentences and extract information from how these sentences are related to the given query. Furthermore, a bipartite graph that relates the sentences into topics can be developed, as in (Parveen et al., 2015).

Sentence representations of sentences within the documents can be further improved by taking into account all the tokens of the sentence rather than just the [CLS] token. Calculating the summation or mean of tokens within the sentence, or the weighted sum of these tokens might end up to a more meaningful representation. Additional training of the model on the dataset also is a step that can be taken to further improve these embeddings.

References

BioASQ (2019). The Challenge. `http://bioasq.org/`.

Chali, Y. and Hasan, S. A. (2012). Query-focused multi-document summarization: Automatic data annotations and supervised learning approaches. *Natural Language Engineering*, 18(1):109–145.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. arxiv, `http://arxiv.org/abs/1810.04805`.

Dong, Y., Shen, Y., Crawford, E., van Hoof, H., and Cheung, J. C. K. (2019). BanditSum: Extractive Summarization as a Contextual Bandit. arxiv, http://arxiv.org/abs/1809.09672.

Ferreira, R., de Souza Cabral, L., Lins, R. D., Pereira e Silva, G., Freitas, F., Cavalcanti, G. D., Lima, R., Simske, S. J., and Favaro, L. (2013). Assessing sentence scoring techniques for extractive text summarization. *Expert Systems with Applications*, 40(14):5755–5764.

Lee, J., Yoon, W., Kim, S., Kim, D., Kim, S., So, C. H., and Kang, J. (2019). BioBERT: A pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*.

Lin, C.-Y. (2004). ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Liu, Y. (2019). Fine-tune BERT for Extractive Summarization. arxiv, `http://arxiv.org/abs/1903.10318`.

Miller, D. (2019). Leveraging BERT for Extractive Text Summarization on Lectures. arxiv, `http://arxiv.org/abs/1906.04165`.

Narayan, S., Cohen, S. B., and Lapata, M. (2018). Ranking Sentences for Extractive Summarization with Reinforcement Learning. arxiv, `http://arxiv.org/abs/1802.08636`.

Parveen, D., Ramsl, H.-M., and Strube, M. (2015). Topical Coherence for Graph-based Extractive Summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1949–1954, Lisbon, Portugal. Association for Computational Linguistics.

PubMed (2019). PubMed. `https://www.ncbi.nlm.nih.gov/pubmed/`.

Reimers, N. and Gurevych, I. (2019). Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. arxiv, `http://arxiv.org/abs/1908.10084`.

Shen, C. and Li, T. (2011). Learning to Rank for Query-Focused Multi-document Summarization. *2011 IEEE 11th International Conference on Data Mining*, pages 626–634.

A Full experimental results of Ranking with Euclidean distance

A.1 Variation A.1: Clustering by question without using the query

The full experimental results for this Variation using BERT-Tiny, BERT-Base and BioBERT are in figures 5, 6 and 7, respectively.

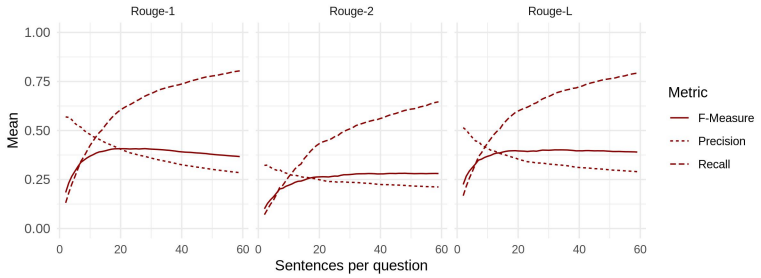

Figure 5: Experimental results for Variation A.1 (Clustering by question without using the query) with the embeddings obtained using BERT-Tiny.

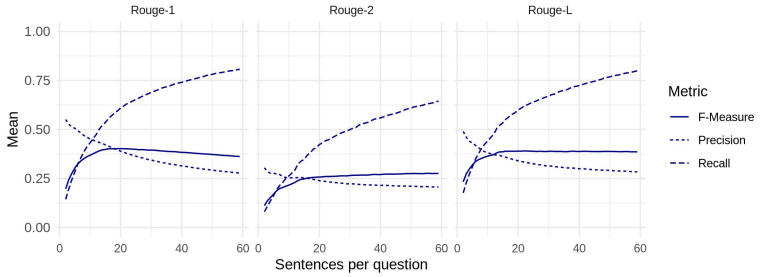

Figure 6: Experimental results for Variation A.1 (Clustering by question without using the query) with the embeddings obtained using BERT-Base.

A.2 Variation A.2: Clustering by question using the query

The full experimental results for this Variation using BERT-Tiny, BERT-Base and BioBERT are in figures 8, 9 and 10, respectively.

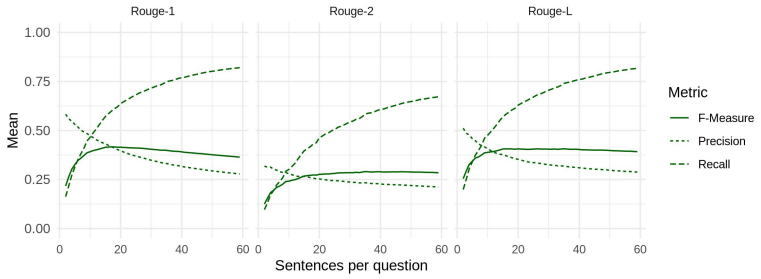

Figure 7: Experimental results for Variation A.1 (Clustering by question without using the query) with the embeddings obtained using BioBERT.

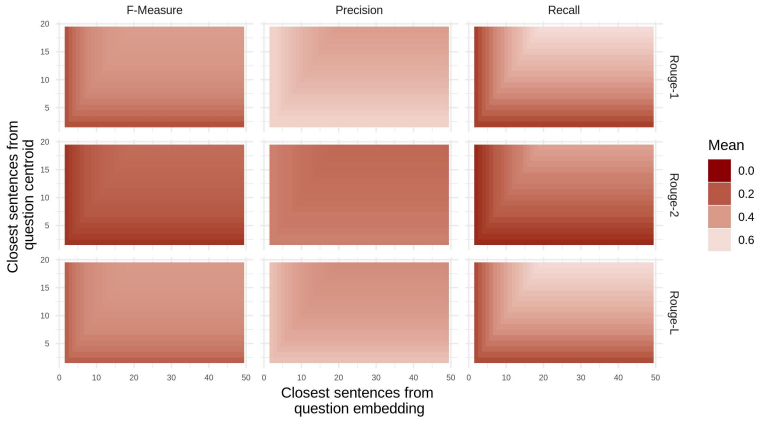

Figure 8: Experimental results for Variation A.2 (Clustering by question using the query) with the embeddings obtained using BERT-Tiny.

A.3 Variation A.3: Clustering by document without using the query

The full experimental results for this Variation using BERT-Tiny, BERT-Base and BioBERT are in figures 11, 12 and 13, respectively.

A.4 Variation A.4: Clustering by document using the query

The full experimental results for this Variation using BERT-Tiny, BERT-Base and BioBERT are in figures 14, 15 and 16, respectively.

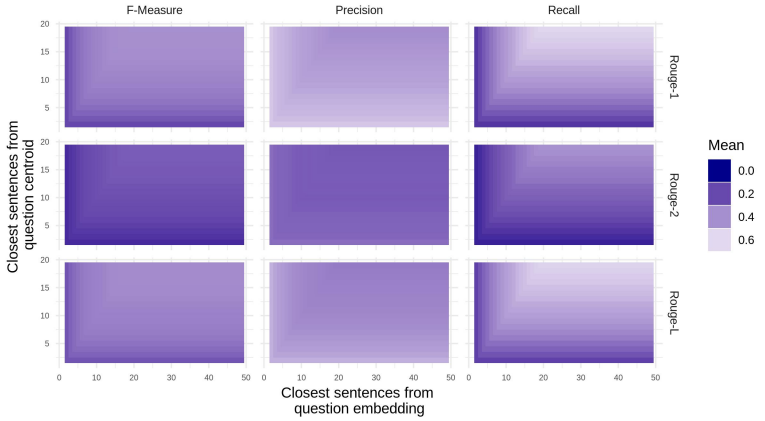

Figure 9: Experimental results for Variation A.2 (Clustering by question using the query) with the embeddings obtained using BERT-Base.

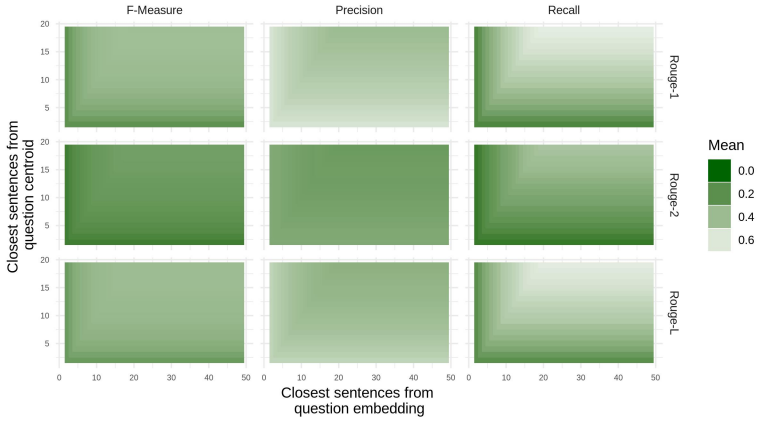

Figure 10: Experimental results for Variation A.2 (Clustering by question using the query) with the embeddings obtained using BioBERT.

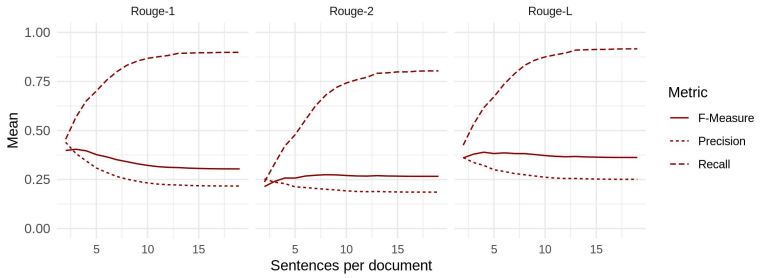

Figure 11: Experimental results for Variation A.3 (Clustering by document without using the query) with the embeddings obtained using BERT-Tiny.

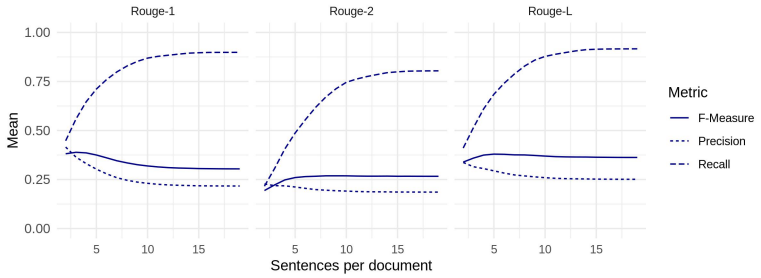

Figure 12: Experimental results for Variation A.3 (Clustering by document without using the query) with the embeddings obtained using BERT-Base.

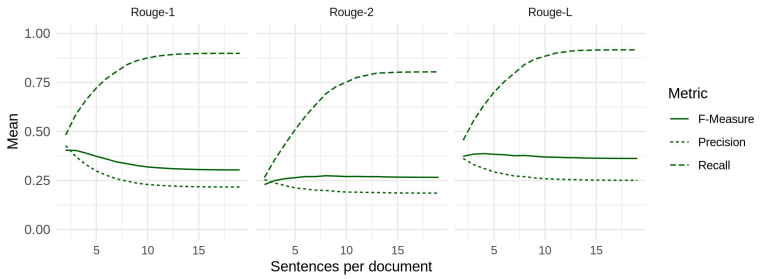

Figure 13: Experimental results for Variation A.3 (Clustering by document without using the query) with the embeddings obtained using BioBERT.

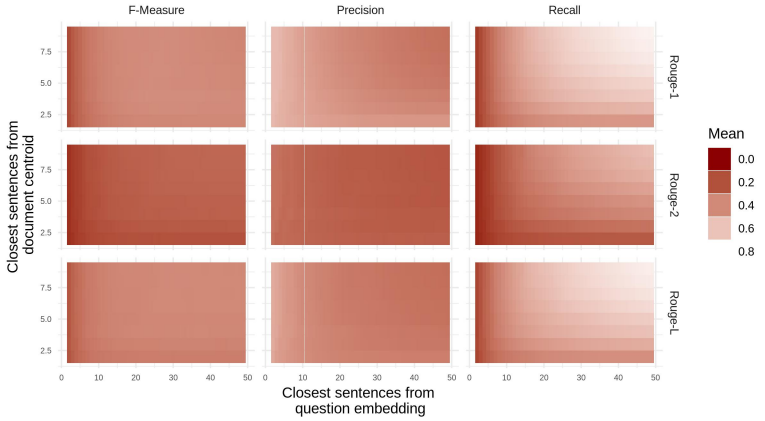

Figure 14: Experimental results for Variation A.4 (Clustering by document using the query) with the embeddings obtained using BERT-Tiny.

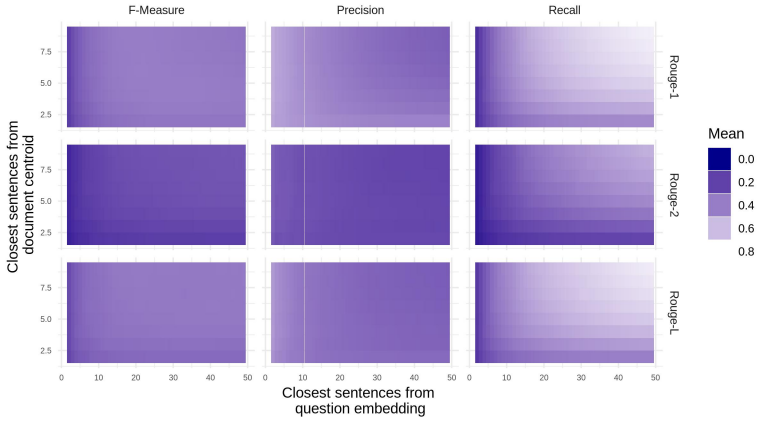

Figure 15: Experimental results for Variation A.4 (Clustering by document using the query) with the embeddings obtained using BERT-Base.

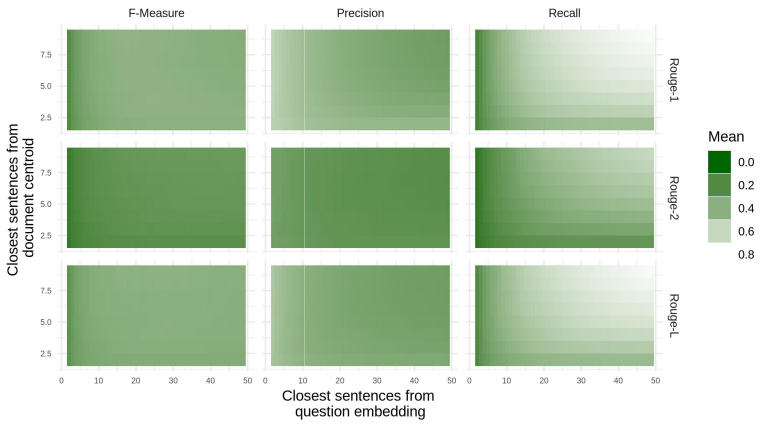

Figure 16: Experimental results for Variation A.4 (Clustering by document using the query) with the embeddings obtained using BioBERT.