Query-focused biomedical text summarization in BioASQ 8B

Jainisha Sankhavara[1][0000−0001−7460−1587] and Prasenjit Majumder[1]

Dhirubhai Ambani Institute of Information and Communication Technology, Gandhinagar, India
{jainishasankhavara,prasenjit.majumder}@gmail.com

Abstract. This paper presents query-sentence matching based and UMLS query-graph based summarization techniques for query-specific biomedical text summarization. The query-specific graphs, constructed using UMLS entities and relations, are used for matching the sentences. The core idea is to find candidate biomedical entities for query expansion which are semantically connected. The graph represents these connections and it was automatically constructed using UMLS knowledge source and biomedical text. The results of the proposed techniques experimented on previous BioASQ dataset are better as compared to the results of baseline techniques. The same techniques are applied on task 8B dataset for ideal answer generation and submitted to BioASQ8. These submitted results gave the highest scores among all participants' submissions for automatic evaluation scores(ROUGE-2 Recall and ROUGE-SU4 Recall).

Keywords: Biomedical text summarization · UMLS · Query-focused summarization.

1 Introduction

Biomedical text and information on the web are growing exponentially nowadays. Text summarization attempts to provide the users with a summarized version of the text with maximum information content in a compact, quick and intelligible way. In recent years, substantial research has been conducted to develop and evaluate various summarization techniques in the biomedical domain. Recent research has focused on a hybrid technique comprising statistical, language processing and machine learning techniques [10].

Automatic text summarization of biomedical text is a promising method for helping clinicians and researchers to efficiently obtain and understand any topic by producing a summary from one or multiple documents. The goal of text summarization is to present a subset of the source text, which expresses the most important points with minimal redundancy. Thus, text summarization

may become an important tool to assist clinicians and researchers with their information and knowledge management tasks. Sometimes, there may exist a query for which the user seeks information and sometimes it may not. In the case of query-focused summarization, the generated summary should have the answer to the query in it. It is a very usual scenario that users want exact answers along with some related details in case of their medical related queries. Therefore, we are focusing here on query-focused biomedical multi-document summarization which will be helpful to clinicians and all other users who are seeking elaborated answers to their medical related queries.

BioASQ[1] organizes challenges which include biomedical semantic indexing and question answering. The question answering task uses benchmark datasets containing development and test questions, in English, along with gold standard (reference) answers constructed by a team of biomedical experts. The participants have to respond with various types of answers. Specifically, task B has questions with their related documents and snippets for which exact answers and ideal answers need to be generated. Here we focus on generating ideal answers for the questions. The ideal answers are paragraph sized summaries with multiple sentences. We are focusing on generating ideal answers using extractive summarization on available snippets.

The remainder of this paper is presented as follows: Section 2 shows the related works. Section 3 describes the baseline methods and the proposed methods for query-focused biomedical text summarization. Sections 4 presents the experiments and results with analysis. Finally, section 6 concludes it.

2 Related work

A lot of research has been carried out in the field of biomedical text summarization. A recent survey on the research in text summarization in the biomedical domain highlights that natural language processing and hybrid techniques were prominently used for summarization of multiple documents [10].

The graph-based summarization using named-entities has been presented as EntityRank algorithm which considers information about named entities in the process of multi-document graph-based summarization [17]. Their results show that the addition of named-entity information increases the performance of graph-based summarizers in the biomedical domain. [11] studied different feature selection approaches for identifying important concepts in a biomedical text and showed that the concept based summarization method outperforms other frequency-based, domain-independent and baseline methods.

Query based biomedical text summarization techniques that rely on external ontology knowledge resource UMLS are proposed in the literature [7, 5, 4, 14, 18, 12, 3]. The ontology-based method of biomedical text summarization performed better when compared to keyword-only methods. [16] observed that an approach for query-focused summarization of medical text based on target-sentence-specific

[1] http://bioasq.org/

and target-sentence-independent statistics along with domain-specific features outperforms other baseline and benchmark summarization systems.

Text summarization approaches often rely on the similarity measure to model the text documents. [1] has studied the impact of the similarity measure on the performance of the summarization methods in the biomedical domain and found that exploiting both biomedical concepts and semantic types improvises the quality of summaries.

Here we propose an approach for query-specific biomedical text summarization which uses ontology knowledge source UMLS [2] to generate a graph of candidate biomedical entities from the query and their semantically connected entities. The importance values of the entities in the query graph are then incorporated in the similarity measure using statistics from the dataset for selecting sentences.

3 Methods

This section describes baseline summarization methods, query sentence matching based method, modified query sentence matching based method using UMLS query graph and modified lexrank using UMLS query-graph.

3.1 Baselines

Two basic approaches of summarization TextRank [9] and LexRank [6] are used as baselines. In both TextRank and LexRank, a graph is constructed with vertex as each sentence in the document. The edges between sentences are based on some form of semantic similarity or content overlap. TextRank uses a very similar measure based on the number of words two sentences have in common while LexRank uses cosine similarity of TF-IDF vectors.

$$sim(s_1, s_2) = \frac{\sum_{w \in s_1, s_2} tf_{w,s_1} tfw, s_2 (idf_w)^2}{\sqrt{\sum_{w \in s_1} (tf_{w,s_1} idf_w)^2} \sqrt{\sum_{w \in s_2} (tf_{w,s_2} idf_w)^2}} \quad (1)$$

where tf_{w,s_i} is the number of occurrences of the word w in the sentences s_i.

In the graph, edges were formed between the sentences having similarity greater than the threshold. In both algorithms, the sentences are ranked by applying PageRank to the resulting graph. A summary is formed by combining the top ranking sentences, using a length cutoff to limit the size of the summary.

3.2 UMLS query graph based lexrank

The UMLS query-graph based lexrank is a modified version of lexrank which uses query-specific graphs generated using UMLS to get the importance of words, matches sentences using weighted cosine similarity measure, generates a graph of sentences and then applies pagerank on the graph.

Query-specific graph has been generated with the use of UMLS entities and relations as described in [15]. From the query, UMLS concepts are extracted and

represented as nodes in the graph. Along with concepts, UMLS also contains relations between entities. These relations for query concepts are used to expand nodes. Each query node gets expanded by its related UMLS concepts considering all types of relations within UMLS. After the node expansion, the expanded graph contains all the related concepts as nodes and relations as edges.

Two nodes in the graph can have an edge between them if and only if those two entities have some relation in UMLS. There are various types of relations present in UMLS and all types of relations are used. There can be some isolated nodes in the graph when any query concept is not related to any other query concept or it does not have any common related concept with another query concept.

The graph is further refined by assigning weights to the edges and removing some of the edges in the graph. The edge weights are calculated based on the co-occurrence value of entities in the text to be summarized. For any edge between two entities, the co-occurrence value of two entities is used as the weight for that edge. The edges whose edge weights are less than some threshold are removed from the graph. Less edge weight for an edge between two entities means those two entities rarely occur together and hence share very less or no context.

In the refined graph, the nodes are weighted using PageRank [13]. Node weight represents the importance of that node in the graph i.e. importance of that entity in the graph for that particular query.

The main difference with lexrank method is UMLS query-graph weighted cosine similarity. The later processing is same as it is in lexrank. The UMLS query graph based weighted cosine similarity is:

$$sim(s_1,s_2)=\frac{\sum_{w\in s_1,s_2}(tf_{w,s_1}+W_w)(tfw,s_2+W_w)(idf_w)^2}{\sqrt{\sum_{w\in s_1}((tf_{w,s_1}+W_w)idf_w)^2}\sqrt{\sum_{w\in s_2}((tf_{w,s_2}+W_w)idf_w)^2}}$$

where,

W_w = importance of concept w from query-graph, if w is in query-graph
= 0, otherwise

$tf_{w,q}$ and $tf_{w,s}$ are the number of occurrences of the word w in query q and sentence s, respectively. idf_w is the inverse of the number of sentences in which word w is present.

The formula incorporates the weights of the extended query terms from query graph in the tf-idf vectors of every sentence containing those terms.

3.3 Query-Sentence matching

The Query Sentence Matching (QSM) based summarization method compares all the sentences with the query and takes top similar sentences to query as summary. The queries and all the sentences in snippets are represented by vectors of tf-idf values of words in the sentences. The similarity measure used to match query vector and sentence vector is cosine similarity as given by equation 1. The only difference here is that the similarity is calculated between query and a sentence instead of similarity between two sentences.

3.4 UMLS query graph based query-sentence matching

The UMLS_querygraph_QSM summarization method is a modified version of QSM which uses query-specific graphs generated using UMLS to get the importance of words. For each query, it generates a query-specific graph as described in [15]. This method uses concepts identified using graph based method along with weights. The weights are incorporated in the similarity measure while ranking the sentences for summary. The UMLS query-graph based cosine similarity between query and sentences are calculated using the following formula:

$$sim(q,s) = \frac{\sum_{w \in q,s} (tf_{w,q} idf_w + W_{w,q})(tf_{w,s} idf_w)}{\sqrt{\sum_{w \in q} (tf_{w,q} idf_w + W_{w,q})^2} \sqrt{\sum_{w \in s} (tf_{w,s} idf_w)^2}}$$

where,

$W_{w,q}$ = importance of concept w from query-graph of q, if w is in query-graph
= 0, otherwise

$tf_{w,q}$ and $tf_{w,s}$ are the number of occurrences of the word w in query q and sentence s, respectively. idf_w is the inverse of the number of sentences in which word w is present.

Here, the weights are only considered for query vector. They are not incorporated in sentence vectors unlike UMLS graph based lexrank. The intuition for updating only query vector was to see it as an query expansion procedure for query-focused text summarization.

4 Experiments and Results

This section describes the experiments performed along with their results. For our experiments, we have used the dataset of BioASQ task 5B phase B and BioASQ task 8B phase B. BioASQ task 5B phase B dataset is used as a benchmark dataset which contains various questions in English, along with gold standard (reference) answers constructed by a team of biomedical experts. The test dataset has five different batches, each containing 100 questions. For each question, the relevant snippets are given and the ideal answer for that question needs to be generated. The ideal answers are paragraph sized summaries so it's a case of multi-document summarization on relevant snippets. The evaluation is done using The ROUGE [8] measures: ROUGE-2 Recall, ROUGE-2 F-measure, ROUGE-SU4 Recall and ROUGE-SU4 F-measure. The same methods are applied on BioASQ task 8B phase B batch 5 dataset and the runs were submitted to BioASQ8 challenge.

4.1 Results on BioASQ 5B

The results on all 5 batches of BIOASQ 5B phase B dataset are presented here. Table 1 and table 2 show a comparison of summarization methods (described in previous section) in terms of ROUGE-2 Recall and ROUGE-2 F-measure, respectively. Table 3 and table 4 shows a comparison of summarization methods in terms of ROUGE-SU4 Recall and ROUGE-SU4 F-measure, respectively.

Table 1. ROUGE-2 Recall results on BIOASQ task 5B dataset. Bold represents highest results and * represents statistically significant diffrence with $p < 0.05$ when compared to baseline lexrank.

	Batch 1	**Batch 2**	**Batch 3**	**Batch 4**	**Batch 5**
textrank	0.5188	0.5322	0.6179	**0.6169**	0.5760
lexrank	0.5716	**0.5618**	0.6256	0.6150	0.6160
lexrank_UMLS_querygraph	**0.5793**	0.5542	**0.6278**	0.6092	**0.6373***
QSM	0.5395	0.5193	0.5828	0.5697	0.5514
UMLS_querygraph_QSM	0.5447	0.5127	0.5895	0.5776	0.5689

Table 2. ROUGE-2 F-measure results on BIOASQ task 5B dataset. Bold represents highest results and * represents statistically significant difference with $p < 0.05$ when compared to baseline lexrank.

	Batch 1	**Batch 2**	**Batch 3**	**Batch 4**	**Batch 5**
textrank	0.1984	0.1857	0.2089	0.2491	0.2185
lexrank	0.2305	**0.2051**	0.2321	**0.2607**	0.2456
lexrank_UMLS_querygraph	**0.2324**	0.2043	**0.2346**	0.2562	**0.2522***
QSM	0.2195	0.1992	0.2158	0.2516	0.2172
UMLS_querygraph_QSM	0.2200	0.1962	0.2174	0.2501	0.2205

Table 3. ROUGE-SU4 Recall results on BIOASQ task 5B dataset. Bold represents highest results.

	Batch 1	**Batch 2**	**Batch 3**	**Batch 4**	**Batch 5**
textrank	0.5419	0.5581	0.6248	**0.6345**	0.5801
lexrank	0.5887	**0.5878**	0.6358	0.6267	0.6169
lexrank_UMLS_querygraph	**0.5951**	0.5786	**0.6384**	0.6234	**0.6360**
QSM	0.5580	0.5432	0.5938	0.5808	0.5628
UMLS_querygraph_QSM	0.5607	0.5399	0.5988	0.5937	0.5785

Table 4. ROUGE-SU4 F-measure results on BIOASQ task 5B dataset. Bold represents highest results.

	Batch 1	**Batch 2**	**Batch 3**	**Batch 4**	**Batch 5**
textrank	0.1958	0.1804	0.2038	0.2419	0.2114
lexrank	0.2253	**0.2013**	0.2279	**0.2518**	0.2384
lexrank_UMLS_querygraph	**0.2270**	0.1999	**0.2305**	0.2480	**0.2439**
QSM	0.2154	0.1935	0.2126	0.2437	0.2117
UMLS_querygraph_QSM	0.2152	0.1917	0.2143	0.2428	0.2148

4.2 Discussion

The results show that UMLS_querygraph_QSM gives an improvement over QSM. The method lexrank_UMLS_querygraph gives an improvement over lexrank for batch 1,3 and 5 of the dataset. For the other two batches, the results are comparable. For batch 5, the ROUGE-2 Recall and ROUGE-2 F-measure results of lexrank_UMLS_querygraph are statistically significantly better than lexrank.

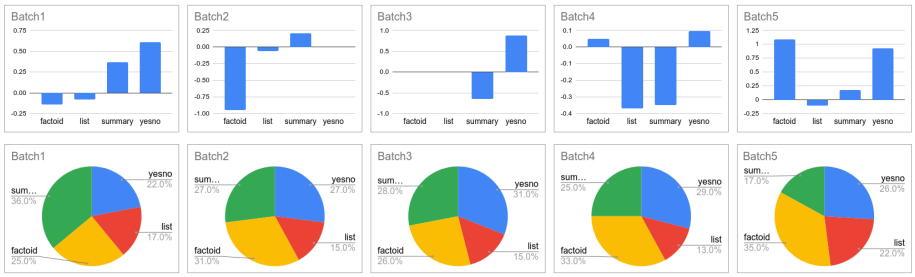

Fig. 1. query wise change in lexrank_UMLS_querygraph with respect to baseline lexrank and distribution of types of the queries

The graphs in the first row of fig. 1 shows the query type wise change in the results of lexrank_UMLS_querygraph as compared to lexrank for every batch of the data while the second row shows the batch wise distribution of the queries based on their types. From the graphs, we can say that the 'yesno' type of questions are getting improved in all batches (considering batch 2 where it is showing zero change: no improvement and no deterioration). The graph of batch 5 indicates that the major part of effectiveness of the method lexrank_UMLS_querygraph comes from the improvements in 'factoid' and 'yesno' type of queries with a small contribution from 'summary' types of the queries. For batch 2 and 4 where lexrank_UMLS_querygraph failed, decrements in 'factoid', 'list' and 'summary' type of queries must be the reason.

4.3 Results on BioASQ 8B

Table 5 shows the results of submitted runs for BioASQ task 8B phase B batch 5 using the techniques described in section 3. Surprisingly, for BioASQ 8B batch 5, simple QSM approach outperformed textrank, lexrank and UMLS querygraph based approaches.

Table 5. BioASQ task 8B Phase B Ideal answer generation results of batch 5

System	R-2 Recall	R-2 F-measure	R-SU4 Recall	R-SU4 F-measure
DAIICT_QSM	**0.6646**	**0.3468**	**0.6603**	**0.3306**
DAIICT_text	0.6627	0.3425	0.6587	0.3261
DAIICT_lex	0.6431	0.3351	0.6399	0.3207
DAIICT_lex_UMLSgraph	0.6411	0.3332	0.6382	0.3190
DAIICT_QSM_UMLSgraph	0.6411	0.3332	0.6382	0.3190

Among all participants' submitted runs, these five submitted runs appeared to be top five(DAIICT_QSM being the highest) for ROUGE-2 Recall and ROUGE-SU4 Recall. For ROUGE-2 F-measure, DAIICT_QSM is second highest considering all participants' runs and it is the third highest in case of ROUGE-SU4 F-measure.

5 Conclusion

This paper presents query sentence matching based summarization techniques and UMLS query graph based summarization techniques which were submitted to BioASQ8 challenge for ideal answer generation in task B on biomedical semantic question answering. These techniques incorporate weights of the candidate biomedical entities from queries and their semantically related entities identified by UMLS and do the matching using tf-idf vectors. The results of the proposed techniques on BioASQ 5B phase B dataset are compared with baselines textrank and lexrank. The analysis shows that the UMLS query graph based method gives comparable results with the baselines and helps to improve 'yesno' type of questions. The results of these techniques on BioASQ task 8B phase B batch 5 dataset were the highest among all participants where simple QSM approach outperformed UMLS graph based QSM as well as UMLS graph based lexrank.

References

1. Azadani, M.N., Ghadiri, N.: Evaluating different similarity measures for automatic biomedical text summarization. In: International Conference on Intelligent Systems Design and Applications. pp. 305–314. Springer (2017)
2. Bodenreider, O.: The unified medical language system (umls): integrating biomedical terminology. Nucleic acids research **32**(suppl_1), D267–D270 (2004)
3. Cao, Y., Liu, F., Simpson, P., Antieau, L., Bennett, A., Cimino, J.J., Ely, J., Yu, H.: Askhermes: An online question answering system for complex clinical questions. Journal of biomedical informatics **44**(2), 277–288 (2011)
4. Chen, P., Verma, R.: A query-based medical information summarization system using ontology knowledge. In: 19th IEEE Symposium on Computer-Based Medical Systems (CBMS'06). pp. 37–42. IEEE (2006)
5. Elhadad, N., Kan, M.Y., Klavans, J.L., McKeown, K.R.: Customization in a unified framework for summarizing medical literature. Artificial intelligence in medicine **33**(2), 179–198 (2005)

6. Erkan, G., Radev, D.R.: Lexrank: Graph-based lexical centrality as salience in text summarization. Journal of artificial intelligence research **22**, 457–479 (2004)
7. Fiszman, M., Rindflesch, T.C., Kilicoglu, H.: Abstraction summarization for managing the biomedical research literature. In: Proceedings of the HLT-NAACL workshop on computational lexical semantics. pp. 76–83. Association for Computational Linguistics (2004)
8. Lin, C.Y.: Rouge: A package for automatic evaluation of summaries. In: Text summarization branches out. pp. 74–81 (2004)
9. Mihalcea, R., Tarau, P.: Textrank: Bringing order into text. In: Proceedings of the 2004 conference on empirical methods in natural language processing. pp. 404–411 (2004)
10. Mishra, R., Bian, J., Fiszman, M., Weir, C.R., Jonnalagadda, S., Mostafa, J., Del Fiol, G.: Text summarization in the biomedical domain: a systematic review of recent research. Journal of biomedical informatics **52**, 457–467 (2014)
11. Moradi, M., Ghadiri, N.: Different approaches for identifying important concepts in probabilistic biomedical text summarization. Artificial intelligence in medicine **84**, 101–116 (2018)
12. Morales, L.P., Esteban, A.D., Gervás, P.: Concept-graph based biomedical automatic summarization using ontologies. In: Proceedings of the 3rd textgraphs workshop on graph-based algorithms for natural language processing. pp. 53–56. Association for Computational Linguistics (2008)
13. Page, L., Brin, S., Motwani, R., Winograd, T.: The pagerank citation ranking: Bringing order to the web. Tech. rep., Stanford InfoLab (1999)
14. Reeve, L., Han, H., Brooks, A.D.: Biochain: lexical chaining methods for biomedical text summarization. In: Proceedings of the 2006 ACM symposium on Applied computing. pp. 180–184. ACM (2006)
15. Sankhavara, J., Dave, R., Dave, B., Majumder, P.: Query specific graph-based query reformulation using umls for clinical information access. Journal of Biomedical Informatics p. 103493 (2020)
16. Sarker, A., Mollá, D., Paris, C.: An approach for query-focused text summarisation for evidence based medicine. In: Conference on Artificial Intelligence in Medicine in Europe. pp. 295–304. Springer (2013)
17. Schulze, F., Neves, M.: Entity-supported summarization of biomedical abstracts. In: Proceedings of the Fifth Workshop on Building and Evaluating Resources for Biomedical Text Mining (BioTxtM2016). pp. 40–49 (2016)
18. Shi, Z., Melli, G., Wang, Y., Liu, Y., Gu, B., Kashani, M.M., Sarkar, A., Popowich, F.: Question answering summarization of multiple biomedical documents. In: Conference of the Canadian Society for Computational Studies of Intelligence. pp. 284–295. Springer (2007)