MultiGBS: A multi-layer graph approach to biomedical summarization

Ensieh Davoodijam
Department of Electrical and Computer Engineering, Isfahan University of Technology, Isfahan 84156-83111, Iran
e.davoodijam@ec.iut.ac.ir

Nasser Ghadiri *
Department of Electrical and Computer Engineering, Isfahan University of Technology, Isfahan 84156-83111, Iran
nghadiri@iut.ac.ir

Maryam Lotfi Shahreza
Department of Computer Engineering, University of Shahreza, Shahreza 86149-56841, Iran
maryam.lotfi@gmail.com

Fabio Rinaldi
Department of Computer Engineering, University of Zurich, CH-8050 Zurich, Switzerland
fabio.rinaldi@uzh.ch

Abstract

Automatic text summarization methods generate a shorter version of the input text to assist the reader in gaining a quick yet informative gist. Existing text summarization methods generally focus on a single aspect of text when selecting the sentences, causing potential loss of essential information. We propose a domain-specific method that models a document as a multi-layer graph to enable processing multiple features of the text at the same time. The features we used in this paper are word similarity, semantic similarity, and co-reference similarity that are modeled as three different layers. The summarizer selects the sentences from the multi-layer graph based on the MultiRank algorithm and length of concepts. The proposed MultiGBS algorithm employs UMLS and extracts concepts and relationships with different tools such as SemRep, MetaMap, and OGER. Extensive evaluation by ROUGE and BertScore shows increased F-measure values. Compared with leveraging BERT as extractive text summarization, the improvements in F-measure are 0.141 for
ROUGE-L, 0.014 for ROUGE-1, 0.018 for ROUGE-2, 0.024 for ROUGE-SU4, and 0.0094 for BertScore.

**Keywords:** Automatic text summarization; text mining; multi-graph text modeling; concept-based summarization; domain-specific summary

1 Introduction

With the rapid development of the Internet and other new technologies, the amount of electronic textual content in the biomedical domain has increased explosively [1]. More than 30 million biomedical articles are cited in PubMed [2]. It is essential to provide improved mechanisms to extract information quickly and efficiently. Text summarization aims at identifying the fundamental, meaningful information in a single document or set of related documents. The goal of automatic text summarization is to present the source text as a shorter version [3]. Summaries can be used by other systems as well, such as information retrieval (IR), question answering, and text classification [4].

In general, existing summarization approaches could be categorized into five approaches: (1) Statistical-based, (2) Topic-based, (3) Discourse-based, (4) Machine learning-based, and (5) Graph-based methods. Despite the complexity of some existing methods, serious challenges still remain, including performance degradation and requiring a large training corpus [4][5][6] [7]. These challenges will be discussed in more detail in Section 2.

In recent years, graph-based approaches have attracted much attention in the summarization domain. There are two fundamental issues in graph-based methods:

- How to construct a graph for summarization?
- How to develop an algorithm for sentence ranking?

In the traditional graph-based algorithms, each node stands for an object, and two nodes are connected by an edge if there exists some kind of relationship between them [8]. Basic graph algorithms only cover one type of relationship, such as word similarity or semantic similarity, and cannot fully capture information and cover several aspects or subtopics. Therefore, to increase the diversity and coverage of summarized results, new algorithms are needed.
In this paper, we propose a novel multi-layer graph-based biomedical text summarizer that models three different types of relationships between sentences as MultiGBS. The system builds an undirected weighted multi-layer graph from the source document. The graph covers semantic relationships, word relationships, and co-reference relationships.

The proposed MultiGBS algorithm uses the UMLS (Unified Medical Language System) biomedical knowledge source to extract the concepts and identify the correlations among them. UMLS is a collection of several biomedical vocabularies and standards [9][10]. MetaMap [11] and OGER [12] are two different tools that identify the concepts and map original text to UMLS concepts. Finally, the proposed method employs the MultiRank algorithm [14] on the multi-layer graph to rank the sentences. This algorithm is similar to PageRank and runs on multi-layer graphs [14].

The performance of the summarization system is evaluated over a corpus of scientific papers in the biomedical domain. The experimental results confirm that the proposed biomedical summarizer outperforms other baselines and well-established methods in terms of the Recall-Oriented Understudy for Gisting Evaluation (ROUGE) metrics and BertScore [15][16][17].

The rest of the paper is organized as follows. Section 2, shortly reviews the history of text summarization and outlines the related work. We present a description of our biomedical text summarization method in Section 3. The results of the performance evaluation are shown in Section 4. Section 5 reports some conclusions and limitations.

2 Related Work

There are different categorization approaches for summarization methods depending on the type of input, output, and the user requirement. In one approach, the summarization methods can be classified into *generic* and *query-oriented* (also referred to as user-oriented or query-focused) methods, based on the purpose of the summaries [18]. A query oriented method creates an outline that, unlike a general summary, includes the content related to a given query. In another categorization approach, concerning the number of the source documents, the summarizers can be categorized into *single-document* versus *multi-document* methods. In the third approach, the output of the summarizer can be classified as *extractive* versus *abstractive* summaries. Extractive summarization chooses the most meaningful subset of the sentences in the document set, while abstractive summarization creates the new sentences, unseen in the sources. Another approach
divides the summaries into indicative versus informative summaries. Informative summaries include sufficient content, and users do not require the original input for understanding. On the other hand, indicative summaries just provide a view of the content, and users still need to review the original document [4][7][18].

The proposed method provides a generic, single-document, informative, and extractive summary. As mentioned in Section 1, there are five categories of extractive approaches for the summary generation that are shortly discussed in the following.

- **Statistically based approaches:** These techniques consider the statistical features such as the position of the sentence, positive keyword or negative keyword, the centrality of the sentence, the similarity of the sentence with the title, length of the sentence, presence of numerical data in the sentence, TF*IDF (Term Frequency–Inverse Document Frequency), etc. [19][20][21]. It has been shown that the use of statistical features alone is not sufficient, but a combination of these features with other methods could provide better results [4][7].

- **Topic-based approaches:** In these methods, the summaries are created by identifying the topic. The topic is the primary concern of the document that is described in five different ways: (1) Topic signatures, (2) Enhanced topic signatures, (3) Thematic signatures, (4) Modeling the documents content structure, and (5)Templates [22]. The algorithms in this category are rather sophisticated and require considerable skills to be used [23].

- **Machine learning-based approaches:** These methods are based on the well-known machine learning algorithms such as classification [24], Hidden Markov Models [25], Bayesian methods, neural networks [22], Support Vector Regression (SVR) and Least Angle Regression [26]. SummaRuNNer is an extractive summarizer that uses a bi-directional recurrent neural network (RNN) for sentence representation and sentence selection [27]. Cheng and Lapata introduced NN-SE as an extractive summarizer. In this model, sentences are represented with convolutional neural networks (CNN) network, and sentences are selected with NN-SE [28]. PriorSum uses the gold standard summaries for training. It merges a multi-layer CNN network with statistical features such as sentence position and average term frequency [29]. A limitation of the existing methods in this category is that the need for an extensive training corpus [4].
- **Discourse-based approaches**: In this group, the summarization methods consider linguistic knowledge [30][31]. Afnan et al. [32] employ the coherence and cohesion of a document. Other approaches try to combine statistical and linguistic techniques. These algorithms have a moderate performance and do not provide significant improvements [4].

- **Graph-based approaches**: Graph-based summarization methods represent the document as a graph with nodes, which show different parts such as terms, phrases, concepts, or sentences, and edges, which describe the similarity relation between them [8]. There are various measures for calculating the similarity between the text units, such as cosine similarity [33], the longest common subsequences [34], the number of common words [35]. LexRank is a well-known multi-document summarization system. It creates a weighted graph based on the predefined threshold and finds the essential sentences using a random walk on the graph [33].

In this paper, we focus on biomedical summarization methods. There are a variety of acronyms, abbreviations, synonyms, or hypernyms in the biomedical domain, which the general text summarization methods do not take into consideration. Some domain-independent summarization systems used frequent itemset mining [36]–[38]. Other methods used UMLS as a domain of knowledge resources and extract the concepts and vocabularies [39]–[43]. BioChain [44] is a typical single document summarization that uses UMLS and creates concept chains that were ranked based on concept frequency. Another method created a simple graph from the input text based on UMLS concepts [45], which measured the similarity between nodes based on “is-a” relationship and clustered the graph based on Genetic Graph-based Clustering (GGC). Nasr et al. [43] created a simple graph and used n-gram based on frequent set mining to create edges. Most existing graph-based methods of biomedical text summarization cannot handle more than one relationship between the text elements simultaneously, thus providing limited accuracy.

The proposed method is a novel biomedical summarization method that uses multi-layer graphs instead of simple graphs, making it possible to handle several types of relationships between the sentences. To this aim, MultiGBS employs different tools such as MetaMap, OGER, and SemRep to extract the concepts and relations from the biomedical domain, which are introduced in Section 4.
3 Methods

MultiGBS is a graph-based method that has two main steps, namely graph creation, and sentence selection, as presented in the following.

3.1 Graph creation

The focus of our approach is to create a graph based on the heterogeneous network concepts, i.e., a network that involves different types of objects or links. There are two types of heterogeneous networks [46]:

- **Multimode networks** that consist of objects of different types with various relationships between them.
- **Multidimensional networks** or **multi-layer networks** that have multiple types of relationships among the same set of objects. Each dimension of the network represents one type of interconnection between objects.

MultiGBS uses multi-layer networks to cover essential aspects and relations between the sentences. A document is modeled as a weighted graph in which each sentence is a node, and the different types of edges represent the various relationships between sentences. A formal definition is given in the following:

**Definition 1.** Let $D$ be a document with a set of sentences $S$. The document can be represented as an undirected weighted graph $G = (V, E, \alpha)$, where $\alpha$ is a set of layers $\alpha=1,2,...,M$, and $E$ shows the quadruple $(u,v,d,w)$ where $w$ is the weight on the edge between $u$ and $v$ in the layer $\alpha$.

Our network has three layers (Fig 1) which each layer represents a different similarity, as below:

- Semantic Layer
- Word Layer
- Co-references Layer
Fig 1: Multi-layer graph with three layers. The red edges show the first layer (Semantic Similarity), the green ones represent the second layer (Word Similarity), and the purple edges display the third layer (Co-reference).

Semantic Layer: MultiGBS extracts concepts and semantic types from UMLS. It is widely used in biomedical text summarization research as a collection of several vocabularies and standards. UMLS has three primary data sources: (1) Meta-thesaurus, (2) Semantic Network, and (3) Specialist Lexicon, which are shortly explained in the following [10].

1. UMLS Meta-thesaurus: A set of various biomedical concepts, names, and synonyms obtained from approximately 200 different vocabularies. It is considered the major UMLS component.
2. UMLS Semantic Network: All concepts in the UMLS Meta-thesaurus can be classified using the Semantic Network component. It defines 133 general categories and 44 relationships between categories.
3. UMLS Specialist Lexicon: A database that gives lexical data and programs for language processing.

To identify and extract the biomedical concepts from a text document, MultiGBS employs two different tools:

MetaMap: is a popular program developed at the National Library of Medicine that maps biomedical text to the concepts of the UMLS Meta-thesaurus [10], [11].

OGER: OntoGene’s entity recognizer that is a text-mining tool for identifying biomedical concepts. It combines a dictionary-based annotator with a corpus-based disambiguation component [12], [13].
After extracting concepts, our summarizer measures the semantic similarity with the n-gram algorithm [47].

**Word Layer**: In this step, unlike the semantic layer, MultiGBS just considers the exact word of sentences and calculates the similarity between sentences based on n-gram.

**Co-reference Layer**: MultiGBS creates a relationship between two sentences based co-reference resolution. It is the task of finding all expressions that refer to the same entity in a text. For example, consider this sentence:

> My favorite sport is football because it a way of life, Sara said.

“my”, and “Sara” belong to the same entity and “football” and “he” belong to the same entity.

SemRep [49] is used to extract co-reference resolution. It is a natural language processing system designed to recover semantic propositions from biomedical text using the specified syntactic analysis and structured domain knowledge from the UML [49].

Finally, a document is represented as a weighted undirected multi-layer graph in which nodes are the sentences, and the edges show the different similarity relations between these sentences. The different weights show the strength of the links between different sentence pairs within the given document.

### 3.2 Sentence selection

After creating the graph, the proposed algorithm needs to select the sentences to create the summary. As mentioned in Section 2 for the extractive summary, the summarization system needs to select the essential sentences from the original text. MultiGBS ranks all the sentences based on the multi-layer network and chooses the top-ranked sentences according to the compression rate. The compression rate is the percentage of original sentences to be shown in the summary. We use an algorithm that aggregates three similarity measures on the multi-layer graph and ranks the sentences. Therefore, we examine two different methods for sentence selection, which are explained in the following.

#### 3.2.1 MultiGBS sentence selection algorithm-Basic

The proposed MultiGBS algorithm uses the MultiRank [14] to rank all sentences. It is based on a random walk algorithm and works on three different layers. The output of this algorithm is the
The centrality $X_i$ of the nodes $i=1, 2, \ldots, n$ that determines the central nodes in highly influential layers. MultiRank algorithm needs to calculate three different matrixes that extract from a multi-layer graph [14].

First of all, the MultiRank algorithm creates three networks from the multi-layer graph [14]:

1. Aggregate network: the aggregate network is the single network that aggregates all layers. A link is created if the nodes have at least a link between them in one layer. This algorithm indicates the adjacency matrix $A^{[\alpha]}$ from the aggregated network. Then it calculates the $W$ weights based on matrix $A$ as total weights:

\[
W^{[\alpha]} = \sum_{i=1}^{N} \sum_{j=1}^{N} A^{[\alpha]}_{ij}
\]  

(1)

2. Bipartite network: this network determines which nodes are connected in which layers. The MultiRank algorithm extracts the matrix $B$ from this network.

\[
B_{ai} = \frac{\sum_j A^{[\alpha]}_{ji}}{W^{[\alpha]}}
\]

(2)

3. Colored network: In this network, the MultiRank algorithm computes the different influences associated with each layer and extracts the matrix $G$.

\[
G_{ij} = \sum_{\alpha=1}^{M} A^{[\alpha]}_{ij} Z^{[\alpha]}
\]

(3)

After that, the centrality $X_i$ of the node $i$ given the influences $Z^{(\alpha)}$ of the layers that is determined by:

\[
X_i = \alpha \sum_{i=1}^{N} \frac{G_{ij}}{k_j} X_j + \beta v_i
\]

(4)

Where $\alpha$ is taken to be $\alpha = 0.85$ and $k_j, v_i, Z^{(\alpha)}$, and $\beta$ are given by:
\[ k_j = \max \left( 1; \sum_{i=1}^{N} G_{ji} \right) \]  
(5)

\[ v_i = \theta \left( \sum_{j=1}^{N} [G_{ij} + G_{ji}] \right) \]  
(6)

\[ \beta = \frac{1}{\sum_{i=1}^{N} v_i} \sum_{j=1}^{N} \left[ 1 - \alpha \theta \left( \sum_{i=1}^{N} G_{ji} \right) \right] X_j \]  
(7)

\[ Z^{(a)} = \frac{1}{\omega} W^{(a)} \sum_{i=1}^{N} B_{ai} X_i \]  
(8)

\( \omega \) is showing a normalization constant.

In this version, top sentences are selected by sorting items of \( X_i \) in ascending order:

\[ \text{Score}(S_i), \leftarrow \text{Sorted}(X_i) \]  
(9)

### 3.2.2 MultiGBS sentence selection algorithm-Enhanced

The MultiGBS-Enhanced algorithm adds the number of concepts for each sentence and proposes \( \text{LenCon}(S_i) \) as follows:

\[ \text{LenCon}(S_i) \leftarrow \text{the number of concepts } (S_i) / \text{the total number of concept in document} \]  
(10)

We use the MultiRank algorithm to calculate \( \text{Sorted}(X_i) \) (similar to MultiGBS-Basic) and calculate the \( \text{Score}(S_i) \) according to Eq. 11. The values of \( \gamma \) and \( \theta \) could be assigned with different weights between 0 and 1. Finally, the scores are normalized based on min-max normalization.

\[ \text{Score}(S_i) \leftarrow \gamma \text{Sorted}(X_i) + \theta \text{LenCon}(S_i) \]  
(11)

The pseudocode of MultiGBS Enhanced is shown in Algorithm 1.
Algorithm 1: MultiGBS Enhanced algorithm

Procedure Create_Graph(document) returns a Multi-layer Graph
input: D // a document
local variables: S = {s_1, s_2, ..., s_n} // list of sentences
C = {c_1, c_2, ..., c_m} // list of concepts
T = {t_1, t_2, ..., t_p} // list of tokens
R = {r_1, r_2, ..., r_q} // list of co-references relationships
For all s_i in S do
  For all s_j in S do
    Layer1 ← Semantic_Sim(C_s_i, C_s_j)
    Layer2 ← Word_Sim(T_s_i, T_s_j)
    Layer3 ← CoRef_Sim(R_s_i, R_s_j)
Function MultiGBS _ Sentence_selection_Enhanced (Multi-layer Graph) returns a sorted list of sentences
input: I = (V, E, α) // α is a set of layers, α = 1, 2, ..., M
local variables: G, W, A // Matrix
W ← Create_Aggregate_network(I)
B ← Create_Bipartite_network(I)
G ← Create_Colored_network(I)
W[α] = ∑_{i=1}^{N} ∑_{j=1}^{N} A_{ij}^{[α]} / W[α] ; G_{ij} = ∑_{α=1}^{M} A_{ij}^{[α]} Z[α] ; Z[α] = 1 / ∑_{i=1}^{N} B_{ui} X_i
X_i = α ∑_{k=1}^{M} G_{ij} X_j + β v_i // the centrality X_i of the node, i given the influences Z[α]
α is taken to be α = 0.85 and k, v, and β are calculated based on W, B, G
S(X_i) ← SortAsc(X_i) // Sort items of X_i in an ascending order
for all v_i do
  LenCon(v_i) ← the number of concepts (v_i) / the total number of concepts in document
  X
  S(i) + β LenCon(v_i)
  Score(v_i) ← γ
Function MultiGBS -ENHANCED(D) returns a summary
Input D // a document
local variables: Multi_Graph // a multi-layer graph
Multi_Graph = Create_Graph(D)
MultiGBS _ Sentence_selection_Enhanced (Multi_Graph)

4 Evaluation

In this section, the MultiGBS algorithm is evaluated with different sentence selection methods. In the following, the overall approach to evaluation, as well as the evaluation criteria and the corpus is described.

4.1 Evaluation method
There are two main categories for text summarization evaluation: intrinsic and extrinsic. The extrinsic evaluation methods try to quantify the information content based on measures such as success rate, time-to-completion, and decision-making accuracy. On the other hand, the intrinsic methods evaluate the quality or the informativeness [48]. A common method for intrinsic evaluation is to compare the content provided by the summarizer with the human models as the reference summary. Creating a reference summary is a difficult, time-consuming task that depends on the experts [48][4][7]. Therefore, we use intrinsic evaluation.

4.2 Evaluation metrics

To measure the effectiveness of our algorithm, ROUGE and BertScore are used. BertScore is based on Bidirectional Encoder Representations from Transformers (Bert)[49] as a pre-trained language representation model. This score does not compare the exact words; instead, it tries to calculate the token similarity based on Bert[15].

ROUGE is a set of metrics that compares an automatically produced summary with one or more human-made summaries. ROUGE is defined in several versions with different applications. It estimates precision, recall, and F-measure. In this work, ROUGE-1, ROUGE-2, ROUGE-L, and ROUGE-SU4 measures are used to evaluate the summarizer [16]. There is a wide range of research works on summarization that evaluated their outcomes by ROUGE based on Document Understanding Conferences (DUC)\(^1\) [50] and the Text Analysis Conferences (TAC)\(^2\)[51]. The BioASQ challenge for the biomedical domain also evaluates the task of automatically summarizing biomedical texts as part of a question answering system [52][53]. The values of recall, precision, and F-measure are reported. As a basic example of how ROUGE works, one can consider this example:

- System Summary 1: The book was found under the bed.
- System Summary 2: The little red book was found under the big funny bed.
- Reference Summary: The book was under the bed.

\(^1\) [http://www-nlpir.nist.gov/projects/duc/index.html](http://www-nlpir.nist.gov/projects/duc/index.html)

\(^2\) [http://www.nist.gov/tac/](http://www.nist.gov/tac/)
Recall = \frac{\text{number of overlapping words}}{\text{total words -- reference summary}} = \frac{6}{6} = 1

Precision 1 = \frac{\text{number of overlapping words}}{\text{total words -- system summary}} = \frac{6}{7} = 0.86

Precision 2 = \frac{\text{number of overlapping words}}{\text{total words -- system summary}} = \frac{6}{11} = 0.55

The precision is used for measuring the unnecessary word in summary. Many research works just consider recall for evaluation and do not report any results for precision [7], [39], [43], [45], [54].

4.3 Evaluation corpus

To the best of our knowledge, there is no widely-accepted standard corpus for biomedical documents and their model summaries. Existing works commonly use a collection of biomedical articles and consider the abstract of each article as the model summary [7], [39], [42], [45], [55], [56]. Therefore, we randomly select 300 biomedical scientific articles from BioMed Central.

4.4 Experimental results

We design our experiments in an incremental manner. The initial experiments show how different features can affect the ROUGE scores. This step is designed to address the following research questions:

Q1) To what extent does the type of graph influence the results?

Q2) To what extent does the length of the sentence influence the results?

Q3) To what extent does the use of different entities extraction strategies influence the results?

Q4) To what extent does the percentage of the summary influence the results?

To answer question Q1, two different types of graphs are created: weighted and unweighted graph. For unweighted graphs, the algorithm processes a pair of sentences that are linked to each other if they have a similarity above the threshold. The different thresholds values of 0.1, 0.2, and 0.3 are examined. On the other hand, the original weights without any threshold are also considered. At this step, the MultiGBS Basic algorithm uses MetaMap for extracting the concepts. Table 1 shows the precision, recall, and F-measure for 150 papers.

Table 1: The ROUGE scores for weighted and unweighted models
As shown in Table 1, the best results are obtained by the "Weighted" method. This function is edge-based, and the properties of nodes (sentences) such as the length, the position of sentences could be added.

For answering the question Q2, the MultiGBS Enhanced algorithm is evaluated. We employ the number of concepts and check the performance. Different weights are assigned to $\gamma$ and $\theta$ in Eq. 11, and the different ROUGE scores (R-1, R-2, R-W-1.2, and R-SU4) for all summarizers are given in Table 2.

Table 2: The ROUGE scores for the summaries with the different weights for the MultiGBS Enhanced (Q2).

| Evaluation Metrics | $\gamma$ | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
|--------------------|----------|---|---|---|---|---|---|---|---|
| $\theta$           | 1        | -1| 0.5| -0.5| 0.25| -0.25| 0.1| -0.1|
| Recall             |          |   |   |   |   |   |   |   |   |
| ROUGE-L            | 0.291    | 0.248| **0.294**| 0.261| 0.293| 0.269| 0.293| 0.282|
| ROUGE-1            | 0.731    | 0.493| **0.732**| 0.541| **0.732**| 0.596| 0.73| 0.661|
| ROUGE-2            | 0.361    | 0.19| **0.363**| 0.221| **0.363**| 0.259| 0.363| 0.305|

As shown in Table 1, the best results are obtained by the "Weighted" method. This function is edge-based, and the properties of nodes (sentences) such as the length, the position of sentences could be added.

For answering the question Q2, the MultiGBS Enhanced algorithm is evaluated. We employ the number of concepts and check the performance. Different weights are assigned to $\gamma$ and $\theta$ in Eq. 11, and the different ROUGE scores (R-1, R-2, R-W-1.2, and R-SU4) for all summarizers are given in Table 2.

Table 2: The ROUGE scores for the summaries with the different weights for the MultiGBS Enhanced (Q2).
The results confirm our hypothesis that the count of concepts can affect the results. The best recall ROUGE scores are reported when \( \theta \) is positive. When \( \theta \) is negative, the precision values are better than other versions. The use of OGER for concept extraction shows better performance than other tools.

For question Q3, different methods are used to extract the entities. First, the UMLS with MetaMap and OGER is employed. Second, BERN is used, which is based on BERT [57] known as a neural biomedical entity recognition and multi-type normalization tool [58]. Three different tools are evaluated on 85 articles from the CRAFT corpus using the metrics precision, recall, and F1-measure [59]. It could be observed that OGER and MetaMap performed better than BERNT based on the results shown in Table 3. Therefore, we try to use the results of MetaMap and OGER and create the multi-layer graphs for other tests.

**Table 3**: Comparison of the NER (Named Entity Recognition performance for MetaMap, OGER, and BERN)

|            | OGER  | MetaMap | BERN  |
|------------|-------|---------|-------|
| Precision  | 0.4778| 0.1635  | 0.3435|
| Recall     | 0.4964| 0.4996  | 0.1514|
| F-measure  | 0.4819| 0.2422  | 0.2026|

For question Q4, the typical size of the summary is between 15% and 35% of the size of the original text [43], [45]. Consequently, it may be observed from Tables 1 and 2 that the best arrangement
for the recall results is achieved by the MultiGBS Enhanced method that shows the improvement resulted from the weighted multi-layer graph. In this step, MultiGBS Enhanced uses OGER and MetaMap for extracting entities and creates three different summaries. The results are shown in Fig. 2 for 150 documents with different compression rates.

![RECALL](image)

**Fig.2** The ROUGE scores for the summaries with different compression rate

Adding more sentences as output can increase recall. Therefore, when the larger size of the summary is used, the recall is better. It should be considered that the size of the abstract is different for the same size of documents. For example, assume two papers with the same size=180 while the size of the abstract is 18 and 32 is the other. This setting is the same for our algorithm and other benchmarks. The results of using OGER are better than MetaMap for ROUGE-2 and ROUGE-SU4, and the best results for OGER-1 is related to the combination of MetaMap and OGER.

### 4.5 Baselines and benchmarks

Summaries generated from the MultiGBS Enhanced algorithm are compared against LexRank and BERT.

**LexRank**: LexRank is a graph-based method that uses TF/IDF cosine similarity and generates a similarity matrix. Then it extracts the most important sentences based on centrality. This idea of selecting meaningful sentences is similar to PageRank[33]. Based on our previous research, LexRank has had better results than other methods in the graph-based method category [43].
**Leveraging BERT:** Leveraging BERT is a new machine learning that uses the BERT model and K-Means clustering [60]. Table 4 shows the results of Bert, LexRank, and our proposed algorithm based on ROUGE and BertScore for 120 documents with the compression rate = 20%

|               | BERT | LexRank | MultiGBS Enhanced |
|---------------|------|---------|-------------------|
| **Recall**    |      |         |                   |
| ROUGE-L       | 0.057| 0.28    | 0.243             |
| ROUGE-1       | 0.557| 0.697   | 0.544             |
| ROUGE-2       | 0.229| 0.351   | 0.235             |
| ROUGE-SU4     | 0.263| 0.38    | 0.262             |
| Bert Score    | 0.8539| 0.8610 | 0.8561            |
| **Precision** |      |         |                   |
| ROUGE-L       | 0.026| 0.113   | 0.142             |
| ROUGE-1       | 0.259| 0.203   | 0.28              |
| ROUGE-2       | 0.109| 0.108   | 0.13              |
| ROUGE-SU4     | 0.124| 0.12    | 0.155             |
| Bert Score    | 0.8279| 0.8378 | 0.8441            |
| **F-measure** |      |         |                   |
| ROUGE-L       | 0.035| 0.158   | 0.176             |
| ROUGE-1       | 0.345| 0.308   | 0.359             |
| ROUGE-2       | 0.145| 0.162   | 0.163             |
| ROUGE-SU4     | 0.165| 0.179   | 0.189             |
| Bert Score    | 0.8406| 0.8492 | 0.85              |

Results show that MultiGBS improves F-measure. The improvements in F-measure are 0.018 for ROUGE-L, 0.051 for ROUGE-1, 0.001 for ROUGE-2, 0.01 for ROUGE-SU4, and 0.0008 for BertScore, compared with LexRank as a graph-based method. In comparing with Leveraging BERT, these improvements are 0.141 for ROUGE-L, 0.014 for ROUGE-1, 0.018 for ROUGE-2, 0.024 for ROUGE-SU4, and 0.0094 for BertScore. F1-measure is maximized if there is a variety of balance between precision and recall. Consequently, MultiGBS takes advantage of a multi-layer graph to cover the different relationships in combination with the length of the sentences to improve F1-measure.
5 Conclusion

In this paper, we have presented MultiGBS as a multi-layer graph model for biomedical text summarization. Every layer in MultiGBS represents a different aspect or similarity measure for the elements of the input document to cover the different relationships among sentences. This enriched model provides more room for improved summarization. The method was evaluated on a collection of 300 papers and compared with different methods such as LexRank and Leveraging BERT. The results show that the proposed summarizer outperforms the other tools according to metrics ROUGE and BertScore.

Further research in this area could be exploiting more similarity measures depending on the context of documents. Moreover, the user requirements may be defined as a complement to the multi-layer graph model. Establishing a gold standard as a group of documents and their summaries as a corpus will also help to improve the evaluation of automatic document summarization.

References:

[1] M. Hilbert and P. López, “The world’s technological capacity to store, communicate, and compute information,” Science, vol. 332, no. 6025, pp. 60–5, Apr. 2011.
[2] “Home - PubMed - NCBI.” [Online]. Available: https://www.ncbi.nlm.nih.gov/pubmed/. [Accessed: 15-Dec-2017].
[3] M. S. Simpson and D. Demner-Fushman, “Biomedical Text Mining: A Survey of Recent Progress,” in Mining Text Data, Boston, MA: Springer US, 2012, pp. 465–517.
[4] E. Lloret and M. Palomar, “Text summarisation in progress: A literature review,” Artif. Intell. Rev., vol. 37, no. 1, pp. 1–41, Jan. 2012.
[5] J. Yao, X. Wan, and J. Xiao, “Recent advances in document summarization,” Knowl. Inf. Syst., vol. 53, no. 2, pp. 297–336, 2017.
[6] M. Gambhir, V. Gupta, M. Gambhir, and V. Gupta, “Recent automatic text summarization techniques: a survey,” Artif. Intell. Rev., vol. 47, no. 1, pp. 1–66, Jan. 2017.
[7] R. Mishra et al., “Text summarization in the biomedical domain: A systematic review of recent research,” J. Biomed. Inform., vol. 52, pp. 457–467, Dec. 2014.
[8] Y. Liu, T. Safavi, A. Dighe, and D. Koutra, “Graph Summarization Methods and Applica-tions: A Survey,” ACM Comput. Surv, vol. 51, 2018.
[9] “Unified Medical Language System (UMLS).”
[10] O. Bodenreider, “The Unified Medical Language System (UMLS): integrating biomedical terminology,” Nucleic Acids Res., vol. 32, no. 90001, pp. 267D – 270, 2004.
[11] “MetaMap - A Tool For Recognizing UMLS Concepts in Text.” [Online]. Available: https://metamap.nlm.nih.gov/. [Accessed: 25-Apr-2019].
[12] M. Basaldella, L. Furrer, C. Tasso, and F. Rinaldi, “Entity recognition in the biomedical domain
using a hybrid approach,” *J. Biomed. Semantics*, vol. 8, no. 1, p. 51, Nov. 2017.

[13] L. Furrer, A. Jancso, N. Colic, and F. Rinaldi, “OGER++: hybrid multi-type entity recognition,” *J. Cheminform.*, vol. 11, no. 1, p. 7, Dec. 2019.

[14] C. Rahmede, J. Iacovacci, A. Arenas, and G. Bianconi, “Centralities of nodes and influences of layers in large multiplex networks,” *J. Complex Networks*, vol. 6, no. 5, pp. 733–752, 2018.

[15] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, and Y. Artzi, “BERTScore: Evaluating Text Generation with BERT,” Apr. 2019.

[16] C. Y. Lin, “Rouge: A package for automatic evaluation of summaries,” *Proc. Work. text Summ. branches out (WAS 2004)*, no. 1, pp. 25–26, 2004.

[17] T. He et al., “ROUGE-C: A Fully Automated Evaluation Method for Multi-document Summarization.”

[18] K. Spärck Jones, “Automatic summarising: The state of the art,” *Inf. Process. Manag.*, vol. 43, no. 6, pp. 1449–1481, Nov. 2007.

[19] V. McCargar, “Statistical Approaches to Automatic Text Summarization,” *Bull. Am. Soc. Inf. Sci. Technol.*, vol. 30, no. 4, pp. 21–25, Jan. 2005.

[20] Y. Ko and J. Seo, “An effective sentence-extraction technique using contextual information and statistical approaches for text summarization,” *Pattern Recognit. Lett.*, vol. 29, no. 9, pp. 1366–1371, 2008.

[21] H. P. Edmundson, “New methods in automatic extracting,” *J. Assoc. Comput. Mach.*, vol. 16, no. 2, pp. 264–285, Apr. 1969.

[22] C. Burges et al., “Learning to rank using gradient descent,” in *ICML 2005 - Proceedings of the 22nd International Conference on Machine Learning*, 2005, pp. 89–96.

[23] S. Harabagiu and F. Lacatusu, “Topic themes for multi-document summarization,” 2005.

[24] J. Kupiec, J. Pedersen, and F. Chen, “Trainable document summarizer,” in *SIGIR Forum (ACM Special Interest Group on Information Retrieval)*, 1995, pp. 68–73.

[25] J. M. Conroy and D. P. O’leary, “Text summarization via hidden Markov models,” in *SIGIR Forum (ACM Special Interest Group on Information Retrieval)*, 2001, pp. 406–407.

[26] J. D. Schlesinger et al., “Understanding Machine Performance in the Context of Human Performance for Multi-Document Summarization,” 2002.

[27] R. Nallapati, B. Zhou, C. N. dos santos, C. Gulcehre, and B. Xiang, “Abstractive Text Summarization Using Sequence-to-Sequence RNNs and Beyond,” Feb. 2016.

[28] J. Cheng and M. Lapata, “Neural summarization by extracting sentences and words,” in *54th Annual Meeting of the Association for Computational Linguistics, ACL 2016 - Long Papers*, 2016, vol. 1, pp. 484–494.

[29] Z. Cao, F. Wei, S. Li, W. Li, M. Zhou, and H. Wang, “Learning summary prior representation for extractive summarization,” in *ACL-IJCNLP 2015 - 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, Proceedings of the Conference*, 2015, vol. 2, pp. 829–833.

[30] W. C. Mann and S. A. Thompson, “Rhetorical Structure Theory: Toward a functional theory of text organization,” *Text*, vol. 8, no. 3, pp. 243–281, 1988.

[31] D. Marcu and D. Marcu, “Discourse Trees Are Good Indicators of Importance in Text,” *Adv. Autom. TEXT Summ.*, pp. 123–136, 1999.

[32] A. U. Khan, S. Khan, and W. Mahmood, “MRST : A New Technique for Information,” *Second
world enformatika Conf. WEC’05, pp. 639–642, 2007.

[33] G. Erkan and D. R. Radev, “LexRank: Graph-based lexical centrality as salience in text summarization,” J. Artif. Intell. Res., vol. 22, pp. 457–479, 2004.

[34] K. Sarkar, K. Saraf, and A. Ghosh, “Improving graph based multidocument text summarization using an enhanced sentence similarity measure,” 2015 IEEE 2nd Int. Conf. Recent Trends Inf. Syst. ReTIS 2015 - Proc., pp. 359–365, Aug. 2015.

[35] R. Mihalcea and P. Tarau, “TextRank: Bringing order into texts,” Proc. EMNLP, vol. 85, pp. 404–411, 2004.

[36] E. Baralis, P. Torino, L. Cagliero, and A. Fiori, “Multi-document summarization exploiting frequent itemsets Categories and Subject Descriptors,” Proc. 27th Annu. ACM Symp. Appl. Comput., pp. 782–786, 2012.

[37] E. Baralis, L. Cagliero, A. Fiori, and P. Garza, “MWI-Sum: A Multilingual Summarizer Based on Frequent Weighted Itemsets,” ACM Trans. Inf. Syst., vol. 34, no. 1, pp. 1–35, 2015.

[38] J.-P. Qiang, P. Chen, W. Ding, F. Xie, and X. Wu, “Knowledg e-Based Systems Multi-document summarization using closed patterns,” Knowledge-Based Syst., vol. 99, pp. 28–38, 2016.

[39] L. Plaza, A. Díaz, and P. Gervás, “A semantic graph-based approach to biomedical summarisation,” Artif. Intell. Med., vol. 53, no. 1, pp. 1–14, 2011.

[40] G.-W. Kim and D.-H. Lee, “Personalised health document summarisation exploiting Unified Medical Language System and topic-based clustering for mobile healthcare,” J. Inf. Sci., p. 016555151772298, Aug. 2017.

[41] L. Plaza, “Comparing different knowledge sources for the automatic summarization of biomedical literature,” J. Biomed. Inform., vol. 52, pp. 319–328, 2014.

[42] M. Moradi and N. Ghadiri, “Quantifying the informativeness for biomedical literature summarization: An itemset mining method,” Comput. Methods Programs Biomed., vol. 146, pp. 77–89, Jul. 2017.

[43] M. Nasr Azadani, N. Ghadiri, and E. Davoodijam, “Graph-based biomedical text summarization: An itemset mining and sentence clustering approach,” J. Biomed. Inform., vol. 84, pp. 42–58, Aug. 2018.

[44] L. H. Reeve, H. Han, and A. D. Brooks, “BioChain: lexical chaining methods for biomedical text summarization,” Sac, pp. 180–184, 2006.

[45] H. D. Menéndez, L. Plaza, and D. Camacho, “A genetic graph-based clustering approach to biomedical summarization,” Proc. 3rd Int. Conf. Web Intell. Min. Semant. - WIMS ’13, p. 1, 2013.

[46] L. Tang, X. Wang, and H. Liu, “Community detection via heterogeneous interaction analysis,” Data Min. Knowl. Discov., vol. 25, no. 1, pp. 1–33, 2012.

[47] G. Kondrak, “N-Gram Similarity and Distance,” Springer, Berlin, Heidelberg, 2005, pp. 115–126.

[48] L. Plaza and J. Carrillo-de-Albornoz, “Evaluating the use of different positional strategies for sentence selection in biomedical literature summarization,” BMC Bioinformatics, vol. 14, no. 1, p. 71, Dec. 2013.

[49] J. Devlin, M.-W. Chang, K. Lee, K. T. Google, and A. I. Language, “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.”

[50] “Document Understanding Conferences.” [Online]. Available: https://duc.nist.gov/. [Accessed: 01-Dec-2019].

[51] “Text Analysis Conference (TAC).” [Online]. Available: https://tac.nist.gov/. [Accessed: 01-Dec-2019].
[52] G. Balikas, A. Krithara, I. Partalas, and G. Paliouras, “BioASQ: A Challenge on Large-Scale Biomedical Semantic Indexing and Question Answering,” Springer, Cham, 2015, pp. 26–39.

[53] P. Vision, G. Paliouras, and A. Krithara, “BioASQ.”

[54] L. Reeve, H. Han, and A. D. Brooks, “BioChain,” Proc. 2006 ACM Symp. Appl. Comput. - SAC ’06, no. August 2014, p. 180, 2006.

[55] M. Moradi, “CIBS: A biomedical text summarizer using topic-based sentence clustering,” J. Biomed. Inform., vol. 88, pp. 53–61, Dec. 2018.

[56] E. Lloret, L. Plaza, and A. Aker, “The challenging task of summary evaluation: an overview,” Lang. Resour. Eval., pp. 1–48, 2017.

[57] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding,” Oct. 2018.

[58] D. Kim et al., “A Neural Named Entity Recognition and Multi-Type Normalization Tool for Biomedical Text Mining,” IEEE Access, vol. 7, pp. 73729–73740, 2019.

[59] M. Bada et al., “Concept annotation in the CRAFT corpus,” 2012.

[60] D. Miller, “Leveraging BERT for Extractive Text Summarization on Lectures,” Jun. 2019.