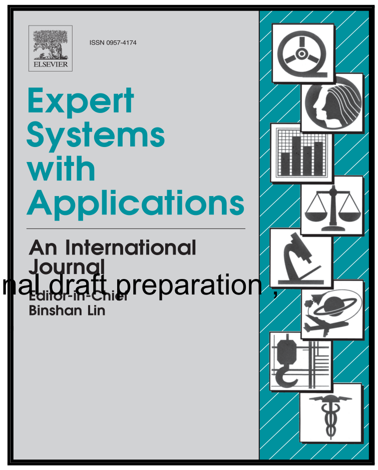

Combine Clustering and Frequent Itemsets Mining to Enhance Biomedical Text Summarization

Oussama ROUANE ConceptualizationMethodologySoftwareWriting- Original draft preparation ,
Hacene BELHADEF SupervisionValidationWriting- Reviewing and Editing ,
Mustapha BOUAKKAZ SupervisionWriting- Reviewing

PII: S0957-4174(19)30396-3
DOI: https://doi.org/10.1016/j.eswa.2019.06.002
Reference: ESWA 12714

To appear in: *Expert Systems With Applications*

Received date: 20 October 2018
Revised date: 31 May 2019
Accepted date: 2 June 2019

Please cite this article as: Oussama ROUANE ConceptualizationMethodologySoftwareWriting- Original draft prepar
Hacene BELHADEF SupervisionValidationWriting- Reviewing and Editing , Mustapha BOUAKKAZ SupervisionWr
Combine Clustering and Frequent Itemsets Mining to Enhance Biomedical Text Summarization, *Expert Systems With Applications* (2019), doi: https://doi.org/10.1016/j.eswa.2019.06.002

Highlights

- A new biomedical text summarization based on clustering and frequent itemsets mining
- Biomedical texts are represented using concepts instead of terms
- This combination enhances the quality of the generated summaries
- The clustering has a crucial impact on the discovered frequent itemsets
- Contribute all clusters in the sentence selection step yields to better performances

Combine Clustering and Frequent Itemsets Mining to Enhance Biomedical Text Summarization

Oussama ROUANE [a, *], Hacene BELHADEF[a], Mustapha BOUAKKAZ[b],

[a] *NTIC Faculty, University of Constantine 2 - Abdelhamid Mehri, Constantine, 25016, Algeria*
[b]*Department of computer science, Faculty of sciences, University of Laghouat , Laghouat, 03000, Algeria*

[*]Corresponding author
Email addresses:
oussama.rouane@univ-constantine2.dz, hacene.belhadef@univ-constantine2.dz, m.bouakkaz@lagh-univ.dz

Abstract

Text summarization has become an important research area, especially in the biomedical domain, where information overload is a major problem. In this paper, we propose a novel biomedical text summarization system that combines two popular data mining techniques: clustering and frequent itemset mining. Biomedical paper is expressed as a set of biomedical concepts using the UMLS metathesaurus. The K-means algorithm is used to cluster similar sentences. Then, the Apriori algorithm is applied to discover the frequent itemsets among the clustered sentences. Finally, the salient sentences from each cluster are selected to build the summary using the discovered frequent itemsets. For the evaluation step, we selected randomly 100 biomedical papers from the *BioMed Central database full-text*, and we evaluated the performances of our system by comparing the resulting summaries with the abstracts of these papers using the ROUGE metrics in term of recall, precision, and F-measure. We also compared the obtained summaries with those achieved by five well-known summarizers: *TextRank, TextTeaser, SweSum, ItemSet Based Summarizer, Microsoft AutoSummarize*, and two baselines: summarization using only the frequent itemsets mining (FRQ-CL), and summarization using only the clustering (CL-FRQ). The results demonstrate that this combination can successfully enhance the summarization performances, and the proposedsystem outperforms other tested summarizers.

Keywords: Biomedical text summarization, Biomedical concepts, Clustering, Frequent itemsets mining, ROUGE metrics

1. Introduction

The development of the World Wide Web, especially in the last two decades has led to an exponential growth of online information. This is also the case in the biomedical domain, e.g., MEDLINE[1] (med), the largest biomedical bibliographic text database contains about 25 million references of journal articles in life sciences that concentrate on biomedicine. However, researchers in this area encountered major difficulties to access to the desired information quickly and efficiently (Afantenos, Karkaletsis & Stamatopoulos, 2005). Text summarization is a promising technique that could aid them to obtain the core information in a given subject by *"condensing the source text with preserving the main ideas from it"* (Mishra, Bian, Fiszman, Weir, Jonnalagadda, Mostafa & Del Fiol, 2014). i.e., text summarization could aid biologists to find general information about a biological concept, e.g., a gene or a disease, from one or multiple documents without reading the entire documents (Shang, Li, Lin & Yang, 2011). Medical doctors frequently use summaries to identify patient's treatments quickly, and to reducing diagnosis time (Reeve, Han, Nagori, Yang, Schwimmer & Brooks, 2006b). Furthermore, summaries are also used to improve indexing and categorization of biomedical papers when it is used as a substitution of abstracts when they are not available (Gay, Kayaalp & Aronson, 2005). The majority of text summarization methods do not consider the characteristics of the domain or the type of documents. They mostly work with units extracted directly from the document itself, such as terms, sentences or paragraphs, etc. Then they rely on data mining or information retrieval techniques to analyze effectively this data. However, in the biomedical domain like any other specific domain, these techniques may not seem to be working well because the literature of this domain has its properties and they should be considered during the summarization process. For this reason, researchers in this domain used domain knowledge resources like ontologies, thesaurus, and taxonomies, etc… to provide meaning to biomedical texts, and then linking

[1] U.S. National Library of Medicine < https://www.nlm.nih.gov/bsd/medline.html>

information within each text to specifications contained in these resources using a markup language and return concepts that express the semantic meaning of texts. Sometimes, they enrich these concepts with their semantic types and link them using the semantic relationships i.e., synonymy, hypernymy, co-occurrence, etc. To build a graph that accurately captures the meaning of the text to be summarized (Menéndez, Plaza & Camacho, 2013). In this work, we combine two data mining techniques: clustering and frequent itemsets mining to produce single summaries (a summary per document), and we treat each document as a set of biomedical concepts instead of terms. We validate our system against five summarizers on a 100 randomly selected biomedical papers from the BioMed Central full-text database. We perform a broad set of comparisons using the ROUGE toolkit in term of precision, recall, and F-measure.

The rest of the paper is organized as follow: Section 2 introduces background on the domain and the related works. In Section 3, we give a brief representation of the parts of the system. Section 4 describes the process of experiments, and in section 5 we present the comparison results against the other tested summarizers with brief discussions. Finally, a conclusion is given in the final section.

2. Background and related work

In this section, we offer basic concepts of the domain, and then we give an overview of the previous works. A summary could be defined as *"a text that is produced from one or more texts, that contains a significant portion of the information in the original text(s), and that is no longer than half of the original text(s)"* (Hovy, 2005). The researchers classified text summarization systems into different factors that should be considered during the summarization process (Afantenos et al., 2005). i.e., text summarization systems can be **single or multiple**: that means if they consider only one document as input otherwise for multiple documents, **monolingual or**

multilingual: in which the input and output are in the same language, or are in different languages, **informative or indicative**: if they return sentences or they return keywords, **generic or user-oriented**: that means if the generated summary contains the important topics discussed in a document or the generated summary respond to a user's question represented by a query that contains a set of keywords, **general or domain-specific**: that entails if they treat document independently to its domain and its type, or they are centred upon various domains like biomedical or newspapers, **extractive or abstractive**: if they produce summaries by picking up most important sentences from the original document, or the selected sentences are combined coherently and compressed to exclude unimportant sections.

In domain-independent text summarization systems, the first work started earlier in 1969 based on what we called *the Edmundsonian paradigm* (Edmundson & P., 1969). This paradigm is based on a superficial analysis of the text like the frequency of terms, the position of sentences, the presence of cue words or phrases, and the similarity of sentences to the titles, etc. These features could also be combined using a linear function to calculate a single score to each sentence, and highly scoring sentences are used to construct the summary. Much progress has not been made because of the unavailability of computational machines. With the upcoming of the internet as a major resource of information, the work on text summarization gained a new interest in 1990. Other successful techniques based on graph representations are also proposed, these techniques in generally represent documents as graphs, where nodes correspond to text units such as words, sentences or even paragraphs, and the edges represent similarity measures between them (e.g., Euclidean distance (Anton, 1994), Jaccard similarity (Jaccard, 1901), Cosine similarity (Singhal, 2001)). Once the graph is created, the important nodes are determined in the graph using different techniques in the literature, and then, the corresponding units are extracted to build the summary, e.g., MEAD is a multi-document summarization system developed by (Radev, Jing & Budzikowska, 2000) based on a technique called centroid based summarizer, which uses the centroid of clusters to identify the sentences that are most central to the topics of these clusters. These centroids are words that have a TF-IDF value (term frequency-inverse document frequency) (Sparck Jones, 1972) above a predefined threshold. The system ranks candidate sentences by calculating their similarities to the centroids, their position values, and their overlaps to the titles. The sentences selection is constrained by the summary length and avoided by the redundancy to the previous ones. LexRank (Radev, 2004) is another well-known multi-document summarizer that identified the most salient sentences in a given corpus of a document using a graph-based ranking model. Firstly, the corpus is represented as an undirected weighted graph where nodes represent sentences as a vector of TF-IDF values, and edges are labelled by the cosine similarities between them. Only edges that have similarities above a predefined threshold are drawn. Then, the PageRank algorithm (Page, Brin, Motwani & Winograd, 1998) is applied to rank sentences in the graph. Finally, a post-processing step built up a summary by adding sentences in their original order but avoiding any sentences that are too similar to the ones that are already added to the summary. LexRank could also be integrated into the MEAD system as a feature to calculate the final score of each sentence. A very similar algorithm is Textrank (Mihalcea & Tarau, 2004), Textrank is applied to single document summaries and it generates a set of keywords or key phrases. So the nodes are represented by keywords or sentences and edges are labelled by co-occurrences or by words overlaps respectively. Another field of techniques used frequent itemsets mining to extract the most informative sentences. These techniques represented document as a transactional data format where each sentence represent a transaction and tokens represent items within the transaction, e.g., ItemSum (Baralis, Cagliero, Jabeen & Fiori, 2012) are the first that tried to exploit frequent itemset mining technique in multi-document summarization. Firstly, they represented the document as a transactional data format, where transaction represents sentences and items represent distinct terms taken by the bag of words (BOW) of document representation, and each term has a support value that represents its frequency in the transactional data format. ItemSum automatically selects the minimal set of the most representative and not redundant sentences to include in the summary that best covers the itemset-based model using a greedy strategy combined with a relevance score that is calculated using the TF-IDF measure. (Baralis, Cagliero, Fiori & Garza, 2015) adopted weighted itemsets instead of traditional (unweighted) itemsets proposed in ItemSum (Baralis et al., 2012) to generate a multilingual document summary. This representation allows them to discriminate between the high and the least relevant terms. Term weights measure term relevance in the analyzed corpus, and it is calculated using a variant of the TF-IDF statistics. To discriminate between sentences that contain and those that do not contain relevant information, MWI-Sum map documents to a weighted transactional data format, and then a frequent weighted itemset mining algorithm is used from the preprocessed data. Finally, the system selects the minimum number of representative sentences that best covers the previously extracted itemsets. Another method proposed by (Baralis, Cagliero, Mahoto & Fiori, 2013) called GraphSum. GraphSum is a multi-document summarization system that is based on building and evaluating a graph of correlated terms. The nodes in this graph represent document terms, and an edge connects every two nodes if they frequently co-occur in the corpus, the weight of the edge will be indicated by the strength of the correlation between these pair of nodes (either positively or negatively). The correlations are extracted using an association rule mining algorithm by implying a minimum support threshold and a maximum negative and a minimum positive correlations threshold. Finally, a variant of PageRank algorithm is applied to rank nodes in the graph, and a greedy strategy is used to determine the best subset of sentences that best cover the model.

However, the biomedical domain like any other scientific domain is difficult to understand for humans because it has many singularities, e.g., the majority of biomedical terms are compound: a compound word is an expression made up of more than one word, e.g., *human being, central nervous system, and collar-bone* (Dzuganova, 2013). The frequent using of acronyms and abbreviations is another critical factor; an abbreviation is a shortened form of a word or phrase (Dzuganova, 2013) while an acronym is initialism pronounced as single word e.g., the acronym*: NF-kB* can simplify the initialism pronounced as single word *nuclear factor kappa-light-chain-enhancer of activated B cells*. The biomedical literature also suffers from the variety of synonyms: words design the same meaning, e.g., *myocardial infarction, heart attack, and MI* (Moradi & Ghadiri, 2017). The vocabulary of this domain also addresses the problem of polysemy (Shortliffe & Cimino, 2014), i.e., the same term corresponds to different meanings according to the context in which is used. In this case, words become ambiguous, e.g., in the genetic domain the word *'to'*, is a very frequent English word, corresponds to two different "*Drosophila genes"* and to the "*mouse gene tryptophan 2,3-dioxygenase"*. Polysemy is resolved at two-level techniques: named entity recognition and word sense disambiguation. The presence of elision that is defined as a phrase with missing words complicates the recognition of the meaning of words, e.g., the phrase "*I have a temperature*" as written by a patient online can mean I have a fever, but text it "*I have a temperature of 98.6*" that means no fever. In this case, external biomedical knowledge is required to infer the presence of fever or not from just a numerical value (Ben Aouicha & Hadj Taieb, 2016). Finally, another important property is that the biomedical vocabulary is highly dynamic in the influx of new terms, e.g., *new drug names*, but also sometimes new disease names, like *SARS* and *H1N1*. This phenomenon also led to the neologism problem, which is represented by newly coined words that would not be expected in a dictionary because they are not universal words (Friedman & Elhadad, 2014).

In recent years, there is a big challenge in the biomedical NLP communities to develop publicly available knowledge resources and tools to enable to machines understand biomedical texts and to aid clinicians and researchers in this field to achieve their different tasks (Fleuren & Alkema, 2015). e.g., **Unified Medical Language System (UMLS)** (National & Us, 2009): is a compendium of over 100 controlled vocabularies related to the biomedical domain, and it can be observed as a comprehensive thesaurus or ontology of all biomedical concepts, it provides a mapping structure that could link all these terms and concepts among these vocabularies. UMLS also offers free software tools that facilitate different natural language processing tasks to understand the meaning of the medical language by computer systems. Currently, UMLS becomes the largest thesaurus in the biomedical domain, and it is maintained by NLM twice a year in May and November (Shams S). There exist three UMLS knowledge sources: the Metathesaurus, the Semantic Network, and the SPECIALIST Lexicon.

- **The Metathesaurus**: forms the backbone of the UMLS and it is created by unifying over 100 controlled vocabularies and classification systems like *CPT, ICD-10- CM, LOINC, MeSH, RxNorm, and SNOMED CT.* It is organized around concepts, each of which represents a meaning and is assigned a *Concept Unique Identifier(CUI)* e.g., the following CUIs are all associated with the term "*cold* ": *C0009443 'Common Cold', C0009264 'Cold Temperature' and C0234192 'Cold Sensation'* (Plaza, Stevenson & D'iaz, 2012).
- **The Semantic Network**: contains a set of broad categories (or semantic types) that provides consistent categorization of all the concepts in the Metathesaurus, which are linked with one another through semantic relationships, e.g., *the CUI C0009443 'Common Cold'* is classified in the semantic type *'Disease or Syndrome'*. There are 127 semantic types and 54 relationships in the UMLS semantic network (National & Us, 2009).
- **The specialist lexicon**: provides the lexicographic information needed for NLP Systems. It includes common English words and biomedical terms in the Metathesaurus. Each entry word or term contains the syntactic, morphological, and orthographic information (National & Us, 2009).

In this domain, researchers have proposed many text summarization systems that are based on domain knowledge resources, e.g., (Reeve, Han & Brooks, 2006a) used concept chains instead of lexical chains, to link semantically related concepts within a biomedical text. Firstly, the text is mapped to biomedical concepts using UMLS Metathesaurus and the obtained concepts are chained them their semantic types in the UMLS semantic network, strong chains are identified, and the most important sentences that contain the most frequent concepts in each chain are selected to form the summary. Another effort done by (Reeve, Han, Nagori, Yang, Schwimmer & Brooks, 2006c) in this work, they used a frequency distribution model of concepts and a context sensitivity approach to select sentences with minimum information redundancy. (Yoo, Hu & Song, 2007) proposed a graphical representation of documents and summarization method (CSUGAR). Firstly, they mapped each document into MeSH (Medical subject headings) descriptors and extending them with all hypernym relationships using the MeSH tree. Then, they constructed a graph for each document where nodes represent MeSH descriptors and edges represent hypernym and co-occurrence relationships and labeled by the strength of the relationships between them. After that, they merged all document graphs into one scale-free graph, and they applied a clustering algorithm to grouping similar MeSH descriptors, documents are assigned to clusters using both their graph representation and the scale-free graph. In the sentence selection step, they constructed a text semantic interaction network (TSIN) of sentences where nodes represent sentences and edges represent the similarity between them, and the strength represent the edit distance between sentences graphs. Finally, sentences that have the maximum betweenness centrality in the graph have been selected to be in the final summary. (Plaza & D'iaz, 2010) studied the effect of lexical ambiguity in the knowledge source on semantic biomedical text summarization approaches by incorporating a word sense disambiguation technique (WSD). To this end, they represented documents as graphs constructed from concepts and their relations derived from the UMLS. Then they applied a degree-based clustering algorithm to find different topics within these documents. They proved that the application of WSD technique to the output of MetaMap improves the quality of the generated summaries significantly. (Plaza, Díaz & Gervás, 2011) addressed the process of the identification of salient sentences in biomedical texts as a graph, all the concepts, and their semantic relationships are returned and merged into one semantic scale-free graph. Then, they applied a degree-based clustering technique to identify various topics in the text. In the sentence selection step, they investigated different heuristics to generate diverse types of summaries. They also noted that they determined some UMLS semantic types that are very generic, and concepts belonging to them should be discarded because they have been founded to be excessively broad and do not contribute to the summary generation. As the best of our knowledge (Moradi & Ghadiri, 2017) are the first that proposed a biomedical text summarization system that combines frequent itemset mining technique with a conceptual representation of biomedical texts. To this aim, they mapped the text into UMLS concepts to construct a transactional data format, and then they applied the Apriori algorithm to discover the frequent itemsets that represent the main subtopics in the text. Finally, the generated frequent itemsets based model is used to score sentences, and the *N* top scoring sentences are put together to form the final summary.

3. Method

This paper presents a novel biomedical summarization system based on a combination of clustering and frequent itemsets mining with a conceptual representation of biomedical text to enhance the quality of the generated summaries. Therefore, we utilize for the clustering task: the K-means algorithm (Macqueen, 1967), and for the mining task: the Apriori algorithm (Agrawal & Srikant, 1994), due to their simplicity and efficiency at the same time. The main objective of this research is to show the effect of the combination of clustering and the frequent itemsets mining techniques enhances the quality of the generated summaries. The proposed system consists of five components: *document pre-processing; sentence representation; sentence clustering; frequent itemsets mining, and sentence evaluation and selection*, as shown in figure 1, where each step is discussed in detail below:

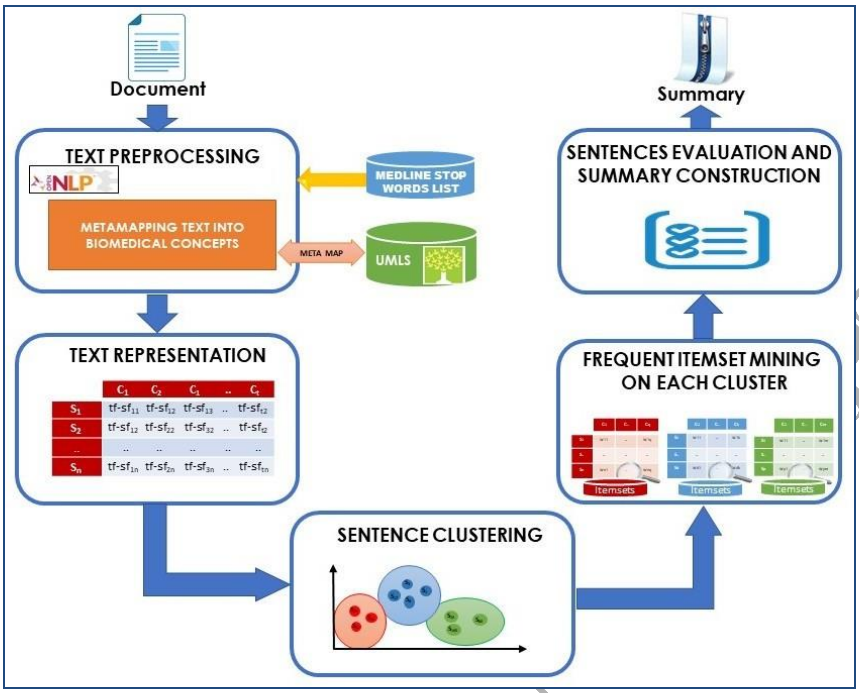
Document
TEXT PREPROCESSING
NLP
METAMAPPING TEXT INTO BIOMEDICAL CONCEPTS
MEDLINE STOP WORDS LIST
META MAP
UMLS
TEXT REPRESENTATION
C1 C2 C3 .. Ct
S1 tf-sf11 tf-sf12 tf-sf13 .. tf-sft2
S2 tf-sf12 tf-sf22 tf-sf32 .. tf-sft2
Sn tf-sf1n tf-sf2n tf-sf3n .. tf-sftn
SENTENCE CLUSTERING
FREQUENT ITEMSET MINING ON EACH CLUSTER
Itemsets
Itemsets
Itemsets
SENTENCES EVALUATION AND SUMMARY CONSTRUCTION
Summary

Figure 1: The global architecture of the proposed system.

3.1. Document pre-processing

To prepare the input document to the following tasks, our system requires several pre-processing steps. The following points illustrate the headlines of these techniques:

3.1.1. Removing irrelevant sections: sections that are considered unimportant for inclusion in the summary are removed manually. We can specify these sections according to the input text and its logical structure. Since our evaluation corpus consists of a set of biomedical papers, i.e., we remove competing interests, acknowledgments, references, headings, images, figures, tables, and titles, etc. We only reserve the abstracts and body sections from each document to process them.

3.1.2. Splitting text into sentences: in this step, we split the text into a set of sentences by detecting terminators. After finishing this step, each document is represented as a set of sentences denoted by $D = \{ S_1, S_2 \ldots S_N \}$.

3.1.3. Tokenizing sentences: each sentence $s \in D$ is expressed as a set of tokens, denoted by $S = \{w_1, w_2 \ldots w_K\}$. Besides, we turn each token to lowercase to facilitate the subsequent processing tasks.

3.1.4. Removing stop words: Stop words are words having no meaning in the text (prepositions, pronouns, etc.). Since our evaluation corpus consists of a corpus of biomedical papers, we use the stop words list related to the biomedical domain given by the PubMed Search engine Pubmed (2005) to remove them from the words sets generated by the precedent step.

3.1.5. Concept recognition: this step consists of mapping the text into concepts from the UMLS Metathesaurus and their corresponding semantic types from the UMLS Semantic Network. We run the MetaMap[2] program over sentences and tokens. In particular, we use the version 2016 and the knowledge release of *UMLS version 2016AA*. We also invoke the WSD module *(-y flag)* to forces MetaMap to return a single concept mapping when a lexical ambiguity is encountered. However, when the returned concepts have the same semantic type, MetaMap may fail to return a unique concept. In this case, we select the first returned mapping.

3.1.6. Removing very generic concepts: we remove concepts from very generic semantic types. These semantic types are *"qualitative concept, quantitative concept, temporal concept, functional concept, idea or concept, intellectual product, mental process, spatial concept, and language"* determined on the empirical study of Plaza et al. (2011) that are not important and do not contribute to the summarization process.

[2] MetaMap - A Tool For Recognizing UMLS Concepts in Text < https://metamap.nlm.nih.gov/ >

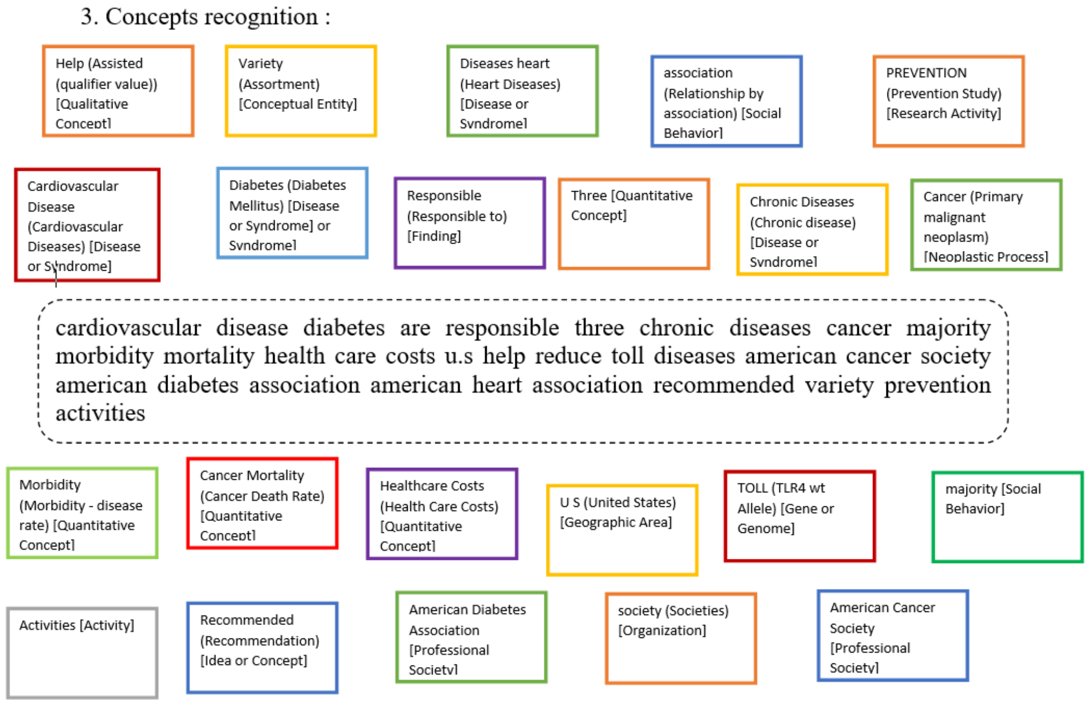
3. Concepts recognition :
Help (Assisted (qualifier value)) [Qualitative Concept]
Variety (Assortment) [Conceptual Entity]
Diseases heart (Heart Diseases) [Disease or Syndrome]
association (Relationship by association) [Social Behavior]
PREVENTION (Prevention Study) [Research Activity]
Cardiovascular Disease (Cardiovascular Diseases) [Disease or Syndrome]
Diabetes (Diabetes Mellitus) [Disease or Syndrome] or Syndrome]
Responsible (Responsible to) [Finding]
Three [Quantitative Concept]
Chronic Diseases (Chronic disease) [Disease or Syndrome]
Cancer (Primary malignant neoplasm) [Neoplastic Process]
cardiovascular disease diabetes are responsible three chronic diseases cancer majority morbidity mortality health care costs u.s help reduce toll diseases american cancer society american diabetes association american heart association recommended variety prevention activities
Morbidity (Morbidity - disease rate) [Quantitative Concept]
Cancer Mortality (Cancer Death Rate) [Quantitative Concept]
Healthcare Costs (Health Care Costs) [Quantitative Concept]
U S (United States) [Geographic Area]
TOLL (TLR4 wt Allele) [Gene or Genome]
majority [Social Behavior]
Activities [Activity]
Recommended (Recommendation) [Idea or Concept]
American Diabetes Association [Professional Society]
society (Societies) [Organization]
American Cancer Society [Professional Society]

Figure 2: Pre-processing step including removing stop words related to the biomedical domain, and the mapping of the sentence to biomedical concepts using the UMLS metathesaurus.

3.2. Text representation

The result of the previews step is a bag-of-concepts (BOC), in which each processed document D consists of a set of sentences $D = \{ S_1, S_2 ... S_N \}$, and each sentence contains a set of distinct concepts $S = \{C_1, C_2 ... C_K\}$. We use the vector space model to represent the document, where sentences in the document represented as vectors of features with associated weights (Salton, Wong & Yang, 1975). In our context, the features were derived from the concepts appearing in the document D. Let S_j be the jth sentence in the document D, which is represented as a vector of concepts with associated weights $(w_{1j}, w_{2j} ... w_{nj})$, where w_{ij} is the numeric weight of the concept i in the sentence j, and n is the total number of distinct concepts in the document D. We define the weight w_{ij} as the product of two values: *"Concept Frequency (CF)"* and *"Sentence Frequency (SF)"*.

Concept Frequency (CF_{ij}): is the proportion of the number of occurrences of the concept C_i in the sentence S_j to the total number of concepts in the same sentence S_j:

$$CF_{ij} = \frac{n_{ij}}{|N_j|} \qquad (1)$$

Where n_{ij} is the number of occurrences of the i^{th} concept in the j^{th} sentence, and $|N_j|$ is the total number of concepts in the j^{th} sentence.

Concepts contained in each sentence do not have the same importance in all sentences. For this reason, we assign to each concept a complement score that denotes its importance in the whole document. To this end, we introduce the *Sentence Frequency weight (SF)*: we weigh each concept based on its distribution in the whole document as follow:

$$SF_j = \frac{|\{S_j | S_j \in D \wedge C_i \in S_j\}|}{|D|} \qquad (2)$$

Where $|S_j| S_j \in D \wedge C_i \in S_j|$ is the number of sentences that contain the concept C_i, and $|D|$ is the total number of sentences in the document *D*. Finally, the *CF-SF* is simply the product of the two values as follow:

$$CF - SF(C_{ij}) = \frac{n_{ij}}{|N_j|} \times \frac{|S_j| S_j \in D \wedge C_i \in S_j|}{|D|} \qquad (3)$$

The key idea of the weight *Concept Frequency Sentence Frequency (CF-SF)*, which we have derived it from the well-known TF-IDF weighting measure (Sparck Jones, 1972) is to determine the score of a concept according to its frequency in a sentence and its distribution through the sentences in the whole document. Since we treat one document in a step, the score of a concept increases if it occurs in a large number of sentences. In other words, the more a concept is frequent in the whole document, the more it is important. This assumption is evident because we cope with the collection of sentences belonging to the same document that ranges over the same topic.

3.3. Sentence clustering

Since we represented sentences as vectors in a vector space, we need to cluster sentences that contain similar information. Similar sentences in a document tend to include similar concepts weights. In this work, the cosine similarity is the most appropriate metric to compute the similarity between two sentence vectors, and it is often used in information retrieval to calculate the similarity between documents and queries (Tan, Steinbach & Vipin Kumar, 2006). The cosine similarity between two sentences S_i and S_j represented by CF-SF vectors is computed using the formula as below:

$$Similarity(S_i, S_j) = \frac{\sum_{i,j=0}^{n} S_i \times S_j}{\sqrt{\sum_{i=0}^{n} S_i^2} \times \sqrt{\sum_{j=0}^{n} S_j^2}} \quad (4)$$

We use the K-means algorithm to achieve the clustering step. K-means is the most well-known algorithm belongs to the set of partitioning clustering techniques. In our work, we aim to divide a set of sentences, based on their features into k-predefined clusters. The idea is to specify k-centroids one for a cluster. Then the centroid of each cluster is formed in such a way that it is mostly closed (in terms of similarity) to all sentences in that cluster. We note that a proper initialization of the centroids is essential. We assign random centroids, and we run the algorithm multiple times to determine the best set of centroids.

3.4. Frequent itemsets mining

According to our context, we have a set of clusters, and each cluster is composed of a set of sentences C_i = {$S_{i1 \ldots} S_{ik}$}, and each one composed of a set of distinct concepts. The BOC (bag of concepts) representation of the j^th sentence belonging to the cluster C_i is the set of all concepts occurring in S_j. To adjust the clustered sentences to the transactional data format, we consider each sentence as a transaction whose items are distinct concepts taken from its BOC representation (i.e., $\mathrm{tr_j} = \{C_{j1} \ . \ . \ . \ C_{jl}\}$ where $tr_j \subseteq S_j$. A transactional representation T of the cluster C_j is the union of all transactions belonging to C_j. To mine the frequent itemsets in each cluster, we employ the Apriori Algorithm. Apriori is initially used for association rule mining, but in our work, we use it to extract the frequent itemsets, then we involve the extracted frequent itemsets in the process of scoring and selecting the most important sentences that capture the maximum information in each cluster. The input of the algorithm is the transactional data format T of clustered sentences and the minimum support threshold. The output is a set of discovered frequent itemsets from each cluster where each itemset contains a set of correlated concepts.

3.5. Sentences evaluation and selection

We remember that extractive text summarization systems work by scoring sentences in the original document and salient sentences that have higher scores are selected to generate the final summary. In this work, we do the same thing, and we put higher scoring sentences in the final summary based on their original order in the document to retain their consecutive meaning. Our system exploits the frequent itemsets models generated from clusters to evaluate and select the most salient sentences using different scoring strategies.

3.5.1. Sentence evaluation

In this work, we score sentences belong to each cluster using the generated frequent itemset models. These models contain the most frequent itemsets with their support values respectively. The support value of each itemset indicates how much the item set is significant. It means that in the comparison between two itemsets extracted from a given cluster. An itemset is assumed more valuable if it has a higher support value than other itemsets. In another hand, if we have two frequent itemsets that have high and equal support value but one is of size N and another of size K where $N > K$. We consider that the frequent itemset of size N is more valuable and important because it includes more frequent items. Hence, it contains supplementary information than the frequent itemset of size *K*. In this case, we can also hypothesize that the size of a frequent itemset is another important factor. Therefore, we define the score of each sentence as the sum of the support values of the frequent itemsets that are covered by a sentence multiplied by their lengths respectively:

$$Score(S_i) = \sum (support(FI_j) \times |FI_j|) \qquad (5)$$

Where *support(FI_j)* is the support value of the frequent itemset FI_j covered by the sentence S_i, and $|FI_j|$ is the size of the same frequent itemset.

3.5.2. Sentence selection

After scoring sentences in each cluster, we sort the top N scoring sentences, where *N* is a number specified by a user and determine the compression rate of the original document. In this subsection, we propose two sentences selection strategies to determine the number of selected sentences from each cluster to construct the summary:

1. **Heuristic 1**: We consider that the cluster with the maximum number of sentences (i.e., the global cluster) represents the main subtopic of the document. We select the top N ranked sentences from only this cluster.
2. **Heuristic 2**: In this heuristic, we contribute all clusters in the final summary by considering the percentage of their sizes to the total number of sentences in a document (i.e., the document size). Thus, for each cluster, the top n_i highly scoring sentences are selected from each cluster, where n_i is relative to the size of the cluster C_i. In this heuristic, we aim to include in the summary information about all the clusters in the document and we do not neglect any cluster in the final summary. We calculate the number of selected sentences using this formula:

$$n_i = \frac{N \times |C_i|}{|D|} \qquad (6)$$

Where N is the total number of sentences in the summary (determined by a compression rate), $|C_i|$ is the number of sentences in the cluster C_i (e.g., the size of the cluster C_i), and $|D|$ is the total number of sentences in the document D.

4. Process of experiments

The purpose of this section is to evaluate the performances of our system and to compare it with other summarizers. This process is accomplished in two steps: 1) a preliminary experiment is to find the best values for different parameters involved in the experiment and 2), the evaluation of the system against other summarizers using these values.

4.1. Evaluation measures

To evaluate the performances of our system, we used a classical method for automatic evaluation of summaries called ROUGE toolkit[3] (i.e., Recall-Oriented Understudy for Gisting Evaluation) (Lin, 2004). ROUGE includes a set of metrics that determine the quality of a summary, by counting the overlapping units, such as n-gram and word sequences between system-generated summaries and human-written summaries (gold summaries). The main advantages of ROUGE are its simplicity and its high correlation with human judgments. We note that we used the version **1.5.5** of ROUGE package, which contains 11 metrics. Here we mention just three metrics that we have used in our experiments: **ROUGE-1, ROUGE-2, and ROUGE-SU4**.

- **ROUGE-1 and ROUGE-2**: compute the 1-gram and the 2-grams common units between human-written summaries and system-

[3] https://rxnlp.com/rouge-2-0/#.XNiKzxRKjIU

generated summaries respectively.

- **ROUGE-SU4**: estimates the "skip bigrams", that is, pairs of units having intervening unit gaps no larger than four units.

For each metric, we present the graphs of the precision, the recall, and the f-measure. The recall in our context means how much of the system summary captures the references summaries. In the precision, what we necessarily measure is how much of the system summary was, in fact, pertinent or needed. The F-Measure is simply an average value between the recall and the precision.

4.2. Corpus

For the process of experiments, we selected a set of 100 biomedical papers from the BioMed Central full-text database randomly. We converted the PDF versions of these papers into a plain-text format to facilitate the processing tasks. We manually removed irrelevant parts like graphics, tables, figures, captions, citation references, and the bibliography, etc. We further split the resulting text into abstract and body sections. We considered the abstracts of papers as reference summaries, and we use the body sections as an input to our system. We compared the resulting summaries against the abstracts. This evaluation strategy has been frequently used in biomedical text summarization because to our knowledge, there is no corpus of reference summaries exists for biomedical papers until now.

4.3. Parameter settings

To simplify the process of experiments, we fix the number of clusters to four clusters. This value is inspired empirically, because we have seen that the majority of biomedical papers are composed of four parts: introduction, methods, results, and discussion. The sentences in each section are semantically more coherent than sentences belonging to other sections. Hence, we only vary the minimum support threshold to determine the best minsup according to ROUGE scores involved in the experiments. We generate all automatic summaries with a compression rate equal to 25% of the original document size. This choice is based on a well-known accepted heuristic of (Hovy, 2005) where the size of an informative summary should be among 15% and 35% of the size of the source text. Although the length of the abstracts in our corpus is on average about 10% of the length of documents, as shown in table 1, a larger size was preferred because the papers used in the experiments are rich in information. Moreover, we discovered that the generated summaries with small sizes are purely informative and suffers from many linguistic problems like coherency, anaphora, etc. We assess the statistical significance of our results using a Wilcoxon signed-rank test with a 95% confidence interval. The Wilcoxon signed-rank test is a non-parametric statistical hypothesis test used to compare two related samples to assess whether their population means ranks differ.

Table 1: Average number of sentences and words in body sections and abstracts in the corpus.

	Average number of sentences	Average number of words
Abstracts	9.77	213
Body sections	108.17	2731.8

4.4. Comparisons with other summarizers

To determine the efficiency of our system, we compare it against five summarizers (three research prototypes: *TextRank, TextTeaser, ItemSet Based Summarizer*, an online summarizers: *SweSum,* and a commercial application: *Microsoft AutoSummarize*). In addition, we implemented two baselines, the first one called **CL-FRQ**, in which, we only apply the clustering without using the frequent itemset mining, and from each resulting cluster we select the top n_i most similar sentences to generate the summary where n_i is proportional to the size of each cluster. In the second baseline called **FRQ-CL**, we apply the frequent itemset mining technique directly to the text without applying the clustering, and we select the most informative sentences that cover the most frequent itemsets from this model using the formula 5 to construct the summary.

5. Results and discussions

5.1. *Parameterizations results*

To assess the impact of the number of clusters and the minimum support thresholds on the performance of our summarizer, we realized a set of experiments on a separate small corpus that contains 20 papers. We found that the sentences in these papers follow a very standardized structure called the IMRAD (Introduction, Method, Results, and discussion) (Sollaci & Pereira, 2004). Table 2 resumes the average size of sections in our corpus. In the experiment, we found that when we set the number of clusters to four in the K-means algorithm, it usually produces a single large cluster and a variable number of small clusters, this result is almost like the real distribution of sentences in this experiment.

Table 2: Average sizes of sections compared to the average sizes of documents in the corpus

Sections	sizes
Background	08%
Methods	42%
Results	25%
Discussion	25%

On the same corpus, we have done another experiment to observe the effect of the clustering on the discovered frequent itemsets. Table 3 reports the average number of all frequent itemsets and the average number of k-itemsets, where k = 1...4, a k-itemsets is an itemset of size *k*. The average numbers are given for each tested support threshold, and whether when we have, or we do not have using the clustering before mining the frequent itemsets.

Firstly, we can observe from the table 3 that there is an inverse relationship between the number of the discovered frequent itemsets and the values of the minimum support thresholds. When we set a higher value of the minimum support threshold, we obtain a fewer number of frequent itemsets and vice versa, and this is an intuitive law in the frequent itemsets mining community. Moreover, another important note is that the clustering has a crucial impact on the number of discovered frequent itemsets. When we use the clustering before the frequent itemset mining, the average number of the discovered frequent itemsets and their average sizes increases significantly. For example, when we set the minsup to 0.1, the average number of the discovered frequent items is on **18.25 vs. to 7** frequent items when we do not use the clustering. In addition, the average number of the discovered frequent itemsets, with and without the clustering respectively, is **(7 vs. 5; for 1-itemsets), (2.25 vs. 1; for 2-itemsets), (0.25 vs. 0; for 3-itemsets)**. We conclude that the obtained frequent itemsets are more correlated when we use the clustering before the frequent itemsets mining.

Table 3: The average number of all frequent items and k-frequent itemsets with (+) and without (-) applying the clustering

Minsup	All frequent items		1-FI		2-FIs		3-FIs		4-FIs	
	- Clustering	+ Clustering	-Clustering	+Clustering	- Clustering	+ Clustering	- Clustering	+ Clustering	- Clustering	+ Clustering
0.02	306	41452	60	95.75	70	335.5	4	619.75	0	925.75
0.04	70	9948	26	42.75	19	105.75	0	179.75	0	272.5
0.06	31	44.5	14	17	7	10.75	0	2	0	0
0.08	25	27.5	10	12.25	6	6.5	0	0.75	0	0
0.1	7	18.25	5	7	1	2.25	0	0.25	0	0
0.12	5	9.75	5	5.5	0	1.75	0	0.25	0	0
0.2	0	5.5	0	2.75	0	1	0	0.25	0	0
0.3	0	0.75	0	0.75	0	0	0	0	0	0

5.2. Evaluation results

In this section, firstly, we present the results of the preliminary experiments, the parameterization of our system. Then, the results of the evaluations comparing our system with other summarizers will be presented later.

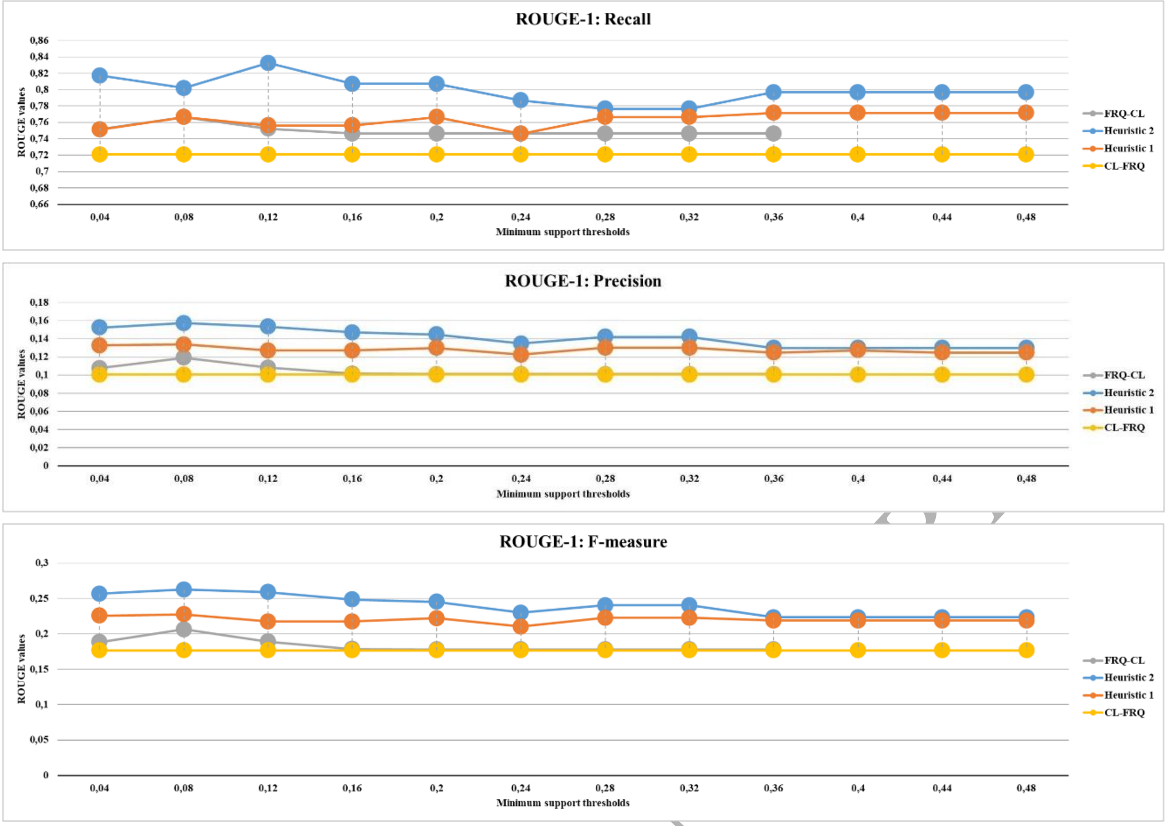

Figure 3: ROUGE-1 values in term of recall, precision, and F-measure

Figures 3, 4 and 5 show the execution of the system (including the two proposed heuristics), and the two baselines when we vary the minimum support thresholds and their effects in the quality of the generated summaries in term of recall, precision and f-measure of ROUGE. As we presented, our system uses two heuristics to select sentences from the clusters. The first heuristic consists of selecting sentences from the global cluster, where the second heuristic contributes all clusters to construct the summary. The first baseline (**CL-FRQ**) clusters similar sentences into k-clusters using the K-means algorithm and selects the most similar sentences from each cluster to build the final summary. We also note that this baseline does not use the minimum support threshold as an input parameter, so their ROUGE scores are constant in all the variations of the minimum support thresholds in the graphs. The second baseline (**FRQ-CL**) generates a frequent itemset model from a set of biomedical concepts and constructs the summary based on this model.

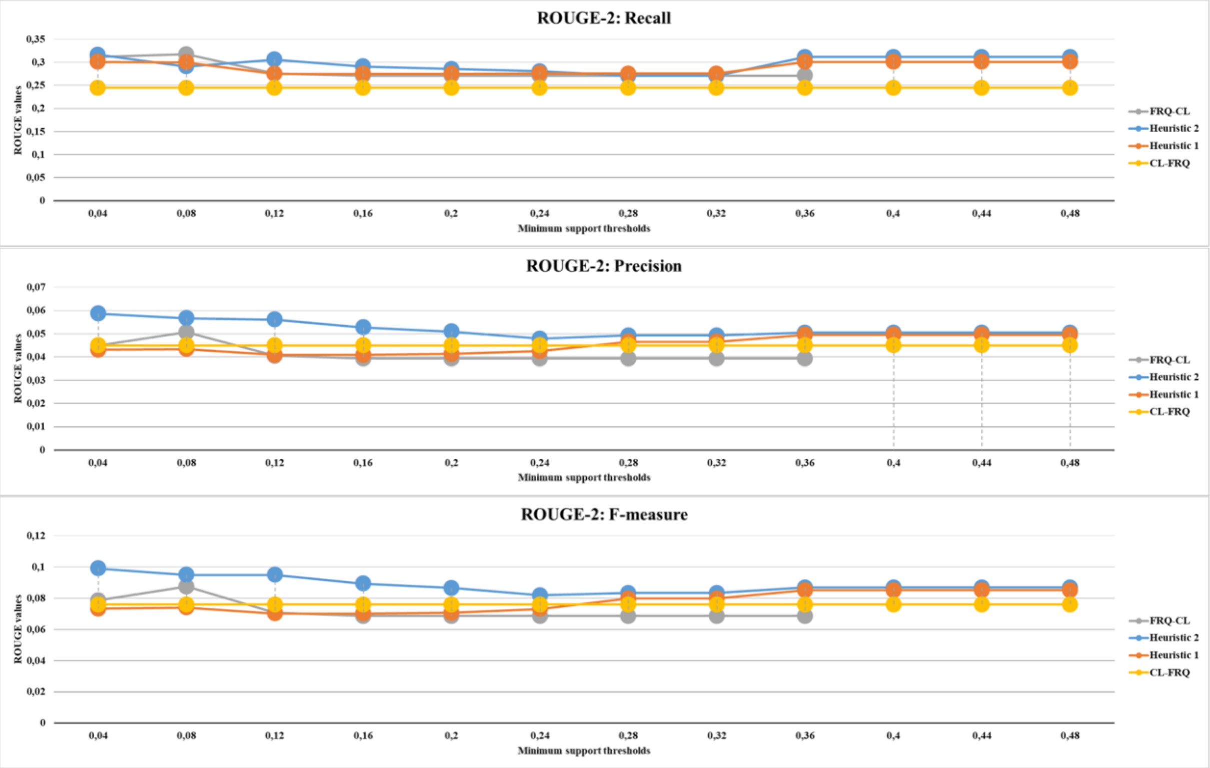

Figure 4: ROUGE-2 values in term of recall, precision, and F-measure

Firstly, we observe according to ROUGE scores, that the value of the minimum support threshold has a crucial impact on the quality of the generated summaries. When we enforce higher minimum support thresholds in each cluster, many itemsets are discarded. Thus, the itemset-based model of each cluster becomes too general to capture all the information in the cluster. Oppositely, when we enforce very low minimum support thresholds, data overfitting occurs, i.e. the generated models are too much specialized to effectively and concisely summarize the whole clusters because it contains too many frequent itemsets. We vary the minimum support threshold on each technique for comparison purpose, e.g., in the first baseline (FRQ-CL), when we set the minimum support threshold to 0.02, we have obtained a ROUGE scores **(R-1: r=0.74619, R-1: p=0.10621; R-2: r=0.30924, R-2: p=0.04486, R-SU4: r=0.37837, R-SU4: p=0.05218)**. However, when we set the minimum support threshold to 0.08, the ROUGE scores increase: **(R-1: r=0.76635, R-1: p=0.11909; R-2: r=0.31753, R-2: p=0.05066, R-SU4: r=0.38922, R-SU4: p=0.05996)**. Moreover, it decreases when we set the minimum support threshold> 0.08. In contrast, the performance of the baseline (FRQ-CL) is significantly better than that of the baseline (CL-FRQ) which got the worst ROUGE scores for **R-1 (r=0.72081, p=0.10058), R-2 (r=0.2449) and R-SU4 (r=0.31904)**, but it performs slightly better than the baseline (FRQ-CL): for **R-SU4 (p=0.05593) and R-2 (p=0.04496)** when the minimum support threshold≥ 0.1.

We can also observe that clustering has a crucial impact on the performances of our summarizer. In both the two heuristics, that are based on the combination of clustering and frequent itemset mining (both with and without using all clusters to generate summaries) have generally high ROUGE scores than the two baselines for all the ROUGE scores.

Regarding the comparison between our two proposed heuristics, the performance of the second heuristic is much better than that of the first heuristic for all ROUGE metrics. For instance, when we set the minimum support threshold to 0.12. The second heuristic reports higher ROUGE scores: **(R-1: r=0.83249, R-1: p= 0.15327; R-2:r=0.30612, R-2:p=0.05613; R-SU4: r=0.38765, R-SU4: p=0.07058)** compared to the first heuristic using the same minimum support threshold value **(R-1: r=0.75635, R-1: p=0.12692; R-2: r=0.25000, R-2: p=0.04092; R-SU4: r=0.37820, R-SU4: p=0.05777)**. However, for the values of minimum support threshold (i.e., 0.36 and above…) the two heuristics have a very close ROUGE scores: **(R-1: p= 0.12975 vs. p=0.12471; R-2: r=0.31122 vs. r=0.30102, R-2: p=0.05045 vs. p=0.0495; R-SU4: r=0.38250 vs. r=0.37479, R-SU4: p=0.06157 vs. p=0.06119)**. Nevertheless, the second heuristic still has the best ROUGE values comparing to that of the first heuristic and to all baselines. Finally, we can also observe that there is no optimal minimum support threshold with regards the obtained ROUGE scores, because we divide each document to a set of sentences, and we use the frequent itemset mining in each cluster so the optimal minsup threshold value is too dependent to each cluster.

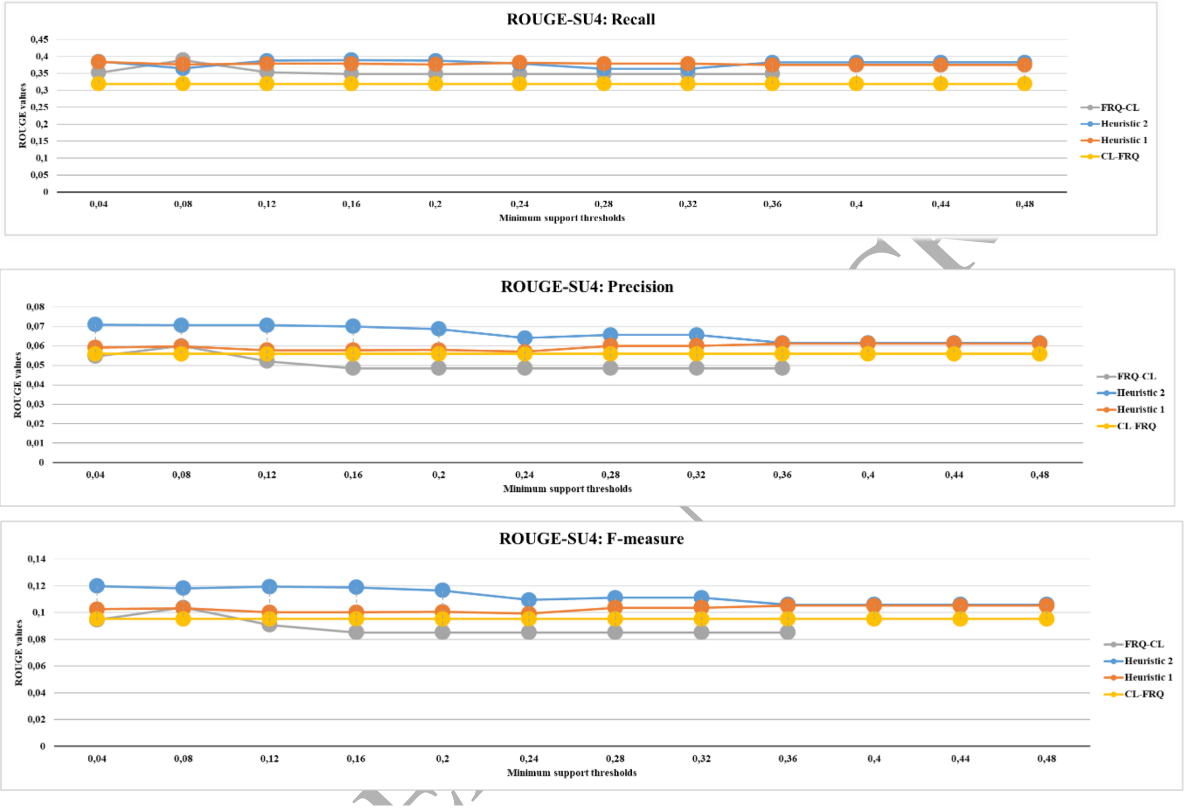

Figure 5: ROUGE-SU4 values in term of recall, precision, and F-measure

5.3. Comparisons with other summarizers

In this section, we compared the summaries generated by the system with those produced by other summarizers on the same corpus. We generated all summaries with a size of 25% of the total number of sentences in documents. Statistically, we used the average values obtained by ROUGE F-measure to simplify the interpretation of the results.

The first summarizer is the Microsoft AutoSummarize[4] (mic); it is a commercial application of the Microsoft Word software, which uses a term-frequency based approach. The second is an online summarizer: SweSum[5] (swe) that is a Swedish text summarizer for newspapers, and it is built on both statistical and linguistic methods, it selects the most important sentences based on a score that combines the position of sentences, sentences tags, sentences containing numerical values, either are contained keywords or not. The rest are three prototypes: (Itemset based summarizer, TextRank, and TextTeaser). Itemset based summarizer (Moradi & Ghadiri, 2017) is a biomedical text summarizer that uses both a frequent itemset model with a conceptual representation of biomedical texts. TextRank (Mihalcea & Tarau, 2004) is a text summarizer that represents text units as a graph, and then it uses the PageRank algorithm to determine the most important units in this graph. TextTeaser[6] (tea) is a machine learning solution that uses some features to score sentences like their relevance to the title, their relevance to keywords in the article, their positions, and their lengths etc.

Table 4 reports the comparison, in terms of ROUGE-1, ROUGE-2 and ROUGE-SU4 F-measures between our system and other summarizers. The summarizers are ranked in decreasing order of ROUGE-2.

Table 4: The different ROUGE scores obtained by the execution of the proposed system including its two sentences selection strategies and other summarizers.

	ROUGE-1	ROUGE-2	ROUGE-SU4
Heuristic 2	**0,23840**	**0,08715**	**0,11456**
Heuristic 1	0,23310	0,08463	0,11108
Itemset based summarizer	0,23243	0,08216	0,11230
TextTeaser	0,23100	0,07936	0,10902
SweSum	0,23253	0,07898	0,11086
TextRank	0,20938	0,07877	0,09898
Microsoft AutoSummarize	0,22152	0,07513	0,10168

[4] Microsoft Word 2007, ed: Microsoft Coporation, 2007

[5] Automatic Text Summarizer < http://swesum.nada.kth.se >

[6] A python text summarizer < https://github.com/IndigoResearch/textteaser >

It may be observed from table 4 that our system with the two proposed heuristics (1 and 2) report higher ROUGE scores than the other tested summarizers. The best results are obtained when using the second heuristic. Therefore, it ranks first in all ROUGE scores. The first heuristic ranks the second in ROUGE-1 and ROUGE-2 but ranks the third in term of ROUGE-SU4 after Itemset based summarizer. However, the worst summarizers are TextRank that ranks the seventh in ROUGE-1 and ROUGE-SU4 and Microsoft AutoSummarize in ROUGE-2.

5.4. Discussions

The key idea of this paper is to show that the combination of clustering and frequent itemsets mining with a conceptual representation of biomedical texts can enhance the quality of the generated summaries compared to those generated by only frequent itemsets mining or clustering, and this is what we found in the previous sections. Our system, with its two proposed sentence selection heuristics produces good ROUGE values compared to the implemented baselines. These improved results are due to the division of the text into similar clusters using the cosine similarity measure and the selection of the most informative sentences from each cluster using the frequent itemsets mining. This idea is absent when we applied it only to determine the most important sentences.

As we have seen in the experimentation process, biomedical documents often have a similar structure, and they are broken down into four clusters with one largest cluster and three other clusters of different sizes. The largest cluster contains the closest sentences to the global subtopic of the document, while the others contain information related to this subtopic, but it also contains other secondary information. When we cluster similar sentences, we try to find sentences that are similar to each other’s in the same cluster and dissimilar to other sentences in other clusters. Which is not the case in the frequent itemsets mining technique; in this, we find the correlations between the concepts appearing in the sentences, and the sentences that cover these frequent itemsets are the most informative sentences in the text. However, is that the generated frequent itemsets-based model really captures all subtopics of the text? Knowing that in the first step the Apriori algorithm takes all concepts at the same level (unweighted concepts). Moreover, sometimes there exist subtopics that are not expressed by many concepts but they have secondary information that should be considered important to appear in the final summary. Therefore, our system ensures the selection of the most important sentences using all the clusters to cover all the subtopics of the document.

In the proposed system, looking at table 4, the number and the sizes of the discovered frequent itemsets in our system are higher than those in the baseline FRQ- CL, in which we do not use the clustering. The reason is that we generate frequent itemsets model in a subset of more similar sentences that share similar information. Therefore, the obtained frequent itemsets are more significant, and our system does not despise any frequent itemsets compared to the baseline FRQ-CL. Thus, the generated models are more precise and informative, and the quality of the generated summaries is increased significantly.

Concerning the comparison between our two sentences selection heuristics, table 4 shows that the second heuristic achieves best ROUGE values, because it selects the maximum number of sentences from the largest cluster while this latter contains global information related to the main topic of the text, but it also includes some sentences from other clusters. Thus, in addition to information about the global subtopic, this heuristic also includes additional information that may be of interest to reader. On the other hand, the first heuristic does not present this information. The reason is that scientific writers ensure consistency in theirs biomedical documents by some redundancy of information by adding other optional information to pass from a topic to another, the majority of the important information is usually founded in the section describing the method while other sections contain secondary information related to this section.

6. Conclusions and future work

In this paper, we have presented a biomedical text summarization system that combines clustering and frequent itemset mining, with a conceptual representation of biomedical texts. We showed that this combination could enhance the quality of the generated summaries. We have proposed different heuristics on how to select the most informative sentences from the clusters, we have evaluated the performances of our system in term of ROUGE scores against two baselines, and five other summarizers, and it was shown that our system outperforms the baselines and other summarizers and the results are promising.

We are considering in future work to make an extension of the proposed system to address a more semantic analysis of biomedical texts by integrating the concept of word embedding in the text representation. Besides, we will try to incorporate a new anti-redundancy technique to reduce the duplicate information to improve the quality of the summaries.

Conflicts of interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper

References

Afantenos, S., Karkaletsis, V., & Stamatopoulos, P. (2005). Summarization from medical documents: A survey. *Artificial Intelligence in Medicine*, *33*, 157–177.

Agrawal, R., & Srikant, R. (1994). Fast Algorithms for Mining Association Rules in Large Databases. *Journal of Computer Science and Technology*, *15*, 487–499.

Anton, H. (1994). *Elementary linear algebra*. John Wiley.

Baralis, E., Cagliero, L., Fiori, A., & Garza, P. (2015). MWI-Sum: A Multilingual Summarizer Based on Frequent Weighted Itemsets. *ACM Transactions on Information Systems*, *34*, 1–35.

Baralis, E., Cagliero, L., Jabeen, S., & Fiori, a. (2012). Multi-document summarization exploiting frequent itemsets. *Proceedings of the 27th Annual …*, (pp. 782–786).

Baralis, E., Cagliero, L., Mahoto, N., & Fiori, A. (2013). GraphSum: Discovering correlations among multiple terms for graph-based summarization. *Information Sciences*, *249*, 96–109.

Ben Aouicha, M., & Hadj Taieb, M. A. (2016). Computing semantic similarity between biomedical concepts using new information content approach. *Journal of Biomedical Informatics*, *59*, 258–275.

Dzuganova, B. (2013). English medical terminology – different ways of forming medical terms. *JAHR - European Journal of Bioethics*, *4*, 55–69.

Edmundson, H. P., & P., H. (1969). New Methods in Automatic Extracting. *Journal of the ACM*, *16*, 264–285.

Fleuren, W. W., & Alkema, W. (2015). Application of text mining in the biomedical domain. *Methods*, *74*, 97–106.

Friedman, C., & Elhadad, N. (2014). Natural Language Processing in Health Care and Biomedicine. In *Biomedical Informatics* (pp. 255–284). London: Springer London.

Gay, C. W., Kayaalp, M., & Aronson, A. R. (2005). Semi-automatic indexing of full text biomedical articles. *AMIA ... Annual Symposium proceedings / AMIA Symposium. AMIA Symposium*, (pp. 271–275).

Hovy, E. (2005). Automated text summarization. In *The Oxford Handbook of Computational Linguistics, Oxford University Press* (pp. 583–598).

Jaccard, P. (1901). Etude de la distribution florale dans une portion des Alpes et du Jura. *Bulletin de la Societe Vaudoise des Sciences Naturelles*, *37* , 547–579.

Lin, C. Y. (2004). Rouge: A package for automatic evaluation of summaries. *Proceedings of the workshop on text summarization branches out (WAS 2004)*, (pp. 25–26).

Macqueen, J. (1967). Some methods for classification and analysis of multivariate observations. *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, *1*, 281–297.

Menéndez, H. D., Plaza, L., & Camacho, D. (2013). A genetic graph-based clustering approach to biomedical summarization. *Proceedings of the 3rd International Conference on Web Intelligence, Mining and Semantics - WIMS '13*, (p. 1).

Mihalcea, R., & Tarau, P. (2004). TextRank: Bringing order into texts. *Proceedings of EMNLP*, *85*, 404–411.

Mishra, R., Bian, J., Fiszman, M., Weir, C. R., Jonnalagadda, S., Mostafa, J., & Del Fiol, G. (2014). Text summarization in the biomedical domain: A systematic review of recent research. *Journal of Biomedical Informatics*, *52*, 457–467.

Moradi, M., & Ghadiri, N. (2017). Quantifying the informativeness for biomedical literature summarization: An itemset mining method. *Computer Methods and Programs in Biomedicine*, *146*, 77–89.

National, B., & Us, M. (2009). UMLS ® Reference Manual. *Health (San Francisco)*, .

Page, L., Brin, S., Motwani, R., & Winograd, T. (1998). The PageRank Citation Ranking: Bringing Order to the Web, .

Plaza, L., Díaz, A., & Gervás, P. (2011). A semantic graph-based approach to biomedical summarisation. *Artificial Intelligence in Medicine*, *53*, 1–14.

Plaza, L., Stevenson, M., & D'iaz, A. (2012). Resolving ambiguity in biomedical text to improve summarization. *Information Processing & Management*, *48*, 755–766.

Plaza, M., Laura and, & D'iaz, A. (2010). Improving Summarization of Biomedical Documents using Word Sense Disambiguation. *Proceedings of the Workshop on Biomedical Natural Language Processing*, (pp. 55–63).

Pubmed, H. (2005). PubMed Help. *October*, (pp. 1–67).

Radev, D. R. (2004). LexRank : Graph-based Lexical Centrality as Salience in Text Summarization, . *22*, 457–479.

Radev, D. R., Jing, H., & Budzikowska, M. (2000). Centroid-based summarization of multiple documents: sentence extraction, utility-based evaluation, and user studies. *Information Processing & Management 40.6 (2004): 919-938.*, *40*, 10.

Reeve, L., Han, H., & Brooks, A. D. (2006a). BioChain. In *Proceedings of the 2006 ACM symposium on Applied computing - SAC '06* (p. 180). New York, New York, USA: ACM Press.

Reeve, L. H., Han, H., Nagori, S. V., Yang, J. C., Schwimmer, T. A., & Brooks, A. D. (2006b). Concept frequency distribution in biomedical text summarization. *Proceedings of the 15th ACM international* *conference on Information and knowledge management - CIKM 06*, (p. 604).

Reeve, L. H., Han, H., Nagori, S. V., Yang, J. C., Schwimmer, T. A., & Brooks, A. D. (2006c). Concept frequency distribution in biomedical text summarization. In *Proceedings of the 15th ACM international conference on Information and knowledge management - CIKM '06* (p. 604). New York, New York, USA: ACM Press.

Salton, G., Wong, A., & Yang, C. S. (1975). A vector space model for automatic indexing. *Communications of the ACM*, *18*, 613–620.

Shams S (). UMLS Knowledge Sources | Machine Learning Medium.

Shang, Y., Li, Y., Lin, H., & Yang, Z. (2011). Enhancing biomedical text summarization using semantic relation extraction. *PLoS ONE*, *6*.

Shortliffe, E. H., & Cimino, J. J. (2014). *Biomedical informatics : computer applications in health care and biomedicine*. (4th ed.). Springer London.

Singhal, A. (2001). Modern Information Retrieval: A Brief Overview. *IEEE Computer Society Technical Committee on Data Engineering*, *24*, 35–42.

Sollaci, L. B., & Pereira, M. G. (2004). The introduction, methods, results, and discussion (IMRAD) structure: a fifty-year survey. *Journal of the Medical Library Association*, *92*, 364–371.

Sparck Jones, K. (1972). A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, *28*, 11–21.

Tan, P.-N., Steinbach, M., & Vipin Kumar (2006). *Introduction to data mining*.

Yoo, I., Hu, X., & Song, I. Y. (2007). A coherent graph-based semantic clustering and summarization approach for biomedical literature and a new summarization evaluation method. *BMC Bioinformatics*, *8*, 1–15.

CRediT author statement

Oussama Rouane: Conceptualization, Methodology, Software, Writing- Original draft preparation.
Hacene Belhadef: Supervision, Validation, Writing- Reviewing and Editing.
Mustapha Bouakkaz: Supervision, Writing- Reviewing.