Artificial Intelligence in Medicine 53 (2011) 1–14

Contents lists available at ScienceDirect

Artificial Intelligence in Medicine

journal homepage: www.elsevier.com/locate/aiim

A semantic graph-based approach to biomedical summarisation

Laura Plaza*, Alberto Díaz, Pablo Gervás

Departamento de Ingeniería del Software e Inteligencia Artificial, Universidad Complutense de Madrid, C/Profesor José García Santesmases, s/n, 28040 Madrid, Spain

ARTICLE INFO

Article history:
Received 12 November 2009
Received in revised form 10 May 2011
Accepted 18 June 2011

Keywords:
Semantic graphs
Biomedical concept annotation
Concept clustering
Unified Medical Language System
Biomedical text summarisation

ABSTRACT

Objective: Access to the vast body of research literature that is available in biomedicine and related fields may be improved by automatic summarisation. This paper presents a method for summarising biomedical scientific literature that takes into consideration the characteristics of the domain and the type of documents.

Methods: To address the problem of identifying salient sentences in biomedical texts, concepts and relations derived from the Unified Medical Language System (UMLS) are arranged to construct a semantic graph that represents the document. A degree-based clustering algorithm is then used to identify different themes or topics within the text. Different heuristics for sentence selection, intended to generate different types of summaries, are tested. A real document case is drawn up to illustrate how the method works.

Results: A large-scale evaluation is performed using the recall-oriented understudy for gisting-evaluation (ROUGE) metrics. The results are compared with those achieved by three well-known summarisers (two research prototypes and a commercial application) and two baselines. Our method significantly outperforms all summarisers and baselines. The best of our heuristics achieves an improvement in performance of almost 7.7 percentage units in the ROUGE-1 score over the LexRank summariser (0.7862 *versus* 0.7302). A qualitative analysis of the summaries also shows that our method succeeds in identifying sentences that cover the main topic of the document and also considers other secondary or "satellite" information that might be relevant to the user.

Conclusion: The method proposed is proved to be an efficient approach to biomedical literature summarisation, which confirms that the use of concepts rather than terms can be very useful in automatic summarisation, especially when dealing with highly specialised domains.

1. Introduction

It is undeniable that information technologies have represented a major milestone in health care practice and in biomedical research. New technologies, such as high-speed networks and massive storage, along with the progressive adoption of the electronic health record (EHR) and the increasing publication of research results in digital journals, are supposed to improve work efficiency by assuring data persistence and the availability of information everywhere and at any time.

Access to biomedical literature has been shown to be beneficial to both health professionals and consumers [1,2]. However, the enormous volume of literature available threatens to undermine the convenience of the information in the absence of easy and effective access technologies [1,3]. Text summarisation may help to manage this information overload [3–5]. Researchers can use summaries to quickly determine whether an article is of interest without having to read the entire document [4]. Physicians can use summaries to identify treatment options, reducing diagnosis time [6]. Moreover, automatic summaries have been shown to improve indexing and categorisation of biomedical literature when used as substitutes for the articles' abstracts [7,8]. Even though the problem of information overload and the benefits of summarisation are common to most scientific disciplines, they are particularly critical in the biomedical domain because physicians and researchers require quick access to up-to-date information relevant to their needs [1].

The majority of summarisation systems are designed to be general-purpose, and for this reason they do not take into account the particular properties of each domain and document type. They usually work with a representation of the document consisting of information that can be directly extracted from the document itself, such as terms, phrases or sentences [9,10]. However, recent studies have demonstrated the benefits of summarisation based on richer representations that make use of domain-specific knowledge sources [5]. These approaches represent the documents using concepts instead of words, and they may also be enriched by using semantic associations among concepts (e.g., synonymy,

* Corresponding author. Tel.: +34 91 394 7576; fax: +34 91 394 7547.
E-mail addresses: lplazam@fdi.ucm.es, lplazamorales@gmail.com (L. Plaza).

0933-3657/$ – see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.artmed.2011.06.005

hypernymy, homonymy or co-occurrence) to improve the quality of the summaries. In particular, the Unified Medical Language System (UMLS) [11] has proved to be a useful knowledge source for summarisation in the biomedical domain [4,5]. Moreover, the need to consider the particular characteristics of the domain and the type of documents is becoming apparent. First, documents in biomedicine are very different from documents in other fields and include very different document types (e.g., patient records, web documents, scientific papers and even e-mailed reports). Second, medical language, despite being highly specialised, is also highly interpretive, and it is constantly expanding [12]. It seems reasonable that these peculiarities should be exploited by the summarisation system.

The main contribution of this work is to show how the use of domain-specific concepts from controlled terminologies and the consideration of the structural properties of the documents provide additional knowledge that may benefit the summarisation process and the quality of the summaries. A graph-based summariser is presented that uses the UMLS to identify concepts and the semantic relations between them to construct a rich semantic representation of the document to be summarised. Three strategies for sentence selection are proposed, each of them aiming to construct a different type of summary according to the type of information in the source that is likely to be included in the summary. Moreover, the summariser deals with several problems derived from the peculiarities of biomedical terminology, such as lexical ambiguity and the use of acronyms and abbreviations.

The paper is organised as follows. Section 2 describes the background and related work on text summarisation and UMLS concept annotation. Section 3 presents the method of summarisation and the evaluation methodology. Section 4 shows the evaluation results and compares the system to other popular summarisers. Section 5 discusses these results. The final section provides concluding remarks and describes future lines of work.

2. Background

2.1. Previous work in summarisation

Text summarisation is the process of automatically creating a compacted version of a given text. Content reduction can be addressed by selection and/or by generalisation of what is important in the source [13]. This definition suggests that two generic groups of summarisation methods exist: those that generate *extracts* and those that generate *abstracts*. Extractive summarisation produces summaries by selecting salient sentences from the original document, and therefore the summaries are essentially composed of material that is explicit in the source. In contrast, abstractive summarisation constructs summaries in which the information from the source has been paraphrased. Although human summaries are typically abstracts, most existing systems produce extracts largely because extractive summarisation has been demonstrated to report better results than abstractive summarisation [14]. This superiority is due to the difficulties entailed by the abstraction process, which usually involves identifying the most prevalent concepts in the source, the appropriate semantic representation of them and the rewriting of the summary through natural language generation techniques.

Extractive methods typically construct summaries based on a superficial analysis of the source. Early summarisation systems were based on what Mani called the *Edmundsonian paradigm* [15]. In this paradigm, sentences are ranked using simple heuristic features, such as the position of the sentences in the document [16], the frequency of their terms [17,18], the presence of certain cue words and indicative phrases [18] or the word overlap between the sentences and the document title and headings [18]. These features are usually combined using a linear weighting function that assigns a single score to each sentence in the document, and the highest scoring sentences are extracted for the summary. More recent approaches also employ machine learning techniques to determine the best subset of features for extraction [19].

Most advanced techniques incorporate graph-based methods. This paper mainly investigates previous work in graph-based summarisation (see [15] for a more thorough study of domain-independent summarisation techniques and [1] for biomedical-focused approaches). Graph-based methods usually represent the documents as graphs where the nodes correspond to text units such as words, phrases, sentences or even paragraphs, and the edges represent cohesion or similarity relations between these units. Once the document graph has been created, salient nodes within it are identified and used to extract the corresponding units for the summary.

LexRank [9] is the best-known example of a graph-based method for multi-document summarisation. It assumes a fully connected and undirected graph for the set of documents to be summarised, in which each node corresponds to a sentence represented by its TF-IDF vector, and the edges are labelled with the cosine similarity between the sentences. Only the edges that connect sentences with a similarity above a predefined threshold are drawn in the graph. The sentences represented by the most highly connected nodes are selected for the summary. A very similar method, TextRank, is proposed by Mihalcea and Tarau [10]. TextRank differs from LexRank in three main aspects: first, it is intended for single-document summarisation; second, the similarity between sentences (i.e., the weight of the edges in the document graph) is measured as a function of their content overlap; and third, the PageRank algorithm [20] is used to rank the nodes in the document graph. Most recently, Litvak and Last [21] proposed a novel approach that uses a graph-based syntactic representation of textual documents for keyword extraction, which can be used as a first step in single-document summarisation. They represent the document as a directed graph, where the nodes represent single words found in the text, and the edges (not labelled) represent precedence relations between words. A hyperlink-induced topic search algorithm [22] is then run on the document graph under the assumption that the top-ranked nodes should represent the document keywords.

Although these approaches are promising, they exhibit important deficiencies that are consequences of not capturing the semantic relationships between terms (synonymy, hypernymy, homonymy and co-occurrence relations). The following sentences illustrate such problems:

1. *Cerebrovascular diseases* during *pregnancy* result from any of three major mechanisms: arterial infarction, haemorrhage or venous thrombosis.
2. *Brain vascular disorders* during *gestation* result from any of three major mechanisms: arterial infarction, haemorrhage or venous thrombosis.

Because the two sentences present different terms, the approaches above are unable to make use of the fact that they have exactly the same meaning. This problem may be solved by dealing with concepts instead of terms and with semantic relations instead of lexical or syntactical ones. Consequently, some recent approaches have adapted existing methods to represent the document at a conceptual level.

For example, in the biomedical domain, Reeve et al. [4] adapt the lexical chaining approach [23] to use UMLS concepts rather than terms and apply it to single-document summarisation. They automatically identify UMLS concepts in the source and chain them so that each chain contains a list of concepts belonging

to the same UMLS semantic type. The concept chains are then scored by multiplying the frequency of the most frequent concept in the chain by the number of distinct concepts in it, and these scores are used to identify the strongest concept chains. Finally, the sentences are scored based on the number of concepts that they contain from strong chains. Yoo and colleagues [24] use the Medical Subject Headings (MeSH) [25] to represent a corpus of documents as a graph, where the nodes are the MeSH descriptors found in the corpus, and the edges represent hypernymy and co-occurrence relations between them. The concepts are clustered to identify groups of documents dealing with the same topic using a degree-ranking method. Each document cluster is then used to produce a single summary. For this purpose, they construct a text semantic interaction network that represents the set of documents to be summarised, using only the semantic relations found in the document cluster. BioSquash [26] is a question-oriented extractive system for biomedical multi-document summarisation. It constructs a graph that contains concepts of three types: ontological concepts (general ones from WordNet [27] and specific ones from the UMLS), named entities and noun phrases. The edges of this graph represent semantic relationships between concepts, but nothing is said about the specific relationships used. A more complex work is presented in Fiszman et al. [5]. They propose an abstractive approach that relies on the semantic predications provided by SemRep [28] to interpret biomedical text and on a transformation step using lexical and semantic information from the UMLS to produce abstracts from biomedical scientific articles. However, these abstracts are presented in a graphical format, and the production of textual summaries using language generation techniques has been relegated to future work.

A recent technique that has proved to be useful for summarisation is *sentence simplification*. Although it is beyond the scope of this work, it is expected to provide future improvement of the methods. Sentence simplification or compression can be considered as a means of creating more space within which to capture important content by producing a simpler and shorter version of a sentence while retaining the relevant information [29]. Sentence simplification approaches have been little explored in the biomedical domain, mainly due to the complexity of the sentences. Recently, the BioSimplify system [30] proved the utility of sentence simplification to improve the output of parsers in biomedical literature, while Jonnalagadda and Gonzalez [31] studied the impact of sentence simplification on the extraction of protein–protein interactions from biomedical articles. Lin and Wilbur [32] showed that sentence compression of biomedical article titles facilitates user decisions regarding whether an article is worth examining in response to an information need. All of these approaches use a series of linguistically motivated trimming rules to remove inessential fragments from the parse tree of a sentence.

2.2. Biomedical domain singularities

Biomedical texts exhibit certain unique attributes that must be taken into account in the development of a summarisation system. First, medical information arises in a wide range of document types [1]: EHR, scientific articles, semi-structured databases, X-ray images and even videos. Each document type presents very distinct characteristics that should be considered in the summarisation process. We focus on scientific articles, which are mainly composed of text but usually contain tables and images that may contain important information that should appear in the summary. Biomedical papers often present the IMRaD structure (*Introduction*, *Method*, *Results and Discussion*), but sometimes also present additional sections such as *abbreviations, limitations of the study* and *competing interests.* In most cases, depending on the summarisation task, this knowledge about the article layout can be exploited to improve the summaries that are generated automatically.

Second, the peculiarities of the terminology make it difficult to automatically process biomedical information [33]. The first challenge is the problem of *synonyms* (the use of different terms to designate the same concept) and *homonyms* (the use of words/phrases with multiple meanings). For instance, the syntagms *coronary failure* and *heart attack* stand for the same concept, while the term *anaesthesia* may refer to either the loss of sensation or the procedure for pain relief. Another handicap to automatic concept recognition is the presence of *neologisms*, which are newly coined words that are not likely to be found in a dictionary (e.g., the term *coumadinise* for the administration of coumadin). Finally, *elisions* and *abbreviations* complicate the automatic processing of medical texts. Elision is the omission of words or sounds in a word or phrase. An example of elision is *white count,* understood by physicians as the *count of white blood cells.* An abbreviation is a shortened form of a word or phrase, for example, the use of *OCP* to refer to *oral contraceptive pills.*

2.3. The use of the UMLS for automatic concept annotation

The UMLS [11,34] is a collection of controlled vocabularies related to biomedicine that contains a wide range of information that can be used for natural language processing (NLP). It consists of three main components: the Specialist Lexicon, the Metathesaurus and the Semantic Network.

The *UMLS Specialist Lexicon* [35] is a database of lexicographic information conceived especially for NLP systems to address the high degree of variability in natural language words. It is intended to be a general English lexicon but also includes many biomedical terms. The lexicon entry for each word records syntactic, morphological and orthographic information. The *UMLS Metathesaurus* [36] comprises a collection of biomedical and health related concepts derived from more than 100 different vocabulary sources, their various names and the relationships among them. The *UMLS Semantic Network* [37] consists of a set of categories (or semantic types) that provides a consistent categorisation of the concepts in the Metathesaurus, along with a set of relationships (or semantic relations) that exist between the semantic types.

Using the UMLS for NLP tasks instead of another biomedical knowledge source (e.g., the Systematized Nomenclature of Medicine-Clinical Terms (SNOMED-CT) [38] or MeSH [25]) offers two main advantages: (1) a broader coverage, as it is a compendium of vocabularies including SNOMED-CT and MeSH and (2) support by a number of resources that aid developers of NLP applications, such as lexical tools, concept annotators and word sense disambiguation algorithms. Moreover, using the UMLS for concept annotation offers two further advantages: (1) it lists more than 15000 entries of ambiguous terms (which attenuate the problems of synonymy and homonymy), and (2) it contains numerous entries for elisions and abbreviations.

To map biomedical text to concepts in the UMLS Metathesaurus, the National Library of Medicine has developed the MetaMap program [39,40]. MetaMap employs a knowledge-intensive approach that uses the Specialist Lexicon in combination with lexical and syntactic analysis to identify noun phrases in text.

Matches between noun phrases and Metathesaurus concepts are computed by generating lexical variations and allowing partial matches between the phrase and the concept. The possible UMLS concepts are assigned scores based on the closeness of the match between the input noun phrase and the target concept. The highest scoring concepts and their semantic types are returned. Fig. 1 shows this mapping for the syntagm *heart attack trial.* The first section in the MetaMap response (*meta candidates)* shows the candidate concepts, whereas the second section (*meta mapping*) shows

Phrase: "heart attack trial"
Meta candidates (8)
827 Trial (clinical trials) [Research activity]
734 Heart attack (myocardial infarction) [Disease or syndrome]
660 Heart [Body part, organ, or organ component]
660 Attack, NOS (onset of illness) [Finding]
660 Attack (attack device) [Medical device]
660 Attack (attack behavior) [Social behavior]
660 Heart (entire heart) [Body part, organ, or organ component]
660 Attack (observation of attack) [Finding]
Meta mapping (901)
734 Heart attack (myocardial infarction) [Disease or syndrome]
827 Trial (clinical trials) [Research activity]

Fig. 1. An example of MetaMap mapping for the syntagm *heart attack trial*. Each candidate mapping is given a score and is represented by its name in the Metathesaurus (in parentheses) and its semantic type in the Semantic Network (in brackets).

the highest scoring candidates. Each candidate is represented by its MetaMap score, its concept name in the Metathesaurus and its semantic type in the Semantic Network.

The UMLS and MetaMap have been used in a number of biomedical NLP applications, including machine translation [41], question answering [42] and information retrieval [43,44]. Erk et al. [41], for instance, modify a simple statistical machine translation system to use information from UMLS concepts and semantic types, thus achieving a significant improvement in translation performance. Overby et al. [42] show that both the UMLS Metathesaurus and the MetaMap program are useful for extracting answers to translational research questions from biomedical text in the field of genomic medicine. Aronson and Rindflesch [43] use MetaMap to expand queries with UMLS Metathesaurus concepts. The authors conclude that query expansion based on the UMLS improves retrieval performance and compares favourably with retrieval feedback. Plaza and Díaz [44] propose a method for the retrieval of similar clinical cases based on mapping the text in EHR onto UMLS concepts and representing the patient records as a set of semantic graphs. Each of these graphs corresponds to a different category of information (e.g., diseases, symptoms and signs or medicaments). These categories are automatically derived from the UMLS semantic types to which the concepts in the records belong.

3. Methods

3.1. Summarisation method

In this section, the concept graph-based summariser is presented. The method accomplishes the task of identifying the N most relevant sentences in a document through seven steps: (1) document preprocessing, (2) concept recognition, (3) sentence representation, (4) document representation, (5) concept clustering, (6) sentence-to-cluster assignment and (7) sentence selection.

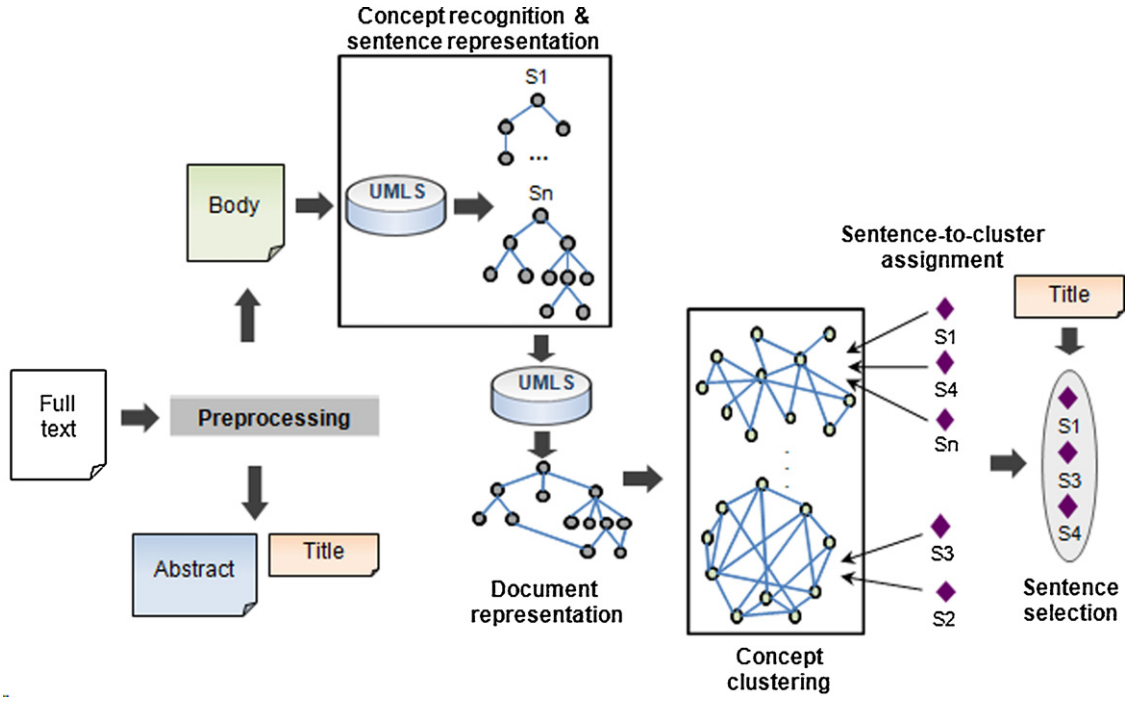

Fig. 2. Summariser architecture. The figure shows the seven steps involved in the algorithm: (1) preprocessing, (2) concept recognition, (3) sentence representation, (4) document representation, (5) concept clustering, (6) sentence-to-cluster assignment and (7) sentence selection.

Each step is discussed in detail in the following subsections. Fig. 2 illustrates the architecture of the summarisation method. Moreover, to clarify how the algorithm works, a complete document example from the BioMed Central corpus [47] (document *cvm-2-6-254.xml*) is elaborated throughout the summarisation process. It is worth mentioning that, although our interest here is to summarise biomedical literature, the summarisation method is generic and may be adapted to work with different types of documents (see [45] and [46] for examples of preliminary applications to summarising news items and tourism-related web sites, respectively).

3.1.1. Document preprocessing

A preliminary step is undertaken to prepare the document for the subsequent steps. This preprocessing involves the following actions:

- First, sections of the document that are considered irrelevant for inclusion in the summary are removed: *competing interests*, *acknowledgments*, *references* and section headings.
- Second, if the document includes an *abbreviations* section, the abbreviations and their expansions are extracted from it. This information is then used to replace these shortened forms in the document body. For example, if the abbreviations section defines *embryonic submandibular* as the expansion of *SMG* for a particular document, and if that document contains the phrase *Survivin may be a key mediator of SMG epithelial cell survival,* then that phrase would become *Survivin may be a key mediator of embryonic submandibular epithelial cell survival*.
- Third, to expand the acronyms and abbreviations not defined in the abbreviations section, the software for abbreviation definition recognition presented in [48] is used. This software is publicly available [49] and allows for the identification of abbreviations and their expansions in biomedical texts with an average precision of 95%. Abbreviations are then replaced by their expansions in the document body.
- Fourth, the title, abstract and body sections are extracted.
- Fifth, using a stop list from Medline [50], generic terms (e.g., prepositions and pronouns) in the body and title sections are removed because they are not useful in discriminating between relevant and irrelevant sentences.
- Finally, the text in the body section is split into sentences using the *tokenizer*, *part of speech tagger* and *sentence splitter* modules of the GATE architecture for text engineering [51].

The preprocessing step can easily be configured to deal with documents of different structures and with unstructured documents. A *config.xml* file allows users to specify, for instance, if the document is not structured and thus the entire text should be considered for the purpose of summarisation; the document sections that have to be ignored; the XML tags (if any) that enclose the title, abstract, body and abbreviations sections; the format used to specify the abbreviations and their expansions; or the stop list to be used.

3.1.2. Concept recognition

The next stage is to map the text in the document to concepts from the UMLS Metathesaurus and semantic types from the UMLS Semantic Network.

The MetaMap program is run over the text in the body section of the document. In particular, the 2009 version of MetaMap is employed, and the 2009AA UMLS release is used as the knowledge base. It is important to note that, in the presence of lexical ambiguity, MetaMap frequently fails to identify a unique mapping for a given phrase [52]. This failure occurs, for instance, for the phrase *Tissues are often cold,* where MetaMap returns three candidate concepts with equal scores for *cold* (*cold sensation*, *common cold* and *cold temperature*). To select the correct mapping for the context in which the phrase appears, MetaMap is invoked using the word sense disambiguation option (*-y* flag). This flag implements the Journal Descriptor Indexing (JDI) methodology described in [53]. This algorithm is based on semantic type indexing, which resolves Metathesaurus ambiguity by choosing a concept with the most likely semantic type for a given context. Using the–*y* flag forces MetaMap to choose a single mapping if there is more than one candidate concept for a given phrase. However, when the candidate concepts share the same semantic type, the JDI algorithm may fail to return a single mapping. In this case, the first mapping returned by MetaMap is selected.

Concepts from very generic UMLS semantic types are discarded because they have been found to be excessively broad. These semantic types are *quantitative concept*, *qualitative concept*, *temporal concept*, *functional concept*, *idea or concept, intellectual product, mental process, spatial concept* and *language*. These types were empirically determined by evaluating the summaries generated using UMLS concepts from different combinations of semantic types.

Table 1 shows the UMLS concepts identified for the sentence S1: The goal of the trial was to assess cardiovascular mortality and morbidity for stroke, coronary heart disease and congestive heart failure, as an evidence-based guide for clinicians who treat hypertension.

3.1.3. Sentence representation

For each sentence in the document, the UMLS concepts returned by MetaMap are retrieved from the UMLS Metathesaurus along with their complete hierarchy of hypernyms (*is_a* relations). All the hierarchies for each sentence are merged, creating a *sentence graph*

Table 1
MetaMap mapping for the sentence The goal of the trial was to assess cardiovascular mortality and morbidity for stroke, coronary heart disease and congestive heart failure, as an evidence-based guide for clinicians who treat hypertension. Ignored concepts of generic semantic types appear crossed out.

Concept	MetaMap score	Semantic type
~~Goals~~	~~1000~~	~~Intellectual product~~
Clinical trials	1000	Research activity
Cardiovascular system	694	Body system
~~Mortality vital statistics~~	~~861~~	~~Quantitative concept~~
~~Morbidity - disease rate~~	~~1000~~	~~Quantitative concept~~
Cerebrovascular accident	1000	Disease or syndrome
Coronary heart disease	1000	Disease or syndrome
Congestive heart failure	1000	Disease or syndrome
~~Evidence of~~	~~660~~	~~Functional concept~~
~~Basis~~	~~660~~	~~Functional concept~~
Clinicians	1000	Prof. or occup. group
~~Treatment intent~~	~~1000~~	~~Functional concept~~
Hypertensive disease	1000	Disease or syndrome

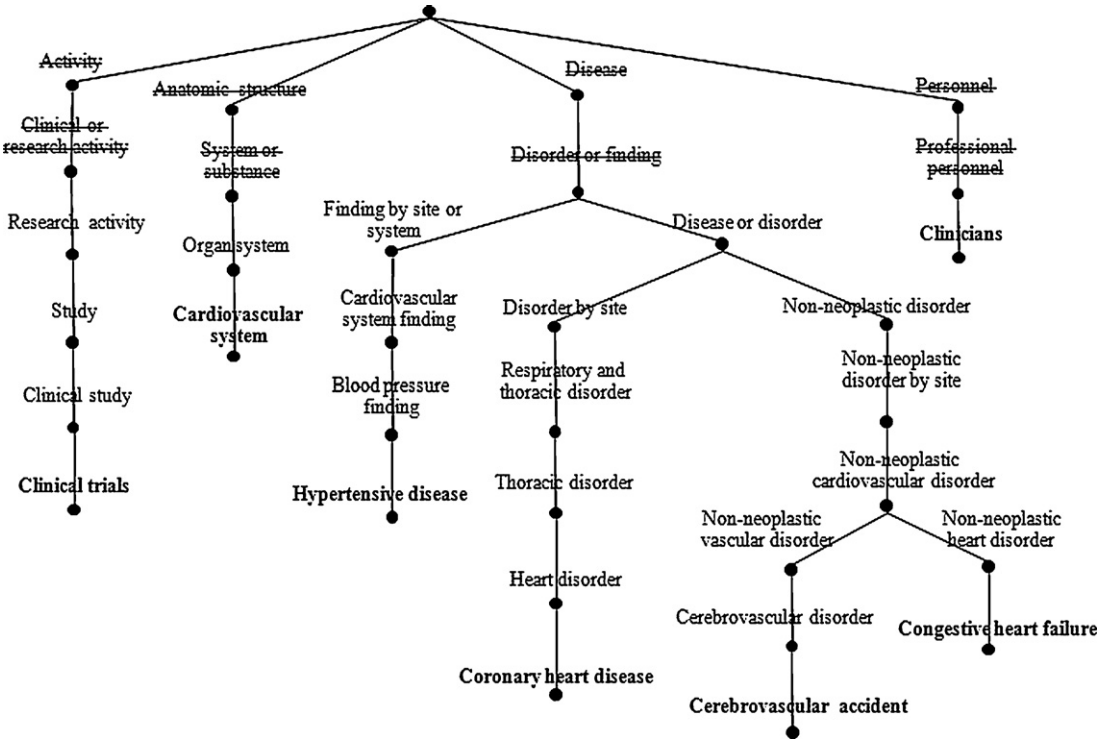

Fig. 3. An example of a sentence graph for the sentence *The goal of the trial was to assess cardiovascular mortality and morbidity for stroke, coronary heart disease and congestive heart failure, as an evidence-based guide for clinicians who treat hypertension.* Very general concepts that are ignored appear crossed out. Final concepts are shown in bold type.

where the edges (temporally unlabelled) represent semantic relations, and only a single vertex is created for each distinct concept in the text. Finally, the two upper levels of this hierarchy are removed, again because they represent very general concepts. Fig. 3 shows the graph for the example sentence used in the previous section.

3.1.4. Document representation

Next, all the sentence graphs are merged into a single *document graph*. This graph can be extended using more specific relationships between nodes to obtain a more complete representation of the document. In particular, in this work, the following sets of relations are tested: (1) no relation (apart from hypernymy), (2) the *associated with* relation between semantic types from the UMLS Semantic Network, (3) the *related to* relation between concepts from the UMLS Metathesaurus and (4) both the *associated with* and *related to* relations. To expand the document graph, only relations that link leaf vertices are added.

Fig. 4 shows an example of a document graph for a simplified document composed of two sentences extracted from the document *cvm-2-6-254.xml* from the BioMed Central corpus:

S1. The goal of the trial was to assess cardiovascular mortality and morbidity for stroke, coronary heart disease and congestive heart failure, as an evidence-based guide for clinicians who treat hypertension
S2. While event rates for fatal cardiovascular disease were similar, there was a disturbing tendency for stroke to occur more often in the doxazosin group, than in the group taking chlorthalidone.

Next, each edge of the document graph is assigned a weight in [0,1], as shown in Eq. (1). The weight of an edge, *e*, representing an *is_a* relation between two vertices, v_i and v_j (where v_i is a parent of v_j), is calculated as the ratio of the depth of v_i to the depth of v_j from the root of their hierarchy. The weight of an edge representing any other relation (i.e., *associated with* or *related to*) between a pair of leaf vertices is always 1.0. Thus, the weighting function attaches greater importance to specific concepts than to general ones.

$$\text{weight}(v_i, v_j) = \phi \quad \text{where} \begin{cases} \phi = \dfrac{\text{depth}(v_i)}{\text{depth}(v_j)} \text{ if } e \text{ represents an is\_a relation} \\ \phi = 1.0 \text{ otherwise} \end{cases} \tag{1}$$

This principle is shown in Fig. 4, where the *is_a* link between the concepts *cardiovascular drug* and *alpha-adrenergic blocking agent* is assigned the weight 1/2 because *cardiovascular drug* is ranked first in its hierarchy and *alpha-adrenergic blocking agent* is ranked second in the same hierarchy. The *related to* link between the leaf concepts *doxazosin* and *chlorthalidone* is assigned the weight 1.0.

3.1.5. Concept clustering

The following step groups the UMLS concepts in the document graph using a *degree-based clustering algorithm* similar to the one proposed in [24]. The aim is to construct sets or clusters of concepts that are closely related in meaning, under the assumption that each cluster represents a different subtheme in the document and that the most central concepts in the clusters (the centroids) give the necessary and sufficient information related to each subtheme.

The working hypothesis is that the document graph is an instance of a *scale-free network* [54]. A scale-free network is a complex network whose degree distribution follows a power law $P(k) \sim k^{-\gamma}$, where k stands for the number of links originating from a given node. The most notable property in this type of networks is that some nodes have a degree that considerably exceeds the average. These highest-degree nodes are often called *hubs*.

The *salience* of each vertex in the graph is then computed. Following [24], the salience of a vertex, v_i, is defined as the sum of the weights of the edges, e_j, that are connected to it, as shown in Eq. (2).

$$\text{salience}(v_i) = \sum_{e_j \,|\, \exists v_k \wedge e_j \text{connect}(v_i, v_k)} \text{weight}(e_j) \tag{2}$$

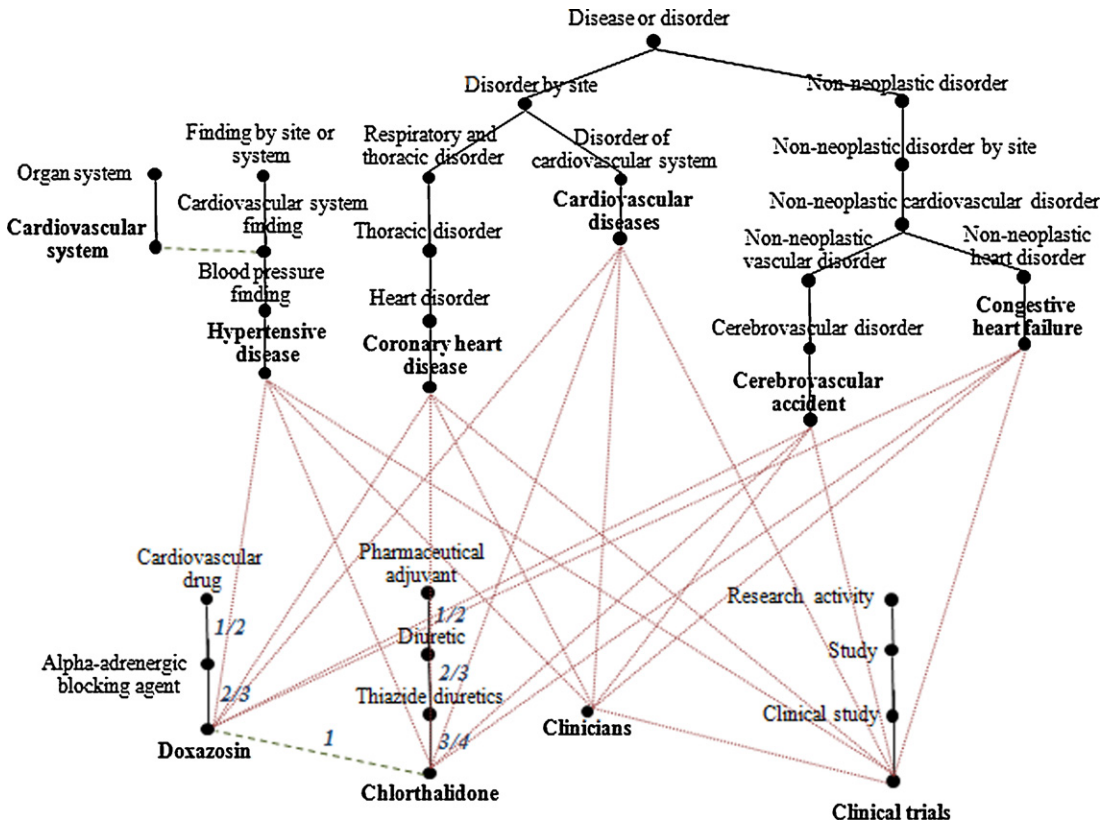

Fig. 4. An example of a simplified document graph from sentences S1 and S2. Continuous lines represent hypernymy relations; dashed lines represent *related to* relations; and dotted lines represent *associated with* relations. The edges of a portion of this graph have been labelled with their weights.

The n vertices with the highest salience (the *hub vertices*) represent the most connected nodes in the graph, taking into account both the number and the weight of the edges. The clustering algorithm starts grouping the hub vertices into *hub vertex sets* (HVSs) [24]. These HVSs can be interpreted as sets of concepts strongly related in meaning and will represent the centroids of the clusters. To construct the HVSs, the clustering algorithm first identifies the pairs of hub vertices that are most closely connected and merges them into a single HVS. Then, for each pair of HVSs, the algorithm checks whether the internal connectivity of the vertices they contain is lower than the connectivity between them. If so, the HVSs are merged. This decision is encouraged by the assumption that the clustering should show maximum intra-HVS connectivity but minimum inter-HVS connectivity. Intra-connectivity for a HVS is calculated as the sum of the weights of all edges connecting two vertices within the HVS, as shown in Eq. (3). Inter-connectivity for two HVSs is computed as the sum of the weights of all edges connecting two vertices, each vertex belonging to one of the HVSs, as shown in Eq. (4).

$$\text{intra-connectivity}(\text{HVS}_i) = \sum_{e_j \,\left|\, \exists v,w \in HVS_i \wedge e_j \text{connect}(v,w)\right.} \text{weight}(e_j) \quad (3)$$

$$\text{inter-connectivity}(\text{HVS}_i, \text{HVS}_j) = \sum_{e_k \,\left|\, \exists v \in HVS_i, w \in HVS_j \wedge e_k \text{connect}(v,w)\right.} \text{weight}(e_k) \quad (4)$$

Once the centroids of the clusters have been determined, the remaining vertices (i.e., those not included in the HVSs) are iteratively assigned to the cluster to which they are more connected. The connectivity between a vertex, v, and a cluster, C_i, is computed as the sum of the weights of the edges that connect the target vertex to the other vertices in the cluster, as shown in Eq. (5). Therefore, the final clusters consist of the HVSs resulting from the clustering algorithm plus the non-HVS vertices that are later attached to them.

$$\text{connectivity}(v, C_i) = \sum_{e_j \,\left|\, \exists w \in C_i \wedge e_j \text{connect}(v,w)\right.} \text{weight}(e_j) \quad (5)$$

Fig. 5 shows two fragments of two clusters from the document example. The purpose of this figure is to give readers an idea of the appearance of the clusters generated by the algorithm. The entire clusters present, respectively, 182 and 27 concepts. The clustering method produces four clusters for the full document. It may be observed that cluster A groups concepts related to diseases, syndromes and findings, as well as concepts regarding chemical and pharmacological substances, while cluster B collects concepts related to population and professional groups.

3.1.6. Sentence-to-cluster assignment

Once the concept clusters have been created, the aim of this step is to compute the semantic similarity between each sentence graph and each cluster. As the two representations are quite different in size, traditional graph similarity metrics (e.g., the edit distance [55]) are not appropriate. Instead, the similarity between a sentence graph and a cluster is computed using a non-democratic voting mechanism, so that each vertex, v_k, within a sentence graph, S_j, assigns a vote to a cluster, C_i, if the vertex belongs to the HVS of that cluster; half a vote if the vertex belongs to the cluster but not to its HVS; and no votes otherwise. The similarity between the sentence graph and the cluster is then calculated as the sum of the

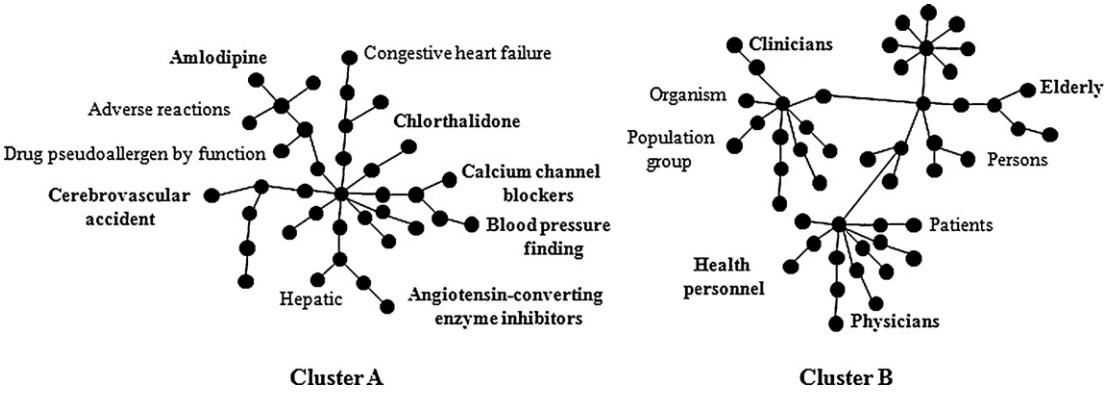

Fig. 5. An example of two fragments of two out of the four clusters extracted from the document example. The hub vertices are shown in bold type. The entire clusters present, respectively, 182 and 27 concepts.

votes assigned to the cluster by all vertices in the sentence graph, as shown in Eq. (6).

$$\text{semantic similarity}(C_i, S_j) = \sum_{v_k \,|\, v_k \in S_j} w_{k,j} \quad \text{where} \quad \begin{cases} v_k \notin C_i \Rightarrow w_{k,i,j} = 0 \\ v_k \in HVS(C_i) \Rightarrow w_{k,i,j} = 1.0 \\ v_k \notin HVS(C_i) \Rightarrow w_{k,i,j} = 0.5 \end{cases} \quad (6)$$

Next, for each cluster, the sentences are ranked in decreasing order of semantic similarity. It should be noted that a sentence may assign votes to several clusters (i.e., it may contain information about different themes).

To illustrate this process, consider the sentence S1 presented in Section 3.1.2 and its sentence graph shown in Fig. 3. The semantic similarity between this sentence graph and cluster A in Fig. 5 is equal to 2.5 because the concepts *blood pressure finding* and *cardiovascular accident* belong to the HVS of the cluster and each receive one vote, while the concept *congestive heart failure* belongs to the cluster but not to its HVS, thus receiving half a vote. The remaining concepts in the sentence graph do not belong to the cluster, and thus they do not receive any vote.

3.1.7. Sentence selection

At this point in, it is important to remember that extractive summarisation works by choosing salient sentences in the original document. In this work, sentence selection is assessed based on the similarity between sentences and clusters, as defined in Eq. (6). The number of sentences to be selected (N) depends on the desired summary compression. Three different heuristics for sentence selection have been investigated:

- *Heuristic 1*: Under the hypothesis that the cluster with the most concepts represents the main theme or topic in the document, the top ranked N sentences from this cluster are selected. The aim of this heuristic is to include in the summary only the information related to the main topic of the document.
- *Heuristic 2*: All clusters contribute to the summary in proportion to their sizes. Therefore, for each cluster, the top ranked n_i sentences are selected, where n_i is proportional to the size of the cluster. The aim of this heuristic is to include in the summary information about all the topics covered in the source.
- *Heuristic 3*: Halfway between the two heuristics above, this heuristic modifies Eq. (6) to compute a single score for each sentence as the sum of the votes assigned to each cluster, adjusted by their sizes, as shown in Eq. (7). Then, the N highest scoring sentences are selected. The aim of this heuristic is to select most of the sentences from the main topic of the document but also to include other secondary information that might be relevant to the user.

$$\text{semantic similarity}(S_j) = \sum_{C_i} \frac{\text{semantic similarity}(C_i, S_j)}{|C_i|} \quad (7)$$

Two additional features, apart from semantic similarity, have been tested in computing the relevance of the sentences: sentence position and similarity to the title. Despite their simplicity, these features are still commonly used in the most recent works on extractive summarisation [56,57].

- *Sentence position*: The position of the sentences in the document has been traditionally considered an important factor in finding the sentences that are most related to the topic of the document [16,56,57]. Sentences close to the beginning and the end of the document are expected to deal with the main theme of the document, and therefore more weight is assigned to them. In this work, a *position* score $\in [0,1]$ is calculated for each sentence as shown in Eq. (8), where M represents the number of sentences in the document and m_j represents the position of the sentence, S_j, within the document.

$$\text{position}(S_j) = \max\left\{\frac{1}{m_j}, \frac{1}{M - m_j + 1}\right\} \quad (8)$$

- *Similarity to the title:* The title given to a document by its author is intended to represent the most significant information in the document, and thus it is frequently used to quantify the relevance of a sentence [57]. In this work, the similarity of a sentence to the title is computed as the proportion of UMLS concepts in common between the sentence and the title, as shown in Eq. (9).

$$\text{title}(S_j) = \frac{\{\text{concepts\_}S_j\} \cap \{\text{concepts\_title}\}}{\{\text{concepts\_}S_j\} \cup \{\text{concepts\_title}\}} \quad (9)$$

The final selection of the sentences for the summary is based on the weighted sum of these feature values, as stated in Eq. (10). The values for the parameters λ, θ and χ must be determined empirically.

$$\text{score}(S_j) = \lambda \times \text{semantic similarity}(S_j) + \theta \times \text{position}(S_j) + \chi \times \text{title}(S_j) \quad (10)$$

It should be noted that the sentences in the summary are placed in the same order in which they appear in the source. Also, because a sentence may assign votes to several clusters or themes, heuristic 2 might include repeated sentences in the summary. To avoid this repetition, the system avoids adding to the summary any sentence that is already part of it. Finally, the tables and figures in the source that are referred to in any sentence belonging to the summary are also included in it.

3.2. Evaluation method

The purpose of the experiment is to evaluate the adequacy of semantic graphs for extractive summarisation and to compare the method with other well-known research and commercial summarisers. The evaluation is accomplished in two phases: (1) a preliminary experiment to find the best values for the different parameters involved in the algorithm and (2) a large-scale evaluation following the guidelines in the 2004 and 2005 Document Understanding Conferences (DUC[1]) [58,59].

3.2.1. Evaluation metrics: ROUGE

Although the evaluation of automatically generated summaries is a critical issue, there is still a controversy as to what the evaluation criteria should be, mainly due to the subjectivity in deciding whether or not a summary is of good quality [60]. Summarisation evaluation methods can be classified into two broad categories, *intrinsic* and *extrinsic*, depending on whether the outcome is evaluated independently of the purpose that the summary is intended to serve. Because the method proposed here is not designed for any specific task, the interest is on intrinsic evaluation. Intrinsic evaluation techniques test the summarisation itself, primarily by measuring two desirable properties of the summary: *coherence* and *informativeness*. Summary coherence refers to text readability and cohesion, while informativeness aims at measuring how much information from the source is preserved in the summary [61].

The automatically generated summaries may be evaluated manually, but this process is both very costly and time-consuming because it requires human judges to read not only the summaries but also the source documents. Besides, to objectively judge a summary has been proven difficult, as humans often disagree on what exactly makes a summary of good quality [62]. As a consequence, the research community has lately focused on the search for metrics to automatically evaluate the quality of a summary. Several metrics have been proposed to automatically evaluate informativeness [63,64]. However, to the best of our knowledge, research in automatic evaluation of coherence is still very preliminary [65,66].

In this work, the recall-oriented understudy for gisting evaluation (ROUGE) package [67] is used to evaluate the informativeness of the automatic summaries. ROUGE is a commonly used evaluation method that compares an automatic summary (called *peer*) with one or more human-made summaries (called *models or reference summaries*) and uses the proportion of *n*-grams in common between the peer and model summaries to estimate the content that is shared between them. The more content shared between the peer and model summaries, the better the peer summary is assumed to be. The ROUGE metrics produce a value in [0,1], where higher values are preferred, as they indicate a greater content overlap between the peer and model summaries. The following ROUGE metrics are used in this work: ROUGE-1 (R-1), ROUGE-2 (R-2), ROUGE-W-1.2 (R-W) and ROUGE-SU4 (R-SU4). R-*N* evaluates *n*-gram occurrence, where *N* stands for the length of the *n*-gram. R-W-1.2 computes the union of the longest common subsequences between the peer and the model summary sentences, taking into account consecutive matches. Finally, R-SU4 evaluates "skip bigrams", that is, pairs of words having intervening word gaps no larger than four words.

It should be noted, however, that ROUGE metrics do not account for text coherence, but merely assess the content of the summaries. An important drawback of ROUGE metrics is that they use lexical matching instead of semantic matching. Therefore, peer summaries that are worded differently but carry the same semantic information may be assigned different ROUGE scores. In contrast, the main advantages of ROUGE are its simplicity and its high correlation with the human judges gathered from previous DUC conferences [67].

3.2.2. Evaluation corpus

To the authors' knowledge, no corpus of model summaries exists for biomedical documents. However, most scientific papers include an abstract (i.e., the author's summary), which can be used as a model summary for evaluation.

In this work, a collection of 300 biomedical scientific articles randomly selected from the BioMed Central full-text corpus for text mining research [47] was used for evaluation. This corpus contains approximately 85,000 papers of peer-reviewed biomedical research, available in XML structured format, which allowed us to easily identify the title, abstract, figures, tables, captions, citation references, abbreviations, competing interests and bibliography sections. As stated in [68], the document sample size is large enough to allow significant evaluation results. The abstracts for the papers were used as reference summaries.

3.2.3. Algorithm parametrisation

A preliminary experiment was performed to determine, according to ROUGE scores, the optimal values for the parameters involved in the algorithm. This preliminary work addressed the following research questions:

1. Which set of semantic relations should be used to construct the document graph? (Section 3.1.4)
2. What percentage of vertices should be considered as hub vertices by the clustering method? (Section 3.1.5)
3. Does the use of traditional criteria (i.e., the position of the sentences and their similarity with the title) improve the quality of the summaries? (Section 3.1.7)
4. Which of the three heuristics for sentence selection produces the best summaries? (Section 3.1.7)

A separate development set was used for this parametrisation. This set consisted of 50 documents randomly selected from the BioMed Central corpus. Again, the abstracts of the papers were used as model summaries.

3.2.4. Comparison with other summarisers

Our approach was compared with three summarisers: two research prototypes (*SUMMA* and *LexRank)* and a commercial application (*Microsoft Autosummarize*). SUMMA [69] is a single- and multi-document summariser that provides several customisable statistical and similarity-based features to score the sentences for extraction. It is one of the most popular research summarisers and is publicly available. The features used for this evaluation include the position of the sentences within the document and within the paragraph, their overlap with the title and abstract sections, their similarity to the first sentence, and the frequency of their terms. Comparison with LexRank [9] will allow us to evaluate whether semantic information provides benefits over merely lexical information in graph-based summarisation approaches. Microsoft Autosummarize [70] is a feature of the Microsoft Word software and is based on a word frequency algorithm. In spite of its simplicity, word frequency is a well-accepted heuristic for summarisation. In addition, two baseline summarisers have been implemented. The first, *lead baseline*, generates summaries by selecting the first *N* sentences from each document. The second, *random baseline*, randomly selects *N* sentences from the document.

All automatic summaries were generated by selecting sentences until the summary is 30% of the original document size. This choice

[1] The DUC conferences (now the Text Analysis Conferences, TAC) are an initiative of the National Institute of Standards and Technology aimed at developing powerful summarisation systems and evaluation methods and at enabling researchers to participate in large-scale experiments.

Table 2
ROUGE scores for different combinations of semantic relations and percentages of hub vertices. The best results for each heuristic and set of relations are shown in italics, while the scores in bold indicate the best results for each heuristic.

		Heuristic 1		Heuristic 2		Heuristic 3	
		R-2	R-SU4	R-2	R-SU4	R-2	R-SU4
Hypernymy	*2%*	*0.3392*	*0.3134*	*0.3168*	*0.3075*	*0.3478*	*0.3227*
	5%	0.3163	0.2835	0.3060	0.3028	0.3414	0.3167
	10%	0.3311	0.2787	0.3033	0.2992	0.3388	0.3153
Hypernymy & associated with	*2%*	*0.3373*	*0.3128*	0.3079	0.3042	0.3502	0.3197
	5%	0.3334	0.3090	*0.3159*	*0.3065*	*0.3512*	*0.3204*
	10%	0.3128	0.3102	0.3025	0.3026	0.3090	0.3015
Hypernymy & related to	*2%*	*0.3394*	*0.3157*	0.3141	*0.3033*	0.3486	0.3240
	5%	0.3126	0.2901	*0.3145*	0.2954	*0.3492*	*0.3251*
	10%	0.3142	0.2954	0.2655	0.2539	0.3298	0.3056
Hypernymy & associated with & related to	*2%*	0.3359	0.3093	0.3149	0.3103	0.3443	0.3198
	5%	**0.3421**	**0.3165**	0.3123	0.3093	**0.3543**	**0.3299**
	10%	0.3316	0.2998	**0.3173**	**0.3110**	0.3516	0.3265

of summary size is based on the well-accepted heuristic that a summary should be between 15% and 35% of the size of the source text [71]. Although the length of the authors' abstracts is, on average, 17% of the length of documents, a larger size was preferred because the documents used for the experiments (i.e., scientific articles) are rich in information. The text in the tables and figures that are included in the summary was not taken into account when computing the summary size.

A Wilcoxon signed-rank test with a 95% confidence interval was used to test the statistical significance of the results.

4. Experimental results

4.1. Parametrisation results

To answer the questions raised in Section 3.2.3, three groups of experiments were performed. The first group was conducted to find the best combination of semantic relations for building the document graph (Section 3.1.4) and the best percentage of hub vertices for the clustering method (Section 3.1.5). Note that both parameters must be evaluated together because the relations influence the connectivity of the document graph and thus the optimum percentage of hub vertices. The results of these experiments are presented in Table 2. For legibility reasons, only R-2 and R-SU4 scores are shown.

It may be observed from Table 2 that the three heuristics behave better when all three semantic relations (i.e., hypernymy, *associated with* and *related to*) are used to build the document graph. However, the best percentage of hub vertices depends on the heuristic. Heuristics 1 and 3 perform better when 5% of the concepts in the document graph are used as hub vertices, while heuristic 2 registers the best outcome when the percentage of hub vertices is set to 10%. Heuristics 1 and 3 achieve slightly better results than heuristic 2, the best result being reported by the third heuristic. It may be also observed that, on average, the *associated with* relationship is more effective than the *related to* relation because the latter links together a relatively low number of concepts and thus produces a quite unconnected document graph. Another interesting result is that the optimal percentage of hub vertices increases with the number of relations (i.e., with the connectivity of the document graph).

The aim of the second group of experiments was to learn if the use of the *positional* and *similarity to the title* criteria to select sentences for the summaries helps to improve the content quality of these summaries (see Section 3.1.7). For these experiments, the percentage of hub vertices was set to 5% for heuristics 1 and 3 and to 10% for the second heuristic. All semantic relations were used to construct the document graph. The ROUGE scores for these tests are presented in Table 3, along with the values for the parameters λ, θ and χ that define the weight of each criterion in the linear function presented in Eq. (10). To determine the values of λ, θ and χ, all possible combinations that arise from varying λ from 0.5 to 1.0 and varying θ and χ from 0.1 to 0.5, at -0.1 intervals, were tested. However, for the sake of brevity, only the combinations that produced the best ROUGE scores are presented. It is worth mentioning that the experiments showed that λ values below 0.7 produce very poor results.

It may be seen from Table 3 that, according to the ROUGE scores, the use of the *positional* and *similarity to the title* criteria does not benefit heuristic 3. In contrast, the results obtained by the second heuristic improve slightly when both criteria are used. Regarding heuristic 1, while the positional criterion does not improve the scores for any of the ROUGE metrics, the effect of the *similarity to the title* criterion is not clear because the use of that criterion increases R-2 but decreases R-SU4. Again, heuristics 1 and 3 behave better than heuristic 2. Table 3 also shows that the *similarity to the title* criterion contributes more to the quality of the summaries than the *positional* one for all the heuristics.

Therefore, it may be concluded from Tables 2 and 3 that the best configuration for heuristic 1 involves using the three semantic relations with 5% of hub vertices and no information about the position of the sentences in the document. However, no definitive conclusions can be drawn about the use of the *similarity to the title* criterion, and thus, both configurations will be tested in the final evaluation. The best configuration for heuristic 2 involves using the three semantic relations with 10% of hub vertices, and both the *sentence position* and *similarity to the title* criteria with weights $\lambda = 0.8$, $\theta = 0.1$ and $\chi = 0.1$, respectively. In turn, heuristic 3 works

Table 3
ROUGE scores for different combinations of sentence selection criteria. The best results for each heuristic are shown in bold type.

				Heuristic 1		Heuristic 2		Heuristic 3	
	λ	θ	χ	R-2	R-SU4	R-2	R-SU4	R-2	R-SU4
Sentence salience	1.0	0.0	0.0	**0.3421**	0.3165	0.3173	0.3110	0.3543	0.3299
Sentence salience & position	0.9	0.1	0.0	0.3395	0.3128	0.3180	0.3118	0.3445	0.3209
Sentence salience & title similarity	0.9	0.0	0.1	0.3412	**0.3199**	0.3226	**0.3124**	0.3458	0.3238
Sentence salience & position & title similarity	0.8	0.1	0.1	0.3393	0.3145	**0.3239**	0.3123	0.3457	0.3221

Table 4
ROUGE scores for different versions of the summariser, two research systems (LexRank and SUMMA), a commercial application (Microsoft AutoSummarize) and two baselines (lead and random). The best score for each metric is shown in bold font. Systems are sorted by decreasing R-2 score.

	ROUGE-1	ROUGE-2	ROUGE-W-1.2	ROUGE-SU4
Heuristic 3	**0.7862**	**0.3543**	**0.2011**	**0.3299**
Heuristic 1	0.7682	0.3421	0.1965	0.3165
Heuristic 1+ sim. with title	0.7645	0.3387	0.1942	0.3132
Heuristic 2	0.7542	0.3239	0.1882	0.3123
LexRank	0.7302	0.3221	0.1865	0.3068
SUMMA	0.7144	0.3199	0.1833	0.2997
Lead baseline	0.6492	0.2587	0.1644	0.2648
AutoSummarize	0.5983	0.2448	0.1357	0.2322
Random baseline	0.4956	0.1732	0.1199	0.2301

best by using the three semantic relations with 5% of hub vertices and no other criterion for sentence selection (i.e., $\lambda = 1.0$, $\theta = 0.0$ and $\chi = 0.0$).

4.2. Evaluation results

To evaluate the summarisation performance, different types of summaries have been generated using (1) the three heuristics for sentence selection with their best configurations concluded in Section 4.1, (2) the SUMMA, LexRank and Microsoft Autosummarize systems, and (3) the lead and random baselines, as explained in Section 3.2.4. The ROUGE scores for all summarisers are presented in Table 4.

Table 4 shows that the three heuristics report higher ROUGE scores than the other summarisers and baselines. The best results are obtained using heuristic 3. Heuristic 1 (both with and without using the similarity with the title criterion) and heuristic 3 significantly improve all ROUGE metrics compared with SUMMA, LexRank, AutoSummarize and both baselines (Wilcoxon signed-rank test, $p < 0.05$). The second heuristic significantly improves all ROUGE metrics with respect to AutoSummarize and both baselines, while the improvement with respect to LexRank and SUMMA is only significant for R-1. On the other hand, it has been found that the first heuristic behaves slightly better without using the similarity to the title criterion. However, the differences with respect to using it are not statistically significant.

Concerning comparison between the three heuristics, the performance of heuristic 3 is significantly better than that of heuristic 2 for all ROUGE metrics (R-1: $p = 0.0005$; R-2: $p = 0.045$; R-W-1.2: $p = 0.002$; R-SU4: $p = 0.0475$) and also better than that of heuristic 1 for R-1 ($p = 0.007$) and R-SU4 ($p = 0.0475$). In contrast, the performance of heuristic 1 is significantly better than that of heuristic 2 for R-1 ($p = 0.021$) and R-2 ($p = 0.045$).

Finally, an important research question that immediately arises is why the ROUGE scores differ so much across documents. This is not shown in the tables (as they present the average results) but it has been observed during the experimentation and can be appreciated in Table 5. This table shows the standard deviation of the different ROUGE scores for the summaries generated by heuristic 3.

Table 5
Standard deviation of ROUGE scores for the summaries generated using heuristic 3.

	Standard deviation
ROUGE-1	0.08001
ROUGE-2	0.11624
ROUGE-W-1.2	0.04816
ROUGE-SU4	0.10135

5. Discussion

In this section, the experimental results presented in Section 4, both for the parametrisation and the final evaluation, are discussed. Various practical applications of the summarisation method are also proposed.

5.1. Algorithm parametrisation

We first discuss the results of the parametrisation performed to determine the optimal values for the parameters involved in the summariser and provide answers to the questions raised in Section 3.2.3. These results were presented in Section 4.1.

First, concerning the set of semantic relations that should be used to build the document graph, Table 2 showed that the three heuristics behave better when all three semantic relations (i.e., hypernymy, *associated with* and *related to*) are used. However, they differ in the optimal percentage of hub vertices used to cluster the concepts in the graph (5% for heuristics 1 and 3 *versus* 10% for heuristic 2). This difference exists because the three heuristics aim to produce different types of summaries. It is worth remembering that the aim of heuristic 2 is to generate summaries covering all topics presented in the source document, regardless of their relative relevance. Thus, it is not sufficient to consider only the concepts dealing with the main topic of the document as hub vertices, but also those dealing with other secondary information.

Second, with respect to the use of traditional criteria for sentence selection (i.e., the position of the sentences in the document and their similarity to the title), Table 3 shows that while heuristic 3 does not benefit from any of these criteria, heuristic 1 produces comparable ROUGE scores regardless of whether or not similarity with the title criterion is used, but such scores decrease when the positional criterion is employed. The results reported by heuristic 2, however, improve when both criteria are used. The reason is that, because heuristic 2 aims to cover all topics in the document, and because frequently some of these topics are irrelevant to the summary, the use of these additional criteria, especially the similarity to the title, biases the selection of sentences toward the information related to the main topic of the document.

We have also found that the similarity to the title criterion contributes more to the quality of the summaries than the positional criterion for all three heuristics. This result is not surprising because scientific papers are not (a priori) expected to present the core information at the beginning and end of the document, as occurs in other types of documents such as news articles. The first sentences in scientific papers usually introduce the problem and motivation of the study, whereas the last sentences provide conclusions and future work. However, the most important information is usually presented in the middle sentences, as part of the method, results and discussion sections. Therefore, it seems that a more appropriate positional criterion would be one that attaches greater importance to sentences belonging to such central sections.

Third, regarding the best heuristic for sentence selection, Tables 3 and 4 showed that heuristic 3 reports the highest ROUGE scores. To understand why this heuristic behaves better than the others, we first examined the authors' abstracts for the 50 documents in the development set. We found that the information in these abstract (i.e., the information considered most important by the authors of the papers) can be classified into three main sections or categories: (1) the background of the study, (2) the method or case presentation and (3) the results and conclusions of the study. The method section includes approximately 58% of the information in the abstract; the results and conclusions section comprises around 25%; and the background section involves less than 17%. We next analysed the clusters generated by the clustering method and found that it usually produces a single large cluster and a variable number of small clusters. The large cluster contains the concepts related to the central topic of the document, while the others include concepts related to secondary information. Although some of the concepts within the large cluster may be found in all three sections of the abstracts, the majority of the concepts in this cluster are usually found in the section describing the method. Therefore, it seems clear that any heuristic for sentence selection that aims to compare well with the authors' abstracts should mainly include information related to the concepts within this large cluster (i.e., information related to the main topic of the document). Hence, heuristic 2 is, by definition, at disadvantage compared with heuristics 1 and 3 when the authors' abstracts are used as model summaries. This result does not mean that heuristic 2 is worse than the others, but rather that it aims to generate a different type of summary.

In spite of this result, the differences among the heuristics are not as remarkable as expected. A careful analysis of the summaries generated by the three heuristics for the 50 documents in the development set suggests that the explanation for this finding is that, given the larger size of the main cluster, the three heuristics extract most of their sentences from this cluster, and hence the summaries generated share most of the sentences in common. Nevertheless, the best results are reported by heuristic 3. It has been determined that this heuristic selects most of the sentences from the most populated cluster, but it also includes some sentences from other clusters. Thus, in addition to the information related to the central topic, this heuristic also includes other secondary or "satellite" information that might be relevant to users. In contrast, the first heuristic fails to present this information, whereas the second heuristic includes more secondary information but sometimes omits some of the core information.

5.2. Comparison with other summarisers

We next discuss the results of the final evaluation and compare our system to other summarisers (see Section 3.2.4). These results were presented in Section 4.2.

Table 4 shows that the ROUGE scores achieved by all variants of the concept graph-based method are significantly better than those of all other summarisers and baselines. These results seem to indicate that using domain-specific knowledge improves summarisation performance compared with traditional word-based approaches, in terms of the informative content quality of the summaries generated. The use of concepts instead of terms along with the semantic relations that exist between them allows the system to identify the different topics covered in the text more accurately and with comparative independence of the vocabulary used to describe them. As a consequence, the information in the sentences selected for the summaries is closer to the information in the model abstracts.

A further test has been performed to compare the performance of heuristic 3 with that of the Reeve et al. [4] summarisation approach, which also employs domain-specific information (UMLS concepts and semantic types) to represent the documents. Reeve et al. address the same problem as the one presented here but uses a different evaluation strategy. They use a corpus of 24 oncology papers to generate summaries with a 20% compression rate and compare the automatic summaries with four model summaries: three models generated by three domain experts, using sentence extraction, and the abstract of the paper. Nonetheless, only the average ROUGE scores for the four models are given. Their best summarisation method reports a R-2 score = 0.12653 and a R-SU4 score = 0.22303. The method proposed here, when run with a 20% compression rate over the 300 documents in the corpus, obtains a R-2 score = 0.2568 and a R-SU4 score = 0.2385. Although these results seem to outperform those reported by Reeve et al., it must be noted that they are not directly comparable due to the use of a different corpus and a different evaluation methodology (in particular, the use of a combination of extracts and abstracts as model summaries).

5.3. Differences across documents

The experiment also showed that the ROUGE scores differ considerably across different documents (see Table 5). To clarify the reasons for these differences, the two extreme cases (that is, the two documents with the highest and lowest ROUGE scores, respectively, for the summaries generated using the third heuristic) were carefully examined. The best case turned out to be one of the largest documents in the corpus, while the worst case was one of the shortest (six pages *versus* three pages). According to the starting hypothesis (i.e., the document graph is an instance of scale-free network), as the graph grows, the new concepts are likely to be linked to other highly connected concepts, and thus the hubs are expected to increase their connectivity at a higher rate. Therefore, the difference in connectivity between the hubs and the remaining vertices is expected to be more marked in large graphs than in modestly sized ones. We think that this fact leads to a better separation of the clusters generated in large graphs than in smaller ones and thus to a better delimitation of the topics covered in the document.

A second interesting difference between the documents is in their underlying subject matter. The best case is published in the *BMC Biochemistry* journal and concerns the reactions of certain proteins over the brain synaptic membranes. In contrast, the worst case is published in the *BMC Bioinformatics* journal and concerns the use of pattern matching for database searching. It has been verified that the UMLS covers the vocabulary in the first document better than the vocabulary in the second one, in terms of both concepts and relations, which leads to a more accurate graph that better reflects the content of the document.

Finally, in the worst-case document, the use of synonyms is quite frequent, which does not occur in the best-case document. For instance, a concept is referred to in the document body as *string searching*, but it is always referred to as *pattern matching* in the abstract. Because the ROUGE metrics are based on the number of word overlaps, the summaries containing synonyms of the terms in the abstract are unreasonably penalised.

5.4. Practical applications

In light of the experimental results, we believe that the summaries generated by the proposed method may help physicians and biomedical researchers in several ways.

First, automatic summaries may be useful in anticipating the contents of the original documents, so that users may decide which of the documents to read further. Even with the author's abstract, there are two main reasons for wanting to generate text summaries from a full-text [4]: (1) the abstract may be missing relevant content

from the full-text, and (2) there is not a single ideal summary, but rather, the ideal summary depends on the user's information needs. In this line of use, an interesting application would be the integration of the summariser within the PubMed search engine and the use of a preview tool that allows users to visualise the summaries and quickly choose the documents that best match what they are looking for without having to read the entire documents. Moreover, the users' queries may be used to guide the summary generation process and thus to bias the summaries toward their information needs. To this end, the similarity of each sentence in the document to the user's query may be computed and added to Eq. (10) as a feature for sentence selection.

Second, extending the method to produce query-driven summaries will also allow us to deal with more challenging types of documents, such as EHR. The use of automatic procedures to summarise the information in EHR is an extremely complex and subtle issue [72] that remains virtually unexplored. First, typing and orthographic errors are quite frequent in EHR, as is the use of non-standard acronyms and abbreviations. Second, the summariser should be able to capture the clinical relevance of the concepts (i.e., diseases, syndromes, etc.) discussed in the record, regardless of their frequency and their relations with other concepts within the record. Third, there is considerable variation in the structure of clinical records, so that exploiting such structure in the summarisation process becomes arduous. Therefore, it is not feasible to think of the automatic summaries as substitutes for the source documents. However, there are other scenarios in which such summaries may be useful. Physicians, for instance, often need to refer to previous patient cases when seeking information for a new or untypical case. Given a query that states the physician's information need and the set of EHR returned by a search engine, each record may be accompanied by an automatically generated summary that highlights the information most relevant to the topic indicated in the user's query. Moreover, if the EHR follows a standard structure, the user may indicate which sections should be ignored and which ones should be given more relevance in generating the summary. Each sentence in the record may be weighted according to the section in which it appears simply by adding a further feature to Eq. (10).

6. Conclusions

In this paper, an efficient approach to biomedical text summarisation has been presented. The method represents the document as a semantic graph using UMLS concepts and relations. In this way, it produces a richer representation than the one provided by traditional models based on terms. The method has been evaluated on a collection of 300 scientific biomedical papers from BioMed Central. It compares favourably with existing approaches, which confirms that the use of domain-specific knowledge can be very useful in automatic summarisation, particularly when dealing with technical or specialised domains. Three different heuristics for sentence selection have been proposed, each aiming to construct a different type of summary according to the type of information in the source that is likely to be included in the summary. It has been found that, when the automatic summaries are evaluated against the authors' abstracts, the best heuristic is the one that selects most of the information from the main topic of the document, but also includes other secondary or "satellite" information that might be relevant to users.

However, it has been found that the efficiency of the algorithm decreases under poor UMLS coverage of concepts in the document to be summarised. Therefore, future work will concentrate on addressing this problem. Specifically, the feasibility of using a general-purpose lexicon (e.g., WordNet [27]) to capture the concepts not covered by UMLS, as in the BioSquash system [26], will be studied.

Moreover, in the short term, we plan to extend the method to produce query-driven summaries. We will also carefully analyse the structure of biomedical scientific papers to weight the sentences according to the section in which they appear. Finally, we will study the possibility of adapting the system to produce query-driven summaries of EHR.

Acknowledgement

This research is funded by the Spanish Government through the FPU program and the projects TIN2009-14659-C03-01 and TSI 020312-2009-44.

References

[1] Afantenos SD, Karkaletsis V, Stamatopoulos P. Summarization from medical documents: a survey. Artificial Intelligence in Medicine 2005;33(2):157–77.
[2] Lau A, Coeira E. Impact of web searching and social feedback on consumer decision making: a prospective online experiment. Journal of the American Informatics Association 2008;11:320–31.
[3] Hunter L, Cohen KB. Biomedical language processing: perspective what's beyond PubMed? Molecular Cell 2006;21(5):589–94.
[4] Reeve LH, Han H, Brooks AD. The use of domain-specific concepts in biomedical text summarization. Information Processing and Management 2007;43:1765–76.
[5] Fiszman M, Rindflesch TC, Kilicoglu H. Abstraction summarization for managing the biomedical research literature. In: Proceedings of the computational lexical semantics workshop at HLT-NAACL 2004. 2004. p. 76–83.
[6] Brooks AD, Sulimanoff I. Evidence-based oncology project. Surgical Oncology Clinics of North America 2002;11:3–10.
[7] Gay CW, Kayaalp M, Aronson AR. Semi-automatic indexing of full text biomedical articles. In: Friedman CP, Ash J, Tarczy-Hornoch P, editors. Proceedings of the American Medical informatics association annual symposium. Willow Grove, PA, USA: Hanley & Belfus Inc.; 2005. p. 271–5.
[8] National Library of Medicine. Technical report, identification of important text in full text articles using summarization. p. 21. http://ii.nlm.nih.gov/resources/Summarization_and_FullText.pdf [accessed 02.10.10].
[9] Erkan G, Radev D. LexRank: graph-based centrality as salience in text summarization. Journal of Artificial Intelligence Research 2004;22:457–79.
[10] Mihalcea R, Tarau P. TextRank – bringing order into text. In: Proceedings of the 2004 conference on empirical methods in natural language processing (EMNLP). 2004. p. 404–11.
[11] Nelson S, Powell T, Humphreys B. The Unified Medical Language System (UMLS) project. In: Kent A, Hall CM, editors. Encyclopedia of library and information science. Marcel Dekker, Inc.; 2002. p. 369–78.
[12] Fleishchman S, Language, medicine D, Schiffirn D, Tannen HE, Hamilton, editors. The handbook of discourse analysis, vol. 24. Malden, MA, USA: Blackwell Publishing; 2004. p. 470–502.
[13] Sparck-Jones K. Automatic summarizing: factors directions. In: Mani MT, Maybury I, editors. Advances in automatic text summarization. Cambridge, MA, USA: The MIT Press; 1999. p. 1–12.
[14] Mani I, Maybury MT, editors. Advances in automatic text summarization. Cambridge, MA, USA: The MIT Press; 1999.
[15] Mani I. Automatic summarization natural language processing, vol. 3. Amsterdam/Philadelphia: John Benjamins Publishing Company; 2001.
[16] Brandow R, Mitze K, Rau LF. Automatic condensation of electronic publications by sentence selection. Information Processing and Management 1995;5(31):675–85.
[17] Luhn HP. The automatic creation of literature abstracts. IBM Journal of Research Development 1958;2(2):159–65.
[18] Edmundson HP. New methods in automatic extracting. Journal of the Association for Computing Machinery 1969;2(16):264–85.
[19] Kupiec J, Pedersen JO, Chen F. A trainable document summarizer. In: Fox E, Ingwersen P, Fidel R, editors. Proceedings of the 18th annual international ACM SIGIR conference on research and development in information retrieval. New York, NY, USA: ACM Press; 1995. p. 68–73.
[20] Brin S, Page L. The anatomy of a large-scale hypertextual web search engine. Computer Networks and ISDN Systems 1998;30(1–7):107–17.
[21] Litvak M, Last M. Graph-based keyword extraction for single-document summarization. In: Bandyopadhyay S, Poibeau T, Saggion H, Yangarber R, editors. Proceedings of the workshop on multi-source multilingual information extraction and summarization (Coling 2008). Stroudsburg, PA, USA: ACL Publications; 2008. p. 17–24.
[22] Kleinberg J. Authoritative sources in a hyperlinked environment. Journal of the ACM 1999;46(5):604–32.
[23] Barzilay R, Elhadad M. Using lexical chains for text summarization. In: Proceedings of the intelligent scalable text summarization workshop (ISTS'97). 1997. p. 10–8.

[24] Yoo I, Hu X, Song I-Y. A coherent graph-based semantic clustering and summarization approach for biomedical literature and a new summarization evaluation method. BMC Bioinformatics 2007;8(9):1–15.
[25] National Library of Medicine. Medical Subject Headings (MeSH). http://www.nlm.nih.gov/mesh/ [accessed 02.10.10].
[26] Shi Z, Melli G, Wang Y, Liu Y, Gu B, Kashani M, et al. Question answering summarization of multiple biomedical documents. In: Kobti Z, Wu D, editors. Proceedings of the 20th Canadian conference on artificial intelligence, lecture notes in computer science 4509. Berlin, Germany: Springer-Verlag; 2007. p. 284–95.
[27] Miller GA, Beckwith R, Fellbaum C, Gross D, Miller K. Introduction to WordNet: an on-line lexical database; 1993. p. 1–86. http://wordnetcode.princeton.edu/5papers.pdf [accessed 02.10.10].
[28] Rindflesch TC, Fiszman M. The interaction of domain knowledge and linguistic structure in natural language processing: interpreting hypernymic propositions in biomedical text. Journal of Biomedical Informatics 2003;36:462–77.
[29] Vanderwende L, Suzuki H, Brockett C, Nenkova A. Beyond SumBasic: task-focused summarization with sentence simplification and lexical expansion. Information Processing and Management 2007;43(6):1606–18.
[30] Jonnalagadda S, Tari L, Hakenberg J, Baral C, Gonzalez G. Towards effective sentence simplification for automatic processing of biomedical text. In: Collins M, Narayanan S, Oardandm DW, Vanderwend L, editors. Proceedings of HLT-NAACL 2009, short papers. ACL Publications, Stroudsburg, PA, USA; 2009. p. 177–80.
[31] Jonnalagadda S, Gonzalez G. Sentence simplification aids protein–protein interaction extraction. In: Rebholz-Schuhmann D, Collier N, Park JC, Wong L, editors. Proceedings of the 3rd international symposium on languages in biology and medicine. 2009. p. 109–14.
[32] Lin J, Wilbur WJ. Syntactic sentence compression in the biomedical domain: facilitating access to related articles. Information Retrieval 2007;10:393–414.
[33] Nadkarni PM. Information retrieval in medicine: overview and applications. Journal of Postgraduate Medicine 2000;46(2):122–66.
[34] National Library of Medicine. Unified Medical Language System (UMLS). http://www.nlm.nih.gov/research/umls/ [accessed 02.10.10].
[35] National Library of Medicine. UMLS Specialist Lexicon fact sheet. http://www.nlm.nih.gov/pubs/factsheets/umlslex.html [accessed 02.10.10].
[36] National Library of Medicine. UMLS Metathesaurus fact sheet. http://www.nlm.nih.gov/pubs/factsheets/umlsmeta.html [accessed 02.10.10].
[37] National Library of Medicine. UMLS Semantic Network fact sheet. http://www.nlm.nih.gov/pubs/factsheets/umlssemn.html [accessed 02.10.10].
[38] International Health Terminology Standards Development Organisation. SNOMED-CT. http://www.ihtsdo.org/snomed-ct/ [accessed 02.10.10].
[39] National Library of Medicine. MetaMap Portal. http://mmtx.nlm.nih.gov/ [accessed 02.10.10].
[40] Aronson AR. Effective mapping of biomedical text to the UMLS Metathesaurus: the MetaMap program. In: Bakken S, editor. Proceedings of the American medical informatics association annual symposium. Willow Grove, PA, USA: Hanley & Belfus Inc.; 2001. p. 17–21.
[41] Eck M, Vogel S, Waibel A. Improving statistical machine translation in the medical domain using the Unified Medical Language System. In: Proceedings of the 20th international conference on computational linguistics. 2004. p. 792–8.
[42] Overby CL, Tarczy-Hornoch P, Demner-Fushman D. The potential for automated question answering in the context of genomic medicine: an assessment of existing resources and properties of answers. BMC Bioinformatics 2009;10(Suppl. 9 (S8)):1–8.
[43] Aronson AR, Rindflesch TC. Query expansion using the UMLS Metathesaurus. In: Masys DR, editor. Proceedings of the American medical informatics association annual symposium. Willow Grove, PA, USA: Hanley & Belfus Inc.; 1997. p. 485–9.
[44] Plaza L, Díaz A. Retrieval of similar electronic health records using UMLS concept graphs. In: Hopfe CJ, Rezgui Y, Métais E, Preece A, Li H, editors. Proceedings of the 15th international conference on applications of natural language to information systems (NLDB 2010), lecture notes in computer science 6177. Berlin, Germany: Springer-Verlag; 2010. p. 296–303.
[45] Plaza L, Díaz A, Gervás P. Automatic summarization of news using WordNet concept graphs. In: Weghorn H, Roth J, Isaías P, editors. Proceedings of the IADIS international conference informatics. Lisbon, Portugal: IADIS Press; 2009. p. 19–26.
[46] Plaza L, Lloret E, Aker A. Improving automatic image captioning using text summarization techniques. In: Sojka P, et al, editors. Proceedings of the 13th international conference on text, speech and dialogue, lecture notes in artificial intelligence 6231. Berlin, Germany: Springer-Verlag; 2010. p. 165–72.
[47] BioMed Central Corpus. http://www.biomedcentral.com/info/about/datamining/ [accessed 02.10.10].
[48] Schwartz A, Hearst M. A simple algorithm for identifying abbreviation definitions in biomedical text. In: Jung T, editor. Proceedings of the pacific symposium on biocomputing, vol. 8. Singapore: World Scientific Publishing; 2003. p. 451–62.
[49] BioText (Abbreviation Definition Recognition Software). http://biotext.berkley.edu/software.html [accessed 02.10.10].
[50] PubMed StopWords. http://www.ncbi.nlm.nih.gov/entrez/query/static/help/pmhelp.html#Stopwords [accessed 02.10.10].
[51] GATE (Generic Architecture for Text Engineering). http://gate.ac.uk/ [accessed 02.10.10].
[52] Shooshan SE, Mork JG, Aronson AR. National library of medicine. Technical report. Ambiguity in the UMLS Metathesaurus; 2009 Edition, p. 46. http://skr.nlm.nih.gov/papers/references/ambiguity09.pdf [accessed 02.10.10].
[53] Humphrey S, Rogers W, Kilicoglu H, Demner-Fushman D, Rindesch T. Word sense disambiguation by selecting the best semantic type based on journal descriptor indexing: preliminary experiment. Journal of the American Society for Information Science and Technology 2006;57:96–113.
[54] Barabasi A, Albert R. Emergence of scaling in random networks. Science 1999;286:509–12.
[55] Levenshtein V. Binary codes capable of correcting deletions, insertions and reversals. Doklady Akademii Nauk SSSR 1966;163(4):845–8.
[56] Bossard A, Généreux M, Poibeau T. Description of the LIPN systems at TAC 2008: summarizing information and opinions. In: Proceedings of the first text analysis conference (TAC 2008). 2008.
[57] Bawakid A, Oussalah M. A semantic summarization system: university of Birmingham at TAC 2008. In: Proceedings of the first text analysis conference (TAC 2008). 2008.
[58] National Institute of Standard and Technology. Document Understanding Conferences (DUC). http://duc.nist.gov/ [accessed 02.10.10].
[59] Litkowski KC. Summarization experiments in DUC 2004. In: Proceedings of the 2004 document understanding conference (DUC 2004). 2004.
[60] Radev D, Lam W, Elebi AC, Teufel S, Blitzer J, Liu D, et al. Evaluation challenges in large-scale document summarization. In: Hinrichs EW, Roth D, editors. Proceedings of the 41st annual meeting on association for computational linguistics. Stroudsburg, PA, USA: ACL Publications; 2003. p. 375–82.
[61] Mani I. Summarization evaluation: an overview. In: Adachi J, Kando N, editors. Proceedings of the 2nd NTCIR workshop on research in Chinese & Japanese text retrieval and text summarization. Tokyo, Japan: National Institute of Informatics; 2001.
[62] Sparck-Jones K, Galliers J. Evaluating natural language processing systems: an analysis and review. Berlin, Germany: Springer-Verlag; 1995.
[63] Tratz S, Hovy E. Summarization evaluation using transformed Basic Elements. In: Proceedings of the text analysis conference (TAC). 2008.
[64] Giannakopoulos G, Karkaletsis V, Vouros G, Stamatopoulos P. Summarization system evaluation revisited: *N*-gram graphs. ACM Transactions on Speech and Language Processing 2008;5(3):1–39.
[65] Pitler E, Nenkova A. Revisiting readability: a unified framework for predicting text quality. In: Lapata M, Ng HT, editors. Proceedings of the 2008 conference on empirical methods in natural language processing. Stroudsburg, PA, USA: ACL Publications; 2008. p. 186–95.
[66] Vadlapudi R, Katragadda R. Quantitative evaluation of grammaticality of summaries. In: Gelbukh A, editor. Proceedings of the 11th international conference on computational linguistics and intelligent text processing (CICLing 2010), lecture notes in computer science 6008. Berlin, Germany: Springer-Verlag; 2010. p. 736–47.
[67] Lin C-Y. ROUGE: a package for automatic evaluation of summaries. In: Moens M-F, editor. Proceedings of workshop on text summarization branches out, post-conference workshop of ACL. Stroudsburg, PA, USA: ACL Publications; 2004. p. 74–81.
[68] Lin C-Y. Looking for a few good metrics: automatic summarization evaluation – how many samples are enough? In: Adachi J, Kando N, editors. Proceedings of the 4th NTCIR workshop on research in information access technologies information retrieval, question answering and summarization. Tokyo, Japan: National Institute of Informatics; 2004.
[69] Saggion H. SUMMA: a robust and adaptable summarization tool. Revue Traitement Automatique des Langues 2008;49(2):103–25.
[70] Microsoft Corporation. Microsoft Office online: automatically summarize a document. http://office.microsoft.com/en-us/word/HA102552061033.aspx [accessed 02.10.10].
[71] Hovy EH. Automated text summarization. In: Mitkov R, editor. The Oxford handbook of computational linguistics. Oxford, United Kingdom: Oxford University Press; 2005. p. 583–98.
[72] Smith CA. Information retrieval in medicine: the electronic medical record as a new domain. In: Proceedings of the American society for information science and technology, vol. 43, no. 1. 2006. p. 1–30.