Survey of Biomedical Semantic Annotation Systems For Text-based Documents

Paper ID: BIO-115

ABSTRACT

This paper examines semantic annotation systems designed for text documents in the biomedical domain, and reviews their mapping algorithm, performance evaluation and failure analysis. Biomedical text annotation focuses on annotating general biomedical concepts, such as disease types, symptoms, treatments and outcomes, rather than annotating biological named entities such as protein and gene names. The output of biomedical research is largely documented as findings in the form of literature written in free-form text format. Free-form text is subject to language variations when describing the same concept. By mapping text units to domain-specific concepts, the effect of language variation can be diminished. Concepts allow physicians and biomedical researchers to find relevant information, stored in large text repositories, to apply to healthcare problems.

Categories and Subject Descriptors

A.1 [**General Literature**]: Introduction and Survey

General Terms

Performance, Design.

Keywords

Biomedical Semantic Annotation, Biomedical Text, Information Extraction.

1. INTRODUCTION

The output of biomedical research is largely documented as findings in the form of literature written in free-form text format [1]. The written texts are then accumulated in large online databases made readily accessible due to recent advances in software and communications. For example, the PUBMED database provided by the United States National Library of Medicine contains over 16 million publications from over 4,800 journals [2]. A key problem in such large free-form text repositories is finding relevant information for physicians and biomedical researchers to apply to healthcare issues. One method to find relevant information is to first annotate each text with domain-specific concepts, and then use the concepts to retrieve and analyze texts. The use of domain-specific concepts overcomes issues such as a language variation.

This paper presents a survey of semantic annotation systems used to perform annotation of text documents in the biomedical domain. Examples of biomedical text sources include biomedical research publications, such as clinical trials in oncology, and patient records. The purpose of biomedical semantic annotation systems is to map free-form biomedical text, typically noun phrases, into specific biomedical concepts. Biomedical concepts are defined by domain experts. Examples of concept resources include the UMLS Metathesaurus, which contains concepts and real-world instances of the concepts, including a concept name and its synonyms, lexical variants, and translations [3]. The UMLS Metathesaurus is derived from over 100 different vocabulary sources resulting in over one million biomedical concepts. Biomedical text annotation focuses on annotating general biomedical concepts, such as disease types, symptoms, treatments and outcomes. This is a similar but different task than biomedical named entity recognition, which focuses on annotating specific biological entities such as protein and gene names.

The rest of paper is organized as follows. Section 2 presents the general annotation approach of biomedical annotation systems. Section 3 presents an overview of each of the systems and describes their approach to mapping free-form text to concepts, accuracy and weaknesses in mapping. Section 4 concludes the paper.

2. GENERAL APPROACH

The basic problem in biomedical text annotation is to map a unit of free-form source text to one or more concept instances stored in the domain resource, such as UMLS MetaMap. A common approach for mapping is to use the following method:

1. Construct a unit of analysis by generating subsets of words in the source text (e.g., phrase, sentence).

2. (Optional) Normalize the source text unit using UMLS [4] by (a) removing possessives, (b) replacing punctuation with spaces, (c) removing stop words, (d) converting words to lower-case, (e) breaking a string into constituent words, and (f) sorting words into alphabetical order.

SAC'08, March 16-20, 2008. Fortaleza, Brazil

3. For each word in the input phrase, build a set of all concepts containing the word.

4. Find the intersection of the concept sets.

5. (Optional) Find the best matching concept based on the common word membership between the source text and concept text.

The unit of analysis is usually a phrase. Noun phrases are typically used because they have more content information [5]. Other units of analysis include sentences [6], and unordered words [7]. Phrases lose contextual value when a single concept is spread across multiple phrases [8], [7]. Sentences overcome the loss of context as compared to phrases, but suffer from: (a) finding invalid concepts when using all permutations of words in the sentence, and (b) since more words are in a sentence than in a phrase, concept identification becomes more computationally complex [8]. It is also possible to eliminate natural language processing and treat the entire text as a set of independent words, which is the approach used by the IndexFinder system [7]. A problem with this bag-of-words approach is that concepts cannot be linked back to their source (e.g., phrase or sentence), since the words are permuted throughout the text. Another problem is the over-generation of concepts, which the authors control with various filters, such as semantic types and word ranges.

Four types of matches between the words of a source text and UMLS concept phrase have been defined in the literature [9]:

1. *None*: there is no match of words.
2. *Simple*: there is an exact match between the words in the text of the source and the UMLS concept text.
3. *Partial*: one or more words do not match exactly.
4. *Complex*: the original source phrase is divided into two or more sets of words which are then mapped to distinct concepts.

The identification of a phrase is typically done using natural language processing (NLP), where a part-of-speech tagger is first executed over a sentence, and then using parts of speech to build a noun phrase. Other approaches include using NLP Heuristics which do not use strict NLP processing [10], moving window where all combinations of a sentence words are generated to build a phrase [11], and using pre-defined block text sizes [6].

3. SYSTEM OVERVIEWS

This section briefly describes several currently available biomedical semantic annotation systems. The systems were chosen from a literature review of biomedical semantic annotation work. The idea is to survey, within a brief format, the algorithmic approaches, evaluation and failure analysis for a set of systems which cover a broad range of approaches.

3.1 Concept Locator

The Concept Locator [8] system uses a phrase-based approach to identify concepts for indexing and retrieval.

Algorithm: Input phrases from the source text are identified using the IBM Intelligent Text Miner's Feature Extraction Tool. Concept Locator uses a subset of UMLS which removes redundant concepts, concepts unrelated to clinical medicine, concepts having eight or more words, and suppressible synonyms (synonyms which result in ambiguity [12]). The algorithm steps are as follows: (1) The input phrase is stripped of stop words and words not appearing in UMLS. Any resulting phrase over five words is rejected. (2) An exact match of concept words to input phrase words is attempted using all words in the input phrase to match all concepts having the same set of words, regardless of word order. If the match results in mapping the input phrase to a distinct concept, the concept is output and the mapping algorithm stops. (3) If no exact match is found, two cases are handled. In the first case, a phrase consists of a single word. If the word is defined as ambiguous by UMLS, no further matching is attempted. In the second case, complete concept word matches for all combinations of words {*N*-1, *N*-2,…2} resulting in distinct concept matches is performed, where N is the total number of words in the source phrase. Words which are not matched are sent back to algorithm step #2. If the subset matching does not result in any concept matches, individual words in the input phrase are matched to single-word concepts. (4) If the input phrase from step 2 or 3 matches several concepts, concept disambiguation is performed by stemming both the words in the concept and the input phrase, and then comparing each for an exact match of the stemmed words. The disambiguation step is on a best-effort basis; that is, it is possible for disambiguation to fail and still result in mapping an input phrase to multiple concepts.

Evaluation: A manual evaluation of the concept mapping output was performed using a corpus of 24 biomedical documents (12 each of discharge summaries and surgery notes). A domain expert manually identified UMLS concepts in the corpus texts, and then compared the Concept Locator concept mappings to a manually-annotated corpus. It was found Concept Locator matched concepts correctly for 76.3% of the phrases.

Failure Analysis: The authors identify three categories of concept-mapping failures: (a) use of noun phrases, (b) UMLS content, and (c) matching algorithm. The use of noun phrases cannot locate concepts spread across two or more noun phrases, resulting in matching two or more concepts rather than a specific single concept. Also, spelling, grammatical errors, and proper names can confuse natural language parsers. UMLS content influences performance because it is incomplete. UMLS does not list all possible word variations, may have missing biomedical concepts, and has redundant concepts. The matching algorithm has built-in limitations, such as five-word maximum phrase length, which causes some phrases not to be mapped.

3.2 KNOWLEDGEMAP

The KnowledgeMap system [13] identifies indexing concepts in biomedical educational texts.

Algorithm: KnowledgeMap's concept identification component is known as KnowledgeMap Concept Identifier (KMCI) and consists of three phases: sentence identification, concept identification, and concept disambiguation. Sentence and noun phrase identification is done by using a natural language parser. Concept identification is done by taking a noun phrase and finding a set of UMLS concepts which match the noun phrase. If no concept match is found, then variants of the words in the noun phrase are generated and the matching process retried. Nearby noun phrases linked by grammar (such as conjunctions and

prepositions) are attempted to match concepts with the current noun phrase. This overcomes the phrase-based problem of identifying concepts occurring over two or more noun phrases. In addition, KMCI will distribute modifying adjectives. For example, "large and small intestine" is converted to "large intestine and small intestine" [13]. If multiple concepts for a phrase are located, disambiguation is attempted. Two resources are used to perform disambiguation. The first resource is an externally-maintained list of concept co-occurrences which occur in MEDLINE abstracts. The second resource is a dynamic list of concepts with an exact match to a source text noun phrase. Disambiguation is performed by discarding concepts which are not similar to either the text's dynamic list of exact-match concepts or which do not co-occur with concepts in MEDLINE abstracts.

Evaluation: Evaluation of KnowledgeMap was done by first having two domain experts manually annotate five biomedical educational texts with important words and phrases. Each text was then split into its component sentences, and each sentence was then submitted to KMCI and MetaMap, a state-of-the-art concept matching system produced by the National Library of Medicine. The domain experts then determined if the concepts for the identified words and phrases were accurate or not. Precision and recall are the evaluation measures. Recall is defined as the number of important words and phrases identified. Precision is defined as the number of correctly identified concepts. The recall is measured at 86% and precision at 92%, which outperforms MetaMap, which has a recall of 81% and a precision of 89% [14].

Failure Analysis: The authors identified nine primary reasons for failure. The biggest failure of concept matching (62% of failures) in KMCI is the lack of a corresponding concept in UMLS. Other sources of failure include acronym/abbreviation/hyphen handling, and overmatching. Overmatching occurs when no exact match exists between the input phrase and UMLS phrases, and additional words in UMLS concept phrases are used to find a match with the input phrase. The disambiguation stage was responsible for only 2% of failures, while accounting for 18% of successful matches.

3.3 IndexFinder

IndexFinder [7] is a concept matching system designed for online indexing applications.

Algorithm: IndexFinder uses a series of in-memory table structures to find all possible concepts by using all combinations of words in the text. IndexFinder treats the words within a text independently regardless of order. The words in a text are first normalized by (a) lowercasing them, (b) removing unknown acronyms/abbreviations, stop words, and words unknown to UMLS, and (c) mapping remaining words to their base form. Next, for each unique word, all UMLS concept phrases containing the word are retrieved. Each retrieved phrase maintains its length and the count of words matched so far. After all words have been evaluated, the retrieved phrases are then evaluated based on their counts. Concepts are extracted for indexing where concept phrases have all matching words with the source phrase.

Evaluation: The authors used a corpus of 5,783 patient reports totaling 10.8MB in size and report a text processing speed of 42.7KB/second. No evaluation was reported on the accuracy of the concepts extracted, although one was planned to evaluate the number of false positives and false negatives.

Failure Analysis: No failure analysis was reported. However, six filters are available to restrict the output. The filters focus on two aspects: (a) restricting the location of text, and (b) restricting the number of concepts generated. In order to generate concepts based on smaller units (i.e., not the full text), the word length can be restricted to less than six words or less than 11 words, as well as all words. In addition, the range filter restricts words to a certain distance, such as ten words. Words outside of the range are not counted in concept matching. The effect is to index only concepts occurring with words a certain distance from each other. To restrict the generation of concepts, concept subsets can be removed if they are contained within a larger concept. In addition, concepts can be included only if they appear within certain UMLS semantic types.

3.4 MetaMap Transfer

MetaMap Transfer (MMTx) [12] is a concept matching system produced by the National Library of Medicine. MMTx is considered a state-of-the-art system for concept matching [14]. MMTx was originally developed to support indexing applications, but has also been used in data mining, decision support, and patient record applications.

Algorithm: The MMTx algorithm consists of five steps. Parsing of the source text by a natural language parser is first done to find noun phrases. Variant generation on each noun phrase's words is done to find each word's acronyms, abbreviations, synonyms, inflectional and spelling variants. UMLS concept phrases containing the word or its variants are identified. Each concept phrase is evaluated to find the best match with the source noun phrase four metrics to measure similarity: centrality, variation, coverage, and cohesiveness [9]. Centrality is the phrase head word used in the concept phrase. Variation is the distance of a word variant from the source word. Coverage measures word overlap between the concept phrase and source noun phrase, ignoring word gaps. Cohesiveness measures similarly to coverage, but factors in sequences of words which co-occur in the concept phrase and the source noun phrase. The four scores are mixed to form a weighted average. The coverage and cohesiveness scores are weighted twice as heavily as centrality and variation. The final stage forms a final mapping of the source text, which may result in mapping a source text into one or more concepts. No explicit disambiguation step is performed.

Evaluation: Several evaluations of MMTx have been performed to date. The National Library of Medicine (NLM) performed a failure analysis of MMTx by using a short evaluation [15]. Five annotators without domain expertise annotated two documents from genetic information Web site. Two documents were chosen as the evaluation corpus size due to (a) the labor intensive activity of annotating documents, and (b) mediating conflicting concept mappings between annotators. The two documents were processed by MMTx and the MMTx concept output was then compared to the manually-generated annotations. The recall score for the two documents was measured at 53%; precision was not calculated.

The University of Washington (UW) also performed an evaluation of MMTx [16] using six domain experts to identify concepts within a corpus of 60 texts. The study used reported a

recall of 53% for exact matches and 93% for partial matches. Precision was reported as 28% for exact matches and 55% for partial matches.

Failure Analysis: The NLM and UW evaluations also concluded with a failure analysis. The NLM study identified thirteen sources of failure. The most common failures resulted from (a) needing implicit knowledge to map a word, (b) the use of broader concepts by annotators because the UMLS Metathesaurus is incomplete, and (c) co-reference resolution. The UW study identified four types of failures: incorrect splitting of a noun phrase, concept ranking and identification failed, and noun phrase breaking which changed the meaning of the phrase.

3.5 PhraseX

The PhraseX program performs noun extraction [17]. A study of UMLS phrases in MEDLINE abstracts used the output of PhraseX to perform simple concept matching [18]. A newer application uses PhraseX as a component of a larger biomedical text indexing application [19].

Algorithm: Noun phrase identification is done by first tagging a sentence's words with their part-of-speech, and then using the barrier word method [20] to delimit a phrase. In this case, tagger output delimits a phrase based on its part-of-speech. For example, a verb ends one phrase and begins another. PhraseX defines three types of successively complex phrases: (a) *simp* – phrases with a head noun, (b) *macro* – phrases with a preposition to the right, and (c) *mega* – using a finite word to divide the sentence into two phrases: one phrase before the verb and one phrase after the verb. Once a phrase is identified, the mapping is done in one of three ways: (a) exact match, (b) exact match with lower casing done to all words in the phrase, and (c) exact matching done after UMLS normalization (lower casing, possessive removal, inflectional variation, and word sorting, among others).

Evaluation: The PhraseX evaluation was performed by downloading all MEDLINE abstracts from Fall 2001 PubMed. The noun phrases were extracted resulting in approximately 175 million unique phrases. The authors report that 63% of the phrases were *simp* phrases, 16% were *macro* phrases, and 21% were *mega* phrases. Each unique phrase was then attempted to be mapped with one of the three matching methods. The result is 6.5% for exact match, 22.5% for lower case match, and 30% for normalized match. There were exact matches only; partial matches were excluded.

Failure Analysis: Five types of failures were identified. 1) The UMLS Metathesaurus contains strings which are not useful for mapping. Examples are long descriptive strings and strings containing codes for what are known as Logical Observations Identifiers, Names and Codes (LOINC). 2) Syntactic analysis of text cannot address ambiguity due to grammar and writing style. 3) Trade names are not always included in the UMLS Metathesaurus. 4) Conservative constraints on the definition of a macro phrase.

3.6 SAPHIRE

The SAPHIRE system [6] was originally developed to support indexing applications, but was later used to support other applications, such as extraction of concepts from patient medical records [21].

Algorithm: The original SAPHIRE algorithm is substantially different from the latest version. The original algorithm uses strict pattern matching where all words in the source text phrase must be present and in the same order as the UMLS phrase [6]. The latest algorithm [22] relaxes the strict requirements and allows for partial matches and out-of-sequence words. The algorithm finds noun phrases using the barrier word method [20]. For each word found, a list of UMLS concepts containing the word is retrieved. UMLS concepts having high-frequency words must also have low frequency words as well or the concepts are excluded. This is to eliminate concepts containing low-content words. The individual word concept lists are merged, and any concept having less than one-half the number of words from the input text is excluded. The resulting set of UMLS concepts is then scored. The highest score occurs if all words in the concept appear in the source text. Word order is ignored. If there is no exact word match, a weighting formula scores the concept based on proportion of words between source text and concept, word proximity, and length of word matches between source text and concept [23].

Evaluation: SAPHIRE has been evaluated in the context of effectiveness as an information retrieval system [24], but a more recent evaluation focusing on finding best concepts rather than all concepts was completed using radiology reports [23]. Fifty radiology reports were processed using SAPHIRE to extract UMLS concepts. Precision and Recall were the evaluation measures. Precision was defined as number of correctly mapped concepts divided by number of total concepts. Recall was defined as the number of correctly mapped concepts divided by total number of correct concepts. Precision was measured at 30% and Recall at 63%.

Failure Analysis: A failure analysis of the radiology report concept matching was performed. The failures affecting recall were due to scoring errors, where the correct concept was scored lower than other concepts, or with issues regarding the barrier method of phrase identification. Failures affecting precision include negation in the phrase which was not identified, and disambiguating competing concepts. A key problem with SAPHIRE's algorithm is that it may return multiple matching concepts for a text segment which, due to partial matching, are incorrect concept mappings [8]. The disambiguation problem was handled by adding a semantic type filter, which filters out concepts belonging to a particular semantic type.

3.7 SENSE

The SENSE (SEarch with New Semantics) system [25] is designed to map user queries to the National Library of Medicine's Medical Subject Heading (MeSH) words. The system addresses the mismatch between user's natural language queries and MeSH index words. The idea is to map user queries to MeSH words, which are indexed by biomedical information retrieval systems.

Algorithm: In order to translate source text into concepts, SENSE translates a user query (a short phrase) into what the authors call *semantic factors*. Semantic factors are base concepts which cannot be decomposed further. SENSE defines 3,400 semantic factors. Semantic factors are constructed from an input phrase by having the Semantic Analyzer component look up phrase words in a knowledge base, which handles variants, such as spelling and plural forms, and produce identical semantic factors for all input

phrases having the same meaning. The output is a suggested list of MeSH words which can be used to perform a search.

Evaluation: No evaluation was performed.

Failure Analysis: No failure analysis was performed because no evaluation was done. However, it should be noted that the purpose of SENSE is to build a list of suggested MeSH words. No effort is made to identify the best matching concept. Therefore, SENSE needs a disambiguation stage to filter out concepts which are not the best match for a phrase.

4. CONCLUSION

In this paper a short survey of semantic annotation systems designed for text documents in the biomedical domain was presented and included reviews of each system's mapping algorithm, performance evaluation and failure analysis. Biomedical text annotation is distinguished from traditional biomedical named entity recognition by identifying general biomedical concepts as opposed to biological entities such as gene names. Biomedical semantic annotation is important in the biomedical domain for applications, such as information retrieval, data mining, and text summarization, which utilize concepts rather than terms to identify important areas of information.

5. REFERENCES

[1] G. Nenadic, H. Mima, I. Spasic, S. Ananiadou and J. Tsujii, "Terminology-driven literature mining and knowledge acquisition in biomedicine," *Int. J. Med. Inf.*, vol. 67, pp. 33-48, 2002.

[2] United States National Library of Medicine, "PubMed," 2006.

[3] United States National Library of Medicine. (2006, 28 March 2006). UMLS metathesaurus fact sheet.

[4] National Library of Medicine, United States. (2006, Specialist lexicon and lexical tools. *2006(August 31),* pp. 1.

[5] P. Elkin, J. Cimino, H. Lowe, D. Aronow, T. Payne, P. Pincetl and G. Barnett, "Mapping to MeSH(the art of trapping MeSH equivalence from within narrative text)." in 1988, pp. 185-190.

[6] W. R. Hersh and R. A. Greenes, "SAPHIRE--an information retrieval system featuring concept matching, automatic indexing, probabilistic retrieval, and hierarchical relationships," *Comput. Biomed. Res.*, vol. 23, pp. 410-425, Oct. 1990.

[7] Q. Zou, W. W. Chu, C. Morioka, G. H. Leazer and H. Kangarloo, "IndexFinder: A method of extracting key concepts from clinical texts for indexing," in *Proceedings of the AMIA Annual Symposium*, 2003, pp. 763-767.

[8] P. Nadkarni, R. Chen and C. Brandt, "UMLS Concept Indexing for Production Databases," *Journal of the American Medical Informatics Association,* vol. 8, pp. 80-91, 2001.

[9] A. R. Aronson, "MetaMap: Mapping Text to the UMLS Metathesaurus," 1996.

[10] P. M. Nadkarni, "Concept locator: a client-server application for retrieval of UMLS metathesaurus concepts through complex boolean query," *Comput. Biomed. Res.*, vol. 30, pp. 323-336, Aug. 1997.

[11] D. Wollersheim, W. Rahayu and J. Reeve. Evaluation of index term discovery in medical reference text. Presented at Proceedings of the International Conference on Information Technology and Applications.

[12] A. R. Aronson, "Effective mapping of biomedical text to the UMLS metathesaurus: The MetaMap program," in *Proceedings of the AMIA Symposium 2001,* 2001, pp. 17-21.

[13] J. C. Denny, P. R. Irani, F. H. Wehbe, J. D. Smithers and A. Spickard 3rd, "The KnowledgeMap project: development of a concept-based medical school curriculum database," *Proceedings of the Annual AMIA Symposium*, pp. 195-199, 2003.

[14] J. C. Denny, J. D. Smithers, R. A. Miller and A. Spickard, ""Understanding" Medical School Curriculum Content Using KnowledgeMap," *Journal of the American Medical Informatics Association,* vol. 10, pp. 351-362, 2003.

[15] G. Divita, T. Tse and L. Roth, "Failure analysis of MetaMap Transfer (MMTx)," *Medinfo,* vol. 11, pp. 763-767, 2004.

[16] W. Pratt and M. Yetisgen-Yildiz, "A study of biomedical concept identification: MetaMap vs. people," *AMIA. Annu. Symp. Proc.*, pp. 529-533, 2003.

[17] National Library of Medicine, United States, "PhraseX and the SPECIALIST Minimal Commitment Parser," vol. 2006, pp. 1, March 16. 2004.

[18] S. Srinivasan, T. C. Rindflesch, W. T. Hole, A. R. Aronson and J. G. Mork, "Finding UMLS Metathesaurus concepts in MEDLINE," *Proc. AMIA. Symp.*, pp. 727-731, 2002.

[19] National Library of Medicine, United States, "Medical Text Indexer," vol. 2006, pp. 1, August 22. 2006.

[20] K. W. F. Tersmette, A. F. Scott, G. W. Moore, N. W. Matheson and R. E. Miller, "Barrier word method for detecting molecular biology multiple word terms," in 1988, pp. 207-211.

[21] W. R. Hersh and L. C. Donohoe, "SAPHIRE International: a tool for cross-language information retrieval," *Proc. AMIA. Symp.*, pp. 673-677, 1998.

[22] W. Hersh and T. J. Leone, "The SAPHIRE server: a new algorithm and implementation," *Proc. Annu. Symp. Comput. Appl. Med. Care.*, pp. 858-862, 1995.

[23] W. R. Hersh, M. Mailhot, C. Arnott-Smith and H. J. Lowe, "Selective Automated Indexing of Findings and Diagnoses in Radiology Reports," *J. Biomed. Inform.*, vol. 34, pp. 262-273, 2001.

[24] W. Hersh, D. H. Hickam, R. B. Haynes and K. A. McKibbon, "Evaluation of SAPHIRE: an automated approach to indexing and retrieving medical literature," *Proc. Annu. Symp. Comput. Appl. Med. Care.*, pp. 808-812, 1991.

[25] Y. L. Zieman and H. L. Bleich, "Conceptual mapping of user's queries to medical subject headings," *Proc. AMIA. Annu. Fall. Symp.*, pp. 519-522, 1997.