Online Biomedical Concept Annotation Using Language Model Mapping

Lawrence H. Reeve[*], PhD[1], Hyoil Han[†], PhD[2], and Ari D. Brooks, MD[3]
[1]IBM Corporation, Marlton, NJ; [2]College of Information Science and Technology, Drexel University, Philadelphia, PA; [3]College of Medicine, Drexel University, Philadelphia, PA

Abstract

We report the results of applying language technology to the bioinformatics problem of online concept annotation of biomedical text. Our online biomedical text concept annotator, CONANN, finds the best-matching biomedical concept for a biomedical text source phrase. Biomedical concepts are defined in resources such as the Unified Medical Language System Metathesaurus. The goal of CONANN is to improve annotation speed without losing annotation accuracy as compared to offline systems, facilitating the use of concept annotation in online environments. In this paper, CONANN has been extended to use language models built from nearly a million biomedical concepts to perform matching of source text phrases to domain-specific concept instances. Intrinsic and extrinsic evaluations show accuracy competitive with a state-of-the-art biomedical text concept annotator with a speed improvement of more than four times. We expect CONANN to be useful in tasks such as question-answering, text summarization, information extraction, and concept-based indexing and retrieval.

Introduction

The task of a biomedical concept annotator is to map each unit of source text, such as a phrase, into one or more domain-specific concepts. Biomedical concepts are defined in resources such as the Unified Medical Language System (UMLS) [1] and National Cancer Institute (NCI) Thesaurus [2]. The biomedicine community maintains large and continuously-updated information sources. For example, United States National Library of Medicine's PubMed service contains in excess of 16 million citations from over 5,000 worldwide biomedicine-related journals (United States National Library of Medicine, 2006). The PubMed service consists of citations and abstracts in addition to linking with full-texts. For physicians and biomedical researchers, finding and using relevant texts within these large resources can be challenging. To address this challenge, annotation systems using domain-specific concepts, rather than terms, have been developed. Examples of such systems include MetaMap Transfer (MetaMap)[3], SAPHIRE[4], and KnowledgeMap[5]. Among the benefits of using concepts, rather than terms, is 1) synonym merging, where synonymous phrases are merged to a single concept, and 2) the use of a domain-specific language for querying. Biomedical concept annotations have been used in applications for indexing and retrieval, data mining, decision support, patient records, medical curriculum searching, and text summarization [3] [5] [6].

In previous work we reported promising initial results of our online concept annotator, CONANN [7]. In this paper, we present the results of extending CONANN to use concept language models to find the best matching domain-specific concept for a biomedical text phrase.

Language Model Concept Mapping

CONANN utilizes biomedical concepts built from pre-defined phrases. A source phrase is a phrase from the source text. A concept instance is phrase belonging to a UMLS concept. Each UMLS concept is associated with one or more synonymous phrases. Candidate phrases are concept instances having words in common with the source phrase. A candidate concept identifies the UMLS concept a candidate phrase belongs to. A concept name is the name given to a particular UMLS concept.

The extended CONANN in this paper uses a unigram language model, which is widely used in information retrieval research [8] [9]. Other language model applications, such as speech recognition, use more than one n-gram to capture the order of the text. The unigram model is more advantageous in concept annotation applications because it allows for gaps in word order when matching a source phrase to a concept instance, where exact matches usually do not occur. In addition, the unigram language model has

[*] This research was conducted when the first author was at Drexel University
[†] Corresponding author

proven an effective approach in information retrieval, outperforming the vector space model [9]. We are not aware of any biomedical concept annotators using a language model approach.

A unigram language model is built for each of the concepts in UMLS. A concept language model contains word identifiers, frequencies of words in a concept, and the probability of the word occurring across all concept instances within the concept, that is, $P(w) = \frac{|w|}{N}$, where w is a word in a particular concept's concept instances, $|w|$ is a count of the number of times the word appears in all of the concept instances of a concept, and N is the total number of words in all of the particular concept's concept instances. In the UMLS resource we used, there were 797,152 concepts defined resulting in the generation of 797,152 corresponding unigram language models.

All of the UMLS concepts are assumed to be independent, even though the UMLS Metathesaurus concepts are organized into a hierarchical form using the UMLS Semantic Network. The Semantic Network categorizes and provides relationships between UMLS concepts. The concept independence assumption allows us to evaluate the effectiveness of language models in the concept mapping task without considering other factors which may influence performance.

The overall annotation strategy of a single biomedical source phrase using CONANN with a language model extension is shown in Figure 2 and is as follows:

1. *Candidate Phrase Generation*: Construct a list of candidate phrases based on the common words between all UMLS concept instances and the source phrase. If only one candidate phrase remains, return its associated concept name.

2. *Candidate Phrase Filtering*: To reduce the number of phrases which must be evaluated by language modeling, the list of candidate phrases is trimmed using a coverage filter, which measures word overlap between a source phrase and a concept instance. The coverage filter is based on the Involvement score used by the MetaMap system [10]. Involvement scoring first computes the normalized word weights of words in common between a source phrase and a candidate phrase in both directions (phrase involvement), and then averages the two phrase involvement values to determine the candidate phrase score.

Once all candidate phrases have been scored using the involvement score, the standard deviation of the involvement phrase scores for the set of candidate phrases is calculated. A threshold value is chosen as the mean candidate involvement score plus two standard deviations. All candidate phrases having involvement phrase scores greater than or equal to the threshold value are passed to the language model concept mapper. By using two standard deviations as a threshold, we capture the top 5% [11] of candidate phrases. If no candidate phrases have a involvement phrase score value greater than or equal to the threshold value, the candidate phrases with the highest involvement phrase score value are passed to the final concept mapper. Two scenarios are possible after the involvement coverage filter is applied. 1) If only one candidate phrase remains, its associated concept name is returned. 2) If more than one candidate phrase remains, they are passed to the final concept mapper.

3. *Final Mapping*: The final concept mapping stage is responsible for finding the best-matching candidate phrase (or phrases in cases of scoring ties) among remaining candidate phrases. CONANN uses a multinomial unigram language model to find the concept most likely to have generated the source phrase. A list of candidate phrases is first retrieved from the coverage filter. Since each candidate phrase belongs to one or more concepts, a list of concepts is generated from the candidate instances to form a set of candidate concepts. Each candidate concept is then assigned a score based on its probability of generating the source phrase. The probability score is calculated as shown in Figure 1, which is a standard unigram language mixture model [8] which combines the source word (w) probability within the concept language model ($M_{concept}$ in Figure 1) with the probability of the word occurring across the entire UMLS phrase collection ($M_{conceptCollection}$ in Figure 1).

$$P(concept) = \prod_{w \in SrcPhrase} ((1-\lambda)P(w \mid M_{concept} + \lambda P(w \mid M_{conceptCollection}))$$

Figure 1: Multinomial Unigram Language Mixture Model for Concept Mapping

We initially set λ=0.5 to balance the concept language model with the collection model. We also evaluated the extreme values of λ=0.1 and λ=0.9, but did not notice any change in the final concept annotation output. To allow for more word variation, we also expanded the source phrase words to include all source phrase word variants of each source phrase word. CONANN uses the source phrase word variants that are provided by the UMLS resources.

.

The highest-scoring candidate concepts are then output as the best-matching concept for the source phrase. In the case of ties, all of the highest-scoring concepts are output.

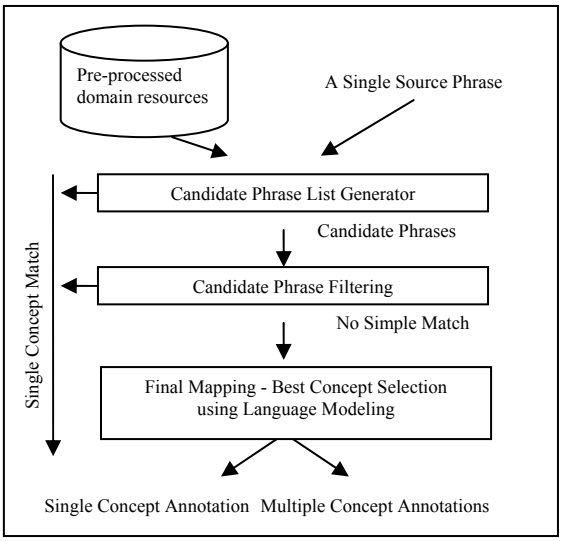

Figure 2: Concept Mapping Process in CONANN with a language model extension
(Adaptation of the old CONANN's architecture in Reeve & Han [7])

Related Work

Most work in semantic annotation for biomedical text is performed to support semantic indexing/retrieval and data mining of biomedical texts [3]. Our work is most closely related to MetaMap [3], KnowledgeMap [5], and SAPHIRE [4]. We focus on scoring candidate phrases for mapping, since that is one of the primary differences between systems, SAPHIRE uses simple and partial mapping, and for candidate phrase scoring combines measures of term overlap, term proximity, and length of term matches. KnowledgeMap uses simple and partial matching, and for candidate phrase scoring uses an exact match approach and if no matches are found, performs iterative variant-word-generation and re-matching. KnowledgeMap also offers a disambiguation stage which uses concept co-occurrence information derived from existing medical texts to find a best-matching concept. MetaMap uses simple, partial and complex mapping. MetaMap scores candidate phrases using a mixture of four different scores: *Centrality*, *Variation*, *Coverage*, and *Coherence*. Compared to SAPHIRE, our CONANN uses simple and partial matching, but does not score every candidate phrase for final mapping. Like KnowledgeMap and MetaMap, we incorporate word variants of the source phrase, but we do not incorporate disambiguation or exact matching as KnowledgeMap does or extensive word variants generation as MetaMap does. Our system reduces computational complexity by deferring complex scoring until after most candidate phrases have been eliminated. In addition, we build a language model of each concept's phrases, whereas existing systems consider each candidate phrase as independent of one another, even from the same concept.

Other related systems include SENSE [12], which translates source and concept instance to low-level semantic factors, then performs exact matching of the semantic factors; Concept Locator [13] which simply sub-divides a phrase and looks for exact matches; PhraseX [14] which focuses on phrase identification and performs an exact match with candidate phrases; and IndexFinder [15] which treats the source text as a bag of words and finds all matching words, regardless of their location. In some systems, such as MetaMap [3], efforts are made to find a best-matching concept, while in other systems, such as IndexFinder [15], all possible concepts are found. The difference is usually determined by the application in which the annotations will be used. For example, finding all concepts within a source text is useful in search and retrieval indexing, while best-matching annotations are useful in applications such as text summarization where the meaning of small text units (e.g. phrases), is needed.

Evaluation Methods

Evaluation of CONANN is done using both intrinsic and extrinsic methods. The intrinsic evaluation is intended to evaluate the speed and accuracy of CONANN against an existing state-of-the-art biomedical concept annotator. The extrinsic evaluation is designed to measure the effect of the annotator's output on a task. We chose concept-based text summarization as the task. A corpus of unique phrases and a subset of texts for summarization was constructed from a citation database of approximately 1,200 oncology clinical trial papers physicians feel are important to the field [16]. The final corpus consists of 1,628 unique phrases for the intrinsic evaluation and 24 texts each with three summaries generated by third-year medical students for use with the extrinsic evaluation.

Intrinsic Evaluation: The intrinsic evaluation is intended to evaluate the speed and accuracy of CONANN against an existing biomedical concept annotator. We use the MetaMap system [3] provided by the United States National Library of Medicine as the baseline system. To measure the amount of time it takes for MetaMap to annotate the corpus of phrases, MetaMap was executed using 1,628 unique

.

phrases from our evaluation corpus. CONANN was then executed against the same set of 1,628 phrases. To measure the accuracy of CONANN as compared to MetaMap, the number of CONANN concept annotations matching MetaMap was divided by the number of phrases, giving a precision metric [17]. The average time to annotate a phrase is calculated by taking the total annotation time of all phrases divided by the total number of phrases annotated.

The average time to annotate a phrase using MetaMap was 208 milliseconds, while for CONANN it was 47 milliseconds, an improvement of speed of nearly 4.5 times. The total time to annotate all phrases in the corpus was 5.67 minutes using MetaMap and 1.28 minutes for CONANN.

We assume that the precision of MetaMap is 1 (i.e. 100%). The precision of CONNAN compared to MetaMap was 0.85. If we relax the concept matching so that the top five concepts produced by CONANN for a phrase were compared to MetaMap, the precision rises to 0.93.

Extrinsic Evaluation: The output of a concept annotator is a list of phrases and their associated domain-specific concepts. This output is an intermediate format, not directly useable by an end-user. The extrinsic evaluation is a complimentary evaluation to the intrinsic, designed to show the usefulness of the concept output in some task. We selected text summarization as the end-user task. We used two probabilistic summarizers, FreqDist [18] and a version of SumBasic [19] modified to use concepts rather than terms. Both summarizers use concept frequency as the sole feature to select salient sentences. Both summarizers' performance is entirely reliant on the frequency of concepts identified in the texts. It is expected if the concept output is accurate, summarization performance will improve because the concepts will have identified important areas within a text. Conversely, if the concept identification is not accurate, text summarization performance will degrade.

The FreqDist and modified version of the SumBasic summarizers were used to generate a summary of each of the 24 texts using the concept output from both MetaMap and CONANN. The system-generated summaries were evaluated using the ROUGE (Recall-Oriented Understudy for Gisting Evaluation) [20] summary evaluation tool. ROUGE provides automated comparison of the sytem-generated summaries with the three summaries produced by medical students. ROUGE is a standard evaluation tool in the text summarization community and uses n-gram co-occurrence to determine the overlap between a summary and the models (e.g., manually created summaries) [21]. An n-gram can be considered as one or more consecutive words. ROUGE scores indicate the n-gram overlap between the source text and the model summaries. ROUGE-2 measures bigram co-occurrence while ROUGE-SU4 measures skip-bigrams with a maximum distance of 4 words.

Table 1 shows the ROUGE-2 and ROUGE-SU4 scores using the FreqDist and SumBasic summarizers with both CONANN and MetaMap annotation output. The use of the CONANN concept annotator with both the FreqDist and SumBasic summarizers provides the best text summarization performance. CONANN with FreqDist shows an improvement of 7% for ROUGE-2 and 2% for ROUGE-SU4. CONANN with SumBasic shows an improvement of 7.8% for ROUGE-2 and 5.6% for ROUGE-SU4.

Table 1: Text Summarization Performance

Summarizer	*ROUGE-2 Score*	*ROUGE-SU4 Score*
FreqDist with MetaMap	0.12080	0.21864
FreqDist with CONANN	0.12897	0.22292
SumBasic with MetaMap	0.10920	0.19868
SumBasic with CONANN	0.11839	0.21053

Conclusion

We presented an extended version of our online biomedical concept annotator, CONANN, which takes a biomedical source phrase, identifies potential matching concepts and phrases in a domain-specific thesaurus, uses a coverage (word overlap) filter to remove unlikely candidate phrases, and maps the source phrase to best-matching concepts using a language modeling approach. An intrinsic evaluation was performed to compare the precision of CONANN's concept output and annotation speed to MetaMap, a state-of-the-art concept annotator. Precision was measured at 0.85 for exact match and 0.93 for relaxed match, with a speed improvement of 4.5 times. An extrinsic evaluation was also performed to measure the usefulness of the concept output in a text summarization task. It showed summaries generated from two different types of summarizers using concepts generated by CONANN slightly outperformed the same summarizers using MetaMap concept output.

The intrinsic and extrinsic evaluations show the use of filtering candidate phrases using word coverage and then mapping source phrases to biomedical

.

concepts using language models is effective and faster than current systems.

Future work includes integrating synonymous word variants into the concept language models and incorporating concept disambiguation. We would also like to consider including semantic relations into concept language models to improve the performance of the final concept mapper. Our goal is to provide a biomedical concept annotator operating at the phrase level which has high accuracy compared to existing systems, and which can operate in an online environment. Such a system would be useful for physician and biomedical research tasks such as personalized text summarization, question-answering, information extraction, data mining and concept-based indexing and retrieval.

References

[1] United States National Library of Medicine, "Unified Medical Language System (UMLS)," 2005.

[2] Center for Bioinformatics, United States National Cancer Institute. 2006, National cancer institute thesaurus.

[3] 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.

[4] 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.

[5] 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.

[6] L. Reeve, H. Han and A. D. Brooks, "BioChain: Using lexical chaining methods for biomedical text summarization," in *Proceedings of the 21st Annual ACM Symposium on Applied Computing, Bioinformatics Track,* 2006, pp. 180-184.

[7] L. H. Reeve and H. Han, "CONANN: An online biomedical concept annotator," in *Proceedings of the 2007 Data Integration in the Life Sciences Conference (DILS'07),* 2007,

[8] C. D. Manning, P. Raghavan and H. Schütze, *Introduction to Information Retrieval.* ,1st ed.Cambridge, England: Cambridge University Press, 2007.

[9] X. Liu and W. B. Croft, "Statistical language modeling for information retrieval," in *Annual Review of Information Science and Technology* , vol. 39, B. Cronin, Ed. American Society for Information Science and Technology, 2005, pp. 1-31.

[10] A. R. Aronson, "MetaMap Evaluation," pp. 1-12, 2001.

[11] H. O. Kiess, *Statistical Concepts for the Behavioral Sciences.* ,Third ed., vol. 1, Boston, MA: Allyn and Bacon, 2002, pp. 568.

[12] 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.

[13] 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.

[14] 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.

[15] 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.

[16] A. D. Brooks and I. Sulimanoff, "Evidence-based oncology project," in *Surgical Oncology Clinics of North America* , vol. 11, Anonymous 2002, pp. 3-10.

[17] 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.

[18] L. Reeve, H. Han, S. V. Nagori, J. Yang, T. Schwimmer and A. D. Brooks, "Concept frequency distribution in biomedical text summarization," in *Proceedings of the ACM Fifteenth Conference on Information and Knowledge Management (CIKM'06),* 2006, pp. 604-611.

[19] A. Nenkova and L. Vanderwende, "The impact of frequency on summarization," Microsoft Research, Redmond, Washington, Tech. Rep. MSR-TR-2005-101, 2005.

[20] C. Lin and E. H. Hovy, "Automatic evaluation of summaries using N-gram co-occurrence statistics," in *Proceedings of 2003 Language Technology Conference (HLT-NAACL 2003),* 2003, pp. 71-78.

[21] National Institute of Standards and Technology (NIST), "Document Understanding Conferences," vol. 2005, 2005.

.