University of Sheffield: Preliminary investigation of a dictionary-based approach to the Biocreative gene and protein identification task

Yikun Guo, Henk Harkema, Ian Roberts, Rob Gaizauskas, Mark Hepple

Department of Computer Science, University of Sheffield, UK
biomed@dcs.shef.ac.uk

1 Introduction

In this paper we describe the system with which we participated in task 1.A of the Biocreative challenge, the identification of protein and gene mentions in sentences drawn from MEDLINE abstracts. The term recognizer we applied to this task is designed to function within a more comprehensive term processing system, which itself will form a part of a larger biomedical information extraction system that is currently under development. The term recognizer is essentially dictionary-oriented: the first step of terminology processing is term look-up in a large scale terminology resource providing access to terminological information aggregated across multiple sources. The next step, which is only partially implemented to date, is terminology parsing in which a rule-based parser tries to build longer terms from shorter ones that have been identified by term look-up.

Even though the full term recognition system is incomplete at the moment, participation in the Biocreative challenge provided a good opportunity to explore various aspects of the performance of the dictionary-based term recognizer as it exists now. Since the decision to participate in the Biocreative challenge was made at a rather late date, there was not much time to tailor our system to the particular context of genes and proteins. Although our entry is considered "unofficial" because it was received late, the Biocreative task 1.A organizers were kind enough score our submitted results.

2 Termino

At the University of Sheffield, we are constructing a general framework for the extraction of information from biomedical text: AMBIT, a system for acquiring medical and biological information from text. Terminology processing in AMBIT begins with term look-up in Termino, a large scale terminology resource for biomedical language processing. Termino includes a relational database which is designed to store a large number of terms together with complex, heterogeneous information about these terms, including information of a morpho-syntactic nature, such as part of speech and morphological class; information of a semantic nature, such as quasi-logical form and links to concepts in ontologies; and provenance information, such as the sources of the information in the database. The design of the database also allows for links to connect synonyms and morphological and orthographic variants to one another and to connect abbreviations and acronyms to their full forms. The contents of Termino are imported from existing, outside knowledge sources, e.g., the HUGO Nomenclature database ([6]) and the UMLS Metathesaurus ([1]). Contents can also be induced from text corpora, e.g., MEDLINE citations, but the database used for the Biocreative task did not contain information obtained in this fashion.

To ensure fast term recognition with Termino's vast terminological database, the system comes equipped with a compiler for generating finite state machines from the strings in the terminological database. This set-up turns Termino into a general terminological resource which is not restricted to any single domain or application. The database can be loaded with terms from multiple domains and compilation can be restricted to particular subsets of strings in the database by selection based on their source, for example. In this way one can produce term recognizers that are tailored towards specific domains or specific applications within domains.

Termino provides uniform access to terminological information aggregated across many sources. The advantage of term recognition with Termino is that it provides immediate entry points into a variety of outside ontologies and other knowledge sources, making the information in these sources available to processing steps subsequent to term recognition. For example, using a recognizer compiled to include terms from the HUGO Nomenclature database and the OMIM database ([4]), Termino will return the HUGO and OMIM database identifiers for gene or protein names it recognizes in a text. These identifiers give access to the information stored in these databases about the gene or protein, including alternative names, gene map locus, related disorders, and references to relevant papers.

A first version of the terminological database has been implemented. It currently contains over 230,000 terms imported from various sources (see section 4 for more details). The compiler to construct finite state recognizers from the database is fully implemented, tested, and integrated into AMBIT. With this implementation we participated in task 1.A of the Biocreative challenge, the identification of gene and protein mentions in sentences drawn from MEDLINE abstracts. Participation in the Biocreative challenge provides a good opportunity to get an understanding of the performance of our system. We will use the evaluation results to direct future research activities to aspects of our system where further development is most likely to result in improved performance. By participating in the challenge we also want to express our support for the organizers' efforts to make available common standards and shared evaluation criteria for comparing different approaches to text mining in bioinformatics.

It should be emphasised that Termino has not been designed to be used as a stand-alone term recognition system but rather as the first component, the lexical look-up component, in a multi-component term processing system. Term look-up as performed by Termino is not the end point of term processing. Term look-up might return multiple possible terms for a given string, or for overlapping strings, and subsequent processes may apply to filter these alternatives down to the single option that seems most likely to be correct in the given context. Mechanisms to do this are currently not in place in our system. Furthermore, more flexible processes of term recognition might apply over the results of look-up. For example, a term *grammar* can be provided for a given domain, allowing longer terms to be built from shorter terms that have been identified by term look-up. Term grammars are a way of alleviating a well-known drawback of dictionary-based approaches to term recognition, namely their inability to deal with novel terms. For the Biocreative task we used a term grammar for protein names that was developed in a previous project ([2]). Due to time constraints, the system was not supplied with a grammar for gene names.

In the remainder of this paper we will describe the architecture of our system (section 3), discuss its performance on the test data (section 4), and then close the paper with some general conclusions and directions for future work (section 5).

3 System Overview

Termino's terminological database is organized as a set of relational tables, each storing one of the types of information mentioned in section 2. The fundamental element of the terminological database is a *termoid*. A termoid consists of a string together with associated information of various kinds about this string. A string can be in more than one termoid. Each termoid, however, pertains to one and only one string. Figure 1 provides a simple illustration of the structure of the terminological database. In the table STRINGS every unique string is assigned a string identifier (*str_id*). In the table TERMOID STRINGS each string identifier is associated with one or more termoid identifiers (*trm_id*). These termoid identifiers serve as keys into the tables holding terminological information. Thus, in the particular example given in figure 1, the database contains the information that in the UMLS Metathesaurus version 2003AC the string *albumin* has been assigned the concept-unique identifier C0001924 (CUI), the lemma-unique identifier L0001924 (LUI), and the string-unique identifier S1965773; in the HUGO database the string *albumin* has been assigned the identifier 399. The PASTA TYPES table indicates that in the term grammars the string *albumin* functions as a constituent of category *protein head*.

STRINGS

string	*str_id*
...	...
albumin	str371
...	...

TERMOID STRINGS

trm_id	*str_id*
...	...
trm023	str371
trm627	str371
trm874	str371
...	...

HUGO

trm_id	*hgnc_id*
...	...
trm023	399
...	...

UMLS

trm_id	*cui*	*lui*	*sui*	*version*
...	...	...	...	...
trm627	C0001924	L0001924	S1965773	2003AC
...	...	...	...	...

PASTA TYPES

trm_id	*major*	*minor*
...	...	...
trm874	protein	head
...	...	...

Figure 1: Structure of the terminological database

Compilation of a finite state recognizer for term recognition proceeds in the following way. First, each string in the terminological database is broken into tokens (see the description of the tokenizer below). Next, starting from a single initial state, a path through the machine is constructed, using the tokens – or their morphological roots, if such exist (see the description of the morphological analyzer below) – of the string to label transitions. New states are only created when necessary. The state reached on the final token of a string will be labeled final and is associated with the termoid identifiers of that string. The resulting finite state recognizer is used in the term matcher module described below.

The protein term grammar used by the system was created manually by splitting lists of terms into their component parts. These parts are then assigned a category, e.g., *protein_pre-modifier*, *protein_head*. The lists of parts and their categories are loaded into Termino. For each decomposition a rule is added to the grammar to recombine the parts, e.g., *protein* → *protein_pre-modifier protein_head* ([3]). The grammar rules also deal with spelling variation involving punctuation marks. For example, mirroring the rule given above, there is also a grammar rule joining a *protein_pre-modifier* and a *protein_head* with a connecting hyphen, viz., *protein* → *protein_pre-modifier - protein_head*. Some of the rules are sensitive to morphological suffixes, e.g., the grammar for protein names contains a rule stating that a token ending in the affix *-ase* is a protein.

The text to be term-tagged is processed by the first six modules of AMBIT's information extraction engine. This engine builds on the PASTA Information Extraction system described in [2]. The engine consists of various modules arranged in a pipeline, where the output of one module serves as the input of the next. Below we will briefly describe the six modules that are involved in term recognition in the order in which they occur in the pipeline.

Tokenizer The text is segmented into tokens, where a token is a contiguous sequence of alphabetic characters, a contiguous sequence of numeric characters, or a punctuation symbol. For example, the string *5-HT3 receptor* is broken into five tokens: *5*, *-*, *HT*, *3*, and *receptor*.

Sentence Splitter This module identifies sentence boundaries in text. Since sentence splitting occurs before term recognition, no term will straddle a sentence boundary.[1]

Morphological Analyzer The morphological analyzer tries to resolve each alphabetic token into a root and suffix. It identifies the English language suffixes *-s*, *-ed*, *-en*, and *-ing*, as well as a set of common biochemical suffixes such as *-ase*, *-yl*, *-ide*, etc. Root forms are only established for tokens containing one of the English suffixes.

Term Matcher The term matcher runs the finite state machine described above over the text, starting from the initial state at each token in the text. For alphabetic tokens with a root form, the labels of transitions in the machine are compared against this root form; in all other cases comparison is against the unanalyzed surface strings of tokens. Each sequence of tokens leading to a final state is annotated with the termoid identifiers associated with this state. Where appropriate the machine will produce multiple termoid identifiers for recognized terms. It will also recognize overlapping and embedded terms.

Term Annotator The termoid identifiers produced by the previous module are used to access the terminological database and retrieve the information contained in the termoids. This module can be parametrized for specific kinds of information to be retrieved from the database.

Terminology Parser The final step of terminological processing is terminology parsing: a rule-based parser tries to combine token sequences into larger terms according to a given set of term grammars. The term grammar rules are sensitive to the results of morphological analysis and term matching and annotation.

4 Results & Discussion

For task 1.A, we compiled a recognizer from Termino containing the following terms: 15,000+ human protein terms (and their assignments to the Gene Ontology) from the European Bioinformatics Institute (http://www.ebi.ac. uk/GOA/), and 36,000+ terms from several gazetteer lists containing terms in the field of molecular biology that were assembled for previous information extraction projects in our NLP group. The system was also supplied with a term grammar for proteins containing about 250 rules. There is no grammar for gene names. As mentioned in the introduction, our submission is considered "unofficial", but it was scored by the Biocreative task 1.A organizers. These "official" performance scores for our system configured as described above are given in the row labeled 'system A' in table 1.[2]

After we submitted our results for evaluation we realized that due to a bug in one of our scripts, many of the 'gene' tags in the submitted results were actually not produced by the AMBIT modules, but originated from the AbGene tagger ([5]) used to tag the text. Hence, the official scores are not an adequate reflection of AMBIT's term processing performance. We repaired this problem, ran the system again over the test data, and rescored the output. The new scores are given in table 1, row 'system B'.

We also discovered that we had erroneously added several thousand incomplete protein names to Termino and included these in the finite state recognizer. We deleted these terms from Termino, recompiled the recognizer, fixed a minor problem with the term grammars, and ran the system again. The scores for this run are given in in table 1, row

[1] The sentence splitter is not used for the Biocreative task, as the test data is already given as single sentences.

[2] Precision = TP / (TP + FP), Recall = TP / (TP + FN), F = (2 · Precision · Recall) / (Precision + Recall).

‘system C’. These scores will be our baseline for further experimentation. The recall score of system C is well below the average recall score of all the open systems that participated in task 1.A (0.71), and the precision score of system C is also below the average precision score (0.71).

	TP	FP	FN	Precision	Recall	F-Measure
system A	4066	3263	1883	0.55	0.68	0.61
system B	994	1322	4955	0.43	0.17	0.24
system C	1070	885	4879	0.55	0.18	0.27
system D	2709	2185	3240	0.55	0.46	0.50
system E	2918	3905	3031	0.43	0.49	0.46
system F	710	599	5239	0.54	0.12	0.19

Table 1: Performance figures

The majority of false positives generated by system C arises for one of the following two reasons. First of all, the notion of relevant term in our application does not entirely concur with the Biocreative guidelines. In particular, our system picks up terms that are considered too general by the Biocreative standard. e.g., *transcription factor* in contexts such as *Nuclear factor kappa B (NF-kappaB) is an important transcription factor for* .... Secondly, many false positives are in fact partial matches of valid terms, e.g., in the gene term *SNAG repressor motif* our system only marks up *repressor*. For cases in which all of its parts have been recognized individually, a missed compound term can be recognized by adding appropriate rules to one of the term grammars. Alternatively, the system could try to detect noun phrases in the text and tag an entire noun phrase in case only part of it has been recognized as a gene or protein.

Unsurprisingly, the large number of false negatives generated by system C are mostly terms that do not occur in any of the term sets included in the recognizer. One way of addressing this issue is to add further terms to the recognizer (see below). However, many of the false negatives are symbolic names such as *CrkL*, *Rep40*, *LIMK2*, and *Jun*. In general, these terms should be recognizable based on their orthographic make-up.[3]

Since the approach we took to task 1.A is essentially dictionary-based, performance depends very much on the sources from Termino that are selected to go into the recognizer. For system D, we took the recognizer from system C and added the 7000+ unique strings that are labeled as genes in the Biocreative training data for task 1.A. The scores of this system can be found in table 1, row ‘system D’. The performance of system D shows that a judicious choice of terms to be included in the recognizer leads to an increase in recall without a significant decrease in precision. It should be noted, however, that this option is not available for term recognition in domains for which there is no annotated training data.

For system E, we extended the recognizer of system D with 36,000 gene names and gene symbols from the generally available HUGO nomenclature database. As can be seen in table 1, recall went up but precision went down. The loss in precision is primarily due to the presence in HUGO of names and symbols such as *WAS*, *AS*, and *DO*, which match frequently occurring common English words. In order to take full advantage of the HUGO terms, recognition should be made sensitive to capitalization patterns, part of speech information, and the context of potential terms.

To test the effect of the term grammars, we ran system C without the protein grammar. The results are in table 1, in row ‘system F’. Using the term grammars improves recall by 50% while precision remains virtually unchanged. We conclude that the protein grammar provides a valuable contribution to the results of term processing.

5 Conclusion & Future Work

In this paper we have presented the results of applying a primarily dictionary-based term recognizer to the Biocreative task 1.A, the identification of gene and protein mentions in sentences drawn from MEDLINE abstracts. The dictionary-based term recognizer has been designed as a component of a more comprehensive term recognition system, which is itself intended to form part of a larger biomedical information extraction system. The full term recognition system is still under development, but we considered it useful to evaluate the dictionary component through participation in the Biocreative challenge, in order to understand the performance levels that could be achieved using only this look-up component.

As was to be expected, we confirmed that dictionary look-up alone is not adequate for the task of gene and protein identification, leading to both false positives and false negatives. False positives arise from: (1) mismatch between gene and protein terms found in standard resources and gene and protein terms as defined for the Biocreative task; (2) dictionary terms occuring as subterms of compound gene and protein terms without the full term being in the

[3] Post-hoc processing of the output of system C in which each word not already tagged as a gene and consisting of a sequence of alphabetic characters followed by a sequence of numeric characters or consisting of a sequence of alphabetic characters followed by a hyphen followed by a sequence of numeric characters such that the sequence of alphabetic characters does not match an entry in the lexicon of the Brill tagger, i.e., is not a ‘non-technical’ English word, was assigned a gene tag produced 1862 true positives, 1339 false positives, and 4087 false negatives – precision = 0.58, recall = 0.31.

dictionary; and (3) gene and protein terms found in dictionaries which are also common English words. False negatives occur because of lack of coverage of the terminological resources.

Aside from these insights into the inadequacies of a dictionary-based approach to the task, our investigations have thrown light on the contributions of various components of our system. In particular supplementing the dictionaries with all of the examples in the training data significantly improved recall without damaging precision, though the fundamental problem of lack of coverage remains. By contrast, the addition of a substantial number of gene names and gene symbols from HUGO improved recall but did reduce precision considerably. Finally, the use of limited grammar rules was demonstrated to have a positive effect on recall without any negative impact on precision.

Based in part on lessons learned from our participation in Biocreative we have identified the following areas for future work. First, lack of coverage, which is inevitable using any manually compiled dictionary, needs to be addressed by some form of rule or pattern-based approach which generalises away from specific terms. This may be done in a variety of ways. Multi-token terms may be analysable as the combinations of subterms from various categories, i.e. there may be a grammar to term construction. The protein grammar we developed for another task and deployed in Biocreative is one such example, capturing generalisations about the internal structure of terms that allow novel combinations of term elements to be recognised. We need to investigate writing grammars for genes and also consider inducing grammars from corpora or from compound terms in term sources such as the UMLS Metathesaurus. Single or multi-token terms may also be recognisable from contextual patterns, patterns which match contexts within which terms are predictably found. Finally, single token terms which are symbolic names or abbreviations (e.g. *CrkL*) should recognisable using a combination of orthographic information and character n-gram models. These are approaches to dealing with the lack of coverage of dictionaries. To deal with false positive (*WAS*, *DO*, etc.) attention needs to be paid to either filtering out such common words from dictionaries or case information needs to be taken into account, or context needs to be taken into account to ascertain whether a term reading is appropriate. We are confident that by taking into account all of these factors, as well as extending the dictionaries themselves, we will be able to build a more complete term recognition system whose performance is substantially better.

Acknowledgements

The authors would like to acknowledge the support of the UK EPSRC (Grant number: GR/R67743) and MRC (Grant number: 60086) which has made possible the work reported here.

References

[1] L. Humphreys, D.A.B. Lindberg, H.M. Schoolman, and G.O. Barnett. 1998. The Unified Medical Language System: An Informatics Research Collaboration. In: *Journal of the American Medical Informatics Association*, 1(5):1-13.

[2] R. Gaizauskas, G. Demetriou, P. Artymiuk, and P. Willett. 2003. Protein Structures and Information Extraction from Biological Texts: The PASTA System. In: *Journal of Bioinformatics*, 19(1): 135-143.

[3] R. Gaizauskas, G. Demetriou, and K. Humphreys. 2002. Term Recognition and Classification in Biological Science Journal articles. In: *Proceedings of the Computational Terminology for Medical and Biological Applications Workshop, 2nd International Conference on Natural Language Processing*, p. 37-44.

[4] Online Mendelian Inheritance in Man, OMIM (TM). 2000. McKusick-Nathans Institute for Genetic Medicine, Johns Hopkins University (Baltimore, MD) and National Center for Biotechnology Information, National Library of Medicine (Bethesda, MD). http://www.ncbi.nlm.nih.gov/omim/.

[5] L. Tanabe and W.J. Wilbur. 2002. Tagging Gene and Protein Names in Biomedical Text. In: *Journal of Bioinformatics*, 18(8): 1124-1132.

[6] H.M. Wain, M. Lush, F. Ducluzeau, and S. Povey. 2002. Genew: The Human Nomenclature Database In: *Nucleic Acids Research*, 30(1): 169-171. (http://www.gene.ucl.ac.uk/nomenclature/.)