City Research Online

City, University of London Institutional Repository

Citation: Gooch, P. & Roudsari, A. (2011). A tool for enhancing MetaMap performance when annotating clinical guideline documents with UMLS concepts. Paper presented at the IDAMAP Workshop at 13th Conference on Artificial Intelligence in Medicine (AIME'11), 02-07-2011 - 06-07-2011, Bled, Slovenia.

This is the unspecified version of the paper.

This version of the publication may differ from the final published version.

Permanent repository link: https://openaccess.city.ac.uk/id/eprint/1036/

Link to published version:

Copyright: City Research Online aims to make research outputs of City, University of London available to a wider audience. Copyright and Moral Rights remain with the author(s) and/or copyright holders. URLs from City Research Online may be freely distributed and linked to.

Reuse: Copies of full items can be used for personal research or study, educational, or not-for-profit purposes without prior permission or charge. Provided that the authors, title and full bibliographic details are credited, a hyperlink and/or URL is given for the original metadata page and the content is not changed in any way.

City Research Online: http://openaccess.city.ac.uk/ publications@city.ac.uk

A tool for enhancing MetaMap performance when annotating clinical guideline documents with UMLS concepts

Philip Gooch*[1] and Abdul Roudsari[1][2]

[1] Centre for Health Informatics, School of Informatics, City University, London, UK,
[2] School of Health Information Science, University of Victoria, BC, Canada

Abstract. We developed a tool that integrates the National Library of Medicine's MetaMap software with GATE, an open-source text analytics framework. The tool allows non-ASCII encoded documents of numerous formats to be annotated with UMLS concepts. We created a GATE pipeline to chunk cardiovascular disease guideline text into default segments (blank-line delimited), XML element content, sentences and phrases, which were sequentially submitted to MetaMap for annotation. XML element, sentence and phrase chunking allowed term extraction and mapping to be completed in around 1/3 of the time taken with default chunking, although with slight loss of accuracy (F1.0s=0.94-0.99). However, phrase chunking allows more complex input to be processed in real time, which is not possible with the other approaches. We discuss the results in relation to use of MetaMap's `--term_processing` option for generating pre- and post-coordinated mappings from composite phrases.

Keywords: biomedical language processing, information extraction, candidate term identification, terminology, concept mapping, informatics applications

1 Introduction

The use of natural language processing (NLP) techniques has recently been proposed as a method for generating tailored clinical decision support (CDS) from free text[1]. Precise identification of biomedical and clinical terms in the text is a crucial step in this process[2].

Term identification typically involves three stages[2]:

1. *recognize* the text string as a possible term (candidate term selection)
2. *classify* the candidate term (e.g. body part, disease, physiological funtion)
3. *map* the term to a single concept (*pre-coordination*) or to qualified, multiple concepts (*post-coordination*) within a standardised vocabulary or ontology.

* Corresponding author.

One of the key tasks in integrating a guideline-based CDS system with an electronic medical record (EMR) is to map clinical terms contained in both guidelines and patient notes to a common, controlled terminology[3][4]. This can facilitate the provision of point of care recommendations within an EMR, and allows encoded guidelines to be shared across institutions.

The Unified Medical Language System (UMLS) Metathesaurus from the National Library of Medicine (NLM)[5] comprises over 100 reference terminologies, such as HL7 v3, SNOMED-CT and LOINC, which have been adopted as international standards for patient data encoding, and form the basis of the information model adopted by at least one formalised guideline model[3].

The NLM's MetaMap software[6] is considered to be the reference tool for identifying biomedical terms in free text and mapping them to UMLS concepts[7]. However, MetaMap only handles unformatted, ASCII text as input, and processing complex phrases can require many hours of computation[6].

2 Background

Our goal in this study was to develop a tool that leveraged the domain coverage of MetaMap, while enabling documents in a variety of non-ASCII formats to be rapidly mined for biomedical terms and mapped to UMLS source vocabulary concepts. We also required that the tool be easily integrated with other NLP pipelines for other information extraction (IE) tasks, such as identifying linguistic patterns in clinical guidelines[8] to extract decision support rules.

In this paper we present an overview of the tool with examples and brief evaluation results of its performance, in terms of annotation accuracy and processing time, in comparison to the standalone MetaMap application, when extracting SNOMED-CT concepts from clinical guideline documents.

3 Methods

GATE[9] is an open-source text analytics framework for processing documents in variety of formats including PDF, RTF, and XML. Using the MetaMap Java API, we developed a plugin that integrates MetaMap with GATE, whereby text is normalized and chunked into blank-line delimited segments, which are submitted to MetaMap server (or multiple servers for parallel processing) and the results converted to GATE annotations and features for further processing.

As MetaMap works with ASCII data, it returns term positions as *byte* offsets relative to the start of the phrase. Annotation offsets in GATE are given as encoding-dependent *character* offsets. To accommodate this, UTF-8 and ISO-8859-1 data in the payload is normalized to its ASCII equivalent by using the `java.text.Normalizer` class[10] to 1) convert the string to its Roman equivalent plus diacritic, 2) stripping the diacritic and 3) converting the resulting string to an ASCII byte stream from which to create a new ASCII-encoded string that forms the input to MetaMap. This process allows terms containing diacritics (e.g. *Guillain-Barré*) to be correctly recognised and mapped by MetaMap.

The plugin provides a number of features designed to optimize the processing of large documents. For example, the document's data payload to MetaMap can be reduced to solely:

- the content of candidate term annotations (identified by an upstream process), or named elements in the original XML markup;
- the first instance of each candidate term, with remaining instances co-referenced against the mapped term;
- named attributes of each candidate term, rather than the original source data. For example, an upstream process might normalize prepositional phrases such as 'cancer of the lung' → 'lung cancer', or verb phrases such as 'pain is severe' → 'severe pain', and store the result as an attribute to the term. This allows these grammatical variants to be treated as equivalent for the purposes of coreferencing.

In this study, we treated all noun (NP), prepositional (PP) and verb phrases (VP) as candidate terms. We used GATE's Sentence Splitter to identify sentences, and combined GATE's Tokenizer and Part of Speech Tagger with Java Annotation Patterns Engine (JAPE) rules to identify NP, PP and VP chunks.

For example, a simple noun phrase can be identified using the following rule:

```
NP = (DT)? (VB)? (ADJ)* (NN)+                    (1)
```

where `DT` represents a determiner ('a', 'the', 'those', 'no'); `VB` a verb; `ADJ` an adjective; `NN` a noun. For example, 'raised$_{VB}$ blood$_{NN}$ pressure$_{NN}$'.

A prepositional phrase can be identified with:

```
PP = (NP) (IN) ((NP) (IN))* (NP)                 (2)
```

where `IN` represents a preposition. For example, 'absence$_{NP}$ of$_{IN}$ pulse$_{NP}$', 'no evidence$_{NP}$ of$_{IN}$ cardiovascular disease$_{NP}$'.

A verb phrase can be identified with:

```
VP = (NP | PP | ADJ+) VG+ (NP | PP | ADJ+)?      (3)
```

where `VG` represents a verb or verb group ('is', 'should be', 'may provide').

We configured the plugin to identify only SNOMED-CT terms, and examined its performance when annotating the content of the 106kB, 11,000 word recommendations section of the *ESC European Guidelines on Cardiovascular Disease Prevention in Clinical Practice* in UTF-8 XML format for different input chunking options (processed in serial): default (blank line delimited seqments), individual XML elements (`p`, `li`, `h1`, `h2`), sentences and NP, PP and VP chunks.

We ran the tests both without and with MetaMap's `--term_processing` option, which treats the input as a single phrase for direct lookup into the Metathesaurus, and provides a mechanism for mapping composite phrases to a single identifier in UMLS.

GATE's Corpus Quality Assurance tool was used to calculate recall, precision and strict *F*-measure scores, where *F*-measure = 2 * precision * recall / (precision + recall).

4 Results

Table 1 shows the time taken to annotate the guideline following default chunking compared with annotation of individual XML elements, sentences and phrases. Table 2 shows recall and precision scores for MetaMap annotations produced from the different lexical units both with and without term processing. The output of the default-chunked input was used as the reference standard.

Table 1. Processing times for different lexical units

	Default	Element	Sentence	Phrase
Time (s) w/o term processing	745	208	213	268
Time (s) w/term processing	$> 10^{5\dagger}$	$> 10^{5\dagger}$	$> 10^{5\dagger}$	325

† Process aborted after 3 hours with no output

Table 2. Recall/precision for Element, Sentence, Phrase (B) vs default chunking (A)

Annotation	Match	Only A	Only B	Overlap	Rec.	Prec.	F1.0s
Element (w/o term processing)	4122	349	154	9	0.92	0.96	0.94
Sentence (w/o term processing)	4393	27	13	9	0.99	1.00	0.99
Phrase (w/o term processing)	4168	128	53	184	0.93	0.95	0.94
Phrase (w/ term processing)	3889	224	118	367	0.87	0.89	0.88

5 Discussion

Inspection of the output suggests that the slight reduction in annotation accuracy resulting from processing by phrase arises from lone verbs not being picked up by our phrase chunker. While identification of relevant verbs is required for extracting 'action' information from clinical guidelines[8], we do not require them to be mapped to UMLS concepts, so perhaps this slight reduction is acceptable. Overall, processing by sentence provided the best trade-off between speed and accuracy.

However, phrase chunking allows rapid term extraction and mapping from input without syntactical structure, such as bulleted lists, which can require many hours of computation if processed directly by MetaMap[6]. Also, as shown in Table 1, by-phrase processing allows the `--term_processing` option to be used on entire documents. This is useful as it allows composite phrases in the guideline to be mapped to their pre-coordinated terms in UMLS, rather than to multiple terms requiring post-coordination. Some examples are shown in Table 3.

In a future paper we extend these techniques to the processing of EMR data, as a first step in automating the analysis of guideline compliance, by comparing SNOMED-CT encoded information from discharge summaries with that extracted from the relevant guidelines.

Table 3. Phrase chunking mappings both without and with (bold) term processing

Phrase	Mappings
management of risk factors	Administration(C0001554); History of- risk factor(C0455624)
	Risk management(C0035649)
foot pulses	Entire foot(C1281587); Pulse(C0391850)
	Pedal pulse(C0232157)
increases in blood pressure	Increase(C0442805); Blood pressure finding(C1271104)
	Elevated blood pressure(C0497247)

6 Conclusion

We have presented a tool and method for rapid extraction of UMLS concepts from clinical guidelines. It utilizes MetaMap's domain coverage strengths and removes some performance bottlenecks. The tool is now available as part of the standard GATE distribution, and can be easily integrated into other NLP pipelines within GATE, allowing it to be customised for more specific tasks.

References

1. Demner-Fushman, D., Chapman, W.W., McDonald, C.J.: What can natural language processing do for clinical decision support? J Biomed Inf 42(5), 760–72 (2009)
2. Krauthammer, M., Nenadic, G.: Term identification in the biomedical literature. J Biomed Inf 37, 512–526 (2004)
3. Tu, S.W., Campbell, J.R. , Glasgow, J., Nyman, M.A. et al.: The SAGE Guideline Model: Achievements and Overview. Journal of the American Medical Informatics Association 14(5), 589–598 (2007)
4. Peleg, M., Keren, S., Denekamp, Y.: Mapping computerized clinical guidelines to electronic medical records: Knowledge-data ontological mapper (KDOM). Journal of Biomedical Informatics 41, 180–201 (2008)
5. National Library of Medicine: UMLS Reference Manual, Bethesda, MD, National Library of Medicine (2009)
6. Aronson, A.R., Lang, F.-M.: An overview of MetaMap: historical perspective and recent advances. J Am Med Inform Assoc 17, 229–236 (2010)
7. Shah, N.H. Bhatia, N. Jonquet, C. Rubin, D. et al.: Comparison of concept recognizers for building the Open Biomedical Annotator. BMC Bioinformatics 10(Suppl 9), S14 (2009)
8. Serban, R., et al.: Extraction and use of linguistic patterns for modelling medical guidelines. Artificial Intelligence in Medicine 39(2), 137–149 (2007)
9. Cunningham, H., Maynard, D., Bontcheva, K., Tablan, V.: GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications. In: Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics (ACL'02), Philadelphia (2002)
10. Sun Microsystems: Normalizer (Java Platform SE 6). `http://java.sun.com/javase/6/docs/api/java/text/Normalizer.html`