Effective Mapping of Biomedical Text to the UMLS
Metathesaurus: The MetaMap Program

Alan R. Aronson, PhD
alan@nlm.nih.gov

National Library of Medicine
Bethesda, MD 20894

**INTRODUCTION**

Many researchers have developed programs to map free text to a biomedical knowledge source including NLM’s MeSH® vocabulary and, more recently, the UMLS Metathesaurus. Examples of such efforts include MicroMeSH [1], CHARTLINE [2], CLARIT [3], SAPHIRE [4, 5], Metaphrase [6] and a recent system developed by Nadkarni et al. [7]. These efforts have been applied to a wide array of applications and have achieved varying degrees of success depending on how well they solve such NLP problems as parsing, lexical variation and ambiguity resolution. The MetaMap approach [8-11] to mapping text is distinguished by its linguistic rigor and reliance on knowledge sources such as the SPECIALIST™ lexicon [12]. We describe the algorithm used by MetaMap, enumerate some of the applications in which MetaMap is being used and discuss current efforts to improve its accuracy.

**METHODS AND IMPLEMENTATION**

**The MetaMap Algorithm**

MetaMap is a highly configurable program that maps biomedical text to concepts in the UMLS Metathesaurus. (Examples cited here use the 2000 edition of the UMLS Knowledge Sources [12].) Options control MetaMap’s output as well as internal behavior such as how aggressive to be in generating word variants, whether or not to ignore Metathesaurus strings containing very common words, and whether to respect or to ignore word order. The description of MetaMap’s algorithm described here is necessarily brief; details can be found in several technical reports at the web address [http://nls3.nlm.nih.gov](http://nls3.nlm.nih.gov).

1. **Parsing**

   Arbitrary text is parsed into (mainly) simple noun phrases; this limits the scope of further processing and thereby makes the mapping effort more tractable. Parsing is performed using the SPECIALIST minimal commitment parser [13] which produces a shallow syntactic analysis of the text. The parser uses the Xerox part-of-speech tagger [14] which assigns syntactic tags (e.g., noun, verb) to words not having a unique tag in the SPECIALIST lexicon.

   Consider the text fragment *ocular complications of myasthenia gravis*. The parser detects two noun phrases: *ocular complications* and *of myasthenia gravis*. A simplified syntactic analysis for *ocular complications* is 

   \[[\text{mod}(ocular), \text{head}(complications)]\]

   Note that the parser indicates that *complications* is the most central part, the *head*, of the phrase. Words with tags such as prepositions, conjunctions and determiners are normally ignored in subsequent processing.

2. **Variant Generation**

   For each phrase, variants are generated using the knowledge in the SPECIALIST lexicon and a supplementary database of synonyms. A variant consists of a phrase word (called a *generator*) together with all its acronyms, abbreviations, synonyms, derivational variants, meaningful combinations of...
these, and finally inflectional and spelling variants [11]. This process, before computation of inflections and spelling variants, is shown pictorially in Figure 1. For efficiency, the generation of inflections and spelling variants is deferred until the variants shown in the figure are computed. The variants of the generator *ocular* are shown in Figure 2. They are arranged hierarchically according to the history of how they were created. Each variant is followed by its part of speech, its distance score from its generator and its history. For example, *ocular* (an adjective) has distance score 0 and empty history because it is a generator, itself.

Similarly, the noun *ophthalmia* has distance score 7 and history “ssd” meaning that it is a derivational variant of a synonym (*ophthalmic*) of a synonym (*eye*) of *ocular*.

3. Candidate Retrieval

The *candidate set* of all Metathesaurus strings containing at least one of the variants is retrieved. This retrieval is controlled by various options including `stop_large_n` which precludes searching for candidates containing either single-character variants with more than 2,000 occurrences in the Metathesaurus and two-character variants with more than 1,000 occurrences. In addition candidate retrieval is made more efficient through the use of special, small indexes whenever possible.

4. Candidate Evaluation

Each Metathesaurus candidate is evaluated against the input text by first computing a mapping from the phrase words to the candidate’s words and then calculating the strength of the mapping using a linguistically principled evaluation function consisting of a weighted average of four metrics: *centrality* (involvement of the head), *variation* (an average of inverse distance scores), *coverage* and *cohesiveness*. The latter two components measure how much of a candidate matches the text and in how many pieces. The candidates are then ordered according to mapping strength.

The nine candidates for the phrase *ocular complications* are shown in Figure 3. If the candidate is not the preferred name of the concept it represents, the preferred name is displayed in parentheses. Note that all of the candidates corresponding to the text *complications* score better than those for *ocular* because they involve the head of the phrase.

---

1. History codes are i (inflection), p (spelling variant), a (acronym/abbreviation), e (expansion of acronym/abbreviation), s (synonym) and d (derivational variant).

2. Distance scores are 0 for spelling variants, 1 for inflections, 2 for synonyms or acronyms/abbreviations and their expansions, and 3 for derivational variants.
5. Mapping Construction

Complete mappings are constructed by combining candidates involved in disjoint parts of the phrase, and the strength of the complete mappings is computed just as for candidate mappings. The highest-scoring complete mappings represent MetaMap’s best interpretation of the original phrase. The highest ranked mappings for the phrase *ocular complications* consist of the Metathesaurus concept ‘Ocular’ and either the concept ‘Complications Specific to Antepartum or Postpartum’. The mappings for *complications* illustrate MetaMap’s most difficult problem: ambiguity. Both concepts have ‘complications’ as one of their strings (ignoring case) and thus cannot be distinguished by MetaMap. This problem is partially addressed in the next section.

Further examples of mappings:
- The text *inferior vena caval stent filter* maps to concepts ‘Vena Cava Filters’ (which has string ‘Inferior Vena Cava Filter’) and ‘Stents’. This is a complete mapping resulting from two partial mappings.
- When using the option `allow_overmatches`, *medicine* maps to any of ‘Alternative Medicine’, ‘Medical Records’, and ‘Nuclear medicine procedure, NOS’. These mappings are *overmatches* because there are words at one or both ends of the Metathesaurus string which do not participate in the match.
- When using the option `composite_phrases`, *pain on the left side of the chest* maps to ‘Left sided chest pain’. Here, a *composite phrase* is a sequence of phrases all but the first of which are prepositional phrases; in addition all but the first prepositional phrases must be of phrases.

### Data Maintenance

MetaMap’s data files must be updated following each release of the UMLS Knowledge Sources. These include tables of precomputed variants, semantic type and MeSH treecode information, and Metathesaurus strings indexed by the words they contain (i.e., word index data). The files requiring the most effort to create are the word index files. The Metathesaurus files (esp. MRCON) are filtered in four ways:

1. **Manual filtering**
   A small number of Metathesaurus strings are problematic and have been manually suppressed before performing other forms of filtering. These include numbers, single alphabetic characters, special cases such as ‘Periods’ for ‘Menstruation’, and ambiguities. The most numerous problems here are the ambiguities; and fortunately the creators of the Metathesaurus have instituted the notion of *suppressible synonyms*, strings which do not express themselves completely or which are abbreviatory or informal. Strings marked as suppressible account for most of the problematic ambiguity in the Metathesaurus. The example above of ‘complications’ for ‘Complications Specific to Antepartum or Postpartum’ is a case in which ‘complications’ is not marked as suppressible but it will most likely be so in the future.

2. **Lexical filtering**
   Lexical filtering is the most benign type of filtering and consists of removing strings for a concept which are effectively the same as another string for the concept. Properties which can make strings effectively the same are:
   - non-essential parentheticals;
   - Metathesaurus multiple meaning designators;
   - NEC/NOS variation;
   - syntactic uninversion (i.e., reordering of strings containing commas unless the string appears to be a list as determined by the presence of a conjunction or preposition);
   - case variation;
   - hyphen variation; and
   - possessives.

   Lexical filtering is accomplished by normalizing all strings for a given concept according to the above criteria and removing all but one string for each set of strings that normalize to the same thing.

3. **Filtering by type**
   In addition to filtering out suppressible synonyms, terms are excluded based on their Term Type (TTY). The excluded types are generally abbreviatory, obsolete or have some kind of internal structure such as laboratory test descriptions in LOINC, one of the constituent Metathesaurus vocabularies.

4. **Syntactic filtering**
   The final kind of filtering is based on applying the parser to the Metathesaurus strings, themselves. Since normal MetaMap processing involves mapping the simple noun phrases found in text, it is highly unlikely that a complex Metathesaurus string will be part of a good mapping. Thus strings consisting of more than one simple phrase are filtered out. Because of their tractability, composite phrases (the ones containing well-behaved prepositional phrases) are exempted from this filtering.

Because MetaMap is used both for highly focused, semantic processing as well as browsing, three data models differing in the degree of filtration are created.

- **Strict Model**: All forms of filtering are applied. This view is most appropriate for semantic processing where the highest level of accuracy is needed. The
Strict Model consists of 706,593 (53%) of the 1,339,497 English Metathesaurus strings; 
• Moderate Model: Manual, lexical and type-based 
  filtering, but not syntactic filtering, are used. This 
  view is appropriate for term processing where 
  input text should not be divided into simple 
  phrases but considered as a whole. The Moderate 
  Model consists of 982,447 (73%) English Meta-
  thesaurus strings; and 
• Relaxed Model: Only manual and lexical filtering 
  are performed. This provides access to virtually all 
  Metathesaurus strings and is appropriate for 
  browsing. The Relaxed Model consists of 
  1,146,962 (86%) English Metathesaurus strings.

Availability
MetaMap is available on the Web for research pur-
who has signed the UMLS license agreement. Both 
interative and batch processing are supported. 
Throughput in batch mode is approximately 1,800 
MEDLINE® citations (over 3MB of text) per hour 
using up to seventeen Sun workstations in parallel.

A Java-based implementation allowing researchers to 
maintain and modify their own copy of MetaMap will 
be available by Summer 2001.

APPLICATIONS
MetaMap was originally developed to improve 
retrieval of bibliographic material such as MEDLINE 
citations. We have explored basic methodologies for 
low-level indexing as well as query expansion and 
have used the statistical IR systems SMART [15] and 
INQUERY [16] to test our methods. We have 
achieved a modest 4% improvement in average preci-
sion using an indexing scheme [8] and a significant 
14% improvement using query expansion [17]. These 
 latter results are comparable to those obtained by 
Srinivasan [18, 19].

MetaMap has also been applied to the following 
efforts:
• a hierarchical indexing project designed in part to 
determine how much of a document is relevant to a 
user’s query [20];
• several data mining efforts in which MEDLINE 
citations or clinical reports are examined to detect 
  - clinical findings [21];
  - molecular binding expressions [22];
  - drugs, genes and relationships between them 
    [23];
  - anatomical terminology [24]; and 
  - arterial branching expressions [25];
• another data mining effort which discovers novel 
  relationships between drugs and diseases in the 
  biomedical literature [26];
• a project which attempts to improve bibliographic 
  retrieval by categorizing users’ queries [27]; and 
• the NLM Indexing Initiative which has developed a 
  system to produce recommended indexing terms 
  for both semi-automatic and fully automatic 
  indexing environments [28].

DISCUSSION
Research has shown that MetaMap is an effective tool 
for discovering Metathesaurus concepts in text. But 
there are two areas in which MetaMap’s performance 
requires improvement: first, detection of idiosyncratic 
text such as chemical names, acronyms and abbrevia-
tions, numeric quantities or similar constructs; and 
second, resolution of ambiguity. The first problem is 
being solved through the use of an extensible, hierar-
chical tokenization regime. The initial implementa-
tion of this regime includes detection of acronyms/
abbreviations and chemical names, the latter based on 
work by Wilbur et al. [29]. Future plans include detec-
tion of numeric quantities and bibliographic refer-
ences. The problem of ambiguity is being investigated 
by developing a word sense disambiguation (WSD) 
test collection for evaluating methods including one 
developed by Humphrey [30, 31] which classifies text 
into a small number of categories, e.g., semantic 
types. Correct classifications can distinguish between 
competing concepts with different semantic types and 
thereby might resolve ambiguities for MetaMap.

Acknowledgements
The author wishes to extend heartfelt thanks to the 
following people whose counsel and efforts have been 
essential in the development of MetaMap: Tom Rind-
flesch, Allen Browne, Guy Divita, Susanne Hum-
phrey, Henny Klein, Alexa McCray, Jim Mork and 
Marc Weeber.

References
1. Elkin PL, Cimino JJ, Lowe HJ, Aronow DB, Payne 
  TH, Pincetl PS and Barnett GO. Mapping to MeSH: 
  The art of trapping MeSH equivalence from within 
2. Miller RA, Gieszczyniewicz FM, Vries JK and 
  Cooper GF. CHARTLINE: Providing bibliographic 
  references relevant to patient charts using the UMLS 
  Metathesaurus knowledge sources. Proc 16th 
3. Evans DA, Ginther-Webster K, Hart M, Lefferts 
  RG and Monarch IA. Automatic indexing using selec-