Available online at www.sciencedirect.com

SCIENCE DIRECT®

Journal of Biomedical Informatics 37 (2004) 512–526

Journal of Biomedical Informatics

www.elsevier.com/locate/yjbin

Methodological Review

Term identification in the biomedical literature

Michael Krauthammer[a,*], Goran Nenadic[b,c]

[a] Department of Biomedical Informatics, Columbia Genome Center, Columbia University, New York, USA
[b] Department of Computation, UMIST, Manchester, UK
[c] National Centre for Text Mining, Manchester, UK

Received 22 July 2004
Available online 8 October 2004

Abstract

Sophisticated information technologies are needed for effective data acquisition and integration from a growing body of the biomedical literature. Successful term identification is key to getting access to the stored literature information, as it is the terms (and their relationships) that convey knowledge across scientific articles. Due to the complexities of a dynamically changing biomedical terminology, term identification has been recognized as the current bottleneck in text mining, and—as a consequence—has become an important research topic both in natural language processing and biomedical communities. This article overviews state-of-the-art approaches in term identification. The process of identifying terms is analysed through three steps: term recognition, term classification, and term mapping. For each step, main approaches and general trends, along with the major problems, are discussed. By assessing previous work in context of the overall term identification process, the review also tries to delineate needs for future work in the field.

Keywords: Term identification; Term recognition; Term classification; Term mapping; Acronym recognition; Biomedical literature

1. Introduction

The current growth of biomedical knowledge has spurred interest in natural language processing (NLP) and information technologies such as information retrieval (IR) and information extraction (IE), which are helpful to cope with an increasingly large body of biomedical articles. These applications depend on term identification as the single most crucial step for accessing information stored in literature. Terms (such as names of genes, proteins, gene products, organisms, drugs, chemical compounds, etc.) are the means of scientific communication as they are used to identify domain concepts: there is no possibility to understand an article without precise identification of terms that are used to communicate the knowledge. A term corresponds to an author's textual representation of a particular concept, and the goal of term identification is to recognize the term and capture its underlying meaning. Automating this process enables the large-scale processing of the biomedical literature by identifying terms across authors and scientific documents.

The identification of terminology in the biomedical literature is one of the most challenging research topics in the last few years both in NLP and biomedical communities. Despite the availability of numerous manually corrected and curated terminological resources, several reports claimed that many term occurrences would not be identified in text if straightforward dictionary/database look-up was used [1–3]. Barriers to successful term identification include extensive lexical variations, which prevent some terms from being recognized in free text,

* Corresponding author. Present address: Department of Pathology, Yale University School of Medicine, P.O. Box 208023, 310 Cedar Street, New Haven, CT 06520-8023, USA.
E-mail address: michael.krauthammer@yale.edu (M. Krauthammer).

1532-0464/$ - see front matter © 2004 Published by Elsevier Inc.
doi:10.1016/j.jbi.2004.08.004

term synonymy (when a concept is represented with several terms), and term homonymy (when a term has several meanings), which create uncertainties regarding the exact term identity. Further, maintenance of terminological resources is complicated by a constantly changing terminology. Some terms typically appear in a very short time period, and some of them do not last for long. New terms are introduced in the domain vocabulary on a daily basis, and—given the number of names introduced around the world—it is practically impossible to have up-to-date terminologies that are produced and curated manually. A related problem is the lack of firm naming conventions. Guidelines do exist for some types of biomedical entities, but they do not impose restrictions to domain experts who are still by no means obliged to use them when coining a new term. Consequently, along with "well-formed" terms, ad hoc names exist, which are problematic for automatic term identification systems. For example, there is a gene name "*bride of sevenless*" (FlyBase [4] ID FBgn0000206) with its acronym "*boss*", as well as a protein that has been named after a Chinese breakfast noodle "*yotiao*" (Swiss-Prot [5] ID Q99996) [6]. Even if biologists start to use exclusively "well-formed" and approved names, there are still a huge number of documents containing "legacy" and ad hoc terms.

Therefore, dynamic approaches are needed to locate and identify terms in documents. Much of the work has been devoted to automatic term recognition (ATR), which is concerned with the tagging of textual units that are related to domain-specific concepts. While covering ATR in great detail, this review also tries to put ATR in context of the overall task of term identification, which goes beyond term recognition to include term classification and term mapping, which are concerned with finding appropriate term categories and links to referent data sources, respectively.

2. Term identification task

We differentiate three main steps for the successful identification of terms from literature: term recognition, term classification, and term mapping (see Fig. 1).

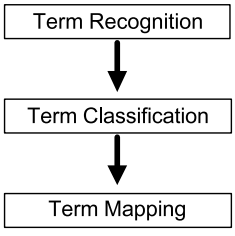

Fig. 1. Term identification consists of three steps: term recognition, term classification, and term mapping.

Term recognition is a non-trivial task of marking single or several adjacent words that indicate the presence of domain concepts. Its main goal is to differentiate between terms and non-terms. As term recognition does not further narrow down the specific meaning of a concept, it is often combined with *term classification* (or term categorization), which assigns terms to broad biomedical classes, such as genes, proteins or mRNAs. Categorized terms are useful for applications that work with specific term classes, such as systems that extract information on protein–protein interactions. Also, term classification is important for ontology management, where terms representing novel concepts are automatically mapped to specific parts of the ontology. While classification helps to establish some broad notion of the nature of a biomedical concept, it is not sufficient for establishing term identity. This is done by *term mapping*, which links terms to well-defined concepts of referent data sources, such as controlled vocabularies or databases. The linking definitely establishes the exact term identity (with respect to the referent data source). Mapped terms are annotated with referent identifiers (IDs) that act as keys to supplementary information such as preferred and synonymous terms, or sequence information. The mapping of terms is essential in any data integration efforts where acquired knowledge on specific biomedical concepts is aggregated across different data sources.

To give an example of the term identification steps, consider a hypothetical sentence such as '*p53 protein suppresses mdm2 expression*' in an article on human signal transduction. We use term recognition to find the term boundaries for the two entities of interest (*p53 protein* and *mdm2*). Then, we categorize the first entity (*p53 protein*) as a protein, while the second entity, *mdm2*, which does not convey any explicit class information, is classified as a gene. Finally, we map the terms to reference databases. In the example above, *mdm2* could be assigned to a reference gene database, such as LocusLink [7], and given a specific database ID (LocusID 4193 for *Homo sapiens*), while *p53 protein* could be linked to a protein repository such as Swiss-Prot (Swiss-Prot ID P04637 for *Homo sapiens*). Of the many challenges in identifying *mdm2* as the LocusID 4193 entity, consider the need for contextual clues to classify it as a gene (as opposed to a protein or other molecular class), and that mapping is complicated by several LocusLink entries for *mdm2* (for different species).

Note that each of the three steps of the identification process can be considered a classification problem. Term recognition is a general binary classification that arranges lexical units from free text into two groups: terms and non-terms. Classification further groups them into broad semantic classes, while mapping attempts to determine the exact "knowledge space" that is assigned to a given term by a fine-grained classification.

So far, we have discussed the major steps in term identification. It is worthwhile to study additional components and underlying resources that are part of the identification process. As can be seen in Fig. 2, term identification is linked to lexical resources and dictionaries, which are compiled from referent databases, such as LocusLink, FlyBase, or SwissProt. They assist the term identification process at different levels: dictionaries are directly applicable for detecting names in texts, while specifically designed lexical resources, such as lists of functional words, are useful for term classification (these resources are optional for methods that work with dictionary-independent surface clues). A *normalization component* interfaces between the dictionaries and the term identification steps. It serves different purposes, such as taking care of lexical variations in dictionary-based term recognition, or selecting a preferred term for term mapping. We will be reviewing different normalization strategies in context of term mapping, which heavily depends on the normalization of term variants (see Section 2.3). There is an additional component (not shown in Fig. 2) that is often associated with term identification: the recognition of acronyms. Acronyms are very common, with many authors defining ad hoc abbreviations for biomedical concepts. The understanding of acronyms is facilitated through automated compilation of acronym dictionaries, which link acronyms to their expanded forms. We will be discussing *acronym recognition* (and the construction of acronym dictionaries) under the topic of term recognition (Section 2.1).

Although methodologically and conceptually clear, the term identification process does not necessarily comply with the sequential order of the steps as depicted in Figs. 1 and 2. Some of the steps can be merged, as in the traditional named entity (NE) recognition task,[1] where term recognition and classification are performed together. Also, if term recognition is based on dictionary/database look-up, then the corresponding term IDs (and, consequently, the term mapping) can be obtained directly from the matching entries (in cases when there is no ambiguity, see Section 2.3.2). Similarly, there are classification algorithms that effectively map terms to specific dictionary entries, blurring the distinction between classification and mapping. We will nevertheless be using this schematic process flow to group and discuss the tremendous amount of published work on term identification. Therefore, the review will be featuring a separate section for term recognition, term classification, and term mapping. We aim at giving a comprehensive overview of general trends, main approaches,[2] and major problems for each of the steps, while giving the reader a chance to understand a specific methodology in the larger context of term identification.

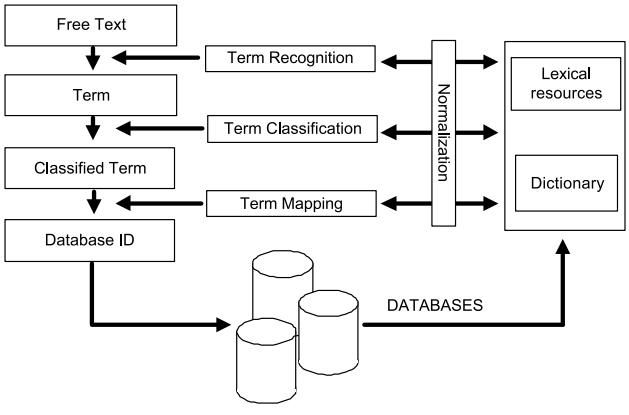

Fig. 2. From text to database IDs: term recognition and classification are essential steps to take before mapping terms to database IDs. Term normalization is important for recognizing variant terms at various stages in the term identification process. Dictionaries and lexical resources are compiled from diverse databases and can be used for tasks such as term recognition or mapping.

2.1. Term recognition

Term recognition denotes a set of procedures that are used to systematically recognize pertinent terms in literature, i.e., to "highlight" lexical units that are related to relevant domain concepts. The performance of ATR systems is typically assessed in terms of precision and recall. *Precision* measures the correctness of the lexical units that are suggested as terms, and is usually measured as the ratio of correct ("true positives") and all suggested units (both "true positives" and "false positives").[3] *Recall* denotes the degree to which concepts in a document are recognized, and is usually measured by the ratio of the correctly recognized terms ("true positives") and all domain-relevant terms occurring in a given document ("true positives" and "false negatives"). Although ATR systems naturally aim at high precision and high recall, there is a trade-off between the two measures: high precision can be typically achieved at lower recall points, and vice versa. The overall performance is typically measured by a single score (called the *F-measure*), which is defined as the harmonic mean of the precision and recall values:

[1] The NE recognition task has been defined within the Message Understanding Conferences (MUCs). The role of NEs and other MUC tasks in biomedical text processing has been discussed by Hirschman and colleagues [2].

[2] In many cases we will provide evaluation of methods as reported by respective authors. However, the corresponding testing sets and evaluation strategies are typically different. A direct comparison of the performance of different methods is therefore problematic.

[3] "True positives" refers to lexical units that are correctly recognized as terms, while "false positives" denote non-term units that are wrongly suggested as terms. Terms that are not recognized are usually referred to as "false negatives."

$$F\text{-}measure = 2 * \frac{precision * recall}{precision + recall}.$$

Since the vast majority of terms are noun phrases (NPs), the main strategy in many ATR systems is to extract specific NPs (typically referred to as *term candidates*) and then to estimate their "termhoods," i.e., likelihood of representing domain-specific concepts. Further, many ATR systems consider multi-word NPs, as the majority of biomedical terms contain several words (e.g., almost 90% biomedical terms in the GENIA[4] corpus are compounds [9]).

In the following sections we will be discussing different approaches to ATR, starting with dictionary-based recognition of biomedical terms. We then examine rule-based (or knowledge engineering) systems that mainly use term internal evidence in order to locate potential terms. We also consider statistical and machine-learning methods that chiefly rely on external evidence presented through surrounding (contextual) information. We further look at hybrid approaches that combine different methods and use a mixture of complementary resources.

As was pointed out in [2], the majority of ATR approaches in the biomedical domain target specific entities (mainly gene and protein names), thus integrating term recognition and term classification. The main reason for performing both tasks in parallel is that it is more difficult to identify features that apply to terms "in general" than features that are specific to individual term classes. Thus, the majority of ATR approaches reviewed here perform both term recognition and term classification. However, we will also mention general ATR approaches that work without semantic knowledge of the domain and that are focused on the term recognition only.

2.1.1. Dictionary-based approaches

Dictionary-based methods for ATR use existing terminological resources in order to locate term occurrences in text. However, as indicated earlier, it has been claimed that many term occurrences could not be recognized in text if straightforward dictionary/database look-up is used [1–3]. Hirschman and associates [2] presented the problems encountered in an experiment with a simple pattern matching used to locate gene references using an extensive list of gene names from FlyBase. They reported on an extremely low precision rate (2% for full articles and 7% for abstracts) with recall in the range 31% (for abstracts) to 84% (for full articles).[5] The main reason for such poor precision was homonymy, as many gene names shared their lexical representation with common English words (e.g., gene names/abbreviations such as *an*, *by*, *can*, and *for*). Even additional filtering and discarding shorter names (which are typically more ambiguous than longer ones) resulted in maximal precision of only 29% (in abstracts). In these experiments, the recall errors (i.e., missed gene names) were mostly due to the fact that some genes appeared only in tables or figures, which were not processed. However, in general, lower recall is typically caused by spelling (or other) variations. For example, Tuason and colleagues [3] reported that name variations could account for up to 79% of the missing genes if straightforward string matching was used. In their experiments with mouse gene names (similar to those reported in [2]), the overall recall was only 36.2%. They indicated that "punctuation" variation (e.g., *bmp-4* and *bmp4*), using different numerals (e.g., *syt4* and *syt iv*) or different transcriptions of Greek letters (e.g., *iga* and *ig alpha*), as well as word order variations (e.g., *integrin alpha 4* and *alpha4 integrin*) were the most frequent causes of the gene name recognition failures (see also Section 2.3, where we discuss term variation and ambiguity in the context of term mapping).

Some ATR approaches combine dictionaries with additional processing to support the term recognition process. Krauthammer and colleagues [10] suggested a method based on approximate string comparison to recognize gene and protein names and their variations. In their approach, both protein dictionaries (compiled from GenBank [11]) and target text are encoded using the "nucleotide" code (a four-letter encoding over the {A, C, G, T} alphabet). Then, the BLAST [12,13] techniques (used for alignment of DNA and protein sequences in databases) are applied to the converted text in order to identify character sequences that are similar (i.e., may be aligned) to existing gene and protein names (also encoded by the corresponding nucleotide codes). In the experiments, the system achieved 78.8% recall with the overall precision of 71.7%.

Tsuruoka and Tsujii [14] suggested a probabilistic generator of spelling variants based on edit-distance operations (namely substitution, deletion, insertion of characters and digits). Only terms with edit distance less or equal to one were considered as spelling variants. The main aims in their approach were to support expansion of (term-based) queries in order to boost IR recall (a set of generated variants was used instead of a single term to retrieve documents), and to augment existing term dictionaries with variants in order to improve dictionary-based recognition of terms in raw corpora. Recently, Tsuruoka and Tsujii [15] further described an adjusted method for approximate string matching against a dictionary of protein terms. In order to address the peculiarities of biomedical terms, they tuned the cost

[4] The GENIA corpus is a manually annotated collection of 2000 biomedical abstracts [8], in which term occurrences are tagged and further classified using the GENIA ontology. The GENIA resources are freely available at http://www-tsujii.is.s.u-tokyo.ac.jp/GENIA/.

[5] In this experiment, precision and recall were calculated by considering only genes that have been curated and (manually) assigned to the (whole) documents by FlyBase curators.

function for edit operations (e.g., substitution of a space with a hyphen (or vice versa) is considerably less "expensive" than substitution of any other two different characters). Also, to tackle the problem of false positive matches, they additionally used a naïve Bayesian classifier (with contextual and term features) trained on protein names found in the GENIA corpus. Using the two-step approach (approximate string matching with filtering false positives) they achieved precision of 73.5% at recall of 67.2% (*F*-measure: 70.2%).

2.1.2. Rule-based approaches

Rule-based approaches generally attempt to recover terms by re-establishing associated term formation patterns that have been used to coin the terms in question.[6] The main approach is to (typically manually) develop rules that describe common naming structures for certain term classes using either orthographic or lexical clues, or more complex morpho-syntactic features. Also, in many cases, dictionaries of typical term constituents (e.g., terminological heads, affixes, and specific acronyms) are used to assist in term recognition. However, knowledge engineering approaches are known to be extremely time-consuming for development, and—since rules are typically very specific—their adjustment to other entities is usually difficult.

A general grammar-based methodology for the recognition of medical terminology was suggested by Ananiadou [18], where a four-level ordered morphology was proposed to describe term formation patterns. The system used a morphological unification grammar and a lexicon with instances of specific affixes, roots, and Greek/Latin neoclassical combining forms.

Gaizauskas and colleagues [1,19,20] used a similar approach with a terminological context-free grammar for the recognition of protein names in EMPATHIE[7] and PASTA[8] systems. Their approach is based on first determining the lexical and morphological properties of the components of domain terms. The morphological analysis is geared to recognize biochemical affixes such as *-ase* or *-in* (indicating possible enzyme or protein names). Look-up in lexical resources compiled from publicly available resource enables the recognition of component categories (such as a protein head) and subcategories (such as a protein modifier). A terminology-parsing step is then used to parse the term components and combine them into single multi-token units. The necessary grammar rules have been developed semi-automatically and manually (to capture multi-word entries with no apparent structure). For example, names from the protein class are described by the following rule:

$$protein \rightarrow protein\_modifier, protein\_head, numeral.$$

A recent evaluation has shown that the overall precision of the recognizer is 84% at 82% recall for the task of recognition of 12 term classes [20].

Several systems used simpler pattern-based approaches based on orthographic and lexical peculiarities of given term classes. For example, Fukuda and colleagues [21] relied mainly on simple lexical patterns and orthographic features for the recognition of protein names. Their system, PROPER (PROtein Proper-noun phrase Extracting Rules),[9] uses the notion of "core" and "feature" components: "core" terms are words that usually bear the core of the meaning, while "feature" terms are keywords that describe the function and characteristics of terms (e.g., *protein*, *receptor*, etc.). For example, in the term "*SAP kinase*," the word *SAP* is a core term, while *kinase* is a feature term. A set of domain-specific filters (which are mainly orthographic) is used for the recognition of "core" terms. Adjacent annotations ("core" and "feature" terms) as well as nouns and/or adjectives between them, are considered part of the same "core-block" and concatenated by application of simple extension rules. For a small-scale experiment, the authors reported very good results (94.7% precision at 98.8% recall).

PROPER influenced many other systems. Narayanaswamy and colleagues [22] similarly consider other types of biomedical names (in particular chemical and source terms). Typical chemical roots and suffixes are used to single out chemicals, while different classes of "feature" terms are used to perform more sophisticated classification. In addition, contextual environments are used for further classification (e.g., the word *expression* in a context such as *expression of CD40* indicates that *CD40* is a protein/gene). Franzen and colleagues [23] developed Yapex (Yet Another Protein Extractor)[10] by adding data sources (e.g., "core" terms compiled from Swiss-Prot), additional heuristic lexical filters and results of syntactic parsing (in order to enhance the detection of name boundaries). They reported better performance compared to PROPER (for strict matching, Yapex's *F*-score was 67.1% compared to PROPER's 40.7%, while the *F*-scores were similar in case of sloppy matching). In order to further improve precision, Hou and Chen [24] considered additional filtering of candidates sug-

[6] While the majority of rule-based methods rely on what is typically inside terms, some methods use "negative" knowledge (i.e., what is outside terms) in order to recognize term boundaries [16]. For example, Blake and Pratt [17] used a stop list (containing common English stop words and some domain-specific expressions) to recognize boundaries of terms: everything between two boundary words was considered as a candidate term.

[7] See http://www.dcs.shef.ac.uk/nlp/funded/empathie.html.

[8] See http://www.dcs.shef.ac.uk/nlp/pasta/.

[9] Available at: http://www.hgc.ims.u-tokyo.ac.jp/service/tooldoc/KeX/intro.html.

[10] A demo is available at http://www.sics.se/humle/projects/prothalt/yapex.cgi.

gested by Yapex using contextual information based on most relevant collocations that appeared with protein names in a training corpus.

Hobbs [25] and Thomas and colleagues [26] customized an existing general NE recognizer (used in general-purpose IE engines Highlight and FASTUS [27]) for detection of protein names. Recognition is carried out in several phrases using a cascade of finite-state transducers, which recognize complex units (such as *3,4-dehydroproline* or *γ-glutamyl proline*) and "basic phrases" that are extended to the surrounding words using (domain-independent) rules for the construction of complex noun groups.

2.1.3. Machine-learning and statistical approaches

A variety of machine-learning (ML) and statistical techniques are used for ATR. While statistical approaches mainly address the recognition of general terms (i.e., keywords [28]), ML-systems are usually designed for a specific class of entities and, thus, integrate term recognition and term classification. ML systems use training data to "learn" features useful for term recognition and classification, but the existence of reliable training resources is one of the main problems as they are not widely available.[11] Apart from that, the main challenge is to select a set of discriminating features that can be used for accurate recognition (and classification) of term instances. Another challenge is detection of term boundaries, which are the most difficult to "learn."

Several supervised ML-methods are exploited for ATR. For example, Collier and colleagues [33] used Hidden Markov models (HMM) and specific orthographic features (e.g., "consisting of letter and digits," "having initial capital letter," etc.) for discovering terms (belonging to a set of 10 classes). Each term candidate was assigned a class of the most similar term from the training set, with respect to the orthographic similarity. To estimate the transition probabilities, maximum-likelihood estimates based on counts on the training data (the GENIA corpus) were used. Results depended on the quality of training resources: for example, for the protein class (which was the most frequent in the training set), the results were encouraging (*F*-score of 75.9%), while, on the other hand, instances of RNAs were very rare, so it was difficult to learn classification features. Similar results (the *F*-measure of 75% for the recognition of *Drosophila* gene names) have been reported by Morgan and colleagues [32], who used HMMs based on local context and simple orthographic and case variations. In addition to orthographic features, Shen and associates [34] experimented with prefix/suffix information, part-of-speech (POS) tags, and noun heads as features. They achieved *F*-scores of 16.7–80% depending on the class (overall *F*-score 66.1%; the protein class *F*-score was 70.8%), and reported that POS tags (obtained by a tagger trained on the biomedical domain) proved to be among the most useful features.

Several authors used support vector machines (SVMs) for the recognition of named entities. Kazama and colleagues [35] trained multi-class SVMs on the GENIA corpus. The corpus has been annotated with so-called B–I–O tags: B-tags denote words that are at the beginning of a term, I-tags such that are inside a term, while O-tags are used for words outside terms. The tags are complemented with the appropriate class information, i.e., a B-PROTEIN-tag denotes a word that is at the beginning of a protein name. The method aims at predicting these composite tags based on position-dependent features (such as POS, prefix, and suffix features), as well as a word cache (captures similarities of patterns with a common keyword) and HMM state features in order to address the data sparseness problem. In general, an *F*-score of 50% was achieved. They reported that considering preceding class and suffix information was helpful, while features related to POS and prefix did not have a positive influence across all experiments conducted. Several authors experimented with additional features for SVM-based term recognition and classification. Takeuchi and Collier [36] considered head-noun features, and reported that their combination with orthographic features gave better performance (*F*-score of 74.2% for 10 classes). Yamamoto and associates [37] combined boundary features (based on morpheme-based tokenization) with morpho-lexical (POS tags, stems), "biomedical" (whether a given word exists in a compiled database of biomedical resources), and syntactic features (head morpheme information). They reported that, individually, "biomedical" features were crucial for recognition of protein names. Lee and colleagues [38], however, suggested strict separation of the recognition and classification steps in the SVM-based NE recognition. For term recognition, they used "standard" features (orthographic, prefix, and suffix information) coupled with a simple dictionary-based refinement of boundaries of the selected candidates (by examining the adjacent words—if they appeared in the dictionary, they were included as part of the term). On the other hand, a set of class-specific "functional" words

[11] Few terminologically tagged biomedical corpora are available (e.g., the GENIA corpus), since it is very time-consuming to produce them manually. Thus, one of the major challenges is the automated creation of tagged corpora that can be used for ML. For example, Hatzivassiloglou and colleagues [29] used the context of "known" occurrences of genes, proteins, and mRNAs as training examples, where "known" occurrences were explicitly disambiguated in text by specifying their class (e.g., *the SB2 gene* clearly means that this occurrence of *SB2* is a gene occurrence). Craven and Kumlien [30], on the other hand, collected a set of instances of sub-cellular locations of proteins from the Yeast Protein Database [31] and then identified sentences from the associated PubMed citations in order to get an annotated corpora. A similar approach has been suggested in [32] by using lists of curated genes from FlyBase and the articles associated with them.

and contextual information were combined as features in the classification phase. They reported that this two-phase model showed better performance compared to the "standard" approach, mainly because discriminative features were selected for each subtask separately.

2.1.4. Hybrid approaches

Many approaches combine different methods (typically rule and statistically based) and various resources (pre-compiled lists of specific terms, words, affixes, etc.) for the term recognition task.

Tanabe and Wilbur [39] presented a protein and gene name tagger, ABGene, which has been trained on Medline abstracts by adapting Brill's POS tagger [40]. Apart from a set of transformation rules for the recognition of single-word gene and protein names, additional filtering and "recovering" of results is performed in order to improve both precision and recall. More precisely, false positive gene/protein names assigned by the tagger are "filtered-out" by an extensive list of pre-compiled general (i.e., non-gene and non-protein) biomedical terms and non-biological terms (obtained by comparing word frequencies in Medline with a general language corpus). On the other hand, false negative tags are "recovered" (i.e., tagged as genes/proteins) by an extensive list of proteins and genes (compiled from the LocusLink database and the Gene Ontology (GO) [41]). Also, context words are consulted: if a word is surrounded by "good" context words, it is tagged as a protein/gene. "Good" context words have been generated by a probabilistic algorithm by assigning Bayesian weights to all non-gene names that co-occurred with known names in the training set. Compound names are also extracted by relying on the combination of frequently occurring components in known multi-word gene names and a set of regular expressions. Overall, ABGene achieved precision in the range of 60–90%.

Similarly, Proux and colleagues [42] used a cascade of finite-state lexical tools to recognize single-word gene names.[12] Their method is based on a morphological POS tagger, which uses a special tag ("guessed") for tokens that cannot be matched with classical word transducers. Most gene names are tagged with the "guessed" tag, and eventually confirmed through contextual analysis (e.g., the presence of a word *gene* next to a candidate token validates its "status" as a gene-name). Special post-processing steps are necessary to recover or remove erroneously tagged tokens, including the use of a dictionary of general expressions from biology. On a small testing corpus (750 sentences obtained from the FlyBase database) they reported precision of 91.4% at the recall point of 94.4%, while when applied on a larger corpus (25,000 abstracts) the system achieved precision of 70%.

Rindflesch and colleagues [43] reported on ARBITER (Assess and Retrieve BInding TERms), which combined several approaches and resources to recognize word sequences that corresponded to binding terms. The approach selects NPs as potential "binding" terms if the NPs map to the UMLS Metathesaurus [44] or GenBank, exhibit "abnormal" morphological characteristics (compared to regular English terms), or contain heads, which are included in a constrained list of words (such as *ligand* or *subunit*). Similarly to PROPER's extension rules (see Section 2.1.2), simple binding terms are joined into complex expressions under specific conditions (e.g., prepositional modification, appositival complementation, etc.). Overall, the reported precision was 79% at 72% recall. A similar approach has been implemented for the recognition of gene, cell, and drug names in the EDGAR system [45], where characteristic words (such as *cell*, *clone*, and *expression*) occurring immediately next to target names are used to help in recognition and classification.

Finally, while the majority of methods address a specific type of entities, a method called C/NC-value, developed by Frantzi and colleagues [46] recognizes general terms. It has been used to recognize terminology in many biomedical sub-domains (e.g., in the domain of nuclear receptors [47] or from yeast corpora [48]). Term candidates are suggested by a set of morpho-syntactic filters, while their termhoods are estimated by a corpus-based statistical measure. The measure amalgamates four numerical characteristics of a candidate term, namely the frequency of occurrence, the frequency of occurrence as a substring of other candidate terms (in order to tackle nested terms), the number of candidate terms containing the given candidate term as a substring, and the number of words contained in the candidate term. The selected list of term candidates is further refined by taking into account the context of candidate terms. Context factors are assigned to candidate terms according to their co-occurrence with top-ranked context words. Experiments performed on a collection of 2082 Medline abstracts have shown the precision of 91–98% for top ranked terms recognized by the C/NC-value method [47]. The method was further augmented by the conflation of different variants of term candidates (e.g., unification of orthographic and inflectional variants, as well as acronyms) prior to the calculation of termhoods [49]. The integration of variants into the ATR process significantly improved both precision and recall of the baseline C/NC-value method [50].

2.1.5. Acronym recognition

It is well known that biomedical terms often appear in shortened or abbreviated forms. With many scientific

[12] Proux and colleagues claimed that only a small percentage of gene names were multi-word units. However, in training/testing corpora described in [23] almost half of all gene/protein names were compounds.

articles defining ad hoc abbreviations, thousands of newly coined acronyms appear yearly in the biomedical literature [51,52]. Therefore, the ability to "understand" acronyms is obviously critical for an NLP system, so the recognition and linking of acronyms and their expanded forms (EFs) is an essential part of term identification. Although there are many existing acronym repositories in the biomedical field [52,53], it has been reported that such resources cover only parts of the acronyms that appear in documents [54].

The discrepancy between curated acronym resources and the wealth of acronyms defined in biomedical articles fostered the development of several acronym recognition systems. In order to locate potential acronym definitions in text, the majority of approaches use pattern matching based on "parenthetical forms" (i.e., occurrences of acronyms within parentheses). Then, an optimal definition candidate string is selected and the candidate EF is analysed with the aim of discovering the relation between a given acronym and the expanded candidate EF (or its substring).

One of the first attempts to compile acronyms from literature was by Yoshida and colleagues [55]. The system, called PNAD-CSS (Protein Name Abbreviation Dictionary - Construction Support System), aimed at the recognition of protein acronyms, and the PROPER system [21] was used for spotting (expanded) target protein names in text. Apart from initial letters of words, they considered the initial characters of syllables in order to match an acronym to a protein name. They reported precision of 98.9% and recall of 95.6%.

Yu and colleagues [54] designed the rules for the recognition of gene/protein symbols and the corresponding full names after the examination of published gene/protein nomenclatures. They combined morphological cues, special "functional" keywords, and positional information. Standard pattern matching rules have been also adapted by two special modifications: numbers and special characters are ignored for mapping short forms to full names, and the identification of special abbreviations and the corresponding forms (such as *Y* for *tyrosine*) has been included. The manual evaluation has shown that the approach achieved 93% precision and 73% recall.

Similar but more general rule-based methods have been also suggested. Liu and colleagues [56] reported on a method (called PW3) for matching three-letter acronyms (including some chemical acronyms). Nenadic and associates [49] introduced a simple rule-based method for discovering and linking acronyms with their EFs from raw text. Matching patterns were modelled by a manually defined grammar that defined common "rules" for coining new acronyms (including using initial letters from affixes used in the corresponding EFs). Also, extracted acronym/EF pairs were grouped so that acronyms sharing "normalized" EFs were conflated by unifying orthographic, structural, and lexical variations. Yu and colleagues [57] presented a pattern matching approach (called AbbRE) that was based on a set of general rules for mapping an abbreviation to its EF. AbbRE applies the rules in a sequence, and prefers a shorter EF for an extracted acronym. They reported an average precision of 95% and recall of 70%. Schwartz and Hearst [58] suggested a general algorithm for the extraction of the shortest corresponding EF for a given acronym. They used only few common constraints, such as the first character of an acronym has to be the first character of the first word in the corresponding EF; EF should be longer than the corresponding acronym; EF should not contain the candidate acronym itself. In the experiments on the MEDSTRACT corpus,[13] they accomplished 99% precision at 84% recall, while on a larger test corpus the method achieved recall of 82% at precision of 95%.

One of the main challenges of the acronym acquisition task is to select an optimal EF: the majority of errors in raw-text based methods are related to the size of the window used for searching for the potential EF. Therefore, additional text pre-processing was used in order to improve the recognition of EFs. For example, Pustejovsky and colleagues [59] based their approach on results of shallow parsing: the size of the window is determined by morpho-syntactic properties and only NPs are considered as candidate EFs. The system, called ACROMED, achieved precision of 98.3% at 72% recall on the MEDSTRACT corpus.

Finally, Chang and colleagues [52] presented a supervised ML approach to acronym recognition that used a binary logistic regression classifier. Feature vectors used for recognition were based on three types of features: features describing acronym patterns (e.g., percentage of lower case letters), features describing how the acronym letters are linked to EFs (e.g., percentages of letters aligned at the beginning of words, on syllable boundaries, etc.), and features related to the alignment (e.g., number of words from an EF used to match letters in a given acronym, the average number of matched characters per word, etc.). The method was also evaluated against the MEDSTRACT corpus: the system achieved 95% precision at 75% recall. This method was used to automatically scan all Medline abstracts and to compile an acronym database.[14]

2.2. Term classification

We have been discussing term recognition as a method to locate lexical units that are related to domain con-

[13] The MEDSTRACT testing corpus (http://www.medstract.org/) contains 100 Medline abstracts with 168 manually marked occurrences of acronyms [59].

[14] Available at http://bionlp.stanford.edu/abbreviation/.

cepts. Term recognition does not further specify the meaning of a term; it is the role of term classification to pinpoint the specific type of a domain concept (such as a gene, protein, or mRNA) that is described by the term. In other words, term classification gives a first clue on the identity of a term, which is an important step towards final term identification. For example, classification may help to select a specific resource useful for term mapping.[15] In technical terms, the classification task is to disambiguate between the possible (broader) senses of terms (if more than one), which is known as term sense disambiguation.

Many term classification systems use functional words, such as *receptor*, *factor*, or *radical* for assigning term categories [20,21,23,60]. However, more often than not, terms do not contain any explicit term category information. In such situation, statistical disambiguation may be warranted. For example, Nobata and colleagues [61] combined the use of functional words with statistical methods for term classification. In their experiment, they compared a naïve Bayesian method with a decision-tree approach for classifying terms into different molecular classes such as protein, DNA, and RNA. In the former, conditional probabilities of word w being assigned to class c have been learnt from category-specific as well as background word lists, the former being compiled from resources such as SwissProt and GenBank, the latter from a large collection of Medline abstracts. The words within a term were then used to determine the class probability. The presence of specific head nouns (acting as functional words) took precedence when determining the term class. The method was tested on 100 manually tagged Medline abstracts (the tag set was derived from the GENIA ontology). The method based on decision-trees relied on three kinds of feature sets (POS information, character type information, and category-specific word lists) and was cross-validated on the same corpus as above. The naïve Bayesian method (F-score 65.8%) showed lower performance than the decision-tree approach (F-score between 87.7 and 90.1%) for classifying terms (assuming perfect term recognition—which has been done manually). They also attempted term classification with automatic term recognition, scoring significantly lower F-scores for the classification task.

Unlike the previous method, which relies on internal evidence for classification, most statistical disambiguation approaches are based on information flanking an ambiguous term. For example, Hatzivassiloglou and associates [29] described a statistical approach for disambiguating between proteins, genes, and mRNAs. They experimented with different ML techniques (naïve Bayesian classification, decision trees, and inductive learning) for term disambiguation, and evaluated several types of classification features (such as words that appeared near a term, positional, morphological, distributional, and shallow syntactic information). They found that using word positional information lowered accuracy (because of data sparseness), while POS information helped the overall accuracy, but only modestly (less than 1%). Overall, their approach showed accuracy between 69.4 and 85% for a two-way classification task (gene/protein) and between 65.9 and 78.1% for a three-way classification task (gene/protein/mRNA). These results compare favourably to a human expert inter-annotator agreement rate of 77.6% when performing the same classification task manually.

Torii and Vijay-Shanker [62] similarly used an unsupervised bootstrapping method (based on decision lists) for learning contextual environments for a given set of classes (namely proteins, chemical names, and sources). Further, Torii and colleagues [60] experimented with term internal (functional words and suffixes) and external (words occurring nearby) sources for the classification of molecular names as chemicals, proteins, and other classes. They also used a term similarity measure (based on lexical resemblance among terms) to measure the distance to previously classified entities. The similarity measure achieved high precision and recall (93 and 84%), and outperformed methods based on internal and external features.

Spasic and associates [63] looked at term classification for the task of ontology management, where it is of interest to automatically expand ontologies with newly discovered terms. They used genetic algorithms to refine verb selectional preferences and to assign classes associated with domain verbs. The class of a novel term is chosen based on co-occurrence with a domain verb, as well as a similarity measure to known terms with established term–class relationships. In an evaluation study involving 28 different classes (a subtree of the UMLS semantic network), the approach achieved a mean classification precision of 64.2% (recall was 49.9%).

Raychaudhuri and colleagues [64] described annotation of *Saccaromyces cerevisiae* gene names with Gene Ontology (GO) codes using a word-based maximum entropy measure. The measure acts as a classifier for journal abstracts, which enables GO mapping for (all) genes that appear in those abstracts. Nenadic and associates [48] further explored how different text-based features influenced the annotation performance using SVMs. The features included document identifiers (i.e., gene–gene co-occurrence within the same document), single words, and automatically extracted terms. The experiment showed that linguistic pre-processing of single words (such as lemmatization and stemming) did not significantly boost the performance. Terms (acting as

[15] In the example presented in Section 1, we classified *mdm2*, in '*p53 protein suppresses mdm2 expression*,' as a gene, and consequently we selected a gene resource (i.e., LocusLink) for the final term identification.

semantic features) improved the performance at low recall points, while document identifiers achieved superior results compared to the other features.

2.3. Term mapping

Term mapping is typically the final step in the term identification process. Its aim is to map a term occurrence to an entry in a referent data source, annotating the term with a referent ID. Term mapping faces two main problems: the extensive variability of lexical term representations, and the problem of term ambiguity with respect to mapping into a data source. The former is linked to the fact that biomedical terms often appear in different surface forms. For example, different orthographic variations (e.g., *NF kappa B*, *NF kappaB*, and *NF-kappa B*), inflectional and morphological variants (e.g., *transcription intermediary factor-2* and *transcriptional intermediate factor 2*), structural variations (e.g., *clones of human* and *human clones*), and lexical alternatives (e.g., *hepatic microsomes* and *liver microsomes*) are very frequent. Since many of such variants are missing from domain resources, it is typically difficult to link term occurrences to referent entries directly (i.e., forms appearing in documents differ from those stored in databases; see [2,3,15,57]). On the other hand, we often encounter term ambiguity with respect to a one-to-many relationship between a term and entries in referent data sources. The ambiguity complicates the mapping of a term, as it is typically not trivial to select an appropriate entry. For example, the term *CAT*, even if previously classified as a protein, has many potential candidate entries in the Swiss-Prot protein database (such as *catalase*, *carnitine o-acetyltransferase*, as well as different *CAT* entries for different species). Tuason and colleagues [3] discuss further issues that are relevant for term mapping. First, there is high ambiguity of biomedical terms with common English words (see also [2]). It seems necessary, therefore, to include a disambiguation step to identify common English words early in the term identification pipeline. Second, terms should be linked to the appropriate species before mapping.[16]

In this Section we will briefly review how research in term normalization and disambiguation tries to overcome the major challenges in term mapping. We start by discussing strategies that deal with the problem of term variability, and then present approaches to term disambiguation.

2.3.1. Handling term variability

We use a broad definition of variability that includes simple variations such as differences in spelling, as well as more complex variation (commonly called synonymy). Recently, there has been work towards a better understanding of the variability issues with regard to biomedical names. For example, Cohen and colleagues [66] have written about variability and normalization of gene and protein names. They differentiate between *contrastive* features, "which can be used to distinguish two samples of natural language with different meaning," and *non-contrastive* variability in the form of spelling variations in synonymous names. They suggested heuristics that allowed the mapping (i.e., conflation) of (synonymous) variants of gene and protein names to a canonical referent. These heuristics included equivalence of vowel sequences, optionality of hyphens and parenthesized material, and case insensitivity. On the other hand, they found "edge effects" (for example, a number at the last position of a protein name) to be contrastive, i.e., changing the meaning (i.e., identity) of a term.

Other approaches to conflation of terminological variants have been also suggested (e.g. [15,67]). For example, Jacquemin and Tzoukermann [68] discussed conflation of multi-word terms by combining stemming and terminological look-up. Stemming was used to reduce words so that conceptually and linguistically related words were normalized to the same stem (thus resolving some orthographic and morphological variations), while a terminological thesaurus might be used for spotting synonyms and linking lexical variants.

The MetaMap program [69], which maps noun phrases identified by the SPECIALIST minimal commitment parser to UMLS Metathesaurus concepts, demonstrates the use of term variation in the process of mapping terms into a domain resource. MetaMap uses a multi-level mapping strategy, which first analyses a target term to "generate" a multitude of variants, such as acronyms, synonyms, and inflectional variants. Each of these derivations of the original term is then mapped against concept names in the Metathesaurus. The method compares the "strength" of the mapping for each term variant, ordering possible mapping candidates. MetaMap has been used in several research projects that depended on mapping to the UMLS Metathesaurus, such as hierarchical indexing, data mining in clinical reports, and automated indexing of documents.[17]

Referent data sources often do not contain the complete set of synonyms of a given concept, complicating the mapping process. There has been work towards automatically finding term synonyms in documents. This work (as well as work on acronyms recognition,

[16] Seewald [65] recently discussed the use of several ML classifiers (naïve Bayesian, SVM, and others) to learn species domains (kingdoms) from Medline abstracts.

[17] MetaMap is available online as MetaMap Transfer (MMTx), at http://mmtx.nlm.nih.gov/.

reviewed in Section 2.1.5) is useful for extending the scope of biomedical dictionaries, which boosts the chance of successfully mapping synonyms. As an example of such work, Yu and Agichtein [70] experimented with four different approaches (namely unsupervised, partially supervised, and supervised ML approaches, as well as a rule-based system) for the extraction of gene and protein synonyms that occurred within the same sentence. The unsupervised ML approach was based on comparison of mutual information of synonym candidates with respect to other words in their neighbouring contexts, while the partially supervised, bootstrap method used a set of seed synonym occurrences to learn "contexts" that indicated occurrence of synonyms (e.g., fragments such as <*GENE*> *also known as* <*GENE*>). The supervised SVM-based method used the same seed occurrences to learn a classifier that classified each textual context surrounding a pair of gene/protein names as "positive" or "negative" with respect to synonymy. Finally, the rule-based system (called GPE) was based on a set of manually defined lexical extraction patterns that indicated typical contexts used to express synonymy. While GPE had high precision with low recall, all ML-approaches traded off precision for higher recall (for example, the precision of 7% at the recall point of 72%). Still, by combining ML-approaches with GPE, the performance significantly improved over all individual approaches.

2.3.2. Handling term ambiguity

The second major problem with term mapping is related to the problem of term ambiguity with respect to referent data sources. Broad classification (reviewed in Section 2.2) can resolve much of term ambiguity, but is useless in situations where a term has different meanings within a specific term class. For example, broad classification may help to disambiguate between *CAT* as a protein, animal, or medical device, but it is ineffective in situations where *CAT* can be mapped to several different protein entries in a protein data source. In such situations, *specific* classification on the level of dictionaries is useful. For example, the work by Liu and associates [71] aimed to disambiguate terms associated with several entries in the UMLS Metathesaurus. Given a term, the method first identifies a set of corresponding UMLS concept identifiers (CUIs), representing the different term senses. Using the UMLS information on relationships between concepts, the method then identifies other UMLS concepts (called the relative CUI set) that have relationships with the original sets of concepts. Using unambiguous concept names of the relative CUI set, the method builds a classifier for each sense of the term. In an evaluation study, the authors experimented with 35 abbreviations with multiple senses in UMLS. The overall precision was 96.8% at 50.6% recall.

Other approaches have also been suggested for mapping ambiguous acronym occurrences[18] to their referent entries. Pustejovsky and colleagues [59] used a simple word-based vector space model for disambiguation of acronyms with multiple meanings (the POLYFIND system). After collecting a set of abstracts for each meaning, a new abstract (with an occurrence of the ambiguous acronym) is compared to each of the corresponding "meaning" sets by using the standard *tf***idf* weighting and the cosine similarity. A set with the highest similarity is used to assign the interpretation to all occurrences in the new abstract. This approach resulted in 97.6% accuracy. Pakhomov [72] used a maximum-entropy classifier on the sentence level by using only the $[-2,+2]$ context window approach to find a correct interpretation of a given acronym. Since he used a set of clinical notes for experiments, he also experimented with features based on the headings (titles) of the sections in which ambiguous acronyms appeared. He reported that there were no significant differences between the two approaches: precision was in average almost 90%. These results suggest that approaches to acronym sense disambiguation—even without any sophisticated information—are promising, but it is obvious that the training resources are needed.

3. Conclusions and challenges

Term identification is crucial for the automated processing of the biomedical literature [2,3,73]. The importance of the topic has triggered fascinating research on the problems of recognizing, classifying, and mapping term occurrences in biomedical texts. From the first descriptions of the term recognition problem (see for example [21]) to the latest published research, there has been a steady improvement of the understanding of the underlying issues and challenges.

Term recognition systems have been developed for many classes of biomedical entities, in particular for gene and protein names. They are based either on internal characteristics of specific classes or on external clues that can support the recognition of word sequences that represent specific domain concepts. Different types of features are used, such as orthographic (capital letters, digits, and Greek letters) and morphological clues (specific affixes and POS tags), or syntactic information from shallow parsing. Also, different statistical measures are suggested for "promoting" term candidates into terms. Discovering acronyms and uncovering their "meaning" is also an essential part of term recognition, since acronyms are very frequent in the biomedical do-

[18] Chang and colleagues [52] claimed that more than one-fifth of all acronyms extracted from Medline were ambiguous.

main. Precision of ATR methods is typically in the 70–90% range, while recall, in the best cases, is around 70%. Still, it is not possible to thoroughly compare different systems as they have different targets, and common test collections are still rare [2]. Some attempts have been made only recently to organize joint evaluation schemes (e.g., the BIOCREATIVE initiative[19]).

Although tremendous work has been done on ATR, some challenges still need additional research. For example, more accurate recognition of term boundaries is needed, as the majority of existing systems address only maximally long-term candidates (which may include some insignificant modifiers, thus complicating subsequent term mapping). Further, recognition of internal term structure and nested (embedded) subterms is essential, in particular since nested terms are common in the biomedical domain.[20] For example, when recognizing the term *leukaemic T cell line Kit225*, it would be useful to have all its nested terms (*cell line*, *T cell line*, *Kit225*, and *leukaemic T cell line*) recognized and highlighted in text. Such information may prove valuable in the subsequent term identification process. Further challenges include handling of both term variation that affect term constituents (e.g., orthographic and morphological variants) and term structure (e.g., recognition of terms that are "encoded" in term coordinations, like terms *estrogen receptor* and *progesterone receptor* in the coordination *estrogen and progesterone receptors* [9]). Finally, recognition of other classes of terms (not only proteins, genes, and chemical compounds) is vital for successful mining of the biomedical literature.

The recognition of lexical units that correspond to domain concepts is not the ultimate goal of term identification: assigning terms to broader biomedical classes and/or to referent databases is an additional challenge. However, the variation and inconsistencies in surface expressions of terms as well as their ambiguity create a major problem for term classification and mapping. *Term classification* is typically based on either functional words that are embedded in concept names, or on contextual characteristics of term occurrences. On the other hand, *term mapping* to referent databases typically needs lexical and morphological "normalization" for matching to existing databases entries, as well as disambiguation for ambiguous terms.

Although the term identification process can be conceptually and methodologically presented through the three steps (recognition, classification, and mapping), in many cases practical solutions merge some of these tasks, blurring the boundaries between them. For example, term recognition and classification are often performed in a single step, where the same features are used to single out term candidates and to categorize them. Also, some researchers have pointed to the dual role of dictionary-based term recognition approaches, which effectively map recognized (unambiguous) terms to the respective dictionary entries [10,15]. Nevertheless, some authors stress the advantages of tackling each step individually, pointing at the different information sources needed to accomplish each sub-task [38,60].[21] It is an open issue whether a clear separation into single steps would improve term identification. Obviously, if separated, it is easier to modularize the term identification task, so that different solutions can be used for each specific problem. For example, if a general, class-independent term recognition method is used, then—in order to successfully categorize entities of a new term class of interest—researchers would have to concentrate only on the design of a classification method. Further, separation would allow for the selection of more relevant and more discriminative features for each of the subtasks.

Also, it seems clear that accurate classification (done prior to term mapping) can be helpful for more accurate linking of ambiguous terms to referent sources. For example, the author of MetaMap discusses the inclusion of statistical disambiguation to resolve situations where terms map to several different concepts in the UMLS [69]. This is a question of practicality: it seems difficult to build a classifier for each ambiguous term in a referent database. The solution might be a step-wise approach, where a broad classification of terms (for example according to UMLS semantic types) maps most of the term occurrences, and where the remaining terms are mapped by individual term classifiers.

Further issues—especially in term mapping—still wait to be addressed. For example, many recognized terms do not appear in referent resources, although highly (conceptually) related entries can be located. Krauthammer and associates [10] have speculated that mapping of such terms can be done to parent concepts of terms. For example, given a database entry *interleukin-2*, it may be possible to map a term such as *interleukin-3*, which is not in the database and is contrastive to *interleukin-2*, to a parent concept of both terms, such as *interleukin*. This would necessitate the inclusion (or generation) of parent terms in the database, as is the case in

[19] BIOCREATIVE (Critical Assessment of Information Extraction systems in Biology) was organized for the first time as a challenge cup in 2003, in which one of the sub-tasks was related to protein/gene name recognition and identification (in the same, shared set of documents). The evaluation showed that the best methods achieved *F*-scores of 80%, with both the best precision and recall values of around 80%. For details see http://www.mitre.org/public/biocreative/.

[20] A recent study by Ogren and colleagues [74] reported that, for example, two-thirds of GO-ontology terms contained another GO-term as a proper substring.

[21] For example, Lee and colleagues [38] reported that POS information was useful for the term recognition task, while it was not effective for classification.

most ontologies. Blaschke and Valencia [73] point to a related problem of terms that refer to families or group of proteins. Without a corresponding entry in a reference database, such family terms cannot be mapped. As an example, consider the (family) name *MAP kinase*, which can map to both *Erk1* and *Erk2* (in humans). The mapping can be further complicated by the fact that it is unclear whether an author refers to the family or either of the entities. Furthermore, in some cases, even a narrow context may not be always sufficient to disambiguate a term (e.g., when a protein name is shared among different species), and wider context (e.g., a whole article) may need to be analysed before terms can be mapped.

Apart from the identification of each and every term occurrence in text, a further challenge is to select the most representative or the most important terms (and entities) that are "discussed" in a given document. This challenge concerns the problems of sophisticated document indexing for improving the quality of information retrieval, which is crucial for database curation[22] and other time-consuming annotation tasks. For this, methods that measure the representativeness of the recognized (and identified) names (e.g. [75,46]) are preferred.

Since biomedical literature is expanding so dynamically, the demand from the user community is directed towards practical and useful systems that are able to identify and link relevant "entities" in literature to databases. Relying exclusively on existing controlled vocabularies to identify terminology in text suffers from both low recall and low precision, as such resources are insufficient for automatic terminology mining. Having in mind the pace of the development in the domain and the rate of coinage of new terms, it is unlikely to expect that any terminology standardization will occur in the near future. Therefore, automatic term identification tools will be for long valuable assets for literature mining and knowledge integration in the biomedical domain.

Acknowledgments

M.K. thanks Dr. Carol Friedman for her help in clarifying important issues in term identification, and G.N. thanks Dr. Sophia Ananiadou for fruitful discussions on problems and challenges in ATR and automatic term management. Both authors are grateful to the anonymous reviewers for their constructive comments.

[22] See also tasks 1B and 2 of the BIOCREATIVE 2003 initiative (http://www.mitre.org/public/biocreative/).

References

[1] Gaizauskas R, Demetriou G, Humphreys K. Term recognition and classification in biological science journal articles. In: Proceedings of Workshop on Computational Terminology for Medical and Biological Applications. Patras, Greece; 2000. pp. 37–44.
[2] Hirschman L, Morgan AA, Yeh AS. Rutabaga by any other name: extracting biological names. J Biomed Inform 2002;35(4):247–59.
[3] Tuason O, Chen L, Liu H, Blake JA, Friedman C. Biological nomenclature: a source of lexical knowledge and ambiguity. In: Proceedings of Pacific Symposium on Biocomputations; 2004. pp. 238–49.
[4] The FlyBase database of the *Drosophila* genome projects and community literature. Nucleic Acids Res 2003;31(1):172–5.
[5] Boeckmann B, Bairoch A, Apweiler R, Blatter MC, Estreicher A, Gasteiger E, et al. The SWISS-PROT protein knowledgebase and its supplement TrEMBL in 2003. Nucleic Acids Res 2003;31(1):365–70.
[6] Collier N, Nobata C, Tsujii J. Automatic term identification and classification in biological texts. In: Proceedings of Natural Language Pacific Rim Symposium. Beijing, China; 1999. pp. 369–74.
[7] Pruitt KD, Maglott DR. RefSeq and LocusLink: NCBI gene-centered resources. Nucleic Acids Res 2001;29(1):137–40.
[8] Ohta T, Tateisi Y, Mima H, Tsujii J. GENIA corpus: an annotated research abstract corpus in molecular biology domain. In: Proceedings of Human Language Technology Conference (HLT 2002). 2002. pp. 73–7.
[9] Nenadic G, Spasic I, Ananiadou S. Mining biomedical abstracts: What is in a term? In: Proceedings of International Joint Conference on NLP. Sanya, China; 2004. pp. 247–54.
[10] Krauthammer M, Rzhetsky A, Morozov P, Friedman C. Using BLAST for identifying gene and protein names in journal articles. Gene 2000;259(1–2):245–52.
[11] Benson DA, Karsch-Mizrachi I, Lipman DJ, Ostell J, Rapp BA, Wheeler DL. Genbank. Nucleic Acids Res 2000;28(1):15–8.
[12] Altschul SG, Gish W, Miller W, Myers EW, Lipman DJ. Basic local alignment search tool. J Mol Biol 1990;215(3):403–10.
[13] Altschul SF, Madden TL, Schaffer AA, Zhang J, Zhang Z, Miller W, et al. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. Nucleic Acids Res 1997;25(17):3389–402.
[14] Tsuruoka Y, Tsujii J. Probabilistic term variant generator for biomedical terms. In: Proceedings of 26th Annual ACM SIGIR Conference. 2003. pp. 167–73.
[15] Tsuruoka Y, Tsujii J. Boosting precision and recall of dictionary-based protein name recognition. In: Proceedings of NLP in Biomedicine, ACL 2003. Sapporo, Japan; 2003. pp. 41–8.
[16] Bourigault D, Gomzalez-Mullier I, Gros C. LEXTER, a Natural language processing tool for terminology extraction. In: Proceedings of EURALEX '96. 1996. pp. 771–9.
[17] Blake C, Pratt W. Better Rules, Fewer Features: A semantic approach to selecting features from text. In: Proceedings of IEEE Data Mining Conference. San Jose, California; 2001. pp. 59–66.
[18] Ananiadou S. A Methodology for automatic term recognition. In: Proceedings of COLING-94. Kyoto, Japan; 1994. pp. 1034–8.
[19] Humphreys K, Demetriou G, Gaizauskas R. Two applications of information extraction to biological science journal articles: enzyme interactions and protein structures. In: Proceedings of Pacific Symposium on Biocomputations. 2000. pp. 505–16.
[20] Gaizauskas R, Demetriou G, Artymiuk PJ, Willett P. Protein structures and information extraction from biological texts: the PASTA system. Bioinformatics 2003;19(1):135–43.

[21] Fukuda K, Tamura A, Tsunoda T, Takagi T. Toward information extraction: identifying protein names from biological papers. In: Proceedings of Pacific Symposium on Biocomputations. 1998. pp. 707–18.
[22] Narayanaswamy M, Ravikumar KE, Vijay-Shanker K. A biological named entity recognizer. In: Proceedings of Pacific Symposium on Biocomputations. 2003. pp. 427–38.
[23] Franzen K, Eriksson G, Olsson F, Asker L, Liden P, Coster J. Protein names and how to find them. Int J Med Inf 2002;67(1–):49–61.
[24] Hou W, Chen H. Enhancing performance of protein name recognizers using collocation. In: Proceedings of NLP in Biomedicine, ACL 2003. Sapporo, Japan; 2003. pp. 25–32.
[25] Hobbs JR. Information extraction from biomedical text. J Biomed Inform 2002;35(4):260–4.
[26] Thomas J, Milward D, Ouzounis C, Pulman S, Carroll M. Automatic extraction of protein interactions from scientific abstracts. In: Proceedings of Pacific Symposium on Biocomputations. 2000. p. 541–52.
[27] Hobbs JR, Appelt D, Bear J, Israel D, Kameyama M, Stickel M, et al. FASTUS: A Cascaded Finite-State Transducer for Extracting Information from Natural-Language Text. In: Finite-State Language Processing. Cambridge: MIT press; 1997. p. 383–406.
[28] Andrade MA, Valencia A. Automatic extraction of keywords from scientific text: application to the knowledge domain of protein families. Bioinformatics 1998;14(7):600–7.
[29] Hatzivassiloglou V, Duboue PA, Rzhetsky A. Disambiguating proteins, genes, and RNA in text: a machine language approach. Bioinformatics 2001;17(Suppl. 1):S97–106.
[30] Craven M, Kumlien J. Constructing biological knowledge bases by extracting information from text sources. In: Proceedings of Int Conf Intell Syst Mol Biol. 1999. pp. 77–86.
[31] Hodges PE, Payne WE, Garrels JI. The Yeast Protein Database (YPD): a curated proteome database for *Saccaromyces cerevisiae*. Nucleic Acids Res 1998;26(1):68–72.
[32] Morgan A, Yeh A, Hirschman L, Colosimo M. Gene name extraction using flybase resources. In: Proceedings of NLP in Biomedicine, ACL 2003. Sapporo, Japan; 2003. pp. 1–8.
[33] Collier N, Nobata C, Tsujii J. Extracting the names of genes and gene products with a hidden markov model. In: Proceedings of COLING 2000. Saarbruecken; 2000. pp. 201–7.
[34] Shen D, Zhang J, Zhou G, Su J, Tan C. Effective adaptation of hidden markov model-based named entity recognizer for biomedical domain. In: Proceedings of NLP in Biomedicine, ACL 2003. Sapporo, Japan; 2003. pp. 49–56.
[35] Kazama J, Makino T, Ohta Y, Tsujii J. Tuning support vector machines for biomedical named entity recognition. In: Proceedings of Workshop on NLP in the Biomedical Domain, ACL 2002. Philadelphia, PA; 2002. pp. 1–8.
[36] Takeuchi K, Collier N. Bio-medical entity extraction using support vector machines. In: Proceedings of NLP in Biomedicine, ACL 2003. Sapporo, Japan; 2003. pp. 57–64.
[37] Yamamoto K, Kudo T, Konagaya A, Matsumoto Y. Protein name tagging for biomedical annotation in text. In: Proceedings of NLP in Biomedicine, ACL 2003. Sapporo, Japan; 2003. pp. 65–72.
[38] Lee K, Hwang Y, Rim H. Two-phase biomedical NE recognition based on SVMs. In: Proceedings of NLP in Biomedicine, ACL 2003. Sapporo, Japan; 2003. pp. 33–40.
[39] Tanabe L, Wilbur WJ. Tagging gene and protein names in biomedical text. Bioinformatics 2002;18(8):1124–32.
[40] Brill E. A simple rule-based part-of-speech tagger. In: Proceedings of ANLP-92. Trento, IT; 1992. pp. 152–5.
[41] Harris MA, Clark J, Ireland A, Lomax J, Ashburner M, Foulger R, et al. The Gene Ontology (GO) database and informatics resource. Nucleic Acids Res 2004;32(1):D258–61.
[42] Proux D, Rechenmann F, Julliard L, Pillet VV, Jacq B. Detecting gene symbols and names in biological texts: a first step toward pertinent information extraction. In: Proceedings of Ninth Workshop on Genome Informatics. 1998. pp. 72–80.
[43] Rindflesch TC, Hunter L, Aronson AR. Mining molecular binding terminology from biomedical text. In: Proceedings of AMIA Symposium. 1999. pp. 127–31.
[44] Humphreys BL, Lindberg DA, Schoolman HM, Barnett GO. The unified medical language system: an informatics research collaboration. J Am Med Inform Assoc 1998;5(1):1–11.
[45] Rindflesch TC, Tanabe L, Weinstein JN, Hunter L. EDGAR: extraction of drugs, genes and relations from the biomedical literature. In: Proceedings of Pacific Symposium on Biocomputations. 2000. pp. 517–28.
[46] Frantzi K, Ananiadou S, Mima H. Automatic recognition of multi-word terms: the c-value/nc-value method. Int J Digit Libr 2000;3(2):115–30.
[47] Ananiadou S, Albert S, Schuhmann D. Evaluation of automatic term recognition of nuclear receptors from medline. Genome Informatics Series 2000.
[48] Nenadic G, Rice S, Spasic I, Ananiadou S, Stapley BJ. Selecting text features for gene name classification: from documents to terms. In: Proceedings of NLP in Biomedicine, ACL 2003. Sapporo, Japan; 2003. pp. 121–8.
[49] Nenadic G, Spasic I, Ananiadou S. Automatic Acronym acquisition and term variation management within domain-Specific texts. In: Proceedings of LREC-3. Las Palmas, Spain; 2002. pp. 2155–62.
[50] Nenadic G, Spasic I, Ananiadou S. Terminology-driven mining of biomedical literature. Bioinformatics 2003;19(8):938–43.
[51] Adar E. S-RAD: A Simple and Robust Abbrevation Dictionary. USA: HP Lab; 2002.
[52] Chang JT, Schutze H, Altman RB. Creating an online dictionary of abbreviations from medline. J Am Med Inform Assoc 2002;9(6):612–20.
[53] Rimer M, O'Connell M. BioABACUS: a database of abbreviations and acronyms in biotechnology and computer science. Bioinformatics 1998;14(10):888–9.
[54] Yu H, Hatzivassiloglou V, Rzhetsky A, Wilbur WJ. Automatically identifying gene/protein terms in Medline abstracts. J Biomed Inform 2003;35(5–6):322–30.
[55] Yoshida M, Fukuda K, Takagi T. PNAD-CSS: A Workbench for Constructing a Protein name abbrevation dictionary. Bioinformatics 2000;16(2):169–75.
[56] Liu H, Aronson AR, Friedman C. A study of abbreviations in MEDLINE abstracts. In: Proceedings of AMIA Symposium. 2002. pp. 464–8.
[57] Yu H, Hripcsak G, Friedman C. Mapping abbreviations to full forms in biomedical articles. J Am Med Inform Assoc 2002;9(3):262–72.
[58] Schwartz AS, Hearst MA. A simple algorithm for identifying abbreviation definitions in biomedical text. In: Proceedings of Pacific Symposium on Biocomputations. 2003. pp. 451–62.
[59] Pustejovsky J, Castano J, Cochran B, Kotecki M, Morrell M, Rumshisky A. Extraction and Disambiguation of Acronym–Meaning Pairs in Medline. In: Proceedings of Medinformatics. 2001.
[60] Torii M, Kamboj S, Vijay-Shanker K. An Investigation of Various Information Sources for Classifying Biological Names. In: Proceedings of NLP in Biomedicine, ACL 2003. Sapporo, Japan; 2003. pp. 113–20.
[61] Nobata C, Collier N, Tsujii J. Automatic term identification and classification in biological texts. In: Proceedings of Natural Language Pacific Rim Symposium. 1999. pp. 369–74.
[62] Torii M, Vijay-Shanker K. Using unlabeled MEDLINE abstracts for biological named entity classification. In: Proceedings of Genome Informatics Workshop 2002. 2002. pp. 567–658.

[63] Spasic I, Nenadic G, Ananiadou S. Using domain-specific verbs for term classification. In: Proceedings of NLP in Biomedicine, ACL 2003. Sapporo, Japan; 2003. pp. 17–24.
[64] Raychaudhuri S, Chang JT, Sutphin PD, Altman RB. Associating genes with gene ontology codes using a maximum entropy analysis of biomedical literature. Genome Res 2002;12(1): 203–14.
[65] Seewald A. Towards recognizing domain and species from MEDLINE publications. In: Proceedings of European Workshops on Data Mining and Text Mining for Bioinformatics. 2003. pp. 51–8.
[66] Cohen KB, Acquaah-Mensah GK, Dolbey AE, Hunter L. Contrast and variability in gene names. In: Proceedings of Workshop on NLP in the Biomedical Domain, ACL 2002. Philadelphia, PA; 2002. pp. 14–20.
[67] Jacquemin C. Spotting and Discovering Terms through NLP. Cambridge, MA: MIT Press; 2001.
[68] Jacquemin C, Tzoukermann E. NLP for Term Variant Extraction: A Synergy of Morphology, Lexicon and Syntax. In: Strzalkowski T, editor. Natural Language Information Retrieval. Boston, MA: Kluwer; 1999. p. 25–74.
[69] Aronson AR. Effective mapping of biomedical text to the UMLS Metathesaurus: the MetaMap program. In: Proceedings of AMIA Symposium. 2001. pp. 17–21.
[70] Yu H, Agichtein E. Extracting synonymous gene and protein terms from biological literature. Bioinformatics 2003;19(Suppl. 1): I340–9.
[71] Liu H, Johnson SB, Friedman C. Automatic resolution of ambiguous terms based on machine learning and conceptual relations in the UMLS. J Am Med Inform Assoc 2002;9(6):621–36.
[72] Pakhomov S. Semi-supervised maximum entropy based approach to acronym and abbreviation normalization in medical texts. In: Proceedings of 40th annual meeting of ACL. 2002. pp. 160–7.
[73] Blaschke C, Valencia A. Molecular biology nomenclature thwarts information-extraction progress. IEEE Intell Syst 2002;17(3): 73–6.
[74] Ogren P, Cohen K, Acquaah-Mensah G, Eberlein J, Hunter L. The compositional structure of gene ontology terms. In: Proceedings of Pacific Symposium on Biocomputations 2004. pp. 214–25.
[75] Hisamitsu T, Tsujii J. Measuring term representativeness. In: Pazienza MT, editor. Information Extraction in the Web Era, LNAI 2700. New York, NY: Springer; 2003. pp. 45–76.