Ulf Leser
is a professor for Knowledge Management in Bioinformatics at the Humboldt-Universität in Berlin. His main topics of research are semantic data integration, text mining and biomedical data management.

Jörg Hakenberg
is a researcher in the group of Ulf Leser. His work in text mining focuses on information extraction for life science applications, especially for systems biology.

Keywords: *text mining, knowledge management, information extraction, machine learning, named entity recognition*

What makes a gene name? Named entity recognition in the biomedical literature

Ulf Leser and Jörg Hakenberg

Date received (in revised form): 21st July 2005

Abstract

The recognition of biomedical concepts in natural text (named entity recognition, NER) is a key technology for automatic or semi-automatic analysis of textual resources. Precise NER tools are a prerequisite for many applications working on text, such as information retrieval, information extraction or document classification. Over the past years, the problem has achieved considerable attention in the bioinformatics community and experience has shown that NER in the life sciences is a rather difficult problem. Several systems and algorithms have been devised and implemented. In this paper, the problems and resources in NER research are described, the principal algorithms underlying most systems sketched, and the current state-of-the-art in the field surveyed.

INTRODUCTION

Over the past years, biology has become an information science.[1] While traditionally characterised by experiments and observations, biology is more and more concerned with the analysis of large amounts of information. Consequently, the way that information is stored, managed, visualised and searched is of growing importance.[2] This can be witnessed both in the ever-growing number of databases[3] and in the rapidly growing number of publications.

At the time of writing, the Medline database[4] contained approximately 15 million scientific abstracts with a growth rate of about 400,000 articles per year. Owing to these immense numbers, it is very difficult for a biologist to keep up with the literature, even in a very specialised area of research. Tools to help researchers cope with the information overload are therefore needed. Furthermore, many new technologies in molecular biology are high-throughput methods, such as gene expression arrays,[5] yeast-2-hybrid screens[6] or protein identification using mass spectrometry.[7] Studies using these techniques typically lead to experimental results for thousands of objects at the same time. For many interesting questions these results need to be augmented with external information. For instance, analysis of microarray data greatly benefits from considering functional annotation of genes in addition to pure expression levels.[8] This additional information is often very difficult to achieve because it is mostly available only in the form of free text, ie in scientific publications, and not in structured databases. Since searching textual resources is much more difficult than searching structured databases, this information is often called 'hidden'. One method to uncover such information hidden in free text is text mining.

Building on techniques from machine learning, natural language processing and pattern matching, text mining is concerned with algorithms and tools for searching large text collections despite the fuzziness of human language. In this context, 'searching' must be understood in a broad sense, including information extraction (for faster search afterwards), document clustering (for searching by similarity) and document classification (for improving text indexing). Two prominent examples of text-mining

Ulf Leser,
Department for Computer Science,
Humboldt-Universität zu Berlin,
Rudower Chaussee 25,
12489 Berlin, Germany

Tel: +49 30 2093 3902
Fax: +49 30 2093 5484
E-mail: leser@informatik.hu-berlin.de

Named entity recognition: prerequisite for text mining systems

applications in the life sciences are the 'Related Articles' functionality in PubMed[9] and the extraction of gene/protein interaction networks from a text collection.[10] However, prior to finding protein interactions, one must first identify proteins. Though this might sound like a trivial task to solve, it is not. Over the past years it has turned out that the recognition of biological objects in written language is very difficult due to many factors, including a general lack of naming conventions, excessive use of abbreviations, frequent usage of synonyms and homonyms, and the fact that biological objects often have names consisting of many single words, such as 'human T-cell leukaemia lymphotropic virus type 1 Tax protein'. The latter example also shows that it is usually not clear where a name starts or ends, even for human readers. Many biologists would argue that the first word 'human' and the last word 'protein' are not part of the protein name itself, while others would claim the contrary.

In this paper, we survey the current state-of-the-art in named entity recognition (NER) for biomedical applications, ie the task of identifying gene, protein, diseases, etc, names in natural text. Though the NER problem is not interesting to biologists itself, NER tools are important building blocks for text-mining tools supporting biologists.[11] Therefore, the NER problem has received considerable attention over the past years. (Note that the NER problem is studied for much longer in information retrieval, but usually on objects such as company or person names. For an overview, search for proceedings of the Message Understanding Conferences (MUC).)

Workshops and competitions on NER

Early papers on the topic are Nobata *et al.*,[12] Craven and Kumlien[13] and Fukuda *et al.*,[14] and the number of publications has grown considerably over the past five years. Several international competitions on NER have been held, such as the shared task at the Joint Workshop on Natural Language Processing in Biomedicine and its Applications[15] and the Critical Evaluation of Information Extraction Systems in Biology.[16] A recent special issue of the *Journal of Biomedical Informatics* was devoted entirely to this topic,[17] including a technical survey on NER systems.[18] NER in the life sciences is also covered briefly in several surveys on general text mining.[11,19]

In the next section, we discuss NER in general, putting a focus on the evaluation of NER. This is followed by a discussion on basic tools and algorithms. A description is then given of several NER systems, grouped by the fundamental method they use.

NAMED ENTITY RECOGNITION

NER actually consists of three different problems – the recognition of a named entity in text, the assignment of a class to this entity (gene, protein, drug, etc), and the selection of a preferred term for naming the object in case that synonyms exist. The latter is especially important if the recognised entities are to be combined with information from other resources, such as databases. In this survey, we concentrate on the first two problems. Further, we make no explicit distinction between these two tasks, as most NER systems are class specific, ie they are designed to find only objects of one particular class or set of classes.

NER tools can be applied to find all kinds of entities, such as gene or protein names, diseases and drugs,[20] mutations[21] or properties of protein structures.[22] However, most current systems concentrate on gene/protein names and do not distinguish between these two classes. Therefore, this paper also focuses on gene and protein names.

Problems in NER

From a semantic perspective, NER is difficult because of two sources of confusion. Recall that names in text represent real-life concepts in our mind. For most objects of our daily language, there exists a social agreement on the nature of real-life concepts (what is a

Concepts, meaning and representation

supermarket?), and on the way these concepts are represented in language (how do I name a supermarket?). However, in domains that are so quickly changing and highly specialised as molecular biology, such agreements do not have enough time to build or are subject to frequent modifications. As a consequence, both the real-life concepts and their textual representations are not unambiguously defined. As an example, there is no community-wide agreement on how a particular gene should be named, with the exception of very prominent genes such as *p53*. Second, the concept denoted by a gene name is usually not clearly defined. One name can stand for a particular gene, may include homologues of this gene in other organisms, may be restricted to a specific splice variant, or may also encompass the protein the gene encodes. This fuzziness is a strength of human language, not a weakness, since it allows us to communicate more concisely. It is much simpler to explain a certain discovery in natural language than represent it in a database.

What makes NER complicated? Synonyms, homonyms, abbreviations and ambiguities

The very same fuzziness is a problem when names in text should be detected automatically. When many synonyms for a given object are in use, information in different articles cannot be mapped to each other any more. For instance, clones during the mapping phase of the Human Genome Project had up to 15 different names,[23] making it very difficult to integrate information from different sources. Similarly, many genes and proteins have more than one name. Furthermore, especially in the beginning of the genomic era, gene names were not distinguished from normal language. Many genes from *Drosophila*, one of the first genomes studied, are named after a specific phenotype of a mutant. Gene names such as 'white' (symbol *w*), 'shaggy' (symbol *ssg*), or 'mind the gap' (symbol *mtg*) make it almost impossible to find gene-related articles using full-text search. The problem gets worse through inconsistent use of variations of names, including prefixes or suffixes with digits or letters ('*MRP2*', '*MRP3*', '*Dbf4p*'), use of special characters ('*Cbp/p300*-interacting transactivator'), or Greek letters ('CCAAT/enhancer binding protein (C/EBP), alpha'). These problems are not likely to disappear in the near future. Even if gene name standards such as those set by the HUGO committee would be used more widely, the large amount of existing publications would still contain 'old' names.

More problems for NER arise from multi-word names and acronyms. Multi-word names are names consisting of more than one word (or token). In the area of gene and protein names, multi-word names are rather the rule than the exception. For instance, in the BioCreative corpus of expert-tagged gene names, 53 per cent of all names consist of more than one token, and similar figures hold for other corpora.[24] Multi-word names are not only harder to find, but in many cases there is no agreement on the exact borders of such names, making evaluation of NER tools difficult (see section on 'Evaluation of NER systems').

Acronyms are abbreviations of names and are very popular in scientific writing because they allow for shorter texts. However, acronyms are difficult to resolve to their true names because they are often homonyms. For instance, the acronym ACE stands for 'angiotensin converting enzyme', 'affinity capillary electrophoresis', 'acetylcholinesterase' and a couple of other things.[25] However, tools for detecting and resolving acronyms typically report an accuracy of over 95 per cent. For a detailed discussion of acronym resolution see Krauthammer and Nenadic.[18]

Evaluation of NER systems

The purpose of an NER system is the automatic analysis of large amounts of text. It is important to know what quality can be expected from a particular method. Therefore, NER systems are evaluated using a 'gold standard', ie a collection of text (a corpus) where all interesting

The rate of annotated text corpora

entities are marked (or tagged or annotated) by a human expert. Such gold standards serve as benchmarks for comparing different systems, and for tuning single systems. The performance of NER tools is typically measured in terms of precision (percentage of true entity names in all entity names found; also called specificity) and recall (percentage of true entity names found compared to all true entity names; also called sensitivity). These two figures are combined to the so-called *F*-measure, defined as the harmonic mean of precision and recall.

Five facts making NER evaluation difficult

In contrast to this simple scheme, evaluating NER tools is actually rather difficult for a number of reasons. First, there are very few corpora available that are sufficiently large to allow the extrapolation of evaluation results to large text collections such as PubMed. Very often tools are evaluated on only between 10 and 100 abstracts. These results cannot be compared robustly with the results of other algorithms.

Second, the mark-up of entity names in the gold standard greatly depends on the particular person performing the annotation. No biologist knows thousands of genes and protein names including all their synonyms, abbreviations and variants. Thus, even an expert will usually not find all names in a text. Therefore, this process should always be performed by more than one person to enable the measurement of inter-annotator consistency,[26] which defines a natural upper bound on the quality achievable by automatic systems. It is also very important that annotators are given clear guidelines before starting their task. It frequently has been observed that annotation behaviour changes over time as annotators gather more experience with their task. Thus, intra-annotator consistency is also an important issue.[27] Unfortunately, none of the three large NER corpora discussed in the section 'Corpora and training data' was generated in such a way that intra- or inter-annotator measures could be derived. Inter-annotator consistency was only measured on smaller corpora, and is reported as being between 75 and 90 per cent for gene and protein names.[22,28] Considering that the best current NER systems reach an *F*-measure around 85 per cent, there is a real danger that all systems reporting better results will only represent an overfitting of the method to the particular gold standard, ie annotator. Robust improvements exceeding an *F*-measure of approximately 85 per cent seem to be only possible within a specific project having a specific focus with a stable team of annotators.

Interannotator agreement: NER is difficult even for humans

This claim is supported by the third problem in evaluating NER systems. There is strong evidence that the severity of the NER problem depends on the particular class of names being searched. For instance, finding genes and protein names is simpler for yeast than for human or mouse, and particularly difficult for *Drosophila*.[29] Some corpora include mRNA or RNA in their scope for 'gene' names, others do not, again leading to problems with varying degrees of difficulty.[30] In particular, it is very hard to differentiate between gene and protein names.

Fourth, there are different definitions of 'true positive', again rendering the comparison of results very difficult. Under a strict evaluation scheme, algorithms need to recognise entity phrases exactly as specified by the annotator, with exact left and right boundaries. Under a loose evaluation scheme, the true and the predicted entity phrases only need to overlap, and various formulas exist for exactly scoring the degree of overlap.[24] Note that missing a border under a strict scheme, ie making a phrases a word too long or a word too short, is punished twice by the standard evaluation, since it results in a false negative (the exact name was not found) and a false positive (a 'false' name was found). The detection of word boundaries has turned out to be one of the most difficult tasks in NER, and thus performance results differ substantially for different evaluation schemes. For

instance, Chang *et al.*[31] report a 25 per cent difference in *F*-measure between strict and partial matches on the Yapex corpus, while we found a 10 per cent difference on the BioCreative corpus,[32] although this corpus partially allows for alternative name boundaries.

Turning NER

Finally, evaluation should always be performed in a task-specific manner, as systems can and must be tuned to meet different needs. For instance, generation of large protein-interaction networks intended for human browsing might use NER tools with partial matches and tuned for high recall, since false or incomplete names are tolerated by human users. In contrast, database curators are often only interested in systems offering high precision to reduce the amount of manual intervention.[33]

TOOLS

NER systems mostly build on techniques from the areas of machine learning and of natural language processing. Many tools in these areas are freely available. With machine learning, we refer to the analysis of data sets to extract models describing the data. For NER, these data sets mostly are text corpora and dictionaries, ie lists of entity names.

Machine learning (ML)

NER and classification

Several machine learning techniques have been used for the NER task. We discuss the three most prominent examples: naive Bayes, support vector machines and hidden Markov models.

Naive Bayes approaches analyse distributions of properties within different classes, thus reflecting that certain properties appear more often in one particular class than in all others. Based on the properties of a new instance, they calculate the probabilities for belonging to either class. This approach can be used to classify words or phrases as being an entity name or not, using properties such as letter frequencies or word length characteristics. Implementations of naive Bayes algorithms, as many other machine learning techniques, are for instance available in the WEKA toolkit.[34]

Support vector machines (SVM) are another widely used technique to solve classification problems. From a given set of feature vectors for positive and negative examples, SVMs deduce linear combinations of features from appropriate examples called support vectors. These support vectors define a hyperplane in the multidimensional feature space, separating (ideally all) positive examples from all negative examples. Freely available SVM implementations include SVM^{light}[35] and libSVM.[36] The latter is particularly useful due to many existing interfaces to scripting languages. The features used for SVM classification in general are the same as for naive Bayes.

Hidden Markov models (HMM) use the order in which features appear in text, such as the order of words in a sentence. They aggregate statistical information from labelled examples to predict the most probable sequence of events (eg series of grammatical features) for a given sentence. To encode typical characteristics of single words, HMMs use derived features similar to those used in other ML approaches. A freely available implementation of HMMs is, for instance, Jahmm.[37]

Natural language processing (NLP)

Many NER systems use meta-information generated by grammatically analysing sentences. For instance, algorithms for part-of-speech (POS) annotation classify tokens in a sentence into nouns, verbs, adjectives, prepositions, etc (see Figure 1). Text chunking provides information on noun and verbal phrases. The subject–object structure can be discovered with so-called shallow parsers. Finally, some systems use full sentence parsing, but they must cope with the fact that only a small fraction of all sentences can be fully parsed using current technology.[38]

POS tagging is the most frequently used NLP technique. One of the first

The/DT codon/NN usage/NN is/VBZ particularly/RB marked/VBN for/IN the/DT gag/NN ,/, pol/NN ,/, and/CC env/JJ genes/NNS ./.

Figure 1: Sentence with part-of-speech tags generated by TnT. Tags are NN: Normal Nomina; IN: preposition or conjunction; CC: conjunction; VBZ/N: verb in present/past tense; DT: determiner; JJ: adjective. Special characters are tagged as themselves

Natural language processing

publicly available POS taggers is the rule-based Brill tagger.[39] Tags'n'Treegrams is based on an HMM and is one of the best performing POS-taggers to date.[40] TreeTagger includes both a tool for POS-tagging and for sentence chunking.[41] All POS-taggers come with models trained on standard corpora, giving them a bias towards sentence structures found in newspaper articles, but can be re-trained on annotated texts dedicated to specific NER tasks. Zhou *et al.* report that POS taggers adapted to the biomedical domain help to improve performance.[42] Clegg and Sheperd present a detailed comparison of standard and adapted taggers on the GENIA corpus.[43] A retrained POS tagger can even be used directly as a biomedical NER tool, as shown in Smith *et al.*[44]

Corpora and training data

Annotated examples for learning rules and models

Virtually all NER systems require annotated corpora to train their models or deduce their rules. The best-known corpora for gene/protein NER are BioCreative,[45] GENIA[46] and Yapex.[24] Table 1 gives some figures describing these corpora. For diseases and treatments, the BioText corpus provides annotated examples.[47] All corpora have been used for the evaluation of NER systems, and for some NER systems measures on more than one corpus have been published. Inter-corpus cross-validations – training on one corpus and evaluating on the other – are especially important to learn about systematic differences between annotators. The differences are surprisingly high. Dingare *et al.*[28] report a 13 per cent difference in the achieved *F*-measure of their system when run on the BioCreative and BioNLP/NLPBA corpus, respectively. Our own experiences with all three NER corpora strongly support this observation (unpublished).

Note that studies on the information distribution in corpora also provide useful insights into language usage in life science publications from a more general perspective. In particular, there is a great difference in the information content of different sections of a publication (introduction, methods, results, etc), as shown by several authors.[48,49] Text-mining systems can exploit this information to adopt specific models for mining abstracts, full papers, patient records, patents, etc.

For all classes of named entities, dictionaries of name lists are used to enhance NER performance – either indirectly by deriving characteristics of the names in the dictionary to build better models, or directly by matching

Table 1: Properties of the three most important corpora for training and evaluating NER systems

	Size	Entity types	Number of tagged entities	Evaluation scheme	Text selection	Metadata shipped with corpus
BioCreative	15,000 sentences	Gene/protein	17,800	Strict, with some alternative word boundaries	Random selection by Genbank curators	PubMed-ID, POS, tokenisation, sentence splitting
GENIA	2,000 abstracts, approx. 19,000 sentences	Various	eg 21,800 protein_molecule, 8,353 DNA_domain_or_region	Strict	PubMed query	PubMed-ID, POS, tokenisation, sentence splitting
Yapex	201 abstracts	Protein	3,711	Strict	PubMed query + at random from GENIA	PubMed-ID

dictionaries against the given text. Such dictionaries can be inferred from curated databases and thesauri, such as UniProt for protein names,[50] IUBMB for enzyme names[51] and MeSH for medical terms.[52]

NER SYSTEMS

In the following, we discuss a number of different NER systems, grouped by their fundamental algorithmic approach. We mostly omit performance measures because they are most often incomparable (see section 'Evaluation of NER systems'). Published values range from 40 per cent up to 95 per cent *F*-measure. The best performing systems in the 2004 BioCreative competition yielded *F*-measures around 83 per cent, and the best performing system in the 2004 BioNLP/NLPBA competition reached about 70 per cent *F*-measure. Table 2 gives a selection of NER tools that are freely available.

Rule-based NER systems use heuristic and machine-learned rules to identify names

Dictionary-based approaches

Dictionary-based NER approaches match text against a fixed name list

Dictionaries essentially are large collections of names, serving as examples for a specific entity class. Matching dictionary entries exactly against text is a simple and very precise NER method, but yields only very low recall. To compensate, one can either use inexact matching techniques, or try to 'fuzzify' the dictionary by automatically generating typical spelling variants for every entry. The extended dictionary is then used for exact matches against the text.[29,53] Notably, even BLAST[54] has been used as an inexact matching algorithm.[55] Characters, numbers and special symbols are all encoded using four-tuples of DNA bases. BLAST then searches for similarities between phrases and a dictionary derived from GenBank gene names that was encoded using the same scheme.

When dictionaries are built from database entries, dictionary-based approaches have the important advantage that term normalisation becomes trivial. For instance, the Whatizit system is able to directly link protein names to their respective UniProt-ID using a dictionary generated from the UniProt database.[33]

Rule-based approaches

Rule-based approaches build on the definition of rules to separate different classes. Early systems used hand-crafted rules to describe the composition of named entities and their context. For instance, surface clues (capital letters, symbols, digits) might be used to extract candidates for protein names.[14] These core terms are expanded according to a set of syntactic rules. As an example, particular functional terms surrounding a candidate are included in the protein name (such as 'kinase' or 'receptor'). A POS-tagger may be used to also provide grammatical information for further expansion rules. For instance, it is a common rule to include all parts of a nominal phrase in the entity name as soon as at least one of its tokens was identified as part of an entity name.

The PASTA system uses a mixture of heuristic and machine-learned rules.[22] Twelve different classes of entities (ranging from 'protein' to 'quaternary structure') are represented by a set of templates. The system consists of two major components, one for terminology

Table 2: List of freely available NER tools

Tool	Recognised entities	Available as	Web page
GAPSCORE	Genes and proteins	Online form and web service	http://bionlp.stanford.edu/gapscore
ABNER	Protein, DNA, RNA, cell line, cell type	Java application and API	http://www.cs.wisc.edu/~bsettles/abner/
KeX	Proteins	Shell and Perl scripts	http://www.hgc.jp/service/tooldoc/KeX/intro.html
AbGene	Genes	Binaries	ftp://ftp.ncbi.nlm.nih.gov/pub/tanabe/AbGene
LingPipe	Genes	Online form and Java API	http://www.alias-i.com/lingpipe/

identification and one for template filling. The usage of heuristic rules, ie rules generated with human intelligence, makes training on a rather small training corpus possible.

Compared with other approaches, rule-based systems are not robust towards unseen names. Another drawback is the time-consuming process of manually deducing and tuning patterns from examples, and the substantial amount of interference between rules. Sophisticated rule-based systems often reach very high precision. However, the more specific the rules are, the lower the achieved recall.

Classification-based approaches rely on characteristic features of phrases

Classification-based approaches

The most popular technique in biomedical NER is to transform the NER task into a classification problem, either for single words or for multi-word phrases. Applying a naive Bayes classifier, Nobata *et al.* trained a model to distinguish between four different categories of named entities: source, protein, DNA and RNA.[12] They also compared their system to a decision tree learner and report that the latter outperformed the naive Bayes model.

YamCha is a word-based classification system using support vector machines.[56] Features are different types of surface clues and morpho-syntactic properties of named entities and their surrounding words, as well as matches of tokens against a dictionary. YamCha transforms the binary decision (named entity or not) into a three-class problem, distinguishing the first word of a named entity from following words (BIO, begin–inside–outside of an entity), and reports that this transformation boosted performance.

GAPSCORE uses features such as dictionary matches, word occurrence, context and word morphology.[31] Word hits are expanded into phrases using POS information. The system computes individual weights for many features by measuring differences in frequency of appearance in positive examples and arbitrary Medline words. The authors compared naive Bayes, SVM and a Maximum Entropy model reporting only minor differences between all three algorithms.

The approach proposed by Curran and Clark[57] applies a maximum entropy tagger to the NER problem. Though no results have been presented for biomedical NER, the technique proved highly successful for non-scientific NER, ie for recognising person names, locations and organisations. We discuss another approach using maximum entropy Markov models for biomedical NER in the next section.

Classification approaches usually consider each word one-by-one, and mostly do not take the order of words into account. Features generated from a word or the surroundings of a word are represented as an order-independent vector. Some systems differentiate between features seen before, inside or after a named entity.[56] Methods including the sequential order of words are discussed in the section 'Sequence-based approaches'.

All machine learning approaches are very sensitive to the selection of features used for training and classification. Choosing the 'right' (ie best performing) set of features is a time-consuming and error-prone task. It is mostly performed in a trial-and-error fashion, testing different combinations of features without any guarantee that another corpus would not require another feature set. Automated feature selection therefore is a very important subproblem of ML-based NER. For instance, we found that the prediction quality of our NER classifier on the BioCreative corpus is practically not effected by removing up to 95 per cent of all features.[32] In addition, this paper discusses a large variety of different features proposed for NER and their respective influences on prediction performance.

Sequence-based approaches

In contrast to classification-based approaches, which only consider words or phrases, sequence-based systems consider

the complete ordered sequence of tokens and part-of-speech tags in a sentence. All these systems perform a statistical analysis on the training corpus, and later deduce the most probable sequence of tags for a given sequence of words.

Kinoshita *et al.*[58] use a simple but efficient way of including sequential information by retraining the TnT-Tagger on a biomedical corpus using standard POS tags plus specific tags for entity names. Some rule-based post-processing stages are added for correcting errors, especially word boundary-related errors. In addition, abbreviation resolution detects missed hits in cases where either the long or short form is tagged.

Dingare *et al.*[30] report on a system using a maximum entropy classifier as the basic component of a maximum entropy Markov model. Using a Viterbi-style algorithm, the system predicts the most probable sequence of single classifications for the tokens of a sentence. Maximum entropy models are able to deal with large numbers of input features: in the case of Dingare *et al.*,[30] approximately 1.25 million features were used.

Conditional random fields are another probabilistic sequence tagging framework.[59] On the BioCreAtIvE corpus, conditional random fields were one of the best performing methods.

Advantages and disadvantages

There exist a number of reasons for using dictionary-, rule- or machine learning-based techniques. These approaches differ in the amount of human intervention needed to build a system, the adaptability to other entity classes, the possibilities to incorporate expert knowledge and, of course, the quality of predictions.

Hybrid approaches try to exploit the advantages of different techniques

Machine learning techniques depend greatly on the existence, size and quality of annotated textual examples. Using a dictionary as the only input to a model saves the laborious task of annotating a corpus, but hinders the inclusion of contextual information. Fixed lists of names, collected from an expert-curated database, also provide unique identifiers for each instance, a problem not solved by machine learning approaches. On the other hand, to improve recall, usage of dictionaries necessitates refinement of entries or fuzzy matching algorithms to achieve robustness towards spelling variants and unseen names, leading to the severe danger of overfitting.

Rule-based systems incorporate expert knowledge in a simple way. Assuming the existence of a proper rule definition language, the model, ie the set of rules, itself is human readable, easy to understand, and can be extended step-by-step, moving from common to more specialised rules. In contrast, machine learning approaches typically only generate large lists of numbers that cannot be interpreted in any useful manner. However, rule-based systems suffer from a strong trade-off between precision and recall. Highly specific rules lead to high precision, but in the extreme case, a new rule must be designed for every example, causing bad recall owing to unseen examples. On the other hand, rules that are too general lead to better recall but low precision.

In addition, once a rule set has been defined it cannot be adapted automatically to different entity classes. In contrast, machine learning approaches can simply be retrained on a different corpus with different classes of entities being marked. This retraining often leads to good results very quickly, though re-engineering of feature generation and selection is necessary to achieve the best possible results. It is natural to consider hybrid approaches to combine the best of all worlds. Such systems are discussed next.

Hybrid approaches

Current NER systems often do not rely only on a single technique, but use processing pipelines with multiple stages (see Figure 2). Pre-processing prepares and enriches texts, using NLP methods for proper sentence-splitting and POS tagging, and adds annotation from external resources. Later, several machine

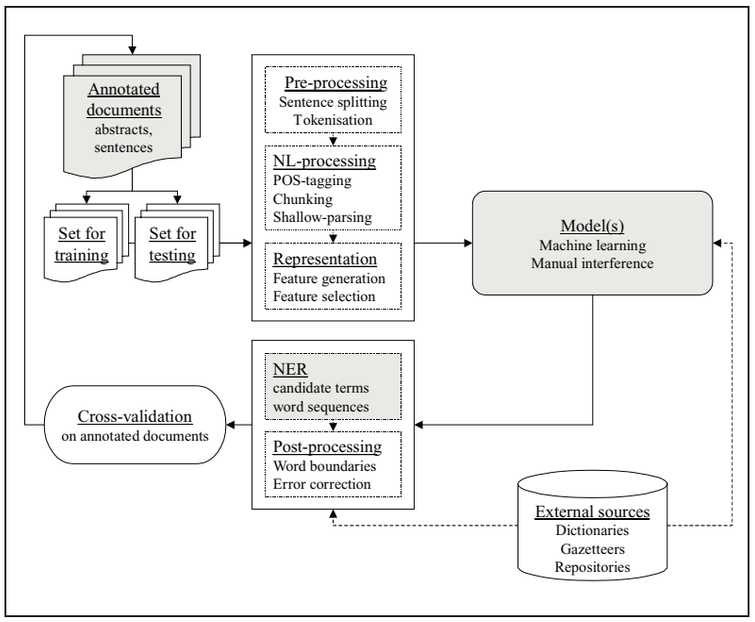

Figure 2: Typical process flow in NER. Training documents are transformed into a different representation (eg vector space model), on which basis models are built, sometimes using external data. These models are applied on a test collection, and detected named entities undergo a post-processing procedure

Post-processing refines predicted candidate names

learning techniques are applied in parallel to mark entities. Rule-based post-processing stages deal with refinement of predicted candidates, resolution of abbreviations and exploitation of multiple occurrences of the same entity within the text.

Whenever more than one classifier is used in parallel, post-processing must include a form of decisions making in case of conflicting results. This can be implemented by using meta-learning, the simplest form of which are voting schemes. The system in Zhou *et al.*[60] uses an ensemble of two HMMs and one SVM classifier with majority voting. The two HMMs are trained on different corpora (BioCreAtIvE and GENIA). This seems to enable the system to properly adapt to different corpus properties. Another meta-learner is described in Mika and Rost.[61] Here, three different SVMs are used that are trained on different corpora using different feature sets. A fourth SVM takes the results of the three original SVMs as features and generates the final result.

Learning from unlabelled examples when there are no corpora available

CONCLUSION

NER in biomedicine has been studied for approximately ten years. In this time, a number of companies have emerged that also offer commercial text-mining tools, usually branded as 'knowledge management' tools. Examples are Temis©, Linguamatics© and ClearForest©. Over those ten years, systems have grown considerably in complexity, starting from simple rule-based pattern matchers to sophisticated, hybrid machine learning classifiers.

It is very likely that this growth in complexity has also led to better performance, though it is not possible to prove this claim, as reasonable comparisons of systems have been performed only in the past few years. As discussed in the section 'Evaluation of NER systems', it is unlikely that much further improvement is possible on the NER problem on general classes, but progress is likely in specialised areas. In particular, species-specific NER is a promising direction, but currently still hindered by the lack of sufficiently large, species-specific corpora. Furthermore, specialised NER systems for diseases, drugs, species names, cell lines, etc probably will soon appear, as the techniques are readily available. It remains to be seen which of these classes will turn out to form easy or difficult problems.

The lack of corpora remains a severe obstacle to further development. Therefore, it is likely that future systems will address learning from unlabelled samples. If at least some examples can be tagged by an automated system with very high precision, NER systems may consider surrounding words to learn reliable context patterns. Using these patterns, more examples could be labelled, thus enriching the corpus step by step.

There is a trend in life science text mining to move away from NER and closer to biology-related tasks, such as detecting functional properties of genes or

Guidelines for authors will support text mining in the future

interactions of proteins. However, NER remains a very important topic since text-mining applications will always perform only as well us the underlying NER methods. One interesting development in this area is the emerging interest of publishers. Ultimately, the NER problem might be solved not by sophisticated algorithms, but by journals forcing authors to tag entity names upon submission of a manuscript.[33,62] However, even if such tools were available, the problem of lacking standards for naming hundreds of thousands of genes and proteins in many organisms remains open, as there will remain millions of already existing and untagged articles, containing the (hidden) knowledge of many decades of biomedical research.

Acknowledgments
This work is supported by the German Federal Ministry of Education and Research (BMBF) under grant contract 0312705B. JH is additionally supported by the German Foreign Exchange Service (DAAD), reference number D/05/26768.

References

1. Kanehisa, M. (2000), 'Post-genome Informatics', Oxford University Press, Oxford.
2. Augen, J. (2001), 'Information technology to the rescue!', *Nature Biotechnol.*, Vol. 19(6), pp. BE39–BE40.
3. Galperin, M. Y. (2005), 'The Molecular Biology Database Collection: 2005 update', *Nucleic Acids Res.*, Vol. 33 (Database issue), pp. D5–24.
4. URL: http://www.ncbi.nlm.nih.gov/entrez
5. Schulze, A. and Downward, J. (2001), 'Navigating gene expression using microarray – a technology review', *Nature Cell Biol.*, Vol. 8(8), pp. E190–195.
6. Legrain, P. and Selig, L. (2000), 'Genome-wide protein interaction maps using two-hybrid systems', *FEBS Lett.*, Vol. 480(1), pp. 32–36.
7. Lin, D., Tabb, D. L. and Yates, J. R. (2003), 'Large-scale protein identification using mass spectrometry', *Biochim. Biophys. Acta*, Vol. 1646(1–2), pp. 1–10.
8. Hvidsten, T. R., Laegreid, A. and Komorowski, J. (2003), 'Learning rule-based models of biological process from gene expression time profiles using gene ontology', *Bioinformatics*, Vol. 19(9), pp. 1116–1123.
9. Wilbur, W. J. and Yang, Y. (1996), 'An analysis of statistical term strength and its use in the indexing and retrieval of molecular biology texts', *Comput. Biol. Med.*, Vol. 26(3), pp. 209–222.
10. Jenssen, T. K., Laegreid, A., Komorowski, J. and Hovig, E. (2001), 'A literature network of human genes for high-throughput analysis of gene expression', *Nature Genet.*, Vol. 28(1), pp. 21–28.
11. Blaschke, C., Hirschman, L. and Valencia, A. (2002), 'Information extraction in molecular biology', *Brief. Bioinformatics*, Vol. 3(2), pp. 1–12.
12. Nobata, C., Collier, N. and Tsujii, J. (1999), 'Automatic term identification and classification in biology texts', in 'Proc. Natural Language Pacific Rim Symposium', November, Beijing, China.
13. Craven, M. and Kumlien, J. (1999), 'Constructing biological knowledge bases by extracting information from text sources', in 'Proceedings of the 7th International Conference on Intelligent Systems for Molecular Biology, AAAI Press, Menlo Park, CA, pp. 77–86.
14. Fukuda, K., Tamura, A., Tsunoda, T. and Takagi, T. (1998), 'Toward information extraction: identifying protein names from biological papers', in 'Proceedings of the 3rd Pacific Symposium on Biocomputing', 4th–9th January, Hawaii, pp. 707–718.
15. URL: http://www.genisis.ch/~natlang/JNLPBA04/
16. URL: http://www.pdg.cnb.uam.es/BioLINK/BioCreative.eval.html
17. Ananiadou, S., Friedman, C. and Tsujii, J. I. (2004), 'Special issue on named entity recognition in biomedicine', *J. Biomed. Inform.*, Vol. 37(6).
18. Krauthammer, M. and Nenadic, G. (2004), 'Term identification in the biomedical literature', *J. Biomed. Inform.*, Vol. 37(6), pp. 512–526.
19. Cohen, A. M. and Hersh, W. R. (2005), 'A survey of current work in biomedical text mining', *Brief. Bioinformatics*, Vol. 6(1), pp. 57–71.
20. Rindflesch, T. C., Tanabe, L., Weinstein, J. N. and Hunter, L. (2000), 'EDGAR: Extraction of drugs, genes and relations from the biomedical literature', in 'Proceedings of the 5th Pacific Symposium on Biocomputing, 4th–9th January, Hawaii, pp. 517–528.
21. Horn, F., Lau, A. L. and Cohen, F. E. (2004), 'Automated extraction of mutation data from the literature: Application of MuteXt to G protein-coupled receptors and nuclear hormone receptors', *Bioinformatics*, Vol. 20(4), pp. 557–568.

22. Gaizauskas, R., Demetriou, G., Artymiuk, P. J. and Willett, P. (2003), 'Protein structures and information extraction from biological texts: the PASTA system', *Bioinformatics*, Vol. 19(1), pp. 135–143.

23. Leser, U., Lehrach, H. and Roest Crollius, H. (1998), 'Issues in developing integrated genomic databases and application to the human X chromosome', *Bioinformatics*, Vol. 14(7), pp. 583–590.

24. Franzen, K., Eriksson, G., Olsson, F. *et al.* (2002), 'Protein names and how to find them', *Int. J. Med. Inf.*, Vol. 67(1–3), pp. 49–61.

25. Adar, E. (2004), 'SaRAD: A Simple And Robust Abbreviation Dictionary', *Bioinformatics*, Vol. 20(4), pp. 527–533.

26. Bayerl, P., Lüngen, H., Gut, U. and Paul, K. (2003), 'Methodology for reliable schema development and evaluation of manual annotation', in 'Proceedings of the Workshop on Knowledge Markup and Semantic Annotation at the 2nd International Conference on Knowledge Capture (K-CAP)', 23rd–25th October, Sanibel Island, FL.

27. Mani, I., Hu, Z., Wu, C. *et al.* (2004), 'Protein name tagging guidelines: Lessons learned', in 'Proceedings of the SIG BioLink, in conjunction with ISMB/ECCB 2004', 31st July–4th August, Glasgow, UK.

28. Dingare, S., Nissim, M., Finkel, J. *et al.* (2004), 'A system for identifying named entities in biomedical text: How results from two evaluations reflect both the system and the evaluation', in 'Proceedings of the SIG BioLink, in conjunction with ISMB/ECCB 2004', 31st July–4th August, Glasgow, UK.

29. Hanisch, D., Fundel, K., Mevissen, H.-T. *et al.* (2004), 'ProMiner: Organism-specific protein name detection using approximate string matching', in 'Proceedings of the EMBO workshop BioCreative: Critical Assessment for Information Extraction in Biology', 28th–31st March, Granada, Spain.

30. Dingare, S., Finkel, J., Manning, C. *et al.* (2004), 'Exploring the boundaries: Gene and protein identification in biomedical text' in 'Proceedings of the EMBO workshop BioCreative: Critical Assessment for Information Extraction in Biology', 28th–31st March, Granada, Spain.

31. Chang, J. T., Schutze, H. and Altman, R. B. (2004), 'GAPSCORE: Finding gene and protein names one word at a time', *Bioinformatics*, Vol. 20(2), pp. 216–225.

32. Hakenberg, J., Bickel, S., Plake, C. *et al.* (2005), 'Systematic feature evaluation for gene name recognition', *BMC Bioinformatics*, Vol. 6(Suppl 1), p.S9.

33. Rebholz-Schuhmann, D., Kirsch, H. and Couto, F. (2005), 'Facts from text – is text mining ready to deliver?', *PLoS Biol.*, Vol. 3(2), p. e65.

34. URL: http://www.cs.waikato.ac.nz/ml/weka/

35. Joachims, T. (1998), 'Text categorization with support vector machines: Learning with many relevant features', in 'Proceedings of the 10th European Conference on Machine Learning', 21st–23rd April, Chemnitz, Germany, pp. 137–114.

36. Fan, R.-E., Chen, P.-H. and Lin, C.-J. (2005), 'Working set selection using the second order information for training SVM', technical report, Department of Computer Science, National Taiwan University.

37. URL: http://www.run.montefiore.ulg.ac.be/~francois/software/jahmm/

38. Daraselia, N., Yuryev, A., Egorov, S. *et al.* (2004), 'Extracting human protein interactions from MEDLINE using a full-sentence parser', *Bioinformatics*, Vol. 20(5), pp. 604–611.

39. Brill, E. (1992), 'A simple rule-based part of speech tagger', in 'Proceedings of the Conference on Applied Natural Language Processing (ANLP92)', Trento, Italy, pp. 152–155.

40. Brants, T. (2000), 'TnT – a statistical part-of-speech tagger', in 'Proceedings of the Conference on Applied Natural Language Processing (ANLP00)', 29th April–4th May, Seattle, WA.

41. Schmid, H. (1995), 'Improvements in part-of-speech tagging with an application to German', in 'Proceedings of the ACL SIGDAT-Workshop', Dublin, Ireland, pp. 47–50.

42. Zhou, G., Zhang, J., Su, J. *et al.* (2004), 'Recognizing names in biomedical texts: A machine learning approach', *Bioinformatics*, Vol. 20(7), pp. 1178–1190.

43. Clegg, A. B. and Sheperd, A. (2005), 'Evaluating and integrating treebank parsers on a biomedical corpus', in 'Proceedings of the Workshop on Software at the 43rd Annual Meeting of the of the Association for Computational Linguistics', 25th–30 June, Ann Arbor, MI.

44. Smith, L., Rindflesch, T. and Wilbur, W. J. (2004), 'MedPost: A part-of-speech tagger for biomedical text', *Bioinformatics*, Vol. 20(14), pp. 2320–2321.

45. Hirschman, L., Yeh, A., Blaschke, C. and Valencia, A. (2005), 'Overview of BioCreAtIvE: critical assessment of information extraction for biology', *BMC Bioinformatics*, Vol. 6 (Suppl 1), p. S1.

46. Kim, J. D., Ohta, T., Tateisi, Y. and Tsujii, J. (2003), 'GENIA corpus – a semantically annotated corpus for bio-textmining', *Bioinformatics*, Vol. 19 (Suppl 1), pp. I180–I182.

47. Rosario, B. and Hearst, M. A. (2004), 'Classifying semantic relations in bioscience text', in 'Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL 2004)', 21st–26 July, Barcelona, Spain.

48. Hakenberg, J., Rutsch, J. and Leser, U. (2005), 'Tuning text classification for hereditary diseases with section weighting', in 'Proceedings of the Symposium on Semantic Mining in Biomedicine (SMBM)', 10–13th April, Hinxton, UK, pp. 34–39.

49. Shah, P. K., Perez-Iratxeta, C., Bork, P. and Andrade, M. A. (2003), 'Information extraction from full text scientific articles: where are the keywords?', *BMC Bioinformatics*, Vol. 4(1), p. 20.

50. Bairoch, A., Apweiler, R., Wu, C. H. *et al.* (2005), 'The universal protein resource (UniProt)', *Nucleic Acids Res*., Vol. 33 (Database issue), pp. D154–159.

51. Fleischmann, A., Darsow, M., Degtyarenko, K. *et al.* (2004), 'IntEnz, the integrated relational enzyme database', *Nucleic Acids Res*., Vol. 32 (Database issue), pp. D434–437.

52. Medical Subject Headings, National Library of Medicine, NIH (URL: http://www.nlm.nih.gov/mesh/).

53. Hanisch, D., Fluck, J., Mevissen, H.-T. and Zimmer, R. (2003), 'Playing biology's name game: Identifying protein names in scientific text', in 'Proceedings of the 8th Pacific Symposium on Biocomputing', 3rd–7th January, Hawaii.

54. Altschul, S. F., Madden, T. L., Schaffer, A. A. *et al.* (1997), 'Gapped BLAST and PSI-BLAST: A new generation of protein database search programs', *Nucleic Acids Res*., Vol. 25(17), pp. 3389–3402.

55. Krauthammer, M., Rzhetsky, A., Morozov, P. and Friedman, C. (2000), 'Using BLAST for identifying gene and protein names in journal articles', *Gene*, Vol. 259(1–2), pp. 245–252.

56. Tomohiro Mitsumori, T., Fation, S., Murata, M. *et al.* (2005), 'Gene/protein name recognition based on support vector machine using dictionary as features', *BMC Bioinformatics*, Vol. 6 (Suppl 1), pp.S8.

57. Curran, J. R. and Clark, S. (2003), 'Language independent NER using a maximum entropy tagger', in 'Proceeding of the 7th Conference on Natural Language Learning', 31st May–1st June, Edmonton, Canada, pp. 164–167.

58. Kinoshita, S., Cohen, K. B., Ogren, P. V. and Hunter, L. (2005), 'BioCreAtIvE Task1A: Entity identification with a stochastic tagger', *BMC Bioinformatics*, Vol. 6(Suppl 1), p. S4.

59. McDonald, R. and Pereira, F. (2005), 'Identifying gene and protein mentions in text using conditional random fields', *BMC Bioinformatics*, Vol. 6 (Suppl 1), p. S6.

60. Zhou, G., Shen, D., Zhang, J. *et al.* (2004), 'Recognition of protein/gene names from text using an ensemble of classifiers and effective abbreviation resolution', in 'Proceedings of the EMBO workshop BioCreative: Critical Assessment for Information Extraction in Biology', 28th–31st March, Granada, Spain.

61. Mika, S. and Rost, B. (2004), 'Protein names precisely peeled off free text', *Bioinformatics*, Vol. 20 (Suppl 1), pp. I241–I247.

62. Mons, B. (2005), 'Which gene did you mean?', *BMC Bioinformatics*, Vol. 6, p. 142.