Chapter 8

Biomedical Named Entity Recognition: A Survey of Machine-Learning Tools

David Campos, Sérgio Matos and José Luís Oliveira

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/51066

1. Introduction

It is well known that the rapid growth and dissemination of the Internet has resulted in huge amounts of information generated and shared, available in the form of textual data, images, videos or sounds. This overwhelming surge of data is also true for specific areas such as biomedicine, where the number of published documents, such as articles, books and technical reports, is increasing exponentially. For instance, the MEDLINE literature database contains over 20 million references to journal papers, covering a wide range of biomedical fields. In order to organize and manage these data, several manual curation efforts have been set up to identify, in texts, information regarding entities (e.g. genes and proteins) and their relations (e.g. protein-protein interactions). The extracted information is stored in structured knowledge resources, such as Swiss-Prot [1] and GenBank [2]. However, the effort required to continually update these databases makes this a very demanding and expensive task, naturally leading to increasing interest in the application of Text Mining (TM) systems to help perform those tasks.

One major focus of TM research has been on Named Entity Recognition (NER), a crucial initial step in information extraction, aimed at identifying chunks of text that refer to specific entities of interest, such as gene, protein, drug and disease names. Such systems can be integrated in larger biomedical Information Extraction (IE) pipelines, which may use the automatically extracted names to perform other tasks, such as relation extraction, classification or/and topic modeling. However, biomedical names have various characteristics that may difficult their recognition in texts [3]:

- Many entity names are descriptive (e.g. “normal thymic epithelial cells”);

- Two or more entity names sharing one head noun (e.g. “91 and 84 kDa proteins” refers to “91 kDa protein” and “84 kDa protein”);
- One entity name with several spelling forms (e.g. “N-acetylcysteine”, “N-acetyl-cysteine”, and “NAcetylCysteine”);
- Ambiguous abbreviations are frequently used (e.g. “TCF” may refer to “T cell factor” or to “Tissue Culture Fluid”).

Consequently, several NER systems have been developed for the biomedical domain, using different approaches and techniques that can generally be categorized as being based on rules, dictionary matching or Machine Learning (ML). Each approach fulfills different requirements, depending on the linguistic characteristics of the entities being identified. Such heterogeneity is a consequence of the predefined naming standards and how faithfully the biomedical community followed them. Thus, it is recommended to take advantage of the approaches that better fulfill the requirements of each entity type:

- Rule-based: names with a strongly defined orthographic and morphological structure;
- Dictionary-based: closely defined vocabulary of names (e.g. diseases and species);
- ML-based: strong variability and highly dynamic vocabulary of names (e.g. genes and proteins).

Applying the best approaches is not possible in all cases, since each approach presents different technical requirements [4]. However, when the appropriate resources are available, ML-based solutions present several advantages over other methods, and provide the best performance results.

The development of ML-based NER solutions integrates various complex steps that incorporate different processing pipelines. Thus, along the past years, a variety of systems were developed using the most different frameworks, techniques and strategies. This chapter gives an overview of ML-based biomedical NER solutions, providing a brief description of the latest and most significant research techniques, and presenting an in-depth analysis of the available systems and frameworks, considering the technical characteristics, provided features and performance outcomes. In the end, future directions and research opportunities on ML-based NER solutions are discussed.

2. Methods

ML-based solutions use statistical models focused on recognizing specific entity names, using a feature-based representation of the observed data. Such approach solves various problems of rule and dictionary-based solutions, recognizing new entity names and new spelling variations of an entity name. However, ML does not provide identifiers from curated resources, which can be solved by using a dictionary in an extra step. Nonetheless, the main drawback of such solutions is the dependency on annotated documents, which are hard and

http://dx.doi.org/10.5772/51066

expensive to obtain. Thus, the absence of such resource for a specific biomedical entity type may limit the applicability of ML solutions.

The development of ML-based solutions requires two essential steps (Figure 1): train and annotate. At first, the ML model must be trained using the annotations present on the annotated documents. This step can take some time depending on the complexity of the model and on the available computational resources. After storing the model in a physical resource, raw documents can be annotated, providing entity names based on the past experience inferred from the annotated documents.

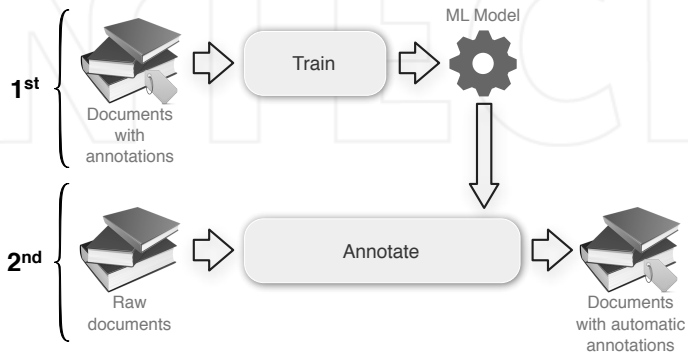


Figure 1. Illustration of the development process of ML-based solutions.

Both training and annotation tasksdepend on various processing steps and resources. Figure 2 presents the pipeline of the required modules to implement ML-based NER solutions:

- Corpora: collection of texts related with the target domain;
- Pre-processing: process the input data in order to simplify the recognition process;
- Feature processing: extract, select and/or induce features from the pre-processed input data;
- ML model: use the generated features to automatically define a set of rules that describe and distinguish the characteristics and patterns ofentity names;
- Post-processing: refinement of the generated annotations, solving problems of the recognition process or extending recognized names;
- Output: input corpora with automatically generated annotations or the extracted information ina structured format.

Each module must perform one or various essential tasks. Moreover, each task can be performed using different algorithms or/and resources, depending on the target goal and predefined requirements. The following sub-sections present the main goals of each module and briefly describe alternative approaches to fulfill the requirements of each task.

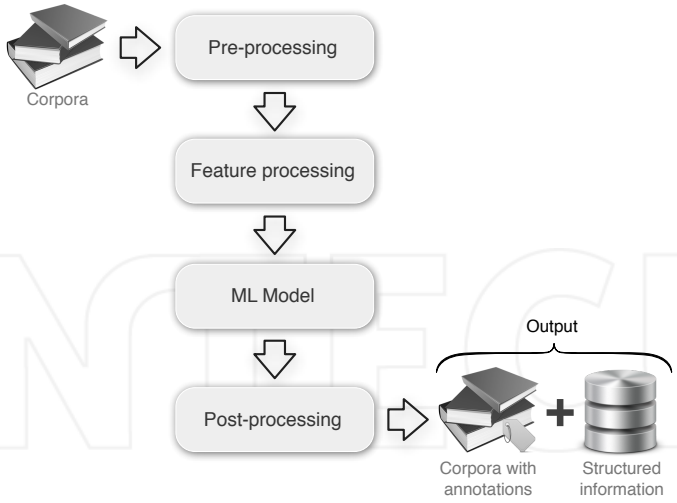


Figure 2. Overall pipeline of the required steps to develop ML-based NER solutions.

2.1. Corpora

In this context, a corpus is a set of text documents that usually contain annotations of one or various entity types. Such annotations are used to train ML models, inferring characteristics and patterns of the annotated entity names. Thus, the trained model is highly dependent on the quality of the annotations present on those corpora. This dependency must be carefully analyzed, since a corpus may contain annotations of a specific entity type but not reflecting the whole spectrum of names. A corpus is also used to obtain performance results, in order to understand the behavior of the system on real-life problems. Such evaluation enables the comparison of distinct solutions to the same problem.

There are two types of annotated corpora, varying with the source of the annotations:

- Gold Standard Corpora (GSC): annotations are performed manually by expert annotators, following specific and detailed guidelines;
- Silver Standard Corpora (SSC): annotations are automatically generated by computerized systems.

Table 1 presents a list of relevant GSC available for the various biomedical entity types. As we can see, most of the research efforts have been on the recognition of gene and protein names, with various corpora containing several thousands of annotated sentences. Such effort is a consequence of two different factors: the importance of genes and proteins on the biomedical domain, and the high variability and no standardization of names. Various challenges were organized for the recognition of gene and protein names, such as BioCreative [5] and JNLPBA [6], and most advances on ML-based NER were achieved in those challenges.

When the amount of annotated documents is not sufficient to reflect the whole spectrum of names, corpora are mostly used for evaluation procedures, as is generally the case for the identification of disorders and species names. The SCAI IUPAC corpus is also a good example of a specific sub-entity type corpus, containing only annotations of chemicals that follow the IUPAC nomenclature. Finally, both AnEM and CellFinder are very recent corpora (released on 2012), showing that the development of manually annotated corpora for the various entity types is still an ongoing work.

Entity	Corpus	Type	Size (sentences)
Gene and Protein	GENETAG [7]	Sentences	20000
	JNLPBA [6] (from GENIA [8])	Abstracts	22402
	FSUPRGE [9]	Abstracts	≈29447*
	PennBioIE [10]	Abstracts	≈22877*
Species	OrganismTagger Corpus [11]	Full texts	9863
	Linnaeus Corpus [12]	Full texts	19491
Disorders	SCAI Disease [13]	Abstracts	≈3640*
	EBI Disease [14]	Sentences	600
	Arizona Disease (AZDC) [15]	Sentences	2500
	BioText [16]	Abstracts	3655
Chemical	SCAI IUPAC [17]	Sentences	20300
	SCAI General [18]	Sentences	914
Anatomy	AnEM[1]	Sentences	4700
Miscellaneous	CellFinder[2]	Full texts	2100

Table 1. List of relevant Gold Standard Corpora (GSC) available for each biomedical entity, presenting the type of documents and its size.
*Each MEDLINE abstract contains on average 7.2±1.9 sentences [19]. We considered the best-case scenario with ≈9 sentences.

As we can see on Table 1, only small sets of documents have been annotated, due to the complexity of generating GSC. The CALBC [20] (Collaborative Annotation of a Large Biomedical Corpus) project aimedto minimize this problem, providing a large-scale biomedical SSC automatically annotated through the harmonization of several NER systems. This large corpus contains one million abstracts with annotations of several biological semantic groups, such as diseases, species, chemicals and genes/proteins.

1 http://www.nactem.ac.uk/anatomy

2 http://www.informatik.hu-berlin.de/forschung/gebiete/wbi/resources/cellfinder

2.2. Pre-processing

Natural Language Processing (NLP) solutions can be accomplished by computerized systems in an effective manner. However, it is necessary firstly to properly delimit the documents into meaningful units. Most NLP solutions expect their input to be segmented into sentences, and each sentence into tokens. Since real-world documents lack such well-defined structure, it is necessary to implement a few methods to perform such tasks.

2.2.1. Sentence splitting

Sentence splitting is the process of breaking a text document into its respective sentences. In the end, each sentence should provide a specific local, logical and meaningful context for future tasks. Various solutions were developed to perform sentence splitting on biomedical documents, such as JSBD [21], OpenNLP[3] and SPECIALIST NLP[4].The best performing solutions can achieve an accuracy of 99.7%.

2.2.2. Tokenization

Tokenisation is the process of breaking a sentence into its constituent meaningful units, called tokens. It is one of the most important tasks of the IE workflow, since all the following tasks will be based on the tokens resulting from this process. Consequently, various tools were developed specifically for the biomedical domain, such as GENIA Tagger [22], JTBD [21] and SPECIALIST NLP. In [23], the authors present a detailed comparison of various biomedical tokenizers. The best solutions achieve accuracies around 96%.

2.2.3. Annotationencoding

In order to internally represent the annotated entity names, it is necessary to use an encoding scheme to give a tag to each token of the text. The simplest is the IO encoding, which tags each token as either being in (tag "I") a particular named entity or outside (tag "O"). This encoding is defective, since it cannot represent two entities next to each other. The BIO encoding is the *de facto* standard, and it extends the IO encoding solving the boundary problem. In this scheme, the "in" tag is subdivided into tag "B", representing the first token or beginning of the entity name, and tag "I" for the remaining tokens. The BMEWO encoding extends the BIO encoding by distinguishing the end of an entity (tag "E") tokens from the middle entity tokens (tag "M"), and adding a new tag ("W") for entities with only one token.

2.3. Feature processing

Feature processing is a crucial NER task, since the predictions will be performed based on the information that they encode, reflecting special phenomena and linguistic characteristics of the naming conventions. Thus, the definition of a rich and carefully selected set of features is required in order to properly represent the target entity names.

3http://opennlp.apache.org

4http://lexsrv3.nlm.nih.gov/Specialist

http://dx.doi.org/10.5772/51066

2.3.1. Linguistic

The most basic internal feature is the token itself. However, in most cases, morphological variants of words have similar semantic interpretations, and can be considered as equivalent. For this reason, stemming or lemmatization can be used to group together all inflected forms of a word, so that they can be analyzed as a single item. The basic idea of stemming is to find the prefix that is common to all variations of the term. On the other hand, lemmatization is a more robust method, because it finds the root term of the variant word (e.g. the lemma of "was" is "be"). Along with normalization techniques, it is also possible to associate each token with a particular grammatical category based on its context, a procedure called Part-of-Speech (POS) tagging. Additionally, chunking can be also used, dividing the text into syntactically correlated parts of words (e.g., noun or verb phrases). These linguistic features only provide a local analysis of the token in the sentence. To complement this, features can be derived from dependency parsing tools to collect the relations between the various tokens in the sentence.

2.3.2. Orthographic

The purpose of orthographic features is to capture knowledge about word formation. For example, a word that starts with a capital letter could indicate the occurrence of an entity name (e.g. in the protein name "MyoD"). Various features can be used, reflecting the presence of uppercase or lowercase characters, the presence of symbols, or counting the number of digits and uppercase characters in a token.

2.3.3. Morphological

Morphological features, on the other hand, reflect common structures and/or sub-sequences of characters among several entity names, thus identifying similarities between distinct tokens. To accomplish this goal, three distinct types of morphological features are commonly considered:

- Suffixes and prefixes:can be used to distinguish entity names. For instance, suffixes like "ase", "ome" and "gen" frequently occur in gene and protein names;
- Char n-grams: are sub-sequences of *n*characters from a given token. It extends suffixes and prefixes by considering sub-sequences of characters in the middle of tokens;
- Word shape patterns: generate a sequence of characters to reflect how letters, digits and symbols are organized in the token. For instance, "Abc: 1234" could be represented by the following patterns: "Abc: *", "Aaa#1111" and/or "a#1".

2.3.4. Context

Higher-level relations between tokens and extracted features can be established through windows or conjunctions of features, reflecting the local context of each token. The application of windows consists of adding features of preceding and succeeding tokens as features of each token. On the other hand, conjunction of features consists of creating new features

by grouping together features of the surrounding tokens. To apply those context methods, it is important to limit the features to use as context information, since using the complete set of features from the surrounding tokens wouldgenerate millions of new features. However, pre-selecting the features used forbuilding the conjunctions may mean that informative conjunctions are not considered. Feature induction solves this problem, by iteratively considering sets of atomic and conjunction features created from the initial feature set. On each iteration, only candidates that provide useful information are included in the updated set of features. Intuitively, features with high gain provide strong evidence for many decisions.

2.3.5. Lexicons

Adding biomedical knowledge to the set of features can further optimize NER systems. To provide this knowledge, dictionaries of specific domain terms and entity names are matched in the text and the resulting tags are used as features. Two different types of dictionaries are commonly used: target entity names (match tokens with dictionaries with a complete set of names of the target entity name), and trigger names (match names that may indicate the presence of biomedical names in the surrounding tokens).

2.4. Machine learning model

As the input to the ML model, each feature should assume the value "1" if it is present on the current token or "0" if it is not (Table 2).

	Feature 1	Feature 2	…	Feature m
Token 1	1	1	…	0
Token 2	0	1	…	0
…	…	…	…	…
Token n	0	0	…	1

Table 2. Illustration of the matrix of features as the input tothe ML model. Each vector defines the features present for the corresponding token.

Each modeling technique uses the feature matrix to create a probabilistic description of the entity names boundaries. The various ML models can be classified as being supervised or semi-supervised, depending on unannotated data being used or not. Supervised learning, which only uses annotated data, has received most research interest in recent years. Consequently, different supervised models have been used on NER systems, namely Conditional Random Fields (CRFs) [24], Support Vector Machines (SVMs) [25], Hidden Markov Models (HMMs) [26] and Maximum Entropy Markov Models (MEMMs) [27]. CRFs have been actively used during the last years, since they present several advantages over other methods. Firstly, CRFs avoid the label bias problem [24], a weakness of MEMMs. In addition, CRFs also have advantages over HMMs, a consequence of their conditional nature that results in relaxation of the independence assumptions [28]. Finally, although SVMs can provide com-

http://dx.doi.org/10.5772/51066

parable results, more time is required to train complex models.Semi-supervised solutions use both annotated and unannotated data, in order to solve the data sparseness problem. Thus, the main goal is to collect features of the unannotated data that are not present in the annotated data, which may contribute to a better identification of the entity names boundaries. There are various approaches to implement semi-supervised solutions, such as Semi-CRFs [29, 30], Semi-SVMs [31], ASO [32] and FCG [33].

2.5. Model Combination

The most recent results on biomedical NER clearly indicate that better performance results can be achieved by combining several systems with different characteristics. As an example, the top five systems of the BioCreative II gene mention challenge [5] used ensembles of NER systems. In order to generate ML models that reflect different characteristics of the annotated data, it is common to use different parsing directions (forward and backward) or different feature sets. Moreover, different approaches can be used to combine the generated annotations, using union, intersection, machine learning [34] or lexicons [35].

2.6. Post-processing

Post-processing techniques are commonly used to solve some recognition mistakes, which may be easily corrected through simple rules or methods:

- Remove or correct recognition mistakes: annotations with an odd number of brackets may be removed or corrected.
- Extend or make annotations more precise: abbreviation resolution methods can be used to extend detected annotations. Moreover, curated dictionaries can be also used to correct generated annotations.
- Remove uninformative terms: some annotations may be known for being non-informative or unwanted terms, and consequently must be removed.

2.7. Evaluation

In order to understand the behavior of the system, it is important to measure the accuracy of the generated annotations. This can be performed by annotating a corpus and then compare the automatic annotations with the ones provided by expert curators. Thus, each automatic annotation must be classified as being a:

- True Positive (TP): the system provides an annotation that exists in the curated corpus;
- True Negative (TN): the non existence of an annotation is correct according to the curated corpus;
- False Positive (FP): the system provides an annotation that does not exist in the curated corpus;
- False Negative (FN): the system does not provide an annotation that is present in the curated corpus.

Exact and fuzzy matching can be used to obtain performance results and to better understand the behavior of the system. With approximate matching we can find the performance when minor and non-informative mistakes are discarded. Such evaluation is important since various post-NER tasks, such as relation extraction and topic modeling, can be performed with imprecise annotations.

Performance results are obtained using three important measures: precision, recall and F-measure. Those measures assume values between 0 (worst) and 1 (best). Precision measures the ability of a system to present only relevant names, and it is formulated as:

$$Precision = \frac{relevant\ names\ recognized}{total\ names\ recognized} = \frac{TP}{TP + FP} \quad (1)$$

On the other hand, recall measures the ability of a system to present all relevant names, and is formulated as:

$$Recall = \frac{relevant\ names\ recognized}{relevant\ names\ on\ corpus} = \frac{TP}{TP + FN} \quad (2)$$

Finally, F-measure is the harmonic mean of precision and recall. The balanced F-measure is most commonly used, and is formulated as:

$$F\text{ - }measure = 2\ \frac{Precision \times Recall}{Precision + Recall} \quad (3)$$

3. Tools

In order to understand and expose the current trends of ML-based NER solutions, it is important to study existing tools and respective characteristics, analyzing their applicability on real life problems. Since dozens of tools are available for the recognition of a specific entity type (e.g. gene and protein), we decided to study the systems that better reflect the overall progress of the domain. On the other hand, some entity types do not have any relevant ML-based systems. For instance, species recognition is already efficiently performed by dictionary-based solutions: LINNAEUS [12] and OrganismTagger [11] already achieve ≈95% and ≈97% of F-measure, respectively. Curiously, OrganismTagger uses a simple ML module that gives a small contribution in the recognition of species strains. Also there are no ML solu-

tions for the recognition of anatomy names, since the AnEM corpus was just released (May, 2012). However, due to the characteristics of the corpus and the importance of anatomy entity type in the biomedical domain, we believe that various solutions will be developed soon.

Table 3 presents an overview of the ML-based systems characteristics that we considered for the various entity types, presenting the used programming languages, features, models and post-processing techniques. The tools presented in the miscellaneous section can be applied to more than one entity type, namely on gene/protein and disorders. However, different entity types have different characteristics, requiring the applicability of distinct techniques and/or feature sets. As we can see, features such as chunking, dependency parsing and conjunctions are only used in gene and protein names recognition, which require a much more complex feature set. Moreover, the BioEnEx authors show that the use of some morphological and orthographic features has a negative impact on the recognition of disorder names [36].

The application of complex techniques is a reflex of the research effort spent on each entity type. As expected, gene and protein names have been the main research target, with eight systems. Four of those systems, including AIIAGMT and IBM Watson, were developed specifically for these entity types. The other four systems were also adapted for the recognition of disorders, such as ABNER and BANNER. It is also interesting to see that the systems developed for the recognition of chemicals are not used in any other entity type or vice-versa. We believe that this is a consequence of various factors: available corpora do not cover the whole spectrum of chemical names or is sparse; and chemical names present different challenges when compared with gene and protein names, namely high variability but different levels of names standardization.

We also studied the various tools in terms of source code availability, since using and adapting already implemented methods may streamline the development of new and improved IE solutions. For instance, one can use Gimli or BANNER to implement new tools for the recognition of different entity types (e.g., BANNER was used in the development of BioEnEx). It is also interesting to see that closed source solutions commonly present more advanced and/or complex techniques. For instance, only IBM Watson applies a semi-supervised algorithm and, with the exception of Gimli, only closed source solutions use model combination strategies.

Regarding used features, we can observe that some features are common to every recognition task, namely some orthographic, morphological and context features. Thus, we may argue that those features are essential in the development of ML-based NER tools. One can also observe that linguistic features are only used in the recognition of entity names that present high variability and low standardization. Moreover, from the results of various tools we can conclude that the use of dictionary matching as features always presents a positive contribution, since adding domain knowledge provides an increased precision.

			Gene and Protein				Chemical		Miscellaneous (gene/protein and disorders)			
Name			**Gimli**[5]	**NERSuite**[6]	**IBM Watson**	**AIIAGMT**	**SCAI**	**ChemSpot**	**BioEnEx**	**BANNER**	**ABNER**	**Lingpipe**
Year			2011	2010	2007	2008	2008	2012	2010	2008	2005	2007
Reference			-	-	[37]	[35]	[17]	[38]	[36, 39]	[40]	[41]	[42]
Programming Language			Java	C++	-	Java, C++	Java	Java	Java	Java	Java	Java
Open source			X	X				X		X	X	X
Features	**Linguistic**	Normalization	X	X		X			X	X		
		POS	X	X	X	X			X	X		
		Chunking	X	X	X							
		Dependency	X						X			
	Orthographic	Capitalization	X	X	X	X	X	X	X	X	X	
		Counting	X	X		X				X		
		Symbols	X	X	X	X	X	X	X	X	X	
	Morphological	Suffix and Prefix	X		X		X	X	X	X	X	
		Char n-grams	X	X		X			X	X		
		WordShape	X	X		X				X	X	
	Context	Windows				X	X	X	X	X	X	
		Conjunctions	X	X	X							
	Lexicons	Target names	X	X	X				X	X	X	
		Trigger names	X			X					X	
Model	**Supervised**		CRF	CRF		CRF	CRF	CRF	CRF	CRF	CRF	HMM
	Semi-supervised				ASO							
Model Combination			X		X	X						
Post-processing	**Parentheses**		X		X	X			X	X		
	Abbreviation		X						X	X		
	Lexicons					X						

Table 3. Overview of the ML-based systems characteristics considering the various target entity types and presenting the used programming languages, features, models and post-processing techniques.

5 http://bioinformatics.ua.pt/gimli

6 http://nersuite.nlplab.org

http://dx.doi.org/10.5772/51066

As stated above, nine of the ten tools use supervised models. From those, all the systems developed in the last four years (from 2008 to 2012) take advantage of CRFs, which shows the success of this modeling technique. However, there is a growing research interest in the application of semi-supervised models, since they may provide more general and non-corpus specific solutions.

Finally, post-processing methods are commonly applied by closed source solutions, with the exception of Gimli and BANNER that already take advantage of several high-end techniques. Thus, we can argue that parentheses processing and abbreviation resolution are essential tasks, since its applicability is independent of the entity type.

Besides the study of the characteristics of each tool, we have also conducted a set of evaluation experiments to help elucidating about the solutions that perform the best. Figure 3 presents a performance comparison of the analyzed tools per entity type.

3.1. Gene and Protein names

Most of the developed solutions are focused on two main corpora, GENETAG and JNLPBA. GENETAG is not restricted to a specific domain, containing annotations of proteins, DNA and RNA (grouped in only one semantic type), which were performed by experts in biochemistry, genetics and molecular biology. This corpus was used in the BioCreative II challenge [5], and it contains 15000 sentences for training and 5000 sentences for testing. For evaluation, the matching is performed allowing alternative names provided by the expert annotators. On the other hand, the JNLPBA corpus is a sub-set of the GENIA corpus, containing 2404 abstracts extracted from MEDLINE using the MeSH terms "human", "blood-cell" and "transcription factor". The manual annotation of these abstracts was based on five classes of the GENIA ontology, namely protein, DNA, RNA, cell line, and cell type. This corpus was used in the BioEntity Recognition Task in BioNLP/NLPBA 2004 [6], providing 2000 abstracts for training and the remaining 404 abstracts for testing. On this challenge, the evaluation was performed using exact matching. Since GENETAG is not focused on any specific biomedical domain, its annotations are more heterogeneous than those of JNLPBA. A brief analysis, considering protein, DNA and RNA classes, shows that GENETAG contains almost 65% of distinct names, as opposed to the 36% found on JNLPBA.

As expected, a model trained on GENETAG provides annotations not focused on any specific biomedical domain, which may be recommended for general real life applications. However, the same semantic group contains annotations of DNA, RNA and gene/protein. On the other hand, a model trained on the JNLPBA corpus may provide annotations optimized for research on human blood cell transcription factors. On this corpus, the various entity types are split on different semantic groups.

Overall, the systems present high performance results on both corpora, where the best performing systems achieve an F-measure of 88.30% on GENETAG and 73.05% on JNLPBA. Both systems, AIIAGMT and Gimli, present complex solutions that include the application

of linguistic and lexicon features and the combination of various CRF models with different characteristics. When comparing open with closed source solutions, there is no significant difference, since both apply high-end techniques and high performance results. Moreover, Gimli is the open-source solution that provides more implemented features,with the best performance results. Nevertheless, BANNER, BioEnEx and NER Suite also present positive outcomes using simpler models and no combination techniques.

The high performance results achieved on this task, on both general and specific corpora, indicate that the recognition of gene and protein names is ready to be used on real life problems, presenting an acceptable error margin.

3.2. Disorder names

Following the UMLS description of disorder, it includes various semantic sub-groups with different interpretations. Thus, a corpus must contain names of diseases, symptoms, abnormalities and dysfunctions in order to reflect the whole spectrum of disorder names in the best way as possible.Some of those sub-groups are important since new diseases are often referred as a set of signs and symptoms until the disease receives an official name.

Only two corpora were used for the development of ML solutions: Arizona Disease Corpus (AZDC) and BioText. AZDC presents a set of annotations that reflect the whole spectrum of disorder names. The annotation process was performed manually by one expert and revised by another. On the other hand, the BioText corpus was annotated for the identification of disease and treatment mentions, not covering the whole spectrum of names. Moreover, the primary goal of this corpus was to explore different types of relationships between diseases and treatments. Thus, a high degree of annotation consistency was not required at the token level. Regarding both corpora, we consider that the amount of annotated sentences (2500 and 3655) may not be sufficient to train an accurate ML model for real life applications. For instance, the AZDC corpus provides 3228 disease mentions. In comparison, GENETAG contains ≈24000 primary gene and protein annotations and more than 17000 alternative mentions for approximate matching.

Due to the restrictions of the available corpora, most solutions for the recognition of disorders are typically implemented through the application of dictionary-based approaches. Nevertheless, various solutions were developed using only ML and the existing corpora. Overall, those solutions present a simpler feature set when comparing with gene and protein models.

As expected, a model trained on the AZDC corpus may provide general annotations considering the whole spectrum of disorder names in one semantic group. On the other hand, a model trained on the BioText corpus only provides annotations of diseases and treatments. However, on both cases, a large amount of disorder names are missed since their statistical description is not present on the training corpus.

Considering exact matching, the best system achieves an F-measure of 81.08% on the AZDC corpus, which is a good performance result. The improved performance of BioEnEx in comparison to BANNER may be justified by the use of dependency parsing and the absence of some orthographic features. On the other hand, the BioText inconsistencies cause the performance of systems to be overly pessimistic. As we can see, the best performing system only achieves an F-measure of 54.84%.

In summary, we believe that the AZDC corpus provides a good benchmark for the recognition of disorder names. However, the implementation of systems based only on that specific corpus is doubtful, due to the small amount of annotated documents. Moreover, we also believe that the BioText corpus is not the best solution for systems comparison, due to the reported inconsistences and achieved results.

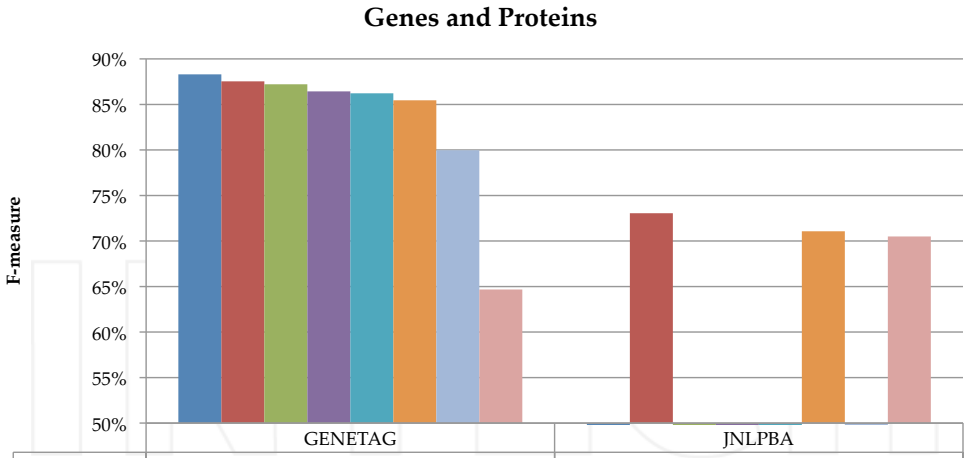
3.3. Chemical names

Chemical names can be divided into two classes [38]: a closed (finite) class for trivial names, and an open (infinite) class for names following strictly defined nomenclatures (e.g. IUPAC names). The SCAI General corpus contains manual annotations of both classes. However, we believe that 914 sentences with only 1206 mentions is not sufficient for development and evaluation procedures, due to the amount and complexity of chemical names. On the other hand, the SCAI IUPAC corpus only contains annotations of chemical names that follow the IUPAC nomenclature. In this case, we believe that 20300 sentences are sufficient for ML models to infer the patterns of the rule-based convention with success.

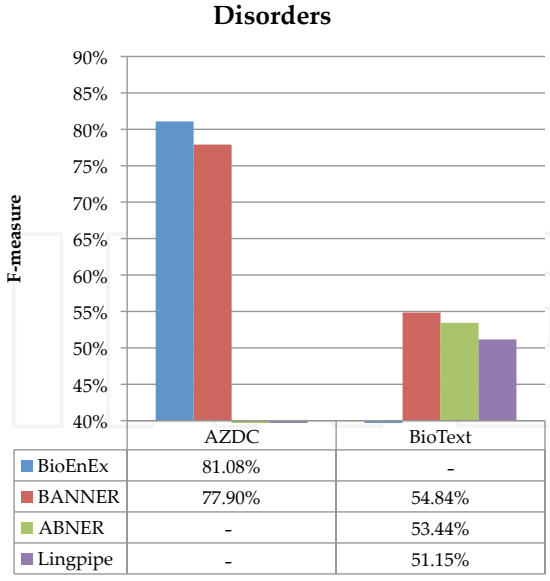
As expected, the systems developed using the IUPAC corpus deliver good results. The best performing solution achieves 85.60% of F-measure using exact matching. As we can see, ChemSpot and SCAI have similar characteristics in terms of model and features. However, the achieved results are quite different. Such difference may be related with specific characteristics of the CRF training. The authors of ChemSpot argue that SCAI uses a third-order CRF (instead of second-order) with optimized meta-parameters, which may over fit the model and consequently provide worse performance on unseen instances.

The same model of ChemSpot that was trained on the IUPAC corpus was tested on the SCAI corpus. As expected, it provides low recall results, since only IUPAC annotations are generated. Overall, it presents an F-measure of 42.60%, which is a good result considering that IUPAC annotations represent around 32% of all names present on the SCAI corpus.

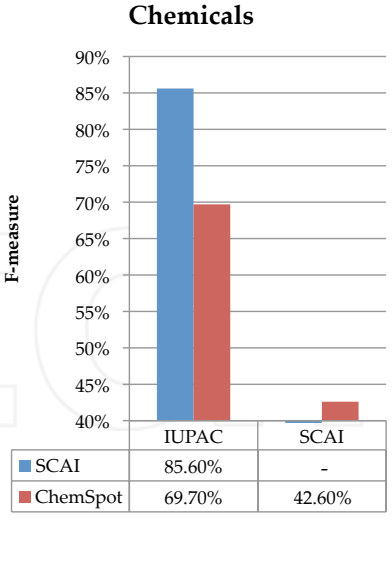
Since the systems for IUPAC entity names provide positive outcomes, we believe that an optimal solution for the recognition of chemical names must be hybrid, combining ML and dictionary-based solutions. Thus, ML may be used for IUPAC names and the dictionary matching for trivial names and drugs. Actually, ChemSpot does exactly that, and achieved an F-measure of 68.10% on the SCAI corpus, presenting an improvement of ≈11% against previous solutions.



	GENETAG	JNLPBA
AIIAGMT	88.30%	-
Gimli	87.54%	73.05%
IBM Watson	87.21%	-
BANNER	86.43%	-
BioEnEx	86.22%	-
NERSuite	85.45%	71.07%
Lingpipe	79.97%	-
ABNER	64.68%	70.50%



	AZDC	BioText
BioEnEx	81.08%	-
BANNER	77.90%	54.84%
ABNER	-	53.44%
Lingpipe	-	51.15%



	IUPAC	SCAI
SCAI	85.60%	-
ChemSpot	69.70%	42.60%

Figure 3. Performance comparison of the various ML-based NER solutions per entity type and corpus.

http://dx.doi.org/10.5772/51066

4. Conclusion

This chapter presented a detailed survey of machine learning tools for biomedical named entity recognition. We started by introducing the various fundamental steps for the development of such tools. Afterwards, we described each step in detail, presenting the required methods and alternative techniques used by the various solutions. Using that knowledge, we presented various tools for each biomedical entity, describing the used methodologies and provided features. Thus, solutions for recognition of gene/protein, disorder and chemical names were studied in detail, exposing the main differences between the various systems characteristics. Such analysis allowed us to expose the current trends of ML-based solutions for biomedical NER, and compare the performance outcomes considering the different systems' characteristics. Thus, we can summarize the current trends by task:

- Corpora: annotated abstracts are the most used corpus type;
- Pre-processing: sentence splitting, tokenization and annotation encoding are fundamental for input data processing;
- Features: most of orthographic, morphological, lexicon and context features are essential in the recognition of any biomedical entity type. Linguistic features and conjunctions present an important contribution in the recognition of non-standardized entity names;
- Model: supervised CRF models are widely used and present positive outcomes on all biomedical entity types;
- Post-processing: parentheses processing and abbreviation resolution are essential tasks and provided positive contributions on all entity types.

Overall, we can argue that the amount and quality of ML-based tools already provide a significant number of advanced features with good performances results. Such results show that most tools are ready to be used on real life applications, providing acceptable error margins.

Regarding future steps, we believe that using full text documents will be fundamental, since they provide more information and completely different challenges, due to the increased ambiguity. Moreover, the application of semi-supervised algorithms will take advantage of the millions of unannotated documents. Such strategy presents various advantages, contributing to the development of ML-based solutions for entity types that have a reduced amount of annotated corpora, and allowing the development of general models, independent of the training corpora and ready to annotate any text with high accuracy. Finally, we also believe that feature induction will be essential, enabling automatic generation of informative features to extract new and unknown characteristics of entity names.

5. Nomenclature

CRFs - Conditional Random Fields
HMMs - Hidden Markov Models
IE - Information Extraction
MEMMs - Maximum Entropy Markov Models
ML - Machine Learning
NLP - Natural Language Processing
POS - Part-of-Speech
SVMs - Support Vector Machines

Acknowledgements

This work received funding from FEDER through the COMPETE programme and from Fundação para a Ciência e a Tecnologia (FCT) under grant agreement FCOMP-01-0124-FEDER-010029. S. Matos is funded by FCT under the Ciência2007 programme.

Author details

David Campos*, Sérgio Matos and José Luís Oliveira

*Address all correspondence to: david.campos@ua.pt

IEETA/DETI, University of Aveiro, Portugal

References

[1] Bairoch, A., & Boeckmann, B. (1992). The SWISS-PROT protein sequence data bank. *Nucleic acids research*, 20, 2019-2022.

[2] Benson, D. A., Karsch-Mizrachi, I., Clark, K., Lipman, D. J., Ostell, J., & Sayers, E. W. (2012). GenBank. *Nucleic acids research*, 40, D48-53.

[3] Zhou, G., Zhang, J., Su, J., Shen, D., & Tan, C. (2004). Recognizing names in biomedical texts: a machine learning approach. *Bioinformatics*, 20, 1178-1190.

[4] Campos, D., Matos, S., & Oliveira, J. L. (2012). Current methodologies for biomedical Named Entity Recognition. *In Biological Knowledge Discovery Handbook: Preprocessing, Mining and Postprocessing of Biological Data (to appear)*, Edited by Elloumi M, Zomaya AY: John Wiley & Sons, Inc.

http://dx.doi.org/10.5772/51066

[5] Smith, L., Tanabe, L. K., Ando, R. J., Kuo, C. J., Chung, I. F., Hsu, C. N., Lin, Y. S., Klinger, R., Friedrich, C. M., Ganchev, K., et al. (2008). Overview of BioCreative II gene mention recognition. *Genome biology*, 9(2), S2.

[6] Kim, J. D., Ohta, T., Tsuruoka, Y., Tateisi, Y., & Collier, N. (2004). Introduction to the bio-entity recognition task at JNLPBA. *In International Joint Workshop on Natural Language Processing in Biomedicine and its Application*, Geneva, Switzerland. Association for Computational Linguistics: 70-75.

[7] Tanabe, L., Xie, N., Thom, L. H., Matten, W., & Wilbur, W. J. (2005). GENETAG: a tagged corpus for gene/protein named entity recognition. *BMC bioinformatics*, 6(1), S3.

[8] Kim, J. D., Ohta, T., Tateisi, Y., & Tsujii, J. (2003). GENIA corpus-semantically annotated corpus for bio-textmining. *Bioinformatics*, 19(1), i180-182.

[9] Hahn, U., Beisswanger, E., Buyko, E., Poprat, M., Tomanek, K., & Wermter, J. (2008). Semantic Annotations for Biology-A Corpus Development Initiative at the Jena University Language & Information Engineering (JULIE) Lab. *In Proceedings of the 6th International Conference on Language Resources and Evaluation*, 28-30.

[10] Kulick, S., Bies, A., Liberman, M., Mandel, M., Mc Donald, R., Palmer, M., Schein, A., Ungar, L., Winters, S., & White, P. (2004). Integrated annotation for biomedical information extraction. *In Workshop on Linking Biological Literature, Ontologies and Databases (Human Language Technology conference)*, Association for Computational Linguistics, 61-68.

[11] Naderi, N., Kappler, T., Baker, C. J., & Witte, R. (2011). OrganismTagger: detection, normalization and grounding of organism entities in biomedical documents. *Bioinformatics*, 27, 2721-2729.

[12] Gerner, M., Nenadic, G., & Bergman, C. M. (2010). LINNAEUS: a species name identification system for biomedical literature. *BMC bioinformatics*, 11, 85.

[13] Gurulingappa, H., Klinger, R., Hofmann-Apitius, M., & Fluck, J. (2010). An Empirical Evaluation of Resources for the Identification of Diseases and Adverse Effects in Biomedical Literature. *In 2nd Workshop on Building and evaluating resources for biomedical text mining*, (7th edition of the Language Resources and Evaluation Conference); Valetta, Malta, 15.

[14] Jimeno, A., Jimenez-Ruiz, E., Lee, V., Gaudan, S., Berlanga, R., & Rebholz-Schuhmann, D. (2008). Assessment of disease named entity recognition on a corpus of annotated sentences. *BMC bioinformatics*, 9(3), S3.

[15] Leaman, R., Miller, C., & Gonzalez, G. (2009). Enabling recognition of diseases in biomedical text with machine learning: Corpus and benchmark. *In 3rd International Symposium on Languages in Biology and Medicine*, Jeju Island, South Korea, 82-89.

[16] Rosario, B., & Hearst, M. A. (2004). Classifying semantic relations in bioscience texts. *In 42nd annual meeting of the Association for Computational Linguistics*, Barcelona, Spain. Association for Computational Linguistics, 430.

[17] Klinger, R., Kolarik, C., Fluck, J., Hofmann-Apitius, M., & Friedrich, C. M. (2008). Detection of IUPAC and IUPAC-like chemical names. *Bioinformatics*, 24, i268-276.

[18] Kolárik, C., Klinger, R., Friedrich, C. M., Hofmann-Apitius, M., & Fluck, J. (2008). Chemical names: terminological resources and corpora annotation. *In Workshop on Building and evaluating resources for biomedical text mining (Language Resources and Evaluation Conference)*, 51-58.

[19] Yu, H. (2006). Towards answering biological questions with experimental evidence: automatically identifying text that summarize image content in full-text articles. *In Proceedings of the AMIA Annual Symposium*, American Medical Informatics Association, 834-838.

[20] Rebholz-Schuhmann, D., Yepes, A. J., Van Mulligen, E. M., Kang, N., Kors, J., Milward, D., Corbett, P., Buyko, E., Beisswanger, E., & Hahn, U. (2010). CALBC silver standard corpus. *Journal of bioinformatics and computational biology*, 8, 163-179.

[21] Tomanek, K., Wermter, J., & Hahn, U. (2007). A reappraisal of sentence and token splitting for life sciences documents. *Studies in health technology and informatics*, 129, 524-528.

[22] Tsuruoka, Y., Tateishi, Y., Kim, J. D., Ohta, T., McNaught, J., Ananiadou, S., & Tsujii, J. (2005). Developing a robust part-of-speech tagger for biomedical text. *Advances in informatics*, 382-392.

[23] He, Y., & Kayaalp, M. (2006). A Comparison of 13 Tokenizers on MEDLINE. Bethesda, MD: The Lister Hill National Center for Biomedical Communications.

[24] Lafferty, J., McCallum, A., & Pereira, F. C. N. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. *In International Conference on Machine Learning*, Williamstown, Massachusetts, USA. Morgan Kaufmann.

[25] Cherkassky, V. (1997). The nature of statistical learning theory. IEEE transactions on neural networks / a publication of the IEEE Neural Networks Council , 8, 1564.

[26] Baum, L. E., & Petrie, T. (1966). Statistical inference for probabilistic functions of finite state Markov chains. *The Annals of Mathematical Statistics*, 37, 1554-1563.

[27] McCallum, A., Freitag, D., & Pereira, F. (2000). Maximum entropy Markov models for information extraction and segmentation. *In International Conference on Machine Learning*, Stanford, California, USA. Morgan Kaufmann, 591-598.

[28] Wallach, H. M. (2004). Conditional random fields: An introduction. *University of Pennsylvania CIS Technical Report MS-CIS-04-21*.

[29] Mann, G. S., & Mc Callum, A. (2007). Efficient computation of entropy gradient for semi-supervised conditional random fields. *In Proceedings of the North American Association for Computational Linguistics*, Rochester, New York, USA. Association for Computational Linguistics, 109-112.

[30] Mann, G., & Mc Callum, A. (2008). Generalized expectation criteria for semi-supervised learning of conditional random fields. *In Proceedings of Association of Computational Linguistics*, Association of Computational Linguistics, 870-878.

[31] Bennett, K., & Demiriz, A. (1999). Semi-supervised support vector machines. *Advances in Neural Information processing systems*, 368-374.

[32] Ando, R. K., & Zhang, T. (2005). A framework for learning predictive structures from multiple tasks and unlabeled data. *The Journal of Machine Learning Research*, 6, 1817-1853.

[33] Li, Y., Hu, X., Lin, H., & Yang, Z. (2011). A Framework for Semisupervised Feature Generation and Its Applications in Biomedical Literature Mining. *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, 8, 294-307.

[34] Campos, D., Matos, S., Lewin, I., Oliveira, J. L., & Rebholz-Schuhmann, D. (2012). Harmonisation of gene/protein annotations: towards a gold standard MEDLINE. *Bioinformatics*, 28, 1253-1261.

[35] Hsu, C. N., Chang, Y. M., Kuo, C. J., Lin, Y. S., Huang, H. S., & Chung, I. F. (2008). Integrating high dimensional bi-directional parsing models for gene mention tagging. *Bioinformatics*, 24, i286-294.

[36] Chowdhury, M., & Faisal, M. (2010). Disease mention recognition with specific features. *In Proceedings of Association for Computational Linguistics*, Association for Computational Linguistics, 83-90.

[37] Ando, R. K. (2007). BioCreative II gene mention tagging system at IBM Watson. *In Proceedings of the Second BioCreative Challenge Evaluation Workshop*, Madrid, Spain, 101-103.

[38] Rocktaschel, T., Weidlich, M., & Leser, U. (2012). ChemSpot: A Hybrid System for Chemical Named Entity Recognition. *Bioinformatics*.

[39] Chowdhury, F. M., & Lavelli, A. (2011). Assessing the practical usability of an automatically annotated corpus. *In Proceedings of the Fifth Linguistic Annotation Workshop*, Portland, Oregon, USA. Association for Computational Linguistics, 101-109.

[40] Leaman, R., & Gonzalez, G. (2008). BANNER: an executable survey of advances in biomedical named entity recognition. *Pacific Symposium on Biocomputing Pacific Symposium on Biocomputing*, 652-663.

[41] Settles, B. (2005). ABNER: an open source tool for automatically tagging genes, proteins and other entity names in text. *Bioinformatics*, 21, 3191-3192.

[42] Carpenter, B. (2007). LingPipe for 99.99% recall of gene mentions. *In Proceedings of the Second BioCreative Workshop*, Madrid, Spain, 307-309.

INTECH