Evaluating and combining biomedical named entity recognition systems

Andreas Vlachos
William Gates Building
Computer Laboratory
University of Cambridge
av308@cl.cam.ac.uk

Abstract

This paper is concerned with the evaluation of biomedical named entity recognition systems. We compare two such systems, one based on a Hidden Markov Model and one based on Conditional Random Fields and syntactic parsing. In our experiments we used automatically generated data as well as manually annotated material, including a new dataset which consists of biomedical full papers. Through our evaluation, we assess the strengths and weaknesses of the systems tested, as well as the datasets themselves in terms of the challenges they present to the systems.

1 Introduction

The domain of biomedical text mining has become of importance for the natural language processing (NLP) community. While there is a lot of textual information available in the domain, either in the form of publications or in model organism databases, there is paucity in material annotated explicitly for the purpose of developing NLP systems. Most of the existing systems have been developed using data from the newswire domain. Therefore, the biomedical domain is an appropriate platform to evaluate existing systems in terms of their portability and adaptability. Also, it motivates the development of new systems, as well as methods for developing systems with these aspects in focus in addition to the performance.

The biomedical named entity recognition (NER) task in particular has attracted a lot of attention from the community recently. There have been three shared tasks (BioNLP/NLPBA 2004 (Kim et al., 2004), BioCreative (Blaschke et al., 2004) and BioCreative2 (Krallinger and Hirschman, 2007)) which involved some flavour of NER using manually annotated training material and fully supervised machine learning methods. In parallel, there have been successful efforts in bootstrapping NER systems using automatically generated training material using domain resources (Morgan et al., 2004; Vlachos et al., 2006). These approaches have a significant appeal, since they don't require manual annotation of training material which is an expensive and lengthy process.

Named entity recognition is an important task because it is a prerequisite to other more complex ones. Examples include anaphora resolution (Gasperin, 2006) and gene normalization (Hirschman et al., 2005). An important point is that until now NER systems have been evaluated on abstracts, or on sentences selected from abstracts. However, NER systems will be applied to full papers, either on their own or in order to support more complex tasks. Full papers though are expected to present additional challenges to the systems than the abstracts, so it is important to evaluate on the former as well in order to obtain a clearer picture of the systems and the task (Ananiadou and McNaught, 2006).

In this paper, we compare two NER systems in a variety of settings. Most notably, we use automatically generated training data and we evaluate on abstracts as well as a new dataset consisting of full papers. To our knowledge, this is the first evaluation of biomedical NER on full paper text instead of

BioNLP 2007: Biological, translational, and clinical language processing, pages 199–200,
Prague, June 2007. ©2007 Association for Computational Linguistics

abstracts. We assess the performance and the portability of the systems and using this evaluation we combine them in order to take advantage of their strengths.

2 Named entity recognition systems

This section presents the two biomedical named entity recognition systems used in the experiments of Section 4. Both systems have been used successfully for this task and are domain-independent, i.e. they don't use features or resources that are tailored to the biomedical domain.

2.1 Hidden Markov Model

The first system used in our experiments was the HMM-based (Rabiner, 1990) named entity recognition module of the open-source NLP toolkit LingPipe[1]. It is a hybrid first/second order HMM model using Witten-Bell smoothing (Witten and Bell, 1991). It estimates the following joint probability of the current token x_t and label y_t conditioned on the previous label y_{t-1} and previous two tokens x_{t-1} and x_{t-2}:

$$P(x_t, y_t | y_{t-1}, x_{t-1}, x_{t-2}) \tag{1}$$

Tokens unseen in the training data are passed to a morphological rule-based classifier which assigns them to predefined classes according to their capitalization and whether they contain digits or punctuation. In order to use these classes along with the ordinary tokens, during training a second pass over the training data is performed in which tokens that appear fewer times than a given threshold are replaced by their respective classes. In our experiments, this threshold was set experimentally to 8. Vlachos et al. (2006) employed this system and achieved good results on bootstrapping biomedical named entity recognition. They also note though that due to its reliance on seen tokens and the restricted way in which unseen tokens are handled its performance is not as good on unseen data.

2.2 Conditional Random Fields with Syntactic Parsing

The second NER system we used in our experiments was the system of Vlachos (2007) that participated in the BioCreative2 Gene Mention task (Krallinger and Hirschman, 2007). Its main components are the Conditional Random Fields toolkit MALLET[2] (McCallum, 2002) and the RASP syntactic parsing toolkit[3] (Briscoe et al., 2006), which are both publicly available.

Conditional Random Fields (CRFs) (Lafferty et al., 2001) are undirected graphical models trained to maximize the conditional probability of the output sequence given the inputs, or, in the case of token-based natural language processing tasks, the conditional probability of the sequence of labels y given a sequence of tokens x. Like HMMs, the number of previous labels taken into account defines the order of the CRF model. More formally:

$$P(y|x) = \frac{1}{Z(x)} exp\{\sum_{t=1}^{T}\sum_{k=1}^{K} \lambda_k f_k(y, x_t)\} \tag{2}$$

In the equation above, $Z(x)$ is a normalization factor computed over all possible label sequences, f_k is a feature function and λ_k its respective weight. y represents the labels taken into account as context and it is defined by the order of the CRF. For a n-th order model, y becomes $y_t, y_{t-1}..., y_{t-n}$. It is also worth noting that x_t is the feature representation of the token in position t, which can include features extracted by taking the whole input sequence into account, not just the token in question. The main advantage is that as a conditionally-trained model CRFs do not need to take into account dependencies in input, which as a consequence, allows the use of features dependent on each other. Compared to HMMs, their main disadvantage is that during training, the computation time required is significantly longer. The interested reader is referred to the detailed tutorial of Sutton & McCallum (2006).

Vlachos (2007) used a second order CRF model combined with a variety of features. These can be divided into simple orthographic features and in

[1] http://www.alias-i.com/lingpipe. The version used in the experiments was 2.1.

[2] http://mallet.cs.umass.edu/index.php/Main_Page

[3] http://www.informatics.susx.ac.uk/research/nlp/rasp/

those extracted from the output of the syntactic parsing toolkit. The former are extracted for every token and they are rather common in the NER literature. They include the token itself, whether it contains digits, letters or punctuation, information about capitalization, prefixes and suffixes.

The second type of features are extracted from the output of RASP for each sentence. The part-of-speech (POS) tagger was parameterized to generate multiple POS tags for each token in order to ameliorate unseen token errors. The syntactic parser uses these sequences of POS tags to generate parses for each sentence. The output is in the form of grammatical relations (GRs), which specify the links between the tokens in the sentence accoring to the syntactic parser and they are encoded using the SciXML format (Copestake et al., 2006). From this output, for each token the following features are extracted (if possible):

- the lemma and the POS tag(s) associated with the token
- the lemmas for the previous two and the following two tokens
- the lemmas of the verbs to which this token is subject
- the lemmas of the verbs to which this token is object
- the lemmas of the nouns to which this token acts as modifier
- the lemmas of the modifiers of this token

Adding the features from the output of the syntactic parser allows the incorporation of features from a wider context than the two tokens before and after captured by the lemmas, since GRs can link tokens within a sentence independently of their proximity. Also, they result in more specific features, since the relation between two tokens is determined. The CRF models in the experiments of Section 4 were trained until convergence.

It must be mentioned that syntactic parsing is a complicated task and therefore feature extraction on its output is likely to introduce some noise. The RASP syntactic parser is domain independent but it has been developed using data from general English corpora mainly, so it is likely not to perform as well in the biomedical domain. Nevertheless, the results of the system in the BioCreative2 Gene Mention task suggest that the use of syntactic parsing features improve performance. Also, despite the lack of domain-specific features, the system is competitive with other systems, having performance in the second quartile of the task. Finally, the BIOEW scheme (Siefkes, 2006) was used to tag the tokenized corpora, under which the first token of a multitoken mention is tagged as B, the last token as E, the inner ones as I, single token mentions as W and tokens outside an entity as O.

3 Corpora

In our experiments we used two corpora consisting of abstracts and one consisting of full papers. One of the abstracts corpora was automatically generated while the other two were manually annotated. All three were created using resources from FlyBase[4] and they are publicly available[5].

The automatically generated corpus was created in order to bootstrap a gene name recognizer in Vlachos & Gasperin (2006). The approach used was introduced by Morgan et al (2004). In brief, the abstracts of 16,609 articles curated by FlyBase were retrieved and tokenized by RASP (Briscoe et al., 2006). For each article, the gene names and their synonyms that were recorded by the curators were annotated automatically in its abstract using longest-extent pattern matching. The pattern matching is flexible in order to accommodate capitalization and punctuation variations. This process resulted in a large but noisy dataset, consisting of 2,923,199 tokens and containing 117,279 gene names, 16,944 of which are unique. The noise is due to two reasons mainly. First, the lists constructed by the curators for each paper are incomplete in two ways. They don’t necessarily contain all the genes mentioned in an abstract because not all genes are always curated and also not all synonyms are recorded, thus resulting in false negatives. The other cause is the overlap between gene names and common English words or biomedical terms, which results in false positives for

[4] http://www.flybase.net/

[5] http://www.cl.cam.ac.uk/ nk304/Project_Index/#resources

abstracts with such gene names.

The manually annotated corpus of abstracts was described in Vlachos & Gasperin (2006). It consists of 82 FlyBase abstracts that were annotated by a computational linguist and a FlyBase curator. The full paper corpus was described in Gasperin et al. (2007). It consists of 5 publicly available full papers which were annotated by a computational linguist and a FlyBase curator with named entities as well as anaphoric relations in XML. To use it for the gene name recognition experiments presented in this paper, we converted it from XML to IOB format keeping only the annotated gene names.

	noisy abstracts	golden abstracts	full papers
abstracts / papers	16,609	82	5
sentences	111,820	600	1,220
tokens	2,923,199	15,703	34,383
gene names	117,279	629	2,057
unique gene names	16,944	326	336
unique non-gene tokens	60,943	3,018	4,113

Table 1: Statistics of the datasets

The gene names in both manually created corpora were annotated using the guidelines presented in Vlachos & Gasperin (2006). The main idea of these guidelines is that gene names are annotated anywhere they are encountered in the text, even when they are used to refer to biomedical entities other than the gene itself. The distinction between the possible types of entities the gene name can refer to is performed at the level of the shortest noun phrase surrounding the gene name. This resulted in improved inter-annotator agreement (Vlachos et al., 2006).

Statistics on all three corpora are presented in Table 1. From the comparisons in this table, an interesting observation is that the gene names in full papers tend to be repeated more frequently than the gene names in the manually annotated abstracts (6.1 compared to 1.9 times respectively). Also, the latter contain approximately 2 unique gene names every 100 tokens while the full papers contain just 1. This evidence suggests that annotating abstracts is more likely to provide us with a greater variety of gene names. Interestingly, the automatically annotated abstracts contain only 0.6 unique gene names every 100 tokens which hints at inclusion of false negatives during the annotation.

Another observation is that, while the manually annotated abstracts and full papers contain roughly the same number of unique genes, the full papers contain 36% more unique tokens that are not part of a gene name (“unique non-gene tokens” in Table 1). This suggests that the full papers contain a greater variety of contexts, as well as negative examples, therefore presenting greater difficultiy to a gene name recognizer.

4 Experiments

We ran experiments using the two NER systems and the three datasets described in Sections 2 and 3. In order to evaluate the performance of the systems, apart from the standard recall, precision and F-score metrics, we measured the performance on seen and unseen gene names independently, as suggested by Vlachos & Gasperin (2006). In brief, the gene names that are in the test set and the output generated by the system are separated according to whether they have been encountered in the training data as gene names. Then, the standard recall, precision and F-score metrics are calculated for each of these lists independently.

		HMM	CRF+RASP
overall	Recall	75.68	63.43
	Precision	89.14	90.89
	F-score	81.86	74.72
seen genes	Recall	94.48	76.32
	Precision	93.62	95.4
	F-score	94.05	84.80
unseen genes	Recall	33.51	34.54
	Precision	68.42	73.63
	F-score	44.98	47.02
seen genes		435	
unseen genes		194	

Table 2: Results on training on noisy abstracts and testing on manually annotated abstracts

		HMM	CRF+RASP
overall	Recall	58.63	61.40
	Precision	80.56	89.19
	F-score	67.87	72.73
seen genes	Recall	89.82	72.51
	Precision	87.83	94.82
	F-score	88.81	82.18
unseen genes	Recall	35.12	53.03
	Precision	69.48	84.05
	F-score	46.66	65.03
seen genes		884	
unseen genes		1173	

Table 3: Results on training on noisy abstracts and testing on full papers

Tables 2 and 3 report in detail the performance of the two systems when trained on the noisy abstracts and evaluated on the manually annotated abstracts and full papers respectively. As it can be seen, the performance of the HMM-based NER system is better than that of CRF+RASP when evaluating on abstracts and worse when evaluating on full papers (81.86 vs 74.72 and 67.87 vs 72.73 respectively).

Further analysis of the performance of the two systems on seen and unseen genes reveals that this result is more likely to be due to the differences between the two evaluation datasets and in particular the balance between seen and unseen genes with respect to the training data used. In both evaluations, the performance of the HMM-based NER system is superior on seen genes while the CRF+RASP system performs better on unseen genes. On the abstracts corpus the performance on seen genes becomes more important since there are more seen than unseen genes in the evaluation, while the opposite is the case for the full paper corpus.

The difference in the performance of the two systems is justified. The CRF+RASP system uses a complex but more general representation of the context based on the features extracted from the output of syntactic parser, namely the lemmas, the part-of-speech tags and the grammatical relationships, while the HMM-based system uses a simple morphological rule-based classifier. Also, the CRF+RASP system takes the two previous labels into account, while the HMM-based only the previous one. Therefore, it is expected that the former has superior performance on unseen genes. This difference between the CRF+RASP and the HMM-based system is substantially larger when evaluating on full papers (65.03 versus 46.66 respectively) than on abstracts (47.02 versus 44.98 respectively). This can be attributed to the fact that the training data used is generated from abstracts and when evaluating on full papers the domain shift can be handled more efficiently by the CRF+RASP system due to its more complex feature set.

However, the increased complexity of the CRF+RASP system renders it more vulnerable to noise. This is particularly important in these experiments because we are aware that our training dataset contains noise since it was automatically generated. This noise is in addition to that from inaccurate syntactic parsing employed, as explained in Section 2.2. On the other hand, the simpler HMM-based system is likely to perform better on seen genes, whose recognition doesn't require complex features.

We also ran experiments using the manually annotated corpus of abstracts as training data and evaluated on the full papers. The results in Table 4 confirmed the previous assessment, that the performance of the CRF+RASP system is better on the unseen genes and that the HMM-based one is better on seen genes. In this particular evaluation, the small number of unique genes in the manually annotated corpus of abstracts results in the majority of gene names being unseen in the training data, which favors the CRF+RASP system.

It is important to note though that the performances for both systems were substantially lower than the ones achieved using the large and noisy automatically generated corpus of abstracts. This can be attributed to the fact that both systems have better performance in recognizing seen gene names rather than unseen ones. Given that the automatically generated corpus required no manual annotation and very little effort compared to the manually annotated one, it is a strong argument for bootstrapping techniques.

A known way of reducing the effect of noise in sequential models such as CRFs is to reduce their order. However, this limits the context taken into account, potentially harming the performance on unseen gene names. Keeping the same feature set, we

		HMM	CRF+RASP
overall	Recall	52.65	49.88
	Precision	46.56	72.77
	F-score	49.42	59.19
seen genes	Recall	96.49	47.37
	Precision	58.51	55.1
	F-score	72.85	50.94
unseen genes	Recall	51.4	49.95
	Precision	46.04	73.4
	F-score	48.57	59.45
seen genes		57	
unseen genes		2000	

Table 4: Results on training on manually annotated abstracts and testing on full papers

trained a first order CRF model on the noisy abstracts corpus and we evaluated on the manually annotated abstracts and full papers. As expected, the performance on the seen gene names improved but deteriorated on the unseen ones. In particular, when evaluating on abstracts the F-scores achieved were 93.22 and 38.1 respectively (compared to 84.8 and 47.02) and on full papers 86.64 and 59.86 (compared to 82.18 and 65.03). The overall performance improved substantially for the abstract where the seen genes are the majority (74.72 to 80.69), but only marginally for the more balanced full papers (72.73 to 72.89).

Ideally, we wouldn't want to sacrifice the performance on unseen genes of the CRF+RASP system in order to deal with noise. While the large noisy training dataset provides good coverage of the possible gene names, it is unlikely to contain every gene name we would encounter, as well as all the possible common English words which can become precision errors. Therefore we attempted to combine the two NER systems based on the evaluation presented earlier. Since the HMM-based system is performing very well on seen gene names, for each sentence we check whether it has recognized any gene names unseen in the training data (potential unseen precision errors) or if it considered as ordinary English words any tokens not seen as such in the training data (potential unseen recall errors). If either of these is true, then we pass the sentence to the CRF+RASP system, which has better performance on unseen gene names.

Such a strategy is expected to trade some of the performance of the seen gene names of the HMM-based system for improved performance on the unseen gene names by using the predictions of the CRF+RASP system. This occurs because in the same sentence seen and unseen gene names may coexist and choosing the predictions of the latter system could result in more errors on the seen gene names. This strategy is likely to improve the performance on datasets where there are more unseen gene names and the difference in the performance of the CRF+RASP on them is substantially better than the HMM-based. Indeed, using this strategy we achieved 73.95 overall F-score on the full paper corpus which contains slightly more unseen gene names (57% of the total gene names). For the corpus of manually annotated abstracts the performance was reduced to 80.21, which is expected since the majority of gene names (69%) are seen in the training data. and the performance of the CRF+RASP system on the unseen data is better only by a small margin than the HMM-based one (47.02 vs 44.98 in F-score respectively).

5 Discussion - Related work

The experiments of the previous section are to our knowledge the first to evaluate biomedical named entity recognition on full papers. Furthermore, we consider that using abstracts as the training material for such evaluation is a very realistic scenario, since abstracts are generally publicly available and therefore easy to share and distribute with a trainable system, while full papers on which they are usually applied are not always available.

Differences between abstracts and full papers can be important when deciding what kind of material to annotate for a certain purpose. For example, if the annotated material is going to be used as training data and given that higher coverage of gene names in the training data is beneficial, then it might be preferable to annotate abstracts because they contain greater variety of gene names which would result in higher coverage in the dataset. On the other hand, full papers contain a greater variety of contexts which can be useful for training a system and as mentioned earlier, they can be more appropriate for evaluation.

It would be of interest to train NER systems on training material generated from full papers. Considering the effort required in manual annotation though, it would be difficult to obtain quantities of such material large enough that would provide adequate coverage of a variety of gene names. An alternative would be to generate it automatically. However, the approach employed to generate the noisy abstracts corpus used in this paper is unlikely to provide us with material of adequate quality to train a gene name recognizer. This is because more noise is going to be introduced, since full papers are likely to contain more gene names not recorded by the curators, as well as more common English words that happen to overlap with the genes mentioned in the paper.

The aim of this paper is not about deciding on which of the two models is better but about how the datasets used affect the evaluation and how to combine the strengths of the models based on the analysis performed. In this spirit, we didn't attempt any of the improvements discussed by Vlachos & Gasperin (2006) because they were based on observations on the behavior of the HMM-based system. From the analysis presented earlier, the CRF+RASP system behaves differently and therefore it's not certain that those strategies would be equally beneficial to it.

As mentioned in the introduction, there has been a lot of work on biomedical NER, either through shared tasks or independent efforts. Of particular interest is the work of Morgan et al (2004) who bootstrapped an HMM-based gene name recognizer using FlyBase resources and evaluate on abstracts. Also of interest is the system presented by Settles (2004) which used CRFs with rich feature sets and suggested that one could use features from syntactic parsing with this model given their flexibility. Direct comparisons with these works are not possible since different datasets were used.

Finaly, combining models has been a successful way of achieving good results, such as those of Florian et al. (2003) who had the top performance in the named entity recognition shared task of CoNLL 2003 (Tjong Kim Sang and De Meulder, 2003).

6 Conclusions- Future work

In this paper we compared two different named entity recognition systems on abstracts and full paper corpora using automatically generated training data. We demonstrated how the datasets affect the evaluation and how the two systems can be combined. Also, our experiments showed that bootstrapping using automatically annotated abstracts can be efficient even when evaluating on full papers.

As future work, it would be of interest to develop an efficient way to generate data automatically from full papers which could improve the results further. An interesting approach would be to combine dictionary-based matching with an existing NER system in order to reduce the noise. Also, different ways of combining the two systems could be explored. With constrained conditional random fields (Kristjansson et al., 2004) the predictions of the HMM on seen gene names could be added as constraints to the inference performed by the CRF.

The good performance of bootstrapping gene name recognizers using automatically created training data suggests that it is a realistic alternative to fully supervised systems. The latter have benefited from a series of shared tasks that, by providing a testbed for evaluation, helped assessing and improving their performance. Given the variety of methods that are available for generating training data efficiently automatically using extant domain resources (Morgan et al., 2004) or semi-automatically (active learning approaches like Shen et al. (2004) or systems using seed rules such as Mikheev et al. (1999)), it would be of interest to have a shared task in which the participants would have access to evaluation data only and they would be invited to use such methods to develop their systems.

References

Sophia Ananiadou and John McNaught, editors. 2006. *Text Mining in Biology and Biomedicine*. Artech House, Inc.

Christian Blaschke, Lynette Hirschman, and Alexander Yeh, editors. 2004. *Proceedings of the BioCreative Workshop*, Granada, March.

Ted Briscoe, John Carroll, and Rebecca Watson. 2006. The second release of the RASP system. In *Proceed-*

ings of the COLING/ACL 2006 Interactive Presentation Sessions.

Ann Copestake, Peter Corbett, Peter Murray-Rust, CJ Rupp, Advaith Siddharthan, Simone Teufel, and Ben Waldron. 2006. An architecture for language processing for scientific texts. In *Proceedings of the UK e-Science All Hands Meeting 2006*.

Radu Florian, Abe Ittycheriah, Hongyan Jing, and Tong Zhang. 2003. Named entity recognition through classifier combination. In Walter Daelemans and Miles Osborne, editors, *Proceedings of CoNLL-2003*, pages 168–171. Edmonton, Canada.

C. Gasperin, N. Karamanis, and R. Seal. 2007. Annotation of anaphoric relations in biomedical full-text articles using a domain-relevant scheme. In *Proceedings of DAARC*.

Caroline Gasperin. 2006. Semi-supervised anaphora resolution in biomedical texts. In *Proceedings of BioNLP in HLT-NAACL*, pages 96–103.

Lynette Hirschman, Marc Colosimo, Alexander Morgan, and Alexander Yeh. 2005. Overview of biocreative task 1b: normalized gene lists. *BMC Bioinformatics*.

J. Kim, T. Ohta, Y. Tsuruoka, Y. Tateisi, and N. Collier, editors. 2004. *Proceedings of JNLPBA, Geneva*.

Martin Krallinger and Lynette Hirschman, editors. 2007. *Proceedings of the Second BioCreative Challenge Evaluation Workshop*, Madrid, April.

Trausti Kristjansson, Aron Culotta, Paul Viola, and Andrew McCallum. 2004. Interactive information extraction with constrained conditional random fields.

J. D. Lafferty, A. McCallum, and F. C. N. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of ICML 2001*, pages 282–289.

Andrew Kachites McCallum. 2002. Mallet: A machine learning for language toolkit. http://mallet.cs.umass.edu.

A. Mikheev, M. Moens, and C. Grover. 1999. Named entity recognition without gazetteers.

A. A. Morgan, L. Hirschman, M. Colosimo, A. S. Yeh, and J. B. Colombe. 2004. Gene name identification and normalization using a model organism database. *J. of Biomedical Informatics*, 37(6):396–410.

L. R. Rabiner. 1990. A tutorial on hidden markov models and selected apllications in speech recognition. In A. Waibel and K.-F. Lee, editors, *Readings in Speech Recognition*, pages 267–296. Kaufmann, San Mateo, CA.

Burr Settles. 2004. Biomedical Named Entity Recognition Using Conditional Random Fields and Novel Feature Sets. *Proceedings of the Joint Workshop on Natural Language Processing in Biomedicine and its Applications*.

D. Shen, J. Zhang, J. Su, G. Zhou, and C. L. Tan. 2004. Multi-criteria-based active learning for named entity recongition. In *Proceedings of ACL 2004*, Barcelona.

Christian Siefkes. 2006. A comparison of tagging strategies for statistical information extraction. In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, pages 149–152, New York City, USA, June. Association for Computational Linguistics.

Charles Sutton and Andrew McCallum. 2006. An introduction to conditional random fields for relational learning. In Lise Getoor and Ben Taskar, editors, *Introduction to Statistical Relational Learning*. MIT Press.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the conll-2003 shared task: Language-independent named entity recognition. In Walter Daelemans and Miles Osborne, editors, *Proceedings of CoNLL-2003*, pages 142–147. Edmonton, Canada.

A. Vlachos and C. Gasperin. 2006. Bootstrapping and evaluating named entity recognition in the biomedical domain. In *Proceedings of BioNLP in HLT-NAACL*, pages 138–145.

A. Vlachos, C. Gasperin, I. Lewin, and T. Briscoe. 2006. Bootstrapping the recognition and anaphoric linking of named entities in drosophila articles. In *Proceedings of PSB 2006*.

Andreas Vlachos. 2007. Tackling the BioCreative2 Gene Mention task with Conditional Random Fields and Syntactic Parsing. In *Proceedings of the Second BioCreative Challenge Evaluation Workshop*.

Ian H. Witten and Timothy C. Bell. 1991. The zero-frequency problem: Estimating the probabilities of novel events in adaptive text compression. *IEEE Transactions on Information Theory*, 37(4):1085–1094.