Named Entity Recognition using Machine Learning techniques

Nuno Miranda[1], Ricardo Raminhos[1], Pedro Seabra[1],
João Sequeira[2], Teresa Gonçalves[2], and Paulo Quaresma[2]

[1] VIATECLA SA, Portugal
nmiranda@viatecla.com, rraminhos@viatecla.com, pseabra@viatecla.com

[2] Universidade de Évora, Portugal
m5071@alunos.uevora.pt, tcg@uevora.pt, pq@uevora.pt

Abstract. Knowledge extraction through keywords and relation creation between contents with common keywords is an important asset in any content management system. Nevertheless, it is impossible to perform manually this kind of information extraction due to the growing amount of textual content of varying quality made available by multiple creators and distributors of information.
This paper presents and evaluates a prototype developed for the recognition of named entities using orthographic and morphologic word attributes as input and Support Vector Machines as the machine learning technique for identifying those entities in new documents.
Since documents are written in the Portuguese language and there was no part-of-speech tagger freely available, a model for this language was also developed using SVMTool, a simple and effective generator of sequential taggers based on Support Vector Machines. This implied adapting the Bosque 8.0 corpus by adding a POS tag to every word, since originally several words were joined into one token with a unique tag and others were split giving rise to more than one tag.

1 Introduction

With the increasing amount of information on the Web it is often difficult to navigate in an objective manner on relevant information. A way to facilitate this navigation is by extracting keywords and creating links between contents with common keywords. Nevertheless, being impossible to perform this task manually (for example, the daily on-line publications of several national newspapers gives rise to some hundreds of documents), the creation of a tool that can recognise useful keywords in textual content is an important asset in any content management.

This tool could be used as a standalone application or embedded as a helper in other applications that make use of keywords or relationships between them, allowing to:

– recognise entities and present them;

– sort the general subject of texts in accordance with relevant stakeholders;
– allow 'surfing' on a set of documents using the relationships created for common entities.

This work presents a tool for identifying useful keywords (named entities) and classifying them under the usual types of `Person`, `Organisation` and `Location` (the "enamex" types). The use of machine learning techniques enables the tool to be language and domain independent[3]. It presents the results obtained using news from the Portuguese daily newspapers OJE[4], Público[5] and Record[6].

The paper is organised as follows: Section 2 introduces the named entity recognition task and Section 3 shows the system's architecture. The part-of-speech tagger and the named entity recogniser are evaluated on Section 4 and Section 5, respectively. Finally, conclusions and future work are discussed in Section 6.

2 The named entity recognition task

Named Entity Recognition (NER) is an information extraction task which seeks to identify and classify elements in text which refer to predefined categories such as person names, organisations, places, time expressions, quantities and monetary values.

While early systems made use of hand-crafted rule-based algorithms [1][20], modern systems most often resort to machine learning techniques using supervised algorithms such as Hidden Markov Models [5], Decision Trees [3][26], Maximum Entropy Models [6], Conditional Random Fields [15] and Support Vector Machines [2][29].

Supervised algorithms use a generic inference procedure (usually referred to as learning) from which they build a classifier. During this phase, several texts are manually treated by identifying (and classifying) named entities from which the algorithm infers the characteristics that define those entities. In the later stage of classification, new texts are submitted to the classifier that will, hopefully, recognise entities within it.

Inference algorithms make use of features: characteristic attributes designed for algorithmic consumption; an example of a feature is a Boolean variable with the value *true* if a word is capitalised and *false* otherwise. The feature vector representation is an abstraction over text where typically each word is represented by one or many Boolean, numeric and nominal values.

Features most often used for the recognition and classification of named entities can be organised along three different axes [17]:

– word-level features

[3] The development was supported by the QREN TV.COMmunity project for the Alentejo region.
[4] `http://www.oje.pt/`
[5] `http://www.publico.pt/`
[6] `http://www.record.xl.pt/`

– list lookup features, and
– document and corpus features

Word-level features are related to the character make-up of words. They specifically could be used to describe word case, punctuation, numerical value and special characters; morphology and part-of-speech tags are also subcategories of common used word-level features. List inclusion is a way to express the relation "is a" (e.g., Paris is a city); general lists (e.g. dictionary, stop-words, common abbreviations), lists of entities and lists of entity cues (e.g. typical words in organisation and location, person title, name prefix) are commonly used lists for recognising enamex types. Document features are defined over both document content and document structure and include features that are beyond the single word and multi-word expression (multiple occurrences, local syntax) by possibly including meta-information about documents and corpus statistics.

A NER system is usually evaluated based on how its output compares with the output of human linguists. It can produce five different errors [17]:

– the system hypothesised an entity where there was none;
– an entity was completely missed by the system;
– the system noticed an entity but gave it the wrong label;
– the system noticed there is an entity but got its boundaries wrong;
– the system gave the wrong label to the entity and got its boundary wrong.

The score given to the system depends on how these errors are counted for. Different evaluation forums account differently. For example:

– in MUC [12] and HUB-4 [8], a system is scored on two axes: its ability to find the correct type and its ability to find exact text, keeping the number of correct answers, the number of actual system guesses and the number of possible entities in the solution for each type. The final score is the harmonic mean of precision and recall calculated over all entity slots on both axes (micro-average f-measure). This measure takes into account all 5 possible types of errors and gives partial credit for errors only occurring in one axis;
– IREX [27] and CONLL [23][24] share a simple scoring protocol: systems are compared based on the micro-averaged f-measure where a named entity is correct only if it is an exact match of the corresponding entity in the solution;
– ACE [10] has a complex evaluation procedure and includes mechanisms for dealing with various evaluation issues (partial match, wrong type, etc.). Basically, each entity type has a parametrized weight and contributes up to a maximal proportion of the final score (e.g., if each person is worth 1 point and each organisation is worth 0.5 point then it takes two organisations to counterbalance one person in the final score).

A good proportion of work in NER research is devoted to the study of the English language but a possibly larger proportion addresses language independence and multilingualism problems: Spanish and Dutch were used in CONLL'02 conference [23] while German and English were chosen for CONLL'03 [24]; Japanese

has been studied in MUC6 [12] and IREX [27] conferences. There are also works for a multitude of languages ranging from Korean [31] to Bulgarian [9] languages.

The Portuguese language is was also studied individually by Palmer [19] and through two evaluation contests: HAREM [25] and HAREM2 [16].

Nadeau and Sekine [17] present a very complete survey of named entity recognition systems as well as other critical aspects of NER such as features and evaluation methods.

3 System architecture

Our NER system consists of two components: a feature extractor and a classifier. The feature extractor converts text into a set of features that are then used by the classifier. The feature extractor reads the input text, creates tokens consisting of either words or selected punctuation marks and encodes each one using a set of 43 binary features – the feature vector; the classifier combines the sources of weak evidence about the candidate token and its context aiming to determine whether it is part of an entity or not. Next sub-sections detail each component.

3.1 Feature extractor

To determine whether a token is an entity, the system uses weak evidence from binary features: 40 word-level (25 orthographic and 15 morphologic) and 3 list-lookup ones. While orthographic attributes encode features like capitalisation and punctuation, morphologic ones indicate word's part-of-speech tag. These attributes are presented in Tables 1 and 2, respectively.

Table 1. Word orthographic features

'!'	'?'	all-uppercase	single-char
'('	')'	all-lowercase	alphanumeric
'"'	','	initial-caps	numeric
';'	':'	upper-lowercase	alphabetic
'+'	'-'	single-uppercase	hifen-word
','	'.'	'e' word ('and')	end-sentence-punct
'≪' or '≫'			

Table 2. Word morphologic features

common-name	determiner	conjunction	det-contraction
proper-name	adverb	prefix	adv-contraction
verb	pronoun	interjection	pron-contraction
adjective	proposition	punctuation	

List-lookup attributes were extracted from the following lists:

- *Label-Lex Portuguese dictionary* [13] (around 940000 words), a lexicon produced by the Language Engineering Laboratory of Instituto Superior Técnico. Since many names are not valid Portuguese words, this resource can identify some potential entities;
- *functional words* (24 words). These are lowercase words that appear inside multi-word entities;
- *rare words* (4995 words). These are words that appear in less than four text documents.

Capitalisation, in conjunction with reliable end-of-sentence boundary detection, is a good indicator for entities. However, determining sentence boundaries is difficult since common boundaries such as periods, question- and exclamation-marks can occur in many different contexts [21].

While the system does not explicitly contain rules for sentence boundary analysis, by using contextual cues, it can partly account for these boundaries. This is accomplished creating a context window that considers the feature vectors for the surrounding words. For example, a window of size 5 considers word-level and list-lookup attributes of five words: the word in question, two previous words and two subsequent ones.

3.2 Classifier

Since an entity may be composed of several words we used the IOB approach, normally used to mark multi-word phrases (see, for example, [23]): label `B` (*begin*) marks the first word and the remainder words are marked with the tag `I` (*inside*); words that do not belong to the expression are marked with the tag `O` (*outside*). Having three entity types (`Person`, `Organisation` and `Location`), this corresponds to define a multi-class classification problem with seven classes (`O` and for each type of entity, `B-xxx` and `I-xxx`).

Since classification is done word by word it is necessary to reconstruct entities after applying the learning algorithm. For that, we considered the most extensive possible entities (without overlap or inclusion of other entities).

For building the word classifier we used Support Vector Machines. This algorithm was chosen between decision trees and variants of the naïve Bayes algorithm by comparing their precision and recall values [22].

Support Vector Machines. Support Vector Machines was motivated by theoretical results from the statistical learning theory: it joins a kernel technique with the structural risk minimisation framework. *Kernel techniques* comprise two parts: a module that performs a mapping from the original data space into a suitable feature space and a learning algorithm designed to discover linear patterns in the (new) feature space. The *kernel function*, that implicitly performs the mapping, depends on the specific data type and domain knowledge of the particular data source. The *learning algorithm* is general purpose and robust.

It's also efficient, since the amount of computational resources required is polynomial with the size and number of data items, even when the dimension of the embedding space (the feature space) grows exponentially [28]. A mapping example is illustrated in Fig. 1a).

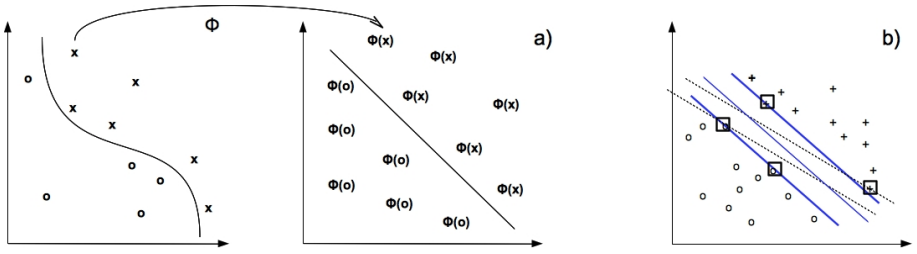

Fig. 1. The SVM approach: kernel transformation and search for maximum margin.

The *structural risk minimisation* (SRM) framework creates a model with a minimised VC (Vapnik-Chervonenkis) dimension [30]. This theory shows that when the VC dimension is low, the expected probability of error is also low, which means good performance on unseen data (good generalisation). In geometric terms, it can be seen as a search to find, between all decision surfaces that separate positive from negative examples, the one with maximum margin (the one having a separating property that is invariant to the most wide translation of the surface). This property is enlighten in Fig. 1b) for a 2-dimensional problem.

4 Part-of-speech tagger

Since documents were written in the Portuguese language and there was no part-of-speech tagger freely available, morphological features were obtained using SVMTool [11], a simple and effective generator of sequential taggers based on Support Vector Machines. Since there was no Portuguese language model available, it was necessary to develop one. Next sub-sections describe the corpus, the experimental setting and the results obtained.

4.1 Corpus

To accomplish a Portuguese POS model it was necessary to adapt the Bosque 8.0 corpus [14] by adding a POS tag for every word, since originally some words were joined into a single token with a unique tag (prepositional phrases) and others were split giving rise to more than one tag (determiner, adverb and pronoun contractions).

Bosque 8.0 is a collection of sentences parsed by the Palavras tool [4] and manually revised by linguists. It was developed by Linguateca (a resource centre for computational processing of Portuguese) and consists of 9368 sentences taken

from the first 1000 extracts of CETEMPúblico and CETENFolha corpora (news from Público and Folha de São Paulo[7], respectively).

CETENFolha sentences (written in Portuguese from Brazil) were withdrawn from the corpus and changes were made by attaching a POS label to every word, a characteristic that does not happen in the Bosque 8.0: contractions are expanded and words members of prepositional phrases are not classified. These changes forced some hand labelling.

Since the NER system makes use of a lexicon (*Label-Lex dictionary*), Bosque POS tags were translated the ones used in Label-Lex [13].

These changes gave rise to a corpus of 133670 words and 17 morphological tags (common-noun, proper-noun, verb, adjective, determiner, pronoun, adverb, preposition, interjection, conjunction, punctuation, prefix, prep-determiner, prep-pronoun, prep-adverb, not-pt and number). Table 3 shows the proportion of words for each major grammatical class.

Table 3. Proportion of words for each major grammatical classes

common-n	proper-n	verb	adjective	determiner	pronoun	adverb	preposition
18.4%	8.5%	12.2%	5.0%	7.6%	5.3%	4.6%	9.4%

4.2 Experimental setting

The tagger was obtained using SVMTool default values for all parameters. The corpus was split using 2/3 of the sentences for training (2983 sentences) and the remaining for testing (1492 sentences). Model's performance was analysed through precision (π), recall (ρ) and F_1 (f_1) measures for each tag (obtained from classification's contingency table: prediction *vs.* manual classification).

4.3 Results

Table 4 shows the maximum, minimum, average and standard deviation for precision, recall and F_1 measures. As can be observed all classes present precision values above 88.6% and recall values above 89.1%.

Table 4. Precision, recall and F_1 values for the seventeen POS tags

	precision	*recall*	f_1
max	.986	.991	.988
min	.886	.891	.888
avg	.958	.941	.949
std	.040	.068	.053

[7] `http://www.uol.com.br/`

Table 5 shows performance values for the major grammatical classes: common-noun, proper-noun, verb, adjective, determiner, pronoun, adverb and preposition as well as their mean and standard deviation values. As can be observed, among these major grammatical classes the one with the best values is the proper-noun class; on the other hand, the one with lower values is the adjective one. This table enables us to say that these values are similar to those obtained by SVMTool for other languages and taggers [11].

Table 5. Performance values for major grammatical classes.

	precision	*recall*	f_1
common-n	.965	.964	.965
proper-n	.986	.991	.988
verb	.973	.976	.974
adjective	.886	.891	.888
determiner	.960	.976	.967
pronoun	.930	.951	.944
adverb	.938	.951	.984
preposition	.991	.977	.984
mean ±stdev	.954 ±.035	.960 ±.031	.962 ±.033

4.4 Related work

FreeLing [18] is an open source language analysis tool suite developed at TALP Research Center, at Universitat Politècnica de Catalunya. The tool suite comprises an efficient POS tagger and a corpus for the Portuguese language.

Tests were performed to compare performance results between SVMTool and FreeLing taggers in the domain of ours texts (OJE, Público and Record Portuguese daily newspapers) using FreeLing POS tags. These tests revelead no significant differences between them.

SVMTool was chosen as the POS tagger because:

- SVMTool is around three times faster than FreeLing;
- its software license is free for use and is not propagative (LGPL);
- FreeLing makes compound words and this was not a desired feature.

5 Named Entity Recogniser

As already mentioned, the Entity Recogniser uses Support Vector Machines as the learning algorithm, where each example characterises a single word by means of a set of word-level and list-lookup attributes. Thus, the classifier will indicate for each word presented, whether or not it belongs to an entity and, if so, what type: `Person`, `Organisation` or `Location`.

Next subsections describe the corpus, the experimental setting and the results obtained.

5.1 Corpus

For the development of the classifier, we built a corpus from a choice of several news from the daily newspapers OJE, Público and Record and manually labeled it according to types of entities considered. The corpus comprises 255 documents totalling 3645 sentences and about 88000 words. Table 6 presents the number of names and unique names tagged for each type of entity.

Table 6. Number of names (total and unique) in the corpus.

	total	*unique*
`Person`	1267	583
`Location`	1041	310
`Organisation`	2507	1005

5.2 Experimental setting

The classifier model was built using SVMLib [7] with a linear kernel and default values for all parameters. SVMLib is an integrated software for support vector classification, regression and distribution estimation that supports multi-class classification.

In order to obtain the best results, we made several experiments: context windows with sizes between 1 and 11 and using or not the list-lookup attributes.

A 10-fold cross-validation procedure was applied over the 255 documents and model's performance was analysed through precision (π), recall (ρ) and F_1 (f_1) measures. All significant tests were made regarding a 95% confidence level.

We also created a classifier model whose task was to identify entity boundaries in the text without classifying them, giving rise to a binary classification problem.

5.3 Results

As already said, several experiments were conducted in order to obtain the best classifier by changing the size of the context window (inclusion of features of the neighbour words) and by including or not list-lookup attributes (dictionary, functional words and rare words). The best results were obtained with a context window of size seven (three words before, the word in question and three words after) using the list-lookup attributes. Figure 2 shows the performance values obtained.

It is possible to observe that the most difficult class to recognise is `Location` with a f_1 value of 49%, while `Organisation` has a value of 73%. Also, for `Person` and `Location` names, precision values are greater than recall ones; for `Organization` names is the other way around.

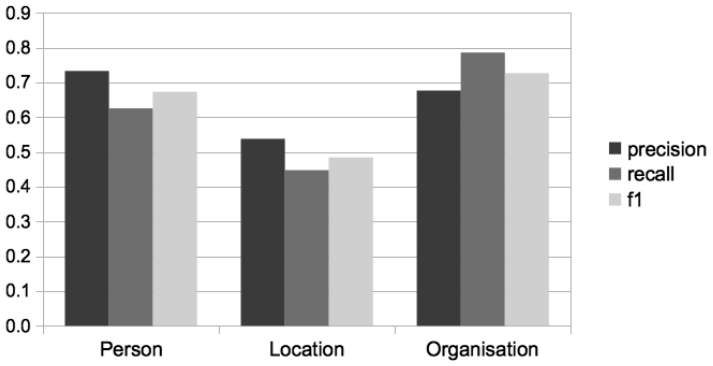

Fig. 2. Performance values for the Entity Recogniser

The model to identify name boundaries was built using the same settings of the recogniser. Table 7 shows the performance values for both models (micro-average values for the recogniser).

Table 7. Precision, recall and F_1 measures for the boundary identifier and the classifier.

	precision	*recall*	f_1
Identifier	.820	.846	.833
Recogniser	.661	.664	.663

As expected, the results for the Identifier are much higher than those obtained for the Recogniser (16% for precision and 18% for recall), since the latter presents a more complex problem.

Table 8 presents the performance values for each of the three studied classes. It is possible to observe that the most difficult class to recognise is `Organisation` with a f_1 value very close to 50%, while `Location` ha a value of 74%.

Table 8. Precision, recall and f_1 measures for the Recogniser (and each entity type)

	precision	*recall*	F_1
`Person`	.582	.701	.636
`Location`	.774	.708	.740
`Organisation`	.497	.492	.494

5.4 Related work

Linguateca, a distributed language resource center for Portuguese Portuguese, promotes the development and evaluation of mechanisms for Named Entity Recognition in the Portuguese language and was responsible for two evaluation contests named HAREM. The last HAREM evaluation was performed in 2008 (HAREM2) and nine different systems with very distinct approaches were presented (many competitors had multiple variants of their own system that were also evaluated). Tables 9, 10 and 11 present the best and worst performance values obtained in HAREM2 for the classes used in this work, namely `Person`, `Location` and `Organization`. It also presents the mean and standard deviation values.

Table 9. Person values HAREM2: Precision, recall and f_1

	precision	*recall*	f_1
max	.768	.716	.632
min	.297	.040	.074
avg	.536	.423	.444
std	.172	.168	.152

Table 10. Location values HAREM2: Precision, recall and f_1

	precision	*recall*	f_1
max	.690	.693	.599
min	.182	.230	.262
avg	.522	.506	.492
std	.170	.106	.114

Table 11. Organization values HAREM2: Precision, recall and f_1

	precision	*recall*	f_1
max	.660	.540	.403
min	.142	.071	.123
avg	.387	.295	.284
std	.159	.138	.069

These results cannot be compared directly with the ones presented in table 8 because they were applied to different corpora. Nevertheless, one can say the recall values obtained in this work are comparable to the best ones of HAREM2; for precision, `Location` entities present better results, but `Person`

and `Organisation` present worse values when compared to the best ones obtained in HAREM2.

6 Conclusions and Future work

This paper presents a system for Named Entity Recognition using Machine Learning techniques. Although it has been applied to the Portuguese language, the system could be considered independent of language and domain: it is only necessary to change the corpus to the chosen language and/or domain. The results are within the values presented in the scientific community for the same type of problem [17][23].

As a side result, but no less important, a part-of-speech tagger for the Portuguese language was obtained. Besides having performance values equivalent to the ones obtained for other languages and taggers, this one has a major advantage over other Portuguese POS taggers: it can be used directly over written text without any pre-processing and produces a tag for each original word.

As future work we intend to compare this system with other systems developed for the Portuguese language using the HAREM2 [16] (last evaluation contest promoted by Linguateca) resources.

In order for the comparison with other systems to be more objective, we intend to test this system with other languages (including English) and other fields other than the daily news. The CONLL shared tasks [23][24] are good resources for comparison.

To improve the system, we intend to create better recognisers through the use of larger corpora and/or the addition of new features (*e.g.* document and corpus features) that could be considered critical to the problem.

References

1. J. Aberdeen, J. Burger, D. Day, L. Hirschman, P. Robinson, and M. Vilain. MITRE: description of the Alembic system used for MUC-6. In *MUC6 '95: Proceedings of the 6th Message Understanding Conference*, pages 141–155, Morristown, NJ, USA, 1995. Association for Computational Linguistics.
2. M. Asahara and Y. Matsumoto. Japanese named entity extraction with redundant morphological analysis. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*, NAACL '03, pages 8–15, Stroudsburg, PA, USA, 2003. Association for Computational Linguistics.
3. S. Baluja, V.O. Mittal, and R. Sukthankar. Applying machine learning for high performance named-entity extraction. In *In Proceedings of the Conference of the Pacific Association for Computational Linguistics*, pages 365–378, 2000.
4. Eckhard Bick. *The Parsing System "Palavras": Automatic Grammatical Analysis of Portuguese in a Constraint Grammar Framework*. PhD thesis, Aarhus University, Aarhus, Denmark, November 2000.
5. Daniel M. Bikel, Scott Miller, Richard Schwartz, and Ralph Weischedel. Nymble: a high-performance learning name-finder. In *In Proceedings of the Fifth Conference on Applied Natural Language Processing*, pages 194–201, 1997.

6. A. Borthwick, J. Sterling, E. Agichtein, and R. Grishman. Nyu: Description of the mene named entity system as used in muc-7. In *In Proceedings of the Seventh Message Understanding Conference (MUC-7*, 1998.
7. C. Chang and C. Lin. *LIBSVM: a library for support vector machines*, 2001. Software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm.
8. N. Chinchor, P. Robinson, and E. Brown. Hub-4 named entity task definition. In *DARPA Broadcast News Workshop*, 1998.
9. J. F. da Silva, Z. Kozareva, and G. P. Lopes. Cluster analysis and classification of named entities. In *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC 2004)*, Lisbon, Portugal, May 2004.
10. G. Doddington, A. Mitchell, M. Przybocki, L . Ramshaw, S. Strassel, and R. Weischedel. The Automatic Content Extraction (ACE) Program–Tasks, Data, and Evaluation. *Proceedings of LREC 2004*, pages 837–840, 2004.
11. J. Giménez and L. Màrquez. SVMTool: A general POS tagger generator based on Support Vector Machines. In *Proceedings of the 4th LREC*, 2004.
12. R. Grishman and B. Sundheim. Message understanding conference-6: a brief history. In *Proceedings of the 16th conference on Computational linguistics - Volume 1*, COLING '96, pages 466–471, Stroudsburg, PA, USA, 1996. Association for Computational Linguistics.
13. IST Laboratório de Engenharia da Linguagem. LABEL-LEX. http://label.ist.utl.pt/pt/labellex_pt.php.
14. Linguateca. Bosque 8.0. http://www.linguateca.pt/floresta/corpus.htmll#bosque.
15. A. McCallum and W. Li. Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003 - Volume 4*, CONLL '03, pages 188–191, Stroudsburg, PA, USA, 2003. Association for Computational Linguistics.
16. C. Mota and D. Santos, editors. *Desafios na avaliação conjunta do reconhecimento de entidades mencionadas: O Segundo HAREM*. Linguateca, 2008.
17. D. Nadeau and S. Sekine. A survey of named entity recognition and classification. *Linguisticae Investigationes*, 30(1):3–26, 2007. Publisher: John Benjamins Publishing Company.
18. Lluís Padró, Miquel Collado, Samuel Reese, Marina Lloberes, and Irene Castellón. Freeling 2.1: Five years of open-source language processing tools. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Bente Maegaard, Joseph Mariani, Jan Odijk, Stelios Piperidis, Mike Rosner, and Daniel Tapias, editors, *LREC'10: Proceedings of the Seventh conference on International Language Resources and Evaluation*, Valletta, Malta, may 2010. European Language Resources Association (ELRA).
19. D. D. Palmer and D. S. Day. A statistical profile of the named entity task. In *ACL Conference for Aplplied Natural Language Processing*, pages 190–193, 1997.
20. L.F. Rau. Extracting company names from text. In *Artificial Intelligence Applications, 1991. Proceedings., Seventh IEEE Conference on*, volume i, pages 29 –32, feb 1991.
21. J. C. Reynar and A. Ratnaparkhi. A maximum entropy approach to identifying sentence boundaries. In *Proceedings of the fifth conference on Applied natural language processing*, ANLC '97, pages 16–19, Stroudsburg, PA, USA, 1997. Association for Computational Linguistics.
22. G. Salton, A. Wang, and C. Yang. A vector space model for information retrieval. *Journal of the American Society for Information Retrieval*, 18:613–620, 1975.

23. E. F. Tjong Kim Sang. Introduction to the conll-2002 shared task: language-independent named entity recognition. In *proceedings of the 6th conference on Natural language learning - Volume 20*, COLING-02, pages 1–4, Stroudsburg, PA, USA, 2002. Association for Computational Linguistics.
24. E. F Tjong Kim Sang and F. De Meulder. Introduction to the conll-2003 shared task: language-independent named entity recognition. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003 - Volume 4*, CONLL '03, pages 142–147, Stroudsburg, PA, USA, 2003. Association for Computational Linguistics.
25. D. Santos and N. Cardoso, editors. *Reconhecimento de entidades mencionadas em português: Documentação e actas do HAREM, a primeira avaliação conjunta na área*. Linguateca, 2007.
26. S. Sekine. Nyu: Description of the japanese ne system used for met-2. In *Proc. of the Seventh Message Understanding Conference (MUC-7*, 1998.
27. S. Sekine and Y. Eriguchi. Japanese named entity extraction evaluation: analysis of results. In *Proceedings of the 18th conference on Computational linguistics - Volume 2*, COLING '00, pages 1106–1110, Stroudsburg, PA, USA, 2000. Association for Computational Linguistics.
28. J. Shawe-Taylor and N. Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press, 2004.
29. K. Takeuchi and N. Collier. Use of support vector machines in extended named entity recognition. In *COLING-02: proceedings of the 6th conference on Natural language learning*, pages 1–7, Morristown, NJ, USA, 2002. Association for Computational Linguistics.
30. V. Vapnik. *Statistical learning theory*. Wiley, NY, 1998.
31. C. Whitelaw and J. Patrick. Evaluating corpora for named entity recognition using character-level features. In Tamás Gedeon and Lance Fung, editors, *AI 2003: Advances in Artificial Intelligence*, volume 2903 of *Lecture Notes in Computer Science*, pages 910–921. Springer Berlin / Heidelberg, 2003.