A highly accurate Named Entity corpus for Hungarian

György Szarvas[1], Richárd Farkas[2], László Felföldi[2], András Kocsor[2], János Csirik[1]

[1] University of Szeged, Department of Informatics
6720, Árpád tér 2., Szeged, Hungary
[2] MTA-SZTE, Research Group on Artificial Intelligence,
6720, Aradi Vértanúk tere 1., Szeged, Hungary
{szarvas, rfarkas, lfelfoldi, kocsor, csirik}@inf.u-szeged.hu

Abstract

A highly accurate Named Entity (NE) corpus for Hungarian that is publicly available for research purposes is introduced in the paper, along with its main properties. The results of experiments that apply various Machine Learning models and classifier combination schemes are also presented to serve as a benchmark for further research based on the corpus. The data is a segment of the Szeged Corpus (Csendes et al., 2004), consisting of short business news articles collected from MTI (Hungarian News Agency, www.mti.hu). The annotation procedure was carried out paying special attention to annotation accuracy. The corpus went through a parallel annotation phase done by two annotators, resulting in a tagging with inter-annotator agreement rate of 99.89%. Controversial taggings were collected and discussed by the two annotators and a linguist with several years of experience in corpus annotation. These examples were tagged following the decision they made together, and finally all entities that had suspicious or dubious annotations were collected and checked for consistency. We consider the result of this correcting process virtually be free of errors. Our best performing Named Entity Recognizer (NER) model attained an accuracy of 92.86% F measure on the corpus.

1. Introduction

A highly accurate Named Entity corpus for Hungarian that is publicly available for research purposes is introduced in the paper, along with its main properties. The results of experiments that apply various Machine Learning models and classifier combination schemes are also presented to serve as a benchmark for further research based on the corpus.

The recognition and classification of proper nouns and names in plain text is of key importance in Natural Language Processing (NLP) as it has a beneficial effect on the performance of various types of applications, including Information Extraction, Machine Translation, Syntactic Parsing/Chunking, etc. Corresponding to this, the identification of phrases that refer to persons, organizations, locations, or proper nouns in general (also called Named Entities or NEs) has been the focus of exhaustive research in the past decade.

Named Entity Recognition (NER) was introduced as shared tasks at the Message Understanding Conferences (Chinchor, 1998) which mainly concentrated on English with newswire texts. Later, research on NER turned to newer domains such as biomedical scientific texts (Kim, 2004) and various languages like Dutch, Spanish and German, with a single system dealing with several languages at the same time (in Computational Natural Language Learning Conferences shared tasks (Tjong, 2002; Tjong, 2003)). There are results on languages like Chinese or Japanese that have unique properties which were published as well. Since the Hungarian language has several interesting, special characteristics (like agglutinativity) which makes many NLP applications rather difficult compared to previously examined European languages, entity recognition in Hungarian texts fits in with these trends.

In the next section we will describe the corpus we built, starting with its general characteristics, than we will discuss the details of the annotation process we followed to ensure a high quality annotation with as few errors as possible. In Section 3 we outline our NER system for Hungarian and in the last section we discuss our recent experiments and make some brief concluding remarks.

2. The corpus

The Named Entity Corpus for Hungarian is a sub corpus of the Szeged Treebank (Csendes et al., 2004),, which contains full syntactic annotations done manually by linguist experts.[1] A significant part of these texts has been annotated with Named Entity class labels in line with the annotation standards used on CoNLL conferences[2]. The corpus is available free of charge for research purposes.

2.1. General properties

Short business news articles collected from MTI (Hungarian News Agency, www.mti.hu) constitute a 200,000 word part of the Szeged Treebank, covering 38 topics concerning the NewsML[3] topic coding standard, ranging from acquisition to stock market changes or to new plants openings.

We provide the text with annotations of person, location, organization names and miscellaneous entities that are proper names but do not belong to the three other classes. Part of speech codes generated automatically by a POS tagger (Kuba, 2004) developed at the University of Szeged have also been added to the database. Furthermore we provide some gazetteer resources in Hungarian (Hungarian first names, company types, list of names of countries, cities, etc.) that we used for experiments to build a model based on the corpus.

The data also has some interesting properties relating to the distribution of class labels which is induced by the domain specificity of the texts – organization class, which turned to be harder to recognize than person names for example, is more frequent than in corpuses used at the CoNLL conferences.

[1] The project was carried out together with MorphoLogic Ltd. and the Hungarian Academy's Research Institute for Linguistics

[2] Visit http://www.cnts.ua.ac.be/conll2003/ner/ for details and similar data in English and German

[3] http://www.newsml.org/IPTC/NewsML/1.2 /documentation/NewsML_1.2-doc-Guidelines_1.00.pdf

	Tokens	Phrases
Non-tagged tokens	200067	–
Person names	1.921	982
Organizations	20.433	10.533
Locations	1.501	1.294
Misc. entities	2.041	1.662

Table 1.: Corpus details

2.2. The annotation process

As annotation errors can readily mislead learning methods, accuracy is a critical measurement of usefulness of language resources containing labelled data that can be used to train and test supervised Machine Learning models for Natural Language Processing tasks. With this we aimed to create a corpus with as low an annotation error rate as possible, which could be efficiently used for training NE recognizer and classifier systems for Hungarian. To guarantee the precision of tagging we set up an annotation procedure with three stages.

In the first stage two linguists labelled the corpus with NE tags and received the same instructions. Both of them were told to use Internet or other sources of knowledge whenever they were confused about their decision. Thanks to this and the special characteristics of texts (domain specificity helps experts to become more familiar with the style and characteristics of business news articles), the resulting annotation was very accurate regarding the inter-annotator agreement rate. We used the evaluation script made for the CoNLL conference shared tasks, which measures a phrase-level accuracy of a Named Entity-tagged corpus. As inter-annotator agreement is expressed in F values, comparing one annotation with another leads to the same result by treating any of the two annotations as the correct one (The precision of comparing A with B is the same as the recall of a comparison between B and A). The corpus showed inter-annotator agreement of 99.6% after the first phase.

In the second phase all words that received different class label were collected for discussion and revision by the two annotators and the chief annotator with several years of experience in corpus annotation. The chief annotator gave instructions to the other two to perform the first phase labelling. Those entities that caused disagreement among the linguists initially received their class labels according to the joint decision of the group.

In the third phase all NEs that showed some kind of similarity to those that had been tagged ambiguously earlier were collected from the corpus for revision even though they received the same labels in the first phase. We did this to ensure the consistency of the annotation procedure. The resulting corpus after the third stage had an agreement rate of 99.89% and 99.77% with the first and second annotations.

Creating error free resources of reasonable size has a very high cost and, in addition, publicly available NE tagged corpuses contain certain annotation errors, so we can say the corpus we developed has a great value for the research community of Natural Language Processing. As far as we know this is the only Hungarian NE corpus currently available, and its size is comparable to those that have been made for other languages.

3. The NER system for Hungarian

The authors carried out some earlier research (Farkas, 2005) on the classification accuracy of NE recognition on Hungarian business news texts and demonstrated that different learning methods (Artificial Neural Network (Bishop, 1996), Support Vector Classifier (Vapnik, 1996), Decision Tree Classifier (Quinlan, 1993)) can achieve over 90% in F measure scores on recognizing the four NE classes.

3.1. Feature set

To build a learning model we collected various types of numerically encodable information describing each term and its context. These consisted of the vector of attributes for the classification (which is partly based on the model described in (Tjong, 2003)).

We designed a model with the following features:

- part-of-speech code (for the word itself and for its +/-4 words neighborhood),
- case code,
- type of initial letter of the word,
- one that tells us if the word contains digits inside the word form,
- one that tells us if the word contains capitalized letter inside the word form,
- one that tells us if the word contains punctuation inside the word form,
- the word at the beginning of a sentence or not,
- the word between quotation marks or not,
- word length,
- if the word is an Arabic or Roman number,
- memory which tells us whether the word in question received an NE tag earlier in the actual document (one article) or not, and what type,
- the ratio of lowercase and capitalized frequency in Szószablya (Halácsy, 2003) term frequency dictionary
- the ratio of mid-sentence uppercase and all uppercase frequency in Szószablya,
- a feature telling us whether the word is in one of the trigger word dictionaries (we used dictionaries of surnames, organization forms, geographic name types and stopwords, and a very small gazetteer containing the names of the world's countries and names of the biggest cities)

3.2. C4.5 decision tree

C4.5 (Quinlan, 1993) is based on the well-known ID3 tree learning algorithm. It is able to learn pre-defined discrete classes from labeled examples. The result of the learning process is an axis-parallel decision tree. This means that during the training, the sample space is divided into subspaces by hyperplanes which are parallel to every axis but one. In this way, we get many n-dimensional rectangular regions that are labeled with class labels and organized in a hierarchical way, which can then be encoded into the tree. Splitting is done by axis-parallel hyper-planes, and hence learning is very fast. One great advantage of the method is time complexity; in the worst case it is $O(dn^2)$, where d is the number of features and n is the number of samples.

3.3. Artificial Neural Networks (ANN)

Since it was realized that, under proper conditions, ANNs can model the class posteriors (Bishop, 1996), neural nets have become evermore popular in the Natural Language Processing community. But describing the mathematical background of the ANNs here is beyond the scope of this article. Besides, we believe that they are well known to those who are acquainted with pattern recognition. In our experiments we used the most common feed-forward multilayer perceptron network with the backpropagation learning rule.

3.4. Support Vector Machines (SVM)

The well-known and widely used Support Vector Machines (Vapnik, 1996) is a kernel method that separates data points of different classes with the help of a hyperplane. The created separating hyperplane has a margin of maximal size with a proven optimal generalization capacity. Another significant feature of margin maximization is that the calculated result is independent of the distribution of the sample points. Perhaps the success and the popularity of this method can be attributed to this property.

3.5. Majority voting

In experiments the results showed that different voting schemes which used trained models of inherently different classifiers are also beneficial to classification accuracy. We applied a decision function that accepts the classification made by any two of the above-mentioned three models as the joint prediction if they coincide (i.e. two models predicted the same class label), and accepts the classification of the best performing individual model if three different predictions were made. This reduced the classification error rate by 17% compared to the best individual model (F measure increased from 85.0% to 87.53%).

3.6. Post processing

Several simple post processing methods can bring about some improvement in system accuracy. Take, for instance, acronym words. They are often easier to disambiguate in their longer, phrase form, so if we find both in the same document we can override the prediction given to the acronym if it does not coincide with the previously met longer form.

As we use a phrase level evaluation method (the one used on the CoNLL conferences), while our NER model is built to classify single terms, misclassifying one term multiples the error rate when the misclassified term is part of a longer phrase. To avoid this it is usual to unify the class labels of a longer NE term sequence, as it can eliminate the multiple error of an accidental misclassification in the middle of the sequence, while unifying the whole sequence with a bad class label does not introduce any further error (one phrase is counted once, so if any of its constituents is misclassified it will be a faulty phrase). As the learning model cannot take into account the specificity of the evaluation script, it is simpler to post process the output of the system to fit the CoNLL evaluation.

3.7. AdaBoost

Since voting schemes like the one mentioned above can be successful only if the individual models are highly accurate in terms of precision (thus coinciding predictions will more often mean error correction), we applied a boosting strategy to improve the precision of the three learning models at the cost of somewhat lower recall.

Boosting was introduced by Schapire as a method for improving the performance of a weak learning algorithm (Schapire, 1990). The algorithm generates a set of classifiers (of the same type) by applying bootstrapping on the original training data set and it makes a decision based on their votes. AdaBoost changes the weights of the training instances provided as input for each inducer based on classifiers that were previously built. The final decision is made using a weighted voting schema for each classifier, whose weights depend on the performance of the training set used to build it. This way we can achieve even more accurate results with majority voting schemes than those with our previous model.

Iterative voting models like boosting can cure typical errors such as those discussed in the post processing section, and thus – as our experiments prove – it seems that post processing cannot bring further significant advance to models based on boosting.

4. Experimental Results

We trained ANN, SVM and C4.5 classifiers for the Hungarian NER, and tested them on a randomly selected 10% segment of the corpus. The remaining 90% data constituted the training set, which was split into a train and development phase test set. We followed the NER tasks of CoNLL conferences in this. Table 2 concludes the results of our latest experiments.

	Term accuracy	**Phrase accuracy**	**Phrase (post processed)**
ANN	92.19%	85.00%	87.39%
SVM	91.44%	84.22%	86.17%
C4.5	91.80%	84.04%	87.83%
VOTING	92.91%	87.53%	89.16%
BOOSTING + C4.5	94.19%	92.77%	92.81%

Table 2.: Summary of the experimental results

The F measure accuracies of the models were 92.19% for ANN, 91.44% for SVM and 91.80% for C4.5 on the evaluation set (the development phase test set gave better results by approximately 2% for each models thanks to the fine tuning of parameters). We also calculated the standard deviations of the models, by repeating the above experiments 10 times on randomly performed train-development-evaluation splits. ANN showed a standard deviation of 1.86% in F measure, while the other two models had slightly higher deviation values (2.43% for SVM, 2.09% for C4.5).

We performed post processing and CoNLL style evaluation as well. While ANN was superior to the other two models in term level and phrase level accuracy

(85.00% to 84.22% of SVM and 84.04% of C4.5), post processing had a different effect on the three models and, surprisingly, C4.5 became the best model for phrase accuracy after post processing its output (87.83% to 87.39% of ANN and 86.17% of SVM).

Majority voting of the three trained models gave an increase in each comparison compared to the base models, proving that these inherently different learners can be effectively employed in decision committees. The voted model had a 92.91% term level accuracy and 87.53% phrase level accuracy without post processing. When post processing was applied to the voted output, the performance increased to 89.16%.

We chose C4.5 for further experiments using the AdaBoost method, as decision trees are common subjects to boosting and often give good results. With a boosting of 20 iterations the C4.5 model increased its accuracy to 94.19% in F measure at the term level, which is an improvement of 2.39%. This, of course, brought about an increase in the phrase level accuracy as well, which became 92.77%. Interestingly, though, post processing had a very small additional benefit on this boosted model which presumes that boosting succeeded to improve the performance of C4.5 on similar examples that were subjects to post processing (the accuracy increased to 92.81%, phrase level).

These results are significantly better than those published for other languages, perhaps due to the domain specific nature of the texts. We also say that our results prove that NER can be performed for the Hungarian language with competitive accuracy, unlike some other Natural Language Processing Tasks.

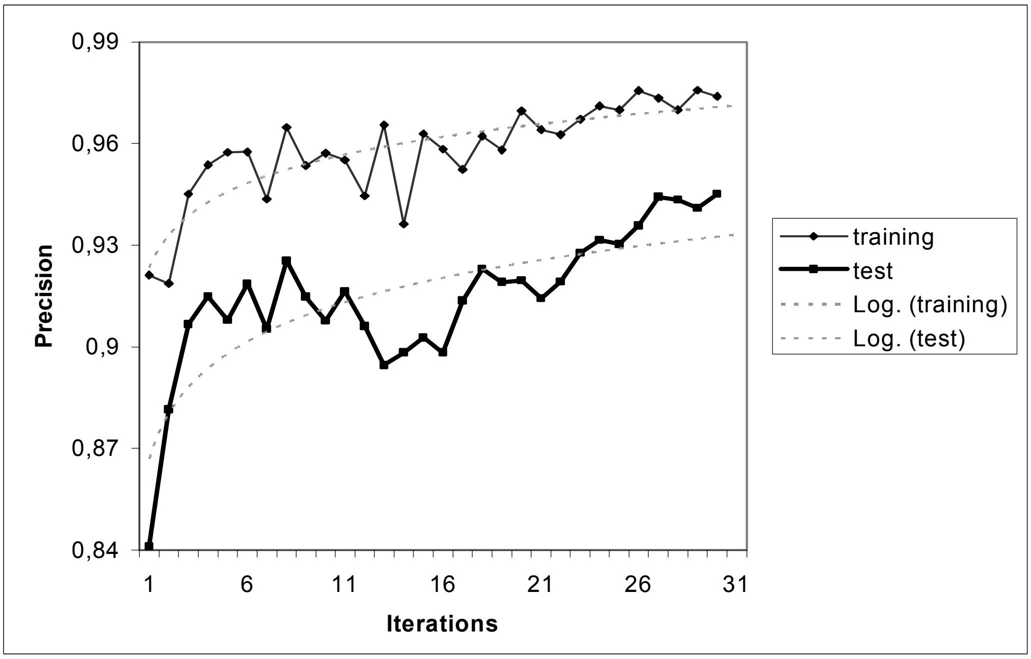

Figure 1.: Change of prediction accuracy (with trend lines) during the iterations of boosting

References

Bishop, C. M. (1996) *Neural Networks for Pattern Recognition*, Oxford Univerisity Press Inc., New York

Chinchor, N. (1998) *MUC-7 Named Entity Task Definition*, in Proceedings of Seventh Message Understanding Conference (MUC-7), Washington

Csendes, D., Csirik, J., Gyimóthy, T. (2004). *The Szeged Corpus: A POS tagged and Syntactically Annotated Hungarian Natural Language Corpus* in Proc. of TSD 2004, Brno, LNAI vol. 3206, pp. 41-49

Farkas, R., Szarvas, Gy., Kocsor, A. (2005) *Named Entity Recognition for Hungarian Using Various Machine Learning Algorithms*, Acta Cybernetica, accepted for publication

Halácsy P., Kornai A., Németh L., Rung A., Szakadát I., Trón V. (2003). *A szószablya projekt* MSZNY 2003, pp. 298–299,

Kim, J-D., Ohta, T., Tsuruoka, Y., Tateisi, Y. (2004) *Introduction to the Bio-Entity Recognition Task* at JNLPBA, Proceedings of JNLPBA-2004, Geneva

Kuba, A., Hócza, A., Csirik, J. (2004). *POS tagging of Hungarian with combined statistical and rule-based methods* in Proc. of TSD 2004, Brno, LNAI vol. 3206, pp. 113-121

Quinlan, J. R. (1993) *C4.5: Programs for machine learning*, Morgan Kaufmann

Schapire, R. E. (1990) *The Strength of Weak Learnability*, Machine Learning, Vol. 5, pp. 197-227

Tjong Kim Sang, Erik. F. (2002) *Introduction to the CoNLL-2002 Shared Task: Language-Independent Named Entity Recognition.* In Proceedings of CoNLL-2002, pp. 155-158., Taipei

Tjong Kim Sang, Erik F. and De Meulder, Fien (2003) *Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition*, in Proceedings of CoNLL-2003, 142-147, Edmonton http://cnts.uia.ac.be/signll/conll.html

Vapnik, V. N. (1998) *Statistical Learning* Theory, John-Wiley & Sons Inc.