Joint Part-of-Speech Tagging and Named Entity Recognition Using Factor Graphs*

György Móra[1] and Veronika Vincze[2,3]

[1] University of Szeged, Department of Informatics,
6720 Szeged, Árpád tér 2., Hungary
gymora@inf.u-szeged.hu
[2] MTA-SZTE Research Group on Artificial Intelligence,
6720 Szeged, Tisza Lajos krt. 103., Hungary
[3] Universität Trier, Linguistische Datenverarbeitung,
54286 Trier, Universitätsring, Germany
vinczev@inf.u-szeged.hu

Abstract. We present a machine learning-based method for jointly labeling POS tags and named entities. This joint labeling is performed by utilizing factor graphs. The variables of part of speech and named entity labels are connected by factors so the tagger jointly determines the best labeling for the two labeling tasks. Using the feature sets of SZTENER and the POS-tagger magyarlanc, we built a model that is able to outperform both of the original taggers.

Keywords: POS tagging, named entity recognition, joint labeling, factor graphs.

1 Introduction

In syntax, proper nouns behave in a similar way to a common noun (e.g. in the sentence *Have you seen "Interview with the vampire"?*, the title of the movie fulfils the function of the object and could be substituted by the common noun *movie*) and hence they are considered a subclass of nouns. Some morphological coding systems give a distinct code to proper nouns (such Np-s* in the Hungarian MSD coding system or NNP in the Penn Treebank system), but the members of multiword proper nouns may belong to other parts of speech on their own (in the above example, we may have a preposition (*with*) and an article (*the*) as well). In such cases, a possible solution is to duplicate the part of speech codes (i.e. to add a proper noun code to every word) because in fact every word with any part of speech code can be part of a proper noun. Thus, in this example all the four words within the title would have the part of speech (POS) code of a proper noun. However, this duplication would make the POS tagging more expensive (each word would have at least two possible codes, from which the POS tagger should select the correct one) and the POS tagger should be able to recognize proper nouns, which is normally the task of a named entity recognition (NER) system.

Here, we propose a solution to solve both problems – POS tagging and named entity recognition – in a parallel way. Our approach separates the two subtasks by assigning

* This work was supported in part by the National Innovation Office of the Hungarian government within the framework of the projects BELAMI and MASZEKER.

P. Sojka et al. (Eds.): TSD 2012, LNCS 7499, pp. 232–239, 2012.

ordinary POS codes to the parts of the proper nouns, which are tagged separately. The two tagging processes work in a parallel way but the label sequences are determined depending on each other. This joint labeling is carried out by utilizing factor graphs. The variables of part of speech and named entity labels are connected by factors, hence the tagger jointly determines the best labeling for both labeling tasks. Sequential models where different subtasks are performed subsequentially usually aggregate tagging errors and one tagger in the processing pipeline may utilize labels created by the previous taggers. Performing the processes in parallel, both taggers can use the other's labels as features. In this paper, we carry out experiments on English and Hungarian texts and we find that parallel labeling can improve the performance and quality of both labeling processes.

2 Morphology and Proper Nouns

Proper nouns are usually considered to be rigid designators, which constantly refer to the same entity [1]. Rigidity here means that the relationship between the designator and the designated is constant but we argue that rigidity can be applied to morphology as well. In agglutinative languages, proper nouns can be inflected or some derivational suffixes could be added, but their base form does not change. It is most salient when a noun with a morphologically irregular behaviour acts as a proper noun as in the following Hungarian examples: *Fodor* 'Fodor as a proper noun' – *Fodort* 'Fodor-ACC' vs. *fodor* 'frill' – *fodrot* 'frill-ACC'.

The common noun *fodor* has a vowel-deleting stem, which means that the last vowel of the stem is deleted before certain suffixes. However, when it functions as a proper noun, this rule is no longer valid (i.e. the last vowel is preserved), which may be exploited in named entity recognition. As an accusative form of *fodor*, the morphological analyzer would expect to get *fodrot*. If it gets *Fodort* as input, it can only analyze this word form with the help of guessing, separating it into the morphs *fodor* 'frill' and *t* 'accusative suffix'. If this lemma is listed in the morphological database, but with a different analysis (fodr+ot vs. fodor+t), then it is highly probable that it is an instance of a proper noun.

Some proper nouns contain an inflectional (or derivational) suffix even within their lemmas. One such case is *McDonald's* in English, where we can see a possessive marker as part of the original name. However, when it comes to speaking about things owned by McDonald's, we get the form *McDonald's'*. If it is supposed that the morphological analyzer does not include a list of companies, this latter form is analyzed by the guesser, whereby the morphs *McDonald's* and *'* are produced. Since the lemma already contains a possessive suffix, it is again suggested that it is a proper noun.

From a morphological analysis view point, named entity recognition that is carried out in parallel can help in accelerating the process. If an element is recognized as a named entity, the morphological analyzer can immediately call the guesser instead of analyzing the element in the traditional way.

3 Joint Labeling Approaches

Different labeling tasks (such as POS tagging, chunking, NER) are usually performed in sequential steps and are defined as separate machine learning problems. Using sequential processing pipelines, a labeler can only use labels as features produced by a previously performed labeling step. Another drawback of methods which use sequential analysis is that the errors made by separate steps may aggregate over the pipeline. To overcome these problems, multiple labeling tasks should be performed in a single step.

Combining the label-spaces of multiple labeling tasks produces a single label-space and the separate machine learning problems become a single machine learning task. If the separated label-spaces were large, the size of the combined space might be intractable. The combined label-space may contain labels which are rare or they do not even occur in the training data and the detection of these combinations cannot be learned properly. A single feature space may not be ideal for all of the labeling subtasks.

In our experiments we utilized an approach based on probabilistic graphical models to perform joint labeling. The MALLET GRMM [2] and FactorIE [3] software packages enable us to define arbitrary conditional dependencies between labels and feature sets instead of using classical linear chain conditional random fields. A token may hold multiple types of labels and feature sets. The conditional dependencies between the labels and the features can be described by factors. Factors between POS and NER labels permit the interaction between labels during the learning and tagging process, but we can still have separate feature sets for each labeling task. This method can be adapted to other tasks such as chunking and it can also handle three or more label types.

Experiments carried out on English language texts reveal that the joint learning of POS and chunk tags gives better results than just performing these task sequentially. Our experiments showed that the accuracy of POS tagging rose from 62.42% to 72.87% and the accuracy of chunking rose from 83.95% to 85.76% by performing joint labeling using the same feature sets as that in the separated cases. The labels act as dynamic features during parallel training, improving the accuracy scores of both labeling tasks. The experiments were performed on a subset of the CoNLL-2000 Shared Task data.

In experiments carried out on the Spanish language dataset of the language independent NER task of the CoNLL2003 Shared Task, we found that both the POS and NER labeling can be improved using joint labeling. With a basic feature set and separate labeling we achieved an accuracy of 88.6% in POS labeling and an F-measure of 39.5 in NER. These scores rose to 88.7% and 42.2, respectively, with our joint labeling approach.

4 Named Entity Recognition

Named entity recognition is a key part of all information extraction systems as named entities are the main building bricks of relations and events. The classification of the entities is a more challenging problem than the simple recognition and it often needs information based on the environment of the token.

The recognition of named entities may be token or sequence-based. The token based approach assigns a label to each individual token independently of the labels of the

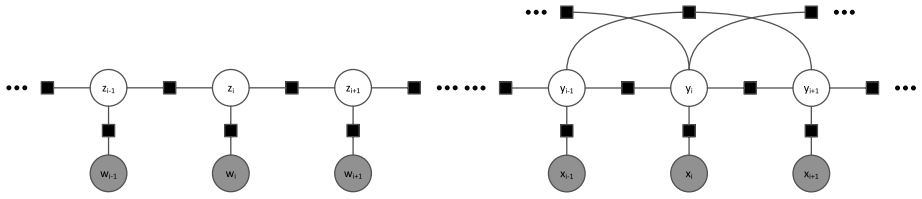

Fig. 1. First-order model used for named entity recognition and second-order model used for POS tagging

other tokens. Support Vector Machines [4] or Maximum Entropy methods [5] are most widely used as machine learning models. The different methods can be combined in a multilayer classification scheme [6].

Sequence labeling approaches like Hidden Markov Models [7] and Conditional Random Fields (CRF) [8] label a sequence of tokens and the most likely sequence is determined instead of separate token classification. The results of the CoNLL-2002 and CoNLL-2003 Shared Tasks on NER revealed that the sequence labeling approaches usually outperform token labeling methods [9,10].

Our NER system is based on the feature set of the SZTENER language independent Named Entity Tagger [11]. It uses a first order CRF machine learning model implemented in the MALLET [2] machine learning tool. The tagger utilizes orthographic, frequency-based and dictionary-based features. In our machine learning settings, we applied the feature-vectors extracted from SZTENER.

In order to compare the two approaches in the same machine learning framework, we implemented a similar first order chain (see Figure 1) in the FactorIE probabilistic programming framework. A modified version of the Gibbs Sampler was employed to train our models.

5 Part of Speech Tagging

POS tagging is a key step in syntactical analysis and many systems use POS codes as features. POS tagging is a token classification task where a label is assigned to each token from a coding system. Here, we used the simplified Hungarian MSD coding system, which is more suitable for machine learning.

Several POS taggers are available for Hungarian (see [12]). Our POS tagger is based on *magyarlanc*, which is a modified version of the Stanford POS Tagger [13]. It utilizes a Cyclic Dependency Network with Maximum Entropy classifier. The feature set adapted to Hungarian consists of character prefixes and suffixes, the word forms and the token patterns of the words.

The Cyclic Dependency network used by the original POS tagger was not directly implementable in FactorIE, but the main structure of dependencies was kept in a factor graph (Figure 1). The resulting model is similar to the NER model, but it has a second order chain and various factors emulating the label and token combination features of the original system.

The set of possible POS codes were added to the model as a second feature vector which allows the learner to incorporate the output of the morphological analyzer, but the possible tags are not limited to these labels. In some cases the tagger chose the correct label despite the fact that the morphological analyzer failed to recommend it.

6 Results

By connecting the graphical models of the NER and POS tagging tasks with factors between the two label sequences, a joint model was created (Figure 2). The NER and POS label variables of the same token and label variables for neighbouring tokens were connected by factors.

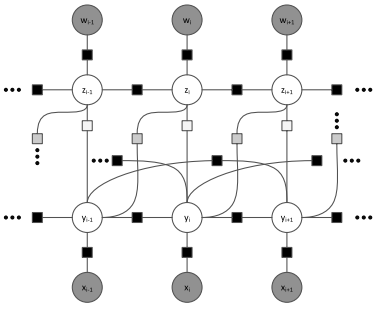

Fig. 2. Unification of the two independent models

We evaluated the original systems and the independent and joint models of our approach on the subcorpus containing business newswire texts of the *Szeged Corpus*, where the gold standard Named Entities are annotated [14,15]. It is a Hungarian language corpus that contains over 220,000 tokens in 9400 sentences. We split the corpus into training and test sets in a 70/30 ratio.

In the original MSD annotation, proper nouns had the code Np-* and multiword NEs were contracted. Before our evaluation, multiword NEs were split into parts and their members were reannotated; moreover, proper and common nouns were not distinguished, both having the POS code 'noun'. Thus, the multiword NE *Magyar Nemzeti Bank* "National Bank of Hungary" was retagged with the A A N POS sequence.

6.1 The Evaluation of Named Entity Recognition

Here, we applied a phrase-based evaluation of named entity recognition. This means that the labeling of multiword NEs was only accepted if all of its members were labeled correctly and no other neighbouring words were marked. For the sake of comparison, all models were trained and evaluated on the same training and test sets, and the same metrics were applied. The phrase-based F-score was used in the evaluation process of the CoNLL-2003 shared task on NER, which we also applied.

Table 1 lists the results obtained by the base model and our joint and sequential NER models running the learning algorithm for 2 and 5 iterations.

Our results confirm that using the same feature space, joint tagging improves the quality of NER compared with the independent model. The independent model improves its performance from 83.86 (which was worse than the base model) to 88.93. The inefficiency of the feature space is indicated by the fact that the results, which are worse than those of the original model, are actually made worse by increasing the number of iterations, most probably due to overfitting. This lack of information may be compensated by the presence of POS codes in the joint learning process.

6.2 The Evaluation of POS-Tagging

POS-tagging was trained and evaluated on a reduced set of MSD codes [16], only those codes being distinguished where the word form does not unambiguously determine the POS-code (e.g. *tőrnek* can mean both "of (the) dagger" and "for (the) dagger"). The reduction of the original set with several hundred codes was necessary because it would have been unfeasible for the machine learning algorithm to treat them properly. Since the original codes can be recovered from the reduced ones, this reduction does not have any substantial effect on the results.

We also evaluated the results concerning just the first character (i.e. the one denoting the main part of speech) of the codes. Hence, it could be seen what the differences were between the two POS-tagging methods that achieved almost the same results on the reduced set of MSD-codes.

In contrast with NER, we used accuracy to measure the performance of the systems, but macro F-scores were also provided for each POS class. Accuracy reflects the average performance of the system, while macro F-score is the average of the F-scores of the classes. If it is only the frequent POS tags that the system identifies correctly, the average of F-scores per class will be low due to the high number of mistagged POS classes with only a few members.

Table 1. Results obtained for NER and POS tagging

		Named Entity			Part of Speech			
					Reduced MSD		Main part of speech	
It.	Model	Precision	Recall	$F_{\beta=1}$	Accuracy	$F_{\beta=1 macro}$	Accuracy	$F_{\beta=1 macro}$
	SZTENER	86.81	88.71	87.75				
	magyarlanc				97.11	67.81	97.98	85.18
2	Independent	86.81	81.11	83.86	97.75	71.03	98.60	84.12
	Parallel	88.57	89.27	88.93	97.78	72.48	98.68	86.32
5	Independent	84.73	81.60	83.13	**98.00**	71.33	98.78	86.44
	Parallel	**89.71**	**90.04**	**89.87**	97.99	**73.32**	**98.81**	**88.77**

The improvement in the POS-tagging results can be primarily attributed to the proper analysis of words that begin with a capital letter. In Hungarian, it is mostly sentence-initial words and named entities that start with a capital letter. With the parallel POS

tagging and NER, sentence-initial named entities were easier to find, so it was easier to assign the proper MSD code to the rest of sentence-initial elements. For instance, the sentence-initial word *Szerinte* was tagged as a noun in the sequential tagging, while it was assigned the proper code 'adverb' in the parallel tagging. The POS tagging of abbreviations rose by 17.68%, which can be attributed to the correct identification of *Dr.* and *Jr.*, which are parts of named entities. The tagging of some NEs ending in pseudo-interjections like *Palotainé* "Mrs. Palotai" was also improved using the parallel NER approach.

Overall, we may conclude that the biggest differences between the systems could be observed in the case of the rare POS classes, while there were no great differences in the case of frequent POS classes. However, the accuracy on the latter class was high (above 97%) when tagging with the sequential model hence the addition of NER did not significantly affect the results.

Although the absolute difference between the accuracies may seem small, in the case of parallel tagging the quality of POS tagging was improved. Macro averages in Table 1 show that the parallel system performs better with POS classes having only a few members, hence it is more balanced. When taking just the main POS into account, it is seen that the parallel system identifies the main POS code slightly better than the sequential system; that is, the errors made by the former are less serious than those of the latter.

7 Conclusions

Here, we presented our system for the joint labeling of part of speech tags and named entities. Our results show that the performance on both tasks can be slightly improved, compared with the traditional sequential models. Although the improvement is less substantial in the case of POS-tagging, our method was still able to raise the overall quality. In our experiments we found that joint labeling is able to exploit labels of one task as features in the other task. These features are not independent of each other from a linguistic point of view, but this joint model is linguistically more feasible than single model approaches. The creation of joint models like this seems to be a promising direction for further research.

References

1. Kripke, S.: Naming and Necessity. Basil Blackwell, Oxford (1980)
2. McCallum, A.K.: Mallet: A machine learning for language toolkit (2002), `http://mallet.cs.umass.edu`
3. McCallum, A., Schultz, K., Singh, S.: FACTORIE: Probabilistic Programming via Imperatively Defined Factor Graphs. In: Advances in Neural Information Processing Systems, vol. 22, pp. 1–9 (2009)
4. Mayfield, J., Mcnamee, P., Piatko, C.: Named entity recognition using hundreds of thousands of features. In: Proceedings of CoNLL 2003, pp. 184–187 (2003)
5. Chieu, H.L., Ng, H.T.: Named entity recognition with a maximum entropy approach. In: Daelemans, W., Osborne, M. (eds.) Proceedings of CoNLL 2003, Edmonton, Canada, pp. 160–163 (2003)

6. Florian, R., Ittycheriah, A., Jing, H., Zhang, T.: Named entity recognition through classifier combination. In: Daelemans, W., Osborne, M. (eds.) Proceedings of CoNLL 2003, Edmonton, Canada, pp. 168–171 (2003)
7. Miller, S., Crystal, M., Fox, H., Ramshaw, L., Schwartz, R., Stone, R., Weischedel, R., Group, T.A.: Algorithms That Learn To Extract Information BBN: Description of The Sift System as Used For MUC-7. In: Proceedings of MUC-7 (1998)
8. Lafferty, J.D., McCallum, A., Pereira, F.C.N.: Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In: Proceedings of the Eighteenth International Conference on Machine Learning, ICML 2001, pp. 282–289. Morgan Kaufmann Publishers Inc., San Francisco (2001)
9. Tjong Kim Sang, E.F.: Introduction to the CoNLL-2002 Shared Task: Language-Independent Named Entity Recognition. In: Proceedings of CoNLL 2002, Taipei, Taiwan, pp. 155–158 (2002)
10. Tjong Kim Sang, E.F., De Meulder, F.: Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition. In: Daelemans, W., Osborne, M. (eds.) Proceedings of CoNLL 2003, Edmonton, Canada, pp. 142–147 (2003)
11. Szarvas, Gy., Farkas, R., Kocsor, A.: A Multilingual Named Entity Recognition System Using Boosting and C4.5 Decision Tree Learning Algorithms. In: Todorovski, L., Lavrač, N., Jantke, K.P. (eds.) DS 2006. LNCS (LNAI), vol. 4265, pp. 267–278. Springer, Heidelberg (2006)
12. Halácsy, P., Kornai, A., Oravecz, C.: HunPos: an open source trigram tagger. In: Proceedings of the ACL 2007 Demo and Poster Sessions, pp. 209–212. Association for Computational Linguistics, Stroudsburg (2007)
13. Toutanova, K., Klein, D., Manning, C.D., Singer, Y.: Feature-rich part-of-speech tagging with a cyclic dependency network. In: Proceedings of NAACL 2003, pp. 173–180. Association for Computational Linguistics, Stroudsburg (2003)
14. Csendes, D., Csirik, J., Gyimóthy, T.: The Szeged Corpus. A POS Tagged and Syntactically Annotated Hungarian Natural Language Corpus. In: Hansen-Schirra, S., Oepen, S., Uszkoreit, H. (eds.) COLING 2004 5th International Workshop on Linguistically Interpreted Corpora, Geneva, Switzerland, pp. 19–22 (2004)
15. Szarvas, Gy., Farkas, R., Felföldi, L., Kocsor, A., Csirik, J.: A highly accurate Named Entity corpus for Hungarian. In: Proceedings of LREC 2006 (2006)
16. Zsibrita, J., Vincze, V., Farkas, R.: Ismeretlen kifejezések és a szófaji egyértelműsítés. In: Tanács, A., Vincze, V. (eds.) MSzNy 2010 – VII. Magyar Számítógépes Nyelvészeti Konferencia, Szeged, Hungary, University of Szeged, pp. 275–283 (2010)