An efficient language independent toolkit for complete morphological disambiguation

László János Laki, György Orosz
MTA-PPKE Hungarian Language Technology Research Group,
Pázmány Péter Catholic University, Faculty of Information Technology and Bionics,
50/a Práter street, 1083 Budapest, Hungary
{laki.laszlo, oroszgy}@itk.ppke.hu

Abstract

In this paper a Moses SMT toolkit-based language-independent complete morphological annotation tool is presented called HuLaPos2. Our system performs PoS tagging and lemmatization simultaneously. Amongst others, the algorithm used is able to handle phrases instead of unigrams, and can perform the tagging in a not strictly left-to-right order. With utilizing these gains, our system outperforms the HMM-based ones. In order to handle the unknown words, a suffix-tree based guesser was integrated into HuLaPos2. To demonstrate the performance of our system it was compared with several systems in different languages and PoS tag sets. In general, it can be concluded that the quality of HuLaPos2 is comparable with the state-of-the-art systems, and in the case of PoS tagging it outperformed many available systems.

Keywords: PoS tagging, lemmatization, SMT decoder, suffix guesser

1. Introduction

Automatic morphological annotation (e.g. complete morphological disambiguation) is a fundamental task in the natural language processing chain; since relatively small differences in the accuracy of morphological annotation can lead to great quality differences at higher levels of linguistic processing. Complete morphological disambiguation is the process to find the lemma and identify the morphosyntactic label of each word of a sentence in one step. Earlier approaches, created for English, did part-of-speech (PoS) tagging and lemmatization separately, thus most of the further implementations also followed the same tendency. However, only few of them carry out complete morphological disambiguation, which is essential in the case of morphologically rich languages. Furthermore, there are only a few PoS taggers that achieve high accuracy amongst grammatically different languages. A language dependent tool may produce high token accuracy for a given corpus. Most algorithms, however are hardly applicable for different sorts of data, thus resulting in poor quality of annotation.

The aim of this study is to introduce a new approach for complete morphological disambiguation, which can be used for different sorts of languages, while producing accuracy scores competing with the ones of language dependent systems.

The structure of our paper is as follows. First of all, we present the difficulties of morphological disambiguation. This is followed by an overview of a language-independent morphological annotation tool that is based on the Moses SMT toolkit called HuLaPos2. Its ability to handle rich language models and the beam-search-based stack decoding algorithm implemented in the Moses decoder make it a promising tool to apply to this task. Finally, to demonstrate the performance of our system, it was tested on several languages. The performance of HuLaPos2 was compared to the available state-of-the-art systems of five different languages. The results of the evaluation are shown in section 5.

2. Motivation and background

2.1. Complete morphological disambiguation

Complete morphological disambiguation is a process to find out the PoS tag and the lemma of each word simultaneously. Several tools exist that accomplish these tasks separately, but only few of them implement complete morphological disambiguation.

PoS tagging is much more difficult and complicated for agglutinative languages (e.g. Hungarian, Turkish, Finnish etc.) than in case of morphologically poor languages (e.g. English, Chinese etc.). The main problem is data sparseness: for example while an English word has about 4-6 different word forms, it has several hundred suffixed word forms in agglutinative languages. Thus, the tagset for Hungarian contains more than a thousand different tags, while this number for English is only a few dozen. There is also a big difference in the number of different word forms in a given corpus.

2.2. SMT decoding – theoretical background

Complete morphological disambiguation can be defined as a translation problem where the source language is the original text and the target language is the morphologically annotated one. It seemed to be a promising choice to use a statistical machine translation (SMT)-based system. Machine translation systems provide a mapping between two languages – be those languages natural or artificial ones – by learning transformation rules from a bilingual parallel corpus using unsupervised learning algorithms. Formally this can be described as follows: if S is a sentence in the source language, its optimal translation, $\hat{T}$ can be identified as

$$\hat{T} = \underset{T}{argmax}\ p(T|S) = \underset{T}{argmax}\ p(S|T)p(T) \quad (1)$$

where $\hat{T}$ maximizes a combination of the language model $p(T)$ and the translation model $p(S|T)$ scores (Jurafsky and

Martin, 2009).
Further on, the decoding model of the SMT method is isomorphic with the noisy-channel models that are close to the one used by PoS taggers based on a hidden Markov model (HMM). Therefore a mapping can be done: the language model ($P(T)$) is the tag transmission probability model and the translation model ($P(S|T)$) is the output (or lexical) probability model. The main difference between these models is the way how probabilities are estimated. The main reasons that motivated us to use an SMT framework, in particular the open-source Moses SMT toolkit (Koehn et al., 2007), to the task of morphological annotation were the following:

1. The Moses training chain is fast to create a translation model (i.e. a lexical probability model) from a given word aligned corpus. Furthermore, in contrast to usual HMM decoders, the translation model may contain long phrases, which allows the system to tag long sequences of words as one unit.

2. The language model (i.e. the tag transmission probability model) is trained with a modified Kneser-Ney smoothing (James, 2000), which has a state-of-the-art performance in the creation of language models.

3. The decoder (tagger) uses an efficient beam-search algorithm applying stack decoding. The advantage of this technique is that the decoding order is not severely left-to-right (in contrast to e.g. the strict left-to-right decoding applied in HMM taggers) due to the reordering ability of the SMT decoder. If the distortion penalty is turned off, far distance jumps are allowed for the decoder.

4. It is easy to integrate a morphological guesser or morphological analyzer as a pre-translator tool into the decoding process.

3. Earlier approaches

The first commonly used statistical taggers were based on Markov models (such as TnT (Brants, 2000) or HunPos (Halácsy et al., 2007)), then starting by Ratnaparkhi (Reynar and Ratnaparkhi, 1997) maximum entropy modeling became a popular methodology and is generally applied to numerous languages. Good examples are the adaptations of the Stanford tagger (Toutanova and Manning, 2000). There are plenty of other supervised learning methods that are shown to be performing well amongst different languages, such as Brill's transformation learner (Brill, 1995), the SVMTool (Giménez and Màrquez, 2004). This is based on Support Vector Machines or TreeTagger (Schmid, 1995), which uses decision trees.
The main advantage of our system is that it operates in a completely language- and tagset-independent manner. Moreover, previous tagger implementations have limited possibilities concerning the selection of decoding order and the units of translation. HMM-based taggers are restricted to left-to-right processing. When assigning an analysis to a word, the system can only rely on the analysis of its left-hand-side neighbors. The system performs a search for the best global analysis at the end of the sentence. The maximum-entropy-based algorithm implemented in e.g. the Stanford parser begins the analysis with the least ambiguous words, and then goes on with analysis of more ambiguous words. The process relies on the analyses of nearest neighbors.
In contrast, the decoder in the tagger described in this paper is in theory able to use arbitrarily long terms as separate translation units, and word analysis can take into account not only left-hand-side, but also right-hand-side neighbors. In addition, the language model used for implementing the tag-transition model – thanks to the improved Kneser-Ney smoothing – proved more effective than the trigram-model used by the HMM-based systems above.
Mora and Peiró (2007) were the first ones to use an SMT decoder as a tagger tool (Mora and Peiró, 2007). The system was designed only for part-of-speech tagging for English (not including lemmatization). A word-frequency-based model and an exception list of 11 word ending patterns were used to manage out-of-vocabulary (OOV) words. A similar approach is described by Laki (Laki, 2012), who applied this method for Hungarian. Parameters and models used by Mora and Peiró for English produced well below state-of-the-art results for Hungarian in a PurePos with morphological analyzer setting (Orosz and Novák, 2013). Due to the agglutinating nature of Hungarian, its analysis requires much more advanced techniques to handle the increased number of OOV words.
In this paper the modified version of Laki's system is described. Some important modifications need to be done to use an SMT as PoS tagger. These changes are presented shortly in the next section; while a longer account can be found in the paper of Laki et al. (Laki et al., 2013).

4. Description of the system

In this section the most important modifications of the SMT framework are described which distinguish the original one from a morphological annotator.

Generalized Lemmatization: For suffixing languages like Hungarian or Turkish, lemmas can be easily represented as a tuple that describes the transformation to be performed on the word form to get the lemma: $\langle cut\#paste \rangle$, where *cut* is how many characters are to be cut from the end of the string, and *paste* is the string to be joined to the end of the result of the cut operation. For example the Hungarian word *hazám* ('my homeland') – where the lemma is *haza* ('homeland') – can be represented as *'2#a'*. Thus, the morphological analysis that is to be identified for each word in a sentence is the lemma representation in conjunction with the morphosyntactic label in form *'2#a#[Nc-sn---s1]'*.

Monotone alignment: In the case of natural languages, unsupervised machine learning algorithms are used in an SMT system as word alignment tools. As a morphological disambiguation tool should use a one-to-one monotone mapping between tokens and their analyses, the Giza++ tool (Och and Ney, 2000) was replaced by a monotone mapping in HuLaPos2.

Phrase extraction: The main advantage of the Moses decoder is that it is able to translate longer phrases as one unit. The maximum length of phrases used in translation

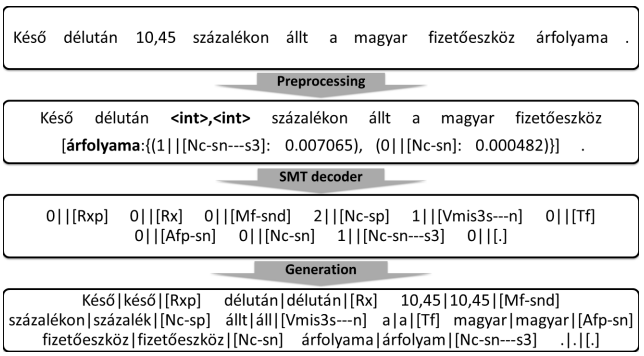

Figure 1: The workflow of HuLaPos2 presented on a Hungarian example *'Late afternoon the exchange rate of the Hungarian currency stood on 10.45 percent.'*

and the order of language models influence the quality of the system. The optimal setting was determined empirically – separately for each language–, where systems were tested using different length settings.

Handling numeric strings: To reduce data sparseness issues, digits, percentages and strings with only uppercase characters were replaced with generic symbols in the training corpus and in the input text as well.

Handling OOV words: To improve tagging and lemmatization of OOV words, a morphological guesser was created, which is similar to the one implemented in HunPos. The guesser learns the *(lemma transformation; tag)* pairs (TTP) for suffixes of rare words found in the training data. A reverse trie of suffixes is built, in which each node has a weighted list of TTPs. The guesser was integrated into the decoder in the following manner. The Moses decoder is able to read a set of predefined translations from the input, which can be used to handle words not seen in the training corpus. In this setup, hapaxes are left intact in the training corpus, while OOV words are pre-translated in the input. This is an optimal solution because the guesser can assign PoS tags and lemmas based on a smoothed interpolated model of various suffix lengths. Moreover, this solution does not force the system to default to a unigram tag transmission probability model when encountering an OOV word. This subsystem is to be used for handling rare unseen words, therefore it should be trained on rare words. The threshold for rare words was set empirically: the highest accuracy was obtained when it was set to 2, i.e. if the guesser is trained only on hapaxes.

Overview of the workflow: Figure 1 shows an analysis of an example sentence, where most of the handled phenomena are represented. The first step is preprocessing, where numeric variables are normalized ($10,5 \rightarrow <int>,<int>$). Thereafter unknown words are preannotated by the guesser (*árfolyama* $\rightarrow (1||[Nc{-}sn{-}{-}{-}s3] : 0.007065), (0||[Nc{-}sn] : 0.000482)$). In the next step translation is created by the Moses decoder from the preprocessed form to the TTPs. Finally, the output form $(token\#lemma\#tag)$ of HuLaPos2 system is generated.

5. Evaluation and results

Based on the available corpora two evaluation schemes were used. In the case of languages where the corpus does not contain lemmas (English, Portugese, Bulgarian), we measured only the accuracy of PoS tagging of our system. In cases where the lemmatization was also included in corpus (Hungarian, Croatian, Serbian), during evaluation we calculated the accuracy of the full morphological annotation including both lemmas and also morphosyntactic tags. We optimized the system for the accuracy of full word level morphological analysis. A detailed comparison of accuracy values is presented in Table 1.

Hungarian: As for Hungarian, PurePos2 (Orosz and Novák, 2013) is the state-of-the-art system. This is a HMM-based complete morphological disambiguation tool which can optionally use an integrated morphological analyzer, which uses the HUMor tagset.

Szeged Corpus 2 (Csendes et al., 2004) is used for the comparison with two different morphosyntactic code sets. The MSD annotation system (Erjavec, 2004) is the original scheme used for annotating this corpus, while HUMor morphosyntactic codes (Novák, 2003) are automatically assigned using conversion algorithms. Szeged Corpus 2 is the only large-scale freely available completely annotated corpus for Hungarian. The corpus contains manually checked and corrected annotation, and it has 64 395 segments (sentences), which contain 1 042 546 tokens. These cover 112 100 different word-form types. The corpus was divided into 10%-10%-80% development, test and training sets. System parameters were tuned on the development set and final testing was performed on the test set. HuLaPos2 was compared to PurePos with (PurePos MA) and without the integrated morphological analyzer. Results show that HuLaPos2 outperforms PurePos without the morphological analyzer, while it still produces better accuracy compared to PurePos with the MA in the case of PoS tagging.

Serbian and Croatian: Regarding Serbian and Croatian, Agić et al. (Agić et al., 2013) created a PoS tagging and lemmatization tool chain in 2013. Their system is a composition of HunPos (Halácsy et al., 2007) and the CST lemmatizer (Jongejan and Dalianis, 2009), which was trained with the SETimes.HR corpus (Croatian newspaper text from Southeast European Times). The SETimes.HR corpus is the dependency treebank of Croatian and Serbian. Both the Croatian and the Serbian part contain about 4000 manually lemmatized and morphosyntactically tagged sentences. The test sets size were 100-100 sentences. Results in table 1 show that in the case of PoS tagging HuLaPos2 outperformed Agić's system, however in the case of lemmatization it can not be said. In Serbian and Croatian inflected forms of words may change not only suffixes, but also prefixes with high frequency. Since our lemmatization tool uses suffix-based TTPs, therefore it is difficult to manage the prefixes with this model, which leads to worse performance of HuLaPos2 in lemmatiztion.

In case of Bulgarian, English and Portugese, systems were trained on available corpora without gold standard lemmatization, therefore the performance of only PoS tagging was compared (see table 2).

Bulgarian: Georgiev et al. (Georgiev et al., 2012) created

Language	System	Token accuracy		
		tagging	lemmatization	full
Hungarian (MSD)	**HuLaPos2**	**99.57%**	**97.24%**	96.84%
	PurePos	96.74%	96.35%	94.76%
Hungarian (HUMor)	**HuLaPos2**	**99.18%**	98.23%	97.62%
	PurePos	96.50%	96.27%	94.53%
	PurePos+MA	98.96%	**99.53%**	98.77%
Croatian	**HuLaPos2**	**93.25%**	96.21%	90.77%
	HunPos+CST	87.11%	**97.78%**	–
Serbian	**HuLaPos2**	**92.28%**	92.72%	86.51%
	HunPos+CST	85.00%	**95.95%**	–

Table 1: Comparing PoS, lemmatization and full morphological annotation values of HuLaPos2 with other well-performing systems

Language	System	Tagging accuracy
Bulgarian	TnT	92.53%
	machine learning	95.72%
	machine learning + lexicon	97.83%
	HuLaPos2	97.86%
	machine learning + lexicon + rules	**97.98%**
Portuguese	**HuLaPos2**	**93.20%**
	HMM based PoS tagger	92.00%
English	TnT	96.46%
	PBT (Mora and Peiró, 2007)	96.97%
	HuLaPos2	97.08%
	Stanford tagger 2.0	97.32%
	SCCN (Søgaard, 2011)	**97.50%**

Table 2: Comparison of results in the case of use of PoS tagging alone

a morphological disambiguation tool based on controlled training for Bulgarian. A morphological lexicon and linguistic rules were built into their system. The tool was trained and tested on BulTreeBank corpus (Chanev et al., 2007). The results in Table 2 show that the performance of HuLaPos2 significantly exceeds the performance of purely statistical-based tools. Despite that our system was not supported by any language specific tool, it outperforms a performance of system with morphological lexicon; furthermore the accuracy of HuLaPos2 gains on the results of Georgiev's best system (controlled training + lexicon + rules).

Portuguese: As regards Portuguese, Maia and Xexéo (2011) created an HMM-based PoS tagger in 2011. Their system was trained on the Floresta Sintá(c)tica Treebank (Freitas et al., 2008) (the first 10% was the test set, the remaining 90% the training set). Maia and Xexéo's system reaching 96.2% accuracy with a 39-tag tagset and 92.0% with a 257-tag tagset extended with inflexion information. HuLaPos2 was trained with the bigger tagset and with these parameters it outperformed their accuracy with more than 1%.

English: Concerning English, the WSJ subpart of the Penn Treebank (Marcus et al., 1993) was used with the commonly used division.[1] Table 2 shows the results of HuLaPos2 and four other systems. Firstly, we could observe that HuLaPos2 exceeded the TnT system and the system of Mora and Pieró (2007). Secondly, HuLaPos2 has comparable results to the state-of-the-art.

6. Conclusion

In this paper, we described a language-independent morphological annotation tool that is based on the Moses toolkit. It performs PoS tagging and lemmatization simultaneously employing a trie-based suffix guesser, which effectively handles the problem of OOV words, typical of morphologically rich languages. The performance of HuLaPos2 was compared to the state-of-the-art systems of five different languages. In the case of PoS tagging, our method outperforms available tools for all investigated languages, but English. Meanwhile its lemmatization accuracy is comparable to them. In the case of English, HuLaPos2 outperforms the TnT system and has a similar performance compared to the best ones.

Acknowledgment

This research was partially supported by the project grants TÁMOP–4.2.1./B–11/2-KMR-2011-0002 and TÁMOP–4.2.2./B–10/1-2010-0014.

[1] http://aclweb.org/aclwiki/index.php?title=POS_Tagging_(State_of_the_art)

7. References

Agić, v., Ljubešić, N., and Merkler, D. (2013). Lemmatization and Morphosyntactic Tagging of Croatian and Serbian. In *Proceedings of the 4th Biennial International Workshop on Balto-Slavic Natural Language Processing*, pages 48–57, Sofia, Bulgaria, August. Association for Computational Linguistics.

Brants, T. (2000). Tnt - a Statistical Part-of-Speech Tagger. In *Proceedings of the Sixth Applied Natural Language Processing (ANLP-2000)*, Seattle, WA.

Brill, E. (1995). Transformation-based error-driven learning and natural language processing: A case study in part-of-speech tagging. *Computational Linguistics*, 21:543–565.

Chanev, A., Simov, K., Osenova, P., and Marinov, S., (2007). *The BulTreeBank: Parsing and conversion*, volume 309 of *Current Issues in Linguistic Theory*, pages 321–330. John Benjamins, Amsterdam & Philadelphia.

Csendes, D., Csirik, J., and Gyimóthy, T., (2004). *The Szeged Corpus: A POS Tagged and Syntactically Annotated Hungarian Natural Language Corpus*, volume 3206 of *Lecture Notes in Computer Science*, pages 41–47. Springer Berlin / Heidelberg.

Erjavec, T. (2004). MULTEXT-East Version 3: Multilingual Morphosyntactic Specifications, Lexicons and Corpora. In *Fourth International Conference on Language Resources and Evaluation, LREC'04*, pages 1535–1538. ELRA.

Freitas, C., Rocha, P., and Bick, E. (2008). Floresta Sintá(c)tica: Bigger, thicker and easier. In *Proceedings of the 8th international conference on Computational Processing of the Portuguese Language*, PROPOR '08, pages 216–219, Berlin, Heidelberg. Springer-Verlag.

Georgiev, G., Zhikov, V., Simov, K. I., Osenova, P., and Nakov, P. (2012). Feature-rich part-of-speech tagging for morphologically complex languages: Application to bulgarian. In Daelemans, W., Lapata, M., and Màrquez, L., editors, *EACL*, pages 492–502. The Association for Computer Linguistics.

Giménez, J. and Màrquez, L. (2004). Svmtool: A general pos tagger generator based on support vector machines. In *In Proceedings of the 4th International Conference on Language Resources and Evaluation*, pages 43–46.

Halácsy, P., Kornai, A., and Oravecz, C. (2007). HunPos: An open source trigram tagger. In *Proceedings of the 45th Annual Meeting of the ACL*, pages 209–212, Stroudsburg. Association for Computational Linguistics.

James, F. (2000). Modified Kneser-Ney smoothing of n-gram models. Technical report.

Jongejan, B. and Dalianis, H. (2009). Automatic training of lemmatization rules that handle morphological changes in pre-, in- and suffixes alike. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 145–153, Suntec, Singapore, August. Association for Computational Linguistics.

Jurafsky, D. and Martin, J. H. (2009). *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition*. Prentice Hall series in artificial intelligence. Prentice Hall, Pearson Education International, Englewood Cliffs, NJ, 2nd edition.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., and Herbst, E. (2007). Moses: Open Source Toolkit for Statistical Machine Translation. In *Proceedings of the ACL 2007 Demo and Poster Sessions*, pages 177–180, Prague. Association for Computational Linguistics.

Laki, L. J., Orosz, G., and Novák, A. (2013). Hulapos 2.0 – decoding morphology. In *12th Mexican International Conference on Artificial Intelligence*, Mexico City, Mexico.

Laki, L. (2012). Investigating the Possibilities of Using SMT for Text Annotation. In Simões, A., Queirós, R., and da Cruz, D., editors, *1st Symposium on Languages, Applications and Technologies*, volume 21 of *OpenAccess Series in Informatics (OASIcs)*, pages 267–283, Dagstuhl, Germany. Schloss Dagstuhl–Leibniz-Zentrum für Informatik.

Marcus, M. P., Santorini, B., and Marcinkiewicz, M. A. (1993). Building a large annotated corpus of english: The Penn Treebank. *Computational Linguistics*, 19(2):313–330.

Mora, G. G. and Peiró, J. A. S. (2007). Part-of-Speech tagging based on machine translation techniques. In *Proceedings of the 3rd Iberian conference on Pattern Recognition and Image Analysis, Part I*, IbPRIA '07, pages 257–264, Berlin, Heidelberg. Springer-Verlag.

Novák, A. (2003). What is good Humor like? In *I. Magyar Számítógépes Nyelvészeti Konferencia*, pages 138–144, Szeged. SZTE.

Och, F. J. and Ney, H. (2000). Improved statistical alignment models. pages 440–447, Hongkong, China, October.

Orosz, G. and Novák, A. (2013). PurePos 2.0: a hybrid tool for morphological disambiguation. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing*, pages 539–545, Hussal, Bulgaria.

Reynar, J. C. and Ratnaparkhi, A. (1997). 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. Association for Computational Linguistics.

Schmid, H. (1995). Improvements In Part-of-Speech Tagging With an Application To German. In *In Proceedings of the ACL SIGDAT-Workshop*, pages 47–50.

Søgaard, A. (2011). Semisupervised condensed nearest neighbor for part-of-speech tagging. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers - Volume 2*, HLT '11, pages 48–52, Stroudsburg, PA, USA. Association for Computational Linguistics.

Toutanova, K. and Manning, C. D. (2000). Enriching the

knowledge sources used in a maximum entropy part-of-speech tagger. In *Proceedings of the 2000 Joint SIGDAT conference on Empirical methods in natural language processing and very large corpora: held in conjunction with the 38th Annual Meeting of the Association for Computational Linguistics - Volume 13*, EMNLP '00, pages 63–70, Stroudsburg, PA, USA. Association for Computational Linguistics.