Morphosyntactic Disambiguation in an Endangered Language Setting

Ens, Jeff

Linköping University Electronic Press
2019

Ens , J , Hämäläinen , M , Rueter , J & Pasquier , P 2019 , Morphosyntactic Disambiguation in an Endangered Language Setting . in M Hartmann & B Plank (eds) , 22nd Nordic Conference on Computational Linguistics (NoDaLiDa) : Proceedings of the Conference . Linköping Electronic Conference Proceedings , no. 167 , NEALT Proceedings Series , no. 42 , Linköping University Electronic Press , Linköping , pp. 345-349 , Nordic Conference on Computational Linguistics , Turku , Finland , 30/09/2019 .

http://hdl.handle.net/10138/305872

Morphosyntactic Disambiguation in an Endangered Language Setting

Jeff Ens[♠] **Mika Hämäläinen**[♢] **Jack Rueter**[♢] **Philippe Pasquier**[♠]
[♠] School of Interactive Arts & Technology, Simon Fraser University
[♢] Department of Digital Humanities, University of Helsinki
jeffe@sfu.ca, mika.hamalainen@helsinki.fi,
jack.rueter@helsinki.fi, ppa12@sfu.ca

Abstract

Endangered Uralic languages present a high variety of inflectional forms in their morphology. This results in a high number of homonyms in inflections, which introduces a lot of morphological ambiguity in sentences. Previous research has employed constraint grammars to address this problem, however CGs are often unable to fully disambiguate a sentence, and their development is labour intensive. We present an LSTM based model for automatically ranking morphological readings of sentences based on their quality. This ranking can be used to evaluate the existing CG disambiguators or to directly morphologically disambiguate sentences. Our approach works on a morphological abstraction and it can be trained with a very small dataset.

1 Introduction

Most of the languages in the Uralic language family are endangered. The low number of speakers, limited linguistic resources and the vast complexity in morphology typical to these languages makes their computational processing quite a challenge. Over the past years, a great deal of work related to language technology for endangered Uralic languages has been released openly on the Giellatekno infrastructure (Moshagen et al., 2014). This includes lexicographic resources, FST (finite-state transducer) based morphological analyzers and CG (constraint grammar) disambiguators.

Despite being a great resource, the Giellatekno infrastructure has tools and data originating from different sources by different authors. Recent research conducted with the resources for Komi-Zyrian, Skolt Sami, Erzya and Moksha has identified a need for proper evaluation of the resources available in the infrastructure, as they are not free of errors (Hämäläinen et al., 2018; Hämäläinen, 2018).

This paper presents a method to learn the morphosyntax of a language on an abstract level by learning patterns of possible morphologies within sentences. The resulting models can be used to evaluate the existing rule-based disambiguators, as well as to directly disambiguate sentences. Our work focuses on the languages belonging to the Finno-Permic language family: Finnish (fin), Northern Sami (sme), Erzya (myv) and Komi-Zyrian (kpv). The vitality classification of the three latter languages is definitely endangered (Moseley, 2010).

2 Motivation

There are two main factors motivating this research. First of all, data is often very scarce when dealing with endangered Uralic langauges. Apart from Northern Sami, other endangered Uralic languages may have a very small set of annotated samples at best, and no gold standard data at worst. As a result, evaluating disambiguated sentences can often only be conducted by consulting native speakers of the language or by relying on the researcher's own linguistic intuition.

Secondly, canonical approaches involving Part-of-Speech (POS) tagging will not suffice in this context due to the rich morphology of Uralic languages. For example the Finnish word form *voita* can be lemmatized as *voi* (the singular partitive of butter), *vuo* (the plural partitive of fjord), *voittaa* (the imperative of win) or *voitaa*[1] (the connegative form of spread butter).

The approach described in this paper, addresses these two issues, as we use a generalized sentence representation based on morphological tags to capture morphological patterns. Moreover, our

[1] A non-standard form produced by the Finnish analyzer

models can be trained on low resource langauges, and models that have been trained on high resource languages can be applied to low or no resource languages with reasonable success.

3 Related Work

The problem of morphological tagging in the context of low-resource languages has been approached using parallel text (Buys and Botha, 2016). From the aligned parallel sentences, their Wsabie-based model can learn to tag the low-resource language based on the morphological tags of the high-resource language sentences in the training data. A limitation of this approach is the morphological relatedness of the high-resource and low-resource languages.

A method for POS tagging of low-resource languages has been proposed by Andrews et al. (2017). They use a bi-lingual dictionary between a low and high-resource language together with monolingual data to build cross-lingual word embeddings. The POS tagger is trained on an LSTM neural network, and their approach performs consistently better than the other benchmarks they report.

Lim et al. (2018) present work conducted on syntactically parsing Komi-Zyrian and Northern Sami using multilingual word-embeddings. They use pretrained word-embeddings for Finnish and Russian, and train word-embeddings for the low-resource languages from small corpora. These individual word-embeddings are then projected into a single space by using bilingual dictionaries. The parser was implemented as an LSTM model and it performed better in a POS tagging task than in predicting syntactic relations. The key finding for our purposes is that including a related high-resource language (Finnish in this case) improved the accuracy.

DsDs (Plank and Agić, 2018) is a neural network based POS tagger for low-resource languages. The idea is to use a bi-LSTM model to project POS tags from one language to another with the help of word-embeddings and lexical information. In a low-resource setting, they find that adding word-embeddings boosts the model, but lexical information can also help to a smaller degree.

Much of the related work deals with POS tagging. However, as the Uralic languages are morphologically rich, a full morphological disambiguation is needed in order to improve the performance of higher-level NLP tools. In addition, we do not want to assume bi-lingual parallel data or access to word embeddings as we want our approach to be applicable for truly endangered languages with extremely limited resources.

4 The Rule-based Tools and Data

We use the morphological FST analyzers in the Gieallatekno infrastructure to produce morphological readings with UralicNLP (Hämäläinen, 2019). They operate on a word level. This means that for an input word form, they produce all the possible lemmas together with their parts-of-speech and morphological readings, without any weights to indicate which reading is the most probable one.

The existing CG disambiguators get the morphological readings produced by the FST for each word in a sentence and apply their rules to remove the non-possible readings. In some cases, a CG disambiguator might produce a fully disambiguated sentence, however these models are often unable to resolve all morphological ambiguity.

In this paper, we use the UD Treebanks for our languages of interest. For Finnish, we use Turku Dependency Treebank (Haverinen et al., 2014) with 202K tokens (14K sentences). The Northern Sami Treebank (Sheyanova and Tyers, 2017) is the largest one for the endangered languages with 26K tokens (3K sentences). For Komi-Zyrian, we use the Komi-Zyrian Lattice Treebank (Partanen et al., 2018) of 2K tokens (189 sentences) representing the standard written Komi. The Erzya Treebank (Rueter and Tyers, 2018) is the second largest endangered language one we use in our research with 15k tokens (1,500 sentences).

5 Sentence Representation

We represent each word as a non-empty set of morphological tags. This representation does not contain the word form itself nor its lemma, as we aim for a more abstract level morphological representation. This representation is meant to capture the possible morphologies following each other in a sentence to learn morphosyntactic inter-dependencies such as agreement rules. This level of abstraction makes it possible to apply the learned structures for other morphosyntactically similar languages.

As we are looking into morphosyntax, we train our model only with the morphosyntactically rele-

vant morphological tags. These are case, number, voice, mood, person, tense, connegative and verb form. This means that morphological tags such as clitics and derivational morphology are not taken into account. We are also ignoring the dependency information in the UD Treebanks as dependencies are not available for text and languages outside of the Treebanks due to the fact that there are no robust dependency parsers available for many of the endangered Uralic language.

Each sentence is simply a sequence of morphological tag sets, represented as a sequence of integers with a special token `SP` demarcating spaces between words. For example the sentence "Nyt on lungisti ottamisen aika." (now it is time to relax), is encoded as $[150, \texttt{SP}, 121, 138, 168, 178, 205, 214, 221, \texttt{SP}\ 150, \texttt{SP}, 25, 138, 158, \texttt{SP}, 31, 138, 158, \texttt{SP}, 165]$.

Equation 1 is used to measure the distance between two sentences containing n words, where x_i denotes set of morphological tags associated with the i^{th} word in x, $||\cdot||$ denotes the number of elements in a set, and $\triangle$ denotes the symmetric difference of two sets. This distance measure is used to approximate the quality of different readings, based on the assumption that the quality of a reading decreases as its distance from the gold standard sentence increases.

$$\texttt{distance}(a, b) = \sum_{i=1}^{n} ||a_i \triangle b_i|| \qquad (1)$$

6 Model

We implement our models using Keras (Chollet et al., 2015), which are trained to rank two sentences encoded as described in Section 5. The model is comprised of a Long Short Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997) layer π and a feed-forward layer ϕ. Given two sentences a and b, the LSTM layer is used to produce the n-dimensional vectors $\pi(a)$ and $\pi(b)$, which are concatenated and passed through the feed-forward layer to produce a single scalar value $\phi(\pi(a), \pi(b))$ indicating the preferred sentence. We train each model with early stopping based on the validation accuracy with a patience of 10 epochs. We use the Adam optimizer (Kingma and Ba, 2014), train the model with batches of size 32, and set $n = 128$.

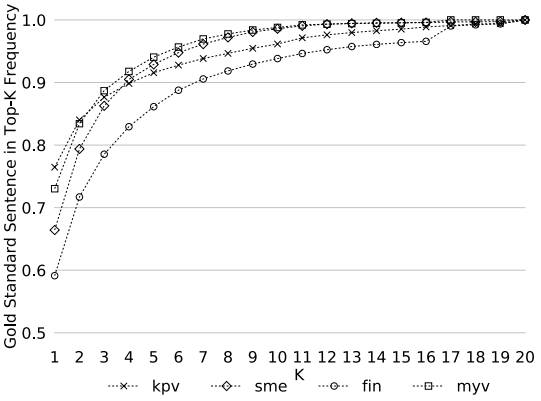

Figure 1: The frequency with which the gold standard sentence is ranked in the top-k with 1000 trials per model averaged over 10 data splits.

7 Evaluation

We produce all the morphological readings for each word in a gold standard sentence (`GSS`) using FST analyzers, and construct incorrect sentences (`INS`) of varying quality by randomly selecting a reading for each word. In order to provide a detailed evaluation, we categorize each sentence based on their distance from the `GSS` using the ranges $[[0, 1), [1, 10), [10, 20), [20, +\infty)]$, which we will refer to as categories G,1,2, and 3. By construction, category G only contains `GSS`. These ranges were chosen so that each bin contains approximately the same number of sentences. We measure the accuracy of the model for each of the $\binom{4}{2} = 6$ possible types of comparisons between sentence categories. To create training, validation and testing data, the set of `GSS` are randomly split before generating `INS`. In cases where two languages are used to train the model, the training data consists of an even number of comparisons from each language to ensure that a larger language does not dominate a smaller language.

Since we are interested in exploring the viability of using high resource to disambiguate low resource languages, we evaluate the models by training on each language and each possible combination of languages, resulting in $4+\binom{4}{2} = 10$ distinct models.

8 Results

In order to ensure that our results are not the artifact of a particular data split, we train each model on 10 random splits of the data. The average ac-

	kpv						sme						fin						myv					
model	Gv1	Gv2	Gv3	1v2	1v3	2v3	Gv1	Gv2	Gv3	1v2	1v3	2v3	Gv1	Gv2	Gv3	1v2	1v3	2v3	Gv1	Gv2	Gv3	1v2	1v3	2v3
kpv	**.93**	**.97**	.97	**.79**	.95	.77	.53	.62	.66	.56	.60	.54	.62	.65	.68	.59	.64	.58	.65	.72	.77	.64	.71	.62
myv	.59	.68	.68	.66	.76	.62	.18	.13	.10	.40	.32	.40	.66	.65	.68	.56	.61	.56	**.95**	**.99**	**.99**	.78	.92	.76
sme	.65	.70	.70	.55	.56	.52	**.93**	**.98**	**.99**	**.73**	**.89**	**.71**	.57	.59	.61	.56	.58	.56	.22	.14	.10	.39	.31	.40
fin	.49	.60	.66	.60	.70	.60	.44	.58	.68	.58	.67	.57	**.88**	**.95**	**.98**	.72	**.85**	.70	.70	.74	.74	.62	.67	.59
kpv+myv	.92	**.97**	**.99**	**.79**	**.96**	**.79**	-	-	-	-	-	-	-	-	-	-	-	-	.92	.98	**.99**	**.80**	**.93**	**.77**
kpv+fin	.90	.95	**.99**	.77	.95	.78	-	-	-	-	-	-	.87	.94	.97	.72	**.85**	.69	-	-	-	-	-	-
kpv+sme	.91	.95	**.97**	.73	.89	.69	.91	.97	**.99**	**.73**	.86	.69	-	-	-	-	-	-	-	-	-	-	-	-
myv+fin	-	-	-	-	-	-	-	-	-	-	-	-	.83	.90	.94	.69	.82	.69	.93	.98	**.99**	.79	.91	.75
myv+sme	-	-	-	-	-	-	.89	.95	.97	**.73**	.86	.70	-	-	-	-	-	-	.86	.94	.96	.75	.87	.72
sme+fin	-	-	-	-	-	-	.90	.96	.98	**.73**	.88	**.71**	.86	.93	.97	**.73**	**.85**	**.71**	-	-	-	-	-	-

Table 1: Model accuracy averaged over 10 data splits with 1000 trials per model.

curacy across data splits is shown in in Table 1, where the accuracy of a single model with respect to a single comparison type is calculated based on 1000 comparisons. The mean standard error was 0.008, and the maximum standard error was 0.054 for these measurements. Figure 1 shows the percentage of times the GSS is ranked in the top-k sentences, given a set of 20 sentences containing 19 randomly selected INS. The $\binom{20}{2} = 190$ pairwise rankings are aggregated using iterative Luce Spectral Ranking algorithm (Maystre and Grossglauser, 2015).

9 Discussion and Future Work

The results in Table 1 demonstrate that our models are as effective for extremely low resource languages like Komi-Zyrian (kpv) as they are for high resource languages like Finnish (fin). Furthermore, there is evidence that training on a higher resource langauge that is genealogically related to a low resource langauge is a viable option. For example, the models trained on Finnish (fin) data performed relatively well when tested on the Erzya (myv) data. In cases where languages are not genealogically close to each other, such as Northern Sami (sme) and Erzya (myv), models perform very poorly when trained on one of these langauges and tested on another.

According to the results, the most difficult comparisons are 1v2 and 2v3. Since Equation 1 is only a proxy for sentence quality, it is possible that for some number of comparisons category 1 sentences are actually lower quality than category 2 sentences. In contrast, Gv1, Gv2, and Gv3 are comparisons against GSS, which are guaranteed to be correct. Consequently, it seems reasonable to conclude that this decrease in performance is partially due to deficiencies in measuring sentence quality.

Figure 1 demonstrates that pairwise rankings can be aggregated to reliably rank sentences based on their quality, as the GSS was frequently in the top-k sentences for small values of k. For example, the kpv, myv and sme models ranked the GSS in the top 3 roughly 86 percent of the time.

Future work may involve experiments with very closely related languages. For instance, out of 9 Sami languages, North Sami is the only one with a UD Treebank. Testing the performance of our system on the other Sami languages while training on North Sami is one of our goals for the future research. However, as due to the lack of gold annotated data, we need to recruit nearly native or native speakers with linguistic knowledge to evaluate our system. This is a time consuming task and it is outside of the scope of this paper.

10 Conclusion

Uralic languages exhibit a high degree of morphological ambiguity, and resources for these languages are often limited, posing difficulties for traditional methods that have been employed successfully on other languages. In order to mitigate these issues, we proposed a representation based on the morphological tags associated with each word in a sentence.

Our experimental results demonstrate that an LSTM based model can accurately rank alternate readings of a single sentence, even when the model is trained on an extremely low-resource language. This technique requires much less effort than developing complex rule-based grammar models for an endangered languages, as our model can be trained on a small set of gold-standard examples. Furthermore, a trained model can be used to disambiguate morphological readings produced by an FST analyzer or to evaluate the output of a CG model.

References

Nicholas Andrews, Mark Dredze, Benjamin Van Durme, and Jason Eisner. 2017. https://doi.org/10.18653/v1/P17-1095 Bayesian modeling of lexical resources for low-resource settings. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1029–1039. Association for Computational Linguistics.

Jan Buys and Jan A. Botha. 2016. https://doi.org/10.18653/v1/P16-1184 Cross-lingual morphological tagging for low-resource languages. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1954–1964. Association for Computational Linguistics.

François Chollet et al. 2015. Keras. `https://keras.io`.

Mika Hämäläinen. 2018. Extracting a Semantic Database with Syntactic Relations for Finnish to Boost Resources for Endangered Uralic Languages. In *Proceedings of the Logic and Engineering of Natural Language Semantics 15 (LENLS15)*.

Mika Hämäläinen. 2019. UralicNLP: An NLP library for Uralic languages. *Journal of Open Source Software*, 4(37):1345. 10.21105/joss.01345.

Mika Hämäläinen, Liisa Lotta Tarvainen, and Jack Rueter. 2018. Combining Concepts and Their Translations from Structured Dictionaries of Uralic Minority Languages. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*. European Language Resources Association (ELRA).

Katri Haverinen, Jenna Nyblom, Timo Viljanen, Veronika Laippala, Samuel Kohonen, Anna Missilä, Stina Ojala, Tapio Salakoski, and Filip Ginter. 2014. https://doi.org/10.1007/s10579-013-9244-1 Building the essential resources for finnish: the turku dependency treebank. *Language Resources and Evaluation*, 48(3):493–531.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. https://doi.org/10.1162/neco.1997.9.8.1735 Long short-term memory. *Neural Comput.*, 9(8):1735–1780.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

KyungTae Lim, Niko Partanen, and Thierry Poibeau. 2018. Multilingual dependency parsing for low-resource languages: Case studies on north saami and komi-zyrian. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

Lucas Maystre and Matthias Grossglauser. 2015. http://dl.acm.org/citation.cfm?id=2969239.2969259 Fast and accurate inference of plackett-luce models. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'15, pages 172–180, Cambridge, MA, USA. MIT Press.

Christopher Moseley, editor. 2010. *Atlas of the World's Languages in Danger*, 3rd edition. UNESCO Publishing. Online version: http://www.unesco.org/languages-atlas/.

Sjur Moshagen, Jack Rueter, Tommi Pirinen, Trond Trosterud, and Francis M. Tyers. 2014. http://www.lrec-conf.org/proceedings/lrec2014/workshops/LREC2014-Workshop-CCURL2014-Proceedings.pdf Open-Source Infrastructures for Collaborative Work on Under-Resourced Languages. The LREC 2014 Workshop "CCURL 2014 - Collaboration and Computing for Under-Resourced Languages in the Linked Open Data Era".

Niko Partanen, Rogier Blokland, KyungTae Lim, Thierry Poibeau, and Michael Rießler. 2018. The first komi-zyrian universal dependencies treebanks. In *Proceedings of the Second Workshop on Universal Dependencies (UDW 2018)*, pages 126–132.

Barbara Plank and Željko Agić. 2018. http://aclweb.org/anthology/D18-1061 Distant supervision from disparate sources for low-resource part-of-speech tagging. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 614–620. Association for Computational Linguistics.

Jack Rueter and Francis Tyers. 2018. Towards an Open-Source Universal-Dependency Treebank for Erzya. In *Proceedings of the Fourth International Workshop on Computatinal Linguistics of Uralic Languages*, pages 106–118.

Mariya Sheyanova and Francis M. Tyers. 2017. Annotation schemes in north sámi dependency parsing. In *Proceedings of the 3rd International Workshop for Computational Linguistics of Uralic Languages*, pages 66–75.