The University of Illinois submission to the WMT 2015 Shared Translation Task

Lane Schwartz, Bill Bryce, Chase Geigle, Sean Massung, Yisi Liu, Haoruo Peng, Vignesh Raja, Subhro Roy and **Shyam Upadhyay**
University of Illinois at Urbana-Champaign
Urbana, IL 61801, USA
lanes@illinois.edu

Abstract

In this year's WMT translation task, Finnish-English was introduced as a language pair of competition for the first time. We present experiments examining several variations on a morphologically-aware statistical phrase-based machine translation system for translating Finnish into English. Our system variations attempt to mitigate the issue of rich agglutinative morphology when translating from Finnish into English. Our WMT submission for Finnish-English preprocesses Finnish data with omorfi (Pirinen, 2015), a Finnish morphological analyzer. We also present results for two other language pairs with morphologically interesting source languages, namely German-English and Czech-English.

1 Introduction

Students enrolled in the Spring 2015 graduate-level course in statistical machine translation (MT) at the University of Illinois were invited to develop MT systems within the context of the 2015 Workshop on Statistical Machine Translation (WMT) shared translation task. Each group of 2-3 students chose one language pair, developed a baseline MT system for that language pair using Moses (Koehn et al., 2007), and chose one specific linguistic dimension along which to experiment. In this work, we present the results of four groups of experiments — two Finnish-English (§3.1 and §3.2), and one each for Czech-English (§4) and German-English (§5).

The first author was the instructor, and the subsequent authors were students in the work described here.

2 Methodology

We use the current stable release (v3) of Moses, a state-of-the-art statistical phrase-based machine translation system.

We trained translation models using the Europarl corpus (Koehn, 2005), using the latest available versions (v7 for German-English and Czech-English, and v8 for Finnish-English), as well as the Common Crawl corpus and News Commentary (v10) corpus for German-English and Czech-English, and the Wiki Headlines corpus for Finnish-English.

We trained a back-off language model (LM) with modified Kneser-Ney smoothing (Katz, 1987; Kneser and Ney, 1995; Chen and Goodman, 1998) on the English Gigaword v5 corpus (Parker et al., 2011) using `lmplz` from KenLM (Heafield et al., 2013).

3 Finnish-English

We tried various morphological tokenization schemes on the *source* language (Finnish) in order to mitigate its strong agglutination. The *target* language (English) was tokenized with the default Moses tokenizer script.

3.1 Finnish tokenization using Morfessor and word-lattices

We begin by adapting the lattice technique of Dyer et al. (2009) to Finnish. We train a standard phrase-based machine translation model on a new corpus: on the source side we concatenate the original data with its one-best segmentation according to a Morfessor (Creutz and Lagus, 2007) model trained on the original data, and on the target side we simply concatenate it with itself. The result is a corpus that is twice as long as the original data, but that aligns both segmented and unsegmented Finnish sentences with their English counterparts. This ensures that we will have phrases in our phrase

Proceedings of the Tenth Workshop on Statistical Machine Translation, pages 192–198,
Lisboa, Portugal, 17-18 September 2015.

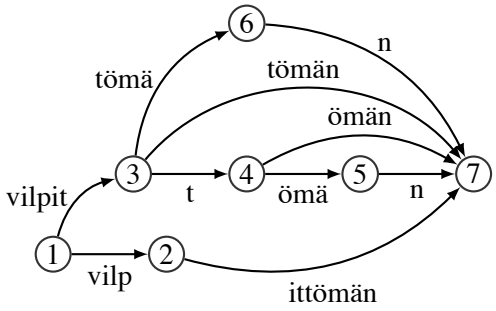

Figure 1: A word lattice that represents the top five segmentations for the Finnish word *vilpittömän*.

table that correspond with both the original unsegmented words as well as for individual morphemes.

At tuning and test time, we then decompose our input into a word lattice input that reflects the uncertainty of the decomposition of each word in the sentence (Dyer et al., 2008). We construct the lattice by considering the top five best segmentations for each word according to our Morfessor model. The start and end of each word in the original sentence is a node, and we place edges and nodes between the two such that the edge is labeled with a string output and its target is a node that represents the partial output of the word thus far. Each of the edges is also weighted with a certain probability, reflecting the likelihood of using that specific edge, given that we are at a specific node.

We calculate edge probabilities as follows. Let $p(v|u, \Theta)$ be the probability of going to node v given that we are at node u under the trained Morfessor model Θ (we only concern ourselves with the case where v is an adjacent to u). Let $\mathbf{s}$ be a segmentation for the current word, represented as a set of edges (n_1, n_2) through the graph. Then, we set

$$p(v \mid u, \Theta) = \frac{\sum_{\mathbf{s}:(u,v)\in\mathbf{s}} p(\mathbf{s} \mid \Theta)}{\sum_{\mathbf{s}':(u,v')\in\mathbf{s}'} p(\mathbf{s}' \mid \Theta)},$$

where the numerator is a summation of the Morfessor segmentation probabilities for segmentations that use the edge (u, v), and the denominator is a summation of the Morfessor segmentation probabilities for all segmentations that pass through node u.

However, Morfessor gives us log likelihood scores for its segmentations. Call these $\ell_{\mathbf{s}}$. We then compute the following, in order to avoid roundoff errors as much as possible:

$$p(v \mid u, \Theta) = \frac{\sum_{\mathbf{s}:(u,v)\in\mathbf{s}} 2^{\ell_{\mathbf{s}}-\ell_{max}}}{\sum_{\mathbf{s}':(u,v')\in\mathbf{s}'} 2^{\ell_{\mathbf{s}'}-\ell_{max}}},$$

where ℓ_{max} is the highest log likelihood segmentation for the current word. This can be seen as simply multiplying the numerator and denominator by the fixed constant $2^{-\ell_{max}}$. The code for performing this lattice generation is freely available online.[1] We use a Morfessor model trained on the Finnish side of the Europarl parallel training data with $\alpha = 0.5$.

System	LM	TM	BLEU	-cased
Baseline	5	5	**16.95**	**15.09**
Morfessor	5	8	15.67	14.88
Hiero	6	5	14.99	14.45
Lattice ($n = 2$)	6	8	14.67	14.00
Lattice ($n = 5$)	6	8	14.68	13.95

Table 1: Results for Finnish-English (§3.1).

Table 1 shows the output of our systems on the testing data from WMT 2015. We report the scores that were obtained from Moses evaluation scripts using multi-BLEU; the numbers in the shared task are slightly different as they use the NIST BLEU scripts. Our baseline is a phrase-based default Moses configuration with the 5-gram language model, and we found this outperformed a hierarchical phrase based configuration with the same maximum phrase length and a 6-gram language model. Among the segmentation methods, using a single one-best segmentation with Morfessor performed the best — the word lattice method had disappointing performance using either the top five or top two best segmentations for the lattice generation. We were unable to combine the word lattice and hierarchical phrase-based approaches together as Moses does not yet support these two features at the same time.

3.2 Finnish tokenization using omorfi

In addition to the experiments described above, we build three variations utilizing omorfi (Pirinen, 2015) to morphologically segment the Finnish data. We use omorfi to decompose each agglutinated Finnish word into its component morphemes and each morpheme to a default case or form. Inflectional morphemes which capture information

[1] `https://github.com/smassung/uiuc-wmt15/tree/master/chase`

```
Istuntokauden
Istuntokauden Istunto#kausi N Gen Sg
```

Figure 2: The first word of Finnish Europarl corpus, as processed by omorfi.

such as the person, number, tense, voice, and mood of verbs as well as the number and case of nouns is lost in the lemmatization, and therefore, when lemmatization has taken place, all of this information is lost to the system. Figure 2 illustrates this process; the token "Istuntokauden" is broken into two morpheme lemmas, separated by a "#" sign. We discard the inflectional information, which here denotes that the original token was a singular noun in genitive case.

As a baseline, we build a system using Moses and provided the data described above with none of the Finnish data having been processed by omorfi. Tuning was done using MERT (Och, 2003).

In the first variation (V1), all Finnish data is first segmented by omorfi. The intuition behind this technique is simply that there are more words in the target text than would align well with agglutinative words in the source text. By using the morphemes of the source language rather than the unsegmented words, the output source tokens might more easily align with the target tokens.

In the second variation (V2), the omorfi-segmented Finnish data from the first variation is concatenated with the unprocessed Finnish. Target language data is concatenated with itself in training to align each target sentence with both the unprocessed and morphologically-analyzed variations of its source sentence. The intuition here is that any Finnish tokens which are their own lemmas (i.e. do not inflect) will potentially align with the same target token twice, and will bear a stronger alignment probability than with other tokens in the translation model. Function words and adpositions would be among those which undergo such double alignment, and which may serve as anchors for the alignment of the entire sentence.

In the third variation (V3), the translation table created during the second variation is consulted during segmentation of the tuning and test data. If an original token could be found in the table before being broken into morphemes by omorfi, then that token is left unprocessed. If a token could not be found, then it was passed to omorfi and the morphemes returned replaced the token in the data.

System	LM	TM	BLEU	-cased
Baseline	5	5	16.14	15.25
V1-omorfi	5	5	14.79	14.00
V2-omorfi	5	5	15.14	14.32
V3-omorfi	5	5	**16.90**	**15.98**

Table 2: Results for Finnish-English (§3.2).

The resulting tuning and testing datasets are thus partially analyzed for morphemes. In this way, more common Finnish agglutinations are retained while less common ones are broken into potentially more common individual morphemes.

Results are shown in Table 2. Only V3 performed better than the baseline of using default Moses tokenization for Finnish. This variation comes closest to a balance between alignment with shorter target phrases — achieved by breaking down agglutinative words into morphemes — and retaining what inflectional information can be retained — since unprocessed and therefore unlemmatized words retain all grammatical inflection.

3.2.1 Variation 1: All data fully processed by omorfi

For the first variation on our system, we pass to omorfi all of the Finnish data described above used for training, tuning, and testing. Therefore, for each token in the text, either the lemma of the original token was returned by omorfi if the token was not found to be an agglutination of stem and morphemes, or, if the token was found to be an agglutination, a lemmatized token of each morpheme was returned, and these new tokens stood in place of the agglutinative token found in the original text.

The intuition behind this technique is simply that there are more words in the target text than would align well with agglutinative words in the source text. By creating more tokens out of the original source tokens, the smaller source tokens might more easily align with the target tokens. The new tokens returned by omorfi were always present in the source text in their non-lemma forms, but because the same morpheme could be added to different stems, the unique word formation may hide a relation between the appearance of that morpheme in a source sentence and a single word of English in the target sentence.

Using only source data which has been fully processed by omorfi in the training, tuning, and testing stages, BLEU scores were 14.00 (case-sensitive) and 14.79 (case-insensitive), that is 1.25 and 1.35

points below the baseline respectively.

3.2.2 Variation 2: Concatenated original source data and omorfi-processed data

For the second variation on our system, we used the same omorfi-processed Finnish data which was used for the first variation. This time, however, the omorfi-processed training, tuning, and testing data was concatenated with the original training, tuning, and testing data respectively. So for example, the data used for training was the original set of sentences from Europarl, followed by the same set of sentences but processed by omorfi as described above. Each of the training, tuning, and testing sets therefore contained exactly twice as many sentences as the original testing data. Likewise, the set of target sentences in each case was twice as many, but the target data was not processed for morphology, such that the second half of the target language training, tuning, and testing sets was exactly the same as the first half.

Designing the datasets in this way effected that, in the case of alignment for example, both the original Finnish sentence was aligned with the English as well as the omorfi-processed Finnish sentence. The intuition here is that Finnish tokens which are their own lemmas (i.e. do not inflect) will potentially align with the same target token twice, and will bear a stronger alignment proability than other tokens in the translation model. Function words and adpositions would be among those which undergo such double alignment, and which may serve as anchors for the alignment of the entire sentence.

For all other words — those for which omorfi returns morphologically analyzed output - two potentially useful alignments could be formed: First, there would be an alignment of the unprocessed source token with several target tokens, and so a phrasal alignment in which the English word aligns with the agglutinative word containing the proper morpheme. Second, there would be an alignment closer to one-to-one between the target word and the proper morpheme lemma returned by omorfi. Concatenating the unprocessed training, tuning, and testing sets in the source language with the omorfi-processed training, tuning, and testing sets respectively resulted in BLEU scores of 14.32 (case-sensitive) and 15.14 (case-insensitive), that is 0.93 and 1.00 points below the baseline respectively.

3.2.3 Variation 3: Consultation of the baseline translation table

For the third and final variation of our system, we preprocess the tuning and testing sets in the source language by consulting the translation table created for the second variation. For each token in the Finnish tuning and testing data, the translation table was consulted for the presence of that token as a unigram. If the token was found in the translation table, then it was rendered as is in the output of this step. If the token was not found in the translation table, then the token was passed to omorfi and the resulting morpheme lemmas were rendered as output. The resulting tuning and testing sets, therefore contained either an agglutinative form as found in the original Finnish or a processed string of morpheme lemmas (or perhaps simply the lemma) returned by omorfi from the original token, but not both.

The intuition here was to overcome the lemmatization process which occurs from passing all of the data through omorfi. It may be the case that different inflections of the same lemma tune better to different English words, but the lemmatization process effects that different English words tune to the same Finnish lemma, causing confusion. Leaving known inflected forms in the tuning and testing data gives this variation an advantage over the first variation. By tuning and testing on known tokens and morphologically analyzing unknown tokens in these datasets, the resulting BLEU scores were 15.98 (case-sensitive) and 16.90 (case-insensitive), 0.73 and 0.76 points above the baseline respectively.

4 Czech-English

For Czech-English, we train baseline phrase-based systems with no special handling of Czech morphology. We also consider experimental variants in which Czech words are morphologically segmented. We use Morphessor (Creutz and Lagus, 2007) for morphological segmentation.

Finally, we consider a re-ranking technique based on the degree of commonality between parts-of-speech (POS) in each source sentence and each respective translation of that source sentence. To this end, we use MorphoDiTa (Straková et al., 2014) and the Stanford CoreNLP toolkit (Manning et al., 2014) to POS tag the Czech and English sentences, respectively. We next construct a dictionary that maps POS tags from one language to tags

in the other. After translating with Moses, each English translation in the n-best list is augmented with a POS intersection score, and rerank taking this new score into account. We define the POS intersection score as simply the number of identical POS tags between a Czech sentence and the hypothesized English translation.

System	BLEU	BLEU-c
Moses trained on Europarl	18.59	17.72
Moses trained on Europarl, Common Crawl and News Commentary	20.69	19.83
Stemming as preprocessing, Moses trained on Europarl	17.88	17.08
Morfessor trained on Europarl, Moses trained on Europarl	16.48	15.74
POS intersection, Moses trained on Europarl	15.68	13.46
Morfessor trained on Europarl, POS intersection, Moses trained on Europarl	13.43	13.74

Table 3: Results for Czech into English.

5 German-English and English-German

For German-English and English-German, we focus primarily on the effects of source clause reordering transformations. In this approach, we transform source language **s** into **s**′, such that the clause structure of sentences in **s**′ more closely follow the clause structure of target language **t**.

5.1 English to German

With the goal of restructuring English source sentences to have more German-like structure, we define the following transformation rules:

1. Detect all clauses in a sentence which might require transformation. We selected spans of text, which were labeled as S or SBAR by the parser. We do not include clauses which begin with "to".
2. For each clause, we apply the following rules in order :
   (a) If there exists a verb phrase (detected by a shallow parser) with "to", we move the remaining portion of the verb phrase (starting with token "to") to the end of the clause.
   (b) If there exists a verb phrase (detected by a shallow parser) with a token with VBN part of speech tag, we move the remaining portion of the verb phrase (starting with VBN token) to the end of the clause.
   (c) If there exists a verb phrase (detected by a shallow parser) starting with a modal verb, we leave the modal verb but move the rest of the verb phrase to the end of the clause.

We used a state-of-the-art shallow parser (Punyakanok and Roth, 2001) in conjunction with a constituent parser (Socher et al., 2013) to implement the above transformation rules. For the purposes of the English-German language pair, we pre-process all English data into equivalent English′ data using the above transformation rules.

We train a German language model on the German side of the Europarl, Common Crawl, and News Commentary corpora, and a translation model on the English′-German Europarl corpus. Our development set for tuning was the WMT newstest data from 2008–2014. Results for the WMT newtest-2015 data set under the baseline (en-de) and restructured (en′-de) conditions are shown in Table 4.

System	BLEU	BLEU-cased	TER
en-de	16.6	16.3	0.933
en′-de	17.9	17.2	0.731

Table 4: Results for English and English′ translated into German.

5.2 German to English

Holmqvist et al. (2011) report improvements on German-English when modifying German text to be more like English. To this end, we utilize a subset of the clause restructuring rules (rules 4 & 6) from Collins et al. (2005):

- If a finite verb (VVFIN) and a particle (PTKVZ) are found in the same clause (subtree labeled as S), then move the particle to precede the verb.
- Before applying rule 6, we first remove all internal VP nodes, and replace them by their

children in the tree. Then, for every clause which dominates a finite verb, infinitival verb and a negative particle (PTKNEG), then the negative particle is moved to directly follow the finite verb.

We used the Stanford Parser (Manning et al., 2014) for parsing German sentences and then applied the relevant rules. The reordered sentences were the yield of the transformed tree. The reordered sentences were then segmented using the jWordSplitter [2] for compound splitting.

We train an English 6-gram language model on the Gigaword corpus, and a translation model on the German′-English Europarl corpus. Our development set for tuning was the WMT newstest data from 2008–2014. Results for the WMT newtest-2015 data set under the baseline (de-en) and restructured (de′-en) conditions are shown in Table 5.

System	BLEU	BLEU-cased	TER
de-en	21.4	22.2	0.938
de′-en	24.9	23.8	0.641

Table 5: Results for German and German′ translated into English.

6 Discussion and Conclusion

Overall, tackling the rich morphology of Finnish proved to be effective in improving upon the baseline, but not by much, and only in the case where the translation model could be consulted as to whether source words in the tuning and testing data were known.

The variation of our Finnish-English system in §3.2.1 breaks down the Finnish data into those components which make up the agglutinated words, treating the morphemes, rather than the original tokens, as the words. In teasing out the morphemes from the original data, more individual word alignments can be created between source and target tokens, but inflectional data such as the case of nouns and the person and tense of verbs, is lost. In this case, different English tokens which may truthfully align to differently inflected forms of the same lemma may instead compete for alignment with the lemma in the translation table, thus creating confusion and resulting in evaluation below the baseline.

[2] http://sourceforge.net/projects/jwordsplitter/

The second variation (in §3.2.2) creates the potential for alignments between agglutinated Finnish words with groups of English words, but also between Finnish lemmas and single English words. While there is more potential for a correct alignments — still despite inflectional information being lost — the approach is still brute force, and there is still confusion created in the translation table since some of the probability given to the correct alignment, whatever that may be, is taken by the alignment of some English words with the agglutinated or non-agglutinated Finnish counterpart.

The third variation (in §3.2.3), while addressing the issue of over-lemmatization created in the first variation, does in fact improve on the baseline. In this final case, inflected forms found in the training data retain their inflection, and so the first person singular form of the verb "to be" in Finnish has greater chance of being translated into "am" rather than the lemmatized form being translated into the most prevalent form of "to be" in the target language training data — "is" for example.

Still the problem of Finnish morphology is very hard for a translation system into English. Our system has only addressed the derivational morphology of Finnish agglutination. We have not at all addressed the inflectional morphology of Finnish, and so much information about the role of certain tokens in the source sentence is lost. Some necessary English words, such as personal pronouns, may be lost on the system because the presence of an English pronoun such as "I" in the best English translation may only be encoded in the inflectional morphology of the Finnish.

In further research, we may try a factored model for our system which encodes not only the lemma or lemmas produced by omorfi, but also the grammatical information from the original inflectional morphology. Further still, our system has not addressed the potential problems of reordering between the source and target languages.

At the very least, a rule could be implemented which places Finnish postpositions in front of their objects as a preprocessing step. As Finnish is a head-final language like English, it is possible that no further rule-based reordering would have to be done, but more research is warranted to make this claim. With these complications yet to be addressed, there is certainly more that we may do in the future to improve evaluation.

References

Stanley F. Chen and Joshua Goodman. 1998. An empirical study of smoothing techniques for language modeling. Technical report, Harvard University.

Michael Collins, Philipp Koehn, and Ivona Kučerová. 2005. Clause restructuring for statistical machine translation. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, pages 531–540, Stroudsburg, PA, USA. Association for Computational Linguistics.

Mathias Creutz and Krista Lagus. 2007. Unsupervised Models for Morpheme Segmentation and Morphology Learning. *ACM Trans. Speech Lang. Process.*, 4(1):3:1–3:34, February.

Christopher Dyer, Smaranda Muresan, and Philip Resnik. 2008. Generalizing word lattice translation. In *Proceedings of ACL-08: HLT*, pages 1012–1020. Association for Computational Linguistics.

Chris Dyer, Hendra Setiawan, Yuval Marton, and Philip Resnik. 2009. The University of Maryland Statistical Machine Translation System for the Fourth Workshop on Machine Translation. In *Proceedings of the Fourth Workshop on Statistical Machine Translation*, StatMT '09, pages 145–149, Stroudsburg, PA, USA. Association for Computational Linguistics.

Kenneth Heafield, Ivan Pouzyrevsky, Jonathan H. Clark, and Philipp Koehn. 2013. Scalable modified kneser-ney language model estimation. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 690–696, Sofia, Bulgaria, August. Association for Computational Linguistics.

Maria Holmqvist, Sara Stymne, and Lars Ahrenberg. 2011. Experiments with word alignment, normalization and clause reordering for smt between english and german. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 393–398, Edinburgh, Scotland, July. Association for Computational Linguistics.

Slava Katz. 1987. Estimation of probabilities from sparse data for the language model component of a speech recognizer. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 35:400–401, March.

Reinhard Kneser and Hermann Ney. 1995. Improved backing-off for *m*-gram language modeling. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP'95)*, pages 181–184, Detroit, Michigan, USA, May. IEEE Computer Society.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic, June. Association for Computational Linguistics.

Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *Proceedings of the 10^{th} Machine Translation Summit (MT Summit X)*, pages 79–86, Phuket, Thailand, September.

Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. 2014. The stanford corenlp natural language processing toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60, Baltimore, Maryland, June. Association for Computational Linguistics.

Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 160–167, Sapporo, Japan, July. Association for Computational Linguistics.

Robert Parker, David Graff, Junbo Kong, Ke Chen, and Kazuaki Maeda. 2011. English Gigaword Fifth Edition LDC2011T07. Philadelphia, Pennsylvania, USA. Linguistic Data Consortium.

Tommi A. Pirinen. 2015. Omorfi — Free and open source morphological lexical database for Finnish. In *Proceedings of the 20th Nordic Conference of Computational Linguistics (NODALIDA '15)*, pages 313–315, Vilnius, Lithuania, May.

Vasin Punyakanok and Dan Roth. 2001. The use of classifiers in sequential inference. In *Advances in Neural Information Processing Systems 14 (NIPS '01)*, pages 995–1001. MIT Press.

Richard Socher, John Bauer, Christopher D. Manning, and Ng Andrew Y. 2013. Parsing with compositional vector grammars. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 455–465, Sofia, Bulgaria, August. Association for Computational Linguistics.

Jana Straková, Milan Straka, and Jan Hajič. 2014. Open-Source Tools for Morphology, Lemmatization, POS Tagging and Named Entity Recognition. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 13–18, Baltimore, Maryland, June. Association for Computational Linguistics.