KLUMSy@KIPoS: Experiments on Part-of-Speech Tagging of Spoken Italian

Thomas Proisl
Computational Corpus Linguistics Group
Friedrich-Alexander-Universität Erlangen-Nürnberg
Bismarckstr. 6
91054 Erlangen, Germany
thomas.proisl@fau.de

Gabriella Lapesa
Institute for Natural Language Processing
Universität Stuttgart
Pfaffenwaldring 5 b
70569 Stuttgart, Germany
gabriella.lapesa@ims.uni-stuttgart.de

Abstract

In this paper, we describe experiments on part-of-speech tagging of spoken Italian that we conducted in the context of the EVALITA 2020 KIPoS shared task (Bosco et al., 2020). Our submission to the shared task is based on SoMeWeTa (Proisl, 2018), a tagger which supports domain adaptation and is designed to flexibly incorporate external resources. We document our approach and discuss our results in the shared task along with a statistical analysis of the factors which impact performance the most. Additionally, we report on a set of additional experiments involving the combination of neural language models with unsupervised HMMs, and compare its performance to that of our system.

1 Introduction

Part-of-speech taggers trained on standard newspaper texts usually perform relatively poorly on spoken language or on written communication that is "conceptually oral", e. g. tweets or chat messages. The challenges of spoken language include non-standard lexis, e. g. the use of colloquial and dialectal forms, and non-standard syntax, e. g. false starts, repetitions, incomplete sentences and the use of fillers. To make things worse, the amount of training data available for spoken language – or non-standard varieties in general – is usually several orders of magnitude smaller than for the usual newspaper corpora. One strategy for coping with this is to incorporate additional resources, e. g. lexica or distributional information obtained from large amounts of unannotated text. Another strategy is to do domain adaptation, i. e. to leverage existing written standard corpora to pretrain an out-of-domain tagger model and to then adapt that model to the target domain using a small amount of in-domain data.

We experiment with these ideas in the context of the EVALITA 2020 shared task on part-of-speech tagging of spoken Italian (Bosco et al., 2020; Basile et al., 2020). The data of the shared task have been drawn from the KIParla corpus (Mauri et al., 2019) and consist of the manually annotated training and test datasets and a silver dataset that has been automatically tagged by the task organizers using a UDPipe[1] model trained on all Italian treebanks in the Universal Dependencies (UD) project.[2] While the silver dataset is annotated with the standard UD tagset (as are the corpora on which the tagger has been trained), the training and test sets use an extended version where tags can optionally be assigned one of two subcategories, .DIA for dialectal forms and .LIN for foreign words.

2 Additional resources

2.1 Corpora

We use a collection of plain text corpora to compute Brown clusters (Brown et al., 1992) that the tagger can use as additional resource.

Ideally, we would use large amounts of transcribed speech for the present task. Since there is no such dataset, we try to use corpora that come close. The closest to authentic speech is scripted speech, therefore we use the Italian movie subtitles from the OpenSubtitles corpus (Lison and Tiedemann, 2016).[3] Computer-mediated communication, e. g. in social media, sometimes exhibits features that are typical of spoken language use. Therefore, we also use a collection of roughly 11.7 million Italian tweets and

[1] http://ufal.mff.cuni.cz/udpipe/1
[2] https://universaldependencies.org/
[3] http://opus.nlpl.eu/OpenSubtitles-v2018.php

ca. 2.7 million Reddit posts (submissions and comments) from the years 2011–2018. We extracted the Reddit posts from Jason Baumgartner's collection of Reddit submissions and comments[4] using the processing pipeline by Blombach et al. (2020). Additionally, we also include all Italian corpora from the Universal Dependencies project and, to further increase the amount of data, a number of web corpora: The PAISÀ corpus of Italian texts from the web (Lyding et al., 2014),[5] the text of the Italian Wikimedia dumps,[6] i. e. Wiki(pedia|books|news|versity|voyage), as extracted by Wikipedia Extractor,[7] and the Italian subset of OSCAR, a huge multilingual Common Crawl corpus (Ortiz Suárez et al., 2019).[8]

We tokenize and sentence split all corpora using UDPipe trained on the union of all Italian UD corpora. We also remove all duplicate sentences. The sizes of the resulting corpora are given in Table 1. As final preprocessing steps, we lowercase all words and normalize numbers, user mentions, email addresses and URLs. Finally, we use the implementation by Liang (2005)[9] to compute 1,000 Brown clusters with a minimum frequency 5.

corpus	complete	deduplicated
oscar	–	13,787,307,218
opensubtitles	795,250,711	378,348,061
paisa	282,631,297	258,679,965
reddit	112,735,958	105,274,620
tweets	152,496,728	148,031,020
ud	672,929	615,057
wiki	578,425,024	560,863,691
wikibooks	12,106,499	11,825,870
wikinews	2,744,317	2,583,135
wikiversity	5,766,859	5,365,924
wikivoyage	3,911,881	3,825,872

Table 1: Sizes of the additional corpora in tokens. OSCAR is already deduplicated on the line level.

2.2 Morphological lexicon

We incorporate linguistic knowledge in the form of Morph-it! (Zanchetta and Baroni, 2005),[10] a morphological lexicon for Italian that contains morphological analyses of roughly 505,000 word forms that correspond to about 35,000 lemmata. In its analyses, Morph-it! distinguishes between derivational features and inflectional features. In total, there are 664 unique feature combinations. We simplify the analyses by stripping away all inflectional features and some of the derivational features, i. e. gender (for articles, nouns and pronouns) and person and number (for pronouns). This results in 39 coarse-grained categories that correspond to major word classes, with some finer distinctions for determiners and pronouns.

3 System description

For our submission to the shared task we use SoMeWeTa (Proisl, 2018), a tagger that is based on the averaged structured perceptron, supports domain adaptation and can incorporate external resources such as Brown clusters and lexica.[11] Its ability to make use of existing linguistic resources allows the tagger to achieve competitive results even with relatively small amounts of in-domain training data, which is particularly useful for non-standard varieties or under-resourced languages (Kabashi and Proisl, 2018; Proisl et al., 2019).

We participate in all three substasks: The main subtask where we use all the available silver and training data, subtask A where we only use the data from the formal register, and subtask B where we only use the informal data. The training scheme is the same for all three subtasks. First, we train preliminary models on the silver data provided by task organizers. Keep in mind that the silver dataset has been automatically tagged. Therefore, it is annotated with the standard version of the UD tagset and not with the extended one that is used in the shared task; in addition, there will be a certain amount of tagging errors in the data. Nevertheless, the dataset provides the tagger with (imperfect) domain-specific background knowledge. In the next step, we adapt the silver models to the union of the Italian UD treebanks, i. e. to high-quality but out-of-domain data. In the final step, we adapt the models to spoken Italian using the manually annotated training data. In every step we train for 12 iterations using a search beam size of 10 and provide the tagger with the Brown clusters and the Morph-it!-based lexicon (Section 2).

[4] https://files.pushshift.io/reddit/
[5] http://www.corpusitaliano.it/
[6] https://dumps.wikimedia.org/
[7] http://medialab.di.unipi.it/wiki/Wikipedia_Extractor
[8] https://oscar-corpus.com/
[9] https://github.com/percyliang/brown-cluster/
[10] https://docs.sslmit.unibo.it/doku.php?id=resources:morph-it
[11] https://github.com/tsproisl/SoMeWeTa

4 Evaluation

4.1 Data preparation and evaluation results

The silver data, training data and the data from the UD treebanks follow UD tokenization guidelines, i. e. contractions such as *parlarmi* (*parlar+mi*) 'to talk+to me' or *della* (*di+la*) 'of+the' are split into their constituents for annotation. This is not the case for the test data where contractions have to be assigned a joint tag, e. g. VERB_PRON or ADP_A. Therefore, we run the test data through the UDPipe tokenizer from Section 2.1, tag the resulting tokens and merge the tags for all tokens that have been split. Table 2 shows the results on the two testsets.[12] On the main task, SoMeWeTa performs reasonably well, only 1–1.4 points worse than the fine-tuned UmBERTo model by Tamburini (2020). On subtasks A and B, it even outperforms that system by a considerable margin.

task	system	formal	informal
main	corrected	92.12	90.11
	gold tokens	92.31	90.66
	Tamburini (2020)	**93.49**	**91.13**
subA	corrected	91.92	89.45
	gold tokens	**92.12**	**89.97**
	Tamburini (2020)	86.47	83.16
subB	corrected	92.37	89.97
	gold tokens	**92.54**	**90.53**
	Tamburini (2020)	89.74	89.52

Table 2: Accuracy scores for our submissions in two variants: (i) With ADP_DET corrected to ADP_A and (ii) based on the true token boundaries instead of on UDPipe tokens.

4.2 Mining tagging accuracy

To get a better insight into the impact of the different experimental variables involved in this study, we carried out feature ablation experiments which targeted the different components of our system, namely the different combinations of training and test data (formal vs. informal) and the different additional resources described in section 2 (use of Brown clusters, Morph-it!, silver data, and UD corpora). We then carried out a linear regression analysis with *tagging accuracy as a dependent variable* and the different *experimental parameters as independent variables (predictors)*. We follow the methodology outlined in Lapesa and Evert (2014) and quantify the impact of a specific predictor (e. g. the use of Brown clusters) as the amount of variance in the dependent variable (tagging accuracy) it accounts for. We considered the following experimental parameters as predictors.

- **setup**: Training/test setup; this predictor encodes the combination of training/test data and has the following values: *all_formal* (i. e. trained on the full set, tested on formal), *all_informal*, *formal_formal*, *formal_informal*, *informal_formal*, *informal_informal*
- **silver**: Use of silver data during training (*yes*, *no*)
- **ud**: Use of UD corpora during training (*yes*, *no*)
- **morph**: Use of Morph-it! (*yes*, *no*)
- **brown**: Use of Brown clusters (*yes*, *no*)

We tested all the possible configurations, i. e. all the combinations of the parameters described above, and, to account for random effects during training, ran each configuration 10 times. This resulted in 960 experimental runs, each corresponding to a single datapoint in our regression analysis. Given that it is reasonable to assume that specific parameter values will influence the performance of other parameters (e. g., use of Morph-it! could boost performance but only if larger corpora are employed), we also test all the 2-way interactions. As a sanity check, we also introduce the number of an experimental run as a predictor (1 to 10, as a categorical variable), in the hope, obviously, of finding no effect for it. Summing up, our regression equation looks as follows:

```
accuracy ~ (setup + silver + ud +
    morph + brown + run)^2
```
[13]

Unsurprisingly, our model achieves an excellent fit to the data, quantified in an Adjusted R-squared of 95.2%. Table 3 lists all significant predictors and interactions, along with their explained variance. Explained variance quantifies the portion of the total R-squared that a specific parameter (or interaction) is responsible for and can be straightforwardly interpreted as the impact that the manipulation of a specific parameter has on the accuracy of our tagger. Reassuringly, we found no effect of experimental run. All other predictors, and

[12] Unfortunately, when preparing our submission, we did not notice that contractions of prepositions (ADP) and determiners (DET) have to be tagged as ADP_A. As a consequence, we mis-tagged all these contractions as ADP_DET. For reference, here are the evaluation results of our faulty submission on the formal/informal test sets: main 87.56/88.24, subA 87.37/87.58, subB 87.81/88.11.

[13] Given that we ran the regression analysis in R, and the equation follows the R syntax in which "^2" denotes all pairwise interactions of the predictors between parentheses.

Predictor	Explained variance
setup	42.06 ***
silver	8.62 ***
ud	12.63 ***
brown	8.76 ***
morph	7.17 ***
setup:silver	1.21 ***
setup:ud	1.08 ***
setup:brown	0.42 ***
setup:morph	0.50 ***
silver:ud	6.00 ***
silver:brown	0.39 ***
silver:morph	1.98 ***
ud:brown	0.03 *
ud:morph	2.48 ***
brown:morph	2.44 ***

Table 3: Regression on tagging a accuracy: predictors and explained variance. Adj. R-squared: 95.2%. Sign. thresholds: ***: 0.001; *: 0.05.

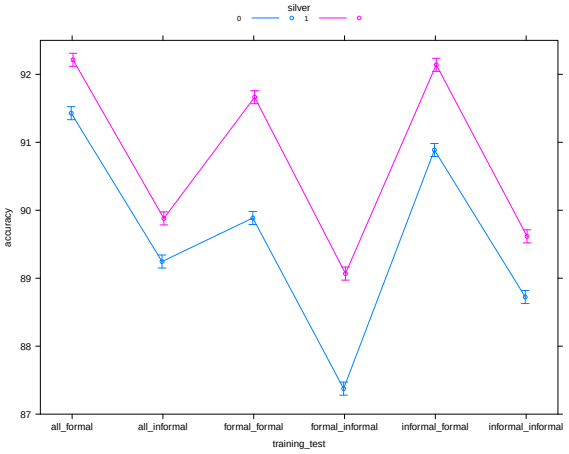

Figure 1: Interaction: setup and silver data

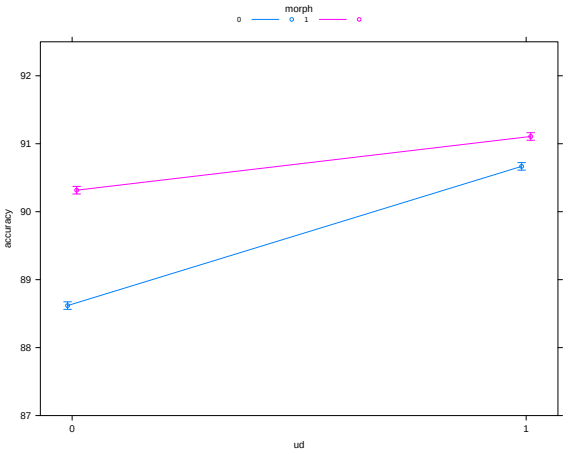

Figure 2: Interaction: UD corpora and Morph-it!

all the corresponding interactions, turned out to be highly significant (with one minor exception). The biggest role is played by the *setup* variable, which alone accounts for 42.06%. Using UD corpora in the training has also a strong impact, with a strong interaction involving the use of silver data (6.00% R-squared). Further strong interactions are found between brown and morph, and brown and UD – probably suggesting that introducing a 3-way interaction would be appropriate here. Given the increased complexity, however, this extension is left for future work.

Now that we have established which parameters or interactions have the strongest impact on model performance, it is time to ask which parameter values ensure the best performance. In our case, given that the system can be assembled incrementally (adding external resources and training data to a basic configuration), asking what the best parameter values are amounts to determining if, for example, the addition of Brown clusters improves performance or is detrimental. Note that the significance of the *brown* predictor in the regression analysis already tells us that the predictor affects performance, ruling out the possibility that it has no impact at all. To visualize the effects in the linear model, we follow Lapesa and Evert (2014) and employ effect displays which show the partial effect of one or two parameters by marginalizing over all other parameters. Unlike coefficient estimates, they allow an intuitive interpretation of the effect sizes of categorical variables irrespective of the dummy coding scheme used.

Let us start with the strongest predictor, *setup*, in its strongest interaction, the one with *silver*. Figure 1 displays the predicted accuracies resulting from the different parameter combinations of the two predictors. Note that, given the excellent fit of the regression model, we can assume predicted accuracy to be a reliable estimate of actual accuracy. Also, note that while we are visualizing the predicted accuracy of a 2-way interaction, we are actually displaying the effect of the individual terms (*setup* and *silver*) and of the interaction (*setup:silver*) jointly. We observe that, unsurprisingly, independently of the use of silver data, training on the whole dataset ensures the best performance on both the formal and informal test sets. The use of silver data (pink line) improves performance, but with differences in the different training/test setups. Interestingly, using the silver data makes the performance gap between the models trained on the whole dataset and those trained on just the informal dataset negligible. Surprisingly, we observe that the best performance is predicted for the formal test set when the informal set is

used. Further experiments on the complementarity of the two subtasks are needed to further clarify this contradiction.

Figure 2 displays the interaction between the use of UD corpora and the integration of Morph-it! in SoMeWeTa. Note that the performance gaps are smaller here than in the previous interaction: this is no surprise, given the smaller explanatory power (explained variance) of the parameters and interactions involved. Morph-it! produces substantial improvements, but again, to a lesser extent if UD corpora are employed: this could either be due to a lower coverage of Morph-it! on the UD corpora, or to the boost in model robustness produced by the introduction of a larger training set. The steep slope of the blue line wrt. the pink one suggests that the presence of a morphological lexicon like Morph-it! can compensate the lack of training data. Let us conclude with the third strongest interaction, the one between the use of Brown clusters and the use of Morph-it!, not shown here for space constraints. It is strikingly similar to the one in Figure 2: Morph-it! improves performance overall, and the steeper improvement in absence of the Brown clusters suggests that the quality of the information encoded in Morph-it! can compensate for the lack of external resources.

In sum, our analysis supports the starting assumption that in a low-resource setting like the one of KIPoS, integrating additional, focussed resources always supports performance.

5 Additional experiments: RoBERTa with unsupervised HMM

Fine-tuned neural language models have been extremely successful in all areas of natural language processing (NLP). Not only can language models trained on huge amounts of plain text be fine-tuned to all NLP tasks, they have also been shown to learn certain linguistic abstractions (Tenney et al., 2019). At least that seems to be the case for English. Languages that are typologically different from English are both more difficult to model with current architectures (Mielke et al., 2019) and seem to be more challenging when it comes to learning linguistic abstractions (Ravfogel et al., 2018). In the experiment described in this section, we extend a state-of-the-art language model architecture to explicitly model part-of-speech information. To this end, we combine a RoBERTa language model (Liu et al., 2019) with an unsupervised neural hidden Markov model (HMM) for part-of-speech induction.

The architecture of the unsupervised HMM follows the LSTM-based variant described by Tran et al. (2016). We directly use the negative logarithm of the observation likelihood determined by the backward algorithm as additional loss for the language model. The embeddings of the best tag sequence (determined using the Viterbi algorithm) are added to the word embeddings before feeding them into the language model. Due to time and resource constraints, we opt for a small to medium-sized model[14] with a total of 45.5 million trainable parameters and train it on 1.9 billion tokens of text (the corpora described in Section 2.1 excluding OSCAR). The model variant with the unsupervised HMM totals 48.7 million trainable parameters. We pre-train and fine-tune both models with the same set of parameters.[15]

The results are summarized in Table 4. Due to the small model size and relatively little training data, the performance of both models is below SoMeWeTa's. (Keep in mind that state-of-the-art language models for Italian like UmBERTo or GilBERTo[16] are based on the same RoBERTa architecture but feature roughly three times as many parameters and have been trained on an order of magnitude more data.) However, the experiment is successful insofar as explicitly modelling part-of-speech information using an unsupervised HMM gives modest gains on both test sets. On the union of the two test sets, this corresponds to a statistically significant improvement from 89.84 to 90.42 (McNemar mid-p test: $p = 0.0133$).

model	formal	informal
RoBERTa	91.28	88.46
RoBERTa+HMM	**91.84**	**89.05**

Table 4: Results for RoBERTa and for RoBERTa with additional unsupervised HMM

[14] We use the RoBERTa implementation from the transformers library (`https://github.com/huggingface/transformers`) with 6 hidden layers, 8 attention heads, a hidden size of 512 and an intermediate size of 2048.

[15] Pretraining for 100,000 steps with a batch size of 500, peak learning rate of 5×10^{-4}, 6,000 warm-up steps and dropout set to 0.1. Fine-tuning to the KIPoS task using the entire training data for 4 epochs with a batch size of 32 and learning rate of 3×10^{-4}

[16] `https://github.com/musixmatchresearch/umberto`, `https://github.com/idb-ita/GilBERTo`

6 Conclusion

This paper started out with the assumption that in low-resource scenarios like the KIPoS shared task the integration of additional resources such as lexica (in our case, Morph-it!) and distributional information from larger corpora (in our case, the Brown clusters) can compensate for the lack of large amounts of training data. Moreover, our strategy also built on the assumption that in a low-resource scenario domain adaptation would be a winning strategy, as it would enable us to exploit larger training sets for written language (out of domain), and then fine-tune the tagger on the spoken language (in domain). The results of our experiments, and the insights gathered from the statistical analysis of our results indicate that both assumptions hold to be true, as far as our contribution to the KIPoS shared task is concerned. In subtasks A and B, where only half the amount of training data was available, this strategy even outperformed a fine-tuned state-of-the-art neural language model. Further work is needed to assess the complementarity of the error profiles of different configurations, taking into the picture also the neural architectures evaluated in Section 4.

References

Valerio Basile, Danilo Croce, Maria Di Maro, and Lucia C. Passaro. 2020. Evalita 2020: Overview of the 7th evaluation campaign of natural language processing and speech tools for italian. In Valerio Basile, Danilo Croce, Maria Di Maro, and Lucia C. Passaro, editors, *Proceedings of Seventh Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2020)*, Online. CEUR.org.

Andreas Blombach, Natalie Dykes, Philipp Heinrich, Besim Kabashi, and Thomas Proisl. 2020. A corpus of German Reddit exchanges (GeRedE). In *Proc. of LREC*, pages 6310–6316, Marseille. ELRA.

Cristina Bosco, Silvia Ballarè, Massimo Cerruti, Eugenio Goria, and Caterina Mauri. 2020. KIPoS@EVALITA2020: Overview of the task on KIParla part of speech tagging. In *Proc. of EVALITA*. CEUR.org.

Peter F. Brown, Vincent J. Della Pietra, Peter V. de Souza, Jennifer C. Lai, and Robert L. Mercer. 1992. Class-based n-gram models of natural language. *Computational Linguistics*, 18(4):467–479.

Besim Kabashi and Thomas Proisl. 2018. Albanian part-of-speech tagging: Gold standard and evaluation. In *Proc. of LREC*, pages 2593–2599, Miyazaki. ELRA.

Gabriella Lapesa and Stefan Evert. 2014. A large scale evaluation of distributional semantic models: Parameters, interactions and model selection. *TACL*, 2:531–546.

Percy Liang. 2005. Semi-supervised learning for natural language. Master's thesis, Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science.

Pierre Lison and Jörg Tiedemann. 2016. Opensubtitles2016: Extracting large parallel corpora from movie and TV subtitles. In *Proc. of LREC*, pages 923–929, Portorož. ELRA.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach.

Verena Lyding, Egon Stemle, Claudia Borghetti, Marco Brunello, Sara Castagnoli, Felice Dell'Orletta, Henrik Dittmann, Alessandro Lenci, and Vito Pirrelli. 2014. The PAISÀ corpus of Italian web texts. In *Proc. of WaC-9*, pages 36–43, Gothenburg. ACL.

Caterina Mauri, Silvia Ballarè, Eugenio Goria, Massimo Cerruti, and Francesco Suriano. 2019. KIParla corpus: A new resource for spoken Italian. In *Proc. of CLiC-it*, Bari.

Sabrina J. Mielke, Ryan Cotterell, Kyle Gorman, Brian Roark, and Jason Eisner. 2019. What kind of language is hard to language-model? In *Proc. of ACL*, pages 4975–4989, Florence. ACL.

Pedro Javier Ortiz Suárez, Benoît Sagot, and Laurent Romary. 2019. Asynchronous pipelines for processing huge corpora on medium to low resource infrastructures. In *Proc. of CMLC-7*, pages 9 – 16, Mannheim. Leibniz-Institut für Deutsche Sprache.

Thomas Proisl, Peter Uhrig, Philipp Heinrich, Andreas Blombach, Sefora Mammerella, Natalie Dykes, and Besim Kabashi. 2019. The_Illiterati: Part-of-speech tagging for Magahi and Bhojpuri without even knowing the alphabet. In *Proc. of NSURL*, Trento.

Thomas Proisl. 2018. SoMeWeTa: A part-of-speech tagger for German social media and web texts. In *Proc. of LREC*, pages 665–670, Miyazaki. ELRA.

Shauli Ravfogel, Yoav Goldberg, and Francis Tyers. 2018. Can LSTM learn to capture agreement? The case of Basque. In *Proc. of BlackboxNLP*, pages 98–107, Brussels, November. ACL.

Fabio Tamburini. 2020. UniBO@KIPoS: Fine-tuning the Italian "BERTology" for the EVALITA 2020 KIPOS task. In *Proc. of EVALITA*. CEUR.org.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019. BERT rediscovers the classical NLP pipeline. In *Proc. of ACL*, pages 4593–4601, Florence. ACL.

Ke M. Tran, Yonatan Bisk, Ashish Vaswani, Daniel Marcu, and Kevin Knight. 2016. Unsupervised neural hidden Markov models. In *Proc. of the Workshop on Structured Prediction for NLP*, pages 63–71, Austin, TX. ACL.

Eros Zanchetta and Marco Baroni. 2005. Morph-it! A free corpus-based morphological resource for the Italian language. In *Proc. of Corpus Linguistics*, Birmingham.