Learning finite state word representations for unsupervised Twitter adaptation of POS taggers*

Julie Wulff
Dpt. of Psychology
University of Southern Denmark
jwulff@health.sdu.dk

Anders Søgaard
Center for Language Technology
University of Copenhagen
soegaard@hum.ku.dk

Abstract

Brown clusters enable POS taggers to generalize better to words that did not occur in the labeled data, clustering distributionally similar seen and unseen words, thereby making models more robust to sparsity effects and domain shifts. However, Brown clustering is a transductive clustering method, and OOV effects still arise. Words neither in the labeled data *nor* in the unlabeled data cannot be assigned to a cluster, and hence, are frequently mis-tagged. This paper presents a simple method of learning finite state automata from Brown clusters that accept and give representations to *truly* unseen words. We show that using automata rather than Brown clusters lead to significant improvements in performance in unsupervised cross-domain POS tagging.

1 Introduction

Out-of-vocabulary (OOV) effects are probably the most common sources of errors in natural language processing. OOV effects arise when supervised models are trained on manually annotated corpora (labeled data) and applied to new text containing words not in the labeled data. The most popular technique to combat OOV effects in the last decade has arguably been Brown clustering (Brown et al., 1992). Other alternatives exist, like word embeddings (Turian et al., 2010), but more than twice as many ACL papers talk about *word clusters* than about word embeddings.

The main problem with Brown clusters – as well as word embeddings – is that they are intended for transductive use, i.e., Brown clusters are used to induce distributional classes (representations) for *observed* words. In other words, while they may minimize OOV effects by bridging between words observed in small labeled corpora and words observed in huge unlabeled corpora, they still do not give us representations for words that we encounter for the first time in our test data. That is, words neither in the labeled nor in the unlabeled data. Such words could, for example, be spelling variants or truly new words (neologisms, etc.).

In newswire most *truly* new words may be proper nouns, but on social media like Twitter we see a lot of linguistic creativity, many spelling variants, and all sorts of neologisms. In our Twitter data, for example, about 40% of the word types were not observed in neither the labeled nor the unlabeled data used to infer our POS tagging model.

This paper presents a relatively simple technique for learning open-ended word representations from Brown clusters, covering also

*The work was done while Julie was a MSc student at University of Copenhagen.

Proceedings of the ACL 2015 Workshop on Noisy User-generated Text, pages 162–166,
Beijing, China, July 31, 2015.

a large portion of the *truly* unknown words. In our experiments, we obtain representations for about 1/4 of these words (1/4 of 40%). The technique, briefly put, is about constructing minimal finite state automata (FSAs) from Brown clusters, collect evidence for productive morpho-phonological alternations (reentrant branchings; see §3), and using these to augment the FSAs. We apply the FSA-based word representations to unsupervised domain adaptation of POS taggers to Twitter data - and show how this leads to significant improvements over a strong baseline system.

2 Related work

FSAs Many of the rules used in phonology and morphology can be analyzed as special cases of regular expressions, and many linguistic descriptions at this level can be compiled into finite state automata (FSAs) (Kaplan and Kay, 1994; Karttunen et al., 1997). Learning minimal FSAs from samples is generally NP-hard (Gold, 1978), and most FSAs used to model phono-/morphotactic constraints have been manually constructed. However, learning a minimal FSA for a fixed set of members of a Brown clusters, is obviously a much easier problem. We extend the FSAs to capture spelling variations better using a simple propagation principle (see §3).

Noeman and Madkour (2010) use FSAs for named entity transliteration, a problem which is very related to ours. They learned transliteration patterns using techniques from phrase-based SMT, but formalized the transliteration grammars by composing FSAs. Similarly, de Vinaspre et al. (2013) use FSAs to learn transliteration of SNOMED CT terms in Basque. Spelling variations and transliteration seem to form a continuum, from non-dialectal spelling variations such as *Facebook/fbook*, over dialectal variations such as *Baltimore/Baltimaw* (observed on Twitter), to cross-language variations such as *München/Munich*.

POS tagging with Brown clusters Brown et al. (1992) introduced the Brown clustering algorithm, which induces a hiearchy of clusters optimizing the likelihood of a hidden Markov model. Each word is assigned to at most one cluster. The algorithm can be used as an unsupervised POS tagger (Blunsom and Cohn, 2011), but Brown clusters have also been used as features in discriminative sequence modeling (Turian et al., 2010).

Ritter et al. (2011) and Owoputi et al. (2013) use Brown clusters induced from a large Twitter corpus to improve a POS tagger trained on a small corpus on hand-annotated tweets (Gimpel et al., 2011). Several recent papers on domain adaptation of POS taggers use discriminative taggers trained with Brown clusters as features as their baseline, e.g., Plank et al. (2014).

3 FSA word representations

Our approach is to learn FSAs from Brown clusters and use statistics over the learned FSAs to propagate non-determinisms, increasing the coverage of our word representations in domains such as Twitter. We explain our word representation algorithm by the following example:

Say we have learned the Brown cluster {cos, coz, coss, cozzz} consisting of short forms of *because*. We can then construct the minimal FSA that accepts only these four strings. However, in this case, we can construct an even smaller FSA if we allow cyclic transitions going out from the first accepting state. This leads to the automaton in Figure 1, which accepts the regular expression $co\{s|z\}^+$. In practice we will introduce cycles whenever we observe character duplication, replacing $a_1 a_2 \ldots a_m$ with a^+. This is supposed to capture productive character du-

plication in Twitter English, e.g.:

(1) Also, crackers and **cheeeese** is the best.

However, spelling variation on Twitter goes beyond character duplication, e.g.:

(2) Jimmy keeps me company in the **baf-room**

We therefore introduce the notion of k-bounded *reentrant branchings* in minimal FSAs. Formally, a k-bounded reentrant branching is a pair of paths p and p' of length at most k such that $\langle s_i, s_j \rangle \in p$, i.e., there is a path of at most k transitions labeled p taking you from s_i to s_j, and $\langle s_i, s_j \rangle \in p'$, and $p \neq p'$. In all our experiments, $k = 3$. From the automaton in Figure 1, we derive the 3-bound reentrant branchings *s-ss*, *s-sss*, *s-z*, *s-zz*, *s-zzz*, *s-zz*, *s-zzz*, *z-ss*, ...

After we have learned FSAs from C Brown clusters, we rank the observed 3-bound reentrant branchings by their frequency. We then take the m most frequent 3-bound reentrant branchings and use them to construct new FSAs. If an FSA F contains a transition labeled s from state s_i to s_j, for example, and $s - z$ is in the top m most frequent nondeterminisms, we create a copy FSA with all the states and transitions of F, as well as with a transition z from s_i to s_j.

From 1000 clusters used in our experiments below, we generate 565,807 3-bound reentrant branchings.

Just like we can construct feature representations over Brown clusters, e.g., a bag-of-words (or bag-of-clusters) representation indicating which Brown clusters have active member words in the current sliding window, we can use FSAs the same way. For example, we can use a sliding window to represent emissions by what unigrams and bigrams occur as neighbors of the target word, as well as what FSAs accept the target or the neighboring words. This way each word can be represented as a binary vector indicating what FSAs accept this word. We use binary features for lexical forms, Brown clusters, as well as the extended set of FSAs.

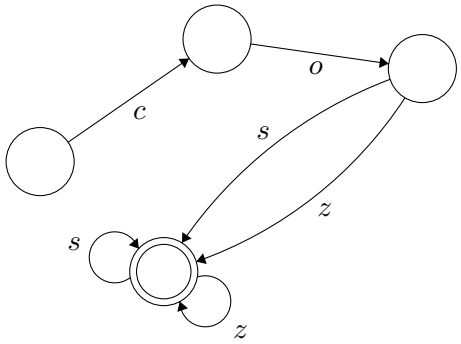

Figure 1: Example FSA for Brown cluster {cos, coz, coss, cozzz}

's	s	s	z
a	e	ie	y
ed	ing	ey	y
ing	n	d	ed
d	s	in	n

Table 1: Top 10 3-bound reentrant branchings in our Twitter clusters

DATA	**baseline**	**FSAs**	**err.red**
FOSTER.DEV	90.0	90.3	0.030
GIMPEL.DEV	74.4	75.0	0.023
FOSTER.TEST	90.0	90.4	0.040
RITTER.TEST	81.8	82.3	0.027
HOVY.TEST	82.2	83.2	0.056

Table 2: Results ($k = 200$, tuned on dev). Effect significant over the entire test data ($p < 0.01$ using Wilcoxon's test)

4 Experiments

We train a linear CRF model on newswire, using a publicly available implementation (CRFsuite),[1] and adapt the feature representation to optimize performance on Twitter data.

Data As our training data we use the OntoNotes 4.0 training split of the Wall Street Journal section of the Penn Treebank. As our held-out data, we use the development sections of Foster et al. (2011) and Gimpel et al. (2011). Our test datasets come from Foster et al. (2011), Ritter et al. (2011) (using the splits from Derczynski et al. (2013)), and Hovy et al. (2014). In other words, one out of three test sets comes from the same sample as one of our development sets, but two come from new ones. This prevents false findings due to over-fitting. All datasets were mapped to the universal tagset presented in Petrov et al. (2011), following Hovy et al. (2014).

Learning CRFsuite uses L-BFGS and L2-regularization by default.

Features Our baseline feature representation uses a combination of unigram, bigram and Brown cluster features, i.e., the CRFsuite default feature model augmented with Brown clusters. The Brown clusters were induced from an in-house Twitter dataset of 57m tweets using Percy Liang's code,[2] after tokenizing the tweets using Twokenize.[3] We use a minimum frequency cut-off at two and induce 1,000 clusters ($C = 1000$). We induce our base FSAs from these clusters using the XFST toolkit.[4] The extended set of FSAs is used to build binary word representations in a sliding window (see above).

The only parameter set is the k-most frequent 3-bound reentrant branchings (set to 200). All other parameters were default in CRFsuite. As already mentioned, we detect 565,807 3-bound reentrant branchings, so by setting $k = 200$ we only use a very small fraction of these. At $k = 200$, the least frequent 3-bound reentrant branchings occur 8 times in our clusters. The most frequent nondeterminism occurs 117 times. The top 10 3-bound reentrant branchings are listed in Table 1. Note that some of these 3-bound reentrant branchings capture inflectional forms, e.g., *ed-ing*, while others capture spelling variations such as *'s-s* and *d-ed*.

[1] http://www.chokkan.org/software/crfsuite/

[2] https://github.com/percyliang/brown-cluster/

[3] http://www.ark.cs.cmu.edu/

[4] http://web.stanford.edu/∼laurik/fsmbook/home.html

5 Results

Our results are presented in Table 2. It is clear that going from Brown clusters to FSAs lead to modest, but consistent improvements across the board. This is not only the case on development data, or test data taken from the same sample as some of our development data (FOSTER.TEST), but across all test sets, including a much newer dataset (HOVY.TEST). The improvements are statistically significant ($p < 0.001$).

Setting $k = 200$ results in 1,699 new words being assigned representations in the annotated Twitter data. Coverage, even with automata representations, was only 69%, showing the need for inductive representations. When we analyze the errors of our FSA-based model, it is clear that most errors are due to known hard cases such as distinguishing between adjectives and adverbs, or distinguishing between adpositions and particles. See Plank et al. (2014) for some discussion.

6 Conclusions

We introduced a new approach to distributional word representations, representing words by the FSAs that accept them. We learn the FSAs from Brown clusters induced from Twitter data, by propagating frequent

3-bound reentrant branchings. The 3-bound reentrant branchings seem to capture morphological rules and known spelling variations well, and lead to significant improvements in POS tagging of Twitter.

References

Phil Blunsom and Trevor Cohn. 2011. A hierarchical Pitman-Yor process HMM for unsupervised part of speech induction. In *ACL*.

P Brown, P DeSouza, R Mercer, D Pietra, and C Lai. 1992. Class based n-gram models of natural language. *Computational Linguistics*, 18:467–479.

Olatz Perez de Vinaspre, Maite Oronoz, Manex Agirrezabal, and Mikel Lersundi. 2013. A finite-state approach to translate SNOMED CT terms into Basque using medical prefixes and suffixes. In *FSMNLP*.

Leon Derczynski, Alan Ritter, Sam Clark, and Kalina Bontcheva. 2013. Twitter part-of-speech tagging for all: overcoming sparse and noisy data. In *RANLP*.

Jennifer Foster, Ozlem Cetinoglu, Joachim Wagner, Josef Le Roux, Joakim Nivre, Deirde Hogan, and Josef van Genabith. 2011. From news to comments: Resources and benchmarks for parsing the language of Web 2.0. In *IJCNLP*.

Kevin Gimpel, Nathan Schneider, Brendan O'Connor, Dipanjan Das, Daniel Mills, Jacob Eisenstein, Michael Heilman, Dani Yogatama, Jeffrey Flanigan, and Noah A. Smith. 2011. Part-of-speech tagging for Twitter. In *ACL*.

Mark Gold. 1978. Complexity of automaton identification from given data. *Information and Control*, 37:302–320.

Dirk Hovy, Barbara Plank, and Anders Søgaard. 2014. When pos datasets don t add up: Combatting sample bias. In *LREC*.

Ron Kaplan and Martin Kay. 1994. Regular models of phonological rule systems. *Computational Linguistics*, 20(3).

Lauri Karttunen, Pierre Chanod, Gregory Grefenstette, and Anne Schiller. 1997. Regular expressions for language engineering. *Natural Language Engineering*, 2(4).

Sara Noeman and Amgad Madkour. 2010. Language independent transliteration mining system using finite state automata framework. In *ACL Workshop on Named Entities*.

Olutobi Owoputi, Brendan O'Connor, Chris Dyer, Kevin Gimpel, Nathan Schneider, and Noah A Smith. 2013. Improved part-of-speech tagging for online conversational text with word clusters. In *NAACL*.

Slav Petrov, Dipanjan Das, and Ryan McDonald. 2011. A universal part-of-speech tagset. CoRR abs/1104.2086.

Barbara Plank, Dirk Hovy, and Anders Søgaard. 2014. Learning POS taggers with inter-annotator agreement loss. In *EACL*.

Alan Ritter, Sam Clark, Mausam Etzioni, and Oren Etzioni. 2011. Named entity recognition in tweets: an experimental study. In *EMNLP*.

Joseph Turian, Lev Ratinov, and Yoshua Bengio. 2010. Word representations: a simple and general method for semi-supervised learning. In *ACL*.