More or less supervised supersense tagging of Twitter

Anders Johannsen, Dirk Hovy, Héctor Martínez Alonso, Barbara Plank, Anders Søgaard
Center for Language Technology
University of Copenhagen, Denmark
Njalsgade 140
ajohannsen@hum.ku.dk, dirk@cst.dk, alonso@hum.ku.dk
plank@cst.dk, soegaard@hum.ku.dk

Abstract

We present two Twitter datasets annotated with coarse-grained word senses (supersenses), as well as a series of experiments with three learning scenarios for supersense tagging: weakly supervised learning, as well as unsupervised and supervised domain adaptation. We show that (a) off-the-shelf tools perform poorly on Twitter, (b) models augmented with embeddings learned from Twitter data perform much better, and (c) errors can be reduced using type-constrained inference with distant supervision from WordNet.

1 Introduction

Supersense tagging (SST, Ciaramita and Altun, 2006) is the task of assigning high-level ontological classes to open-class words (here, nouns and verbs). It is thus a coarse-grained word sense disambiguation task. The labels are based on the lexicographer file names for Princeton WordNet (Fellbaum, 1998). They include 15 senses for verbs and 26 for nouns (see Table 1). While WordNet also provides catch-all supersenses for adjectives and adverbs, these are grammatically, not semantically motivated, and do not provide any higher-level abstraction (recently, however, Tsvetkov et al. (2014) proposed a semantic taxonomy for adjectives). They will not be considered in this paper.

Coarse-grained categories such as supersenses are useful for downstream tasks such as question-answering (QA) and open relation extraction (RE). SST is different from NER in that it has a larger set of labels and in the absence of strong orthographic cues (capitalization, quotation marks, etc.). Moreover, supersenses can be applied to any of the lexical parts of speech and not only proper names. Also, while high-coverage gazetteers can be found for named entity recognition, the lexical resources available for SST are very limited in coverage.

Twitter is a popular micro-blogging service, which, among other things, is used for knowledge sharing among friends and peers. Twitter posts (tweets) announce local events, say talks or concerts, present facts about pop stars or programming languages, or simply express the opinions of the author on some subject matter.

Supersense tagging is relevant for Twitter, because it can aid e.g. QA and open RE. If someone posts a message saying that some LaTeX module now supports "drawing trees", it is important to know whether the post is about drawing natural objects such as oaks or pines, or about drawing tree-shaped data representations.

This paper is, to the best of our knowledge, the first work to address the problem of SST for Twitter. While there exist corpora of newswire and literary texts that are annotated with supersenses, e.g., SEMCOR (Miller et al., 1994), no data is available for microblogs or related domains. This paper introduces two new data sets.

Furthermore, most, if not all, of previous work on SST has relied on gold standard part-of-speech (POS) tags as input. However, in a domain such as Twitter, which has proven to be challenging for POS tagging (Foster et al., 2011; Ritter et al., 2011), results obtained under the assumption of available perfect POS information are almost meaningless for any real-life application.

In this paper, we instead use predicted POS tags and investigate experimental settings in which one or more of the following resources are available to us:

- a large corpus of unlabeled Twitter data;
- Princeton WordNet (Fellbaum, 1998);
- SEMCOR (Miller et al., 1994); and
- a small corpus of Twitter data annotated with supersenses.

We approach SST of Twitter using various degrees of supervision for both learning and domain adaptation (here, from newswire to Twitter). In

*Proceedings of the Third Joint Conference on Lexical and Computational Semantics (*SEM 2014)*, pages 1–11,
Dublin, Ireland, August 23-24 2014.

weakly supervised learning, only *unlabeled* data and the lexical resource WordNet are available to us. While the quality of lexical resources varies, this is the scenario for most languages. We present an approach to weakly supervised SST based on type-constrained EM-trained second-order HMMs (HMM2s) with continuous word representations.

In contrast, when using *supervised* learning, we can distinguish between two degrees of supervision for domain adaptation. For some languages, e.g., Basque, English, Swedish, sense-annotated resources exist, but these corpora are all limited to newswire or similar domains. In such languages, **unsupervised domain adaptation** (DA) techniques can be used to exploit these resources. The setting does not presume labeled data from the target domain. We use discriminative models for unsupervised domain adaptation, training on SEMCOR and testing on Twitter.

Finally, we annotated data sets for Twitter, making **supervised domain adaptation** (SU) experiments possible. For supervised domain adaptation, we use the annotated training data sets from both the newswire and the Twitter domain, as well as WordNet.

For both unsupervised domain adaptation and supervised domain adaptation, we use structured perceptron (Collins, 2002), i.e., a discriminative HMM model, and search-based structured prediction (SEARN) (Daume et al., 2009). We augment both the EM-trained HMM2, discriminative HMMs and SEARN with type constraints and continuous word representations. We also experimented with conditional random fields (Lafferty et al., 2001), but obtained worse or similar results than with the other models.

Contributions In this paper, we present two Twitter data sets with manually annotated supersenses, as well as a series of experiments with these data sets. These experiments cover existing approaches to related tasks, as well as some new methods. In particular, we present type-constrained extensions of discriminative HMMs and SEARN sequence models with continuous word representations that perform well. We show that when no in-domain labeled data is available, type constraints improve model performance considerably. Our best models achieve a weighted average F1 score of 57.1 over nouns and verbs on our main evaluation data set, i.e., a 20% error reduction over the most frequent sense baseline. The two annotated Twitter data sets are publicly released for download at `https://github.com/coastalcph/supersense-data-twitter`.

n.Tops	n.object	v.cognition
n.act	n.person	v.communication
n.animal	n.phenomenon	v.competition
n.artifact	n.plant	v.consumption
n.attribute	n.possession	v.contact
n.body	n.process	v.creation
n.cognition	n.quantity	v.emotion
n.communication	n.relation	v.motion
n.event	n.shape	v.perception
n.feeling	n.state	v.possession
n.food	n.substance	v.social
n.group	n.time	v.stative
n.location	v.body	v.weather
n.motive	v.change	

Table 1: The 41 noun and verb supersenses in WordNet

2 More or less supervised models

This sections covers the varying degree of supervision of our systems as well as the usage of type constraints as distant supervision.

2.1 Distant supervision

Distant supervision in these experiments was implemented by only allowing a system to predict a certain supersense for a given word if that supersense had either been observed in the training data, or, for unobserved words, if the sense was the most frequent sense in WordNet. If the word did not appear in the training data nor in WordNet, no filtering was applied. We refer to the distant-supervision strategy as *type constraints*.

Distant supervision was implemented differently in SEARN and the HMM model. SEARN decomposes sequential labelling into a series of binary classifications. To constrain the labels we simply pick the top-scoring sense for each token from the allowed set. Structured perceptron uses Viterbi decoding. Here we set the emission probabilities for disallowed senses to negative infinity and decode as usual.

2.2 Weakly supervised HMMs

The HMM2 model is a second-order hidden Markov model (Mari et al., 1997; Thede and Harper, 1999) using logistic regression to estimate emission probabilities. In addition we constrain

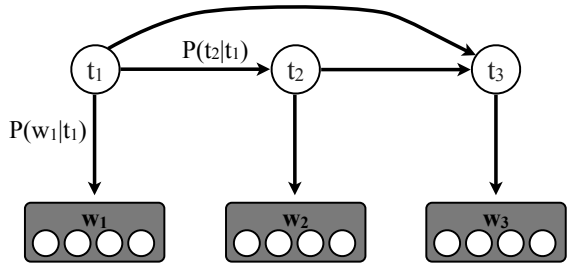

Figure 1: HMM2 with continuous word representations

the inference space of the HMM2 tagger using type-level tag constraints derived from WordNet, leading to roughly the model proposed by Li et al. (2012), who used Wiktionary as a (part-of-speech) tag dictionary. The basic feature model of Li et al. (2012) is augmented with continuous word representation features as shown in Figure 1, and our logistic regression model thus works over a combination of discrete and continuous variables when estimating emission probabilities. We do 50 passes over the data as in Li et al. (2012).

We introduce two simplifications for the HMM2 model. First, we only use the most frequent senses ($k = 1$) in WordNet as type constraints. The most frequent senses seem to better direct the EM search for a local optimum, and we see dramatic drops in performance on held-out data when we include more senses for the words covered by WordNet. Second, motivated by computational concerns, we only train and test on sequences of (predicted) nouns and verbs, leaving out all other word classes. Our supervised models performed slightly worse on shortened sequences, and it is an open question whether the HMM2 models would perform better if we could train them on full sentences.

2.3 Structured perceptron and SEARN

We use two approaches to supervised sequential labeling, structured perceptron (Collins, 2002) and search-based structured prediction (SEARN) (Daume et al., 2009). The structured perceptron is a in-house reimplementation of Ciaramita and Altun (2006).[1] SEARN performed slightly better than structured perceptron, so we use it as our in-house baseline in the experiments below. In this section, we briefly explain the two approaches.

[1] https://github.com/coastalcph/rungsted

2.3.1 Structured perceptron (HMM)

Structured perceptron learning was introduced in Collins (2002) and is an extension of the online perceptron learning algorithm (Rosenblatt, 1958) with averaging (Freund and Schapire, 1999) to structured learning problems such as sequence labeling.

In structured perceptron for sequential labeling, where we learn a function from sequences of data points $x_1 \dots x_n$ to sequences of labels $y_1 \dots y_n$, we begin with a random weight vector $\mathbf{w}_0$ initialized to all zeros. This weight vector is used to assign weights to transitions between labels, i.e., the discriminative counterpart of $P(y_{i+1} \mid y_i)$, and emissions of tokens given labels, i.e., the counterpart of $P(x_i \mid y_i)$. We use Viterbi decoding to derive a best path $\hat{\mathbf{y}}$ through the corresponding $m \times n$ lattice (with m the number of labels). Let the feature mapping $\Phi(\mathbf{x}, \mathbf{y})$ be a function from a pair of sequences $\langle \mathbf{x}, \mathbf{y} \rangle$ to all the features that fired to make $\mathbf{y}$ the best path through the lattice for $\mathbf{x}$. Now the structured update for a sequence of data points is simply $\alpha(\Phi(\mathbf{x}, \mathbf{y}) - \Phi(\mathbf{x}, \hat{\mathbf{y}}))$, i.e., a fixed positive update of features that fired to produce the correct sequence of labels, and a fixed negative update of features that fired to produce the best path under the model. Note that if $\mathbf{y} = \hat{\mathbf{y}}$, no features are updated.

2.3.2 SEARN

SEARN is a way of decomposing structured prediction problems into search and history-based classification. In sequential labeling, we decompose the sequence of m tokens into m classification problems, conditioning our labeling of the ith token on the history of $i - 1$ previous decisions. The cost of a mislabeling at training time is defined by a cost function over output structures. We use Hamming loss rather than F_1 as our cost function, and we then use stochastic gradient descent with quantile loss as a our cost-sensitive learning algorithm. We use a publicly available implementation.[2]

3 Experiments

We experiment with weakly supervised learning, unsupervised domain adaptation, as well as supervised domain adaptation, i.e., where our models are induced from hand-annotated newswire *and* Twitter data. Note that in all our experiments,

[2] http://hunch.net/~vw/

we use *predicted POS tags* as input to the system, in order to produce a realistic estimate of SST performance.

3.1 Data

Our experiments rely on combinations of available resources and newly annotated Twitter data sets made publicly available with this paper.

3.1.1 Available resources

Princeton WordNet (Fellbaum, 1998) is the main resource for SST. The lexicographer file names provide the label alphabet of the task, and the taxonomy defined therein is used not only in the baselines, but also as a feature in the discriminative models. We use the WordNet 3.0 distribution.

SEMCOR (Miller et al., 1994) is a sense-annotated corpus composed of 80% newswire and 20% literary text, using the sense inventory from WordNet. SEMCOR comprises $23k$ distinct lemmas in $234k$ instances. We use the texts which have full annotations, leaving aside the verb-only texts (see Section 6).

We use a distributional semantic model in order to incorporate distributional information as features in our system. In particular, we use the neural-network based models from (Mikolov et al., 2013), also referred as *word embeddings*. This model makes use of skip-grams (n-grams that do not need to be consecutive) within a word window to calculate continuous-valued vector representations from a recurrent neural network. These distributional models have been able to outperform state of the art in the SemEval-2012 Task 2 (Measuring degrees of relational similarity). We calculate the embeddings from an in-house corpus of $57m$ English tweets using a window size 5 and yielding vectors of 100 dimensions.

We also use the first $20k$ tweets of the $57m$ tweets to train our HMM2 models.

3.1.2 Annotation

While an annotated newswire corpus and a high-quality lexical resource already enable us to train, we also need at least a small sample of annotated tweets data to evaluate SST for Twitter. Furthermore, if we want to experiment with supervised SST, we also need sufficient annotated Twitter data to learn the distribution of sense tags.

This paper presents two data sets: (a) supersense annotations for the POS+NER-annotated data set described in Ritter et al. (2011), which we use for training, development and evaluation, using the splits proposed in Derczynski et al. (2013), and (b) supersense annotations for a sample of 200 tweets, which we use for additional, out-of-sample evaluation. We call these data sets RITTER-{TRAIN,DEV,EVAL} and IN-HOUSE-EVAL, respectively. The IN-HOUSE-EVAL dataset was downloaded in 2013 and is a sample of tweets that contain links to external homepages but are otherwise unbiased. It was previously used (with part-of-speech annotation) in (Plank et al., 2014). Both data sets are made publicly available with this paper.

Supersenses are annotated with in spans defined by the BIO (Begin-Inside-Other) notation. To obtain the Twitter data sets, we carried out an annotation task. We first pre-annotated all data sets with WordNet's most frequent senses. If the word was not in WordNet and a noun, we assigned it the sense *n.person*. All other words were labeled *O*.

Chains of nouns were altered to give every element the sense of the head noun, and the BI tags adjusted, i.e.:

> Empire/B-n.loc State/B-n.loc Building/B-n.artifact

was changed to

> Empire/B-n.artifact State/I-n.artifact Building/I-n.artifact

For the RITTER data, three paid student annotators worked on different subsets of the pre-annotated data. They were asked to correct mistakes in both the BIO notation and the assigned supersenses. They were free to chose from the full label set, regardless of the pre-annotation. While the three annotators worked on separate parts, they overlapped on a small part of RITTER-TRAIN (841 tokens). On this subset, we computed agreement scores and annotation difficulties. The average raw agreement was 0.86 and Cohen's κ 0.77. The majority of tokens received the *O* label by all annotators; this happended in 515 out of 841 cases. Excluding these instances to evaluate the performance on the more difficult content words, raw agreement dropped to 0.69 and Cohen's κ to 0.69.

The IN-HOUSE-EVAL data set was annotated by two different annotators, namely two of the authors of this article. Again, for efficiency reasons they worked on different subsets of the data, with an overlapping portion. Their average raw agreement was 0.65 and their Cohen's κ 0.62. For this data set, we also compute F_1, defined as usual as the harmonic mean of recall and precision. To

compute this, we set one of the annotators as gold data and the other as predicted data. However, since F_1 is symmetrical, the order does not matter. The annotation F_1 gives us another estimate of annotation difficulty. We present the figures in Table 3.

3.2 Baselines

For most word sense disambiguation studies, predicting the most frequent sense (MFS) of a word has been proven to be a strong baseline. Following this, our MFS baseline simply predicts the supersense of the most frequent WordNet sense for a tuple of a word and a part of speech. We use the part of speech predicted by the LAPOS tagger (Tsuruoka et al., 2011). Any word not in WordNet is labeled as *noun.person*, which is the most frequent sense overall in the training data. After tagging, we run a script to correct the BI tag prefixes, as described above for the annotation ask.

We also compare to the performance of existing SST systems. In particular we use SenseLearner (Mihalcea and Csomai, 2005) as a baseline, which produces estimates of the WordNet sense for each word. For these predictions, we retrieve the corresponding supersense. Finally, we use a publicly available reimplementation of Ciaramita and Altun (2006) by Michael Heilman, which reaches comparable performance on gold-tagged SEMCOR.[3]

3.3 Model parameters

We use the feature model of Paaß and Reichartz (2009) in all our models, except the weakly supervised models. For the structured perceptron we set the number of passes over the training data on the held-out development data. The weakly supervised models use the default setting proposed in Li et al. (2012). We have used the standard online setup for SEARN, which only takes one pass over the data.

The type of embedding is the same in all our experiments. For a given word the embedding feature is a 100 dimensional vector, which combines the embedding of the word with the embedding of adjacent words. The feature combination $f_{\mathbf{e}}$ for a word w_t is calculated as:

$$\mathrm{f}_{\mathbf{e}}(w_t) = \frac{1}{2}(\mathbf{e}(w_{t-1}) + \mathbf{e}(w_{t+1})) - 2\mathbf{e}(w_t),$$

where the factor of two is chosen heurestically to give more weight to the current word.

We also set a parameter k on development data for using the k-most frequent senses inWordNet as type constraints. Our supervised models are trained on SEMCOR+RITTER-TRAIN or simply RITTER-TRAIN, depending on what gave us the best performance on the held-out data.

4 Results

The results are presented in Table 2. We distinguish between three settings with various degrees of supervision: weakly supervised, which uses no domain annotated information, but solely relies on embeddings trained on unlabeled Twitter data; unsupervised domain adaptation (DA), which uses SemCor for supervised training; and supervised domain adaptation (SU), which uses annotated Twitter data in addition to the SemCor data for training.

In each of the two domain adaptation settings, SEARN and HMM are evaluated with type constraints as distant supervision, and without for comparison. SEARN without embeddings or distant supervision serves as an in-house baseline.

In Table 3 we present the WordNet token coverage of predicted nouns and verbs in the development and evaluation data, as well as the inter-annotator agreement F_1 scores.

All the results presented in Table 2 are (weighted averaged) F_1 measures obtained on predicted POS tags. Note that these results are considerably lower than results on supersense tagging newswire (up to 80 F_1) that assume gold standard POS tags (Ciaramita and Altun, 2006; Paaß and Reichartz, 2009).

The re-implementation of the state-of-the-art system improves slightly upon the most frequent sense baseline. SenseLearner does not seem to capture the relevant information and does not reach baseline performance. In other words, there is no off-the-shelf tool for supersense tagging of Twitter that does much better than assigning the most frequent sense to predicted nouns and verbs.

Our weakly supervised model performs worse than the most frequent sense baseline. This is a negative result. It is, however, well-known from the word sense disambiguation literature that the MFS is a very strong baseline. Moreover, the EM learning problem is hard because of the large label set and weak distributional evidence for super-

[3] http://www.ark.cs.cmu.edu/mheilman/questions/SupersenseTagger-10-01-12.tar.gz

	RITTER DEV	RITTER EVAL	IN-HOUSE EVAL
Wordnet noun-verb token coverage	83.72	70.22	41.18
Inter-annotator agreement (F1)	81.01	69.15	61.57

Table 3: Properties of dataset.

senses.

The unsupervised domain adaptation and fully supervised systems perform considerably better than this baseline across the board. In the unsupervised domain adaptation setup, we see huge improvements from using type constraints as distant supervision. In the supervised setup, we only see significant improvements adding type constraints for the structured perceptron (HMM), but not for search-based structured prediction (SEARN).

For all the data sets, there is still a gap between model performance and human inter-annotator agreement levels (see Table 3), leaving some room for improvements. We hope that the release of the data sets will help further research into this.

4.1 Coarse-grained evaluation

We also experimented with the more coarse-grained classes proposed by Yuret and Yatbaz (2010). Here our best model obtained an F_1 score for mental concepts (nouns) of 72.3%, and 62.6% for physical concepts, on RITTER-DEV. The overall F_1 score for verbs is 85.6%. The overall F_1 is 75.5%. Note that this result is not directly comparable to the figure (72.9%) reported in Yuret and Yatbaz (2010), since they use different data sets, exclude verbs and make different assumptions, e.g., relying on gold POS tags.

5 Error analysis

We have seen that inter-annotator agreements on supersense annotation are reliable at above .60 but far from perfect. The Hinton diagram in Table 2 presents the confusion matrix between our annotators on IN-HOUSE-EVAL.

Errors in the prediction primarily stem from two sources: out-of-vocabulary words and incorrect POS tags. Figure 3 shows the distribution of senses over the words that were not contained in either the training data, WordNet, or the Twitter data used to learn the embeddings. The distribution follows a power law, with the most frequent sense being *noun.person*, followed by *noun.group*, and *noun.artifact*. The first two are related to NER categories, namely *PER* and *ORG*, and can be expected, since Twitter users frequently talk about new actors, musicians, and bands. Nouns of communication are largely related to films, but also include Twitter, Facebook, and other forms of social media. Note that verbs occur only towards the tail end of the distribution, i.e., there are very few unknown verbs, even in Twitter.

Overall, our models perform best on labels with low lexical variability, such as quantities, states and times for nouns, as well as consumption, possession and stative for verbs. This is unsurprising, since these classes have lower out-of-vocabulary rates.

With regards to the differences between source (SEMCOR) and target (Twitter) domains, we observe that the distribution of supersenses is always headed by the same noun categories like *noun.person* or *noun.group*, but the frequency of out-of-vocabulary stative verbs plummets in the target domain, as some semantic types are more closed class than others. There are for instance fewer possibilities for creating new time units (noun.time) or stative verbs like *be* than people or company names (*noun.person* or *noun.group*, respectively).

The weakly supervised model HMM2 has higher precision (57% on RITTER-DEV) than recall (48.7%), which means that it often predicts words to not belong to a semantic class. This suggests an alternative strategy, which is to train a model on sequences of purely non-*O* instances. This would force the model to only predict *O* on words that do not appear in the reduced sequences.

One important source of error seems to be unreliable part-of-speech tagging. In particular we predict the wrong POS for 20-35% of the verbs across the data sets, and for 4-6.5% of the nouns. In the SEMCOR data, for comparability, we have wrongly predicted tags for 6-8% of the annotated tokens. Nevertheless, the error propagation of wrongly predicted nouns and verbs is partially compensated by our systems, since they are trained on imperfect input, and thus it becomes possible for the systems to predict a noun supersense for a verb and viceversa. In our data we have found e.g. that the noun *Thanksgiving* was incorrectly tagged as a verb, but its supersense was correctly predicted to be *noun.time*, and that the verb *guess* had been mistagged as noun but the system

	Resources				Results		
	Token-level		Type-level		RITTER		IN-HOUSE
	SemCor	Twitter	Embeddings	Type constraints	DEV	EVAL	EVAL
General baselines							
MFS	-	-	-	+	47.54	44.98	38.65
SENSELEARNER	+	-	-	-	14.61	26.24	22.81
HEILMAN	+	-	-	-	*48.96*	*45.03*	*39.65*
Weakly supervised systems							
HMM2	-	-	-	+	47.09	42.12	26.99
Unsupervised domain adaptation systems (DA)							
SEARN (Baseline)	+	-	-	-	48.31	42.34	34.30
SEARN	+	-	+	-	52.45	48.30	40.22
SEARN	+	-	+	+	56.59	**50.89**	40.50
HMM	+	-	+	-	52.40	47.90	40.51
HMM	+	-	+	+	**57.14**	**50.98**	**41.84**
Supervised domain adaptation systems (SU)							
SEARN (Baseline)	+	+	-	-	58.30	*52.12*	36.86
SEARN	+	+	+	-	**63.05**	57.09	42.37
SEARN	+	+	+	+	62.72	**57.14**	**42.42**
HMM	+	+	+	-	57.20	49.26	39.88
HMM	+	+	+	+	60.66	51.40	41.60

Table 2: Weighted F1 average over 41 supersenses.

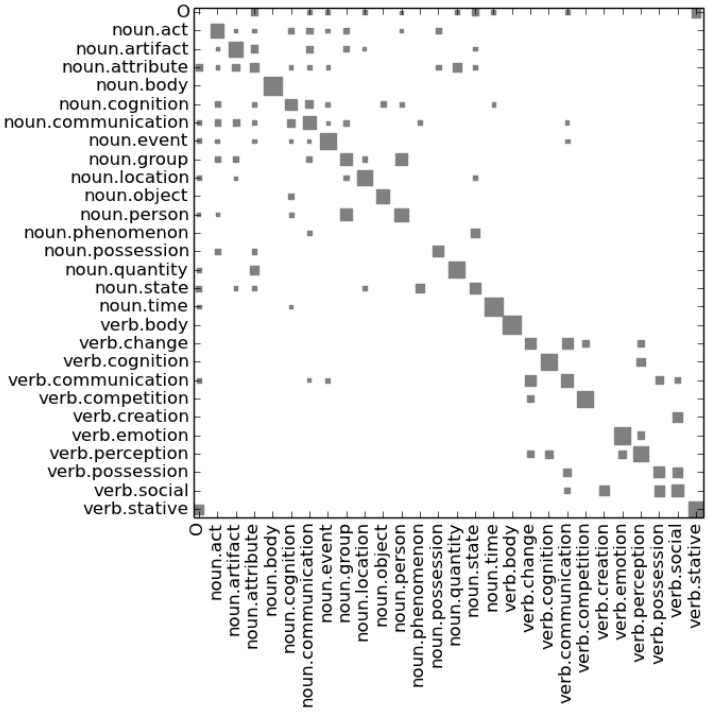

Figure 2: Inter-annotator confusion matrix on TWITTER-EVAL.

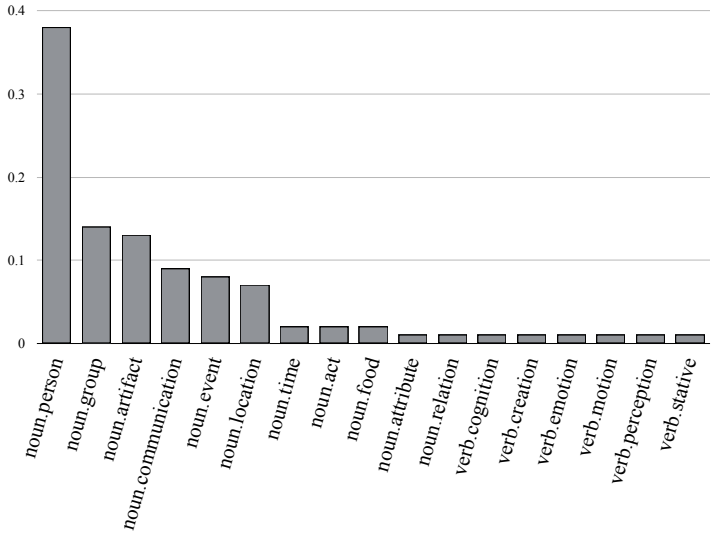

Figure 3: Sense distribution of OOV words.

still predicted the correct *verb.cognition* as supersense.

6 Related Work

There has been relatively little previous work on supersense tagging, and to the best of our knowledge, all of it has been limited to English newswire and literature (SEMCOR and SENSEVAL).

The task of supersense tagging was first introduced by Ciaramita and Altun (2006), who used a structured perceptron trained and evaluated on SEMCOR via 5-fold cross validation. Their evaluation included a held-out development set on each fold that was used to estimate the number of epochs. They used additional training data containing only verbs. More importantly, they relied on gold standard POS tags. Their overall F_1 score on SEMCOR was 77.1. Reichartz and Paaß (Reichartz and Paaß, 2008; Paaß and Reichartz, 2009) extended this work, using a CRF model as well as LDA topic features. They report an F_1 score of 80.2, again relying on gold standard POS features. Our implementation follows their setup and feature model, but we rely on *predicted* POS features, not gold standard features.

Supersenses provide information similar to higher-level distributional clusters, but more interpretable, and have thus been used as high-level features in various tasks, such as preposition sense disambiguation, noun compound interpretation, and metaphor detection (Ye and Baldwin, 2007; Tratz and Hovy, 2010; Tsvetkov et al., 2013). Princeton WordNet only provides a fully developed taxonomy of supersenses for verbs and nouns, but Tsvetkov et al. (2014) have recently proposed an extension of the taxonomy to cover adjectives. Outside of English, supersenses have been annotated for Arabic Wikipedia articles by Schneider et al. (2012).

In addition, a few researchers have tried to solve coarse-grained word sense disambiguation problems that are very similar to supersense tagging. Kohomban and Lee (2005) and Kohomban and Lee (2007) also propose to use lexicographer file identifers from Princeton WordNet senses (supersenses) and, in addition, discuss how to retrieve fine-grained senses from those predictions. They evaluate their model on all-words data from SENSEEVAL-2 and SENSEEVAL-3. They use a classification approach rather than structured prediction.

Yuret and Yatbaz (2010) present a weakly unsupervised approach to this problem, still evaluating on SENSEVAL-2 and SENSEVAL-3. They focus only on nouns, relying on gold part-of-speech, but also experiment with a coarse-grained mapping, using only three high level classes.

For Twitter, we are aware of little previous work on word sense disambiguation. Gella et al. (2014) present lexical sample word sense disambiguation annotation of 20 target nouns on Twitter, but no experimental results with this data. There has also been related work on disambiguation to Wikipedia for Twitter (Cassidy et al., 2012).

In sum, existing work on supersense tagging and coarse-grained word sense disambiguation for English has to the best of our knowledge all focused on newswire and literature. Moreover, they all rely on gold standard POS information, making previous performance estimates rather optimistic.

7 Conclusion

In this paper, we present two Twitter data sets with manually annotated supersenses, as well as a series of experiments with these data sets. The data is publicly available for download.

In this article we have provided, to the best of our knowledge, the first supersense tagger for Twitter. We have shown that off-the-shelf tools perform poorly on Twitter, and we offer two strategies—namely distant supervision and the usage of embeddings as features—that can be combined to improve SST for Twitter.

We propose that distant supervision implemented as type constraints during decoding is a viable method to limit the mispredictions of supersenses by our systems, thereby enforcing predicted senses that a word has in WordNet. This approach compensates for the size limitations of the training data and mitigates the out-of-vocabulary effect, but is still subject to the coverage of WordNet; which is far from perfect for words coming from high-variability sources such as Twitter.

Using distributional semantics as features in form of word embeddings also improves the prediction of supersenses, because it provides semantic information for words, regardless of whether they have been observed the training data. This method does not require a hand-created knowledge base like WordNet, and is a promising technique for domain adaptation of supersense tagging.

References

Taylor Cassidy, Heng Ji, Lev-Arie Ratinov, Arkaitz Zubiaga, and Hongzhao Huang. 2012. Analysis and enhancement of wikification for microblogs with context expansion. In *COLING*, volume 12, pages 441–456.

Massimiliano Ciaramita and Yasemin Altun. 2006. Broad-coverage sense disambiguation and information extraction with a supersense sequence tagger. In *Proc. of EMNLP*, pages 594–602, Sydney, Australia, July.

Michael Collins. 2002. Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms. In *EMNLP*.

Hal Daume, John Langford, and Daniel Marcu. 2009. Search-based structured prediction. *Machine Learning*, pages 297–325.

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

Christiane Fellbaum. 1998. *WordNet: an electronic lexical database*. MIT Press USA.

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*.

Yoav Freund and Robert Schapire. 1999. Large margin classification using the perceptron algorithm. *Machine Learning*, 37:277–296.

Spandana Gella, Paul Cook, and Timothy Baldwin. 2014. One sense per tweeter and other lexical semantic tales of Twitter. In *EACL*.

Upali Kohomban and Wee Lee. 2005. Learning semantic classes for word sense disambiguation. In *ACL*.

Upali Kohomban and Wee Lee. 2007. Optimizing classifier performance in word sense disambiguation by redefining word sense classes. In *IJCAI*.

John Lafferty, Andrew McCallum, and Fernando Pereira. 2001. Conditional random fields: probabilistic models for segmenting and labeling sequence data. In *ICML*.

Shen Li, João Graça, and Ben Taskar. 2012. Wiki-ly supervised part-of-speech tagging. In *EMNLP*.

Jean-Francois Mari, Jean-Paul Haton, and Abdelaziz Kriouile. 1997. Automatic word recognition based on second-order hidden Markov models. *IEEE Transactions on Speech and Audio Processing*, 5(1):22–25.

Rada Mihalcea and Andras Csomai. 2005. Senselearner: Word sense disambiguation for all words in unrestricted text. In *Proceedings of the ACL 2005 on Interactive poster and demonstration sessions*, pages 53–56. Association for Computational Linguistics.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In *NIPS*.

George A. Miller, Martin Chodorow, Shari Landes, Claudia Leacock, and Robert G. Thomas. 1994. Using a semantic concordance for sense identification. In *Proceedings of the workshop on Human Language Technology*, pages 240–243. Association for Computational Linguistics.

Gerhard Paaß and Frank Reichartz. 2009. Exploiting semantic constraints for estimating supersenses with CRFs. In *Proc. of the Ninth SIAM International Conference on Data Mining*, pages 485–496, Sparks, Nevada, May.

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

Frank Reichartz and Gerhard Paaß. 2008. Estimating Supersenses with Conditional Random Fields. In *Proceedings of ECMLPKDD*.

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

Frank Rosenblatt. 1958. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386–408.

Nathan Schneider, Behrang Mohit, Kemal Oflazer, and Noah A Smith. 2012. Coarse lexical semantic annotation with supersenses: an arabic case study. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*, pages 253–258. Association for Computational Linguistics.

Scott Thede and Mary Harper. 1999. A second-order hidden Markov model for part-of-speech tagging. In *ACL*.

Stephen Tratz and Eduard Hovy. 2010. Isi: automatic classification of relations between nominals using a maximum entropy classifier. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 222–225. Association for Computational Linguistics.

Yoshimasa Tsuruoka, Yusuke Miyao, and Jun'ichi Kazama. 2011. Learning with lookahead: can history-based models rival globally optimized models? In *CoNLL*.

Yulia Tsvetkov, Elena Mukomel, and Anatole Gershman. 2013. Cross-lingual metaphor detection using common semantic features. *Meta4NLP 2013*, page 45.

Yulia Tsvetkov, Nathan Schneider, Dirk Hovy, Archna Bhatia, Manaal Faruqui, and Chris Dyer. 2014. Augmenting english adjective senses with supersenses. In *Proc. of LREC*.

Patrick Ye and Timothy Baldwin. 2007. Melb-yb: Preposition sense disambiguation using rich semantic features. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 241–244. Association for Computational Linguistics.

Deniz Yuret and Mehmet Yatbaz. 2010. The noisy channel model for unsupervised word sense disambiguation. *Computational Linguistics*, 36:111–127.