Similarity Based Genre Identification for POS Tagging & Dependency Parsing Experts

Atreyee Mukherjee
Indiana University
atremukh@indiana.edu

Sandra Kübler
Indiana University
skuebler@indiana.edu

Abstract

POS tagging and dependency parsing achieve good results for homogeneous datasets. However, these tasks are much more difficult on heterogeneous datasets. In (Mukherjee et al., 2016, 2017), we address this issue by creating genre experts for both POS tagging and parsing. We use topic modeling to automatically separate training and test data into genres and to create annotation experts per genre by training separate models for each topic. However, this approach assumes that topic modeling is performed jointly on training and test sentences each time a new test sentence is encountered. We extend this work by assigning new test sentences to their genre expert by using similarity metrics. We investigate three different types of methods: 1) based on words highly associated with a genre by the topic modeler, 2) using a k-nearest neighbor classification approach, and 3) using perplexity to determine the closest topic. The results show that the choice of similarity metric has an effect on results and that we can reach comparable accuracies to the joint topic modeling in POS tagging and dependency parsing, thus providing a viable and efficient approach to POS tagging and parsing a sentence by its genre expert.

1 Introduction

POS tagging and dependency parsing can be performed reliably on homogeneous datasets such as the Penn Treebank. However, both POS taggers and parsers are often used on out-of-domain data, which generally results in considerably lower accuracies. Domain adaptation provides a potential for improving out-of-domain accuracy if annotations in the target domain are available. In our work, we assume a more flexible approach in which POS tagging and parsing of an individual sentence is performed by the closest genre expert where the genres are identified automatically prior to training.

In (Mukherjee et al., 2016, 2017), we approach this problem by creating genre/domain experts using topic modeling: We use Latent Dirichlet Allocation (LDA) (Blei et al., 2003; Blei, 2012) for an unsupervised clustering of sentences into topics. We assume that these topics correspond to genres; previous experiments (Mukherjee et al., 2016) have shown that the topic modeler models the split into genres in a very similar way to the original split, with error rates around 2%. We then train one expert per topic. I.e., we train the expert on all the training sentences that were assigned to the corresponding topic. During testing, we assign test sentences to topics, which means that they are POS tagged and parsed by the corresponding expert. We tested the approach on an artificial, heterogeneous corpus, consisting of a balanced mix of sentences from the WSJ portion of the Penn Treebank (financial news) (Marcus et al., 1994) and from the GENIA corpus (biomedical abstracts) (Tateisi and Tsujii, 2004). For POS tagging, we show a moderate increase in performance over a competitive baseline of training on the full training set, and a considerable increase for dependency parsing.

However, in our previous approach, assigning test sentences to the relevant training topic experts is handled in the simplest possible way: We perform topic modeling on the combination of training and test data. Since topic modeling clusters the data but does not create a predictive model, we cannot assign sentences to topics after the initial clustering. This means that each time a new test

Proceedings of Recent Advances in Natural Language Processing, pages 519–526,
Varna, Bulgaria, Sep 4–6 2017.
https://doi.org/10.26615/978-954-452-049-6_068

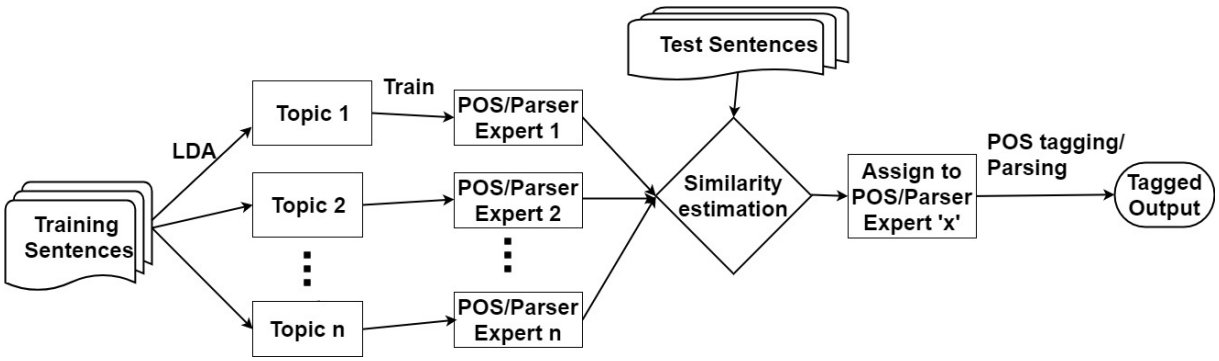

Figure 1: Overview of the architecture of the POS tagging and parsing experts.

sentence is encountered, the topic modeler is rerun to consistently determine the appropriate genres across training and test sentences. In order to avoid retraining, we propose to use similarity estimation techniques to determine which genre the test sentence belongs to. In this setup, we only create the experts once, and then assign new sentences to genres in an asynchronous fashion. For a new test sentence, we evaluate the similarity of the test sentence to the sentences of a genre and then assign it to the expert with the highest similarity score. We investigate a range of different techniques, based on 1) words closely associated with a topic by the LDA, 2) k-nearest neighbors, or 3) perplexity models to estimate the similarity.

Our results indicate that for word based similarity metrics, we reach results very similar to the joint topic modeling approach, thus proving the feasibility of an asynchronous approach. In the case of unigram-based perplexity, parsing accuracy even surpasses the joint modeling approach.

The remainder of the paper is structured as follows: Section 2 outlines our system architecture in greater detail, and section 3 discusses related work. In section 4, we describe the similarity methods, section 5 describes the setup for the experiments, and section 6 shows the results of our experiments. In section 7, we draw conclusions and discuss future steps.

2 Architecture

Following our previous work, we use LDA (Blei et al., 2003; Blei, 2012) to generate genres and train genre experts. But then, instead of having the test sentences clustered along with the training sentences, we assign test sentences to the genres via *similarity metrics*. The test sentences are consequently annotated by the corresponding genre expert. The complete architecture of our approach is shown in Figure 1.

For the current experiments, we use 2 topics, parallel to the 2 domains, and assign each training sentence via hard clustering to the genre for which LDA showed the highest probability[1]. We then test the different similarity metrics: We compute the similarity of a test sentence to the training sentences of the individual experts and then assign the sentence to the expert for tagging/parsing for which it has the highest similarity. We use the following similarity metrics:

- Topic words from LDA: LDA does not only cluster sentences into genres, it also determines which words are highly correlated with each genre. Thus, we can utilize these words along with their probabilities for a specific genre. We sum over all genre words that we find in the sentence, weighted by their probability, and then assign the sentence to the topic that has the highest score.

- k-nearest neighbors: There exist a wide range of metrics to calculate similarity between two feature vectors. However, since we compare a sentence to a set of genre sentences, we decided to use memory-based classification using the k-nearest neighbors to classify each test sentence into the relevant class (or genre, in our case). This has the advantage over a pure similarity metric that we have a principled way of handling the comparison to a set

[1]We have experimented with more topics for POS tagging and shown that we reach good results when using soft clustering rather than hard clustering (Mukherjee et al., 2016). The same holds true for experiments within the Penn Treebank, where we model WSJ internal topics, which are less distinct in their textual characteristics, but the experts still show an improvement over a full training baseline.

of sentences. Additionally, we do not consider the whole search space, as we would if we used a centroid.

- Perplexity: Another obvious choice for determining the similarity of a sentence to a set of sentences is language modeling. We calculate the perplexity of a test sentence with regard to set of sentences of an expert. We then assign a sentence to the expert for which it has the lowest perplexity.

3 Related Work

Our work cannot exactly be called domain adaptation since we automatically determine a range of genres and then train experts per genre while domain adaptation starts from a general model and adapts it to a specific genre. However, the two research questions are closely related and generally face the same problems. Our work is closest to the work by Plank and van Noord (2011) and McClosky et al. (2010). McClosky et al. address the problem of parse adaptation in the case of multiple sources. In their setting, a parser learns the domain differences and various statistics from being trained on datasets from multiple domains. Our approach is comparable to the extent that both approaches profit from a range of domains. Plank and van Noord (2011) adopt an approach where they build a highly specialized training set that is most similar to an out-of-domain document. They create a specialized expert each time the parser encounters a new document. Our approach is more general than Plank and van Noord's in that we do not create an expert for every new document, but rather create experts per genre. In contrast, our approach is more fine grained in that we assign individual sentences to genres rather than complete documents. Plank and van Noord (2011) use the topic distribution from LDA as features for determining the most similar training set. This is comparable to our approach of assigning test sentences to the proper genre (but not to the creation of genres).

Domain adaptation has been studied more extensively in parsing than in POS tagging. For POS tagging, Blitzer et al. (2006) have a similar setup to ours, where they train on WSJ data and test on MEDLINE abstracts. They use structural correspondence learning by identifying "frequently occurring" pivot features which can appropriately represent source as well as target domain. The problem of adapting to a new domain is compounded in cases where no adequate data from the target domain is available. Differences in annotation scheme between the source and target domain can pose additional challenges (Dredze et al., 2007). In our case, GENIA follows a similar annotation scheme as WSJ with a few differences in assigning POS tags to names.

Agreement based approaches and co-training have been employed for domain adaptation of POS tagging. In the agreement-based method adopted by Clark et al. (2003), a Markov model tagger and a maximum entropy tagger are used. For a sentence to be included in the training set, both the taggers have to reach a unanimous decision. Sagae and Tsujii (2007) apply a similar approach but using MaxEnt and SVM to simulate an iteration of co-training. Kübler and Baucom (2011) extended this further by demonstrating that an agreement in terms of word sequences rather than complete sentences is more robust and achieves better results.

In the CoNLL 2007 shared task on domain adaptation for dependency parsing, Attardi et al. (2007) adapt an error correction approach to revise mistakes caused by the base parser in the target domain. Kawahara and Uchimoto (2008) employ a single parser approach using a second order MST parser and combining labeled data from the known domain with unlabeled data of the new domain by simple concatenation and judging the efficacy of the resulting most reliable parses. Finkel and Manning (2009) devise a model for dependency parsing by using a hierarchical Bayesian prior based on the notion that different domains may have different features specific to each domain. Instead of applying a constant prior over all the parameters, a hierarchical Bayesian global is used.

4 Similarity Estimation

There are different ways of determining the similarity of a sentence to the sentences in a genre. We investigate methods based on the topic words from LDA, k-nearest neighbor approaches, and perplexity.

4.1 Topic Words from LDA

LDA provides a list of the words most closely associated with a topic and assigns a weight to each word. Thus, we can use the words that are highly correlated with each topic as good indicators for a sentence belonging to the genre represented by

the topic. Additionally, we utilize the probabilities provided by LDA as weights to determine a word's contribution to the similarity. For this experiment, we select the top 50/100/200 words from each topic. I.e., we assume that these words can be considered to be the most representative words in their respective domain. Then, for each sentence, we check how many of those words occur in the current sentence and add up their weights. We then assign the sentence to the topic with the higher value. Since we only look at a small number of words, we have to consider the cases where a sentence does not contain any of the topic words. We resolve these cases by extending beyond the top words and considering all the words in the training set for each genre. If there is a tie in values, we assign the sentence randomly to one of the experts.

4.2 k-Nearest Neighbors

In this method, we perform k-nearest neighbor classification to assign test sentences to topic experts. We create a feature vector by using the top 50/100 words from each genre, as determined by LDA, and assigning the weights as values. We tune the classification parameters on the validation set. For the setting using 50 words, the highest accuracy corresponds to a setting with 7 nearest neighbors, Dice coefficient as the distance metric, and gain ratio for feature weighting. When we increase the number of words per genre to 100, we use Dice coefficient, gain ratio, and 3 nearest neighbors.

4.3 Perplexity-Based Similarity

As the third set of methods, we turn to language modeling and use perplexity as a measure to determine the similarity, i.e., we assign test sentence to the expert which has a lower perplexity. Perplexity is calculated based on unigrams, bigrams, or trigrams.

5 Experimental Setup

5.1 Dataset

We create our corpus manually by combining the Wall Street Journal (WSJ) (Marcus et al., 1994) section of the Penn Treebank and the GENIA corpus (Tateisi and Tsujii, 2004). This gives us an artificial balanced corpus for which we know to which genre a sentence belongs. While WSJ consists of newspaper reports, GENIA consists of biomedical abstracts from Medline.

For the WSJ corpus, we use the POS annotation and syntactic annotations from the treebank. The GENIA Corpus is annotated on different linguistic levels, including POS tags, syntax, coreference, and events, among others. We use GENIA 1.0 trees (Ohta et al., 2002) created in the Penn Treebank format[2]. Both treebanks are converted to dependencies using pennconverter (Johansson and Nugues, 2007).

Following our data split in Mukherjee et al. (2016, 2017), we create a balanced dataset comprising 17 181 sentences from each corpus for the training set and 850 sentences for the test set. Since GENIA is rather small and since there is no standard data split for GENIA, we decided to extract the last 850 sentences for the test set, and the 850 sentences before that for the validation set. The remaining 17 181 sentences are used for training. For WSJ, we chose the same number of sentences for the training, validation, and test set, the training sentences are selected randomly from sections 02-21 and the validation and test sentences from section 22 and 23 respectively.

5.2 Baselines

In Mukherjee et al. (2016, 2017), we have used two baseline cases: The first baseline considers the entire training set and does not employ any topic modeling. Since the topic experts have access to only a fraction of the data, a second and more comparable baseline consists of randomly distributing training and test sentences into sets that correspond in size to the genres. We use these baselines and add a third, which is more relevant to our current setting. In this case, we use the experts trained on the genres but then randomly assign test sentences to the experts. This allows us to gauge how important a correct assignment to the corresponding expert is.

5.3 Topic Modeling

Probabilistic topic modeling is a class of unsupervised algorithms which detects the thematic structure in volumes of documents (Blei, 2012). We use Latent Dirichlet Allocation (LDA), a generative probabilistic model that approximates the underlying hidden topical structure of a collection of texts based on the distribution of words in the documents (Blei et al., 2003).

[2]http://nlp.stanford.edu/ mcclosky/biomedical.html

Similarity metric	Setting	Accuracy
joint LDA		98.94
topic words	50	97.59
	100	97.53
	200	97.35
perplexity	unigrams	**99.76**
	bigrams	84.71
	trigrams	81.53
k-NN	50	90.59
	100	91.18

Table 1: Accuracy of genre assignment for different similarity metrics.

We use the topic modeling toolkit MALLET (McCallum, 2002). The topic modeler in MALLET implements Latent Dirichlet Allocation clustering documents into a predefined number of topics.

5.4 POS Tagging and Parsing

For part of speech tagging, we use the Markov model POS tagger TnT (Trigrams'n'Tags) (Brants, 2000). We use TnT mainly because of its speed and because it allows the manual inspection of the trained models (emission and transition frequencies). For the parsing experiments, we use the dependency parser of the MATE Tools[3], a Java implementation of a graph-based parser (Bohnet, 2010). For the parsing experiments, we use gold POS tags.

5.5 Similarity Estimation

For the k-nearest neighbor estimation, we use the Tilburg Memory-Based Learner (TiMBL) (Daelemans et al., 2010). For the perplexity models, we derive n-grams of the training experts using Laplace smoothing, in NLTK (Bird et al., 2009), and compute the perplexity of the training experts to a test sentence.

5.6 Evaluation

We use the script `tnt-diff` that is part of TnT to evaluate the POS tagging results and the CoNLL shared task evaluation script[4] for evaluating the parsing results.

6 Experimental Results

Genre Assignment. We first investigate how well the different similarity metrics can assign the test sentences to the correct genre. I.e., we calculate accuracy in terms of whether a GENIA sentence is assigned to the GENIA genre, and a WSJ sentence to the WSJ genre. Table 1 shows the results of classification accuracy of using different similarity estimation techniques. The reference here is the joint LDA for training and test data, with an accuracy of 98.94%.

Perplexity based on unigrams reaches the highest accuracy, reaching an accuracy of 99.76%, thus surpassing the joint LDA. Surprisingly, using bi- or trigrams instead decreases accuracy by 15-20 points absolute. We assume that this is due to data sparsity since the language model is trained on a relatively small dataset. The second highest accuracy is reached by the methods based on topic words. Here, the number of words considered does not seem to make a significant difference. The k-NN approach performs at around 91%. These results indicate that using single words without context (i.e., the context in bi- and trigrams) provides the most reliable information. The language model has an additional advantage, potentially because it can smooth over unseen words.

POS Tagging. Table 2 shows the accuracies of the POS tagging experiments for different similarity metrics. We notice that assigning the test sentences randomly to genres has a detrimental effect, and we reach an accuracy of 91.36%, which is more than 5 points absolute lower than the random split baseline. This difference shows how important it is that sentences are assigned to the correct genre.

Perplexity based on unigrams and the method using topic words reach comparable accuracies to the original topic expert results based on the joint LDA clustering. These results follow the same

[3] code.google.com/p/mate-tools

[4] http://ilk.uvt.nl/conll/software/eval.pl

Setting	Similarity metric	Accuracy
full training		96.69
random split		96.41
topic experts + random test		91.36
joint LDA		96.95
topic words	50	96.80
	100	96.81
	200	96.81
perplexity	unigrams	**96.92**
	bigrams	95.64
	trigrams	95.22
k-NN	50	96.05
	100	96.09

Table 2: Results for the POS tagging experiments.

Setting	Similarity metric	LAS	UAS
full training		88.67	91.71
random split		87.84	90.86
topic experts + random test		82.17	88.13
joint LDA	-	90.51	92.14
topic words	50	90.30	92.07
	100	90.30	92.07
	200	90.30	92.07
perplexity	unigrams	**90.54**	**92.16**
	bigrams	88.33	91.13
	trigrams	87.50	90.68
k-NN	50	89.45	91.82
	100	89.52	91.84

Table 3: Attachment scores for the dependency parsing experiments.

trends as the genre assignment accuracies, but the differences between the methods are smaller. The perplexity setting using unigrams does not only surpass the joint LDA scores but also all the baselines: by nearly 0.3 percent points for the full training set, by 0.5 percent points for the random split, and by 5 percent points for the random test assignment.

Dependency Parsing. Table 3 shows labeled and unlabeled attachment scores of the dependency parses. These results mirror the trends in the POS tagging experiments: Randomly assigning sentences to genres results in the lowest LAS score of 82.17%, and using perplexity based on unigrams reaches the highest LAS of 90.54%. This LAS is considerably higher than the full training baseline of 88.67%. Note that the differences between the accuracies based on different similarity metrics are considerably more pronounced than in the POS tagging experiments. This mirrors the trend that we have previously seen for the joint LDA assignment. It is also interesting to see that the results for all the topic word settings are the same. This is due to 5 sentences that did not contain any of the topic words and thus had to be randomly assigned to one genre.

We now have a closer look at the two best settings, i.e., the perplexity experiment using unigrams and 50 topic words: We separate the sentences that were assigned to the wrong genre from the correctly assigned ones and evaluate them separately. For the unigram setting, 4 sentences were assigned to the wrong genre, for the 50 topic words, 41 sentences. The results are shown in Table 4. These results show that the sentences that were assigned to the wrong genre receive POS and dependency analyses with significantly lower accuracies, the difference to the correct ones ranging between 5 points absolute for POS tag-

	setting	Correct genre	Incorrect genre	Overall
POS tagging (acc.)	unigram	96.92	91.84	96.92
	topic words 50	98.23	88.59	96.80
parsing (LAS)	unigram	90.54	85.72	90.54
	topic words 50	90.55	76.19	90.30

Table 4: Results for POS tagging and dependency parsing when we separate incorrectly assigned sentences from correct ones.

ging and 10-14 points for parsing. This corroborates our findings that the correct assignment to a genre is of utmost importance, which also corroborates our conclusion that the genre experts model genre-specific information. If they did not, misassigning sentences would not have any impact.

7 Conclusion

Using topic modeling to create experts can be very beneficial, but this approach is only viable if we can assign new sentences asynchronously to genres without having to retrain the LDA to determine genres that include the new sentences. We have investigated similarity based methods for assigning the new sentences to genres. More specifically, we have investigated the following methods: 1) using topic words that LDA associates with a genre, 2) k-nearest neighbor models, and 3) perplexity in language models. A baseline that assigns test sentences randomly to genres performs poorly, thus showing that the correct assignment to genres is indispensable.

Our results show that the perplexity model based on unigrams surpasses the accuracy of a joint LDA model that assigns the sentences synchronously. For POS tagging, the accuracy of the unigram perplexity model is very close to that of the joint LDA. For parsing, the unigram perplexity model outperforms the joint LDA model. Using the 50 topic words to assign sentences to their genre reaches accuracies close to the best performing model. This shows that word-based methods are more robust in comparison to bigram and trigram methods, which should be able to profit from more context but also face data sparsity issues.

For the future, we plan to investigate models with a more dynamic mix of genres during the POS tagging and parsing process. I.e. rather than creating independent experts, we will investigate methods to create a POS tagger and parser that have access to expert views during each decision about the next POS tag or parsing step. We will also investigate whether we can integrate gold POS tags or dependency information into the topic modeling process, so that the topic modeler has access not only to the specialized lexical information but also to the linguistic information that it is ultimately tasked to distinguish.

References

Giuseppe Attardi, Felice Dell'Orletta, Maria Simi, Atanas Chanev, and Massimiliano Ciaramita. 2007. Multilingual dependency parsing and domain adaptation using DeSR. In *Proceedings of the CoNLL Shared Task Session of EMNLP-CoNLL 2007*. Prague, Czech Republic, pages 1112–1118.

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit*. O'Reilly Media.

David M. Blei. 2012. Probabilistic topic models. *Communications of the ACM* 55(4):77–84. https://doi.org/10.1145/2133806.2133826.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent Dirichlet Allocation. *Journal of Machine Learning Research* 3:993–1022.

John Blitzer, Ryan McDonald, and Fernando Pereira. 2006. Domain adaptation with structural correspondence learning. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Sydney, Australia, pages 120–128.

Bernd Bohnet. 2010. Top accuracy and fast dependency parsing is not a contradiction. In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING)*. Beijing, China, pages 89–97.

Thorsten Brants. 2000. TnT–a statistical part-of-speech tagger. In *Proceedings of the 1st Conference of the North American Chapter of the Association for Computational Linguistics and the 6th Conference on Applied Natural Language Processing (ANLP/NAACL)*. Seattle, WA, pages 224–231.

Stephen Clark, James Curran, and Miles Osborne. 2003. Bootstrapping POS-taggers using unlabelled data. In *Proceedings of the Seventh Conference on Natural Language Learning (CoNLL)*. Edmonton, Canada, pages 49–55.

Walter Daelemans, Jakub Zavrel, Ko van der Sloot, and Antal van den Bosch. 2010. TiMBL: Tilburg memory based learner – version 6.3 – reference guide. Technical Report ILK 10-01, Induction of Linguistic Knowledge, Computational Linguistics, Tilburg University.

Mark Dredze, John Blitzer, Partha Pratim Talukdar, Kuzman Ganchev, João Graca, and Fernando Pereira. 2007. Frustratingly hard domain adaptation for dependency parsing. In *Proceedings of the CoNLL Shared Task Session of EMNLP-CoNLL 2007*. Prague, Czech Republic, pages 1051–1055.

Jenny Rose Finkel and Christopher D. Manning. 2009. Hierarchical Bayesian domain adaptation. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL)*. pages 602–610.

Richard Johansson and Pierre Nugues. 2007. Extended constituent-to-dependency conversion for English. In *Proceedings of NODALIDA 2007*. Tartu, Estonia, pages 105–112.

Daisuke Kawahara and Kiyotaka Uchimoto. 2008. Learning reliability of parses for domain adaptation of dependency parsing. In *Proceedings of the Third International Joint Conference on Natural Language Processing (IJCNLP)*. Hyderabad, India.

Sandra Kübler and Eric Baucom. 2011. Fast domain adaptation for part of speech tagging for dialogues. In *Proceedings of the International Conference on Recent Advances in NLP (RANLP)*. Hissar, Bulgaria.

Mitchell Marcus, Grace Kim, Mary Ann Marcinkiewicz, Robert MacIntyre, Ann Bies, Mark Ferguson, Karen Katz, and Britta Schasberger. 1994. The Penn Treebank: Annotating predicate argument structure. In *Proceedings of the ARPA Human Language Technology Workshop, HLT 94*. Plainsboro, NJ, pages 114–119.

Andrew Kachites McCallum. 2002. Mallet: A machine learning for language toolkit. `http://mallet.cs.umass.edu`.

David McClosky, Eugene Charniak, and Mark Johnson. 2010. Automatic domain adaptation for parsing. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*. Los Angeles, CA, pages 28–36.

Atreyee Mukherjee, Sandra Kübler, and Matthias Scheutz. 2016. POS tagging experts via topic modeling. In *Proceedings of the 13th International Conference on Natural Language Processing*. Varanasi, India, pages 120–128.

Atreyee Mukherjee, Sandra Kübler, and Matthias Scheutz. 2017. Creating POS tagging and dependency parsing experts via topic modeling. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*. pages 347–355.

Tomoko Ohta, Yuka Tateisi, and Jin-Dong Kim. 2002. The GENIA corpus: An annotated research abstract corpus in molecular biology domain. In *Proceedings of the Second International Conference on Human Language Technology Research*. San Francisco, CA, pages 82–86.

Barbara Plank and Gertjan van Noord. 2011. Effective measures of domain similarity for parsing. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*. Portland, OR, pages 1566–1576.

Kenji Sagae and Jun'ichi Tsujii. 2007. Dependency parsing and domain adaptation with LR models and parser ensembles. In *Proceedings of the CoNLL Shared Task Session of EMNLP-CoNLL 2007*. Prague, Czech Republic, pages 1044–1050.

Yuka Tateisi and Jun'ichi Tsujii. 2004. Part-of-speech annotation of biology research abstracts. In *Proceedings of 4th International Conference on Language Resource and Evaluation (LREC)*. Lisbon, Portugal.