Approximating Learning Curves for Active-Learning-Driven Annotation

Katrin Tomanek and Udo Hahn

Jena University Language & Information Engineering (JULIE) Lab
Friedrich-Schiller-Universität Jena
Fürstengraben 30, D-07743 Jena, Germany
{katrin.tomanek|udo.hahn}@uni-jena.de

Abstract

Active learning (AL) is getting more and more popular as a methodology to considerably reduce the annotation effort when building training material for statistical learning methods for various NLP tasks. A crucial issue rarely addressed, however, is when to actually stop the annotation process to profit from the savings in efforts. This question is tightly related to estimating the classifier performance after a certain amount of data has already been annotated. While learning curves are the default means to monitor the progress of the annotation process in terms of classifier performance, this requires a labeled gold standard which – in realistic annotation settings, at least – is often unavailable. We here propose a method for committee-based AL to approximate the progression of the learning curve based on the disagreement among the committee members. This method relies on a separate, unlabeled corpus and is thus well suited for situations where a labeled gold standard is not available or would be too expensive to obtain. Considering named entity recognition as a test case we provide empirical evidence that this approach works well under simulation as well as under real-world annotation conditions.

1. Introduction

State-of-the-art NLP components are increasingly based on supervised machine learning methods. This raises the need for large amounts of training data. While for the general language English newspaper domain syntactic (Marcus et al., 1993), semantic (Palmer et al., 2005; Pustejovsky et al., 2003), and even discourse (Carlson et al., 2003; Miltsakaki et al., 2008) annotations are increasingly made available, any language, domain, or genre shift pushes the severe burden on developers of NLP systems to supply comparably sized high-quality annotations. Even inner-domain shifts, such as, e.g., moving from hematology (Ohta et al., 2002) to the genetics of cancer (Kulick et al., 2004) within the field of molecular biology may have drastic consequences in the sense that entirely new meta data sets have to produced by annotation teams. Thus, reducing the human efforts for the creation of adequate training material is a major challenge.

Active learning (AL) copes with this problem as it intelligently selects the data to be labeled. It is a sampling strategy where the learner has control over the training material to be manually annotated by selecting those examples which are of high utility for the learning process. AL has been successfully applied to speed up the annotation process for many NLP tasks without sacrificing annotation quality (Engelson and Dagan, 1996; Ngai and Yarowsky, 2000; Hwa, 2001; Tomanek et al., 2007a).

Once we decide to use AL for meta-data annotation and a reasonable, stable level of annotation quality is reached – after having run through only a fraction of the documents compared with the traditional annotation approach where a randomly and independently selected amount of documents is sequentially annotated – an obvious question turns up: When do we stop the annotation process to cash in the time savings? Stopping after a certain amount of time has elapsed or a certain amount of data has been annotated is clearly not the best choice since such criteria, easily applicable though, do not take into account how well a classifier trained on the annotated data really performs. An optimal stopping condition for any annotation would be to locate that point in time when no further improvement in terms of classifier performance can be achieved by additional annotations. Since learning curves show the classifier performance at different time steps, i.e., for different amounts of annotated training examples, we can observe that progression. Given this observation data we may stop the annotation process when the learning curve completely converges and is not ascending any more.

In most real-world annotation scenarios, however, such a well-defined stopping point based on the convergence of classifier performance does not exist. Instead, additional annotations often result in slight improvements of the classifier's performance. Accordingly, one should rather consider the *trade-off* between further annotation efforts and gains in classifier performance to decide whether additional annotations are worth the effort for targeted application.

This trade-off can be read from the learning curve which, unfortunately, will not always be available. Re-sampling strategies, e.g., cross-validation or bootstrapping, usually applied to estimate classifier performance, assume independently and identically distributed (*i.i.d.*) examples to sample from. But examples selected by means of AL do not meet this requirement. So, to estimate classifier performance a separately annotated gold standard with i.i.d. examples is often used to obtain a learning curve for AL. Yet, this solution comes with expensive extra annotation work.

We present an approach to approximate the progression of the learning curve without the need for a labeled gold standard. We situate our discussion in the context of a simulation and a real-world annotation scenario and will find out that the second scenario imposes some restrictions on the configuration of the approach. The paper is structured as follows: In Section 2., we describe our approach in detail. Other work on stopping conditions for AL-based annotation is discussed in Section 3. Experimental results for the task of named entity recognition are presented in Section 4.

2. Approximating the Learning Curve

Given the idea that from the learning curve one can read the trade-off between annotation effort and classifier performance gain, we here propose an approach to approximate the progression of the learning curve which comes at no extra annotation costs. This approach is designed for use in committee-based AL (Seung et al., 1992). A committee consists of k classifiers of the same type trained on different subsets of the already labeled (training) data. Each committee member then makes its predictions on the pool of unlabeled examples, and those examples on which the committee members express the highest disagreement are considered most informative for learning and are thus selected for manual annotation.

To calculate the disagreement among the committee members several metrics have been proposed including the vote entropy (Engelson and Dagan, 1996) as possibly the most well-known one. Our approach to approximating the learning curve is based on the disagreement within a committee. However, it is independent of the actual metric used to calculate the disagreement. Although in our experiments we considered the NLP task of named entity recognition (NER) only, our approach is not limited to this scenario and can be expected to be applicable to other tasks as well.

In Tomanek et al. (2007a) we introduced the *selection agreement* (SA) curve – the average agreement amongst the selected examples plotted over time. When the SA values are close to '1', the committee members almost perfectly agree. So, any further AL iteration would resemble a random selection. Experiments have shown that at the point where the SA curve converges on values close to '1' the respective learning curve converges on its maximum value as well so that further annotations would have (almost) no impact on the classifier performance. As a result, we concluded that we can derive, from the SA curve, the point where the classifier performance is not increased any more by further annotation efforts. Hence, when this curve approaches values of '1' it can be interpreted as a stopping signal for annotation.

However, this positive finding is due to an inherent feature of AL *simulations*. In typical simulation settings, the pool of annotation items is of a very limited size – normally only a few thousand examples. This is so because for simulations, a pre-annotated corpus is used and the manual annotation is simulated by just moving selected examples from the pool to the training set unveiling the labels. As a consequence, the total number of positive and hard examples, which are preferentially selected by AL, is rather limited.

In the NER scenario, examples to be selected are complete sentences. Sentences containing (many) entity mentions can be considered as "positive" ones. Especially when very infrequent entity classes are to be annotated, a corpus will consist of a large proportion of "negative" examples which contain no entity mentions at all. In our experiments, we observed that sentences which contained many and complex entity mentions were already selected in early AL iterations. Thus, the more AL iterations are run, the less hard and positive examples are left in the pool. As a consequence, only in early iterations, AL really has choices to select useful examples.

The SA curve is directly affected by this *simulation effect* and thus cannot be used as a reliable approximation of the learning curve in a real-world annotation scenario where the pool will be much larger and much more diverse. In such a setting there will always be useful (and, by this, hard) examples which AL may find, thus keeping the selection agreement constantly high.

The solution we propose is to calculate the average agreement for each AL iteration on a separate *validation set* which should reflect the real data distribution and must not be used in the annotation process itself. As for most NLP tasks there is no limit to unlabeled data and no annotations are required, the validation set comes at no extra costs. Plotted over time we get the *validation set agreement* (VSA) curve. This curve is based on the same data in each AL iteration making the agreement values comparable between different AL iterations. Since the examples of the validation set are not used in the annotation process we can further guarantee that this curve is only affected by the benefit the selected and labeled examples have on training a classifier. Now, from a VSA curve which is only slightly ascending between selected measurement points we can infer that the respective learning curve has only a low slope at these positions, too. Although interpreting the actual agreement values of the VSA curve is still problematic, its progression behavior can be used to estimate whether further annotation is worth the human labeling effort. In Section 4., we will provide empirical evidence that the VSA curve is indeed an adequate approximation of the progression of the learning curve and that the SA curve fails in the real-world annotation scenario where examples are selected from a much larger pool.

3. Related Work

While there is a large body of work on AL proper, there are only few papers reporting on stopping criteria or methods to monitor the progress of AL-driven annotations. Schohn and Cohn (2000) consider an AL approach for Support Vector Machines (SVM) where examples are selected according to their proximity to the hyperplane. They propose to stop the annotation process when, in the current AL iteration, none of the unlabeled examples are closer to the hyperplane than the support vectors. While this approach is restricted to AL for SVMs, Vlachos (2008) presents a stopping criterion for uncertainty-based AL (Cohn et al., 1996) in general. The confidence of the classifier at the current AL iteration is estimated on a large, separate validation set. The author reports that such a confidence curve follows a rise-peak-drop pattern: It rises in the beginning, then reaches its maximum values, and after that it constantly drops. The stopping condition is then defined as the point when the confidence curve starts dropping, i.e., the point when the learning curve has converged. This approach is similar to ours in that it employs the usefulness measure of the AL selection and in that it applies a separate validation set to calculate the confidence curve on. However, while it provides an exact *stopping* condition, it cannot provide a means to estimate the progression of the learning curve. This is equally important since, in practice, one might want to stop the annotation before such a final stopping condition is met, e.g., when the

trade-off between additional annotation costs and gain in classifier performance is falling below some threshold.
For uncertainty-based AL, further stopping criteria employing a confidence estimate of the current classifier were proposed by Zhu et al. (2008). The first one is based on an uncertainty measurement on all unlabeled examples of a pool, the second one uses the prediction accuracy on the selected examples, and the final one builds on the classifier's expected error on all unlabeled examples. Since these approaches are not based on a separate validation set we assume that their reported success rates are largely due to the simulation effect, i.e., the limited number of 'hard' examples in a simulation data set. Whereas the first and the third criterion could also be applied in a separate, unlabeled validation set to avoid this shortcoming, the second one would require an annotated validation set – not really an advantage over plotting a learning curve. Further on, Zhu et al. use their approaches as stopping condition by comparing the respective values against a *fixed* threshold. We find this problematic because a priori chosen or heuristically determined values are highly task- and data-dependent. In a real-world annotation scenario it is almost impossible to adequately define such values in advance.
While all the above-mentioned approaches focus on single-classifier AL strategies, ours is tailored to committee-based AL.

4. Experiments

To empirically test whether our proposed approach works well as an approximation of the learning curves we ran several experiments both in a pure simulation mode, where the manual annotation was simulated by unveiling the labels already assigned in the simulation corpus, and in a real-world scenario where human annotators were asked to annotate the sentences selected by AL. For both scenarios the selection agreement (SA) and the validation set agreement (VSA) was calculated for each AL iteration.

4.1. Experimental Settings

For our experiments on approximating the learning curves for AL-based selection, we chose named entity recognition (NER) as the annotation task in focus. We employed the committee-based AL approach described in Tomanek et al. (2007a). The committee consists of $k = 3$ Maximum Entropy (ME) classifiers (Berger et al., 1996). In each AL iteration, each classifier is trained on a randomly[1] drawn (sampling without replacement) subset $L' \subset L$ with $|L'| = \frac{2}{3}$, L being the set of all examples seen so far. Disagreement is measured by vote entropy (Engelson and Dagan, 1996). In our NER scenario, complete sentences are selected by AL. While we made use of ME classifiers during the selection, we employed a NE tagger based on Conditional Random Fields (CRFs) (Lafferty et al., 2001) during evaluation time to determine the learning curves. We have already shown that in this scenario, ME classifiers perform equally well for AL-driven *selection* as CRFs when using the same features.

[1]The random selection of the training material for the classifiers explains why our agreement curves sometimes have outliers: A suboptimally sampled committee results in suboptimal classifiers and thus in high agreement values.

scenario	corpus	seed	pool	gold
simulation	CoNLL	20	14,000	3,453
simulation	PBVAR	20	10,020	1,114
real annotation	CDANTIGEN	853	$\approx$ 2 m	2,165
real annotation	CYTOREC	256	$\approx$ 2 m	2,165

Table 1: Corpora used for the experiments (size of seed set, pool, and gold standard in the number of sentences)

This effect is truly beneficial, especially for real-world annotation projects, due to much lower training times and, by this, shorter annotator idle times (Tomanek et al., 2007a).
For the AL simulation, we employed two simulation corpora: The CoNLL corpus, based on the English data set of the CoNLL-2003 shared task (Tjong Kim Sang and De Meulder, 2003), which consists of newspaper articles annotated with respect to person, location, and organisation entities. This pool consists of about 14,000 sentences. As validation set and as gold standard for plotting the learning curve we used CoNLL's evaluation corpus which sums up to 3,453 sentences. The PBVAR corpus consists of biomedical abstracts and was derived from the PENNBIOIE corpus (Kulick et al., 2004) by keeping only those annotations related to variation event mentions. We have randomly split this corpus into a pool set and a validation/gold set. In our simulations, 20 sentences were selected in each AL iteration and the simulations were started with a random seed set of 20 sentences. Our results are averaged over three independent runs.
For the real-world annotation scenario, we considered two sub-corpora from the entity annotations described in (Hahn et al., 2008): The cytokine and growth factor receptors corpus (CYTOREC) is annotated with various entity subclasses of special receptor entities, while the antigens corpus (CDANTIGEN) contains annotations of various immunologically relevant antigen entities. For both annotation projects, the pool from which AL selected the examples to be labeled consisted of approximately 2 million sentences taken from PUBMED[2] abstracts, the validation set and gold standard was composed of 2,165 sentences. In each AL iteration, 30 sentences were selected for manual annotation. The corresponding seed sets were considerably larger than in our simulations and were assembled by the heuristic described by Tomanek et al. (2007b). Table 1 summarizes the corpora used for our experiments.

4.2. Results

Figures 1 and 2 display the learning and agreement curves for the CoNLL and the PBVAR corpus, respectively. The learning curves are depicted for both AL (solid line) and random selection (dashed line) revealing the increase in annotation efficiency when AL is used to select the examples to be annotated. As for the agreement curves, we plot both the exact agreement values (dots) and a curve obtained by local polynomial regression fitting (solid line).
On the CoNLL corpus, the learning curve converges on its maximum f-score ($\approx 84\%$) after about 125,000 tokens. This is reflected by the SA curve which is not ascending any

[2]http://www.ncbi.nlm.nih.gov/

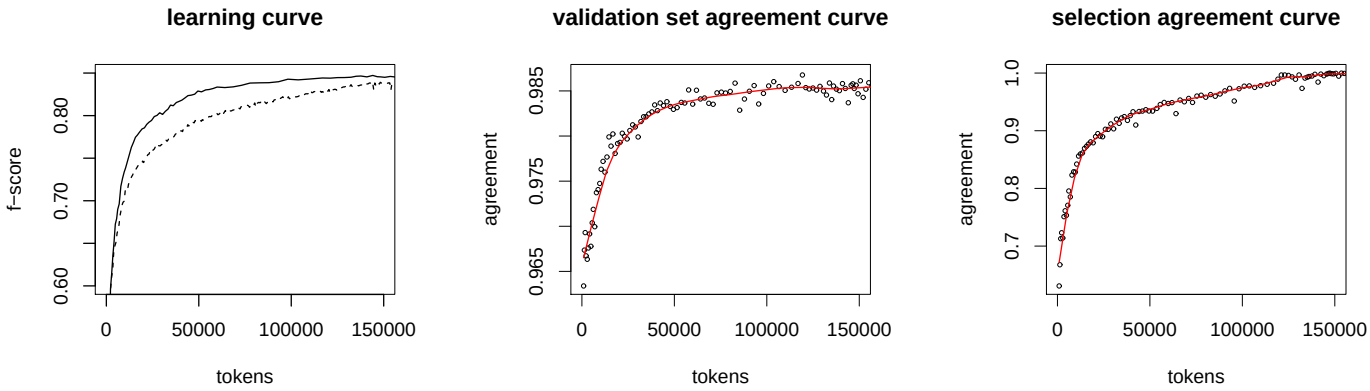

Figure 1: Learning curves (solid line: AL selection, dashed line: random selection) and agreement curves for CoNLL

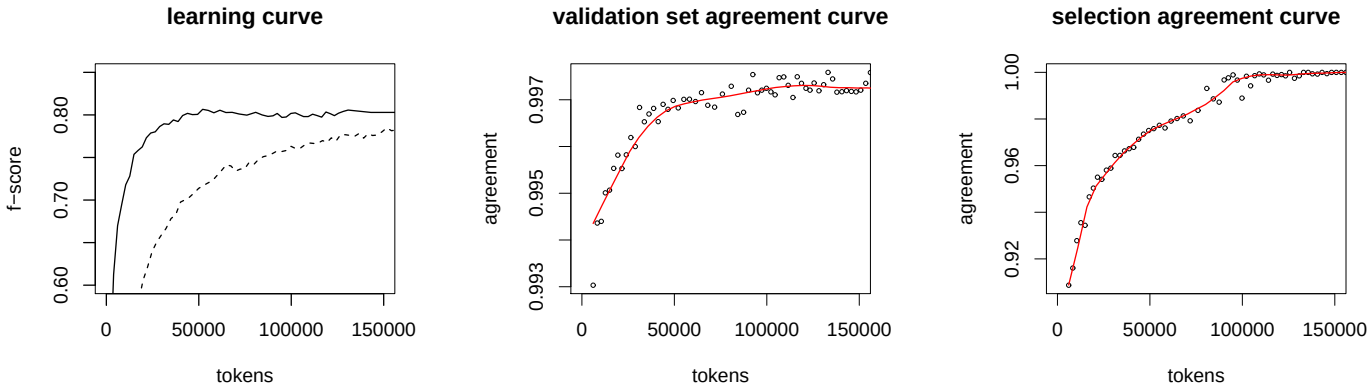

Figure 2: Learning curves (solid line: AL selection, dashed line: random selection) and agreement curves for PBVAR

more at about the same number of tokens. A similar pattern is depicted in the VSA curve though it provides an even clearer picture of the progression of the learning curve. It is only slightly ascending after about 50,000 tokens, i.e., at a time when the slope of the learning curve already becomes very low. From both the learning and the VSA curve we can read that after 50,000 tokens any additional annotation is very costly compared to its benefits in terms of increased classifier performance.

On the PBVAR corpus, the maximal f-score ($\approx$ 80%) is reached after approximately 50,000 tokens, then there is a small decline which after about 100,000 tokens stabilizes at the maximum value. The SA curve reached values around '1' after about $100,000$ tokens, but is misleading here since it does not reflect that the learning curve had already reached a maximum before. The VSA curve, however, more comprehensively approximates the behavior of the learning curve. It has a clear bend after some 50,000 tokens and converges after approximately 100,000 tokens.

Figures 3 and 4 display the learning and agreement curves for our experiments in the real-world annotation scenario. No learning curve for random selection is shown since only AL selection was performed to avoid unnecessary human efforts. Further, in this scenario, the agreement was not calculated during the selection to keep selection time as short as possible but was calculated afterwards for this experiment.[3] On both corpora, agreement as well as learning curves start with the complete seed set (256 sentences, with about 10,00 tokens for CYTOREC and 853 sentences, with some 35,000 tokens for CDANTIGEN).

On the CDANTIGEN corpus, after $80,000$ tokens being annotated the learning curve has not completely converged but additional annotations do not pay off either. The VSA curve mirrors this behavior since it keeps on ascending with a low slope, though the SA curve remains quite obscure, here. A similar behavior can be observed for the CYTOREC corpus. The learning curve is only slightly ascending after about 65,000 tokens have been annotated. This is nicely mirrored by the VSA curve. Again, the SA curve is almost impossible to interpret: Though its slope decreases a bit after roughly 40,000 tokens, it keeps ascending thereafter. Both SA curves exhibit an oscillating behavior that does not contain any clue to guide stopping decisions.

We have seen that in the simulation scenario the two agreement curves (SA and VSA) share a similar curve progression due to the simulation effect (cf. Figure 1 for the

[3]Due to the randomness when sampling the committee (see above), we averaged over three runs where we calculated the agreement curves. After every fifth AL iteration (i.e., 150 sentences selected) we calculated both the SA and the VSA curves.

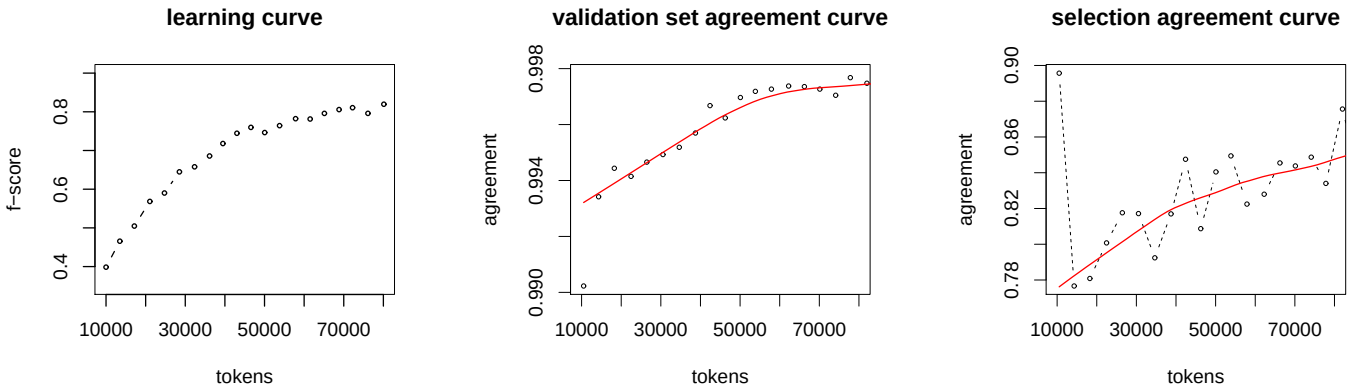

Figure 3: Learning curve (AL selection) and agreement curves for CYTOREC

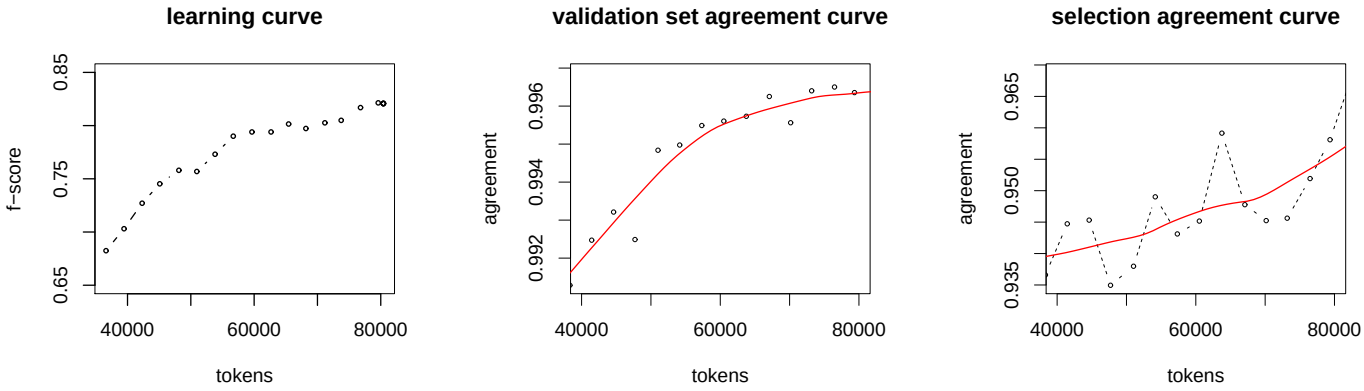

Figure 4: Learning curve (AL selection) and agreement curves for CDANTIGEN

CONLL corpus). But even in the simulation scenario the SA curve might be problematic and hence misleading as can be concluded from our experiments on the PBVAR corpus. In the real-world annotation scenario these SA curves are clueless to approximate the progression of the learning curve. However, our experiments suggest (see Figures 3 and 4) that the VSA curve is a good estimator for the progression of the learning curve and also works in practice, while the SA curve fails as a reliable predictor in our real-world annotation scenario. Still, even the more predictive VSA curves merely *guide* but do not *finalize* stopping decisions. So it is left to the annotation manager's over-all assessment to balance the trade-off between annotation costs and expectable quality gains for the learner.

5. Conclusions

In this paper, we discussed an approach to approximate the progression of the learning curve for AL-driven annotation. Such an approximation can be used to estimate the relative quality gains of further annotation efforts. This might render valuable decision support for the question when to actually stop an annotation process, in practice, and is especially helpful when a learning curve is not available due to the absence of a labeled gold standard.

We have deliberately refrained from defining a fixed stopping condition for AL-driven annotations. In practice, further annotation efforts will mostly result in *some*, although mild, classifier improvement. Whether the respective gain justifies the efforts (and costs) depends on the task at hand. As far as the learning curve and its approximation is concerned, the relative gains can be estimated. Such an approach might be more adequate for practical use cases rather than a single-point stopping condition which does not incorporate trade-off considerations of any sort.

Further, we have discussed that AL simulations are subject to the simulation effect. From our experiments we conclude that approaches to monitor the progress (in whatever manner) of AL-driven annotation should always be based on a separate validation set instead of the material directly involved in the AL training process. As the validation set does not need to be labeled and for almost all NLP applications unlabeled material is available in virtually unlimited volumes this approach comes at not extra costs.

Acknowledgements

This research was funded by the EC within the BOOTSTREP project (FP6-028099), and by the German Ministry of Education and Research within the STEMNET project (01DS001A-C).

6. References

Adam L. Berger, Stephen A. Della Pietra, and Vincent J. Della Pietra. 1996. A maximum entropy approach to natural language processing. *Computational Linguistics*, 22(1):39–71.

Lynn Carlson, Daniel Marcu, and Mary Ellen Okurowski. 2003. Building a discourse-tagged corpus in the framework of Rhetorical Structure Theory. In Jan van Kuppevelt and Ronnie Smith, editors, *Current Directions in Discourse and Dialogue*, pages 85–112. Dordrecht: Kluwer Academic.

David A. Cohn, Zoubin Ghahramani, and Michael I. Jordan. 1996. Active Learning with statistical models. *Journal of Artifical Intelligence Research*, 4:129–145.

Sean Engelson and Ido Dagan. 1996. Minimizing manual annotation cost in supervised training from corpora. In *ACL'96 – Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics*, pages 319–326. University of California at Santa Cruz, CA, USA, 24-27 June 1996.

Udo Hahn, Elena Beisswanger, Ekaterina Buyko, Michael Poprat, Katrin Tomanek, and Joachim Wermter. 2008. Semantic annotations for biology: A corpus development initiative at the Jena University Language & Information Engineering (JULIE) Lab. In *LREC 2008 – Proceedings of the 6th International Conference on Language Resources and Evaluation*. Marrakech, Morocco, June 28-30, 2008. Paris: European Language Resources Association (ELRA). *(this volume)*.

Rebecca Hwa. 2001. On minimizing training corpus for parser acquisition. In *CoNLL-2001 – Proceedings of the 5th Natural Language Learning Workshop*, pages 84–89. Toulouse, France, 6-7 July 2001.

Seth Kulick, Ann Bies, Mark Liberman, Mark Mandel, Ryan McDonald, Martha Palmer, Andrew Schein, Lyle Ungar, Scott Winters, and Pete White. 2004. Integrated annotation for biomedical information extraction. In *BioLink 2004 – Proceedings of the HLT-NAACL 2004 Workshop 'Linking Biological Literature, Ontologies and Databases: Tools for Users'*, pages 61–68. Boston, MA, USA, May 2004.

John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML-2001 – Proceedings of the 18th International Conference on Machine Learning*, pages 282–289. Williams College, MA, USA, June 28 - July 1, 2001.

Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of English: The PENN TREEBANK. *Computational Linguistics*, 19(2):313–330.

Eleni Miltsakaki, Livio Robaldo, Alan Lee, and Aravind Joshi. 2008. Sense annotation in the Penn Discourse Treebank. In Alexander F. Gelbukh, editor, *CICLing 2008 – Proceedings of the 9th International Conference on Computational Linguistics and Intelligent Text Processing*, pages 275–286. Haifa, Israel, February 17-23, 2008. Berlin: Springer.

Grace Ngai and David Yarowsky. 2000. Rule writing or annotation: Cost-efficient resource usage for base noun phrase chunking. In *ACL'00 – Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics*, pages 117–125. Hong Kong, China, 1-8 August 2000.

Tomoko Ohta, Yuka Tateisi, and Jin-Dong Kim. 2002. The GENIA corpus: An annotated research abstract corpus in molecular biology domain. In *HLT 2002 – Human Language Technology Conference. Proceedings of the 2nd International Conference on Human Language Technology Research*, pages 82–86. San Diego, CA, USA, March 24-27, 2002.

Martha Palmer, Daniel Gildea, and Paul Kingsbury. 2005. The Proposition Bank: An annotated corpus of semantic roles. *Computational Linguistics*, 31(1):71–106.

James Pustejovsky, Patrick Hanks, Roser Saurí, Andrew See, Robert Gaizauskas, Andrea Setzer, Dragomir Radev, Beth Sundheim, David Day, Lisa Ferro, and Marcia Lazo. 2003. The TIMEBANK corpus. In *Proceedings of the Corpus Linguistics 2003 Conference*, pages 647–656. Lancaster University, U.K., 28-31 March 2003.

Greg Schohn and David Cohn. 2000. Less is more: Active Learning with Support Vector Machines. In *ICML 2000 – Proceedings of the 17th International Conference on Machine Learning*, pages 839–846. Stanford, CA, USA, June 29 - July 2, 2000.

H. Sebastian Seung, Manfred Opper, and Haim Sompolinsky. 1992. Query by committee. In *COLT'92 – Proceedings of the 5th Annual Conference on Computational Learning Theory*, pages 287–294. Pittsburgh, PA, USA, July 27-29, 1992. New York, NY: ACM Press.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CONLL-2003 shared task: Language-independent named entity recognition. In *CoNLL-2003 – Proceedings of the 7th Conference on Computational Natural Language Learning*, pages 142–147. Edmonton, Canada, 2003.

Katrin Tomanek, Joachim Wermter, and Udo Hahn. 2007a. An approach to text corpus construction which cuts annotation costs and maintains reusability of annotated data. In *EMNLP-CoNLL 2007 — Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 486–495. Prague, Czech Republic, June 28-30, 2007.

Katrin Tomanek, Joachim Wermter, and Udo Hahn. 2007b. Efficient annotation with the Jena ANnotation Environment (JANE). In *The LAW at ACL 2007 – Proceedings of the Linguistic Annotation Workshop*, pages 9–16. Prague, Czech Republic, June 28-29, 2007.

Andreas Vlachos. 2008. A stopping criterion for active learning. *Computer Speech and Language*, 22:295-312.

Jingbo Zhu, Huizhen Wang, and Eduard Hovy. 2008. Learning a stopping criterion for Active Learning for word sense disambiguation and text classification. In *IJCNLP 2008 – Proceedings of the 3rd International Joint Conference on Natural Language Processing*, pages 366–372. Hyderabad, India, January 7-12, 2008.