There's no Data like More Data? Revisiting the Impact of Data Size on a Classification Task

Ines Rehbein, Josef Ruppenhofer

Saarland University
Saarbrücken, Germany
{rehbein,josefr}@coli.uni-sb.de

Abstract

In the paper we investigate the impact of data size on a Word Sense Disambiguation task (WSD). We question the assumption that the knowledge acquisition bottleneck, which is known as one of the major challenges for WSD, can be solved by simply obtaining more and more training data. Our case study on 1,000 manually annotated instances of the German verb *drohen* (threaten) shows that the best performance is not obtained when training on the full data set, but by carefully selecting new training instances with regard to their informativeness for the learning process (Active Learning). We present a thorough evaluation of the impact of different sampling methods on the data sets and propose an improved method for uncertainty sampling which dynamically adapts the selection of new instances to the learning progress of the classifier, resulting in more robust results during the initial stages of learning. A qualitative error analysis identifies problems for automatic WSD and discusses the reasons for the great gap in performance between human annotators and our automatic WSD system.

1. Introduction

Nowadays, supervised ML-based approaches are the mainstay of many NLP applications, relying on a reasonably sized amount of manually annotated training data. The creation of high quality language resources, however, is time-consuming and costly. Recently, two different approaches have been used successfully to minimise the problem. The first applies semi-supervised techniques to bootstrap more training data from unannotated text (Yarowsky, 1995). The second, referred to as *Active Learning* (AL) (Cohn et al., 1996), tries to minimise human annotation effort by carefully selecting the most informative training instances, thus reducing the size of the training data while preserving accuracy.

Bootstrapping assumes that adding enough (even noisy) training data can solve most of the classification problems in NLP. By contrast, AL assumes that more training data does not always improve results, and that a smaller number of high-quality training instances reduces the time and cost of annotation. In the paper we address whether NLP classification systems can be improved by simply enlarging the amount of training data, or if there is an upper bound that we cannot exceed with sheer size. We chose Word Sense Disambiguation as a typical classification task in NLP, and present experiments in an AL setting.

The contribution of the paper is two-fold: we present MaJo, a new toolkit for WSD using AL which combines an easy-to-use graphical user interface with support for feature exploration.[1] We propose an improved method for uncertainty sampling, taking into account what the classifier has learned so far, increasing the performance of AL in the early training stages. On a theoretical level, we contribute to the ongoing discussion on the impact of training size on classification tasks, arguing that quality is better than quantity, and that further improvements in WSD cannot be gained by using ever more training data. We show that a large amount of training data for a verb whose senses are very easy to distinguish for humans still does not allow the classifier to discriminate between the word senses, and discuss the main reasons for the gap in performance between our WSD system and human annotators.

The paper is structured as follows. Section 2. gives a brief overview of the Active Learning paradigm and discusses related work investigating the impact of training size on a classification task. Section 3. describes the AL process with MaJo. In Section 4. we present experiments on WSD, using a simulated AL approach, and describe our improved sampling method based on uncertainty sampling. In Section 5. we discuss our results and their impact on the question of the appropriate size of training data. The last section concludes and outlines future work.

2. Active Learning

The basic idea in AL is to reduce the amount of human annotation by selecting new training instances according to their informativeness for the machine learning classifier. Selected instances are passed to a human annotator, the *oracle*, who assigns the correct label. Instead of annotating a large number of instances, Active Learning seeks to select those instances from a large pool of sentences which provide important information for the machine learner. Thus guiding the learning process by providing the information the classifier still needs to learn, a smaller number of instances should suffice to achieve the same accuracy as on a larger training set of randomly selected training examples.

2.1. Related Work

Although AL has been shown to be useful for WSD in general (e.g. Chen et al. (2006)), some open issues remain. Recent work has explored the impact of different parameters on the performance of AL. Among them are the size of the seed data; techniques for selecting new examples to be labeled by the human oracle; possible stopping criteria for

[1] MaJo is freely available for research purposes (http://www.coli.uni-saarland.de/projects/salsa).

when to end the AL process (e.g. Vlachos (2008), Bloodgood and Shanker (2009)). Other key issues include the grain size of sense distinctions and the distribution of word senses in the data. It is not yet clear whether AL works only for coarse-grained sense distinctions (Dang, 2004) or also for fine-grained ones (Chan and Ng, 2007).

3. MaJo - A Graphical User Interface for WSD using Active Learning

The MaJo toolkit for supervised Word Sense Disambiguation (WSD) provides a testing environment for exploring the questions outlined in Section 2.1. Its graphical user interface allows users with little or no programming skills to investigate the interaction between different feature sets and settings for sampling and to systematically explore the impact of the different parameters on the Active Learning task. MaJo features a flexible plugin architecture which implements a number of interfaces to off-the-shelf NLP tools and linguistic resources for extracting training data from the web (Yahoo! search API), for preprocessing (Stanford POS Tagger (Toutanova et al., 2003), Stanford Parser (Klein and Manning, 2003), Berkeley Parser (Petrov and Klein, 2007), MaltParser (Nivre et al., 2006)), for extracting semantic features (WordNet (Fellbaum, 1998), GermaNet (Kunze and Lemnitzer, 2002)) and for classification (OpenNLP MaxEnt 2.5[2]). The architecture can easily be extended to incorporate additional components for preprocessing and feature extraction, or to implement new machine learning algorithms for training. At the moment the system provides working interfaces for English and German, but it can easily be extended to other languages.

The graphical user interface guides the user through the learning process and enables the user to systematically explore the benefit gained from different feature types for WSD. The toolkit supports manual annotation of selected instances and re-trains the system on the extended data set. MaJo also provides the means to evaluate the performance of the system against a gold standard.

The approach to AL with MaJo is as follows (Figure 1): First the system is trained on a small seed data set containing sentences with disambiguated instances of a specific target word. In the AL phase, users can either load unannotated sentences from a text file, enter new instances by hand, or generate new example sentences from the WWW. Users can specify a threshold for uncertainty sampling (Lewis and Gale, 1994): all sentences for which the confidence of the prediction by the classifier is below the threshold are presented to the user for annotation. The idea behind this sampling method is that the classifier's low confidence shows that it has yet to learn how to treat these examples, and by adding them to the training set we allow the classifier to just do that. After having been assigned the correct word sense, the new instances are added to the seed data and a new model is trained on the combined data set. The process can be repeated, adding more and more new instances to the training set. For a more detailed description of the MaJo toolkit see Rehbein et al. (2009).

[2] `http://maxent.sourceforge.net`

4. Experiments

4.1. Data

We selected 1,000 newspaper sentences containing the German verb *drohen* (threaten). The data was annotated manually by two expert annotators, following the annotation scheme of the German Salsa Corpus (Burchardt et al., 2006), a newspaper corpus annotated within the framework of frame semantics (Baker et al., 1998) . We selected *drohen* because it has only 3 different word senses which can be easily distinguished by human annotators (Table 2). The three word senses, or frames, are RUN_RISK-SALSA, COMMITMENT and the proto-frame DROHEN1-SALSA.[3] While there is a dominant frame which accounts for the majority of occurences of *drohen* (threaten) in our data set (RUN_RISK-SALSA captures around 50% of all word senses of *drohen* in our data), the distribution is not unduly skewed and still provides us with a large number of examples for the other two word senses of *drohen*.

frame	freq.	example
drohen1-salsa	243	*Die Mieten drohen zu steigen.* Rent increases are imminent.
Commitment	256	*Sie drohte ihm mit dem Messser.* She threatened him with the knife.
Run_risk	501	*Ihr drohen 3 Jahre Gefängnis.* She is facing 3 years in prison.
	1,000	training: 750, test: 250

Table 2: Word senses (frames) for *drohen* (threaten)

The three word senses (or frames) are defined as follows.

1. The RUN_RISK frame describes a situation where a *Protagonist* is exposed to a potentially dangerous situation that may end in a *Bad_outcome* for him- or herself. Please note that the original FrameNet RUN_RISK frame does not include the verb *threaten*. RUN_RISK-SALSA (hence called RUN_RISK) is an enhanced version of the FrameNet RUN_RISK frame with the three core frame elements *Action*, *Bad_outcome* and *Endangered_entity*. In German, *Endangered_entity* is usually filled by a dative NP.

   (1) **Ihr**$_{Endangered\_entity}$ drohen
   Her threathen
   3 Jahre Gefängnis$_{Bad\_outcome}$
   3 years prison.
   "She is facing 3 years in prison."

2. The COMMITMENT frame is defined as follows: "A *Speaker* makes a commitment to an *Addressee* to carry out some future action." The two core frame elements, addressee and speaker, are constrained to the semantic type *sentient*. The announced action includes desirable as well as less desirable events. Often these sentences include a *mit*-PP (with-PP) expressing the *message* or an instrument.

[3] Proto-frames are created to capture those meanings in German which are not yet covered by existing FrameNet frames.

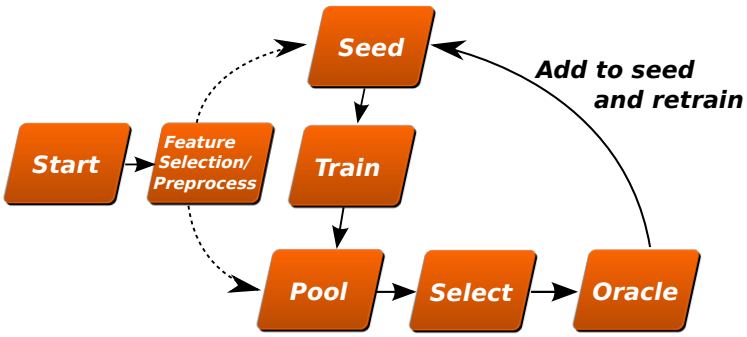

Figure 1: Active Learning loop with MaJo

	Feature Class	Description	Parameter	
(1)	WordRangeContext	bag-of-word context	window size	
(2)	POSTagContext	bag-of-POS-tag context	window size	Berk./Stan. POS Tagger
(3)	ClauseFunDep	words or POS tags for given functional dependencies	functional dependency	MaltParser
(4)	SentencePhrase FunDep	words or POS tags for children of a specific syntactic category	syntactic category	Berk./Stan. Parser
(5)	WordNet/GermaNet (Super)POSTag	WordNet relations for (super-ordinate) POS tags	max. depth, sem. relation	Berk./Stan. POS Tagger
(6)	WordNet/GermaNet FunDep	WordNet relations for specific functional dependencies	max. depth, sem. relation	MaltParser

Table 1: Off-the-shelf software components implemented in MaJo

(2) **Sie**$_{Speaker}$ droht **ihm**$_{Addressee}$ (PP mit dem Messer).
She threatens him (PP with the knife).
"She threatens him with the knife"

(3) **Sie**$_{Speaker}$ droht **ihm**$_{Addressee}$ (PP mit Vergeltung).
She threatens him (PP with revenge).
"She threatens to take revenge on him"

3. The third frame, DROHEN1-SALSA, shows strong semantic similarity to RUN_RISK-SALSA while being syntactically distinct. Is is realised by an infinite verb form with *zu* (to) and comprises the two core frame elements *Bad_outcome* and *Cause*.

(4) **Die Krise**$_{Cause}$ droht **Arbeitsplätze zu vernichten**$_{Bad\_outcome}$.
The crisis threatens jobs to eliminate.
"The crisis threatens to eliminate jobs."

Human annotators do not have problems to distinguish these frames. We computed Inter-Annotator Agreement on 100 sentences annotated by both annotators. The two annotators agreed on 99% of the data. This means that our data set does not include annotation noise resulting from controversial cases which could mislead the classifier.
In the experiment we want to investigate the following questions: (1) How many training instances do we need to get a reasonable performance for an automatic WSD system? (2) For an easy verb like *drohen*, can we get results comparable to human judgments when having access to a large amount of training data?

4.2. Feature Tuning

We used MaJo for tuning the features for the verb *drohen* and afterwards extracted the features from the training set (750 sentences). We kept the remaining 250 sentences for evaluation, running our experiments in a 5-fold cross validation setting. Our tuned feature set includes the following feature classes and settings:

1. bag-of-word context (5 words to the left/right)
2. bag-of-POS-tag context (3 POS tags to the left/right)
3. word forms for token assigned one of the following functional dependencies: main predicate (ROOT), subject (SUBJ), accusative object (OBJA), dative object (OBJD), genitive object (OBJG), prepositional object (OBJP), prepositional modification (PP), discontinuous morphemes (PART), prenominal attribute (ATTR), auxiliary phrases (AUX) and split verb prefixes (AVZ)

4. POS tags for all children of VZ phrases (VPs with zu-marked infinitives).

To our surprise we found that the features based on GermaNet did not perform well (Table 3). Best performance was obtained by lexical context features (bag-of-word context (0.705) and word forms for token with selected grammatical functions (0.700)). Somewhat lower were results for POS context (0.635), while features based on GermaNet hypernyms extracted for all nouns, adjectives and verbs only achieved an accuracy of 0.575, and the feature set based on GermaNet hypernyms for arguments did not perform well at all (0.475).

There are two possible explanations. First, the grammatical functions used for identifying the candidates for the GermaNetMalt feature plugin have been assigned by the statistical parser and thus are error-prone. Results for the Malt parser in the PaGe shared task on parsing German (Kübler, 2008) report an f-score of 90.2% for subjects and 80.0% for accusative objects, while for dative objects the MaltParser achieves 49.7% f-score only. This explains the better performance for the GermaNet features based on superordinate POS tags (Table 3). A second reason for the poor performance of the semantic GermaNet based features is founded in the more syntactically motivated frame distinctions. The DROHEN1-SALSA frame can be identified by looking for an infinite verb with *zu* (to), while RUN_RISK-SALSA usually has an indirect object. As we mentioned above, the automatic identification of dative NPs in German does not work very well. However, the protagonist in the COMMITMENT frame is often realised by a personal pronoun. While case syncretism certainly is a problem for German, at least the masculine personal pronoun *ihm* (him) is unambiguous. Unfortunately, personal pronouns are not useful for extracting informative features from GermaNet.

feature plugin	**settings**	**accuracy**
WordRangeContext	5 words left/right	0.705
ClauseFunDep	ROOT, SUBJ, OBJA, OBJP, PART, ATTR, AUX, AVZ	0.700
POSTagContext	3 words left/right, ignore punctuation	0.635
GermaNetSuperPOS	Noun, Adj., Verb, hyperonymy, depth=4	0.575
GermaNetMalt	SUBJ, OBJA, OBJD, OBJG hyperonymy, depth=3	0.475

Table 3: Accuracy for individual feature types (all results on fold 5)

Note that our tuned feature set outperforms the performance of Shalmaneser, a state-of-the-art WSD system (Erk and Padó, 2006) trained on the same data set.

We divided the 1,000 instances into a training set (750 instances) and a test set (250 instances). The initial seed data set contained 9 sentences (3 instances for each word sense) taken from the SALSA corpus.

4.3. Results

As a baseline we randomly selected n new instances from the pool of training data (*random sampling*). For *uncertainty sampling* we used the confidence score of a maximum entropy classifier.[4]

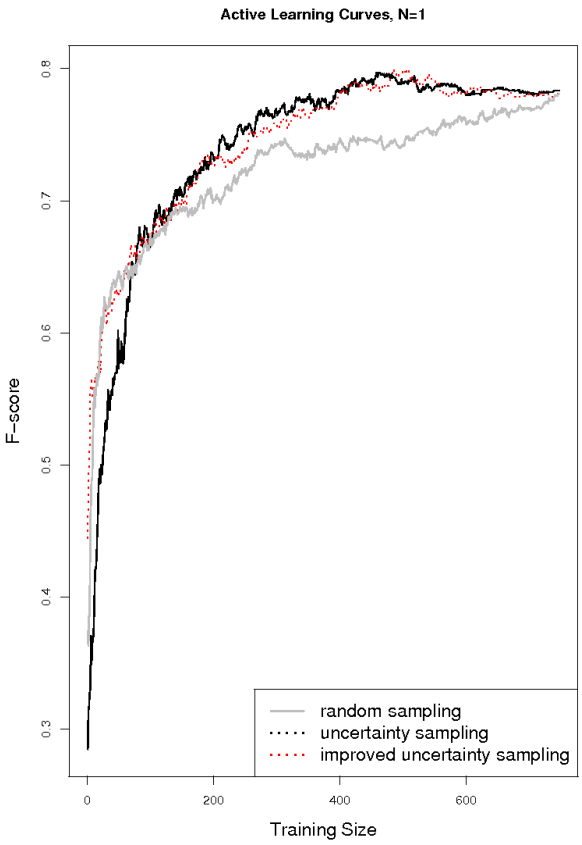

Figure 2: Learning curves for random sampling, uncertainty sampling and improved uncertainty sampling (5-fold cross validation)

Figure 2 shows learning curves for our experiments. We experimented with a varying number of training instances (1, 2, 3, 5, 10) added in each iteration to the training set (Table 4). Results for our WSD system are far below the results of our human annotators. Comparing the curve for AL with uncertainty sampling to the learning curve for random sampling, we see that in the beginning the randomly selected training instances yield better results. After adding approximately 75 new training instances, however, the AL approach outperforms random sampling.

4.4. Improved Uncertainty Sampling

The results suggest that early in the learning process when our classifier is trained on a very small seed data set, it is not beneficial to add the instances with the lowest classifier confidence. Instead, we propose a dynamic version of uncertainty sampling, taking into account how much the classifier has learned so far. When trained on a small seed set only, many of the predictions made by the classifier are made with low confidence. During training, the minimum confidence score for predicting word senses for unseen text

[4] `http://maxent.sourceforge.net`

inst./iteration	RAND	UNC	IMP_UNC
N=1	78.1 (747)	79.7 (464)	80.0 (508)
N=2	78.2 (746)	79.1 (536)	79.0 (500)
N=3	78.6 (744)	79.5 (465)	79.0 (504)
N=5	78.3 (705)	78.9 (565)	79.3 (530)
N=10	78.1 (670)	78.8 (500)	79.4 (490)

Table 4: Best F-score and number of training instances added to the seed data for each AL setting (5-fold cross validation)

increases, and the gap between minimum and maximum confidence scores closes. We make use of this and, instead of adding the n instances with the lowest confidence score, we compute a threshold as follows:

$$threshold = max_{Confidence} - min_{Confidence} \quad (1)$$

Then we sort all instances in the pool according to the confidence score for the classifier prediction and add the next n instances with a confidence score below the threshold. As training proceeds, the threshold drops. If there are no instances with a confidence value below the threshold, we select the n instances with the lowest confidence predicted by the classifier. Figure 2 shows the learning curves for our baseline (*random sampling*, RAN), for *uncertainty sampling* (UNC) and for our improved method (IMP_UNC). The improved uncertainty sampling shows comparable performance to UNC when trained on more data, while in the beginning the learning curve is steeper.

5. Discussion

We achieved our best result of 80% F-score for automatically disambiguating between the three senses of *drohen* using the improved uncertainty sampling (N=1, 508 instances added). Trained on the same number of instances, the basic uncertainty setting yields 78.8% F-score, while for random sampling on an equally sized data set we achieve 74.9% F-score. When training on all 750 instances, results get worse (78.1%). This shows that some sentences in the training set are not informative enough to improve the classifier, or even add noise to the data and so harm the learning process. There are only marginal differences between the best results for UNC and IMP_UNC (Figure 2). While the careful selection of new training instances (UNC, IMP_UNC) does improve results for all settings (N=1,2,3,5,10), performance remains nearly 20% lower than that of human annotators. This shows that simply adding more instances to the training set does not solve the knowledge acquisition bottleneck.
The results outlined above raise the following questions:

1. What is the reason for the great gap in performance between human annotators and our automatic WSD system?
2. Which are the instances the system gets wrong?
3. What is the impact of the different sampling methods on the learning process?

In Section 5.1. we give a qualitative error analysis, outlining reasons for the difference in accuracy between human annotation and the WSD system. In Section 5.2. we compare the different sampling methods and their impact on the training sets at different stages of the training process.

5.1. Error Analysis

How can we explain that the task of word sense disambiguation for *drohen*, being that easy for human annotators, cannot be solved sufficiently by the machine learning classifier?
To answer this question, we looked at the errors made by the classifier (Table 5). Results for our Active Learning approaches after 500 iterations are considerably higher than for random sampling (78.9 (UNC) and 79.6 (IMP) versus 74.2 (RAND)).

	f1	f2	f3	f4	f5	avg.	sd
RAND	76.0	74.0	71.2	76.8	72.8	**74.2**	2.3
UNC	79.2	77.2	79.6	79.6	78.8	**78.9**	1.0
IMP	79.2	78.0	78.8	80.4	81.6	**79.6**	1.4

Table 5: F-scores for different sampling methods and for individual folds after 500 iterations

The most common mistake made by the classifier is to assign RUN_RISK instead of the COMMITMENT frame. This is one of the typical problems for Machine Learning and reflects the class imbalance problem (Japkowicz and Stephen, 2002; Zhu and Hovy, 2007), where a highly skewed distribution of classes in the data causes the classifier to overuse the dominant class. In fact, the frames assigned by the classifier show a strong bias towards RUN_RISK, which is the most frequent frame in our data set. The classifier trained on the random selection of training data has the strongest bias and assignes the RUN_RISK frame to 62.5% of the instances in the test set. For basic and improved uncertainty sampling, the bias is less strong, but they still overgeneralise and assign the dominant frame to around 58% of all instances in the test set (Table 6), while the "real" distribution in the pool data shows a frequency of around 50% (Table 8) for the RUN_RISK frame. The ranking of different error types is the same for all three sampling methods (Table 7).

Frame	RAND	UNC	IMP
Commitment	17.6	20.8	21.0
Run_risk	19.9	21.3	21.0
drohen1-salsa	62.5	57.9	58.0

Table 6: Avg. frame distribution in classifier output

Aside from the class imbalance problem, which can be somewhat alleviated by controlling the distribution of classes in the training set (as done by the AL sampling methods), what else is responsible for the great gap in performance between human annotators and the WSD toolkit? For illustration let us look at some examples. Examples 5a and 5b show two very similar sentences, however, in the first one the predicate *drohen* should be classified as RUN_RISK while the second instance belongs to the COMMITMENT frame. In example 5a we have the dative NP

correct	predicted	RAND	UNC	IMP
drohen1-salsa	Commitment	3.7	3.8	2.0
Run_risk	drohen1-salsa	6.5	8.0	7.1
Commitment	drohen1-salsa	8.7	9.8	9.0
Run_risk	Commitment	9.6	12.1	13.3
drohen1-salsa	Run_risk	27.5	27.3	29.0
Commitment	Run_risk	44.0	39.0	39.6

Table 7: Percentage of error types averaged over all five folds

Erdogan-Partei which is threatened by a ban. In example 5b the NP *Baath-Partei* is in the nominative case and so the agent of the threat event. Without lexical context, it is nearly impossible to decide if the noun *Partei* (political party) is the subject or the object of the predicate *drohen* (threaten). For disambiguation we need to know the grammatical function of each of the NPs.

(5) a. Erdogan-Partei$_{DAT}$ droht Verbot$_{NOM}$
Erdogan party is threatened by ban
"RUN_RISK frame"

b. Baath-Partei$_{NOM}$ droht USA$_{DAT}$
Baath party threatens USA
"COMMITMENT frame"

For a syntactic parser, however, it is hard to identify the correct grammatical function, as German has a semi-free word order which allows the subject to be in sentence-initial position as well as occuring after the verb. Case information can help, but case syncretism in German not always allows for an unambiguous interpretation. Most statistical parsers, in fact, would incorrectly analyse the first NP as the subject, as this is the unmarked position for the subject argument. This would mislead the classifier to treat example 5a like 5b and assign it the COMMITMENT frame.

While for examples 5a and 5b syntactic information could support the identification of the correct word sense, in the following examples 6a and 6b syntactic knowledge does not help at all, as both NPs are in fact subject NPs. What we need for disambiguation is semantic knowledge which tells us that *home secretary* has the semantic type *sentient* and can fill the core frame element *Speaker* of the COMMITMENT frame, while *coup* is an event or action which might be the *Cause* in the RUN_RISK frame.

(6) a. Innenminister$_{NOM}$ droht.
home secretary threatens
"COMMITMENT frame"

b. Staatsstreich$_{NOM}$ droht.
coup threatens
"RUN_RISK frame"

Sometimes even semantic knowledge is not sufficient to disambiguate between different frames or word senses. Consider example 7a: without context it is not possible to know if *Asylbewerber* (asylum seeker) is in the nominative case or in the dative case. The semantic type of the second argument is ambiguous, too, as *Heim* (institution) can stand for the actual building, the institution, or metonymically for the inhabitants of the institution. The context information in 7b and 7c gives humans necessary clues how to interpret the sentence, an automatic analysis, however, is not straightforward.

(7) a. Asylbewerber$_{???}$ droht Heim$_{???}$
asylum seeker threatens institution

b. Asylbewerber$_{DAT}$ droht Heim$_{NOM}$ oder Ausweisung
asylum seeker threatens institution or eviction
"Asylum seeker is threatened to be institutionalised or expelled."
RUN_RISK frame

c. Asylbewerber$_{NOM}$ droht Heim$_{DAT}$ mit Feuer
asylum seeker threatens institution with fire
"Asylum seeker threatens institution with fire."
COMMITMENT frame

These examples outlined some of the problems for automatic WSD, explaining why the performance of automatic WSD systems stand back so far behind human annotation. Our experiments showed that even a large number of training instances is not sufficient to learn automatic frame distinctions. While syntactic features can support WSD to a great extent (Chen and Palmer, 2005), for fine-grained sense distinctions as the ones in FrameNet and its German counterpart SALSA, we also need to have access to semantic information and world knowledge.

5.2. Differences between the three sampling methods

Another important question concerns the impact of the three sampling methods on the selection of new data. Figure 2 shows a similar learning curve for random sampling and for the improved uncertainty sampling during the initial phase of learning, which both yield better performance than basic uncertainty sampling. However, after adding around 60 annotated instances to the seed data, all three sampling methods achieve comparable accuracy. From that point on random sampling is outperformed by both, uncertainty sampling as well as improved uncertainty sampling. This observation implies that the training sets for random sampling and the improved uncertainty sampling contain similar instances during early training, while in later stages of the learning process basic and improved uncertainty sampling seem to produce comparable training sets. To test this assumption we compared the average distribution of frames in the five folds for each sampling method after 30, 60, and 500 iterations.

Table 8 shows that the training set created by random sampling best reflects the distribution of frames in the pool. The more training iterations, the more similar the distribution of frames in the RAND training set and in the pool. This leads to an improved performance during the initial training iterations (as the test set is taken from the same population and thus shows the same distribution of frames).

Uncertainty sampling results in a more balanced training set, concentrating on those instances which are hard to learn

frame	RAND	UNC	INC_UNC	POOL
30 iterations				
drohen1-salsa	29.6	24.8	34.5	24.4
Commitment	15.2	34.5	20.0	24.9
Run_risk	55.2	40.7	45.5	50.7
60 iterations				
drohen1-salsa	26.2	30.0	29.3	24.4
Commitment	22.6	31.0	27.7	24.9
Run_risk	51.2	39.0	43.0	50.7
500 iterations				
drohen1-salsa	24.1	27.4	27.4	24.4
Commitment	24.4	29.2	29.6	24.9
Run_risk	51.5	43.4	43.0	50.7

Table 8: Avg. distribution of frames (%) in training sets for different sampling methods after 30, 60 and 500 iterations of training

for the classifier. As a result, the benefit we get from Active Learning becomes noticable only after adding a sufficient number of new instances. In our setting this is after around 60 iterations.

Improved uncertainty sampling is a compromise between random sampling and basic uncertainty sampling. It still reflects the frame distribution in the pool data, but also takes the actual learning process into consideration. Therefore it overcomes the weakness of basic uncertainty sampling during early training. Unfortunately, selecting better instances during the first iterations does not seem to improve results in the later training phases. After 500 training iterations random sampling perfectly mirrors the distribution in the pool data, while the proportion of frames in the data sets created by basic and improved uncertainty sampling is nearly identical, explaining the similar performance of both methods after the initial phases of training. There is a lingering suspicion that the size of our pool is not yet big enough to allow the difference between basic uncertainty sampling and the improved uncertainty sampling to become visible. This, however, can only be tested on a larger data set and therefore must await further investigation.

Another important difference between the training sets generated by the three sampling methods concerns the variance in the different folds of our training sets. The Active Learning approaches create more homogeneous data sets where results obtained on the individual folds show a much lower standard deviation than the ones for the different folds for random sampling (Table 5). This observation is in line with (Ertekin et al., 2007), who showed that SVM based Active Learning, where the most informative instances are considered to be the ones closest to the hyperplane, can provide the learner with more balanced data sets. As a result the AL approaches produce more reliable data sets for training, yielding more consistent results.

6. Conclusions and Future Work

We presented an environment for Active Learning in a WSD task, providing a user friendly GUI which supports processing steps like feature selection, sampling methods for selecting new training instances from the pool, and an interface for manually adding correct labels to the selected training instances. At present, MaJo implements a basic environment for AL, using uncertainty sampling and a maximum entropy classifier. We plan to integrate other ML algorithms as well as more sophisticated sampling methods. We also intend to expand the feature set used for WSD.

In the paper we questioned the assumption that the knowledge acquisition bottleneck, which has been described as the major impediment to solving the WSD problem (Gonzalo and Verdejo, 2006), can be solved by simply obtaining more and more data. While the size of the training set certainly has a crucial impact on the performance of WSD systems, we showed that the claim "There's no data like more data" is not entirely true.

Our case study on the German verb *drohen* (threaten) showed that an increase in size for the training set does not always correspond to an increase in performance. Less, but carefully selected examples can not only yield comparable performance and thus reduce costs for human annotation, they can even outperform classifier performance on a randomly selected data set, as Active Learning improves the quality of the data itself, resulting in more reliable and consistent training sets.

7. Acknowledgements

We would like to thank Marcel Köster and Jonas Sunde who implemented the MaJo toolkit. This work has partly been funded by the German Research Foundation DFG (grant PI 154/9-3).

8. References

C.F. Baker, C.J. Fillmore, and J.B. Lowe. 1998. The berkeley framenet project. In *Proceedings of the 17th international conference on Computational linguistics*, Morristown, NJ, USA. Association for Computational Linguistics.

M. Bloodgood and V. Shanker. 2009. A method for stopping active learning based on stabilizing predictions and the need for user-adjustable stopping. In *Proceedings of CoNLL-2009*, Boulder, Colorado.

A. Burchardt, K. Erk, A. Frank, A. Kowalski, S. Padó, and M. Pinkal. 2006. The salsa corpus: a german corpus resource for lexical semantics. In *Proceedings of LREC-2006*, Genoa, Italy.

Y.S. Chan and H.T. Ng. 2007. Domain adaptation with active learning for word sense disambiguation. In *Proceedings of ACL-2007*.

J. Chen and M. Palmer. 2005. Towards robust high performance word sense disambiguation of english verbs using rich linguistic features. In *IJCNLP*.

J. Chen, A. Schein, L. Ungar, and M. Palmer. 2006. An empirical study of the behavior of active learning for word sense disambiguation. In *Proceedings of NAACL-2006*, New York, NY.

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

H.T. Dang. 2004. *Investigations into the role of lexical semantics in word sense disambiguation*. PhD dissertation, University of Pennsylvania, Pennsylvania, PA.

K. Erk and S. Padó. 2006. Shalmaneser - a flexible toolbox for semantic role assignment. In *Proceedings of LREC-2006*, Genoa, Italy.

Şeyda Ertekin, Jian Huang, L'eon Bottou, and Lee Giles. 2007. Learning on the border: active learning in imbalanced data classification. In *CIKM '07: Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 127–136, New York, NY, USA. ACM.

Christiane Fellbaum, editor. 1998. *WordNet: An Electronic Lexical Database (ISBN: 0-262-06197-X)*. MIT Press, first edition.

Julio Gonzalo and Felisa Verdejo, 2006. *Automatic Acquisition of Lexical Information and Examples*, chapter 9. Kluwer.

Nathalie Japkowicz and Shaju Stephen. 2002. The class imbalance problem: A systematic study. *Intelligent Data Analysis*, 6(5):429–449.

Dan Klein and Chris Manning. 2003. Accurate unlexicalized parsing. In *41st Annual Meeting of the Association for Computational Linguistics (ACL-03)*, Sapporo, Japan.

Sandra Kübler. 2008. The page 2008 shared task on parsing german. In *ACL Workshop on Parsing German (PaGe-08)*, Columbus, OH.

Claudia Kunze and Lothar Lemnitzer. 2002. Germanet - representation, visualization, application. In *Proceedings of the 3rd International Conference on Language Resources and Evaluation (LREC-02)*.

D.D. Lewis and W.A. Gale. 1994. A sequential algorithm for training text classifiers. In *Proceedings of ACM-SIGIR*, Dublin, Ireland.

Joakim Nivre, Johan Hall, and Jens Nilsson. 2006. Maltparser: A data-driven parser-generator for dependency parsing. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC-06)*, Genoa, Italy.

Slav Petrov and Dan Klein. 2007. Improved inference for unlexicalized parsing. In *Proceedings of the Human Language Technology Conference and the 7 th Annual Meeting of the North American Chapter of the Association for Computati onal Linguistics (HLT-NAACL-07)*, Rochester, NY.

Ines Rehbein, Josef Ruppenhofer, and Jonas Sunde. 2009. Majo - a toolkit for supervised word sense disambiguation and active learning. In *Proceedings of the 8th Workshop on Treebanks and Linguistic Theories (TLT-8)*, Milano, Italy.

Kristina Toutanova, Dan Klein, Christopher D. Manning, and Yoram Singer. 2003. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology (NAACL-03)*, Morristown, NJ, USA. Association for Computational Linguistics.

A. Vlachos. 2008. A stopping criterion for active learning. *Compututer Speech and Language*, 22(3).

D. Yarowsky. 1995. Unsupervised word sense disambiguation rivaling supervised methods. In *Proceedings of ACL-1995*, Cambridge, MA.

J. Zhu and E. Hovy. 2007. Active learning for word sense disambiguation with methods for addressing the class imbalance problem. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, Prague, Czech Republic.