# A Comparative Study of Uncertainty Based Active Learning Strategies for General Purpose Twitter Sentiment Analysis with Deep Neural Networks

Nils Haldenwang(✉), Katrin Ihler, Julian Kniephoff, and Oliver Vornberger

Institute of Computer Science, Media Computer Science Group,
University of Osnabrück, Osnabrück, Germany
{nils.haldenwang,kihler,jkniepho,oliver}@uos.de

**Abstract.** Active learning is a common approach when it comes to classification problems where a lot of unlabeled samples are available but the cost of manually annotating samples is high. This paper describes a study of the feasibility of uncertainty based active learning for general purpose Twitter sentiment analysis with deep neural networks. Results indicate that the approach based on active learning is able to achieve similar results to very large corpora of randomly selected samples. The method outperforms randomly selected training data when the amount of training data used for both approaches is of equal size.

## 1 Introduction

General purpose Twitter sentiment analysis was introduced as a new sentiment classification task by Haldenwang and Vornberger (2015). The main difference to other popular Twitter sentiment analysis tasks – such as SemEval, Nakov et al. (2016) – lies in the omission of filtering the Twitter stream with regard to certain topics or types of messages. Hence, the data set consists of a representative sample of the public Twitter stream, which is relevant for applications such as monitoring the sentiment of individuals, regions or the general, unfiltered public Twitter stream.

Systems based on deep neural networks are prevalent in the related Twitter sentiment analysis tasks (Deriu et al. 2016, Rouvier and Favre 2016, Xu et al. 2016). Therefore, it seems reasonable to investigate their feasibility for general purpose Twitter sentiment analysis.

Acquiring a sufficient amount of manually annotated data for the training of deep neural networks to perform the aforementioned task is very labor intensive. One possibility to deal with low amounts of manually annotated data is the use of *distant supervision* approaches based upon emoticons as originally introduced by Pak and Paroubek (2010). Distant supervision has already successfully been used in the training process of various deep learning architectures for Twitter sentiment analysis (Severyn and Moschitti 2015, Deriu et al. 2016, Xu et al. 2016).


G. Rehm and T. Declerck (Eds.): GSCL 2017, LNAI 10713, pp. 208–215, 2018.
https://doi.org/10.1007/978-3-319-73706-5_18

While noisy labels based on emoticons provide a good starting point for the training of a deep learning system, it is probably beneficial to use manually annotated training data for the specific task to achieve satisfying results.

A common approach to reduce the manual effort is *active learning*. Settles (2010) summarizes the idea of active learning as follow: "[...] a machine learning algorithm can achieve greater accuracy with fewer training labels if it is allowed to choose the data from which it learns [...]". Given a large corpus of unlabeled data points, the learner may choose the samples from which it hopes to gain the most insights from. The labels of the chosen data points are queried from an *oracle*, in this case a human annotator. The remainder of this paper describes a study the authors conducted to assess the feasibility of various metrics for measuring the potential information gain for unlabeled samples and then choosing the samples that are to be annotated.

## 2 Experimental Setup

In this section we first introduce the initial deep neural network that is the starting point for all experiments and illustrate how it was parametrized. Secondly, the active learning strategies which are evaluated are described. Finally, the experimental procedure is presented.

### 2.1 Initial Deep Neural Network

The classifier used in these experiments is a convolutional neural network. Its basic architecture is described in Zhang and Wallace (2015). First, the tokenized tweet is transformed into a list of dense *word embeddings*. The resulting *sentence matrix* is then convolved with a certain set of *filters* of potentially varying *region sizes*. After that, the resulting *feature maps*, which are vectors describing certain "higher order features" of the tweet, activate a 1-max-pooling layer via a possibly non-linear activation function. Lastly, this pooling layer is densely connected to the output layer using softmax activation and optional dropout regularization. In contrast to Zhang and Wallace (2015), our output layer has three neurons, reflecting the fact that we want to differentiate the three classes *positive*, *negative* and *uncertain*.[1]

All weights of the network were initialized randomly except for the embedding layer, where we used *word2vec* vectors (cf. Mikolov et al. (2013)) of dimension $d = 100$ trained on a dataset of approximately 33 million tweets collected between June 2012 and August 2013 by Neubauer (2014). After some minimal preprocessing[2], this dataset contained 624,015 unique tokens, of which we used the 200,000 most frequent ones in the network. The parameters were chosen as follows: The model used was the skip-gram model, the window size was 5 words, the subsampling threshold was $t = 10^{-5}$; negative sampling was used with $k = 5$

[1] See Haldenwang and Vornberger (2015) for further details.

[2] replacing @-mentions and URLs by generic tokens and removing "non-words".

"noise words" and we ran two iterations of the algorithm. Most of these values were recommended by Mikolov et al. (2013), where one can also find explanations for the parameters. The rest of the network's hyperparameters was found using a search guided by the best practices laid out in Zhang and Wallace (2015): We first evaluated networks with only one region size $r \in \{2, 3, 4, 5, 6, 8, 10\}$ and $n \in \{50, 325, 600\}$ filters. The activation function $f$ between the convolution and pooling layers was chosen from the set {id, tanh, RelU[3]} and the dropout rate (Srivastava et al. 2014) was $p \in \{0, 0.25, 0.5\}$.

We evaluated all of these combinations based on their average *macro-$F_1$-score* in a tenfold cross-validation using the dataset from Haldenwang and Vornberger (2015). First, each network was trained using a *distant supervision* procedure with noisy labels based on emoticons in the dataset of Neubauer (2014). Note, that the distant super vision approach only consists of positive and negative tweets, since there is no reliable noisy label for uncertain tweets. Next, the network's parameters were further refined by using the positive and negative tweets from the datasets of the SemEval competitions (Nakov et al. 2013, Rosenthal et al. 2014, 2015) for training.[4] The networks were trained using the Adagrad (Duchi et al. 2011) algorithm. Both datasets were presented once (one *epoch*) in a batch size of 50 tweets.

The best configuration turned out to be $r = 2$, $n = 50$, $f = \tanh$ and $p = 0.25$ with an average $F$-score of $F_1 \approx 0.56$. We also tried adding bigger filters to this configuration in multiple ways, but none of the resulting configurations could significantly surpass the above, so we do not go into further details of this process here. For the following experiments with regard to active learning, we used the version of this network that was only trained on the noisy labels, to properly reflect one of the constraints of this approach: not to have a big supply of manually labeled tweets in advance.

### 2.2 Investigated Active Learning Strategies

As a strategy to query the best suited tweets to label for the network, we decided to investigate *uncertainty sampling*, a strategy originally devised by Lewis and Gale (1994) which is both easy to implement and understand and thus commonly used. With this strategy, each tweet is assigned an uncertainty value which defines how uncertain the network is in finding the correct label for the tweet. The most uncertain tweets are then chosen to be labeled.

For a problem with three (or more) classes such as ours, there are different metrics available to calculate uncertainty. These metrics differ in how many of the class probabilities they take into account. In the following a short description for each of the metrics provided. A more thorough introduction and comparison can be found in the literature survey of Settles (2010).

[3] Mahendran and Vedaldi (2015).

[4] The neutral class does not match with the desired uncertain class and hence is ommitted here.

The *confidence* metric can be used to choose the tweet $x^*_{LC}$ whose label the network is *least confident* about:

$$x^*_{LC} = \underset{x}{\operatorname{argmin}}\, P_\theta(\hat{y}|x)$$

The confidence is defined as the probability that the class label $\hat{y}$ chosen by the network $\theta$ is correct as considered by the network itself (and as such is the highest of the three probabilities for the three class labels).

The *margin* metric also takes the second highest probability into account by calculating the difference between the probabilities of the two class labels $\hat{y_1}$ and $\hat{y_2}$ the network believes to be most likely correct:

$$x^*_M = \underset{x}{\operatorname{argmin}}\, P_\theta(\hat{y_1}|x) - P_\theta(\hat{y_2}|x)$$

A tweet with a smaller margin would be considered more uncertain since the network has difficulties choosing between the labels $\hat{y_1}$ and $\hat{y_2}$.

Finally, the *entropy* metric considers the probability for all class labels $\hat{y_i}$ to calculate the amount of informativity each tweet has to offer to the network:

$$x^*_H = \underset{x}{\operatorname{argmax}} - \sum_i P_\theta(\hat{y_i}|x) \log P_\theta(\hat{y_i}|x)$$

In our experiment we compare the effect of these metrics to find out which is most helpful for our use case.

To speed up the labeling process, we query and label the tweets in batches of 20. However, since the uncertainty values are not recalculated after picking a tweet for a batch, this could lead to the tweets in the batch being very similar to one another since they all occupy the same uncertain region of the feature space. To avoid this, we introduce diversity as a second criterion to our querying process as described in (Patra and Bruzzone 2012):

First, we choose the 60 most uncertain tweets which we then reduce to 20 both uncertain and diverse tweets by clustering them with kernel $k$-means into 20 clusters and picking the most uncertain tweet from each cluster.

### 2.3 Experimental Procedure and Data Usage

For each of the uncertainty metrics described above, the experiment is initialized with a copy of the initial deep neural network that was pretrained with the aforementioned distantly supervised data only. The corpus of unlabled tweets to chose from consisted of 100,000 tweets that were randomly sampled from the 33 million dataset of Neubauer (2014). First, all tweets in the unlabeled corpus are classified by the network and then 20 tweets are chosen to be annotated using the previously mentioned strategy. Next, after the 20 tweets are labeled by the human annotator, 10 training iterations are performed with the newly annotated

tweets. This procedure is then repeated until 1,000 tweets are annotated for each uncertainty metric.

Additionally, we generated a random baseline by training a copy of the initial neural network with randomly selected, manually annotated tweets in batches of 20 with 10 training iterations.

Each generated network was then evaluated using the reliable general purpose Twitter sentiment analysis data set from Haldenwang and Vornberger (2015) as a test set. The resulting macro $F_1$-score is reported.

## 3 Results

Figure 1 shows a visualization of the experimental results. A notable observation is the effectiveness of just labeling 100 tweets, the classification performance almost doubles for all metrics. This drastic increase in performance is a strong indication that even small amounts of manually annotated data are very beneficial in addition to the noisy labeled training data. Note, that the initial score is rather low, because the network was just pretrained with positive and negative data and, hence, missclassified all uncertain samples. When measuring the score for just the positive and negative classes after pretraining, it was $F_1 \approx 0.637$. Hence, pretraining with the distantly supervised data provides a useful basis for the network's parameters.

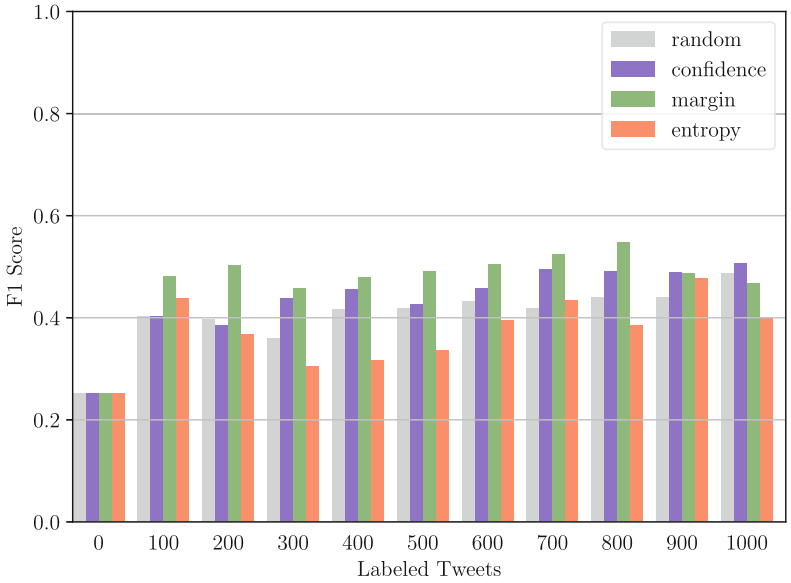


**Fig. 1.** Experimental results showing the macro $F_1$-score of the investigated metrics in steps of 100 manually annotated tweets.

The random baseline yields solid results but seems to always be outperformed by either the confidence or margin metric. The entropy metric performs worse than random in almost all cases. Moreover, it seems to be the most unstable with the strongest fluctuations in performance.

While the margin metric takes the lead for the first 800 annotated tweets, its effectiveness drastically drops at 900 and 1,000. Below 800 the confidence metric performed consistently worse than the margin metric but does not seem to suffer as severe a performance drop and at 1,000 labeled tweets takes the lead.

Overall, the best performance achieved was $F_1 \approx 0.55$ by the margin metric at 800 manually annotated tweets. The differences in classification behaviour when compared to the other metrics and the random baseline were significant. Moreover, the result is on par with training the same initial network with about 25,000 manually annotated tweets from a related domain (SemEval) and about 8,000 manually annotated tweets for the problem at hand (Haldenwang and Vornberger 2015), as was presented in Sect. 2.1, while only using a fraction of the training data.

## 4 Conclusions and Outlook

The results indicate that two out of three investigated uncertainty based active learning strategies consistently seem to surpass random sample selection for the investigated task.

Overall, the performance of the investigated strategies seems to be fluctuating a lot. After a certain point (more than 800 labeled tweets) the performance of all three active learning strategies seems to deteriorate or converge with the random baseline. In future work the study has to be extended to verify the aforementioned trend.

Moreover, a problem that can occur with purely uncertainty based metrics lies in their affinity to favor outliers since those are often of high uncertainty (Settles and Craven 2008). This selection of outliers may be what causes the deterioration at the last steps, since the outliers probably do not add any useful information for the correct classification of the non-outliers and may be harmful for the overall generalization of the system. In future work we plan on investigating active learning strategies which do not purely rely on the uncertainty but also take the *density weight* into account, as was suggested by Settles and Craven (2008). The basic idea is to not only select uncertain samples but also take into account the density of samples in the surrounding area to select data points which are representative for as many other uncertain samples as possible. Hopefully, this strategy can prevent pure outliers from being selected, increase the information gain and reduce the fluctuations.

Combining deep convolutional neural networks with active learning based on uncertainty sampling seems to be a promising approach for general purpose Twitter sentiment analysis which can drastically reduce the amount of manual annotation that is needed to achieve sufficient results.

## References

Deriu, J., Gonzenbach, M., Uzdilli, F., Lucchi, A., De Luca, V., Jaggi, M.: Swisscheese at semeval-2016 task 4: sentiment classification using an ensemble of convolutional neural networks with distant supervision. In: Proceedings of the 10th International Workshop on Semantic Evaluation, SemEval 2016, San Diego, California, pp. 1124–1128, June 2016. Association for Computational Linguistics. http://www.aclweb.org/anthology/S16-1173

Duchi, J., Hazan, E., Singer, Y.: Adaptive subgradient methods for online learning and stochastic optimization. J. Mach. Learn. Res. **12**, 2121–2159 (2011)

Haldenwang, N., Vornberger, O.: Sentiment uncertainty and spam in Twitter streams and its implications for general purpose realtime sentiment analysis. In: Proceedings of the International Conference of the German Society for Computational Linguistics and Language Technology, GSCL 2015, pp. 157–159 (2015)

Lewis, D.D., Gale, W.A.: A sequential algorithm for training text classifiers. In: Croft, B.W., van Rijsbergen, C.J. (eds.) SIGIR 1994, pp. 3–12. Springer, London (1994). https://doi.org/10.1007/978-1-4471-2099-5_1

Mahendran, A., Vedaldi, A.: Understanding deep image representations by inverting them. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5188–5196. IEEE (2015)

Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in Neural Information Processing Systems, pp. 3111–3119 (2013)

Nakov, P., Rosenthal, S., Ritter, A., Wilson, T.: SemEval-2013 task 2 : sentiment analysis in Twitter. In: Proceedings of the 7th International Workshop on Semantic Evaluation (SemEval 2013), in conjunction with the 2nd Joint Conference on Lexical and Computational Semantics, *SEM 2013, vol. 2, pp. 312–320 (2013)

Nakov, P., Ritter,A., Rosenthal, S., Sebastiani, F., Stoyanov, V.: Semeval-2016 task 4: sentiment analysis in Twitter. In: Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016), San Diego, California, pp. 1–18. Association for Computational Linguistics, June 2016. http://www.aclweb.org/anthology/S16-1001

Neubauer, N.: Semantik und Sentiment: Konzepte, Verfahren und Anwendungen von Text-Mining. Dissertation, Universität Osnabrück (2014). https://repositorium.uni-osnabrueck.de/handle/urn:nbn:de:gbv:700--2014060612524

Pak, A., Paroubek, P.: Twitter as a corpus for sentiment analysis and opinion mining. In: LREC (2010)

Patra, S., Bruzzone, L.: A cluster-assumption based batch mode active learning technique. Pattern Recogn. Lett. **33**(9), 1042–1048 (2012)

Rosenthal, S., Ritter, A., Nakov, P., Stoyanov, V.: Semeval-2014 task 9: sentiment analysis in Twitter. In: Proceedings of the 8th International Workshop on Semantic Evaluation, SemEval 2014, Dublin, Ireland, pp. 73–80. Association for Computational Linguistics and Dublin City University, August 2014

Rosenthal, S., Nakov, P., Kiritchenko, S., Mohammad, S., Ritter, A., Stoyanov, V.: Semeval-2015 task 10: sentiment analysis in Twitter. In: Proceedings of the 9th International Workshop on Semantic Evaluation, SemEval 2015, Denver, Colorado, pp. 451–463. Association for Computational Linguistics, June 2015

Rouvier, M., Favre, B.: Sensei-lif at semeval-2016 task 4: polarity embedding fusion for robust sentiment analysis. In: Proceedings of the 10th International Workshop on Semantic Evaluation, SemEval 2016, San Diego, California, pp. 202–208. Association for Computational Linguistics, June 2016. http://www.aclweb.org/anthology/S16-1030

Settles, B.: Active learning literature survey. University of Wisconsin, Madison (2010)
Settles, B., Craven, M.: An analysis of active learning strategies for sequence labeling tasks. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp. 1070–1079. Association for Computational Linguistics (2008)
Severyn, A., Moschitti, A.: Unitn: training deep convolutional neural network for Twitter sentiment classification. In: Proceedings of the 9th International Workshop on Semantic Evaluation, SemEval 2015, Denver, Colorado, pp. 464–469. Association for Computational Linguistics, June 2015
Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: a simple way to prevent neural networks from overfitting. J. Mach. Learn. Res. **15**(1), 1929–1958 (2014)
Xu, S., Liang, H., Baldwin, T.: Unimelb at semeval-2016 tasks 4a and 4b: an ensemble of neural networks and a word2vec based model for sentiment classification. In: Proceedings of the 10th International Workshop on Semantic Evaluation, SemEval 2016, San Diego, California, pp. 183–189. Association for Computational Linguistics, June 2016. http://www.aclweb.org/anthology/S16-1027
Zhang, Y., Wallace, B.: A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification. arXiv preprint arXiv:1510.03820 (2015)