

User-based active learning

Christin Seifert* and Michael Granitzer*[†]

*Knowledge Management Institute, University of Technology, Graz, Austria

[†]Know-Center GmbH, Graz, Austria

christin.seifert@tugraz.at, mgrani@know-center.at

Abstract—Active learning has been proven a reliable strategy to reduce manual efforts in training data labeling. Such strategies incorporate the user as oracle: the classifier selects the most appropriate example and the user provides the label. While this approach is tailored towards the classifier, more intelligent input from the user may be beneficial. For instance, given only one example at a time users are hardly able to determine whether this example is an outlier or not. In this paper we propose user-based visually-supported active learning strategies that allow the user to do both, selecting and labeling examples given a trained classifier. While labeling is straightforward, selection takes place using a interactive visualization of the classifier’s a-posteriori output probabilities. By simulating different user selection strategies we show, that user-based active learning outperforms uncertainty based sampling methods and yields a more robust approach on different data sets. The obtained results point towards the potential of combining active learning strategies with results from the field of information visualization.

Keywords—active learning, simulation, user behavior, visualization

I. INTRODUCTION

The success of supervised learning depends to a large degree on the quantity and quality of the provided training data set. Active learning has become popular to overcome the training data scarcity problem by increasing the efficiency of the labeling process itself. While there are different active learning strategies, as for example active learning strategies optimized towards particular hypothesis classes, the general approach to active learning is basically the same: The learner itself selects the most beneficial unlabeled example. After obtaining the correct label for this example from the user, the learner updates its classification hypothesis and starts the next round by again selecting the most beneficial example, i.e. the example the learner benefits most from. Generally speaking, humans become labeling machines on examples selected by the learner. While this approach sounds quite natural, users have to fully trust the active learning strategy which is not always the case [15]. Moreover, recent work shows that, depending on the application scenario, some active learning strategies require more labeled instances than passive learning or have been beaten by the random baseline ([10], [3], [12], [6]). An interesting finding has been made in [1], where it is shown that the efficiency of active learning correlates with the proficiency of the annotator.

In our work, we extend the active learning paradigm, in particular approaches based on pool-based uncertainty sampling, towards a user-based active learning approach by including information visualization techniques. In uncertainty based sampling, the learner selects the examples from the pool of unlabeled examples where the learner is least certain how to label. Our main motivation concerns a better utilization of human intelligence which goes beyond simple labeling and is more robust towards proficiencies of an annotator. Therefore, we develop an interactive visualization of the classifier’s results on the unlabeled data allowing the user to select examples and label them afterwards. Hence, we combine the classifiers suggestions with the human capability in rapidly selecting appropriate examples by visual means. By simulating picking strategies of users observed in preliminary experiments, we show that our visual active learning approach outperforms classical uncertainty sampling. Moreover, experimental results show that the quality of the learner can be devised by visual means. This indicates that there has to be a tighter, visual integration between learner and humans in the labeling process.

We contribute to the field of active learning by (i) developing new, visually inspired active learning strategies which outperform entropy and least confidence uncertainty sampling strategies on average, (ii) showing that our visualization technique allows to estimate negative effects of an uncertainty based active learning strategy, (iii) empirically confirming the findings in [10], [3], [12], [6], that entropy and least confidence uncertainty sampling can be outperformed by random sampling in various settings, and (iv) showing that through changing labeling strategies, our visual active learning strategies could yield further improvements and more robust results. Additionally, the proposed setup allows for formulating and evaluating additional user strategies that might outcome of subsequent user experiments. Overall we demonstrate, that utilizing humans to only label examples in active learning settings is suboptimal. Hence, active learning can benefit from including visualization techniques.

The paper is structured as follows: We introduce our user-based active learning approach in the next section. In the following experimental section, we compare the deduced active learning strategies to uncertainty based active learning strategies. The paper ends by summarizing our findings and pointing to future research directions.

II. USER-BASED, VISUALLY-SUPPORTED ACTIVE LEARNING APPROACH

Classical uncertainty sampling for active learning can be summarized as follows: Given a training set \mathbf{X}_l , the classification model is built and applied to unlabeled examples, denoted as \mathbf{X}_u . By analyzing the classification results on \mathbf{X}_u using a particular uncertainty sampling strategy, one example is selected to be labeled by the human. The labeled example is added to the training set \mathbf{X}_l and the process starts again. Thereby, the uncertainty sampling strategy should choose the example the current model is least confident in. Two different uncertainty sampling strategies have been regularly used in the past, namely entropy sampling and least confidence sampling. Let y denote the label of an example and x the example itself. Then, for least confidence sampling the example selected as the most informative is given by $x_c = \arg \min_x (\arg \max_y P(y|x))$. In case of maximum entropy sampling [13], [12], the example is selected as $x_e = \arg \max_x (-\sum_i P(y_i|x) \log(y_i|x))$. When comparing the user-based active learning models to the uncertainty sampling methods above we will focus on serial active learning, i.e., rebuilding and evaluating the classifier after each labeled instance, because simple uncertainty-based strategies have been shown to be outperformed by passive learning (random sampling) in batch-mode settings [6].

In user-based active learning the user itself is given the possibility to select the next sample to label, decide whether to label it and finally assign the label. To enable the user to find (select) appropriate examples we visualize the current classifier decisions for the unlabeled examples. Once an item is selected the user can then decide, whether to label it and eventually provide the correct label (by drag and drop). We identified four crucial properties for a visualization supporting effective user-based active learning: First, the visualization should allow to judge problematic behavior of classification models like for example biases towards particular classes. Second, fast and easy identification of false and/or problematic examples, e.g. outliers should be supported. Third, users should be able to rapidly select and label batches of examples, not only single examples. Fourth, the visualization should be classifier independent and usable on top of any multi-class, single-label classification task which provides a-posteriori probabilities. This requirement comes from practical reasons. As outlined in a recent survey on the use of active learning strategies in the NLP domain [15], practitioners highly appreciate classifier independent strategies.

In the next section we show our approach to such an interactive visualization supporting user-based active learning. We then describe the active learning strategies we derived from observing users interacting with the visualization. We then experimentally compare these strategies with classical uncertainty-based active learning. The various experiments

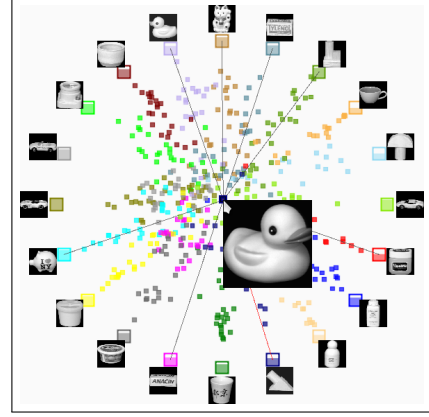


Figure 1: Example visualization on Coil-20 image date set

we present are not feasible with real users due to the time required. Therefore, we do not employ real-users but rather simulate their behavior observed in preliminary studies.

A. Visualizing Classifier Decisions

In order to fulfill the visualization properties outlined above, we used the visualization strategy for multi-class, single-label classification problems described in [11], which we briefly summarize in this section. Note that the batch selection is currently not implemented. As depicted in figure 1, the classifier visualization displays classes as uniquely colored squares equally distributed around the perimeter of a circle in random order. The filling level of the squares is proportional to the number of examples assigned to the specific class. The placement of unlabeled samples ($\in \mathbf{X}_u$) inside the circle depends on the a-posteriori probabilities of the classification model trained on \mathbf{X}_l . A test example is colored according to the class with the maximum a-posteriori probability. The following algorithm describes the layout of the visualization:

Require: classifier, set of classes C , \mathbf{X}_u

$\alpha \leftarrow 0$

$\Delta\alpha \leftarrow \frac{2\pi}{|C|}$

for all $c \in C$ **do**

$\vec{p}_c \leftarrow (\cos(\alpha), \sin(\alpha))$

$\alpha \leftarrow \alpha + \Delta\alpha$

end for

for all $x \in \mathbf{X}_u$ **do**

$\{(c, v)\} \leftarrow \text{classify}(x)$, v .. confidence for class c

$\vec{p}_x \leftarrow (0, 0)^T$

for all $(c, v) \in \{(c, v)\}$ **do**

$\vec{d}_c \leftarrow v \cdot \vec{p}_c$

$\vec{p}_x \leftarrow \vec{p}_x + \vec{d}_c$

end for

end for

return positions of examples (p_x) and classes (p_c) in the unit square

Intuitively, one can think of springs attached between examples and classes. Each spring attracts the example towards the position of the class in the plane with a force proportional to the a-posteriori probability $P(y|x)$ of assigning example x to class y . The final position of the example corresponds to the equilibrium state where all spring forces sum up to zero. For example, a perfectly classified test example which has the probability distribution $p = (p_1, \dots, p_c)$, $p_i = 1$ and $p_j = 0 \quad \forall j \neq i, 1 \leq i, j \leq c$ lies exactly on the location of the class item associated with class i (c denotes the number of classes). On the contrary, a test example with an uniform a-posteriori probability is placed in the center of the circle. Because the classifier visualization is a projection from c -dimensional space in the two-dimensional space, the visualization is ambiguous for $c > 2$. However, the goal of the visualization is not to present an unambiguous solution, but to enable the user to distinguish between examples that could be clearly classified versus examples with no confident prediction.

B. User-Based Active Learning Models

Given the previously introduced classification visualization, we analyzed how users utilize the visualization for labeling training examples in a preliminary study. We identified two primary strategies which inspired the development of two different active learning strategies. In order to simulate these strategies in our experiments, we derived mathematical models from the observed strategies. In the experiments we distinguish between selection strategies (deciding which example to select next) and labeling strategies (whether or not to actually label an example). We implemented two different selection strategies, named “gaussian” and “convhull” and two different labeling strategies, named “all” and “misses”.

1) *Simulated selection strategies:* In the first selection model, “gaussian”, we simulate user selecting examples from the center of the visualization. This selection model assumes that the user tries to select the most ambiguous samples, i.e. those samples classes are mostly competing for. These samples are placed in the center of the visualization. To simulate this behavior we select an example with probability according to a bivariate Gaussian distribution over the center of the visualization. This means, the probability $p(x, y)$ to select an example at position (x, y) is proportional to $p(x, y) \approx \mathcal{N}_2(\mu_1, \mu_2, \Sigma)$ with $\mu_1 = \mu_2 = (0.5s, 0.5s)$, $\Sigma = \text{diag}(0.1s)$ and s being the diameter of the circle. However, the center does not contain the most ambiguous examples in all cases. For example, if during active learning only two classes have assigned training examples, the remaining test examples are distributed along the line connecting both classes which not necessarily runs through the center.

The second selection model “convhull” simulates user who do not pick from the center of the visualization but

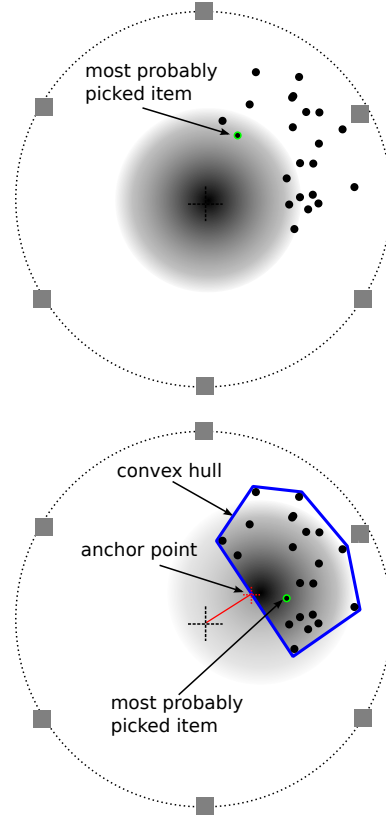


Figure 2: User picking models (top: “gaussian”, bottom: “convhull”). Gray squares represent classes, black dots unlabeled samples, gray gradient the probability of picking.

rather judge the distribution of the samples and then select the presumably most ambiguous sample based on this distribution. In particular the “convhull” selection model calculates the convex hull around the visible unlabeled examples and centers a bivariate Gaussian distribution around the point with the smallest distance to the center of the circle. If the center of the visualization is inside the convex hull then the center of the visualization is taken as the central point of the Gaussian; hence, the “convhull” selection model becomes the “gaussian” selection model described above. The covariance matrix for the “convhull” model is set the same as in the “gaussian” model. The convex hull is calculated using Graham’s algorithm [4]. An example for both user selection models, “convhull” and “gaussian”, is shown in figure 2.

Note that our selection models are neither complete nor exhaustive, and currently only represent simple static user behavior. For example, we do not consider changes in the picking behavior over time. However, as we will show in our experiments, changing picking strategies may become essential to overcome negative effects resulting from following only one particular selection strategy. By visualizing classification results, users are enabled to judge the current

situation and to change their selection strategy appropriately.

2) *User Labeling Strategies*: In addition to the selection behavior, we observed that user have different labeling strategies. For example, if the user is uncertain of the correct label (e.g., outliers, deficient user’s knowledge), users may reject to label an example. Also, a user may assume that labeling misclassified example provides more information than the labeling of correctly classified examples. In our simulation we model two cases: (i) the user always label the selected example and (ii) the user labels only mis-classified examples. The first model is called “all”, because the user assigns a label to each example presented by the selection strategy. The second labeling model is called “misses” indicating that the user only labels examples that are mis-classified by the current classifier. This latter strategy means that the user investigates examples one after another and only labels mis-classified examples. The “misses” strategy may fail in simulations in situations where no more misclassified examples occur in the visualization (the test data set is perfectly classified). Further, it is likely that no user would have the patience to investigate dozens of examples before eventually labeling one. Therefore, the “misses” strategy uses a threshold to limit the number of unsuccessful subsequent investigations. If this threshold is reached the example is labeled, independent of whether it was mis-classified or not.

We are aware that this extension of the different selection strategies is not comparable to traditional active learning approaches. In the traditional settings the user does not get an idea how well the example in question is currently classified and whether this decision is correct. We introduced the labeling strategies to investigate their influence in combination with the user-based selection strategies. Our hypothesis is that the error rate is decreased more rapidly when the “misses”-strategy is applied. In this experiments we do not consider the time required for the user to determine whether an example is labeled correctly or not. This time may vary depending on the data set and the visual representation of a data item. For example in image classification a user can recognize the class of an image more easily than in case of text documents. We report these experiments to get an idea what the best case gain in terms of accuracy would be while not considering the time required to perform this strategy.

III. EXPERIMENTS

We compared the derived user-based active learning strategies to classical active-learning strategies and passive learning on four different data sets and three different classifiers.

A. Data Sets

We performed the experiments on the COIL-20 image data set [8], a German text data set APA [7] (27570 documents, referred to as APA), the R8 subset of the Reuters-

21578 data set ¹ (7674 documents, referred to as REU-8) and the 20-Newsgroup data set² after removing duplicated entries (18828 documents, referred to as 20NEWS). All text data sets were annotated using a natural language processing module based on OpenNLP ³. We used the Part-Of-Speech tags to construct the noun vector spaces and applied TF-IDF weighting. All of these data sets were split into training and evaluation subsets, both sets are fully labeled.

B. Classifier Setup

For each of the data sets we trained two different classifiers. For the image data set COIL-20 we used a linear SVM with standard parameter settings and a KNN algorithm. For the KNN algorithm we set $K = 7$, because it resulted in the best performance in the experiments described in [14]). For the text data sets we used the LibLinear package [2] (LIBLIN) and the class-feature-centroid classifier [5] (CFC), both with standard settings. To estimate the a-posteriori distribution for each data set we used majority weighting for KNN and the sigmoidal fitting described in [9] for SVM, LIBLIN and CFC.

C. Evaluation approach

For all selection modes the classifier was bootstrapped with c randomly chosen samples (c = number of classes). The learner was trained in serial-mode, i.e., one example at a time was labeled and added to the training set. Afterwards, the model was retrained and the test data set reclassified. After every c examples we evaluated the classification error on the evaluation data set. To avoid random biases due to initialization or within the different selection strategies, we run every experiment ten times and averaged the results.

D. Results and Discussion

Figure 3 compares the classification error for the APA, REU-8, COIL-20 and 20NEWS data set respectively.

1) *Comparison of Active Learning Strategies*: Most noteworthy, there is no clear winning active learning strategy over all data sets. Moreover, we can not confirm that the compared uncertainty based sampling strategies have a clear advantage over random sampling.

In particular, on the APA data set (see figure 3a for CFC), both non-user-based learning strategies, “entropy” and “minconf”, have not been able to improve the classification accuracy. The random baseline outperforms “entropy” and “minconf” by a large margin. The user-based active learning strategies tend to be slightly worse at the beginning compared to the random baseline but could beat the random baseline by a significant fraction at the end.

On the REU-8 data set the classifiers showed quite different behaviors (see figure 3b for LIBLIN). The CFC seems

¹<http://www.daviddlewis.com/resources/testcollections/reuters21578/>

²<http://people.csail.mit.edu/jrennie/20Newsgroups/>

³<http://opennlp.sourceforge.net>

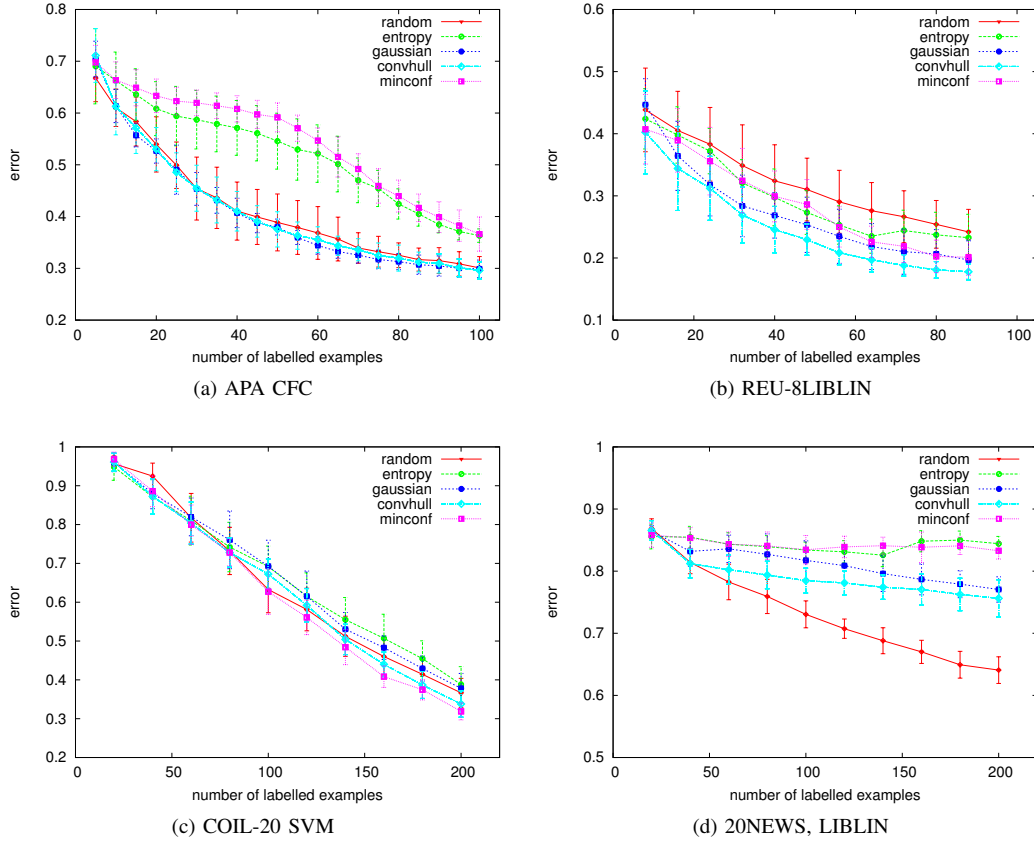


Figure 3: Error rates, averaged over 10 runs, showing mean and standard deviation

to converge faster, since after 20 examples the classification error drops below 0.3 independent of the selection strategy. Moreover the standard deviation significantly decreases after 30 examples for all selection strategies, indicating a more stable behavior. The LIBLIN classifier shows a different behavior. The random baseline is outperformed by all other strategies by a large margin. Moreover, the classical active learning strategies are outperformed by the user selection strategies, while the “convhull” strategy performs best.

The COIL-20 data set is the only data set where classical active learning strategies outperform the user-based (see Figure 3c). In particular “minconf” outperforms the random baseline in case of the SVM classifier while entropy-selection outperforms the random baseline in case of the KNN-7 classifier (figure omitted due to space constraints). Our user-based strategies, especially the “convhull” strategy performs quite similar to the random baseline.

The 20NEWS data set shows a special behavior (see figure 3d for LIBLIN). All active learning strategies got beaten by the random baseline by a large margin of nearly 20%. However, user-based active learning strategies perform better than classical ones.

On all data sets, classical active learning approaches have

a smaller standard deviation over the runs than our visual approaches. However, on most data sets our user-based methods perform still better taking worst-case runs, i.e. those with the highest standard deviation, into account.

From these results we can draw the following conclusions: (i) Our visually inspired active learning strategies outperform classical uncertainty based sampling strategies in most cases (20NEWS, Reu-8 and APA). (ii) We can confirm the findings summarized in [12], that the best sampling strategy depends on the application, the data set and the classifier and that the random baseline is not always outperformed.

Especially the last finding points towards a crucial property in active learning: users have to be able to find out whether the chosen active learning strategy works or not and to switch the active learning strategy as needed. The selection using the interactive visualization allows such switching of active learning strategies. However, while our results provide evidence that user-based active learning will work, the effects of real users have to be investigated in user studies.

2) *Qualitative Classification Analysis:* Efficient active learning requires to detect negative effects of the active learning strategy early in the labeling process. Hold out sets

may allow to do so, too. However, hold out sets have to be constructed prior to active learning and their construction suffers from the same problem as creating training sets. Intelligent visualization techniques may allow users to detect such patterns by visual means and without the need of a hold out set. Although quantitative insights are usually hard to grasp by such techniques, qualitative properties can be derived rapidly given appropriate visualization techniques. Hence, given the classification visualization we ask whether negative effects of the chosen active learning strategy can be detected qualitatively.

Figure 4 shows a sequence of visualizations along the steps taken by the CFC classifier on the APA data set for the different active learning strategies. As it can be seen in figure 4d, left, the initialization starts by randomly selecting examples resulting in a triangular shape. Since two classes do not have training examples, they do not attract any examples. Contrary to this starting figure, better classifiers should move all examples towards their position on the circle, resulting in a star-like shape (see figure 4d, right). Comparing the classification visualization for each step, i.e., each increase in training examples, shows the differences between the active learning strategies. In the example of the APA data set and the CFC classifier (see figure 4c for “gaussian”, figure 4b for entropy and figure 4a for “random”), one can observe that entropy-based selection ignores two classes to a large degree (remember, the initial samples were randomly chosen). Below 15 examples (3 steps in the APA Classifier) entropy-based selection focused only on two classes, which is not the case for “random” and “gaussian” selection. As outlined above (see figure 4, left), entropy-based selection does not work at all while the “gaussian” method performs similarly well as the random baseline. Hence, the relatively static layout in the “entropy” case up to 55 examples may be used as an indicator for an ill-chosen active learning strategy.

While user studies have to be undertaken to confirm these findings, we can conclude that including a visualization of the classifier’s decision bears the potential to identify ill-chosen active learning strategies. Hence, it supports users in switching active learning strategies without creating hold out sets.

3) *Evaluation of User Labeling Strategies:* As outlined in section II-A, we do not only want to evaluate different possible selection strategies as substitute or extension of active learning models, but also analyze whether different labeling strategies have an impact or not. Basically, uncertainty-based active learning does hardly allow a user to abstain from labeling. However, users might decide to ignore correctly classified examples and focus on wrongly classified examples as part of their labeling strategy. The simulation of the strategy that labels only mis-classified samples relies on performing “gaussian” or “convhull” selection until a wrongly classified example is selected as labeling candi-

Table I: Effect of labeling only mis-classified examples. “+”: increases accuracy, “o”: no effect

		gaussian	convhull
20NEWS	CFC	+	+
	LIBLIN	+	o
REU-8	LIBLIN and CFC	+	+
APA	LIBLIN	+	o
	CFC	o	o
COIL-20	SVM and KNN7	+	+

date. Figure 5 shows the comparison between this labeling strategy, denoted with the suffix “misses”, and the standard user-based selection strategies (denoted by the suffix “all”) for the REU-8 and the COIL data set.

The results are partly surprising. We expected to see a clear gain in accuracy when comparing the standard “all” strategies to the respective “misses” strategies. The hypothesis of improving the accuracy when applying the “misses” strategy can for instance be confirmed for REU-8 LIBLIN “gaussian” (see figure 5, left). In this case labeling only mis-classified examples leads to a significant improvement of the classifier accuracy. On the contrary, there is no effect on the accuracy when changing the “all” to the “misses” labeling strategy, for example in the setting APA CFC using “gaussian” selection (see figure 5, right). Table I summarizes the effect of applying the “misses” strategies for all combinations of data sets and classifiers.

One possible explanation for the above described behavior is the following: The COIL-20 data set contains no noise and no outliers. The “misses” strategy on this data set labels data items which are no outliers, are uncertain and mis-classified and thus provides more (correct) information to the classification model than the “all” strategy which leads to improved accuracy. On the contrary, the text data sets (REU-8, APA, 20NEWS) contain outliers. Applying the “misses” strategy could result in selecting outliers or noisy data items. The similar behavior of “misses” and “all” for the CFC algorithm could be explained by the properties of the CFC algorithm: The influence of one example on the class centroid is relatively small, the CFC regularizes well. Such neither the additional information of (uncertain and mis-classified) nor selection of outliers have strong influence on the accuracy. In case of the LIBLIN classifier (hard margin), outlier selection can result in a strong change of the hyperplane and thus in loss of generalization ability.

IV. CONCLUSION & OUTLOOK

From conducted experiments and user simulations we conclude, that utilizing humans to only label examples in active learning settings is suboptimal. Giving users a more active role in terms of a visual selection of examples

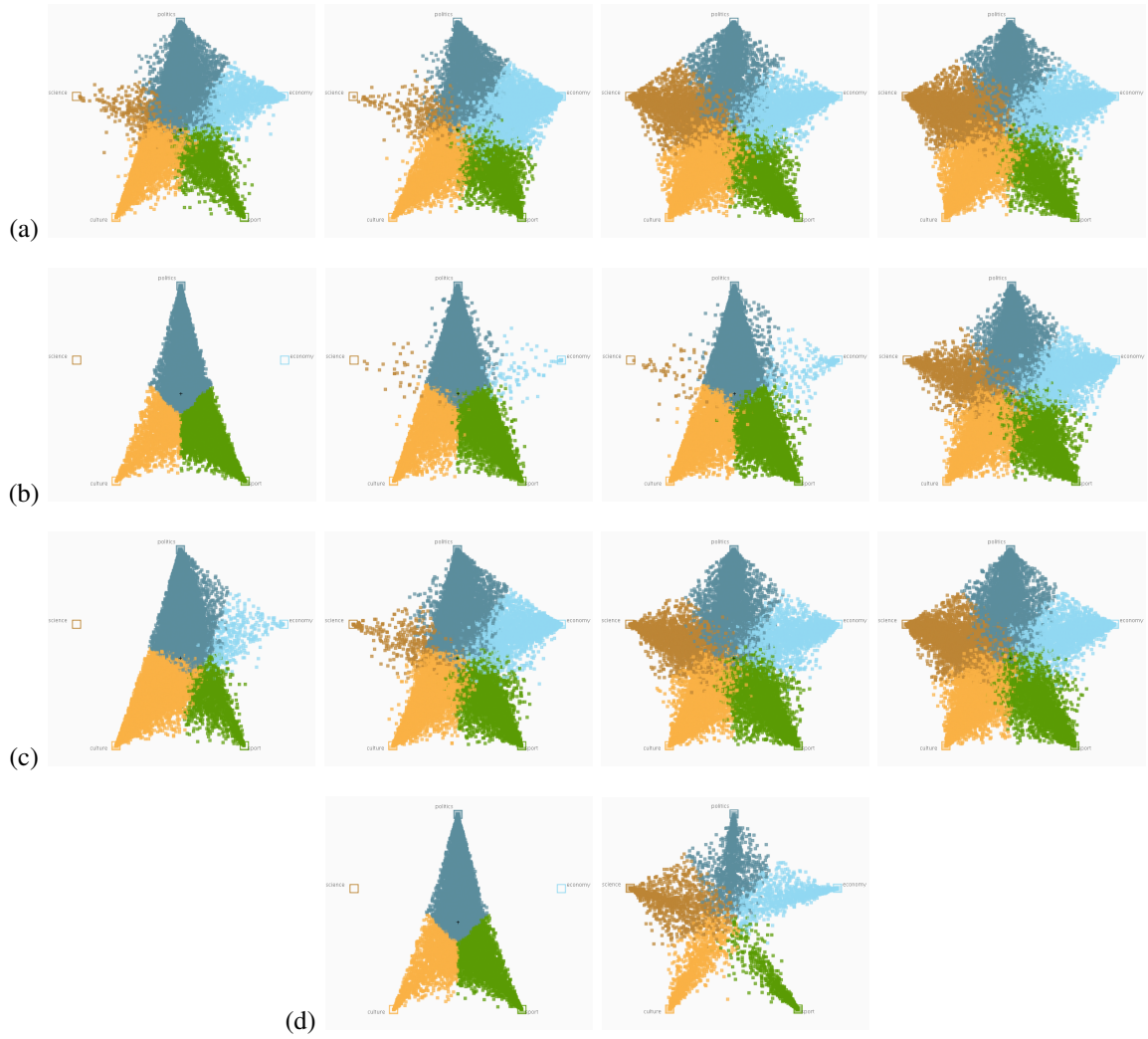


Figure 4: Visualization of APA-CFC after 10, 15, 55, and 80 training examples (left to right) using different selection strategies (a) random, (b) maxentropy, (c) gaussian. Bottom row: initial state (left) and the fully trained classifier (right).

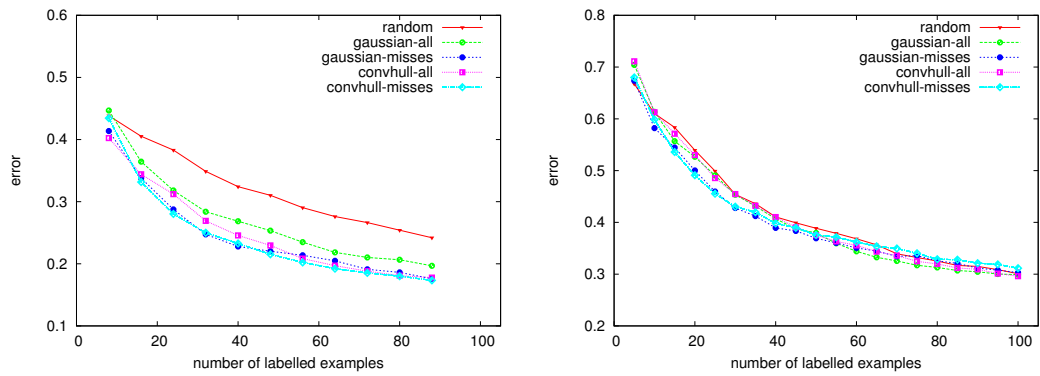


Figure 5: Comparing labeling strategies. left: REU-8 LIBLIN, right: and APA CFC. Error rates, averaged over 10 runs.

and in adapting their labeling strategies on top of tailored visualization techniques could increase labeling efficiency.

We presented a framework for simulating different user selection and labeling strategies for active learning. The basis for the user's decision is a visualization of the a-posteriori probabilities of the unlabeled samples. We simulated two different user selection strategies and compared them to uncertainty-based and random selection. With our experiments we showed that trained users would perform better than greedy active learning strategies in all cases and at least not perform worse than the random strategy while given more choice in the active learning process. The experiments for the different labeling strategies showed that the benefit of these strategies depends on the data set and the classifier.

The huge amount of simulations could not have been done with real users. (The complexity of the evaluation: 4 data sets, 3 classifier, 2 user picking and 2 user labeling strategies, 10 runs for each setting. $4 \times 3 \times 2 \times 2 \times 10 = 480$ user. 100 samples to label on average. Estimated user labeling time for on sample: 3 s for image data sets, 30 s for text data set per example. 50 min per user for one text data set.). Instead the obtained statistical results show a direction for user experiments and allow an estimate what to expect in real user studies. In turn, we plan to derive new user-based selection models from the user studies which then can be statistically evaluated using the simulation framework.

Additionally, batch selection of examples may further improve the labeling efficiency. This could be easily integrated into the visualization by the means of lasso selection and dragging of multiple examples. While this would enable fast labeling, it would also introduce noise in the training data. By using lasso selection not only correct examples would be chosen but also (nearby positioned) false examples. The resulting noisy training data set could be cleaned again by visualizing the training data (re-classify and show the differences of labels and classifiers decisions). Misclassified examples could then easily be filtered out and corrected.

Acknowledgments

This work has been partially funded by the European Commission as part of the TEAM IAPP (grant no. 251514) within the People Programme of FP7. The Know-Center is funded within the Austrian COMET Program - Competence Centers for Excellent Technologies - under the auspices of the Austrian Ministry of Transport, Innovation and Technology, the Austrian Ministry of Economics and Labor and by the State of Styria. COMET is managed by the Austrian Research Promotion Agency (FFG).

REFERENCES

- [1] Jason Baldridge and Alexis Palmer. How well does active learning actually work?: Time-based evaluation of cost-reduction strategies for language documentation. In *Proc. of the 2009 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 296–305, Morristown, NJ, USA, 2009. Association for Computational Linguistics.
- [2] Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. Liblinear: A library for large linear classification. *J. Mach. Learn. Res.*, 9:1871–1874, 2008.
- [3] Caroline Gasperin. Active learning for anaphora resolution. In *Proc. of the NAACL Workshop on Active Learning for Natural Language Processing (HLT)*, pages 1–8, Morristown, NJ, USA, 2009. Association for Computational Linguistics.
- [4] Ronald L. Graham. An efficient algorithm for determining the convex hull of a finite planar set. *Inf. Process. Lett.*, 1(4):132–133, 1972.
- [5] Hu Guan, Jingyu Zhou, and Minyi Guo. A class-feature-centroid classifier for text categorization. In *Proc. of the International conference on World Wide Web (WWW)*, pages 201–210, New York, NY, USA, 2009. ACM.
- [6] Yuhong Guo and Dale Schuurmans. Discriminative batch mode active learning. In *Proc. of Advances in Neural Information Processing Systems (NIPS)*, 2007.
- [7] Elisabeth Lex, Christin Seifert, Michael Granitzer, and Andreas Juffinger. Efficient cross-domain classification of weblogs. *International Journal of Computational Intelligence Research*, to appear.
- [8] Samer A. Nene, Shree K. Nayar, and Hiroshi Murase. Columbia object image library (coil-20). Technical Report CUCS-005-96, Department of Computer Science, Columbia University, February 1996.
- [9] John C. Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In *Advances in Large Margin Classifiers*, pages 61–74. MIT Press, 1999.
- [10] Andrew I. Schein and Lyle H. Ungar. Active learning for logistic regression: an evaluation. *Mach. Learn.*, 68(3):235–265, 2007.
- [11] Christin Seifert and Elisabeth Lex. A novel visualization approach for data-mining-related classification. In *Proc. of the International Conference on Information Visualisation (IV)*, pages 490–495. Wiley, July 2009.
- [12] Burr Settles. Active learning literature survey. Computer Sciences Technical Report 1648, University of Wisconsin–Madison, 2010.
- [13] Claude E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423, 1948.
- [14] Yang Song, Jian Huang, Ding Zhou, Hongyuan Zha, and C. Lee Giles. Iknn: Informative k-nearest neighbor pattern classification. In *Proc. of the European conference on Principles and Practice of Knowledge Discovery in Databases (PKDD)*, pages 248–264, Berlin, Heidelberg, 2007. Springer.
- [15] Katrin Tomanek and Fredrik Olsson. A web survey on the use of active learning to support annotation of text data. In *Proc. of the NAACL Workshop on Active Learning for Natural Language Processing (HLT)*, pages 45–48, Morristown, NJ, USA, 2009. Association for Computational Linguistics.