Active Learning to Classify Email

Bryan Klimt, Shyamsundar Jayaraman, Yiming Yang
Language Technologies Institute, School of Computer Science
Carnegie Mellon University
5000 Forbes Ave., Pittsburgh, PA 15213
{bklimt,shyamj,yiming}@cs.cmu.edu

Abstract

While the technique of active learning has been applied successfully in improving text classification, its use in email classification has still not been explored. This paper examines several of the state-of-the-art algorithms for active learning with support vector machines as they are applied to email folder classification. We also introduce several extensions to these methods specifically designed to improve the quality of active learning when used for email folders. We evaluated the relative accuracy of these algorithms using a large publicly available email corpus. Our results show that current methods for active learning used in text classification work poorly for email foldering, but by taking chronological information, such as receipt time, into account, we can improve upon them significantly.

1 Introduction

With the rising number of emails arriving in users' inboxes on a daily basis, helping them to sort that mail is becoming increasingly important. One approach to alleviating this problem is the design of algorithms that can automatically learn from a few examples the way in which a person organizes his or her mail. These algorithms can then use this learned model to sort the rest of that person's mail. Several algorithms have been devised for automatic email classification, but support vector machines (SVM) have been shown to work particularly well for this task. However, the quality of SVM in classifying email depends greatly on which documents are used as the training examples. Automatically choosing which documents to train on, known as *active learning*, has been extensively studied in other domains, but not in email classification. Using active learning, we could greatly reduce the number of emails a user would have to label in order to maximize the automatic classification of his email. In this paper, we explore the application of active learning techniques to SVM for email folder classification. We examine the state-of-the-art algorithm for active learning for text classification, and its performance on email. We also propose an extension to the algorithm to make it more suitable for the domain of email folder classification. The performance of these algorithms is then compared with each other, and with that of a common baseline.

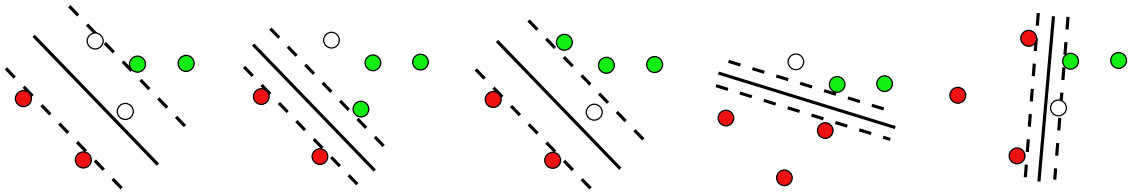
2 Algorithms

We chose Support Vector Machines as the basis for this work based on their previous success in email folder classification. SVM was found to outperform other learning algorithms for email folder classification in [Brutlag and Meek, 2000], especially for folders with a lot of messages. In [Klimt and Yang, 2004], it was found that there is a correlation between the volume of email a person receives and how much email they have in each of their folders. This indicates that SVM should perform best for users with a large amount of email. These are exactly the users that active learning can help the most.

Many approaches to active learning have been proposed, and it has been well proved that the number of required training examples can be drastically reduced, while maintaining a high level of accuracy [Iyengar *et al.*, 2000]. However, using active learning with SVM is a relatively recent phenomenon. In 2000, a simple approach was proposed for doing this [Schohn and Cohn, 2000; Tong and Koller, 2001].

To understand this method, one must first understand how SVM works. In a binary SVM classifier, training examples are mapped into a vector space. A hyperplane is then found which divides the vector space such that positive training examples fall on one side of the hyperplane and negative training examples fall on the other side of it. Since many such decision boundaries could exist, the one is chosen that will maximize the minimum distance from the boundary to the nearest positive and negative training examples. Twice the distance from the separating hyperplane to the nearest training example is known as the margin.

In hard-margin SVM, new training examples given to the classifier will have no effect on the location of this hyperplane if they fall outside of this margin, and would not provide new information for classification. However, if a new training example is within the margin, the hyperplane will move, and information will be gained. For the most part, soft-margin SVM will behave similarly. As shown in figure 2, the closer to the hyperplane an unlabeled document is, the more likely its label will improve. If the document chosen is labeled with the class that the SVM would have assigned it, then the document closer to the decision boundary would decrease the



(a) SVM with potential examples

(b) Closer example labeled as more probable class

(c) Farther example labeled as more probable class

(d) Closer example labeled as less probable class

(e) Farther example labeled as less probable class

Figure 1: SVM with Potential New Training Example Labels

margin more, thus refining the precision of the SVM. If the document chosen is labeled differently that SVM would have predicted, then the accuracy of the SVM is improved regardless of which training example was labeled. However, this is more likely to happen with the document closest to the decision boundary, since this is the document for which SVM is least sure of its prediction.

Thus, the approach proposed by [Schohn and Cohn, 2000] chooses examples to be labeled by picking those closest to the hyperplane first. This approach has a solid theoretical background as explained in [Tong and Koller, 2001], and performs very well in empirical testing [Schohn and Cohn, 2000]. In fact, it was shown to have higher accuracy with only a portion of the available training examples than a classifier trained with all of them. It was also shown that near-optimal performance can be reached when there are no longer any training examples that fall within the margin [Tong and Koller, 2001].

In some domains, however, active-learning performance with SVM can be improved even further, by taking training *diversity* into account. For example, consider the case in figure 2, where you have two new unlabeled documents, and you are only allowed to ask the user to label one of them. One of the documents is slightly closer to the decision boundary, but it is also very near several other training examples, which have all been identically labeled. The other document is slightly farther away from the decision plane, but there are no labeled documents anywhere near it in the hyperspace. In this case, you would likely do better by choosing the second document, as its label is less predictable, by virtue of its being in a sparse area of the hyperspace. Taking advantage of this, it has been found that the heuristic above for SVM active learning can be improved by linearly combining the distance of the new document to the hyperplane with the distance to other training examples. This technique improves the diversity of the labeled examples, and improves the performance of the active learning [Brinker, 2003].

Email classification differs from text classification in many ways, though, and this is true for active learning as well. To be useful for email classification, some additional issues must be addressed with these approaches. First, it must be adapted for use with multi-class classification, rather than just binary

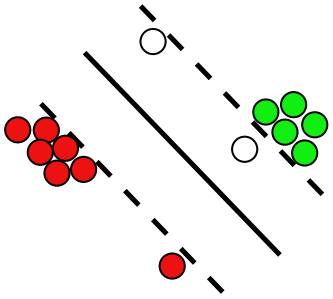


Figure 2: "Diversity" of Training Examples'

classifiers. A method for this extension is given in [Yan *et al.*, 2003], which provides a generalized framework for solving the problem. One approach they describe is to use the one-versus-rest method to reduce the problem to binary classifiers. Then, they choose the training example which is the closest to any of the classifiers' margins. In other words, they choose the example with the minimum distance to the hyperplane from all combinations of classifiers and training examples. This method must be similarly extended to deal with the multiple classifiers used for different sections of an email message, as in [Klimt and Yang, 2004].

3 Dataset

When evaluating automatic email folder classification, it is important to test on a dataset that accurately models the users whom the system would benefit. For that reason, we must choose data that is both realistic and has many users with a large amount of email, such that active learning could be useful for them. Furthermore, the users in the dataset must have manually sorted their email into folders (using their own organizational strategies). Then, we can compare our classifications with their folder labels. We have therefore chosen to use the freely available Enron Corpus [Klimt and Yang, 2004] for these experiments, as it is the only email corpus we are aware of which meets these requirements. The Enron corpus, which became available during court proceedings

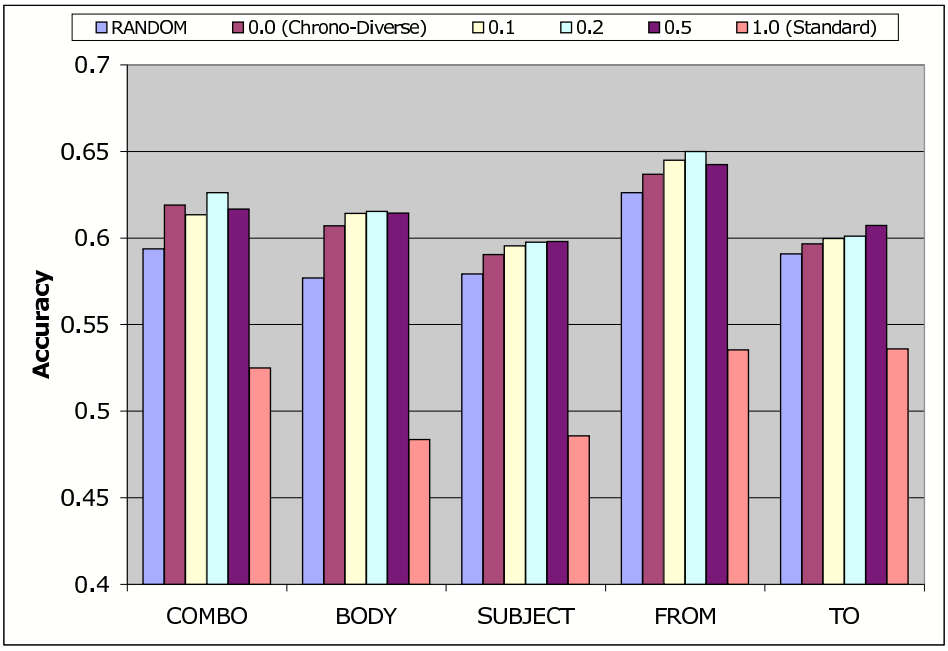


Figure 3: Accuracy with Active Learning Methods

against the Enron corporation, contains 200,399 messages of 158 users. Almost all of these users use folders for sorting their email, and they exhibit a wide variety of organizational strategies. This dataset is currently available online at `http://www-2.cmu.edu/~enron/`.

Because of time constraints and the long runtime of SVM, we were limited in the amount of data we could use for the experiments in our project. For our data, we chose the 100 of the 158 users in the Enron corpus with the least email. Although we would like active learning to be useful for users with the most messages, the techniques described in this paper were found to be intractable for some of the users in this dataset with hundreds of thousands of messages. Improving the efficiency of these techniques will therefore be an important step in future work. Furthermore, the emails in the Enron dataset range from 1997 to 2002, so in practical applications, the algorithm may have a considerable amount of time to run for users with a lot of email. The largest user in our dataset still has about 2000 messages, which is enough for active learning to be useful.

4 Experiments

For our experiments, we devised a scenario in which an email classifier is allowed to choose a small portion of a user's email to have them label. The classifier is then trained on these examples, and is tested by the accuracy of its classification of the remaining messages. With the classifier, we tried several active learning algorithms for choosing which emails should be used for training. The email classifiers used were based on those in [Klimt and Yang, 2004]. A separate SVM classifier was trained on each of the different parts of the email message, such as "from", "subject", etc.. The scores for each of these SVMs were then combined using a linear function that was learned from the training examples using least squares regression. Multi-class classification was handled using the standard one-vs-rest approach. The parameters used in these experiments are approximately the same as in [Klimt and Yang, 2004]. However, the algorithms had to be modified to be trained incrementally, to accommodate the active learning. Words were canonicalized using the Porter stemmer for English [Porter, 1980]. We then applied the "ltc" term-weighting scheme as defined by [Buckley *et al.*, 1995], such that the value w_{ik} for term k in the feature vector for email i is given by

$$w_{ik} = \frac{(\log(f_{ik}) + 1.0) * \log(N/n_k)}{\sqrt{\sum_{j=1}^{t}[(\log(f_{ij}) + 1.0) * \log(N/n_j)]^2}}$$

where f_{ik} is the frequency of term k in email i, N is the number of emails that have been seen, and n_k is the total number of emails we have seen so far which contain the term k. This function also results in feature vectors that are normalized to unit length.

Basically, three algorithms are investigated, in addition to the baseline. For our baseline, the classifier picked which emails to have labeled completely randomly. This is a common baseline in active learning research. In our first experimental algorithm, which we will call "Standard", new training examples were chosen that were, on average, closest to the decision boundaries of the classifiers. This is the method proposed in [Schohn and Cohn, 2000; Tong and Koller, 2001], which has been shown to be the state-of-the-art in SVM active learning for text classification, extended for one-vs-rest as described above. The results for this algorithm, as described in the next section, motivated our second algorithm, called "Chrono-Diverse". This algorithm simply picks each training example such that it is maximally distant from

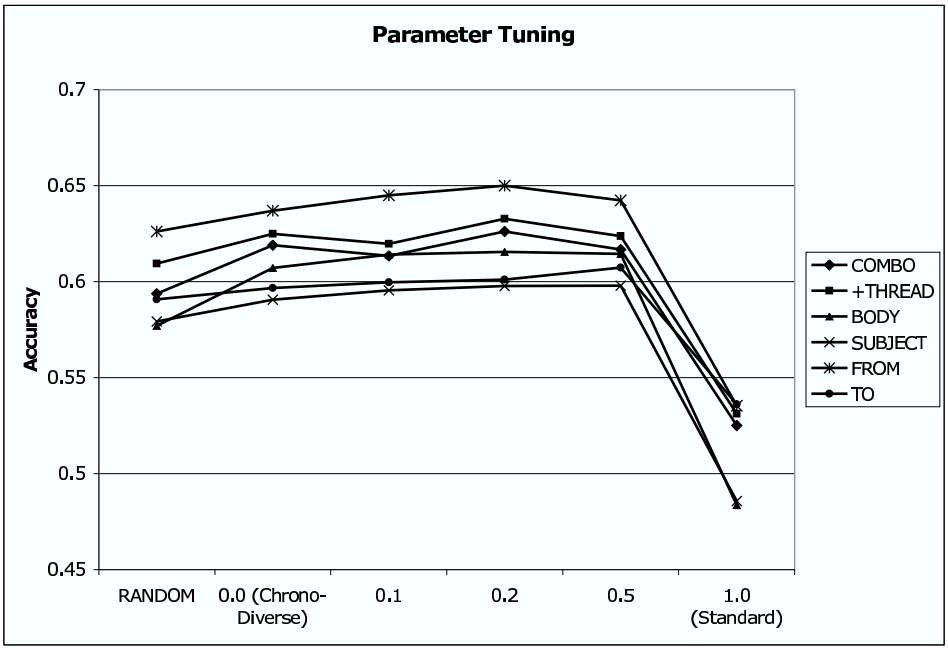


Figure 4: Tuning of parameter λ

all of the other training examples with respect to the time the email was received. This means that the least recent email is chosen first, followed by the most recent. Then, the email closest to the middle of the time span is chosen, and so forth, similar to a binary search. Finally, we tried a "Combined" approach, which ranked potential training examples by taking a linear combination of the score from the other two active learning algorithms. This introduces a tunable parameter, λ, used in the linear combination:

$$Score = \lambda \cdot Score_{Standard} + (1 - \lambda) \cdot Score_{ChronoDiverse}$$

Thus, "Chrono-Diverse" is equivalent to "Combined" with $\lambda = 0.0$.

We evaluated each algorithm by training it on 10% of a user's data and measuring its accuracy in classifying the other 90% of the messages. To train with larger portions of data would have been interesting, but was not tractable with these methods. We trained each classifier as detailed in [Klimt and Yang, 2004], with a separate SVM for each of the sections of the email. A linear regression was then used to find the weights to combine their classification score. We evaluated the scores for the classifier for each section. To make hard decisions, rank-based thresholding was used, only taking the most highly ranked folder as the prediction. We measured the classification accuracy for each user, and averaged them for the final score.

5 Results and Analysis

The results of our experimental runs are given in Figure 3. Each classifier was tried using random selection, as well as our three other algorithms, with $\lambda = 0.1$, 0.2, and 0.5 for "Combined". Shockingly, "Standard", the state-of-the-art SVM active learning method for text classification, performed horribly, even much worse than random selection.

Further investigation showed that the reason for this was that, when the classifier did not have much training data, it found that all of the training documents would probably be about equally informative. Thus, choosing which email to have labeled was decided by a very small difference in scores. This is not the right approach, however, as it is known that users tend to change the way they organize their mail over time, and this information is not utilized in the "Standard" method. Thus, we introduced the other extreme, the "Chrono-Diverse" method. Since this algorithm looks at emails spread evenly across time, it captures the diversity of the foldering strategies the user has used, and performs much better overall. This is consistent with the concept of diversity as a desirable goal in active learning, as introduced in [Brinker, 2003].

The obvious weakness of the "Chrono-Diverse" method, though, is that when a large amount of training data has been seen, the relative informativeness of the individual emails could be more useful than continuing to sample smaller time segments. This is the motivation for the "Combined" model, which initially relies on chronological diversity, but eventually uses the traditional method for selection. The parameter for the "Combined" model, λ, must be chosen empirically. Our tuning of λ is shown in Figure 4. We found that 0.2 was optimal for our dataset, but that the results were stable as long as λ was relatively small. We would have liked to further refine this parameter, but time did not permit it.

One final aspect of the results to notice is the relatively poor performance of the "COMBO" classifier, while it has been shown to be the best classifier in previous experiments [Klimt and Yang, 2004]. Our hypothesis for its low score is that it suffers from the change to being incrementally trained. When

choosing the weights for linearly combining its component classifiers, the "COMBO" classifier will give low weights to a classifier that has performed poorly at classifying previous examples. Thus, it punishes some of its component classifiers for failing on old training examples, when they would correctly classify those examples if they were given to it now. In the future, we will test this hypothesis by changing the "COMBO" classifier to retrain on all of the data every time one of its component classifiers is updated.

6 Conclusions and Future Work

From these experiments, it is clear that the current state-of-the-art techniques for active learning with SVM for text classification are inappropriate for email folder classification, when applied alone. However, combining them with chronological information greatly improves their performance. Furthermore, it is clear that, initially, time information is actually *more* important than the heuristics employed by the previous methods.

Active learning for email folder classification with SVM remains an open challenge. For future research, we will need to improve the efficiency of incremental SVM training to allow these methods to be applied to larger datasets. In addition, more analysis needs to be done to see under which circumstances these methods do relatively well or poorly. For one, we would like to see how their performance improves when different amounts of mail are used, when the performance improvement would become saturated, and how to adjust the sampling strategy accordingly to minimize user interaction and maximize effective learning by the system.

Acknowledgments

This material is based upon work supported by the Defense Advanced Research Projects Agency (DARPA) under Contract No. NBCHC030029.

References

[Brinker, 2003] K. Brinker. Incorporating diversity in active learning with support vector machines. In *Proceedings of the Twentieth International Conference on Machine Learning*, pages 59–66, 2003.

[Brutlag and Meek, 2000] J. Brutlag and C. Meek. Challenges of the email domain for text classification. In *Proceedings of the Seventeenth International Conference on Machine Learning*, pages 103–110, 2000.

[Buckley *et al.*, 1995] C. Buckley, G. Salton, J. Allan, and A. Singhal. Automatic query expansion using SMART: TREC 3. In *Proceedings of the Third Text REtrieval Conference*, pages 69–80. NIST Special Publication, 1995.

[Iyengar *et al.*, 2000] V. Iyengar, C. Apte, and T. Zhang. Active learning using adaptive resampling. In *Proceedings of the Sixth ACM SIGKDD*, pages 91–98, 2000.

[Klimt and Yang, 2004] B. Klimt and Y. Yang. The Enron corpus: A new dataset for email classification research. In *Proceedings of the Fifteenth European Conference on Machine Learning*, pages 217–225, 2004.

[Porter, 1980] M. Porter. An algorithm for suffix stripping. *Program*, 14(3):130–137, 1980.

[Schohn and Cohn, 2000] G. Schohn and D. Cohn. Less is more: Active learning with support vector machines. In *Proceedings of the Seventeenth International Conference on Machine Learning*, pages 839–846, 2000.

[Tong and Koller, 2001] S. Tong and D. Koller. Support vector machine active learning with applications to text classification. *Journal of Machine Learning Research*, 2:45–66, 2001.

[Yan *et al.*, 2003] R. Yan, J. Yang, and A. Hauptmann. Automatically labeling data using multi-class active learning. In *Proceedings of the International Conference on Computer Vision*, pages 516–523, 2003.