Modeling Personalized Email Prioritization: Classification-based and Regression-based Approaches

Shinjae Yoo
Computational Science Center
Brookhaven National Lab.
Upton, NY, USA
sjyoo@bnl.gov

Yiming Yang
Language Technologies Inst.
Carnegie Mellon University
Pittsburgh, PA, USA
yiming@cs.cmu.edu

Jaime Carbonell
Language Technologies Inst.
Carnegie Mellon University
Pittsburgh, PA, USA
jgc@cs.cmu.edu

ABSTRACT

Email overload, even after spam filtering, presents a serious productivity challenge for busy professionals and executives. One solution is automated prioritization of incoming emails to ensure the most important are read and processed quickly, while others are processed later as/if time permits in declining priority levels. This paper presents a study of machine learning approaches to email prioritization into discrete levels, comparing ordinal regression versus classifier cascades. Given the ordinal nature of discrete email priority levels, SVM ordinal regression would be expected to perform well, but surprisingly a cascade of SVM classifiers significantly outperforms ordinal regression for email prioritization. In contrast, SVM regression performs well – better than classifiers – on selected UCI data sets. This unexpected performance inversion is analyzed and results are presented, providing core functionality for email prioritization systems.

Categories and Subject Descriptors

I.7.m [**Computing Methodologies**]: Document and Text Processing—*Miscellaneous*; I.5.4 [**Computing Methodologies**]: Pattern Recognition—*Applications*

General Terms

Algorithms, Experimentation, Human Factors

1. INTRODUCTION

Email overload is an endemic problem, especially for the busiest and most productive professionals, managers and executives, and trends indicate aggravation rather than alleviation of the problem. Spira and Goldes report that a typical worker receives 200 non-spam email messages per day [21] and managers receive increasing numbers as their responsibilities broaden. NSF program managers, for instance, report 500 to 1000 non-spam emails per day. Productivity is compromised by constantly reading email streams loaded with low-priority announcements and acknowledgments, or alternatively by not reading email frequently and ignoring high-priority urgent emails. Hence, a partial solution is to develop automated email prioritization software, whose output would be binning incoming email streams into discrete priority levels, enabling the time-challenged user to query only the highest priority emails frequently, and lower priorities in declining order if and when she has the time to do so.

An additional major challenge is that unlike spam filtering, where the vast majority of email users would agree on what constitutes spam, e.g. Viagra commercials and penny-stock scams, non-spam email prioritization is very much user dependent. A patient receiving an email from her physician with the latest lab test results may be of crucial importance to her. But the same email sent back to the lab acknowledging receipt of the test results would be considered much less important by the lab technician. Hence, we need to mine each user's email data and solicit priority judgments in order to build personalized email prioritization systems. Given privacy issues and personalized priority preferences, data mining and priority elicitation is best done in massively distributed environments, replicating the process of each user, e.g. on a cloud or at the client side. This paper explores two different machine learning approaches to personalized email prioritization (PEP).

Although there has been significant and successful work in spam filtering, past progress on directly addressing PEP has been frustratingly slow. There are two main reasons: (1) privacy issues making it difficult to share open data sets of personal email, and (2) personalization entailing per-user data sparsity of priority-labeled emails. With respect to the first issue, possibly none of us would make our entire mailbox openly available to researchers; nor would email providers such as Google or Yahoo do likewise with their clients' email, for good reason. Due to the lack of personal priority labels, we could not use Enron corpus for PEP [11], which has been widely used as an email research benchmark dataset. Although Google recently started email prioritization service, they have not released the details of their models and the prioritization service has only two levels, whether it is important or not. In a nutshell, there are no publicly available dataset with very personal priority labels. To make progress on PEP, there is no other way except privately collecting PEP datasets from each individual subject, entails strict IRB (Institutional Review Board) supervision. It is an expensive and time consuming process, but we collected a modest dataset for our research. We should note, however, that once a PEP system is developed and proved

CIKM'11, October 24–28, 2011, Glasgow, Scotland, UK.

effective, no data sharing is required for its widespread application, since it trains on each user's emails and priority judgments, without cross-talk. The latter issue – limited priority judgments per user – requires us to investigate machine learning methods with a degree of robustness and suggests future work in active learning.

This paper addresses how to model and predict personal email priority. Specifically, we systematically analyze the effect of (1) multiple importance levels (ordinal regression) [23] and (2) personalization of email priority, using a five level Likert scale [14]. We investigate two machine learning approaches (1) ordinal regression, primarily SVM-based, and (2) classification-based, primarily a cascade of SVM classifiers. Intuitively, regression-based approaches look promising and appear to be natural choice for personalized email prioritization. To our best knowledge, there are only a handful of previous research efforts after the first mention of email overload in 1982 [3] and there have been no conclusive results on how to model personal priority. In 1999, Horvitz et al. [8] built an email alerting system which used Support Vector Machines to classify newly arrived email messages into two categories, i.e., high or low in terms of utility. In contrast to Horvitz et al., Hasegawa and Ohara [7] chose the alternate approach, using linear regression [13] but only looked at two priority levels, high and low. Recently Yoo et al. [23] applied classifiers in five-level priority prediction, presenting a basis for the present work. In summary, there have been no systematic comparisons between classification-based and regression-based approaches in terms of email priority modeling. Also there was no consideration of personalization except in Yoo et al., which serves as an initial basis for the present work.

The main contribution of this paper is a thorough comparison of ordinal regression versus classifier cascades as the underlying machine learning engine for PEP. We present the first thorough and systematic study with both regression-based approaches and classification-based approaches (including our new approaches) addressing the PEP problem based on personal importance judgments of multiple users and further analyzing on ordinal regression benchmark dataset for general performance and synthetic dataset for controlled study. Especially for personalized email prioritization, we extended the dataset of Yoo et. al. [23] from seven to 19 subjects. Specifically, our contributions in this paper include:

1. We summarized and analyzed the advantages, disadvantages and their assumptions of the regression-based approaches and classification-based approaches with the perspective of personalized email prioritization and general ordinal regression problems in Section 2 and 3.

2. Based on our analysis, we propose new approach to handle personalized ordinal regression problems in Section 3. Our proposed models take advantages of both classification-based approaches and regression-based approaches. It is based on how to model more flexible models like classification based approaches and yet we want to take advantages of partial ordinal relations if there are.

3. We evaluate two approaches with our proposed methods in three different dataset: personalized email prioritization dataset, ordinal regression benchmark dataset [2] and synthetic dataset (Section 4). The PEP dataset strongly favors classification-based approaches but ordinal regression benchmark dataset follows the underlying assumption of SVOR, and performs better on those data sets (by an 8% to 16% margin). Using an additional synthetic dataset, we can control the environment to allows us to tell which method is better under given conditions

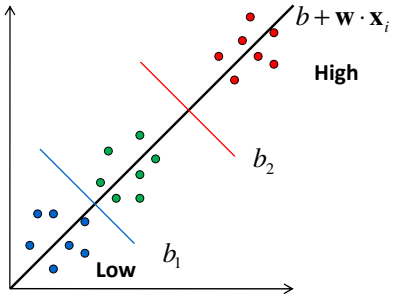


Figure 1: Three ordinal levels with a regression model and two separating thresholds

2. REGRESSION-BASED APPROACHES

2.1 Pure Regressions

The natural choice to handle ordinal response variables such as priority levels, survey answers or movie preference ratings is an ordinal regression model. After applying standard regression such as linear regression [13] or support vector regression [4], we may map the response variable to certain ordered discrete values, such as priority levels. For instance, SVR (Support Vector Regression) optimizes the following conditions:

$$\min_{\mathbf{w},b,\xi,\xi^*} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n}(\xi_i + \xi_i^*) \tag{1}$$

subject to

$$\begin{aligned}(\mathbf{w}\cdot\mathbf{x_i} - b) - y_i \le \varepsilon + \xi_i, \xi_i \ge 0, \forall i \\ (\mathbf{w}\cdot\mathbf{x_i} - b) - y_i \ge -\varepsilon - \xi_i^*, \xi_i^* \ge 0, \forall i\end{aligned} \tag{2}$$

where $\mathbf{w} \in R^d$ is a row weight vector and $\mathbf{x_i} \in R^d$ is a column vector for the input, ε is the margin for regression, ξ_i and ξ_i^* are slack variables, C is a regularization parameter and b is the intercept of a regression model. In case of prediction, we pick the closest level l from the predicted score of $\mathbf{w}\cdot\mathbf{x_i} - b$.

There are two important assumptions we need to address when we model ordinal regression problems by using the pure regression models. The first assumption is that one weight vector $\mathbf{w}$ defines the whole ordinal relations among different levels from Equation 1. As shown in Figure 1, the decision hyperplanes are parallel to each other and orthogonal to the weight vector $\mathbf{w}$. We call it *one model assumption* because there is only one weight vector $\mathbf{w}$ compared to multiple weight vectors of classification-based approaches. Since it is biased to have only one model or parallel decision hyperplanes, it is economical and it could be less sensitive to the noisy data than multiple models as shown in Figure 2 where we have three hyperplanes and they are not parallel. Since PEP (Personalized Email Prioritization) has to handle

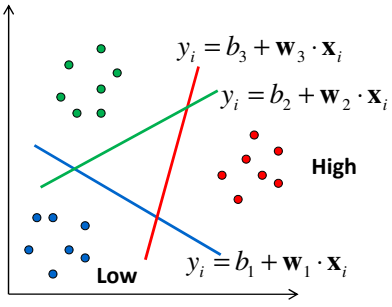


Figure 2: Three ordinal classes with three hyperplanes (OVA)

limited amount of training data, it would be attractive to have only one model to represent whole priority relations. However, if the assumption does not hold, the performance of regression models may not be guaranteed. In other words, the decision hyperplanes may not be parallel. In practice, PEP has to handle personalized priorities and the user defined priority is not necessarily satisfying this assumption. If a priority is based on a task or topic, then it could be closer to classification than regression.

The second underlying assumption is that the pure regression approaches assume *the fixed equal distance* between adjacent ordinal levels. This assumption could be less critical than *one model assumption* but it is still affecting the accuracy of prediction because regression models predict to the closest level. For instance, the difference between *important* and *very important* could be smaller than the difference between *neutral* and *important*.

2.2 Ordinal Regressions

Rather than modeling an ordinal regression problem through the pure regression, we may explicitly model ordinal regression. Ordinal regression models drop the second assumption, *the fixed equal distance* between adjacent levels. Therefore, it provides multiple thresholds which tell us the predicted priority levels as shown in Figure 1, although it still learns one regression weight vector $\mathbf{w}$. These thresholds allow us to have different distances among different levels. For example, Support Vector Ordinal Regression (SVOR) [2] learns a model weight vector $\mathbf{w}$ and $r-1$ thresholds when we have r priority levels.

More specifically, SVOR optimizes the following conditions:

$$\min_{\mathbf{w},b,\xi,\xi^*} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{j=1}^{r-1}\sum_{i=1}^{n_j}\left(\xi_i^j + \xi_i^{*j}\right) \tag{3}$$

subject to

$$\begin{aligned}(\mathbf{w}\cdot\mathbf{x_i^j} - b_j) &\le -1+\xi_i^j, \xi_i^j \ge 0, \forall i,j\\ (\mathbf{w}\cdot\mathbf{x_i^j} - b_{j-1}) &\ge 1-\xi_i^{*j}, \xi_i^{*j} \ge 0, \forall i,j\\ b_{j-1} &\le b_j, \text{for} j = 2,\cdots,r-1.\end{aligned} \tag{4}$$

where n_j is the number of training emails which belong to priority level j, b_j is the threshold for j or lower level threshold, and $\mathbf{x_i^j}$ is j^{th} priority level email. The formulation of SVOR is quite similar to SVR but SVOR has $r-1$ thresholds, b_j, compared to only one intercept b of SVR.

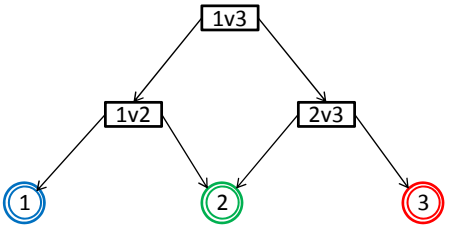


Figure 3: Decision DAG (Directed Acyclic Graph) for One vs. One multi-class classification. The rectangular represents a OVO classifier and the double circle shows the final decision. When testing a decision node, take the left child if the left-hand class is more probable than the right-hand class.

3. CLASSIFICATION-BASED MODELS

3.1 Multi-class Classification

We can drop *one model assumption* by treating the ordinal regression problem as multi-class classification problems and thus we may have multiple models for each priority level. Multi-class classification provides the most flexible model but there are no predefined relations among different priority levels. Although there are numerous ways to build multi-class classifiers from binary classifiers, we focus on three popular approaches: OVA (One vs. All), OVO (One vs. One), and DAGSVM [18].

One vs. All (OVA), also known as One vs. Rest (OVR), is the most common way to handle the multi-class classification problem, Figure 2. OVA treats remaining classes as negatives and thus we need r models if we have r priority levels. When testing, we choose the most confident priority level as our prediction.

One vs. One (OVO), also known as all pairs, build all possible pairs of binary classifiers [9, 15] such as (1 vs. 2), (1 vs. 3), ..., ($r-1$ vs. r). When testing, each classifier votes and the majority class will be the predicted class. Although One vs. One (OVO) classification requires $r\cdot(r-1)/2$ classifiers, each classifier has less amount of training examples than OVA classifiers and thus overall training time is reduced [9].

Instead of majority voting, we may use decision DAG (Directed Acyclic Graph) during testing as shown in Figure 3. We call it DAG instead of DAGSVM [18] because we may apply it to different classifiers instead of just SVM. DAG is faster than OVO during prediction because it requires only $r-1$ test. Although Plattet et al. [18] reported the order of classes from DAG did not affect final results, we sorted the order of priority levels as shown in Figure 3.

3.2 Order Based DAG

Although regression models make use of priority relations, their models are not flexible due to *one model assumption*. It could be critical for personalized email prioritization because each person might have different assumption about the priority levels. Multi-class classification provides flexibility because they allow multiple models among the different priority levels. However, they ignore the ordinal relations among the priority levels. Therefore, we propose models which have both the flexibility of multi-class classification models and the ordinal relations of regression models.

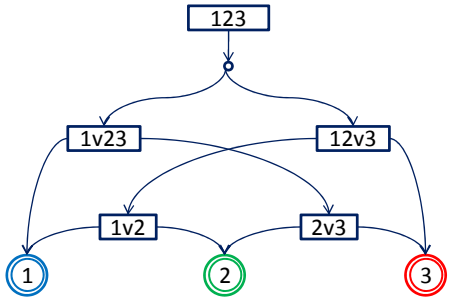


Figure 4: Decision DAG (Directed Acyclic Graph) for three level Order-Based (OB) classification. The rectangular represents a OB classifier and the double circle shows the final decision. When testing a decision node, take the left child if the left-hand class is more probable than the right-hand class.

Rather than directly predicting each priority level, we may use the order information for guiding better specific cases. Figure 4 shows the decision directed acyclic graph (DAG) for Order-Based (OB) classification models. When there are multiple paths available from top nodes to leaf nodes, any path may guide to the correct decision as long as each node's decision is correct. Since there are multiple choices available, we can always choose the **most confident** decision node among candidate decision nodes, OB-MC or we may do **majority voting**, OB-MV. For instance, when we have three priority levels, we can start from both "12 vs 3" and "1 vs 23" of Figure 4. For a testing email x_i, suppose that an SVM classifier trained "12" as positive and "3" as negative training classes (12 vs 3) and the classifier predicted 0.7 but SVM trained with "1" as positive and "23" as negative training labels (1 vs 23) and predicted -0.9. In case of OB-MC, we follow "1 vs 23" decision path because -0.9 is more confident than 0.7 and the next decision node is "2v3" instead of "1" due to the negative prediction score. OB-MV test all possible paths and then majority voting will determine which one is our final decision. If there are even votes, we may test even votes results using one vs remaining even vote node classification. For instance, "12 vs 3" predicted "1" for final decision but "1 vs 23" ended up with "3". Then we choose the better one out of "1 vs 3".

Order-Based approaches have multiple flexible models as classification-based models but they also have model bias to the order of priority levels as regression-based models, resulted in robust modeling to the noisy data. If the priority levels have no relations (perfect for classification) or satisfy ordinal regression assumption (perfect for regression), our proposed order-based approaches may not be able to outperform than these two extreme approaches. However, if users have set any form of partial ordinal relations, then our proposed models have a potential to improve the prediction accuracy.

When we apply r level prioritizer, the total number of basic classifier is $\sum_{k=1}^{r}(r-k+1)\cdot(k-1)$. The classification models listed above can be paired with any kinds of classification algorithm and we tested SVMs and Regularized Logistic Regression depending on dataset.

Table 1: Collected Dataset

User	# of emails	# of train	# of test
1	1750	150	1600
2	503	150	353
3	519	150	469
4	989	150	839
5	275	150	125
6	279	150	129
7	234	150	84
8	899	150	749
9	408	150	258
10	404	150	254
11	282	150	132
12	863	150	713
13	758	150	608
14	476	150	326
15	2989	150	2839
16	569	150	419
17	816	150	666
18	582	150	432
19	1126	150	1076
Avg	774.8	150	624.8

Table 2: PEP Evaluation results with p-values

# of tr	Base(b) MAE	SVOR(o) MAE	SVOR(o) *p-val*(b)	OB-MV MAE	OB-MV *p-val*(b)	OB-MV *p-val*(o)
30	1.1560	1.1340	0.3576	**0.9980**	* 0.0148	* 0.0288
60	1.1560	1.0736	0.1362	**0.9185**	* 0.0010	* 0.0197
90	1.1560	1.0459	0.0844	**0.8837**	* 0.0004	* 0.0189
120	1.1560	1.0441	0.0746	**0.8791**	* 0.0003	* 0.0141
150	1.1560	1.0480	0.0902	**0.8689**	* 0.0002	* 0.0143

(a) Macro MAE Results

# of tr	Base(b) MAE	SVOR(o) MAE	SVOR(o) *p-val*(b)	OB-MV MAE	OB-MV *p-val*(b)	OB-MV *p-val*(o)
30	1.0887	1.0992	* 0.0000	**0.9700**	* 0.0000	* 0.0000
60	1.0887	1.0647	* 0.0000	**0.8597**	* 0.0000	* 0.0000
90	1.0887	1.0406	* 0.0000	**0.8140**	* 0.0000	* 0.0000
120	1.0887	1.0278	* 0.0000	**0.8083**	* 0.0000	* 0.0000
150	1.0887	1.0259	* 0.0000	**0.7907**	* 0.0000	* 0.0164

(b) Micro MAE Results

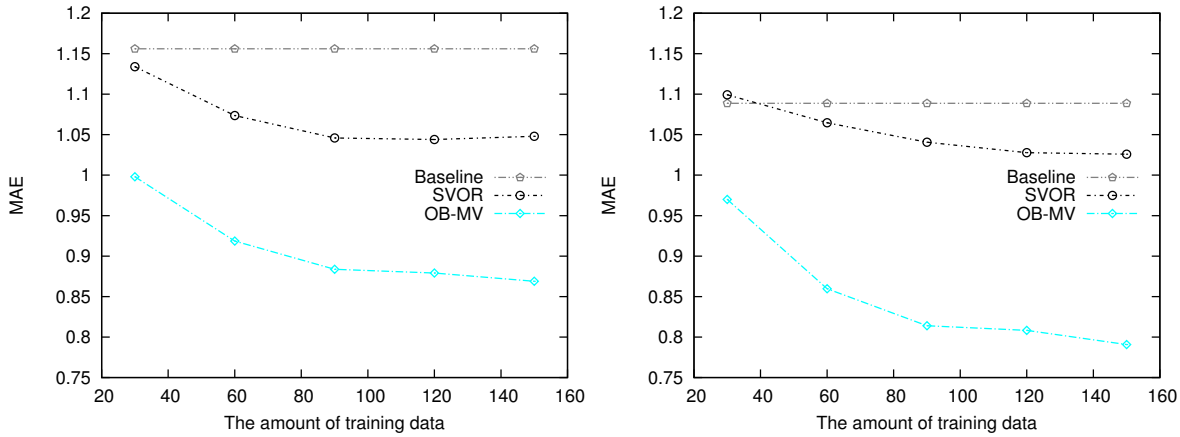
4. EXPERIMENTS AND ANALYSIS

We evaluated the regression-based approaches and classification-based approaches on three different datasets.

4.1 Personalized Email Prioritization

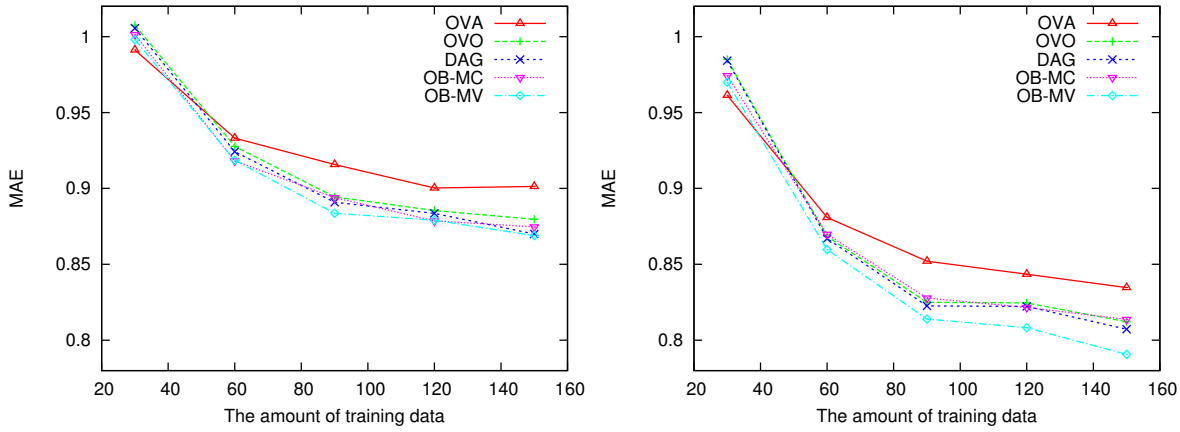
4.1.1 Dataset and Preprocessing

We extended the personalized email prioritization datasets of Yoo et al. [23] from 7 experimental subjects to 19 experimental subjects using our developed Thunderbird Add-ons. The original datasets were collected from Carnegie Mellon University but we recruited additional subjects from local churches and universities. We finally collected data from two faculties, six staff members, ten students, two pastors and one job seeker. We asked the subject to label at least 400 non-spam emails during one month period and suggested labeling 800 non-spam emails (or equivalently labeling 40 emails per day). We applied five importance levels (not important at all, not important, neutral, important, and very



(a) Macro MAE (b) Micro MAE

Figure 5: Macro and Micro Average MAE Learning Curves with Baseline, SVOR and OB-MV



(a) Macro MAE (b) Micro MAE

Figure 6: Macro and Micro Average MAE Learning Curves Among Classification-based Approaches.

important). Table 1 shows the summary statistics of the final collected emails with labels. We split the first 150 email messages as training and the rest as testing based on the timestamp of email messages. If we did not reserve the first 150 email messages as training, then we could build prioritization models from future data and it would not be realistic.

We applied the email address canonicalization [23]. Then we preprocessed email messages by tokenization but we did not remove stop words or apply stemming. The basic features were the tokens in the sections of from, to, and cc address, title, and body text of email messages. We used *ltc* term weighting to construct a document vector per message.

Since we want to show improvement on limited amount of training data through learning curves, we randomly shuffled 150 training examples ten times and choose every 30 training email increments from 30 emails to 150 emails.

4.1.2 Estimation Models

For classification-based approaches, we used linear SVM classifiers for our base classifiers. Each classifier took the vector representation of each message as its input, and produces a score with respect to a specific importance level. We used the SVM^{light} software package and tuned the margin parameter C in the range from 10^{-3} to 10^{3} with ten-fold cross validation of training data.

For regression-based approaches, we tested only SVOR with implicit constraints [2] with linear kernel. We tested explicit constraints SVOR and other non-linear kernels but they showed worse results than implicit constraints SVOR with linear kernels. Again we only tuned regularization parameters with the same ranges of SVM classifiers.

Our baseline always predict priority level 3 out of 5 levels, which is the most common priority level on our data collection and which minimizes MAE.

4.1.3 Evaluation Metric

We use MAE (Mean Absolute Error) as the main evaluation metric, which is standard in evaluating systems that produce multi-level discrete predictions. MAE is defined as:

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i| \quad (5)$$

where N is the number of messages in the test set, y_i is true priority level and $\hat{y}_i$ is predicted priority level. Since we have five levels of importance, the MAE scores range from zero

(the best possible) to four (the worst possible). There are two conventional ways to compute the performance average over multiple users. One way is pooling the test instances from all users to obtain a joint test set, and computing the MAE on the pool. This way has been called *micro-averaged* MAE. The other way is to compute the MAE on the test instances of each user and then take the average of the per-user MAE values. This way has been called as *macro-averaged* MAE. The former gives each instance an equal weight, and is possibly dominated by the system's performance on the data of a user who has the largest test set. The latter gives each user an equal weight instead. Both methods can be informative; therefore we present the evaluation results in both metrics.

We also applied a paired t-test on macro-averaged MAE and Wilcoxon signed rank test on micro-averaged MAE.

4.1.4 Results and Analysis

First of all, surprisingly, the state-of-the-art regression-based approach, SVOR, showed significantly worse performance than the performance of classification based approach, OB-MV, shown in Figure 5 and Table 2. The performance gap is not only significant but also it is statistically significant regardless of the types of significance test. It is evident that SVOR performance among machine learning models suggested that *one model assumption* did not hold on personalized email prioritization.

Second, we could validate the machine learning approaches significantly improve over baseline. In other words, we could make use of machine learning approach to improve the prediction performance of personal importance.

Third, among the classification methods, the evaluation results show that there are not many distinctions among classification based methods on Figure 6. However, OVA showed the worst performance except 30 trainings and others did notably better. Also our proposed order based approaches, especially OB-MV, showed the overall best performances among the classification approaches and the difference was statistically significant. We conjecture that order based approaches could take advantages of the partial order relations. Between DAG and OVO, DAG showed significantly better statistically but it was on limited ranges.

Suppose that we might have very limited amount of training data (less than 30 messages) and we might not be sure about *one model assumption*, we might use OVA. However, we may want to try order-based DAGs when we have more emails available. If we have to choose from popular classification-based approaches, then DAGs are a good choice given enough amount of training email messages.

4.2 Benchmark Experiments

4.2.1 Dataset and Experimental Setups

Our next research question was whether our proposed order-based approaches would work well or not on a benchmark dataset. Therefore, we tested order-based approaches along with other approaches to ordinal regression benchmark datasets generated from UCI dataset [1][1]. [1] used two collections of datasets but we tested only one of them because the size of the other collection was too small to test different training set size. The datasets were normalized to be zero mean and unit variance for each feature. The response variable was split into 10 ordinal levels using equal-size binning. Note that this procedure will satisfy ordinal regression assumption but does not guarantee fixed equal distance assumption. We randomly selected training from 25 instances to 300 instances by 25 increments and then tested on the remaining. The training and testing splits were repeated 100 times independently. Table 3 summarizes datasets and their statistics.

[1]http://www.gatsby.ucl.ac.uk/ chuwei/ordinalregression.html

Table 3: UCI Ordinal Regression Benchmark Dataset Statistics

Data Sets	Features	Instances
Bank Domains(1)	8	8192
Bank Domains(2)	32	8192
Computer Activities(1)	12	8192
Computer Activities(2)	21	8192
California Housing	8	15640
Census Domains(1)	8	16784
Census Domains(2)	16	16784

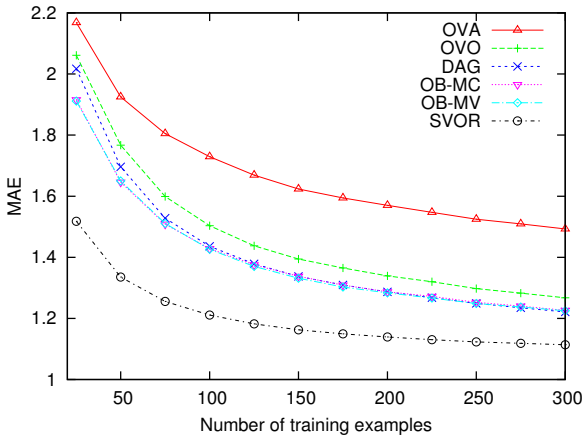
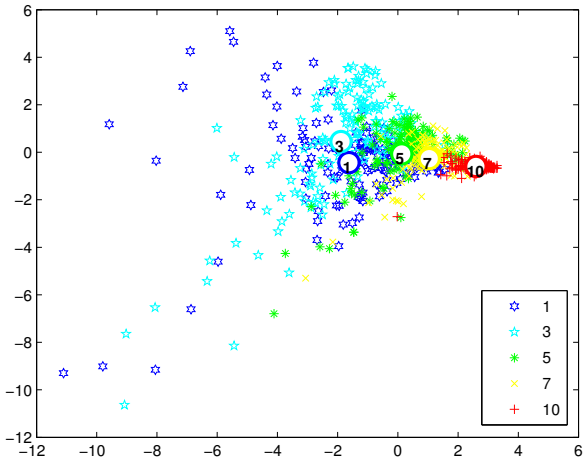


Figure 7: UCI 7 Dataset Average MAE Results

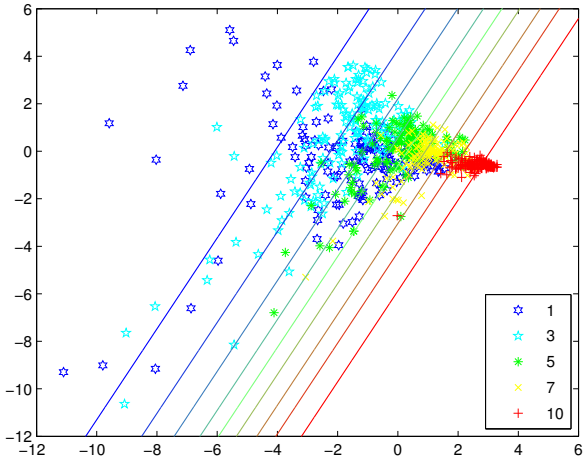
For classification-based approaches, we could not use SVM classifiers as our base classifiers due to the slow speed of SVM classifiers and thus we used Regularized Logistic Regression [22] due to its convergence properties and comparable accuracies and we got similar performance with regularized logistic regression performance compared to SVM classifier on benchmark datasets and [13] reported both of them showed similar performance. Again we tuned regularization parameter λ from 10^{-8} to 10^{-1}. We applied the same SVOR settings as in personalized email prioritization.

4.2.2 Results and Analysis

On the contrary to personalized email prioritization datasets, we got quite different results from UCI benchmark datasets, shown in Figure 7. First of all, SVOR showed the best performance regardless of training sizes and datasets and OVA showed the worst performance in most cases. As we observed in personalized email prioritization datasets, DAG is better than OVO. Order-Based DAGs showed better performance than DAG but the improvement is limited to the limited training size. With the limited amount of training data, order information was more helpful but with enough training data, DAG performance is similar to OB-DAG. The main

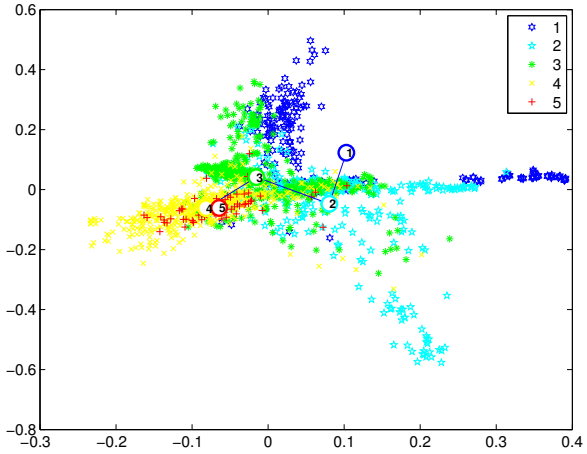


(a) PCA projection with Centroids

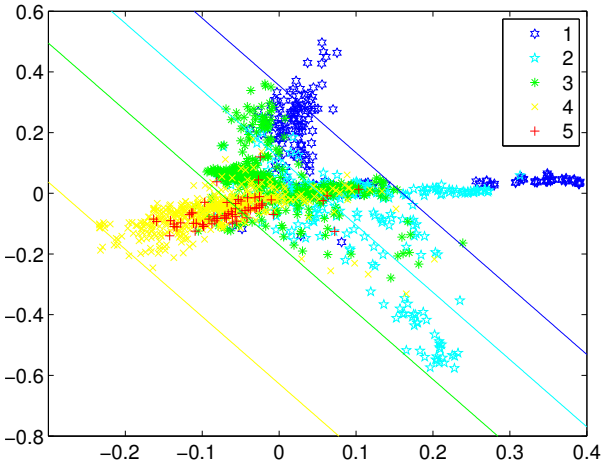


(b) PCA projection with Ordinal Regression Decision Hyperplanes

Figure 8: Computer Activities (2) on two the most correlated reduced dimensions with the response levels. The drawn lines are threshold for each ordinal levels and the fixed equal distance assumption do not hold here. Ordinal regression thresholds well captured different levels except level 1.



(a) PCA projection with Centroids



(b) PCA projection with Ordinal Regression Decision Hyperplanes

Figure 9: One user of email prioritization datasets was projected on two most correlated reduced direction with the response levels. The drawn lines are threshold for each ordinal levels. Ordinal regression thresholds captured different levels to some degree but it was not as good as CPU Activity (2).

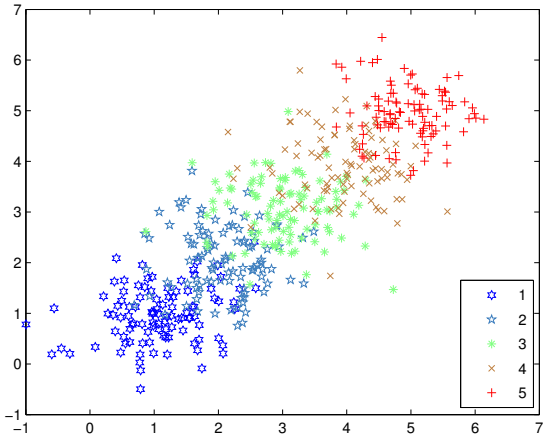
difference between personalized email prioritization datasets and UCI datasets is whether the datasets satisfy *one model assumption* or not, discussed in the rest of Experiments Section.

4.3 Principle Component Analysis

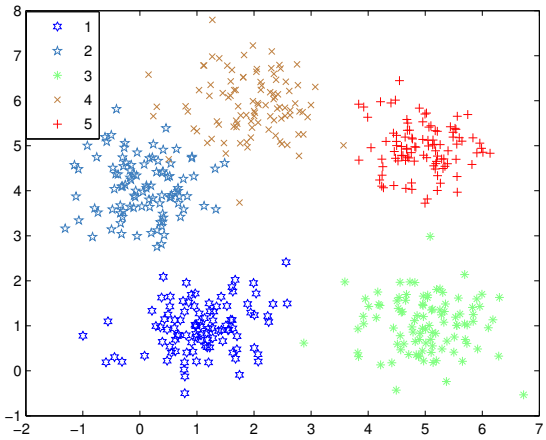
However, it was not clear why SVOR outperformed on certain datasets but it did not outperform on the email prioritization datasets. To answer this question, we used Principal Component Analysis (PCA), which is one of the most popular dimensionality reduction approaches. We projected Email Prioritization and UCI datasets onto the two most correlated reduced dimensions with the ordinal response variable by using Pearson Correlation Coefficients. Note that this projection should be the best projection for regression based approach. We also learned decision hyperplanes of SVOR models from the projected two dimensional datasets and drew decision hyperplanes in Figure 8 and 9.

Among seven ordinal regression benchmark datasets, we focused on the Computer Activities (2) dataset because the dataset well characterized ordinal regression conditions and with the same reason, we chose one user from email prioritization datasets. We observed the data distribution looks quite different. First, the centroids of Computer Activities (2) on Figure 8(a) were well aligned as a linear line according to the ordinal levels (except level 1), resulted in good alignment with SVOR decision hyperplanes compared to email prioritization datasets where the centroids were not well aligned as the linear line, so that we had better distribution for classification decision hyperplanes. We would like to point out that there were still partial ordinal relations from email prioritization datasets, which confirmed why our proposed order-based approaches worked better than other classification approaches.

In summary, this analysis tells us whether the dataset follows *one model assumption* or not. Computer Activities (2)

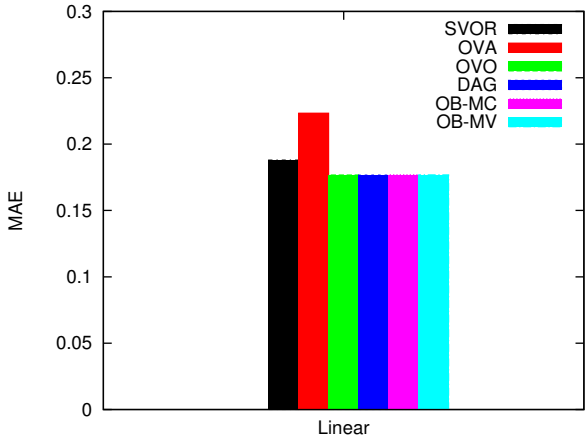


(a) Linearly Aligned Centroids on $y = x$

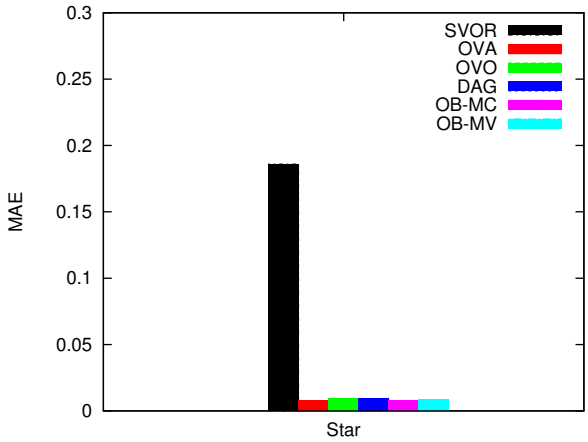


(b) Star-shaped Centroids

Figure 10: Two synthetic data generation conditions (Linear and Star)



(a) Linear Results



(b) Star Results

Figure 11: Experiment results of two synthetic data conditions

follows *one model assumption* pretty well, so that regression-based approaches outperformed classification based approaches. However the tested email prioritization dataset seemed not well fitted with *one model assumption*, which resulted in better classification performance.

4.4 Synthetic Experiments

4.4.1 Dataset and Experimental Setups

Although we reflected the correlations between reduced dimensions and the response variable on PCA, our two dimensional PCA analysis may not be perfect. Through our synthetic analysis experiments, we present that what we discovered is still valid on the controlled study.

We generated two dimensional Gaussian data distributions with the centroids on (1,1), (2,2), (3,3), (4,4) and (5,5) as shown in Figure 10(a). Note that it satisfies *one model assumption* and *fixed equal distance assumption* perfectly. To control the linearity of the centroid distribution, we shifted centroids from (2,2) to (0,4), from (4,4) to (2,6) and from (3,3) to (5,1), shown in Figure 10(b), which does not satisfy two regression assumptions. We repeated the above procedures 100 times independently and reported the average results. We apply the same evaluation strategy of UCI ordinal regression benchmark datasets to this synthetic datasets.

4.4.2 Results and Analysis

First of all, with linearly aligned centroids, SVOR did not show the better performance. However, SVOR showed better performance than OVA approaches. All classification approaches except OVA showed better performance than SVOR. But with more difficult cases (high signal-to-noise ratio), we observed SVOR showed better results than any other classification-based approaches, which was omitted due to limited space.

When the centroids are not linearly aligned, classification based approaches showed significantly better results than SVOR. Therefore, to be the best condition for SVOR, noisy and linearly aligned centroids are required, which is favorable for *one model assumption*.

4.5 Discussion

Although we presented only two simulation conditions for our synthetic data experiments, we could not test all possible conditions such as diverse training set size, different levels of ordinality, the skewed size distribution of different priority levels or different kinds of deviation conditions. However,

we present here the most interesting results, which may not be the best but we believe it still delivers what we would like to show in this paper.

One would wonder why we did not try other kinds of regression-based approaches such as Support Vector Regression [4], Gaussian Process Ordinal Regression (GPOR) [1] or classification approaches such as Half-against-half [12]. What we chose are state-of-the-art or the most popular approaches and we believe they will have representative characteristics. For instance, suppose that GPOR was slightly better than SVOR but it would not change our main observations.

Non-linear classifiers or additional features could improve each individual approach performance but they could not change our main observation too. For instance, our test results on personalized email prioritization showed that non-linear SVOR showed severely worse results than linear-SVOR but OVA classification results were slightly improved and again overall observation was not changed.

Due to the difficulties of data collection procedures and limited data availability, our collected dataset could not reveal long-term relations such as topic or priority drifting over time here. It is an interesting topic to be investigated.

5. RELATED WORK

Pang and Lee [17] tried both OVA classification and Support Vector Regression approaches in sentiment ordinal regression problem but they also applied a simple form of re-ranking to the output of classifiers or regressions. They reported regression approaches were better for four levels but three levels preferred OVA classification and the re-ranking showed consistent improvements. Unfortunately they evaluated on only four users of Internet movie reviews and as we pointed out OVA is worst choice for classification and SVOR is better than SVR [2] due to fixed equal distance assumption. Therefore, it is hard to generalize their observation but applying re-ranking to ordinal regression might have potential to improve ordinal regressions.

Frank and Hall [6] tried to model cumulative odds of general ordinal regression by decision tree classifiers. We believe it was the first approach to handle ordinal regression problems by using classification-based approaches. However, when we tested this on our personal email prioritization datasets, it showed quite poor performance and most predictions directed to the highest or lowest priority due to its strong bias of this model assumption. Therefore, we did not include this as one of our baseline approaches.

Qin et al. [19] share the same line of thought with ours in ranking approaches. They tried multiple models instead of only one model in ranking algorithm. They also observed that ranking dataset is not satisfying one model assumption and proposed a model based on multiple models and how to aggregate multiple model results.

There are some works which utilize decision tree structures [10, 12, 5, 20, 16]. Since there are so many ways to build tree structures, most of them, [10, 12, 5, 16] utilize a form of clustering algorithms at each level. Most of them start from one root node and split it into two classes and keep splitting until there are only two classes left. However, their main concern is scalability or speeding up of multi-class classification instead of robustness of classification. In our proposed order based DAGs, we tried all possible decision paths at each level and resulted in more robust estimation with the assumption of ordinality of classification data.

6. CONCLUSIONS

Personalized email prioritization requires effective mapping from a high-dimensional input feature space to ordinal output variables. We presented a comparative study of two types of supervised learning approaches: ordinal regression-based and classification-based approaches, including a classifier cascade. Our conceptual analysis and empirical evaluations show that the effectiveness of ordinal-regression based methods crucially depend on the separability of priority classes by parallel hyperplanes, which may be too restrictive for personalized email prioritization. Classification-based methods, on the other hand, offer more general and robust solutions when complex decision boundaries are needed because they allow multiple non-parallel hyperplanes as decision functions. With the proposed OB-MV and OB-MC schemes, we effectively combine the outputs of different binary classifiers into email priority predictions, yielding significant improvements over the results of SVOR, a state-of-the-art method among ordinal-regression based approaches. Our experiments with synthetic datasets and ordinal-regression benchmark datasets further support our conclusions, and provide additional insights regarding when regression-based methods work better and when classification-based methods work better.

7. REFERENCES

[1] W. Chu and Z. Ghahramani. Gaussian processes for ordinal regression. *Journal of Machine Learning Research*, 6:1019–1041, 2005.

[2] W. Chu and S. S. Keerthi. New approaches to support vector ordinal regression. In L. D. Raedt and S. Wrobel, editors, *Machine Learning, Proceedings of the Twenty-Second International Conference (ICML 2005), Bonn, Germany, August 7-11, 2005*, volume 119 of *ACM International Conference Proceeding Series*, pages 145–152. ACM, 2005.

[3] P. J. Denning. ACM President's letter: Electronic junk. *Communications of the ACM*, 25(3):163–165, 1982.

[4] H. Drucker, C. J. C. Burges, L. Kaufman, A. J. Smola, and V. Vapnik. Support vector regression machines. In M. Mozer, M. I. Jordan, and T. Petsche, editors, *NIPS*, pages 155–161. MIT Press, 1996.

[5] B. Fei and J. Liu. Binary tree of svm: a new fast multiclass training and classification algorithm. *IEEE Transactions on Neural Networks*, 17(3):696–704, 2006.

[6] E. Frank and M. Hall. A simple approach to ordinal classification. In *EMCL '01: Proceedings of the 12th European Conference on Machine Learning*, pages 145–156, London, UK, 2001. Springer-Verlag.

[7] T. Hasegawa and H. Ohara. Automatic priority assignment to E-mail messages based on information extraction and user's action history. In R. Loganantharaj and G. Palm, editors, *Intelligent Problem Solving, Methodologies and Approaches, 13th International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems, IEA/AIE 2000, New Orleans,*

Louisiana, USA, June 19-22, 2000, Proceedings, volume 1821 of *Lecture Notes in Computer Science*, pages 573–582. Springer, 2000.

[8] E. Horvitz, A. Jacobs, and D. Hovel. Attention-sensitive alerting. In K. B. Laskey and H. Prade, editors, *UAI '99: Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence, Stockholm, Sweden, July 30-August 1, 1999*, pages 305–313. Morgan Kaufmann, 1999.

[9] C.-W. Hsu and C.-J. Lin. A comparison of methods for multi-class support vector machines. *IEEE Transactions on Neural Networks*, 13:415–425, 2002.

[10] J. Kittler and F. Roli, editors. *Multiple Classifier Systems, Second International Workshop, MCS 2001 Cambridge, UK, July 2-4, 2001, Proceedings*, volume 2096 of *Lecture Notes in Computer Science*. Springer, 2001.

[11] B. Klimt and Y. Yang. The Enron corpus: A new dataset for email classification research. In J.-F. Boulicaut, F. Esposito, F. Giannotti, and D. Pedreschi, editors, *Machine Learning: ECML 2004, 15th European Conference on Machine Learning, Pisa, Italy, September 20-24, 2004, Proceedings*, volume 3201 of *Lecture Notes in Computer Science*, pages 217–226. Springer, 2004.

[12] H. Lei and V. Govindaraju. Half-against-half multi-class support vector machines. In N. C. Oza, R. Polikar, J. Kittler, and F. Roli, editors, *Multiple Classifier Systems*, volume 3541 of *Lecture Notes in Computer Science*, pages 156–164. Springer, 2005.

[13] F. Li and Y. Yang. A loss function analysis for classification methods in text categorization. In *Proceedings of ICML-03, 20th International Conference on Machine Learning*, Washington, DC, 2003. Morgan Kaufmann Publishers, San Francisco, US.

[14] R. Likert. A technique for the measurement of attitudes. *Archives of Psychology*, 140:1–55, 1932.

[15] Y. Liu, Y. Yang, and J. G. Carbonell. Boosting to correct inductive bias in text classification. In *CIKM*, pages 348–355. ACM, 2002.

[16] G. Madzarov, D. Gjorgjevikj, and I. Chorbev. A multi-class svm classifier utilizing binary decision tree. *Informatica*, 33(2):233–241, 2009.

[17] B. Pang and L. Lee. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *ACL*. The Association for Computer Linguistics, 2005.

[18] J. C. Platt, N. Cristianini, and S. J. Taylor. Large margin DAGs for multiclass classification. In S. A. Solla, T. K. Leen, and K. R. Müller, editors, *Advances in Neural Information Processing Systems*, volume 12. MIT Press, 2000.

[19] T. Qin, X.-D. Zhang, D.-S. Wang, T.-Y. Liu, W. Lai, and H. Li. Ranking with multiple hyperplanes. In W. Kraaij, A. P. de Vries, C. L. A. Clarke, N. Fuhr, and N. Kando, editors, *SIGIR*, pages 279–286. ACM, 2007.

[20] A. Ramanan, S. Suppharangsan, and M. Niranjan. Unbalanced decision trees for multi-class classification. In *International Conference on Industrial and Information Systems, 2007. ICIIS 2007.*, pages 291–294, Aug. 2007.

[21] J. B. Spira and D. M. Goldes. Information overload: We have met the enemy and he is us, 2007. Basex Inc.

[22] Y. Yang, S. Yoo, J. Zhang, and B. Kisiel. Robustness of adaptive filtering methods in a cross-benchmark evaluation. In R. A. Baeza-Yates, N. Ziviani, G. Marchionini, A. Moffat, and J. Tait, editors, *SIGIR*, pages 98–105. ACM, 2005.

[23] S. Yoo, Y. Yang, F. Lin, and I.-C. Moon. Mining social networks for personalized email prioritization. In J. F. E. IV, F. Fogelman-Soulié, P. A. Flach, and M. J. Zaki, editors, *KDD*, pages 967–976. ACM, 2009.

8. ACKNOWLEDGMENT

This work is supported in parts by the Defense Advanced Research Project Agency (DARPA) under contract NBCHD030010, by the National Science Foundation (NSF) under grant IIS_0704689 and by Department of Energy under contract DE-AC02-98CH10886. Any opinions, findings, conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the sponsors.