Coarse-to-Fine, Cost-Sensitive Classification of E-Mail

Jay Pujara
jay@cs.umd.edu

Lise Getoor
getoor@cs.umd.edu

Abstract

In many real-world scenarios, it is necessary to make judgments at differing levels of granularity due to computational constraints. Particularly when there are a large number of classifications that must be done in a real-time streaming setting and there is a significant difference in the time required to acquire different subsets of features, it is important to have an intelligent strategy for optimizing classification accuracy versus computational costs. Accurate and timely email classification requires trading off the classification granularity with the feature acquisition costs. To solve this problem, we introduce a *Granular Cost-Sensitive Classifier (GCSC)* which modulates the cost of feature acquisition with the granularity of the classification, allowing inexpensive classification at a coarse level and more costly classification at finer levels of granularity. Our approach can classify messages with greater accuracy while incurring a lower feature acquisition cost relative to baseline classifiers that do not make use of cost information.

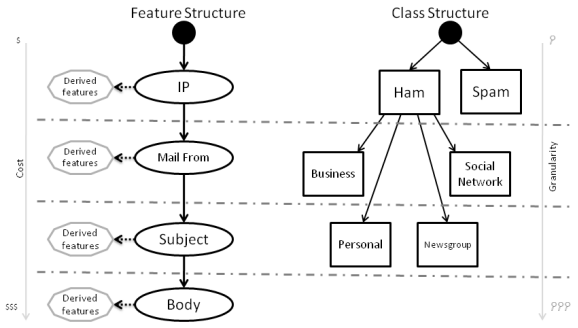
1 Introduction

Electronic mail has become an integral part of daily communication, filling needs ranging from the delivery of financial statements to personal correspondence. The huge volume, and multifarious uses of email have an impact on users, who must dedicate time to organize and categorize the influx of messages. The popularity of the medium also requires service providers to operate at extremely large scale, handling billions of messages, totaling terabytes of data, each day.

This immense scale provides correspondingly outsized computational challenges. For example, the vast majority of attempts to send e-mail consist of unwanted marketing messages ("spam"), which providers must reject or place in a separate folder. Moreover, webmail systems are increasingly attempting to use more involved processing to make messages easier to organize and provide convenience functions. Examples of this functionality include detecting meeting invitations, creating slideshows from attached photographs, annotating metadata from social network profiles, recognizing shipment tracking information, and predicting relevancy to a user. Because only a subset of these processing steps are necessary for a given message, providers need to understand the type of message being sent.

In these scenarios, the goal of a mail system is to classify a message into a category and take an appropriate action. These categories occur at differing scales with a defined structure: at a coarse scale messages are considered undesirable ("spam") or desirable ("ham"), while desirable messages can be further classified at a finer scale as "business communication" or "social network notifications". Messages containing "business communication" may possibly be "shipment notifications" and as a result are candidates for extracting package tracking information.

This multi-level classification task is not suitable for conventional methods, which rely on static feature vectors or full-text classification of each message. Because so many message deliveries are attempted, mail systems cannot consider each message in its entirety before deciding the disposition of a message; fast decisions must be made using a restricted feature set. Additionally, mail systems



(a)

Command state	Associated features
<Connect>	$\phi_1 \rightarrow$ IP address of remote host
HELO	Identity of remote host
MAIL FROM	$\phi_2 \rightarrow$ Sender of message
RCPT TO	Recipient of message
DATA	$\phi_3 \rightarrow$ Message incl. headers

(b)

Figure 1: (a) Relationship of feature cost and class granularity (b) SMTP Commands

must be sensitive to load and malicious attacks, foregoing more computationally demanding features in favor of making many decisions. Given some of the complicated enhancements mail providers apply to messages, eliminating those that are not germane to a message early in its delivery process can result in a significant decrease in computational costs.

Fortunately, e-mail messages also adhere to a protocol that allows features to be acquired incrementally, at differing costs. The structure imposed by feature costs and dependencies provides a compelling parallel to the structure of message categories. Relating these two structures provides the promise of making coarse-to-fine category judgments (Figure 1). Ideally, cheap features can be used to make coarse judgments and progressively expensive features can be used for classification at a finer level. By using cost-sensitive methods that emphasize incremental acquisition of features on the basis of acquisition cost, the ability to make granular classifications becomes a more computationally tractable problem.

2 Approach

Problem Description We consider a classification setting, where the goal is to learn a mapping from instances $\mathbf{X}$ to classes $\mathbf{Y}$, $h : \mathbf{X} \rightarrow \mathbf{Y}$. Here, each instance $\mathbf{x}$ is described by a set of features $\langle \phi_1(\mathbf{x}) \ldots \phi_n(\mathbf{x}) \rangle$, and each label Y comes from a set of p class labels, $\{y_1 \ldots y_p\}$. Given training data $\mathcal{D} = \{\langle \mathbf{x}_1, y_1 \rangle, \ldots, \langle \mathbf{x}_m, y_m \rangle\}$, we attempt to learn a hypothesis h which minimizes a loss function that we will describe shortly.

Two special properties of the classification problem we propose are the structure of features and the structure of classes. Features often have an acquisition ordering structure imposed by natural dependencies or artificial protocols in the domain. For example, determining whether a circuit is functioning correctly requires establishing that current is properly flowing through the components before testing each component, and in processing email the sender of a message always precedes the body. These feature acquisition dependencies can be formalized and are related to the cost of feature acquisition. We say that a feature $\phi_j \prec \phi_i$, if, in order to acquire ϕ_i, we must have already acquired ϕ_j. For the purposes of this paper, we assume a simple linear order over the features acquired, i.e., $\phi_1 \prec \phi_2 \prec \ldots \prec \phi_n$. We use $\Gamma^k = \{\phi_1, \ldots, \phi_k\}$ to denote the set of features which must be acquired in order to acquire feature ϕ_k and we use $C_\phi(\Gamma^k)$ to denote the cost of acquiring feature set Γ^k (we assume that the cost does not depend on the particular instance). We refer to Γ^k as "feature level k."

Similarly, class labels exist in a hierarchy and have dependencies which can influence misclassification costs. We assume a hierarchy over the class labels $y_1, \ldots, y_p$. We use $y_j < y_i$ to denote that label y_j is an ancestor of y_i in the classification hierarchy. We use $C_Y(y_i, y_l)$ to denote the cost of misclassifying an instance with true class label y_i as y_l. We assume, for example, that $C_Y(y_i, y_j) \leq C_Y(y_i, y_l)$ when $y_j < y_i$ and $y_l \not< y_i$.

Our goal is to learn a model, h that minimizes both misclassification cost (C_Y) and feature acquisition cost (C_ϕ). For any instance x_i, we may acquire a different subset of features, which we denote

Γ_i^k. As a result, our hypothesis operates over incomplete instances. To denote the classification result of the hypothesis for instance $\mathbf{x}_i$ over a subset of features Γ_i^k we write $h(\mathbf{x}_i, \Gamma_i^k) = \widehat{y_{ik}}$.

We introduce a loss function, $L(h, \mathbf{x}_i, \Gamma_i^k)$. This loss function depends on both misclassification cost and the feature acquisition cost for instance $\mathbf{x}_i$. We define the loss function as follows:

$$L(h, \mathbf{x}_i, \Gamma_i^k) = \alpha(C_\phi(\Gamma_i^k)) + (1 - \alpha)(C_Y(y_i, h(\mathbf{x}_i, \Gamma_i^k)))$$

where $0 \leq \alpha \leq 1$. Here the α is a parameter that modulates the contribution of the misclassification cost and feature cost to the total loss. An interesting feature of this loss function is that increasing the influence of misclassification cost will decrease the influence of feature acquisition cost.

Granular Cost-Sensitive Classifier Given this problem setting, how can we learn a classifier which minimizes both the cost of feature acquisition and misclassification? A broad intuition is that the classifier must know when it needs more information, that is the classifier must choose the appropriate feature level for each instance such that the decrease in expected misclassification cost outweighs the cost of acquiring features. A general solution is to define a measure of uncertainty and acquire additional features for instances where the classifier has a high uncertainty measure. Our approach is to train a classifier which can provide a decision margin that serves as a measure of the distance from a given instance to the decision boundary. We use $M(h, \mathbf{x}_i, \Gamma_i^k)$ to denote the margin for a particular instance $\mathbf{x}_i$ when features Γ_i^k are acquired. By using this margin as a measure of uncertainty, we can decide which instances would benefit from the acquisition of additional features. This goal is achieved by learning the optimal margin, m_k, for each Γ^k, where acquiring features when $M(h, \mathbf{x}_i, \Gamma_i^k) < m_k$ minimizes the total loss. To understand the loss for a given instance $\mathbf{x}_i$ and candidate margin m_k, we introduce a cost-sensitive loss, L_c:

$$L_c(h(\mathbf{x}_i)) : \begin{cases} L(h, \mathbf{x}_i, \Gamma_i^k) & \text{if } M(h, \mathbf{x}_i, \Gamma_i^k) > m_k; \\ \min_{r>k} L(h, \mathbf{x}_i, \Gamma_i^r) & otherwise \end{cases}$$

When the margin is greater than m_k, the loss is based on using Γ_i^k to provide the classification result. If the margin is below m_k, we choose the best loss across all feature levels greater than k. Learning the margin m_k that minimizes L_c for each feature level k, we can classify instances with incremental feature acquisition using Algorithm 1.

Algorithm 1 Cost-Sensitive Classification Algorithm

Given instance X_i and learned margins $(m_0 \ldots m_{n-1})$
$k \leftarrow 0$
repeat
  $k \leftarrow k + 1$
  $\Gamma_i^k \leftarrow \Gamma^k$
  $\widehat{y_{ik}} = h(\mathbf{x}_i, \Gamma_i^k)$
until $M(h, \mathbf{x}_i, \Gamma_i^k) > m_k$ or $k = n$
return $\widehat{y_{ik}}$

One motivation of cost-sensitive classification is to provide a framework to use increasingly costly features when attempting to predict more granular labels. This is accomplished by adjusting the parameter α of the loss function for tasks of different granularity. For example, we can choose $\alpha = \alpha_c$ when performing a coarse classification task, and $\alpha = \alpha_f$ when performing a fine-grained classification task. By choosing a relatively high value for α_c, we can favor low-cost solutions over granular labels. As candidates are rejected in the coarse task, we can use a lower value for α_f to produce more accurate class predictions. In general, for each level of a class hierarchy we can differentially choose α to modulate the importance of misclassification cost and feature acquisition cost due to computational constraints.

3 Experimental Evaluation

E-Mail Domain E-mail is generally delivered to Mail Transfer Agents (MTAs) using the Simple Mail Transfer Protocol (SMTP)[1]. This protocol defines a conversation consisting of a well-ordered

Class	Count
Spam	531
Business	187
Social Network	233
Newsletter	174
Personal/Other	102

(a) Message Categories, Counts

Feature	C_n	C_s	C
ϕ_1	0	5.035	.168
ϕ_2	3	6.640	.322
ϕ_3	8	7.299	.510

(b) Feature Classes and Costs

Figure 2: Description of Mail Dataset

set of commands (Figure 1). This well-ordered set of commands corresponds to the strongly ordered features, $\phi_1 \ldots \phi_n$ considered in the problem description. After each command the MTA must send a response code, indicating whether the command was successful or resulted in an error, and update its internal state.

The first piece of information available to the MTA is the IP address of the remote sender, which arrives as soon as the sender connects. As the conversation continues, the sender will send the "MAIL FROM" command and provide an e-mail address. The sender must provide one or more recipients using the "RCPT TO" command. Finally, the sender will enter the "DATA" state and send the message. Usually the message will consist of headers and content. The headers contain metadata about the message, including routing information as well as items such as the date and time the message was sent, the sender's name, and the subject.

The structure of this conversation lends itself to coarse-to-fine processing. The IP address can differentiate between known hosts with a history of sending messages of a specific type and unknown hosts. In particular, known senders of spam can be given an error after the first command (e.g., after acquiring the first level features), effectively blocking the sender. Given the vast quantities of spam that flow through networks, the ability to use few features and quickly reject undesirable messages can provide a significant efficiency gain. Many major senders also use different sending addresses or special metadata in the header that allows refinement of class categories such as "shipment notification" or "shared photographs" in advance of receiving the complete message.

Dataset We evaluated on data sampled from a month-long time window of e-mail data exchanged by users of Yahoo! Mail. To create a tractable experiment, the feature set was limited to four types of features: remote IP address, sender mailfrom domain, sender mailfrom address, and tokens from the message subject. We representatively sampled approximately 100 feature values from each of the four feature types, yielding 432 features. We then randomly sampled messages from the same month of e-mail data, restricting the sample to messages containing at least one of the selected features values. For each of the 432 features, up to three messages were randomly selected, yielding a total 1227 messages. The full feature vectors, encompassing all possible IP, sender and subject features found in the 1227 training messages, were constructed for these messages.

Sampled messages were categorized into five categories: "business", "social network", "newsletter", "personal/unknown" and "spam". The labels were generated editorially by a human expert. The frequency of each category is shown in Figure 2(a).

Misclassification and Feature Costs For misclassification cost, we used a basic (0,1) loss function. For cost at different feature levels, we defined three feature sets, IP features (ϕ_1), sender features (ϕ_2), and subject features (ϕ_3). Each of these feature sets was attributed a cost, C (Figure 2(b)). Costs were normalized to their fractional share of the cost of the entire feature vector, so that acquiring all features for a message corresponds to incurring a cost of 1. Two factors were considered when computing the cost of a feature: the network traffic necessary to acquire the feature (C_n) and the storage required to hold all possible feature values (C_s). In practice these costs can be weighted by their computational impact; here we weighted them equally. The normalized combination of these two costs is shown as C. Since MTAs communicate with senders via TCP, the network cost can be quantified in packets exchanged with the sender. The storage cost of features was computed by calculating the entropy of feature values in the sampled messages. The entropy represents the number of bits needed to optimally encode the feature values, and this is the best-case scenario for storing feature information.

Table 1: Baseline results for misclassification and feature acquisition costs, comparing progressive feature classes and Granular Cost-Sensitive Classifier, 10-fold cross validation

Classifier, Features	Coarse C_Y	Fine C_Y	Overall C_Y	Overall C_ϕ
NB, Γ^1	.104	.244	.185	.168
SVM, Γ^1	.153	.305	.242	.168
GCSC, $\alpha_c = .7, \alpha_f = .1$	.100	.207	.162	.206
NB, Γ^2	.098	.214	.164	.490
SVM, Γ^2	.125	.247	.167	.490
GCSC, $\alpha_c = .3, \alpha_f = .05$	.091	.174	.141	.479
NB, Γ^3	.090	.176	.144	1
SVM, Γ^3	.097	.196	.150	1
GCSC, $\alpha_c = .15, \alpha_f = .01$	.088	.175	.140	.511

Classifier Implementation and Baseline To evaluate the performance of our cost-sensitive approach, a baseline classifier was trained on the full feature vector as well as progressive sets of features at each cost level. The baseline classifier was used to evaluate the cost and accuracy on a coarse learning task, differentiation between spam messages and non-spam messages. Messages judged non-spam were then used in a fine-grained task of predicting a message category. Results from two baseline classifiers are presented: a naive Bayes classifier (NB) and a multiclass SVM implementation (SVM) [2]. The classifiers were trained using all features, as well as the set of features at each cost level. Evaluation was conducted using 10-fold cross-validation.

The baseline naive Bayes classifier was then augmented to implement the granular cost-sensitive algorithm discussed earlier. Specifically, the classifier margin was expressed as the ratio of the probabilities of the two most likely classes. For each feature level, the margin and loss were calculated for all training instances. These values were used to learn optimal margins for the IP and sender classes. At evaluation time, additional features were acquired only if the decision margin was below the optimal margin learned on the training data.

Results An important consideration in results is whether the classifier could complete a coarse-to-fine learning task and minimize misclassification cost and feature acquisition cost. We compare the results of the Granular Cost-Sensitive Classifier to the baseline classifiers for these two metrics in Table 1. When considering misclassification cost, the cost-sensitive classifier was able to achieve a significantly lower C_Y relative to the baseline classifier at feature level Γ^2 without any significant increase in feature acquisition cost ($p = .05$). When considering feature acquisition cost, the cost-sensitive classifier was able to obtain a significantly lower C_ϕ compared to the baseline classifier using feature level Γ^3 without any significant increase in misclassification cost ($p = .05$).

These results show how the choice of α parameters can manage tradeoff between cost and accuracy. Higher values of α result in the acquisition of fewer features, allowing the classifier to surpass the baseline classifier in feature cost. Lower values of α allow the classifier to acquire more features, which let the classifier succeed in producing fewer misclassifications. These results depend on the delicate relationship between α_c and α_f; choosing a higher α_c can result in misclassifications at the coarse level, which limits the ability of the classifier when making fine judgments. Lower values of α_c require paying a high price for each coarse classification, but allow successful classification at a fine level with no additional feature acquisition. The tradeoff presented by the α parameter and interplay between α_c and α_f is seen in Figure 3.

4 Related Work

Many of the ideas present in this work have been considered separately. Computational cost of classification has been reduced following the approach of [3]: instances are initially processed through computationally efficient classifiers for a coarse judgment and those instances judged interesting during the coarse classification are reconsidered with increasingly computationally demanding models. However, these approaches generally do not relate cost to a hierarchical multiclass classification, nor do they focus on feature costs or interdependencies in their analysis. The problem of hierarchical text classification as initially presented in [4] supports feature sets tailored to the classification task at a particular level. Subsequent work has considered hierarchical classification of e-mail [5].

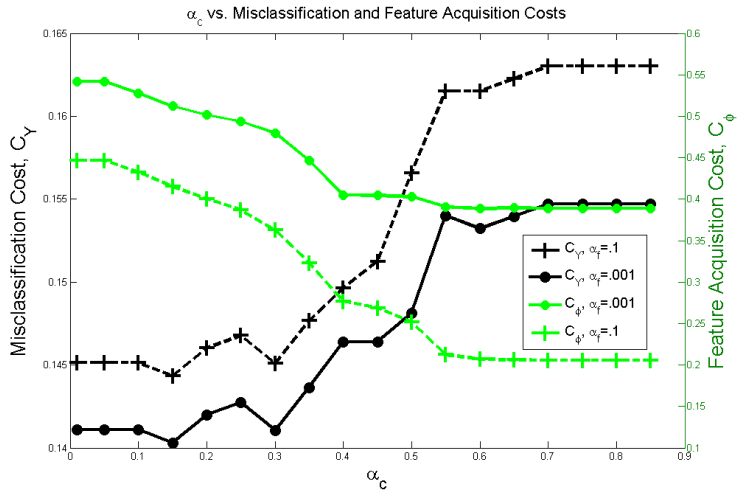


Figure 3: This figure shows the relationship of α_c to misclassification and feature acquisition costs for $\alpha_f = .1$ and $\alpha_f = .001$. As α_c is increased, misclassification cost increases and feature acquisition cost decreases. While the costs for differing values of α_f are similar for low values of α_c, at high values of α_c, the choice of α_f plays a large role in optimizing feature acquisition cost relative to misclassification cost

No work to our knowledge has approached the selection of feature sets in the hierarchy from the perspective of feature costs. Finally, the question of actively acquiring features to improve classifier performance both during training [6] and at test time [7] has been studied, but has not explored the relationship between differing levels of granularity and feature cost. The contribution of this paper, then, is the synthesis of these facets: applying an active feature acquisition model to a hierarchically structured output with particular attention to the relationship of feature cost and the granularity of output.

5 Conclusion

This work presents the Granular Cost-Sensitive Classifier which allows the ability to make both coarse judgments while incurring a low feature-acquisition cost and more granular judgments while dedicating more resources to feature acquisition. A surprising benefit is that cost-motivated feature selection produces superior results; the framework is able to decrease both misclassification cost and feature acquisition cost relative to cost-insensitive baseline classifiers.

References

[1] J. Postel. RFC 821: Simple mail transfer protocol, August 1982.

[2] Thorsten Joachims. Svm-light support vector machine. `http://svmlight.joachims.org/svm\_multiclass.html`, 2008.

[3] P Viola and M Jones. Rapid object detection using a boosted cascade of simple features. *Computer Vision and Pattern Recognition*, 1:511–518, 2001.

[4] D Koller and M Sahami. Hierarchically classifying documents using very few words. In *Proceedings of ICML*, pages 170–178, 1997.

[5] Julia Itskevitch. *Automatic Hierarchical E-Mail Classification Using Association Rules*. PhD thesis, Belorussian State Polytechnic Academy, 2001.

[6] P. Melville, M. Saar-Tsechansky, F. Provost, and R. Mooney. Active feature-value acquisition for classifier induction. In *Proceedings of ICDM*, pages 483–486. IEEE, 2004.

[7] X Chai, L Deng, Q Yang, and C X Ling. Test-cost sensitive naive bayes classification. In *Proceedings of ICDM*, pages 51–58, 2004.