Multiclass-Multilabel Learning when the Label Set Grows with the Number of Examples

Ofer Dekel
Microsoft Research
oferd@microsoft.com

Ohad Shamir
The Hebrew University
ohadsh@cs.huji.ac.il

November 2009

Technical Report
MSR-TR-2009-163

We discuss multiclass-multilabel classification problems in which the set of candidate labels is extremely large. Most existing multiclass-multilabel learning algorithms expect to observe a statistically significant sample from each class, and fail if they receive only a handful of examples per class. We propose and analyze the following two-stage approach: first use an arbitrary (perhaps heuristic) classification algorithm to construct an initial classifier, then apply a simple but principled method to augment this classifier by removing harmful labels from the label set. A careful theoretical analysis allows us to justify our approach under some reasonable conditions (such as label sparsity and power-law distribution of label frequencies), even when the training set does not provide a statistically significant representation of most classes. Surprisingly, our theoretical analysis continues to hold even when the number of labels exceeds the sample size. We demonstrate the merits of our approach on the ambitious task of categorizing the entire web using the 1.5 million categories defined on Wikipedia.

Microsoft Research
Microsoft Corporation
One Microsoft Way
Redmond, WA 98052
http://www.research.microsoft.com

1 Introduction

In multiclass-multilabel classification, our goal is to assign one or more labels, from a given set of k labels, to instances in some domain. An example of a multiclass-multilabel problem is document categorization, which is the problem of assigning one or more topics to each document in a corpus (e.g. [11]). Multiclass-multilabel problems are also abundant in other fields, such as computer vision [5] and computational biology [3]. If a training set of m labeled examples is available, a multiclass-multilabel classifier can be learned using a supervised machine learning algorithm. Typically, learning algorithms for multiclass-multilabel problems are developed and analyzed under the assumption that k is held constant as m grows. In this paper, we consider a different version of the multiclass-multilabel problem, where the label set grows with the number of examples (i.e. $k \geq \Omega(m)$). For example, this situation occurs when the set of labels is a so-called *Folksonomy*, a set of labels that emerges from a collaborative tagging or a social tagging scheme.

The concrete problem that motivates this work is the problem of categorizing the entire web using the set of categories defined by the Wikipedia website. At the very bottom of every Wikipedia article there is a short list of categories, and we define our label set to be the union of these lists. The Wikipedia articles themselves can be used as training examples, since they are labeled web pages. When new articles are added to Wikipedia, they often introduce new categories. As of today, Wikipedia contains 2.9 million articles and almost 1.5 million categories.

The Internet provides many other examples where k grows with m. For instance, photo sharing websites allow users to annotate their photos with keywords. The implied classification task is to recommend keywords whenever new photos are uploaded. Assuming that the site does not impose any restrictions on the keywords that may be used, the set of distinct keywords is likely to grow as more and more photos are uploaded to the site.

For such datasets, applying standard techniques is problematic for two reasons: First, since k scales with m, many labels occur only a handful of times in the training set, so we do not have a statistically significant sample from each class; Second, the concrete values of m and k we deal with are very large, to the point that most existing multiclass-multilabel learning algorithms become computationally intractable. For example, the most common approach to multiclass-multilabel learning is to train a separate binary classifier for each class. For the datasets we have in mind, such an approach is both statistically untenable (due to the small number of examples per class) and computationally impractical (as it requires maintaining millions of hypotheses for all the classes). Similar problems are encountered for other standard approaches, such as those based on ranking (e.g. [2, 6, 9]).

In practice, the only realistic solution is to turn to much simpler classification algorithms, such as nearest neighbor methods. For example, consider once again the problem of categorizing the entire web using Wikipedia categories, and assume that we have access to the log of an Internet search engine. We can use the log to construct a *click graph*, a bipartite graph whose vertices include all web pages and all queries ever issued to the search engine. An edge is drawn between a query Q and a web page W if enough users issued the query Q and then clicked on W. We can use the

graph distance induced by the click graph to define a metric over web pages. Given a set of labeled Wikipedia pages, we can label the entire web using a nearest neighbor type algorithm over this metric. In other words, labels are propagated from the labeled Wikipedia pages along the edges of the click graph to the rest of the web. Such simple algorithms can be implemented in a way that is almost entirely insensitive to the size of the label-set, but their simplicity often comes at the cost of lower classification accuracy.

One way to improve the accuracy of a simple classification algorithm is to add a separate post-learning step. Taking such a two-stage approach is also common in other areas of machine learning. For example, in ranking problems, it is common to run a simple and fast algorithm to obtain an initial ranking, and then to run a more accurate re-ranking algorithm only on the top few results. In this paper, we focus on a post-learning step that modifies a classifier by pruning certain labels from its output. In other words, the original classifier outputs a set of labels, and the post-learning step deletes labels from this set. The intuition behind this approach is that in such massive multiclass-multilabel datasets, many labels are inherently noisy and hard to learn, and attempting to predict them decreases the overall accuracy of our classifier.

We propose and analyze, both theoretically and experimentally, a simple label-pruning method. The method is based on comparing the number of true-positives and false-positives of each predicted label, and discarding labels where their ratio exceeds a certain threshold. Returning to the example of categorizing the web, the initial nearest neighbor algorithm is likely to find that many web pages about classical composers turn out to be close neighbors of the Wikipedia article on *Mozart*. The nearest neighbor classifier indiscriminantly assigns all of the Wikipedia categories that are associated with the article on *Mozart* to all of these pages. One of these categories is *People Born in 1756*, which is likely to have many false positives across the validation set. Intuitively, the label *People Born in 1756* is incompatible with the click-graph based metric we have chosen, namely, our metric is unlikely to put different web pages from this class in close proximity to each other. In this situation, our label-pruning method removes this label from the set of labels the classifier is allowed to output.

While our method is simple and straightforward to implement, its analysis is quite tricky, since it is based on premises that appear to be statistically unacceptable. After all, our basic assumption is that most labels are very rare, so the decision to dump a label may be based on statistically insufficient evidence. Say that we see two false-positives and one true-positive of a given label in our validation set: can we confidently decide to remove that label? The key to the formal analysis of our technique is to think of its overall effect rather than considering its effect on each individual label. Indeed, we cannot conclusively evaluate each label and our technique will most certainly mistake some good labels for bad ones. Nevertheless, we can show that our pruning criterion removes more bad labels than good ones, and overall improves the accuracy of our classifier, under mild conditions that often hold in practice. Concretely, we assume that the label frequencies follow a power-law distribution and that every example only belongs to a bounded number of different classes.

To our knowledge, our theoretical approach is unique and quite distinct from previous analyses of multiclass-multilabel learning algorithms. Most previous such analyses build on techniques originally developed for the analysis of binary classification algo-

rithms, and therefore require at least some degree of label-wise convergence.

We conclude our paper with a set of experiments, in which we validate our approach on the task of categorizing web pages using the set of 1.5 million Wikipedia categories. The simplicity of our approach enables us to perform these experiments on a single server, without requiring a large cluster computer.

Related Work

The work in [12] deals with multiclass classification and bears similarities to our work, in that the space of possible classes can be very large compared to the size of the dataset. However, the analysis there is specific to multiclass rather than multiclass-multilabel learning (i.e. each instance is assigned only a single label), and focuses on large margin classifiers with a particular rule for choosing the label of each instance.

A more closely related paper is [8], which also deals with massive multiclass-multilabel classification. It proposes a clever method, where a predictor is trained on a compressed representation of the original label vectors. The original labels are reconstructed using techniques from compressed sensing. The problem setting and some of the assumptions made (such as label sparsity) are similar to our work. However, the approach of [8] applies only to learning algorithms that regress on a real-valued compressed label vector. This is often not the case with algorithms designed for massive datasets, such as the click-graph based approach described earlier. In contrast, our approach makes no assumptions about the learning algorithm.

2 Setting and Notation

We assume that the learning task at hand is a supervised multiclass-multilabel problem. Formally, let $\mathcal{X}$ be an arbitrary measurable space, $\mathcal{Y} = \{0,1\}^k$, and let $\mathcal{D}$ be an unknown distribution on the product space $\mathcal{X} \times \mathcal{Y}$. Each element $(\mathbf{x}, \mathbf{y})$ in this space is composed of an instance $\mathbf{x}$ and a vector of indicators $\mathbf{y} = (y_1, \ldots, y_k)$ that represents the set of labels associated with $\mathbf{x}$. We assume that label vectors sampled from $\mathcal{D}$ are *sparse*, namely, that $\Pr(\sum_j y_j \leq s) = 1$ for some constant s. A classifier is a function $h : \mathcal{X} \to \mathcal{Y}$, that maps an instance $\mathbf{x}$ to a label vector $\hat{\mathbf{y}} = h(\mathbf{x})$. We restrict our discussion to classifiers that output sparse label vectors, namely $\sum_j \hat{y}_j \leq s$. We evaluate the accuracy of a classifier using a loss function $\ell(h(\mathbf{x}), \mathbf{y})$, which measures the disparity between the predicted label set and the actual label set. In this paper, we focus on a simple weighted loss function that is parameterized by $\gamma \in (0,1)$ and defined as

$$\frac{1}{s} \sum_{j=1}^{k} (1-\gamma)\, \mathbb{1}(\hat{y}_j = 0, y_j = 1) + \gamma\, \mathbb{1}(\hat{y}_j = 1, y_j = 0), \tag{1}$$

where $\mathbb{1}(\cdot)$ is the indicator function. The parameter γ controls the importance of false negatives vs. false positives, and the normalization by s ensures that the loss is always bounded in $[0,1]$. Our ultimate goal is to obtain a classifier h with a small risk, which is defined as $\mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}}[\ell(h(\mathbf{x}), \mathbf{y})]$.

We distinguish between two phases of the learning process. In the *learning phase*, we use some learning algorithm to obtain an *initial classifier* h. We then perform a *post-learning phase*, where we find a *label transformation function* $\varphi : \mathcal{Y} \to \mathcal{Y}$, such that the final classifier is the composition $\varphi \circ h$. In this paper, we focus on the post-learning phase, and make no assumptions on the learning phase or on the quality of the initial classifier. From the perspective of the post-learning phase, the initial classifier h is simply a predefined function. For simplicity, we assume that the data used to train h is independent from the data used in the post-learning phase to train φ.

In principle, the label transformation function φ can be arbitrarily complex. In this paper, we focus on the simple set of *label pruning rules*. Formally, a label pruning rule $\varphi_{\tilde{\mathbf{y}}}$ corresponds to an element $\tilde{\mathbf{y}} \in \{0,1\}^k$, and is defined as $\varphi_{\tilde{\mathbf{y}}}(\mathbf{y}) = \max\{\mathbf{y} - \tilde{\mathbf{y}}, \mathbf{0}\}$. In words, we simply remove the labels represented by $\tilde{\mathbf{y}}$ from the set of labels represented by $\mathbf{y}$. Such rules are simple to implement and are particularly useful in massive multilabel problems, where many labels are both inherently noisy and very rare. In such cases, refraining from predicting these labels can actually improve the final classifier's performance.

The four basic quantities we work with are the risk and the empirical risk of h and of $\varphi \circ h$. Letting S denote an i.i.d. sample of size m from $\mathcal{D}$, we define the *initial empirical risk* $\hat{R}_0 = 1/m \sum_{(\mathbf{x},\mathbf{y})\in S} \ell(h(\mathbf{x}), \mathbf{y})$, the *initial risk* $R_0 = \mathbb{E}_{(\mathbf{x},\mathbf{y})}[\ell(h(\mathbf{x}), \mathbf{y})]$, the *final empirical risk* $\hat{R}_\varphi = 1/m \sum_{(\mathbf{x},\mathbf{y})\in S} \ell(\varphi \circ h(\mathbf{x}_i), \mathbf{y})$, and the *final risk* $R_\varphi = \mathbb{E}_{(\mathbf{x},\mathbf{y})}[\ell(\varphi \circ h(\mathbf{x}), \mathbf{y})]$. Our goal is to find a pruning rule φ such that R_φ is as small as possible.

For the analysis, we need to describe these quantities in an alternative form, as specified in the following easy-to-prove lemma:

Lemma 1. *For a given classifier $h(\cdot)$, define*

$$\begin{aligned}
\hat{p}_{j,11} &= \frac{1-\gamma}{m} \sum_{(\mathbf{x},\mathbf{y})\in S} \mathbb{1}(h(\mathbf{x})_j = y_j = 1) \\
\hat{p}_{j,10} &= \frac{\gamma}{m} \sum_{(\mathbf{x},\mathbf{y})\in S} \mathbb{1}(h(\mathbf{x}) = 1, y_j = 0) \\
\hat{p}_{j,\neq} &= \frac{1}{m} \sum_{(\mathbf{x},\mathbf{y})\in S} (1-\gamma)\, \mathbb{1}(h(\mathbf{x})_j = 0, y_j = 1) \\
&\qquad\qquad + \gamma\, \mathbb{1}(h(\mathbf{x})_j = 1, y_j = 0)\,.
\end{aligned}$$

Let $p_{j,\neq}, p_{j,11}$, and $p_{j,10}$ be the expected values (over the sample S) of $\hat{p}_{j,\neq}, \hat{p}_{j,11}$, and $\hat{p}_{j,10}$ respectively. Also, for a fixed pruning rule $\varphi(\cdot)$, let $\mathbb{1}$(label j pruned) be an indicator that equals 1 if and only if the pruning rule $\varphi(\cdot)$ removes label j. Then it

holds that

$$\hat{R}_0 = \frac{1}{s}\sum_{j=1}^{k}\hat{p}_{j,\neq} \ , \ \ R_0 = \frac{1}{s}\sum_{j=1}^{k}p_{j,\neq},$$

$$\hat{R}_\varphi = \frac{1}{s}\sum_{j=1}^{k}\Big(\hat{p}_{j,\neq} + \mathbb{1}(\text{label } j \text{ pruned})\,(\hat{p}_{j,11} - \hat{p}_{j,10})\Big),$$

$$R_\varphi = \frac{1}{s}\sum_{j=1}^{k}\Big(p_{j,\neq} + \mathbb{1}(\text{label } j \text{ pruned})\,(p_{j,11} - p_{j,10})\Big).$$

3 The Pruning Method

Recall that our goal is to reduce the final risk R_φ. The expression for R_φ given in Lemma 1 suggests that R_φ can be reduced by removing those labels for which $p_{j,10} > p_{j,11}$. Unfortunately, $p_{j,11}$ and $p_{j,10}$ are unknown quantities that depend on $\mathcal{D}$, and we must resort to using their empirical counterparts $\hat{p}_{j,11}$ and $\hat{p}_{j,10}$. Specifically, our simple label pruning procedure proceeds as follows: given a sample S, calculate $\hat{p}_{j,11}$ and $\hat{p}_{j,10}$, and choose the label pruning rule φ that removes all labels for which $\hat{p}_{j,11} < \hat{p}_{j,10}$. In other words, this procedure prunes any label for which the ratio of false positives to true positives exceeds $(1-\gamma)/\gamma$. Notice that this makes φ a random function that depends on the randomness of the sample S. For the theoretical analysis, it will be convenient to view R_φ and $\hat{R}_\varphi$ as random variables, which depend on the random draw of S.

This algorithm essentially attempts to decrease the final empirical risk $\hat{R}_\varphi$ in lieu of R_φ. However, notice that in our setting (where k scales with m), we cannot assume that each and every $\hat{p}_{j,11}, \hat{p}_{j,10}$ is an accurate estimate of $p_{j,11}, p_{j,10}$. In fact, our analysis shows that $\hat{R}_\varphi$ is generally *not* a good estimator of R_φ. Nevertheless, we can prove that our method reduces the final risk R_φ compared to the initial risk R_0 with high probability, under mild conditions.

4 Theoretical Analysis

Our pruning procedure works by making the empirical risk $\hat{R}_\varphi$ as small as possible. In this section, we show that that this is also likely to make R_φ smaller than R_0. The straightforward theoretical approach would be to show that for reasonably large samples, $\hat{R}_0$ is close to R_0 and $\hat{R}_\varphi$ is close to R_φ. While the first premise is easy to show via a large deviation inequality, it turns out that $\hat{R}_\varphi$ does not necessarily converge to R_φ when the number of labels grows with the number of examples. This is implied by the following theorem and the discussion which follows. Its proof is a simple consequence of the definitions, and is omitted due to lack of space.

Theorem 1. *$\mathbb{E}[R_\varphi - \hat{R}_\varphi]$ is lower bounded by*

$$\frac{1}{s}\sum_{j=1}^{k} \Pr(\text{label } j \text{ pruned})(p_{j,11} - p_{j,10}).$$

If we were to assume that k is fixed, we might expect $\hat{p}_{j,11}, \hat{p}_{j,10}$ to converge to $p_{j,11}, p_{j,10}$ uniformly for all $j = 1, \ldots, k$. Since our method prunes labels for which $\hat{p}_{j,11} < \hat{p}_{j,10}$, we would have that $\Pr(\text{label } j \text{ pruned})(p_{j,11} - p_{j,10})$ converges to a non-positive quantity uniformly for any j, and thus our lower bound would converge to 0. However, when we assume that k grows with m, $\hat{p}_{j,11}, \hat{p}_{j,10}$ need not converge uniformly to $p_{j,11}, p_{j,10}$, and the correlation between $\Pr(\text{label } j \text{ pruned})$ and the sign of $(p_{j,11} - p_{j,10})$ can remain weak regardless of the sample size. To give a concrete example, if we take $\gamma = 1/2, s = 10$ and assume that $p_{j,11} = s/3k, p_{j,10} = s/6k$ for all j, then we have by the theorem above that

$$\mathbb{E}[R_\varphi - \hat{R}_\varphi] \geq \frac{1}{6k}\sum_{j=1}^{k} \Pr(\text{label } j \text{ pruned}).$$

It can be shown that when $m, k \to \infty$ but (say) $m/k = 3$, the right hand side above converges to a strictly positive constant. Therefore, it is possible that our lower bound will remain larger than some positive constant regardless of sample size, which implies that $\hat{R}_\varphi$ does not converge to R_φ in such cases.

This observation precisely captures the difficulty of working with a sample that does not sufficiently represent many of the individual classes in the problem, and is the reason why most existing algorithms are inadequate when the number of labels is not fixed. Nevertheless, we can show that it is possible to analyze the behavior of R_φ directly. Specifically, we prove that R_φ is well behaved when the training set is large enough, even when k is very large and grows with m. Namely, although the empirical quantities do not necessarily correspond to their expected values, we can still provide high probability guarantees that our pruning method reduce the overall risk of the classifier. In a nutshell, the analysis consists of proving that $|R_\varphi - \mathbb{E}[R_\varphi]|$ is small with high probability (where the expectations are taken over the random draw of the sample S), and then directly proving that $\mathbb{E}[R_\varphi]$ is strictly smaller than R_0, under mild conditions.

The first part of the proposed approach is formalized in the following theorem. Informally, it states that when m is large enough, R_φ is arbitrarily close to its expectation with arbitrarily high probability. Note that this bound does not depend at all on k, the number of labels.

Theorem 2. *For any fixed $\epsilon > 0$, it holds that*

$$\begin{aligned}&\Pr\left(|R_\varphi - \mathbb{E}[R_\varphi]| > \frac{2m^{-1/6+\epsilon}}{\gamma(1-\gamma)} + m^{2/3}\exp\left(-m^{2\epsilon}\right)\right)\\ &\quad\leq 2sm^{2/3}\exp\left(-m^{2\epsilon}\right),\end{aligned}$$

The proof is presented in the appendix. Intuitively, the idea is to distinguish between labels for which $|p_{j,11} - p_{j,10}|$ is large, and labels for which this difference is small. The first type of labels are more common in the data, and thus we can reliably estimate $p_{j,11} - p_{j,10}$ and decide whether to prune them or not. On the other hand, there cannot be too many such labels, because $\sum_j p_{j,11} + p_{j,10}$ is a bounded quantity. This effectively limits the dimensionality of the problem regardless of the parameter k. Whenever $|p_{j,11} - p_{j,10}|$ is small, the pruning process is noisy and prone to errors, but it can be shown that these cases do not influence R_φ too much. A careful formalization of these ideas, using Bernstein and McDiarmid's large deviation bounds, allows us to show that R_φ concentrates around its expectation with high probability, regardless of k.

Next, we need to show that $R_0 - \mathbb{E}[R_\varphi]$ is strictly positive, to prove that our method indeed reduces the final risk. It turns out that the exact value of $R_0 - \mathbb{E}[R_\varphi]$ is highly dependent on the specific values of $p_{j,11}$ and $p_{j,10}$ for each j. Intuitively, if labels for which $p_{j,10} > p_{j,11}$ are pruned with high probability and labels for which $p_{j,10} \leq p_{j,11}$ are pruned with low probability, we expect $R_0 - \mathbb{E}[R_\varphi]$ to be large. Although it is possible to provide positive lower bounds on $R_0 - \mathbb{E}[R_\varphi]$ in terms of these quantities, they are not particularly enlightening. Instead, the theorem below will allow us to characterize a mild condition, under which we can expect $R_0 - \mathbb{E}[R_\varphi]$ to be strictly positive. A proof appears in the appendix.

Theorem 3. *The difference $R_0 - \mathbb{E}[R_\varphi]$ is at least*

$$\frac{1}{s} \sum_{j:p_{j,10} \geq p_{j,11}} (p_{j,10} - p_{j,11}) - \frac{1}{s} \sum_{j=1}^{k} \sqrt{\frac{p_{j,11} + p_{j,10}}{m}}.$$

Moreover, if we assume that $p_{j,10} + p_{j,11}$ are sorted in descending order, and there exists some $r \neq 2$ such that $p_{j,10} + p_{j,11} \leq O(j^{-r})$ for all j, then $R_0 - \mathbb{E}[R_\varphi]$ is at least

$$\frac{1}{s} \sum_{j:p_{j,10} \geq p_{j,11}} (p_{j,10} - p_{j,11}) - O\left(\sqrt{\frac{k^{\max\{2-r,0\}}}{m}}\right).$$

The requirement that $r \neq 2$ is for technical reasons and can easily be treated separately.

What does this theorem tell us? The non-negative term $\sum_{j:p_{j,10} \geq p_{j,11}} (p_{j,10} - p_{j,11})$ can be arbitrarily small, but we can expect it to be lower bounded by a positive constant (independent of m, k) if a fixed fraction of the labels are such that $p_{j,10} \geq p_{j,11}$, and if $p_{j,10} - p_{j,11}$ is proportional to $p_{j,10} + p_{j,11}$. So we turn our attention to the term

$$\frac{1}{s} \sum_{j=1}^{k} \sqrt{\frac{p_{j,11} + p_{j,10}}{m}} \ ,$$

which can indeed be large in the regime where k scales with m. For example, if $p_{j,11} + p_{j,10} = s/k$ for all j, the above equals $\sqrt{k/sm} \geq \Omega(1)$, and Thm. 3 may become vacuous. Luckily, assuming that $p_{j,11} + p_{j,10}$ is equal for all j is unrealistic. By

definition, $p_{j,11} + p_{j,10}$ is closely related to the probability that our learned hypothesis labels a random instance with label j. If the marginal label distribution of the classifier is similar to the marginal label distribution of the data, then this distribution is often observed to follow a *power law*, which corresponds to the assumption that $p_{j,10} + p_{j,11} \leq O(j^{-r})$ for all j. Under this assumption, we obtain the second statement in Thm. 3. This power-law behavior, sometimes known as Zipf's law, is a very well known and well documented phenomenon for many rank vs. frequency datasets (see examples in [10, 1, 7]), and in particular for the applications we have in mind. We verify this property in our experiments, presented below.

Overall, this lower bound implies that if we let $m, k \to \infty$, we can expect $R_0 - \mathbb{E}[R_\varphi]$ to be positive whenever m grows faster than k^{2-r}. In particular, if $r > 1$ (which happens quite often in practice, including in our experiments), we obtain the interesting result that the lower bound remains meaningful, even when the number of labels k grows faster than the number of examples m.

5 Experiments

We applied our technique to the task of categorizing web pages using the 1.5 million categories defined in Wikipedia. As mentioned in the introduction, we first used search engine logs to create a click graph, which is a bipartite graph between queries and web pages. A link between query Q and web page W indicates that a sufficiently large number of users issued the query Q and then clicked on the link to page W. Next, we randomly split the set of Wikipedia articles into three sets: 50% training, 30% validation, and 20% test. Each Wikipedia article is associated with a set of categories and also corresponds to a node in the click graph. Next, we propagated the categories from each Wikipedia training article along the edges of the click graph, to all of the web pages that have a query in common with that article (namely, to all web pages whose distance to the training article is 2). We call the resulting labeling of the web *labeling A*. The rationale behind this labeling procedure is the assumption that two web pages that were clicked on (by different people, at different times) after the same query are likely to share many topics. Next, we propagated the categories along the edges of the click graph a second time, extending the reach of each category to all pages with graph distance 4 from the original article. We call this *labeling B*.

We repeated the process described above a second time, this time seeded with a larger set of labels per Wikipedia training article. We used the fact that Wikipedia categories are themselves categorized by higher-level categories. For example, the Wikipedia article on *Dogs* is associated with the category *Domesticated Animals*, and the latter is associated with the category *Animals*. We added all of these second-order categories to each Wikipedia article. We propagated the extended category sets along the edges of the click graph as before, to obtain *labeling C*. We then performed a second iteration of label-propagation to obtain *labeling D*.

We applied our label-pruning technique independently to each of the four initial labellings. Namely, we revealed the true categories of the Wikipedia validation articles and compared them to the propagated labels in the four versions of our experiment. For each label we counted true and false positives, and decided which labels to prune.

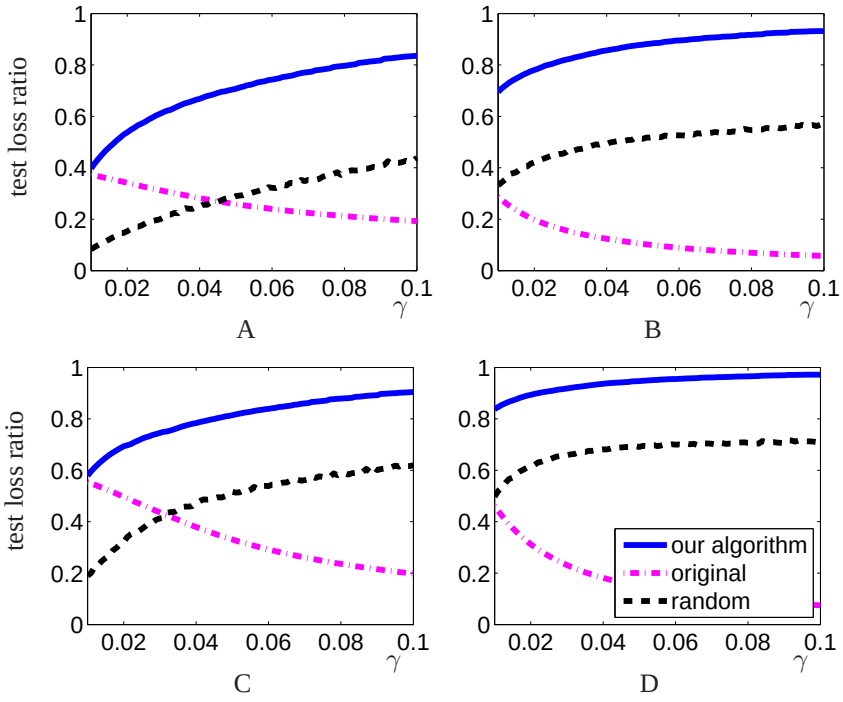

Figure 1: Ratios between the best attainable test loss and the test loss attained by three different techniques, on four different initial labellings.

The set of Wikipedia categories is problematic in that it is over-complete. Many categories have duplicates or near-duplicates; some articles are labeled by one category while other articles are labeled by its near-duplicate category. Also the false-positives in all four labellings significantly outnumber the true-positives. For these reasons, false-positives should be treated with great suspicion. When we see a false-positive, either our classification is wrong or the Wikipedia editor may have simply neglected to add this category. Spot-checking reveals that many false-positives are actually quite reasonable. On the other hand, false-negatives should always be taken seriously: a human editor explicitly added a category to the article and our algorithm concluded that it is not relevant. To correct this imbalance, we set γ in Eq. (1) to give more weight to false-negatives. Specifically, we set γ to values between 0.01 and 0.1.

After using the validation set to identify and remove harmful labels, we revealed the categories in the Wikipedia test set, and evaluated the performance of our algorithm. For each of the four labellings and for each value of γ, we also calculated an *oracle* pruning which provides a lower bound on the test loss of any possible pruning algorithm. This was done by cheating and finding the best pruning on the test set (in terms of each γ-weighted loss). The loss attained by the oracle varies greatly with γ, so it is meaningless to plot absolute loss values for different values of γ on the same figure. To get a coherent visualization of our results, we plotted the ratio between the

oracle loss and the loss of our algorithm. The performance of our algorithm is shown in solid lines in Fig. 1. Values close to 1 indicate that our test loss is very close to the loss of the ideal pruning.

For comparison, the plots in Fig. 1 also show the performance of two other simple algorithms. The first is the algorithm that performs no pruning and just keeps the initial labeling. The second is an algorithm that uses our method to determine how many labels to remove, and then removes labels randomly. These experimental results clearly show the amount of improvement achieved by our algorithm. Despite the statistical challenge of generalizing with only a handful of examples per class, our algorithm performs very well across a wide range of γ.

Finally, using a simple least-squares fitting technique, we validated that all four datasets satisfy the power-law assumption used in our theoretical analysis (see Thm. 3 and the discussion which follows). Namely, when we sort the labels by frequency in the data, we see that the frequency of label y_j is proportional to j^{-r}, with $r \approx 1.3$ for labeling A; $r \approx 1.6$ for labeling B; $r \approx 1.9$ for labeling C; and $r \approx 2.3$ for labeling D.

6 Conclusions

In this paper, we studied the problem of massive multiclass-multilabel learning, where the set of labels scales with the number of available training examples. This setting is very relevant when the label-set results from a collaborative tagging scheme, such as Wikipedia categories or keywords in media hosting websites. In this regime, the standard assumption of a fixed label set is too simplistic, and straightforward generalizations of methods for binary classification (such as multiclass SVM) may be impractical.

Motivated by the computational issues faced by practitioners in this area, we proposed and analyzed a *post-learning* method on top of any desired learning algorithm, which for our purposes can be treated as a black-box. Our experiments demonstrate that the method works quite well on real-world, large scale data.

Theoretically, this setting poses a challenge, since we cannot hope to get statistically significant data on each and every label. As far as we know, this setting violates the assumptions underlying most previous theoretical work on multiclass-multilabel learning. Nevertheless, a careful analysis allows us to justify our approach, using some non-trivial but mild sufficient conditions, such as sparsity of labels per instance and a power-law behavior of the label frequencies.

While our approach seems to work in practice, and has some interesting theoretical properties, the algorithm we have focused on is obviously a very simple one, and several extensions immediately come to mind. One direction is to utilize additional knowledge about label dependencies, rather than treating each label separately. Also, we have dealt only with very simple label transformation rules, which prune a subset of labels (i.e. "if label A appears, remove it"). However, it is possible to envision more complex rules, such as "if labels A and B appear, but not label C, replace label D by label E". Understanding how to implement these extensions effectively and in a theoretically justified manner, even when there are as many labels as examples, remains a

topic for future research.

References

[1] Lada A. Adamic and Bernardo A. Huberman. Zipf's law and the internet. *Glottometrics*, 3:143–150, 2002.

[2] Yonatan Amit, Ofer Dekel, and Yoram Singer. A boosting algorithm for label covering in multilabel problems. In *AISTATS*, 2007.

[3] Zafer Barutcuoglu, Robert E. Schapire, and Olga G. Troyanskaya. Hierarchical multi-label prediction of gene function. *Bioinformatics*, 22(7):830–836, 2006.

[4] Stéphane Boucheron, Gábor Lugosi, and Olivier Bousquet. Concentration inequalities. In Olivier Bousquet, Ulrike von Luxburg, and Gunnar Rätsch, editors, *Advanced Lectures on Machine Learning*, volume 3176 of *Lecture Notes in Computer Science*, pages 208–240. Springer, 2004.

[5] Matthew R. Boutell, Jiebo Luo, Xipeng Shen, and Christopher M. Brown. Learning multi-label scene classification. *Pattern Recognition*, 37(9):1757–1771, 2004.

[6] Koby Crammer and Yoram Singer. A family of additive online algorithms for category ranking. *Journal of Machine Learning Research*, 3:1025–1058, 2003.

[7] Xavier Gabaix. Zipf's law for cities: An explanation. *The Quarterly Journal of Economics*, 114(3):739–767, August 1999.

[8] Daniel Hsu, Sham Kakade, John Langford, and Tong Zhang. Multi-label prediction via compressed sensing. In *NIPS*, 2009.

[9] Guy Lebanon and John D. Lafferty. Conditional models on the ranking poset. In *NIPS*, pages 415–422, 2002.

[10] Christopher D. Manning and Hinrich Schütze. *Foundations of Statistical Natural Language Processing*. MIT Press, 2002.

[11] Andrew McCallum. Multi-label text classification with a mixture model trained by em. In *AAAI 99 Workshop on Text Learning*, 1999.

[12] Tong Zhang. Class-size independent generalization analysis of some discriminative multi-category classification. In *NIPS*, pages 1625–1632, 2004.

A Technical Proofs

A.1 Proof of Thm. 2

We need the following two lemmas. The first lemma follows directly from Bernstein's inequality (see for instance [4]). We note that using an inequality that relies on variance is crucial to obtain a non-trivial bound with our proof technique. The second lemma follows directly from the definitions. The proofs are omitted due to lack of space.

Lemma 2. *For any j, if $p_{j,10} \leq p_{j,11}$, then $\Pr(\hat{p}_{j,10} > \hat{p}_{j,11})$ is at most*

$$\exp\left(-\frac{m(p_{j,10}-p_{j,11})^2}{2((1-\gamma)p_{j,11}+\gamma p_{j,10}+|p_{j,10}-p_{j,11}|/3)}\right),$$

A similar bound holds on $\Pr(\hat{p}_{j,10} \leq \hat{p}_{j,11})$ if $p_{j,10} \geq p_{j,11}$.

Lemma 3. *It holds that*

$$\sum_{j=1}^{k} |p_{j,11}-p_{j,10}| \;\leq\; \sum_{j=1}^{k} p_{j,11}+p_{j,10} \;\leq\; s.$$

Let $\alpha > 0$ be an arbitrary parameter to be specified later, and define the label subsets $J_1 = \{j : |p_{j,11}-p_{j,10}| \leq \alpha\}$, $J_2 = \{1,\ldots,k\} \setminus J_1$. We have by definition of the pruning procedure and Lemma 1 that $|R_\varphi - \mathbb{E}[R_\varphi]|$ is at most

$$\frac{1}{s}\left|\sum_{j\in J_1}(p_{j,11}-p_{j,10})\left(\mathbb{1}_{\hat{p}_{j,10}>\hat{p}_{j,11}} - \Pr\left(\hat{p}_{j,10}>\hat{p}_{j,11}\right)\right)\right|$$
$$+\frac{1}{s}\sum_{j\in J_2}|p_{j,11}-p_{j,10}|\left|\mathbb{1}_{\hat{p}_{j,10}>\hat{p}_{j,11}} - \Pr\left(\hat{p}_{j,10}>\hat{p}_{j,11}\right)\right|. \tag{2}$$

Focusing on the first line in the expression, note that if we change any single instance in our sample, at most $2s$ terms will change by at most $|p_{j,11}-p_{j,10}| \leq \alpha$. Therefore, the expression in the first line will change by at most 2α. Applying McDiarmid's inequality, and noting that the expectation of what's inside the absolute value is zero, we get that with probability of at least $1-\delta$, it is upper bounded by

$$\sqrt{2m\alpha^2\log(1/\delta)}. \tag{3}$$

Turning to the second line in Eq. (2), and applying Lemma 2, we get that for any j, with probability of at least $1-g(m,p_{j,11},p_{j,10})$, it holds that

$$|\,\mathbb{1}(\hat{p}_{j,10}>\hat{p}_{j,11}) - \Pr(\hat{p}_{j,10}>\hat{p}_{j,11})| \leq g(m,p_{j,11},p_{j,10}),$$

where $g(m,p_{j,11},p_{j,10})$ equals

$$\exp\left(-\frac{m(p_{j,11}-p_{j,10})^2}{2((1-\gamma)p_{j,11}+\gamma p_{j,10}+|p_{j,10}-p_{j,11}|/3)}\right).$$

Let $c > 0$ be another parameter to be determined later. If $c((1-\gamma)p_{j,11}+\gamma p_{j,10}) \leq |p_{j,10}-p_{j,11}|$, we can upper bound $g(m,p_{j,11},p_{j,10})$ by

$$\exp\left(-\frac{mc^2((1-\gamma)p_{j,11}+\gamma p_{j,10})^2}{2\left((1-\gamma)p_{j,11}+\gamma p_{j,10}+|p_{j,10}-p_{j,11}|/3\right)}\right).$$

Dividing the numerator and denominator of the fraction in the exponent by $(1-\gamma)p_{j,11}+\gamma p_{j,10}$, and using the easily verified fact that for any $a>0, b>0, \gamma \in (0,1)$ it holds that $|a-b|/((1-\gamma)a+\gamma b) \leq 1/(\gamma(1-\gamma))$, we get the upper bound

$$\exp\left(-\frac{mc^2((1-\gamma)p_{j,11}+\gamma p_{j,10})}{2(1+1/3\gamma(1-\gamma))}\right). \tag{4}$$

On the other hand, we always have

$$|\,\mathbb{1}(\hat{p}_{j,10} > \hat{p}_{j,11}) - \Pr(\hat{p}_{j,10} > \hat{p}_{j,11})| \leq 1 \tag{5}$$

with probability 1. Applying Eq. (4) and Eq. (5) on the second line of Eq. (2), we get a probabilistic upper bound for it, of the form

$$\sum_{j \in J_{2,1}} \frac{|p_{j,11} - p_{j,10}|}{s} \exp\left(-\frac{mc^2((1-\gamma)p_{j,11} + \gamma p_{j,10})}{2(1 + 1/3\gamma(1-\gamma))}\right) + \frac{1}{s} \sum_{j \in J_{2,2}} |p_{j,11} - p_{j,10}|, \tag{6}$$

where $J_{2,1} = \{j \in J_2 : c \leq \frac{|p_{j,10} - p_{j,11}|}{(1-\gamma)p_{j,11} + \gamma p_{j,10}}\}$, and $J_{2,2} = \{j \in J_2 \setminus J_{2,1}\}$. By a union bound, Eq. (5) holds with probability at least

$$1 - \sum_{j \in J_1} \exp\left(-\frac{mc^2((1-\gamma)p_{j,11} + \gamma p_{j,10})}{2(1 + 1/3\gamma(1-\gamma))}\right). \tag{7}$$

We now make four observations. First, by Lemma 3, $\sum_j |p_{j,11} - p_{j,10}| \leq s$, so there can be at most s/α labels j such that $|p_{j,11} - p_{j,10}| > \alpha$. Second, it is easy to verify that if $|p_{j,11} - p_{j,10}| > \alpha$ (which holds for any $j \in J_{2,1}$), then $(1-\gamma)p_{j,11} + \gamma p_{j,10} > \alpha\gamma(1-\gamma)$. Third, for any $j \in J_{2,2}$, $|p_{j,11} - p_{j,10}| < c((1-\gamma)p_{j,11} + \gamma p_{j,10})$. Fourth, $\sum_{j \in J_{2,2}}((1-\gamma)p_{j,11} + \gamma p_{j,10}) \leq s$ by Lemma 3 and the fact that $\gamma \in (0,1)$. Applying these four observations on Eq. (6) and Eq. (7), we can weaken this bound to the form

$$\frac{1}{\alpha} \exp\left(-\frac{mc^2\alpha\gamma(1-\gamma)}{2(1 + 1/3\gamma(1-\gamma))}\right) + c,$$

which holds with probability of at least

$$1 - \frac{s}{\alpha} \exp\left(-\frac{mc^2\alpha\gamma(1-\gamma)}{2(1 + 1/3\gamma(1-\gamma))}\right).$$

To get the theorem statement, we combine this with the bound in Eq. (3), substitute into Eq. (2), choose $\alpha = m^{-2/3}$, $\delta = sm^{2/3} \exp\left(-m^{2\epsilon}\right)$ (for some $\epsilon > 0$), let

$$c = m^{-1/6+\epsilon} \sqrt{\frac{2(1 + 1/3\gamma(1-\gamma))}{\gamma(1-\gamma)}},$$

and perform some straightforward simplifications. □

A.2 Proof of Thm. 3

We have that $R_0 - \mathbb{E}[R_\varphi]$ equals

$$\frac{1}{s} \sum_{j=1}^{k} (p_{j,10} - p_{j,11}) \Pr(\hat{p}_{j,10} > \hat{p}_{j,11}). \tag{8}$$

For any j, if $p_{j,10} - p_{j,11} \geq 0$, we have by Lemma 2 that $\Pr(\hat{p}_{j,10} \geq \hat{p}_{j,11})$ is lower bounded by

$$\begin{aligned}
&1 - \exp\left(-\frac{m(p_{j,10} - p_{j,11})^2}{2((1-\gamma)p_{j,10} + \gamma p_{j,11}) + |p_{j,10} - p_{j,11}|/3}\right) \\
&\geq 1 - \exp\left(-\frac{m(p_{j,10} - p_{j,11})^2}{2(p_{j,10} + p_{j,11} + (p_{j,10} + p_{j,11})/3)}\right) \\
&= 1 - \exp\left(-\frac{3m(p_{j,10} - p_{j,11})^2}{8(p_{j,10} + p_{j,11})}\right).
\end{aligned}$$

If $p_{j,10} - p_{j,11} \leq 0$, we have by Lemma 2 in a similar manner that

$$\Pr(\hat{p}_{j,10} > \hat{p}_{j,11}) \leq \exp\left(-\frac{3m(p_{j,10} - p_{j,11})^2}{8(p_{j,10} + p_{j,11})}\right).$$

Substituting these results into Eq. (8), we get that $R_0 - \mathbb{E}[R_\varphi]$ is lower bounded by

$$\begin{aligned}
&\frac{1}{s} \sum_{j:p_{j,10} \geq p_{j,11}} (p_{j,10} - p_{j,11}) \\
&\qquad - \frac{1}{s} \sum_{j=1}^{k} |p_{j,10} - p_{j,11}| \exp\left(-\frac{3m(p_{j,10} - p_{j,11})^2}{8(p_{j,10} + p_{j,11})}\right). \quad (9)
\end{aligned}$$

In order to upper bound the second line in the expression (with something which does not depend on $p_{j,10} - p_{j,11}$), it is enough to upper bound for any j the expression

$$\max_{|p_{j,10} - p_{j,11}|} |p_{j,10} - p_{j,11}| \exp\left(-\frac{3m(p_{j,10} - p_{j,11})^2}{8(p_{j,10} + p_{j,11})}\right). \quad (10)$$

For that, it is sufficient to find the maximal value of the function $f(x) = x \exp(-3mx^2/8p)$, where $p := p_{j,11} + p_{j,10}$, for any $x \in [0, p]$. It can be verified that this function is maximized at $x = \sqrt{4p/3m}$. Substituting this value for $|p_{j,10} - p_{j,11}|$ in Eq. (10), we get an upper bound of the form $\sqrt{4(p_{j,10} + p_{j,11})/3m \exp(1)}$. Substituting this bound in Eq. (9), and simplifying by noting that $\sqrt{4/3 \exp(1)} \approx 0.7 < 1$, we get the required lower bound

$$\frac{1}{s} \sum_{j:p_{j,10} \geq p_{j,11}} (p_{j,10} - p_{j,11}) - \frac{1}{s} \sum_{j=1}^{k} \sqrt{\frac{p_{j,11} + p_{j,10}}{m}}$$

on $R_0 - \mathbb{E}[R_\varphi]$. To derive from it the second inequality in the theorem, notice that under the assumptions stated there, $\sum_{j=1}^{k} \sqrt{p_{j,11} + p_{j,10}}$ is at most $C \sum_{j=1}^{k} j^{-r/2}$ for some constant C. This sum is $O(k^{1-r/2})$ if $r < 2$, $O(\log(k))$ if $r = 2$, and $O(1)$ if $r > 2$. Ignoring the case $r = 2$ for simplicity, we upper bound the different cases by $O(\sqrt{k^{\max\{2-r,0\}}})$, and the inequality stated in the theorem follows. □