Learning Not to Try Too Hard

Bill McDowell
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213
forkunited@gmail.com

Noah A. Smith
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213
nasmith@cs.cmu.edu

Abstract

We describe an extension to margin-based linear models for multiclass classification in which the cost function is learned alongside the parameters. The intuition is to exploit estimates of the relative "ease" of avoiding different types of errors, allowing the learner to focus on easier distinctions while "giving up" on harder ones.

Note to collaborators: please redefine the command `\forpubnote` *so that it does not suppress internal notes.*

1 Introduction

Discriminative learning algorithms are often motivated by their ability to trade off among different kinds of prediction mistakes with different costs. The cost of a mistake is usually taken to be fully defined by the task; i.e., human system designers are trusted to encode this knowledge prior to learning. Information about the inherent ease of avoiding some errors vs. others is generally not taken into account. Closely related to this, and critically important in domains where the data are constructed by humans, is the problem that the outputs in the training data may be unreliable. For example, if the training dataset is produced by asking humans to label instances, and two labels are insufficiently well defined for human labelers to distinguish them, then a learner might be forgiven for conflating them.

We consider situations where human intuition about relative costs of different errors is insufficient. In a margin-based linear modeling framework, we propose a method for incorporating **learning of the cost function** alongside learning of the model. Our approach introduces explicit estimates of the "ease" of avoiding each type of error (for a particular model family). For error types that are "just too hard," our model is offered the possibility of giving up in favor of making other, less challenging predictions more accurately.

In preliminary experiments with text classification, we find that our method performs as well as a baseline SVM, offering no clear gain. It does, however infer cost functions that appear qualitatively reasonable.

2 Background and Notation

In a prediction problem, let $\mathcal{X}$ denote the input space, $\mathcal{Y}$ denote the output space, and assume N training instances $\{(x_1, y_1), \ldots, (x_N, y_N)\}$. We assume a linear model and prediction function:

$$\hat{y} = \operatorname*{argmax}_{y \in \mathcal{Y}} \Big(f(x, y; \mathbf{w}) \triangleq \mathbf{w}^\top \mathbf{g}(x, y) \Big) \tag{1}$$

where $\mathbf{w} \in \mathbb{R}^D$ are the parameters to be learned and $\mathbf{g} : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}^D$ is the feature vector function. We will let $\mathcal{M} = \{f(\cdot, \cdot; \mathbf{w}) \mid \mathbf{w} \in \mathbb{R}^D\}$ denote the model family under consideration, given a fixed choice of $\mathbf{g}$.

Our approach, which assumes $\mathcal{Y}$ is categorical, is based on the soft margin formulation of multiclass support vector machines [1–3]. Tsochantaridis et al. [4] and Taskar et al. [5] generalized this framework to allow for differences in costs between different kinds of mistakes, as found when $\mathcal{Y}$ is structured. Let the cost function $\Delta : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ be such that $\Delta(y, y')$ is the cost of predicting y when the correct label is y'. We use the "margin rescaling" variant of the multiclass SVM:

$$\min_{\boldsymbol{\xi} \geq 0, \mathbf{w}} \frac{\lambda}{2}\|\mathbf{w}\|_2^2 + \frac{1}{m}\sum_{i=1}^{N} \xi_i \quad \text{s.t.} \quad \forall i, \forall y \in \mathcal{Y} \setminus \{y_i\}, f(x_i, y_i; \mathbf{w}) - f(x, y, \mathbf{w}) \geq \Delta(y, y_i) - \xi_i \tag{2}$$

This objective seeks $\mathbf{w}$ that minimizes misclassifications while maximizing the margin between correct and incorrect instances. Further, the more incorrect an (x, y) pair is, the greater the margin should be. This problem is often transformed into an unconstrained one corresponding to direct minimization of the regularized average hinge loss:

$$\min_{\mathbf{w}} \frac{\lambda}{2}\|\mathbf{w}\|_2^2 + \sum_{i=1}^{N} -f(x_i, y_i; \mathbf{w}) + \max_{y \in \mathcal{Y}} f(x_i, y; \mathbf{w}) + \Delta(y, y_i) \tag{3}$$

We introduce some notation for errors. We let $\mathcal{S} \subseteq 2^{\mathcal{Y} \times \mathcal{Y}}$ be a collection of prediction error classes that exhausts $\mathcal{Y}^2$ (i.e., $\bigcup_{S \in \mathcal{S}} S = \mathcal{Y}^2$); the error classes need not be mutually exclusive. We let $e_S \in \mathbb{R}$ denote an estimate of the "ease" with which a learner searching in $\mathcal{M}$ can successfully avoid errors in class S. Then we let:

$$\Delta(y, y') = \sum_{S \in \mathcal{S} : (y, y') \in S} e_S = \mathbf{e}^\top \mathbf{s}(y, y') \tag{4}$$

where $\mathbf{e}$ is a vector of the e_S and $\mathbf{s}$ is a binary vector of length $\mathcal{S}$ indicating which error class(es) each possible confusion in $\mathcal{Y} \times \mathcal{Y}$ belongs to.

In this paper, we consider two prediction error classes, corresponding to unordered and ordered pairs of outputs. We denote them $\mathcal{S}^{\mathrm{u}}$ and $\mathcal{S}^{\mathrm{o}}$, respectively.

3 Cost Learning Model

Previous work assumes Δ follows intuitively from the prediction task. For example, in natural language dependency parsing, the number of words attached to the wrong parent (Hamming distance for the parse tree) is a sensible choice. We propose to parameterize Δ and learn its parameters jointly with $\mathbf{w}$. This learned cost function should encode distances between outputs from the perspective of the ease with which a model in the family $\mathcal{M}$ can distinguish between them. This joint learning setup is expected to be particularly useful when some classes of errors are difficult or impossible for a model in the class to resolve, due to unreliable annotations or an insufficient choice of features $\mathbf{g}$.

3.1 Ease

We desire a model that estimates prediction ease $\mathbf{e}$ while estimating predictive model parameters $\mathbf{w}$. We have used the term "ease" with respect to an arbitrary model in the family $\mathcal{M}$, but it is more sensible to consider the particular model we seek to estimate. We propose that, for error class S and a model with parameters $\mathbf{w}$, ease e_S should be inversely related to the number of of margin violations involving S that $f(\cdot, \cdot; \mathbf{w})$ makes in the training data assuming that argmax always gives a single label, breaking ties arbitrarily:

$$v_S(\mathbf{w}, \mathbf{e}) \triangleq \left|\left\{i \in \{1, \ldots, N\} \mid \left(y_i, \mathrm{argmax}_{y \in \mathcal{Y}} f(x_i, y; \mathbf{w}) + \mathbf{e}^\top \mathbf{s}(y, y_i)\right) \in S\right\}\right| \tag{5}$$

The intuition is that, when $v_S(\mathbf{w}, \mathbf{e})$ is large, it is because it is not easy for the model to shrink. Of course, we should also take into account that the distribution of the data may make some errors

more frequent, inflating the size of the set in Eq. 5 even if S is "easy." Further, *infrequently* observed labels are generally expected to be harder to predict. Yet for an S that includes errors on a rarely occurring class, the set in Eq. 5 will necessarily be small, regardless of how easy it is. We therefore propose the following condition for e_S:

$$e_S = \max\left(0, 1 - \frac{v_S(\mathbf{w}, \mathbf{e})}{n_S}\right) \tag{6}$$

where n_S is a fixed, *a priori* upper bound on the count of S errors, v_S. This has the desirable property that if $v_S \geq n_S$, i.e., S is too difficult to shrink, then ease e_S goes to zero and the model is allowed to give up on S. It also keeps $e_S \in [0, 1]$, ensuring interpretability of the ease relative to the maximum possible value of $e_S = 1$.

3.2 Objective

Our approach is a modification to the SVM objective in Eq. 3; it is a joint optimization of $\mathbf{w}$ and $\mathbf{e}$:

$$\min_{\mathbf{e}\geq 0, \mathbf{w}} \frac{\lambda}{2}\|\mathbf{w}\|_2^2 + \frac{1}{2}\|\mathbf{e}\|_{\mathbf{n}}^2 - \mathbf{e}^\top \mathbf{n} + \sum_{i=1}^{N} -f(x_i, y_i; \mathbf{w}) + \max_{y\in\mathcal{Y}} f(x_i, y; \mathbf{w}) + \mathbf{e}^\top \mathbf{s}(y, y_i) \tag{7}$$

where $\mathbf{n}$ is the vector of upper bounds on prediction error frequencies and $\|\mathbf{e}\|_{\mathbf{n}}^2 = \sum_{S\in\mathcal{S}} n_S e_S^2$.[1] The modification amounts to (1) including $\mathbf{e}$ as a free variable and (2) regularizing it with a quadratic penalty (second term in Eq. 7) and a linear "reward" (third term in Eq. 7). The linear term selects which e_S should be nonzero—equivalently, are not impossibly difficult. Setting $e_S = 0$ amounts to giving up on S errors.[2]

Most importantly, Eq. 7 is minimized at ease $\mathbf{e}$ that matches Eq. 6. To see this, we define $C(\mathbf{w}, \mathbf{e})$ as the objective minimized in Eq 7, and derive the optimal ease values $\mathbf{e}^*$ such that $\mathbf{0} \in \partial C(\mathbf{w}^*, \mathbf{e}^*)$. The subdifferential ∂C is given by:[3]

$$\begin{aligned}\partial C(\mathbf{w}, \mathbf{e}) = &\frac{\lambda}{2}\nabla(\|\mathbf{w}\|_2^2) + \frac{1}{2}\nabla(\|\mathbf{e}\|_{\mathbf{n}}^2) - \nabla(\mathbf{e}^\top \mathbf{n}) + \\ &\sum_{i=1}^{N} -\nabla(f(x_i, y_i; \mathbf{w})) + \mathbf{Co}(\nabla F^{\Delta}_{\max}(x_i, y_i; \mathbf{w}, \mathbf{e}))\end{aligned} \tag{8}$$

Where addition and subtraction operators are overloaded to perform additions and subtractions over sets, $\mathbf{Co}(\cdot)$ gives the convex hull, and $\nabla F^{\Delta}_{\max}$ is defined as:

$$\nabla F^{\Delta}_{\max}(x, y; \mathbf{w}, \mathbf{e}) = \{\nabla f(x, \hat{y}; \mathbf{w}) + \nabla \mathbf{e}^\top \mathbf{s}(y, \hat{y}) \mid \hat{y} \in \operatorname*{argmax}_{y'\in\mathcal{Y}} f(x, y'; \mathbf{w}) + \mathbf{e}^\top \mathbf{s}(y, y')\} \tag{9}$$

We are only interested in the value of $\mathbf{e}$, so we can consider Eq. 8 reduced to a simpler subdifferential with respect to a single dimension e_S of $\mathbf{e}$:

$$\partial_{e_S} C(\mathbf{w}, \mathbf{e}) = n_S e_S - n_S + \sum_{i=1}^{N} \mathbf{Co}\left(\frac{\partial F^{\Delta}_{\max}(x_i, y_i; \mathbf{w}, \mathbf{e})}{\partial e_S}\right) \tag{10}$$

Given that $\mathbf{0} \in \partial C(\mathbf{w}^*, \mathbf{e}^*)$, Eq 10 leads to:

$$e_S^* \in 1 - \frac{\sum_{i=1}^{N} \mathbf{Co}(\frac{\partial F^{\Delta}_{\max}(x_i, y_i; \mathbf{w}^*, \mathbf{e}^*)}{\partial e_S})}{n_S} \tag{11}$$

[1] $\|\cdot\|_{\mathbf{n}}$ norm is a Mahalanobis norm, as described in [6].

[2] We use the term "giving up" loosely. Setting $e_S = 0$ implies a reversion to a cost-unaware hinge loss as found in the structured perceptron [7]. The learner will become less concerned with S errors, relative to others, but it will still try to improve the score of the observed output. **[PLEASE CHECK THIS –NAS]**

[3] See [8] for background on subgradients.

If we allow the argmax in Eq. 9 to arbitrarily break ties as in Section 3.1, and we constrain $e^*_S \geq 0$ as in Eq. 7, then Eq. 11 suggests that e^*_S takes on the value given by Eq. 6 when the objective is minimized. So our objective chooses values of $\mathbf{e}$ according to our intuitions from Section 3.1.

3.3 Constants $\mathbf{n}$

The appropriate choice for the normalization vector $\mathbf{n}$ in Eq. 7 depends on the prediction classes in $\mathcal{S}$ and the types of bias we seek to avoid when estimating $\mathbf{e}$. For $\mathcal{S}^{\mathrm{u}}$ and $\mathcal{S}^{\mathrm{o}}$, we are most concerned with unbalanced marginal distributions over labels. Let c_y be the frequency of the label y in the training data, $|\{i \mid y_i = y\}|$. We propose two choices of $\mathbf{n}$, both based on the training data:

1. **Logical** $\mathbf{n}$: an upper bound on $v_S(\cdot, \cdot)$ based on frequencies in the training data. For $\mathcal{S}^{\mathrm{u}}$, let $n_{S_{\{y,y'\}}} = c_y + c_{y'}$. For $\mathcal{S}^{\mathrm{o}}$, let $n_{S_{y,y'}} = c_y$ where $S_{y,y'}$ corresponds to an erroneous label of y' in place of the correct y.
2. **Expected** $\mathbf{n}$: an upper bound calculated by assuming that our learner can perform better than a random classifier that uses label proportions observed in the training data. For $\mathcal{S}^{\mathrm{u}}$, let $n_{S_{\{y,y'\}}} = 2c_y c_{y'}/N$. For $\mathcal{S}^{\mathrm{o}}$, let $n_{S_{y,y'}} = c_y c_{y'}/N$.

The **Logical** choice will tend to dramatically overestimate the maximum count of each prediction error, but we might choose it over **Expected** if we believe that the random classifier would not give a baseline rate at which our model is biased to predict certain labels by the label distribution independent of the inputs. A third option, not explored here, might use a more sophisticated model to estimate bounds on error counts.

4 Experiments

We implemented the multiclass SVM (Eq. 3) and variations of our method—which we refer to as normalized cost learning (NCL; Eq. 7)—and we compared each of these implementations on standard text classification datasets in several experiments. In each experiment, we used a standard train/test split for the dataset, but with 10% of the training set held out to perform a grid search for λ over 16 values in $[10^{-6}, 50]$, choosing the value that gives the best accuracy, and then fixing λ and retraining on the whole training set. For training, ran stochastic gradient descent (SGD) with a learning rate determined by AdaGrad for 50 passes over random permutations of the data [9, 10]. We observed that during the last ten passes, accuracy varied by < 0.01 and fewer than 10% of predictions changed.

4.1 Datasets

We considered two datasets with relatively large output label sets: 20 Newsgroups (20NG; 20 category labels corresponding to newsgroups)[4] and Reuters-21578 (R52; 52 topic labels).[5] The 20NG categories are organized hierarchically by the providers of the dataset, as shown in Figure 1a; we refer to the groupings as "topical clusters." We followed [11] in preprocessing the text corpora (including downcasing, removing symbols, etc.), and we let features in $\mathbf{g}$ correspond to tf-idf computed over unigrams [12].

The 20NG dataset consists of 18,846 documents, sorted by date, with 60% used for training. Though the categories are roughly uniformly distributed, the topics vary greatly in their relatedness, following a hierarchical labeling scheme (e.g., *rec.autos* and *rec.motorcycles* are likely more closely related than either to *sci.space*). This offers a way to measure the effectiveness of NCL at learning "ease": the less closely related two categories are, the greater the ease in learning to distinguish them.

The R52 dataset contains 9,100 documents; we use the ModApte split (70% training). The label distribution is skewed, with 43% of documents assigned to *earn* and 37 topics receiving fewer than 50 examples.

[4] `http://qwone.com/~jason/20Newsgroups`

[5] `http://www.csmining.org/index.php/r52-and-r8-of-reuters-21578.html`

Table 1: Micro-averaged accuracies of different learners. The "clusters" column for 20NG corresponds to a coarser score where errors are counted only between high-level topical clusters shown in Figure 1a.

Learner	20NG		R52
	full	clusters	
SVM	0.834	0.920	0.945
NCL: $\mathcal{S}^{\text{o}}$, none	0.834	0.919	0.946
NCL: $\mathcal{S}^{\text{u}}$, none	0.832	0.921	0.945
NCL: $\mathcal{S}^{\text{o}}$, logical	0.836	0.920	0.948
NCL: $\mathcal{S}^{\text{u}}$, logical	0.830	0.919	0.948
NCL: $\mathcal{S}^{\text{o}}$, expected	0.830	0.920	0.949
NCL: $\mathcal{S}^{\text{u}}$, expected	0.820	0.911	0.946

4.2 Results

Table 1 shows the micro-averaged accuracies on the Reuters and 20 newsgroups tasks for the SVM baseline model and NCL with various prediction error classes ($\mathcal{S}^{\text{u}}$; $\mathcal{S}^{\text{o}}$) and normalization constants ($\mathbf{1}$, i.e., none; logical; expected). NCL does not appear to have performed better overall, giving approximately the same accuracy as the SVM. A difference is not even apparent when we consider a less strict accuracy measure that does not penalize errors within higher-level clusters. **[CONDENSED A LOT HERE, PLEASE CHECK –NAS]**

We inspected the learned cost functions from the $\mathcal{S}^{\text{u}}$ versions of NCL to see whether the learned costs reflect the topical groupings suggested by the hierarchy in Figure 1a. To observe the relationship between the learned costs and the topical clusters, we constructed average linkage hierarchical clusters (UPGMA) using the values of learned cost function weights $\mathbf{e}$ as distances [13]. The resulting cost-learned hierarchical clusterings shown in Figures 1b, 1c, and 1d are qualitatively similar to to the topical clusters in Figure 1a.

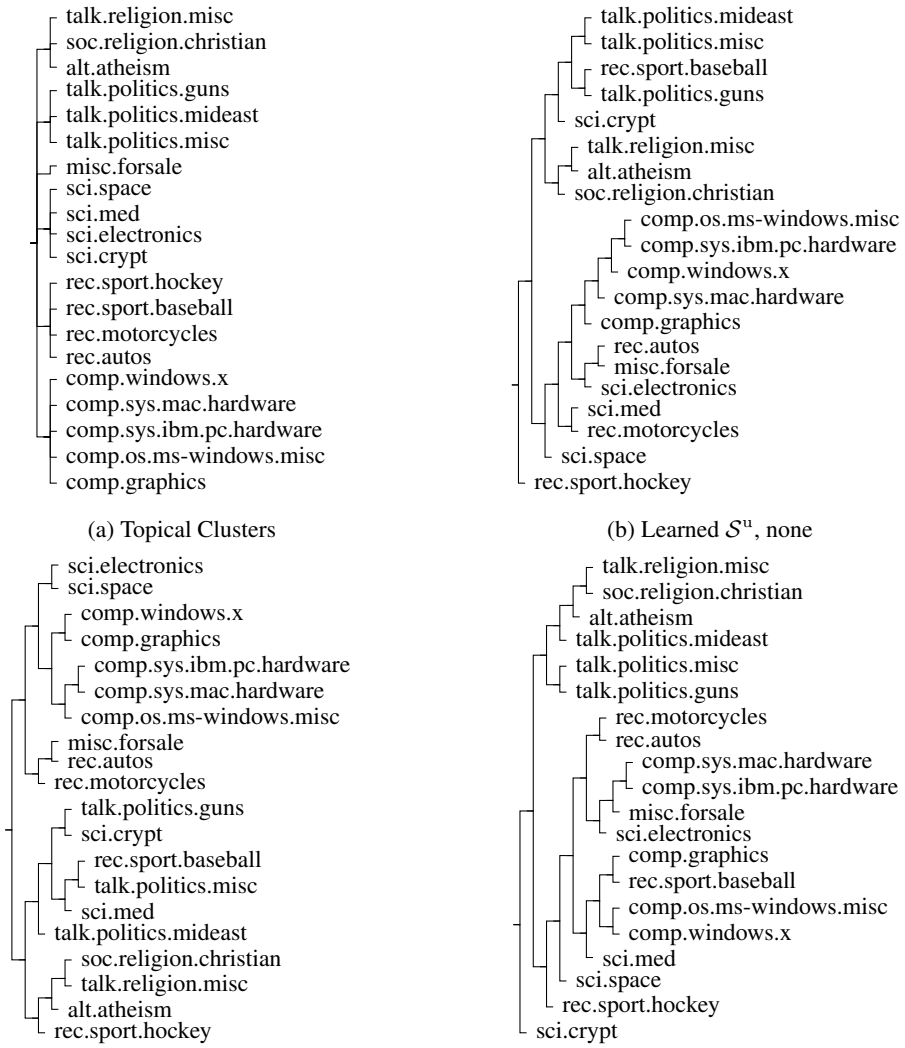
There is evidence that the normalization affects the learned $\mathbf{e}$ as expected from Eq. 6 in that the "expected" normalization constants give components of $\mathbf{e}$ in $[0.759, 1]$, the "logical" constants give component of $\mathbf{e}$ in $[0.907, 1]$, and the "none" constants gives components of $\mathbf{e}$ in $[0.999, 1]$.

4.3 Discussion

Given that NCL's learned cost function seems to align well with the hierarchical nature of the categories, it is surprising that NCL does not give an increased accuracy over the SVM baseline in the "clusters" column of Table 1. This could be an artifact of the 20NG dataset—for example, the SVM's choice of weights might be near optimal for distinguishing between categories in separate topical clusters given its features, and so there might be nothing that NCL can do to improve these weights.

It is also possible that NCL adjusts the weights given the learned costs in a sub-optimal way. The baseline multiclass SVM from [2] generalizes the binary maximum margin-principle from [1] in a way that does not preserve the binary principle's original geometric interpretation. Furthermore, [4] shows both "slack-rescaling" and "margin-rescaling" methods for incorporating the cost function into the SVM, and we've only experimented with the possibly inferior "margin-rescaling" method. Investigations of the generalization from binary to multi-class, the generalization from uniform to varying cost, and the interaction between these might suggest ways to use the cost learning to gain feature weights which generalize better.

Improvements might also be possible with alternative ways to estimate the normalization constants $\mathbf{n}$ or different conditions applied to "ease."



(a) Topical Clusters

(b) Learned $\mathcal{S}^{\mathrm{u}}$, none

(c) Learned $\mathcal{S}^{\mathrm{u}}$, logical

(d) Learned $\mathcal{S}^{\mathrm{u}}$, expected

Figure 1: Clusterings of 20 Newsgroup categories. 1a shows the topical clusters suggested at `http://qwone.com/~jason/20Newsgroups/`. 1b, 1c, and 1d show hierarchical clusterings constructed using distances from the "ease" $\mathbf{e}$ learned by NCL with $\mathcal{S}^{\mathrm{u}}$ prediction classes and "none" ($\mathbf{1}$), "logical," and "expected" normalization constants.

5 Related Work

We believe that the approach proposed here may bear some connection to self-paced learning [14], which provided some direct inspiration, as well as curriculum learning [15], confidence-weighted learning [16], inter-annotator agreement analysis [17], discriminative state-splitting [18], finite-state output encodings [19], and metric learning [20].

6 Future Work

In future work, richer representations of prediction error types ($\mathcal{S}$) might be pursued. For example, types might be constructed based on frequencies of labels, with the rarest labels forming a group. For structured output spaces such as natural language parsing, the domain might suggest groups of errors; post hoc analysis of $\mathbf{e}$ might, in turn, suggest ways to improve the model through feature engineering.[6] Our framework is easily extended to let these classes depend on the input or metadata as well, allowing very rich parameterizations of learnable cost functions. Recall that these classes need not be mutually exclusive.

Alternative ways to estimate $\mathbf{n}$ might also be considered, such as using a more sophisticated model to estimate bounds on error frequencies in the training set. Characterizations of ease might be developed, for example, from stability measure from learning theory [21], which might offer insight into the generalizability of predictions involving a particular label.

We concede that our notion of "ease" merges several concepts that might be treated separately. These include the reliability of the labels in training data, the distinctiveness of the labels given the model family (choice of features), the learnability of each label given the number of instances it has in the training set, and the overall similarity of the training distribution to the "true" one. We believe it is an open theoretical question how these various notions might relate to learning guarantees.

Finally, given our results on the text-classification data, it would be good to experiment with other datasets, perhaps with a larger number of labels, and with richer model families. Synthetic data experiments might elucidate the approach, as well.

Acknowledgments

This work was supported by NSF grant IIS-1054319.

References

[1] Vladimir N. Vapnik. *Statistical Learning Theory*. Wiley, 1998.

[2] Koby Crammer and Yoram Singer. On the algorithmic implementation of multiclass kernel-based vector machines. *The Journal of Machine Learning Research*, 2:265–292, 2002.

[3] Jason Weston and Chris Watkins. Multi-class support vector machines, 1998.

[4] Ioannis Tsochantaridis, Thomas Hofmann, Thorsten Joachims, and Yasemin Altun. Support vector machine learning for interdependent and structured output spaces. In *Proceedings of ICML*, 2004.

[5] Ben Taskar, Carlos Guestrin, and Daphne Koller. Max-margin markov networks. In *NIPS*, 2003.

[6] Prasanta Chandra Mahalanobis. On the generalized distance in statistics. *Proceedings of the National Institute of Sciences (Calcutta)*, 2:49–55, 1936.

[7] Michael Collins. Discriminative training methods for hidden Markov models: Theory and experiments with perceptron algorithms. In *Proceedings of EMNLP*, 2002.

[8] S. Boyd and L. Vandenberghe. Subgradients, 2008. `http://see.stanford.edu/materials/lsocoee364b/01-subgradients_notes.pdf`.

[9] G. George Yin and Harold Joseph Kushner. *Stochastic approximation and recursive algorithms and applications*. Springer, 2003.

[10] John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *The Journal of Machine Learning Research*, 12:2121–2159, 2011.

[6] We note an interesting parallel to the *ceteris paribus* reasoning suggested by inspection of linear model weights $\mathbf{w}$; inspecting $\mathbf{e}$ shows, "all other things equal," a scaling of error types by ease-of-avoidance.

[11] Ana Cardoso-Cachopo. *Improving Methods for Single-label Text Categorization*. PhD thesis, Instituto Superior Tecnico, Universidade Tecnica de Lisboa, 2007.
[12] Man Lan, Chew Lim Tan, and Hwee-Boon Low. Proposing a new term weighting scheme for text categorization. In *Proceedings of AAAI*, 2006.
[13] Pierre Legendre and Loic FJ Legendre. *Numerical ecology*, volume 20. Elsevier, 2012.
[14] M. Pawan Kumar, Benjamin Packer, and Daphne Koller. Self-paced learning for latent variable models. In *NIPS*, 2010.
[15] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of ICML*, 2009.
[16] Mark Dredze, Koby Crammer, and Fernando Pereira. Confidence-weighted linear classification. In *Proceedings of ICML*, 2008.
[17] Barbara Plank, Dirk Hovy, and Anders Søgaard. Learning part-of-speech taggers with inter-annotator agreement loss. In *Proceedings of EACL*, 2014.
[18] Slav Petrov. *Coarse-to-fine natural language processing*. Springer, 2011.
[19] Edward Loper. *Encoding structured output values*. PhD thesis, University of Pennsylvania, 2008.
[20] Eric P. Xing, Michael I. Jordan, Stuart Russell, and Andrew Y. Ng. Distance metric learning with application to clustering with side-information. In *NIPS*, 2002.
[21] Sayan Mukherjee, Partha Niyogi, Tomaso Poggio, and Ryan Rifkin. Learning theory: stability is sufficient for generalization and necessary and sufficient for consistency of empirical risk minimization. *Advances in Computational Mathematics*, 25(1-3):161–193, 2006.