Learning with feature feedback: from theory to practice

Stefanos Poulis
UC San Diego
spoulis@eng.ucsd.edu

Sanjoy Dasgupta
UC San Diego
dasgupta@cs.ucsd.edu

Abstract

In supervised learning, a human annotator only needs to assign each data point (document, image, etc.) its correct label. But in many situations, the human can also provide richer feedback at essentially no extra cost. In this paper, we examine a particular type of *feature feedback* that has been used, with some success, in information retrieval and in computer vision. We formalize two models of feature feedback, give learning algorithms for them, and quantify their usefulness in the learning process. Our experiments also show the efficacy of these methods.

1 INTRODUCTION

In supervised learning, obtaining a labeled training data set can be costly: a human labeler needs to scrutinize each data point and determine its label. One approach to reducing this expense is *active learning*: the learner intelligently and adaptively decides which points should be labeled. There are several active learning methods that work well in practice [Settles, 2012] and enjoy theoretical guarantees [Cohn et al., 1994, Dasgupta, 2011]. Here we consider a strategy complementary to active learning: can the human, while examining the data point, provide not just the label but also the identity of one or more relevant features?

Consider, for example, a document classification problem in which a labeler assigns each document x to a category y ("sports", "politics", and so on). While making this determination, the labeler might also be able to highlight a few words that are highly indicative of the label (e.g. "Congress", "Obama", "filibuster"). Figure 1a illustrates feature feedback. This additional information can often be provided with minimal extra effort and might be useful for learning a classifier. Early work in information retrieval that advocates this auxiliary feedback is that of Croft and Das [1990]. Since then, there have been several experimental studies of different methods for exploiting this feedback [Raghavan et al., 2005, Dayanik et al., 2006, Druck et al., 2008, Raghavan and Allan, 2007, Settles, 2011].

Alternatively, consider a computer vision system that is learning to recognize different animals. Whenever it makes a mistake—classifies a "zebra" as a "horse", say—a human labeler corrects it. While doing this, the labeler can also, at no extra cost, highlight a part of the image (the stripes, for instance) that distinguishes the two animals. Recent work on recognizing different species of birds, for instance, has used this effectively [Branson et al., 2010].

This kind of feedback is not trivial to model. For one thing, it is potentially quite ambiguous. Let's return to the example of a document about "politics" in which the labeler highlights the word "filibuster". This word is, indeed, predictive of the label, but it is also so specific that it might not apply to very many documents. Should "filibuster" be treated as a proxy for a whole collection of words that co-occur with it, or possibly a proxy for an entire *topic*? This seems reasonable, but what is the right level of granularity for the topic, or the cluster of co-occurring words?

Similarly, in the computer vision example, suppose a labeler decides that a bird is a particular type of robin and provides additional feedback by clicking on its breast (whose color, for instance, might be a deciding factor). The learner may have some higher-level representation of the image, for instance a hierarchical parts decomposition, in which case it will in general be unclear which of these features the user is referring to: several features, at different scales, might be candidates.

In both the text classification and vision examples, we see that there is the raw input x (document, image), as well as an intermediate representation z (clusters

Appearing in Proceedings of the 20^{th} International Conference on Artificial Intelligence and Statistics (AISTATS) 2017, Fort Lauderdale, Flordia, USA. JMLR: W&CP volume 54.

of words, hierarchical parts decomposition) that the labeler can not access directly. After deciding on the label y, the labeler indicates one or more coordinates in x; these indirectly and noisily reference a subset of features in z, of which some might be relevant to y and some not. Under this scenario there is some *vagueness* in the labeler's intent: he/she directly acts on coordinates in x whose selection triggers a subset of coordinates in z. Figure 1b illustrates vague feature feedback for the example of document classification.

Contributions. In contrast to the richness of theoretical results for label-based active learning, there is little theoretical work on feature feedback. In this paper we formalize two models of feature feedback and give learning algorithms for them, along with theoretical guarantees. We also provide experimental corroboration of the efficacy of these methods.

The first model we study is a probabilistic generalization of disjunctions. For concreteness, we define this model specifically in the document-topic setting, but it applies more generally to the x-z-y situation described above: the label y of each document x is assumed to be probabilistically generated from the unnamed intermediate-level features z. We call this the *probabilistic disjunction model* (PDM). If we only had documents and labels, we could try to find a maximum-likelihood fit for the generative model, but we show that this is an NP-hard problem. On the other hand, feature feedback makes learning tractable. We give an efficient algorithm that exploits this feedback to learn a PDM. Under simple assumptions, we are able to guarantee the correctness of this algorithm, as well as quantify its label complexity.

The PDM model is not as expressive as linear separators, which are commonly used in document classification. To address this, the second situation we study is learning linear separators from feature feedback. We suggest a novel approach for incorporating information that a particular feature is relevant: reducing the degree of regularization on that feature. This is algorithmically simple, and we show, using seminal results of Bartlett and S.Mendelson [2003] and Kakade et al. [2009], that it leads to better generalization bounds.

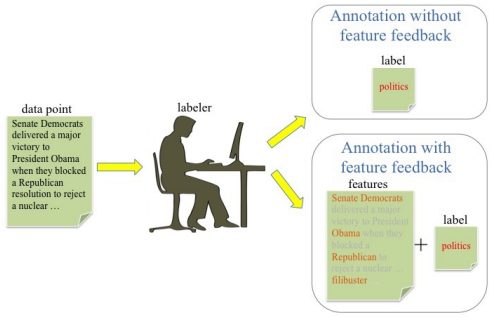
The regularization approach to feature feedback has the drawback of not directly modeling vagueness in the labeler's intent. We incorporate this in a bootstrapped PDM algorithm in which a PDM is first fit to data, using a small amount of feature feedback, and is then used to label whichever documents it is confident about. This augmented training set is then used to train a linear separator (or any other model of interest).

The rest of the paper is organized as follows. In section 2 we review previous work in learning with feature feedback. In section 3 we lay out the problem of learning probabilistic disjunctions. We give our hardness result and in turn, a simple and efficient algorithm for learning a PDM. Then, in section 4 we study linear separators. We first provide an improved generalization bound for feature feedback. We then derive a practical algorithm for learning a linear support vector machine with feature feedback (SVM-FF). To cope with the limitations of the PDM and SVM-FF we give the bootstrapped PDM. In section 6 we present a variety of simulation experiments comparing these methods (PDM, SVM-FF and bootstrapped PDM) on several benchmark text categorization data sets. We then conduct a user study to assess feature feedback in a situation with human annotators and conclude.

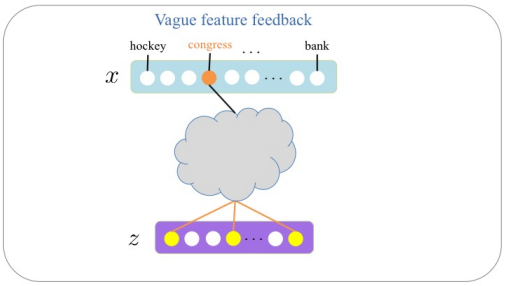
2 RELATED WORK

There is a lot of work on incorporating domain knowledge into learning, for instance by using this knowledge to construct a preliminary classifier or to set Bayesian hyperparameters (Schapire et al. [2002], Wu and Srihari [2004], and Dayanik et al. [2006]).

For feature feedback more specifically, the feedback model closest in spirit to ours is probably that of Druck et al. [2007], whose *generalized expectation criteria* framework incorporates user-supplied feature-label relationships into the objective function for learning. Another line of work develops the idea of *annotator rationales* (Zaidan et al. [2007], Zaidan and Eisner [2008], Donahue and Grauman [2011]), in which the labeler highlights regions of the document that serve as explanations of the label; these are then used to generate *contrast examples* (same document, but with these regions removed) and the learning procedure asks for each document to be distinguished from its contrasting version. This framework involves denser annotation than we have in mind. A related form of "contrast example" is considered by Sun and DeJong [2005], who incorporate this into an SVM framework and provide generalization bounds—though these are weaker and less general than our bounds, which have less requirements on the feedback and apply to any linear model. Later work by Small et al. [2011] developed the *constrained weight-space* SVM framework by allowing annotators to provide ranked features. The final research thread in the topic includes work developed in Melville et al. [2004, 2005], Raghavan et al. [2006], Sindhwani et al. [2009], where active models are considered to incorporate feature feedback into learning. The framework there is to identify the *most informative features* to be shown to the human, when asked to label an example.



(a) Annotation with and without feature feedback.



(b) Vague feature feedback: selecting a word in x indirectly and noisily triggers a subset of the latent features z.

Figure 1: Models of Feature Feedback

In the above works feature feedback is explicit: information about particular (feature,label) associations does not propagate to others. With the exception of Sun and DeJong [2005], there is also a lack of theoretical analysis of the efficacy of the various methods.

3 A PROBABILISTIC DISJUNCTION MODEL (PDM)

In this section, we define a stochastic model that generates the label $y \in \{1, 2, \ldots, k\}$ of any document d. The model makes use of an intermediate-level representation that, for concreteness, we think of as referring to topics.

Suppose we have a set of T "topics" as well as a procedure for representing any document as a convex combination of these topics. The details of how this is done are irrelevant. We will assume that every topic $t \in \{1, 2, \ldots, T\}$ either has an associated label $\ell(t) \in \{1, 2, \ldots, k\}$ or has $\ell(t) = ?$. In the former case, the topic is a strong predictor of the corresponding label. In the latter case, the topic is ambiguous, for instance, an overly general topic. We will denote the set of predictive topics as $P = \{t : \ell(t) \neq ?\}$ and we will assume that every document assigns non-zero probability to at least one predictive topic.

The stochastic model for labeling a document works like a probabilistic disjunction:

- Let $\theta = (\theta_1, \ldots, \theta_T)$ be the topic representation of the document.
- Pick a predictive topic at random: choose $t \in P$ with probability proportional to θ_t.
- The label of the document is $\ell(t)$.

Suppose there is no feature feedback; that is, the learner has access only to a collection of (document, label) pairs. A reasonable objective, under the above stochastic model, is to find the assignment $\ell : \{1, 2, \ldots, T\} \to \{1, 2, \ldots, k, ?\}$ that maximizes the likelihood of the data. But we can show that merely finding an assignment with non-zero likelihood is NP-hard.

Theorem 1. *The following problem is NP-complete: Given a collection of labeled documents, where each document is represented as a distribution over topics, and where $k = 2$ (binary labels), find an assignment $\ell : [T] \to \{0, 1, ?\}$ with non-zero likelihood.*

(Proof in Section A.1.) Feature feedback makes this intractability go away, as we will see next.

3.1 Learning a PDM with Feature Feedback

The interactive labeling process works as follows: (a) The labeler gets a batch of (say) 10 documents. (b) For each document: he/she assigns it a label and chooses a predictive word (or maybe several words). This is then repeated until the budget for human interaction runs out.

The goal of the learner is to identify the correct mapping $\ell : [T] \to \{1, 2, \ldots, k, ?\}$. A scheme for doing this is shown in Algorithm 1. Roughly, when the user identifies a document as having label y and relevant words $w_1, \ldots, w_c$, the algorithm picks a set of topics $S \subseteq [T]$ triggered by these words and increments a counter n_{ty} for each $t \in S$. This n_{ty} is a count of how many times the user has suggested that topic t is predictive of label y.

The specific mechanism for choosing the set S based on the feedback, corresponding to the function `select-topics` in the pseudocode, is not relevant for the theoretical results we establish below. Here is the strategy we use in our experimental work: given feedback words $w_1, \ldots, w_c$ for document x, obtain topic

distributions for each of these words in the context of document x; call these $p_1, \ldots, p_c$. Add topic t to the selected set S if $(p_1(t) + \cdots + p_c(t))/c$ exceeds some predefined threshold.

Algorithm 1 Probabilistic Disjunction Model (PDM)

Input: Collection of unlabeled documents U
Initialize: $n_{ty} = 0, \forall t, y$
Labeled data set $L = \emptyset$
repeat
  Draw next batch $B \subset U$ of documents at random
  $U = U \setminus B$
  for each document $x \in B$ **do**
    Receive label y, relevant words $w_1, \ldots, w_c$
    Add (x, y) to L
    $S =$ select-topics$(x, w_1, \ldots, w_c)$
    for $t \in S$ **do**
      $n_{ty} = n_{ty} + 1$
    end for
  end for
until budget runs out

Assigning a label to each topic. This is summarized in Algorithm 2. The total amount of feedback received for topic t is $n_t = \sum_y n_{ty}$. If this exceeds some fixed amount n_o, and moreover there is a specific label y for which $n_{ty} \geq \lambda n_t$, then we assign $\widehat{\ell}(t) = y$. Here λ is a fixed fraction. In all other cases, we set $\widehat{\ell}(t) =?$.

Labeling a new document. This prediction rule is shown in Algorithm 3. Once topics are labeled, the estimated set of predictive topics is $\widehat{P} = \{t : \widehat{\ell}(t) \neq ?\}$. Let θ be the topic distribution for the new document. The conditional probability that this document has label y can be estimated as $\frac{\sum_{t:\widehat{\ell}(t)=y} \theta_t}{\sum_{t \in \widehat{P}} \theta_t}$.

3.2 Theoretical Guarantees

Correctness of topic labeling. In order to show that the topic labeling algorithm recovers the true labels $\ell(t)$ with high probability, we do not need the full strength of the PDM assumption. What we require is that the topics selected by the user are not systematically misleading. On each round, the machine associates a set of user-selected topics S with a label y. Some of these associations may be spurious, for instance, due to polysemy that the user inadvertently overlooks. But the same spurious associations should not occur repeatedly.

To formalize this, first observe that the two sources of randomness in topic labeling are: (1) the uniform distribution over documents in the corpus, and (2) the possibly stochastic mechanism by which the human selects helpful words from a document.

Assumption 1. *For any topic t and any label $y \neq \ell(t)$, if we pick a document at random, ask the human for the label and for helpful words, and look at the induced set of selected topics,*

$$\Pr(\textit{label} = y \mid \textit{topic } t \textit{ is selected}) \leq \lambda/2.$$

Meanwhile, for any predictive topic $t \in P$,

$$\Pr(\textit{label} = \ell(t) \mid \textit{topic } t \textit{ is selected}) \geq 2\lambda.$$

Algorithm 2 Topic labeling assignment (TLA)

Input: $n_{ty} \forall t, y, \lambda, n_o$
for each topic t **do**
  $\widehat{\ell}(t) =?$
  $n_t = \sum_y n_{ty}$
  if $n_t \geq n_o$ **then**
    $y = \arg\max_{y'} n_{ty'}$
    if $n_{ty} \geq \lambda n_t$ **then**
      $\widehat{\ell}(t) = y$
    end if
  end if
end for

Algorithm 3 PDM prediction rule

Input: Topic representation $\theta \in [0,1]^T$ of document d
Initialize: $\pi = 0^k$
Label topics according to TLA (Algorithm 2)
for each topic t **do**
  if $\widehat{\ell}(t) \neq ?$ **then**
    $\pi(\widehat{\ell}(t)) \leftarrow \pi(\widehat{\ell}(t)) + \theta_t$
  end if
end for
Normalize π to sum to 1

Theorem 2. *Pick any $0 < \delta < 1$. Suppose Assumption 1 holds and that we set $n_o \geq (6/\lambda)\ln(Tk)/\delta$. Then with probability at least $1 - \delta$, for all $t \in [T]$ with $n_t \geq n_o$, we have $\widehat{\ell}(t) = \ell(t)$.*

(Proof in Section A.2.)

Label complexity. In order to quantify the amount of feedback needed to recover the true labels ℓ, we require that the user doesn't systematically avoid any informative topics, as follows.

Assumption 2. *There is an absolute constant c_o for which the following holds. Pick any t, y such that $\ell(t) = y$. Then for any document with topic distribution θ and label y, if we solicit feature feedback and look at the induced set of topics,*

$$\Pr(\textit{topic } t \textit{ is selected}) \geq c_o \frac{\theta_t}{\sum_{t':\ell(t')=y} \theta_{t'}}.$$

Let $\theta(x) = (\theta_1(x), \ldots, \theta_T(x))$ be the topic distribution for any document x. We define the *prevalence* of a predictive topic $t \in P$ as

$$\gamma_t = \mathbb{E}_x \left[\frac{\theta_t(x)}{\sum_{t' \in P} \theta_{t'}(x)} \right],$$

where the expectation is over a uniform-random choice of x from the corpus. Roughly, γ_t tells us how common topic t is relative to other predictive topics, and thereby how easy it is to estimate $\ell(t)$.

Theorem 3. *Suppose documents are labeled according to the PDM process. Under Assumption 2, for any $t \in P$, the expected number of labels needed for $\ell(t)$ to be set is at most $n_o/(c_o\gamma_t)$.*

(Proof in Section A.3.) For fixed constants λ and δ, we need $n_o = O(\ln Tk)$. If all predictive topics are equally prevalent then they each have $\gamma_t = 1/|P|$. In this case, the number of rounds of interaction needed is $O(|P| \ln(Tk))$. This shows the benefit of feature feedback when only a small fraction of the topics are predictive (that is, $|P| \ll T$).

4 LEARNING LINEAR SEPARATORS WITH FEATURE FEEDBACK

We now study feature feedback in the setting where the goal is to learn a linear classifier by minimizing a loss function and a regularization penalty. Given a data set $\{(x_i, y_i)\}_{i=1}^n \subset \mathbb{R}^p \times \mathcal{Y}$, the optimization is:

$$\widehat{w} = \arg\min_w \frac{1}{n} \sum_{i=1}^n \ell(w \cdot x_i, y_i) + \lambda \|w\|^2,$$

where $\ell(\cdot)$ is a loss function and $\|\cdot\|$ is some norm. For SVMs, for instance, ℓ is the hinge loss and $\|\cdot\|$ is the 2-norm.

We suggest a novel approach to incorporating information about relevant features: reduce the regularization along those specific dimensions. To achieve this, we take the regularization norm $\|\cdot\|$ to be a *Mahalanobis* norm, given by a $p \times p$ positive definite matrix A:

$$\|x\|_A = \sqrt{x^T A x} = \|A^{1/2} x\|_2.$$

In the absence of feature feedback, A is the identity matrix I_p, giving the 2-norm. But if we find that features $R \subset [p]$ are relevant, we downweight the diagonal matrix in those dimensions: we set $A_{jj} = 1/c$ for relevant features j and $A_{jj} = 1$ otherwise, for some $c > 1$.

We next study the statistical benefit of this estimator.

4.1 Improved Generalization Error Bounds

Let's start with a generalization bound for learning linear classifiers chosen from some set $\mathcal{F}$. Write the empirical loss function as

$$\widehat{\mathcal{L}}(w) = \frac{1}{n} \sum_{i=1}^n \ell(w \cdot x_i, y_i)$$

(regularization is incorporated by restricting $\mathcal{F}$ to vectors of bounded norm). When the training data (x_i, y_i) comes i.i.d. from an (unknown) underlying distribution, the following seminal result shows the relation of $\widehat{\mathcal{L}}(w)$ to the true loss $\mathcal{L}(w) = \mathbb{E}_{x,y}\ell(w \cdot x, y)$:

Theorem 4. *[Bartlett and S.Mendelson, 2003] Suppose the loss function ℓ is Lipschitz in its first argument and is upper-bounded by a constant M_ℓ. Then for any $\delta > 0$, with probability $\geq 1 - \delta$ over the choice of data,*

$$\forall f \in \mathcal{F}: \quad \mathcal{L}(f) \;\leq\; \widehat{\mathcal{L}}(f) + 2R_n(\mathcal{F}) + M_\ell \sqrt{\frac{\log 1/\delta}{2n}},$$

where $R_n(\mathcal{F})$ is the Rademacher complexity of $\mathcal{F}$.

The key term here is $R_n(\mathcal{F})$. In our setup, let w^* be a sparse target classifier of interest and define a feature as being relevant if it is set in w^*. Using a powerful result of [Kakade et al., 2009], we obtain the following.

Theorem 5. *Let $R = \{j \in [p] : w_i^* \neq 0\}$ denote the relevant features of w^*.*

- *We can write any x in terms of its relevant and other components, $x = (x_R, x_o)$.*
- *Let A be the diagonal matrix whose jth entry is $1/c$ if $j \in R$ and 1 otherwise.*

Then, for the family of linear separators $\mathcal{F} = \{w : \|w\|_A \leq \|w^*\|_A\}$, we have

$$R_n(\mathcal{F}) \;\leq\; \|w^*\|_2 \cdot \max_{x \in \mathcal{X}} \sqrt{\left(\frac{1}{c}\|x_o\|_2^2 + \|x_R\|_2^2\right)} \sqrt{\frac{2}{n}}.$$

(Proof in Section B.1.) In situations where the x_o (the irrelevant portion of the data) has significant norm, this downweighting by a factor of c substantially reduces the generalization error bound. We note that our approach to reducing the degree of regularization is analogous to increasing the prior on these features in a Naive Bayes model, as was done in Settles [2011].

4.2 Practical Linear Models with Feature Feedback

Given training data $\{(x_i, y_i)\}_{i=1}^n \subset \mathbb{R}^p \times \mathcal{Y}$ consider the SVM problem with our Mahalanobis regularizer:

Algorithm 4 SVM with feature feedback (SVM-FF)

Input: $c < 1$, unlabeled data set U
Initialize: $L = \emptyset, S = \emptyset, A = I_p$
repeat
  Draw next batch $B \subset U$ of documents
  $U = U \setminus B$
  for each document $x \in B$ **do**
    Receive label y, words $s = \{w_1, \ldots, w_d\}$
    Add (x, y) to L
    for $j \in s$ **do**
      $A_{jj} = c$
      Add j to S
    end for
    Train linear SVM on $\{(A^{-1/2}x, y) : (x, y) \in L\}$
  end for
until budget runs out

Algorithm 5 Bootstrap PDM

Input: Unlabeled data set U, τ_0 (optionally, $c < 1$)
Initialize: $L = \emptyset$ (optionally, $A = I_p$)
repeat
  Draw next batch $B \subset U$ of documents
  $L = L \cup B$; $U = U \setminus B$
  Train PDM (Algorithm 1) on L
  (optionally, update A as in Algorithm 4)
  for each document $x \in U$ **do**
    $I = \emptyset$ (documents with inferred labels)
    Predict $\pi(\cdot)$ over labels according to Algorithm 3
    Predict $\widehat{y} = \arg\max_{y' \in \{1,\ldots,k\}} \pi(y')$
    if $\pi(\widehat{y}) \geq \tau_0$ **then**
      Add $(x, \widehat{y})$ to I
    end if
  end for
  Train any classifier on $\{(x, y) : (x, y) \in L \cup I\}$
  (optionally, train linear SVM as in Algorithm 4)
until budget runs out

$$\begin{aligned} \underset{w}{\text{minimize}} \quad & \frac{1}{2}\|w\|_A^2 + C\sum_{i=1}^{N}\xi_i \\ \text{subject to} \quad & \xi_i \geq 0, y_i(x_i^T w) \geq 1 - \xi, \forall i. \end{aligned}$$

A straightforward derivation shows the following.

Lemma 6. *Pick any positive definite $p \times p$ matrix A. Then, learning a linear SVM on instances $\{(x_i, y_i)\}_{i=1}^n$ with Mahalanobis regularizer $\|w\|_A$ is equivalent to learning a linear SVM on modified instances $\{(A^{-1/2}x_i, y_i)\}_{i=1}^n$ with $\|w\|_2$ regularization.*

(Proof in Section C.) An SVM algorithm with feature feedback (SVM-FF) is given in Algorithm 4. For each supplied feature, the corresponding diagonal entries of A are set to a particular value $c < 1$ and every labeled and unlabeled example is weighted by $A^{-1/2}$. Then, a standard linear SVM is trained on the weighted labeled instances.

5 BOOTSTRAPPING THE PDM

The feedback in the regularization approach is explicit: the regularization will only be applied to features that the labeler selects. Let's return to the "filibuster"–"politics" example in the introduction. Even though the word "filibuster" is a good predictor for "politics" it is a fairly uncommon word. Hence, not that many documents will be affected by reducing the regularization on it. On the other hand, vague feature feedback facilitated by the PDM is richer: feedback on "filibuster" propagates to other words in the same topic. To incorporate vague feedback into a linear classifier, we introduce the *bootstrapped* PDM (Algorithm 5). Given a labeled data set L and an unlabeled data set U, the algorithm fits a PDM on L and uses this fitted PDM to predict on U. It then infers the labels of a set $I \subseteq U$ of data points for which it is confident. We say that the PDM is confident on an instance x if its prediction $\widehat{y}$ has estimated conditional probability $\pi(\widehat{y}) \geq \tau_0$ (recall the notation of Algorithm 3), where τ_0 is a parameter to be set. One can then train any classifier on $L \cup I$. If the classifier of choice is a linear SVM, one can apply the mixed regularization, by multiplying every example by $A^{-1/2}$ and training a linear SVM on this weighted data set of labeled and inferred points.

6 EXPERIMENTS

We conducted experiments on the following 6 benchmark text categorization data sets. **20 NewsGroups**: Set of approximately 20,000 documents, partitioned evenly across 20 newsgroups, containing postings about politics, sports, technology, religion, science etc. **Reuters-21578**: The most widely used collection for text categorization research. Documents with less than or with more than one label were eliminated, resulting in **R8** (8 classes) and **R52** (52 classes). **webkb**: Data set that contains web pages collected from computer science departments of various universities. **cade**: Web pages from the CADE Web Directory, which points to Brazilian web pages classified by human experts in 12 classes, including services, education, sciences, sports, culture etc. **ohsumed**: Medical abstracts from the MeSH (Medical Subject Headings) data set, belonging to 23 cardiovascular disease categories. For further details on the data sets, see section D.1 of the Appendix. The first five data sets were already processed [Cardoso-Cachopo, 2007]; we processed **ohsumed** in the same manner (stemming, removal of stop words and words shorter than two characters). As we are interested in single label documents, we only kept data points that had only one label. For each document we obtained its tf-idf and topic representations. For the latter we trained a Latent Dirichlet Allocation model using the

collapsed Gibbs sampler [Griffiths and Steyvers, 2004]. The number of topics was 10 times the number of classes in each data set.

Oracle features. To simulate the labeler's feedback, we first generated a list of *oracle* features for each class as follows. We first trained a logistic regression classifier with ℓ_1 regularization and took all the feature weights that were positive. We then looked at the level of correlation between these features and the class labels. Specifically, for various thresholds α, we considered feature j as feedback for class k if $P(k|j)$, the conditional probability of label k given the presence in the document of word j, was at least α. We then tested our models for various values of α. Feature feedback on a document applied if it contained any of the words in the list of its label. An example of feature feedback for the **20ng** dataset using the PDM is shown in figure 4 in the appendix.

Experimental setup. We compared our models to a linear SVM without feedback. To choose the cost C of all SVM classifiers, we only tuned the SVM without feedback by optimizing the macro-F_1 score on the grid $\{1, 10, 100, 1000\}$. We then set C for the SVM-FF and bootstrap PDM models to that value. On the first few batch iterations we used 2-fold cross validation and continued with 5-fold in later iterations. We set the rest of the parameters for PDM, SVM-FF, and bootstrap PDM as follows: $\lambda = \frac{1}{10}$, $n_o = 2$, $c = \frac{1}{20}$ and $\tau_0 = .75$.

Discussion of simulation results. Figures 2 (a-c) show learning curves for the first 500 data points for each training data set, divided into 20 batches. For each batch iteration, we report macro-F_1 score on the test set. (See D.2 for a more detailed exposition of the experimental results.) Across the board, we find that feedback on a few predictive words helps significantly. To get a feel of the *amount* of feature feedback see figures 11- 12 in D.2. Vague feature feedback (PDM, bootstrap PDM) is particularly helpful when the labeled data set is small. Generous feature feedback (i.e. $\alpha \geq .5$) helps fast convergence when data are scarce but has a somewhat adverse effect when plenty of labeled samples are available. However, this improves for $\alpha \geq .9$. Interestingly, in addition to its superior performance, SVM-FF produces a solution that is much sparser than that of the SVM, as seen in figure 2d. This makes sense intuitively, as feature feedback helps the learning algorithm to focus on important dimensions.

Small *vs* large data regimes. The simulation results illustrate that the benefits of feature feedback diminish asymptotically. We note that since we are learning a linear classifier, in the limit of enough labeled data, we can simply run SVM. Also, the degree of regularization in the SVM-FF can be adjusted so that $c \to 1$ as the sample grows. Hence, our methods are well suited to–the fairly common–situation where the amount of labeled data is limited.

Human experiment. To get a sense of the feature feedback that humans tend to provide and to quantify the difference in the benefits of a *selected* feature *vs* a *random* feature, we conducted a small human study involving 5 annotators. We considered a subset of the **20ng** data set that included points with classes *talk.politics.mideast*, *comp.graphics*, *sci.med*, *rec.autos* and *misc.forsale*. The annotators provided the labels of a randomly chosen set of 50 points along with a number of features via an interface. (See D.3 for details). For class k, call S_k, N_k the set of features that annotators selected and did not select, respectively. In table 1 we show $\bar{p}_{S_k} = \frac{1}{|S_k|}\sum_{j\in S_k} P(k|j)$ and $\bar{p}_{N_k} = \frac{1}{|N_k|}\sum_{j\in N_k} P(k|j)$, where the $P(k|j)$'s are the conditional probabilities described earlier.

Table 1: Results of Human Experiment

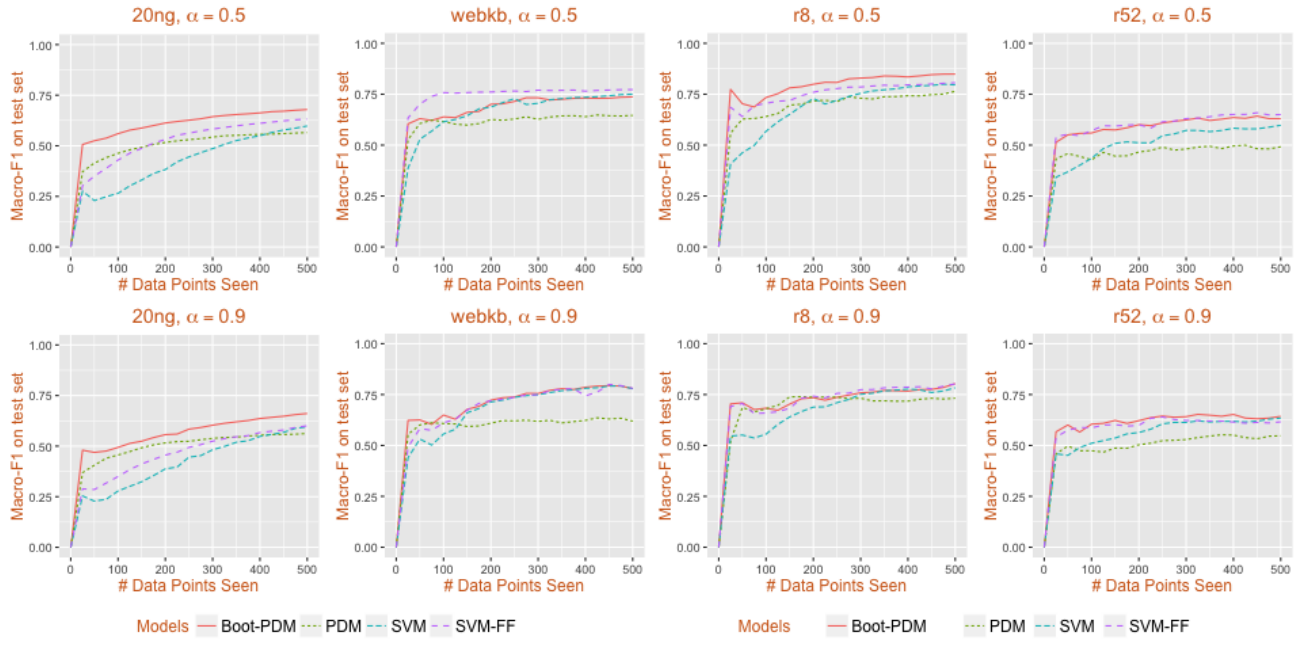
	$\bar{p}_{S_k}$	$\bar{p}_{N_k}$
misc.forsale	0.63	0.76
rec.autos	0.95	0.82
sci.med	0.96	0.78
comp.graphics	0.83	0.66
talk.politics.mideast	0.98	0.74

Note that $\bar{p}_{S_k}$ is smaller than $\bar{p}_{N_k}$ only for the class *misc.forsale* because some annotators confused documents about items for sale with documents with class *comp.graphics* and *rec.autos*. This is not a surprising effect and we expect to diminish with more labeled data and with a larger pool of annotators. Across the board, we find that humans tend to provide words that are highly predictive of the label.

Conclusion

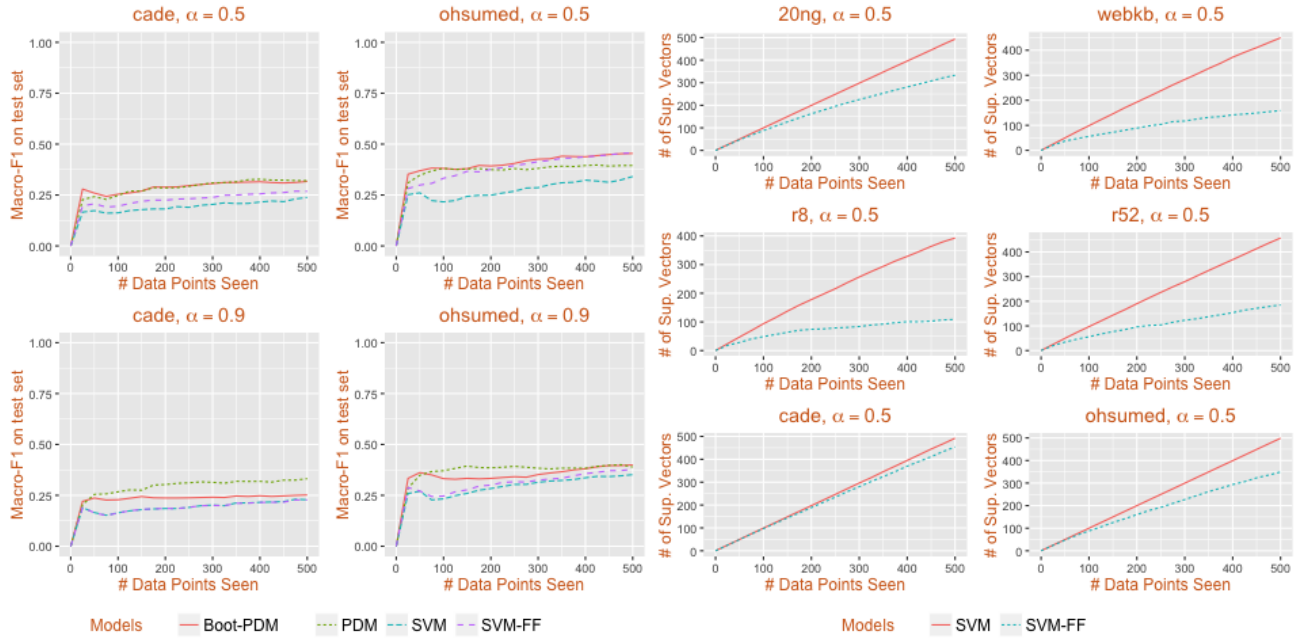
In this paper, we formalized a problem that has been largely studied empirically. We introduced vague feature feedback, a novel approach that models the ambiguity that is inherent in the intent of the labeler. We quantified the benefits of this feedback and demonstrated improvements in small data regimes.

There are several directions following our work. We propose to study models of feature feedback that (i) operate in the *active learning* setting (ii) are adaptable to the varying levels in the quality of the feedback. Also, we propose to study models that make use of feature feedback that helps to *discriminate* between classes.



(a) 20ng-Webkb

(b) R8-R52



(c) cade-ohsumed

(d) Number of Support vectors of SVM vs SVM-FF

Figure 2: (a) to (c): Learning Curves at Different Values of α. (d): Number of Support Vectors.

Acknowledgements

The authors are grateful [to the funding], for feedback from the anonymous reviewers, and for help with the human experiment from Julaiti Alafate, Zack Lipton, Christopher Tosh and Sharad Vikram.

References

P. Bartlett and S.Mendelson. Rademacher and gaussian complexities: Risk bounds and structural results. *The Journal of Machine Learning Research*, 3:463–482, 2003.

S. Branson, C. Wah, B. Babenko, F. Schroff, P. Welinder, P. Perona, and S. Belongie. Visual recognition with humans in the loop. In *European Conference on Computer Vision*, 2010.

A. Cardoso-Cachopo. *Improving methods for single-label text categorization*. PhD thesis, Instituto Superior Tecnico, Universidade Tecnica de Lisboa, 2007.

D. Cohn, L. Atlas, and R. Ladner. Improving generalization with active learning. *Machine Learning*, 15 (2):201–221, 1994.

W. Croft and R. Das. Experiments with query acquisition and use in document retrieval systems. In *Proceedings of the 13th International Conference on Research and Development in Information Retrieval*, pages 349–368, 1990.

S. Dasgupta. Two faces of active learning. *Theoretical Computer Science*, 412(19):1767–1781, 2011.

A. Dayanik, D. Lewis, D. Madigan, V. Menkov, and A. Genkin. Constructing informative prior distributions from domain knowledge in text classification. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 493–500, 2006.

J. Donahue and K. Grauman. Annotator rationales for visual recognition. In *2011 IEEE International Conference on Computer Vision*, pages 1395–1402, 2011.

G. Druck, G.Mann, and A. McCallum. Reducing annotation effort using generalized expectation criteria (technical report 2007-62). *University of Massachusetts, Amherst*, 2007.

G. Druck, G. Mann, and A. McCallum. Learning from labeled features using generalized expectation criteria. In *Proceedings of ACM Special Interest Group on Information Retrieval*, 2008.

T. Griffiths and M. Steyvers. Finding scientific topics. *Proceedings of the National Academy of Sciences*, 101 (suppl 1):5228–5235, 2004.

S. Kakade, K. Sridharan, and A. Tewari. On the complexity of linear prediction: Risk bounds, margin bounds, and regularization. In *Advances in Neural Information Processing Systems*, pages 793–800, 2009.

D. Lewis, Y. Yang, T. Rose, and F. Li. Rcv1: a new benchmark collection for text categorization research. *Journal of Machine Learning Research*, 5:361–397, 2004.

P. Melville, M. Saar-Tsechansky, F. Provost, and R. Mooney. Active feature-value acquisition for classifier induction. In *Fourth IEEE International Conference on Data Mining*, pages 483–486, 2004.

P. Melville, M. Saar-Tsechansky, F. Provost, and R. Mooney. An expected utility approach to active feature-value acquisition. In *Fifth IEEE International Conference on Data Mining*, pages 4–pp, 2005.

H. Raghavan, O. Madani, and R. Jones. Interactive feature selection. In *Proceedings of the 19th International Joint Conference on Artificial Intelligence*, pages 841–846, 2005.

H. Raghavan, O. Madani, and R. Jones. Active learning with feedback on features and instances. *The Journal of Machine Learning Research*, 7:1655–1686, 2006.

Hema Raghavan and James Allan. An interactive algorithm for asking and incorporating feature feedback into support vector machines. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 79–86. ACM, 2007.

R. Schapire, M.Rochery, M. Rahim, and N. Gupta. Incorporating prior knowledge into boosting. In *Proceedings of the 19th International Conference on Machine Learning*, volume 2, pages 538–545, 2002.

B. Settles. Closing the loop: fast, interactive semi-supervised annotation with queries on features and instances. In *Empirical Methods in Natural Language Processing*, 2011.

B. Settles. Active learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 6(1): 1–114, 2012.

V. Sindhwani, P. Melville, and R. Lawrence. Uncertainty sampling and transductive experimental design for active dual supervision. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 953–960, 2009.

K. Small, B. Wallace, T. Trikalinos, and C. Brodley. The constrained weight space svm: learning with ranked features. In *Proceedings of the 28th International Conference on Machine Learning*, pages 865–872, 2011.

Q. Sun and G. DeJong. Explanation-augmented svm: an approach to incorporating domain knowledge into

svm learning. In *Proceedings of the 22nd International Conference on Machine Learning*, pages 864–871, 2005.

X. Wu and R. Srihari. Incorporating prior knowledge with weighted margin support vector machines. In *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 326–333, 2004.

O. Zaidan and J. Eisner. Modeling annotators: A generative approach to learning from annotator rationales. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 31–40, 2008.

O. Zaidan, J. Eisner, and C. Piatko. Using "annotator rationales" to improve machine learning for text categorization. In *Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings*, pages 260–267, 2007.

Appendix

A Theoretical results for the probabilistic disjunction model

A.1 Proof of Theorem 1

Proof. The problem is clearly in NP. To show hardness, we will use a reduction from 3SAT.

Given a 3SAT instance $\phi(x_1, \ldots, x_q) = C_1 \wedge C_2 \wedge \cdots \wedge C_p$, where each clause C_j is a disjunction of three literals, create the following topic labeling problem:

- There are $2q$ topics: $t_1, \ldots, t_q, t'_1, \ldots, t'_q$. Think of t_i as corresponding to the positive literal x_i and t'_i the negative literal $\overline{x_i}$.
- For each variable x_i, create a document d_i whose topic distribution has probability $1/2$ on t_i and on t'_i and zero elsewhere.
- For each clause C_j, create a document d'_j that puts $1/3$ probability on (the t_i or t'_i corresponding to) each of the literals in C_j.
- The data set consists of document-label pairs $(d_i, 0), (d_i, 1), (d'_j, 1)$: a total of $p + 2q$ labeled documents.

Now, suppose there is an assignment $\ell : \{t_1, \ldots, t_q, t'_1, \ldots, t'_q\} \to \{0, 1, ?\}$ with nonzero likelihood. Then for each labeled document (d, y) there is at least one topic t such that $\theta_t^{(d)} > 0$ and $\ell(t) = y$. Now, document d_i appears with label 0 as well as with label 1. Therefore, one of $\ell(t_i), \ell(t'_i)$ must be 0 and one of them must be 1. If $\ell(t_i) = 0, \ell(t'_i) = 1$, we will assign $x_i = 0$. If $\ell(t_i) = 1, \ell(t'_i) = 0$, we will assign $x_i = 1$. To see that this is a satisfying assignment, pick any clause C_j. The corresponding document d'_j has label 1; therefore at least one of the three topics corresponding to its literals must be assigned label 1 under $\ell(\cdot)$. Hence that literal is assigned a value of 1.

Conversely, if ϕ is satisfiable, then the mapping

$$\ell(t_i) = 0, \ell(t'_i) = 1 \text{ if } x_i = 0$$
$$\ell(t_i) = 1, \ell(t'_i) = 0 \text{ if } x_i = 1$$

has nonzero likelihood. □

A.2 Proof of Theorem 2

Proof. First, fix any t, y with $\ell(t) \neq y$. Under Assumption 1, each time topic t is selected, there is less than a $\lambda/2$ probability that the label is y. Conditioned on n_t, the expected value of n_{ty} is therefore at most $\lambda n_t/2$, and by a multiplicative Chernoff bound,

$$\Pr(n_{ty} \geq \lambda n_t) \leq e^{-n_t \lambda/6},$$

which is $\leq \delta/(Tk)$ if $n_t \geq n_o$.

Likewise, for any predictive feature $t \in P$, the expected value of $n_{t,\ell(t)}$ is at least $2\lambda n_t$. Again using a multiplicative Chernoff bound,

$$\Pr(n_{t,\ell(t)} < \lambda n_t) \leq e^{-n_t \lambda/6}.$$

Taking a union bound over all pairs $(t, y) \in [T] \times [k]$, we conclude that with probability at least $1 - \delta$, the following holds whenever $n_t \geq n_o$:

- If $y \neq \ell(t)$ then $n_{ty} < \lambda n_t$.
- If $t \in P$ then $n_{t,\ell(t)} \geq \lambda n_t$.

Therefore, $\widehat{\ell}(t) = \ell(t)$ for $t \in P$ and ? otherwise. □

A.3 Proof of Theorem 3

Proof. Pick any predictive topic $t \in P$, and let $y = \ell(t)$. For a document x chosen at random,

$$\begin{aligned}\Pr_x(\text{topic } t \text{ selected}) &\geq \Pr_x(\text{document label} = y)\Pr_x(\text{topic } t \text{ selected} \mid \text{document label} = y) \\ &\geq \mathbb{E}_x\left[\frac{\sum_{t':\ell(t')=y}\theta_{t'}(x)}{\sum_{t'\in P}\theta_{t'}(x)} \cdot c_o \frac{\theta_t(x)}{\sum_{t':\ell(t')=y}\theta_{t'}(x)}\right] \;=\; c_o\gamma_t.\end{aligned}$$

Therefore, the expected number of documents that need to be seen before n_t reaches n_o is at most $n_o/(c_o\gamma_t)$. □

B Incorporating feature feedback through regularization

B.1 Proof of Theorem 5

Recall that we wish to bound $R_n(\mathcal{F})$. The powerful results of [Kakade et al., 2009] achieve this for a wide range of cases: for any $\mathcal{F} = \{w : \|w\| \leq W\}$, where $\|\cdot\|$ satisfies a strong convexity property. Specifically, they show

$$\mathbb{R}_n(\mathcal{F}) \;\leq\; W \cdot \max_{x\in\mathcal{X}} \|x\|_* \cdot \sqrt{\frac{2}{n}}$$

where $\mathcal{X}$ is the input space, and $\|\cdot\|_*$ is the dual norm of $\|\cdot\|$.

We now apply this bound to our setting, where our regularizer norm is $\|\cdot\|_A$ for positive definite A.

Lemma 7. *Pick any positive definite $p \times p$ matrix A and consider the Mahalanobis norm $\|\cdot\|_A$ on $\mathbb{R}^p$.*

1. *The function $\|\cdot\|_A^2$ is 2-strongly convex. In particular, for any $u, v \in \mathbb{R}^p$ and $0 \leq \alpha \leq 1$,*

$$\alpha\|u\|_A^2 + (1-\alpha)\|v\|_A^2 - \|\alpha u + (1-\alpha)v\|_A^2 \;=\; \alpha(1-\alpha)\|u-v\|_A^2.$$

2. *The dual norm of $\|\cdot\|_A$ is $\|\cdot\|_{A^{-1}}$.*

Proof. The first assertion follows directly by expanding the expression. For the second, we note that the dual norm of $\|\cdot\|_A$ is defined by

$$\|x\|_* = \sup_{\|y\|_A \leq 1} x \cdot y.$$

We will show that this is $\|x\|_{A^{-1}}$.

First, take

$$y = \frac{A^{-1}x}{\sqrt{x^T A^{-1} x}}.$$

Then

$$\|y\|_A^2 = y^T A y = \frac{x^T A^{-1} A A^{-1} x}{x^T A^{-1} x} = 1$$

so $\|y\|_A = 1$. Moreover, $x \cdot y = \sqrt{x^T A^{-1} x} = \|x\|_{A^{-1}}$.

Conversely, pick any y with $\|y\|_A \leq 1$. Then

$$x \cdot y \;=\; x^T A^{-1/2} A^{1/2} y \;=\; (A^{-1/2}x)^T (A^{1/2}y) \;\leq\; \|A^{-1/2}x\|_2 \|A^{1/2}y\|_2 \;=\; \|x\|_{A^{-1}}\|y\|_A \;\leq\; \|x\|_{A^{-1}}.$$

□

If w^* is the sparse target classifier, the function class of interest is $\mathcal{F} = \{w : \|w\|_A \leq \|w^*\|_A\}$ and by [Kakade et al., 2009] we have

$$R_n(\mathcal{F}) \;\leq\; \|w^*\|_A \cdot \max_{x\in\mathcal{X}} \|x\|_{A^{-1}} \sqrt{\frac{2}{n}}$$

Let $R = \{i \in [p] : w_i^* \neq 0\}$ denote the relevant features. We can split any x into its relevant and other components, $x = (x_R, x_o)$, and when we downweight the diagonal R-entries of A by a factor of c, we get

$$\|x\|_{A^{-1}}^2 = \|x_o\|_2^2 + c\|x_R\|_2^2$$

whereas

$$\|w^*\|_A^2 = \frac{1}{c}\|w\|_2^2$$

(assuming we have captured all the features on which w^* is non-zero). Thus

$$R_n(\mathcal{F}) \;\leq\; \|w^*\|_2 \cdot \max_{x \in \mathcal{X}} \sqrt{\left(\frac{1}{c}\|x_o\|_2^2 + \|x_R\|_2^2\right)} \sqrt{\frac{2}{n}}.$$

C Proof of Lemma 6

Proof. Consider the optimization problem for computing the support vector classifier using the mixed regularizer.

$$\begin{aligned} \underset{w}{\text{minimize}} \quad & \frac{1}{2}\|w\|_A^2 + C\sum_{i=1}^{N} \xi_i \\ \text{subject to} \quad & \xi_i \geq 0, y_i(x_i^T w) \geq 1 - \xi, \forall i. \end{aligned} \tag{1}$$

The primal Lagrange function L_P of (1) is

$$\begin{aligned} L_P = \frac{1}{2}\|w\|_A^2 + C\sum_{i=1}^{N} \xi_i - \sum_{i}^{N} \mu_i \xi_i \\ + \sum_{i}^{N} \alpha_i [y_i(x_i^T w) - (1 - \xi_i)], \end{aligned}$$

where the scalars α_i, μ_i are the lagrange multipliers. It easy to see that the Lagrange dual function L_D is

$$L_D = \sum_{i=1}^{N} \alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j x_i^T A^{-1} x_j.$$

□

D Experiments

D.1 Data sets

20 NewsGroups: The 20-Newsgroups collection is a set of approximately 20,000 newsgroup documents, partitioned evenly across the 20 different newsgroups. The documents are postings about politics, sports, technology, religion, science etc., and contain subject lines, signature files, and quoted portions of other articles. Some of the newsgroups are very closely related to each other (e.g., IBM computer system hardware *vs* Macintosh computer system hardware), while others are unrelated (e.g., misc for sale *vs* social religion and christian). A processed version of the data set was obtained. The original data set can be found in Jason Rennie's website. [1].

Reuters-21578: This is the most widely used collection for text categorization research. The documents appeared on the Reuters newswire in 1987 and were manually classified into several topics by personnel from Reuters Ltd. See Lewis et. al Lewis et al. [2004] for further details on the data set. Sub-collections **R10** (10 classes with the highest number of topics) and **R90** (at least one positive and one training example) are usually considered for text categorization tasks. As our goal here was to consider single-labeled data, all the documents with less than or with more than one label were eliminated, resulting in **R8** (8 classes) and **R52** (52 classes).

webkb: This data set contains web pages collected from computer science departments of various universities in January 1997 by the World Wide Knowledge Base project of the CMU text learning group [2].

cade: The documents in this collection correspond to web pages extracted from the CADE Web Directory, which points to Brazilian web pages classified by human experts in 12 classes, including services, education, sciences, sports, culture etc.

ohsumed: This data set includes medical abstracts from the MeSH (Medical Subject Headings) categories of the year 1991 [3] on 23 cardiovascular disease categories. We only considered documents with a single label.

For each data set we only considered tokens that occurred at least 3 times. Figure 3 below provides a summary of the data as they were used in the experiment.

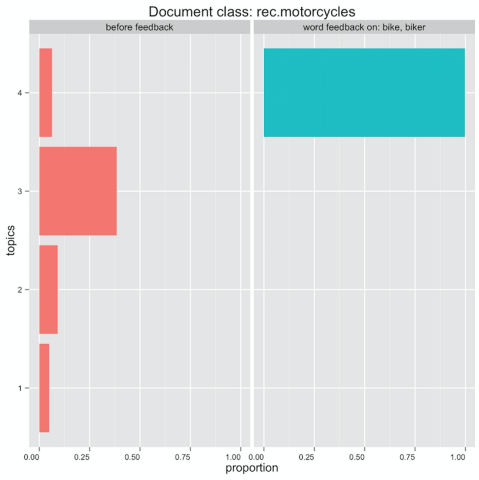
	# tokens	# training docs	# test docs	# topics	# classes
20 NewsGroups (20ng)	33,223	11,293	7,528	200	20
Reuters 8 (R8)	7,744	5,485	2,189	80	8
Reuters 52 (R52)	8.868	6,532	2,568	520	52
cade	68,983	27,322	13,661	120	12
webkb	7,644	2,803	1,396	40	4
ohsumed	13,627	3,357	4,043	230	23

Figure 3: Summary of the datasets and the number of topics used in the experiment

[1] http://qwone.com/ jason/20Newsgroups/
[2] http://www.cs.cmu.edu/afs/cs.cmu.edu/project/theo-4/text-learning/www/index.html
[3] ftp://medir.ohsu.edu/pub/ohsumed

D.2 Results

An example of a PDM from the **20ng** dataset is shown in figure 4. Figures 5 - 10 show our experimental results for each one of the data sets in more detail. Figures 11- 12, show the *amount* of feedback over time.



	Topic 1	Topic 2	Topic 3	Topic 4
1	gener	air	unit	bike
2	process	heat	engin	dod
3	thi	temperatur	cross	ride
4	sinc	water	bnr	motorcycl
5	effect	cold	adjust	bmw
6	anoth	pressur	link	rider
7	requir	hot	pre	helmet
8	real	fan	replac	sun
9	result	effect	nick	drink
10	case	ga	put	biker

Figure 4: Left : Topic representation of a document with the class **rec.motorcycles** before and after feature feedback, on the oracle features **bike** and **biker** Right: Descriptive words of the topics that are present in the document.

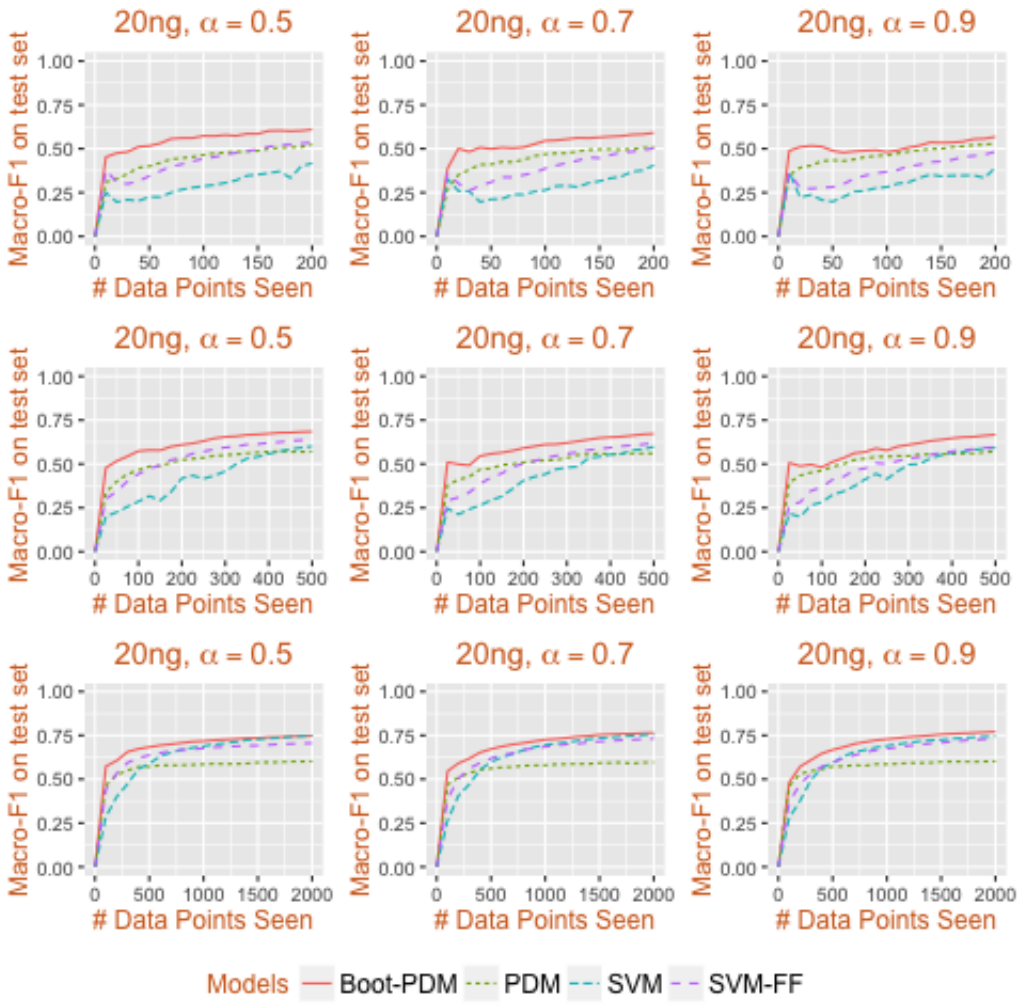


Figure 5: 20ng

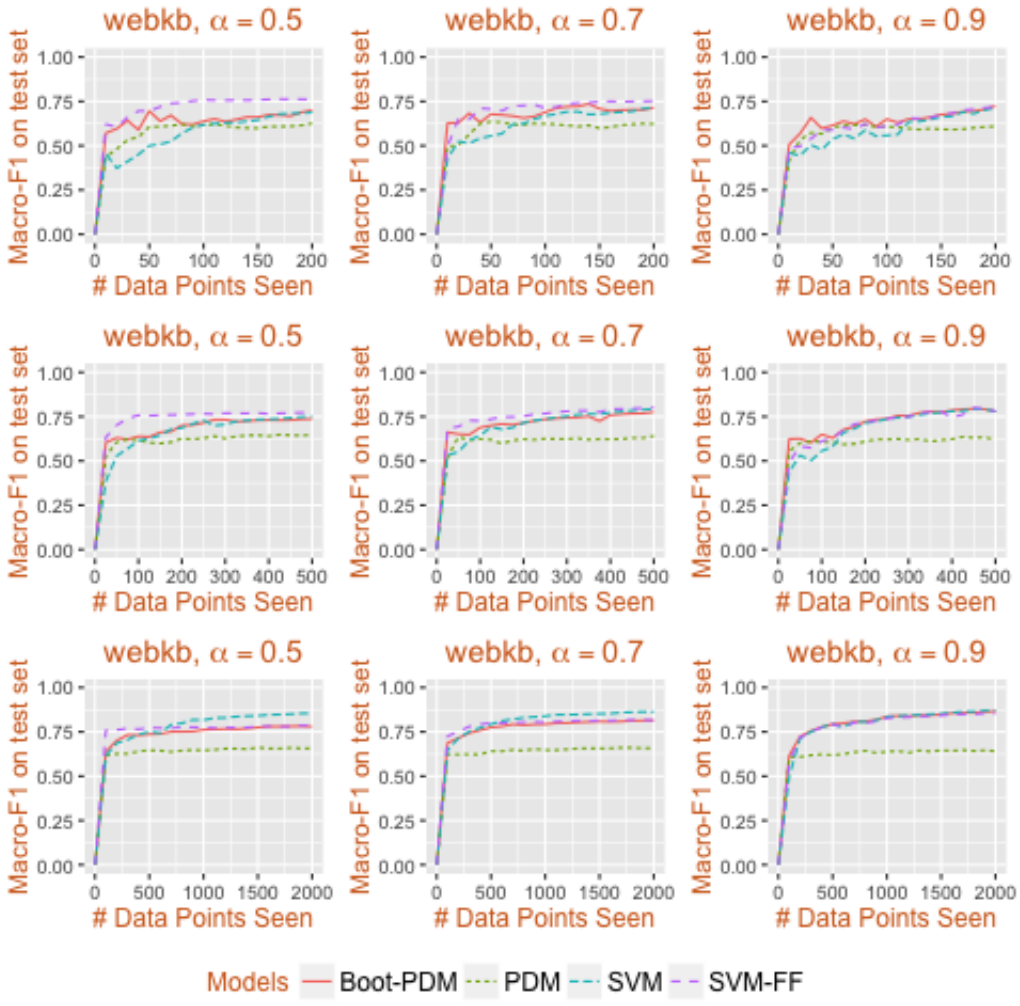


Figure 6: webkb

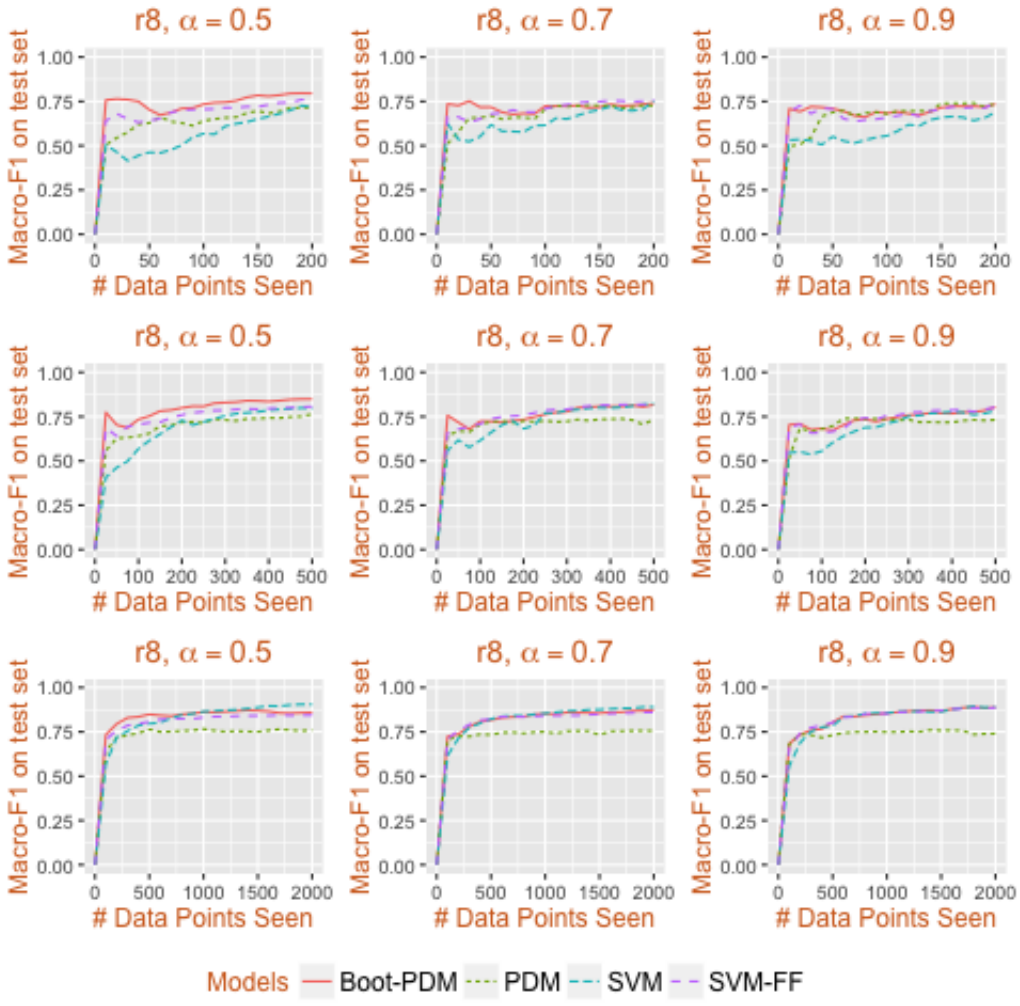

r8, α = 0.5
r8, α = 0.7
r8, α = 0.9
Macro-F1 on test set
Data Points Seen
Models
Boot-PDM
PDM
SVM
SVM-FF

Figure 7: R8

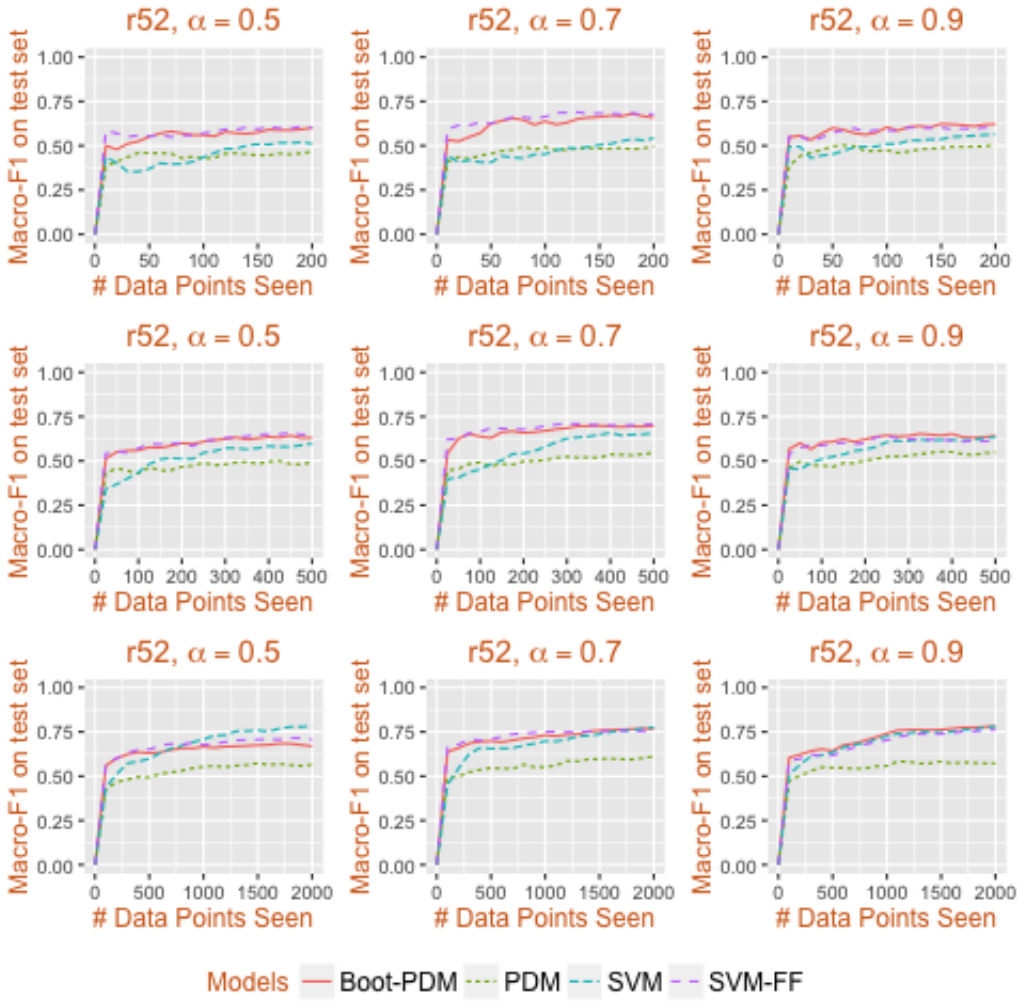

r52, α = 0.5
r52, α = 0.7
r52, α = 0.9
Macro-F1 on test set
Data Points Seen
Models
Boot-PDM
PDM
SVM
SVM-FF

Figure 8: R52

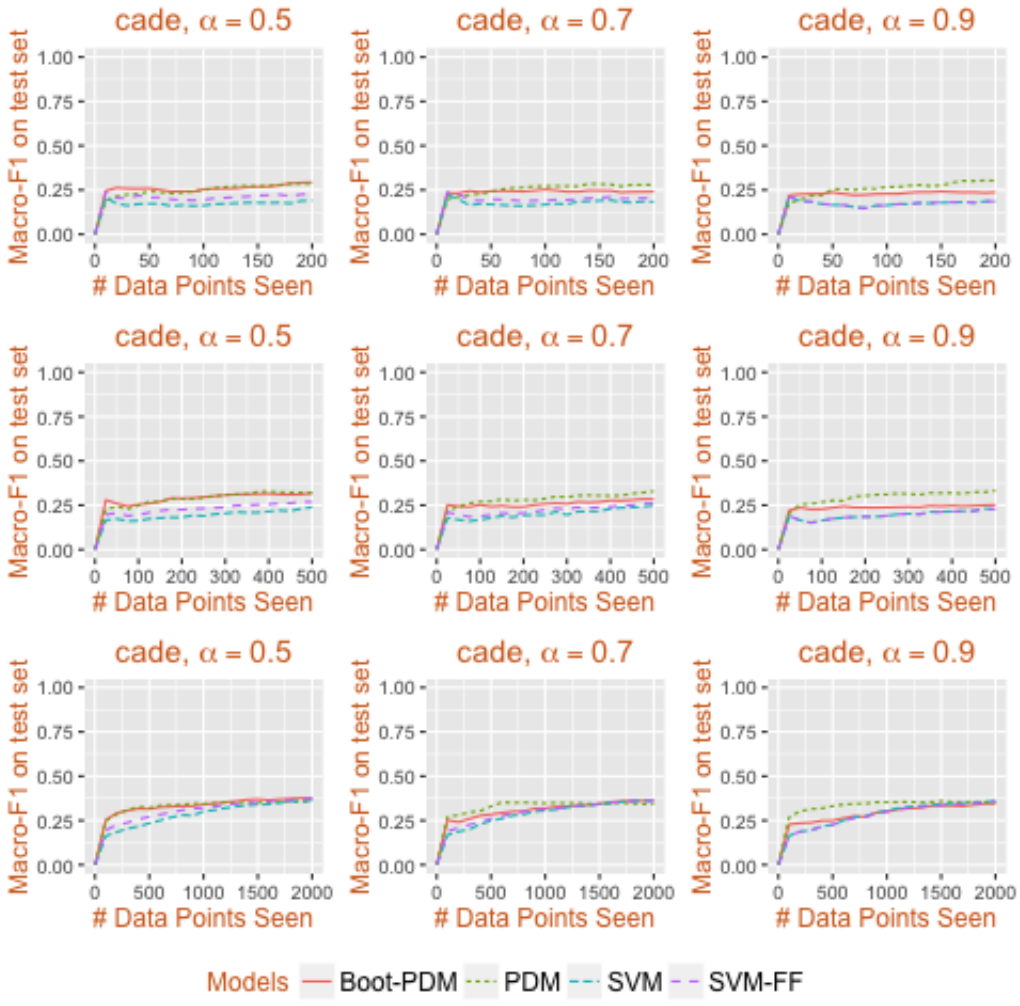


Figure 9: Cade

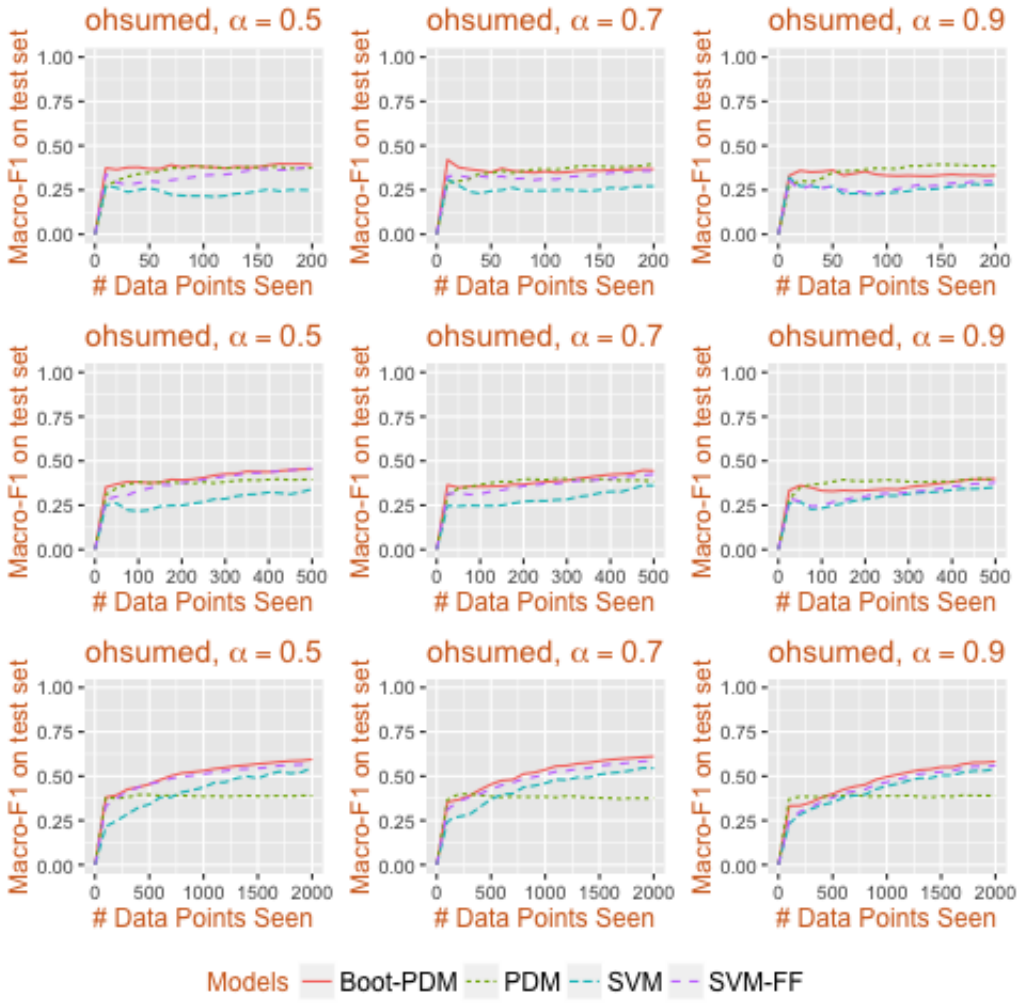


Figure 10: Ohsumed

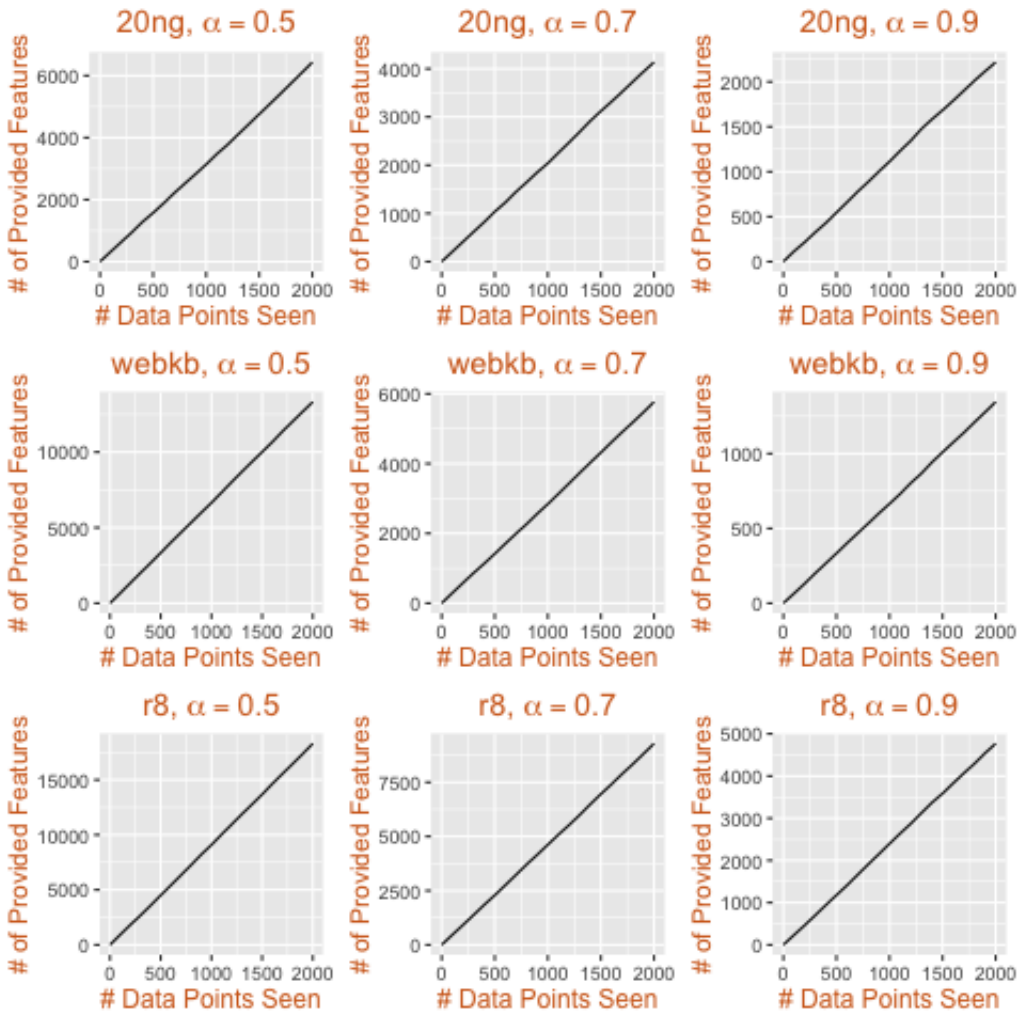


Figure 11: Amount of Feature Feedback

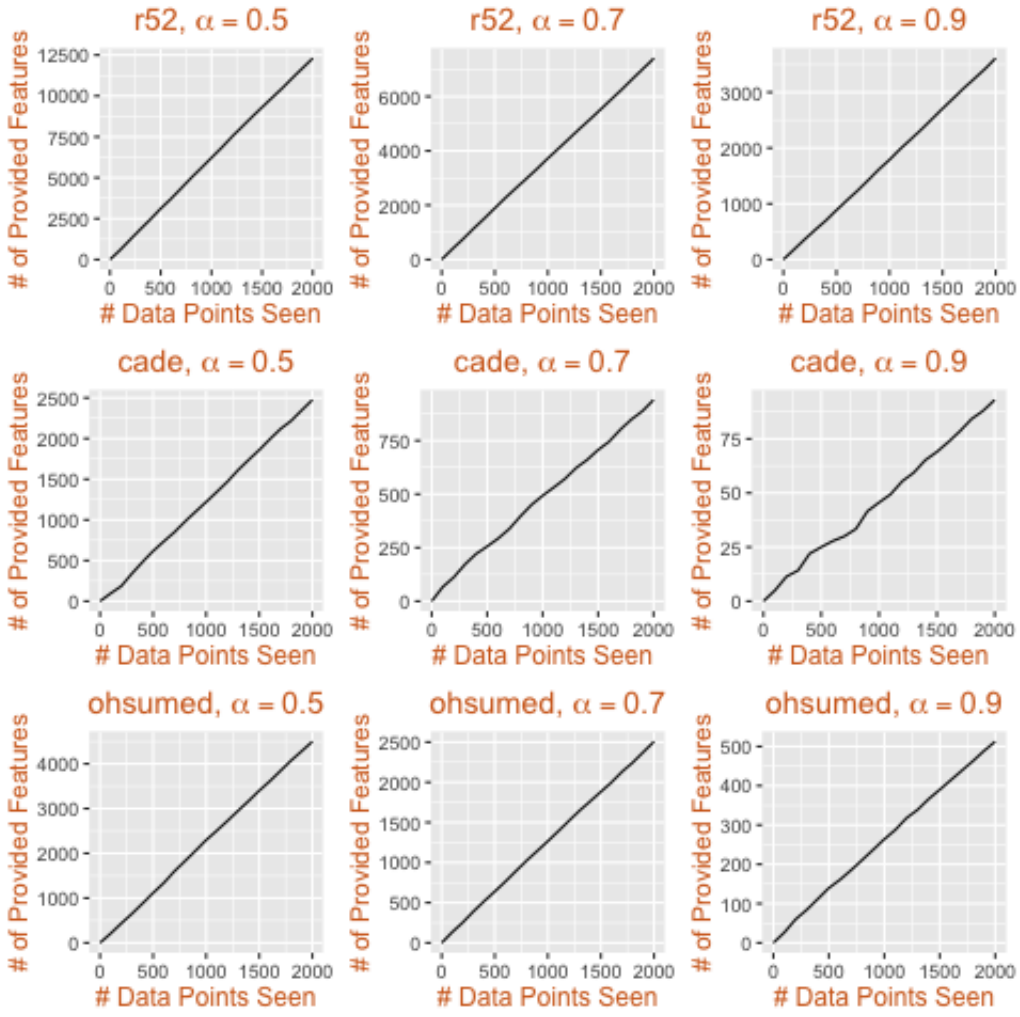


Figure 12: Amount of Feature Feedback

D.3 Human Experiment.

Figure 13 depicts the interface that was used to solicit labels and feature feedback from human annotators. Annotators were given the option to select a number of features from a list. They were also given the ability to insert a feature from the document that was not in the list.

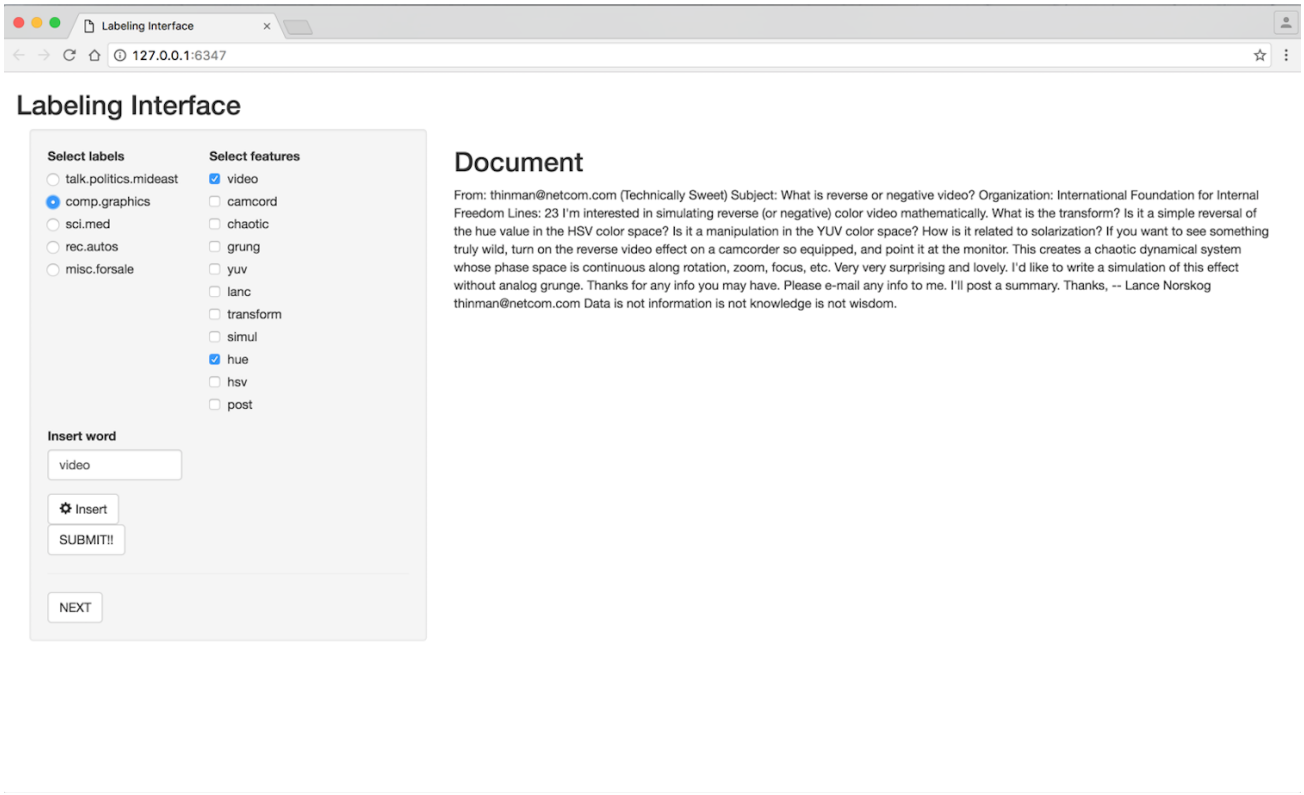


Figure 13: Interface used in Human Experiment