Active Learning Approaches for Labeling Text:
Review and Assessment of the Performance of Active Learning Approaches*

Blake Miller† Fridolin Linder‡ Walter R. Mebane, Jr.§

April 18, 2018

Abstract

In the case where concepts to measure in corpora are known in advance, supervised methods are likely to provide better qualitative results, model selection procedures, and model performance measures. In this paper, we illustrate that much of the expense of manual corpus labeling comes from common sampling practices such as random sampling that result in sparse coverage across classes, and duplicated effort of the expert who is labeling texts (it does not help your model's performance to label a document that is very similar to a document the expert has already labeled). In this paper we outline several active learning methods for iteratively modeling text and sampling articles based on model uncertainty with respect to unlabeled posts. We show that with particular care in sampling unlabeled data, researchers can train high performance text classification models using a fraction of the labeled documents one would need using random sampling. We illustrate this using several experiments on three corpora that vary in size and domain type (Tweets, Wikipedia talk sections, and Breitbart articles).

*Work supported by NSF award SES 1523355.

†Department of Political Science, University of Michigan, Haven Hall, Ann Arbor, MI 48109-1045 (E-mail: blakeapm@umich.edu).

‡Department of Political Science, Pennsylvania State University, Pond Laboratory, State College, PA 16801 (E-mail: fridolin.linder@psu.edu).

§Professor, Department of Political Science and Department of Statistics, University of Michigan, Haven Hall, Ann Arbor, MI 48109-1045 (E-mail: wmebane@umich.edu).

1 Introduction

In the past few years, automated text analysis methods have gained significant attention in political science.[1] Though automatic text analysis has grown in popularity, this growth has been lopsided, with very little attention paid to supervised learning approaches. There are many unsupervised applications in the political science literature: there are 9 published political science works listed on the structural topic model (STM) website and there are dozens of political science papers that use latent dirichlet allocation (Roberts, Stewart, Tingley, Airoldi, et al., 2013; Blei, Ng, & Jordan, 2003). Conversely, there are just a handful of supervised applications: Drutman & Hopkins (2013) classifies corporate emails as political or non-political; Ceron, Curini, Iacus, & Porro (2014) uses supervised methods to track public opinion on social media in Europe; and Workman (2015) and Collingwood & Wilkerson (2012) use supervised models to code policy agendas. The relative scarcity of supervised applications to automated text analysis is perhaps due to the difficulty and costliness of labeling texts, which is not necessary for unsupervised models. Costliness of labeling texts is often advanced as an argument for using unsupervised models in political science research (Quinn, Monroe, Colaresi, Crespin, & Radev, 2010).

In this paper, we demonstrate that much of the cost that comes from labeling texts is a result of randomly sampling documents for an expert to label when classes are imbalanced. For example, a political scientist may wish to identify texts from a newspaper corpus that reference a terrorist attack. Since terrorist attacks are rare events, we can expect that a random sample of newspaper articles will contain only a small number of articles about terrorist attacks and an expert labeler will spend much of his/her time labeling irrelevant documents. Active learning approaches instead draw samples of documents from a set of unlabeled documents using an uncertainty measure, usually derived from the decisions of classifier(s). Documents are coded in batches, with each batch representing the set of documents the classifier was most 'uncertain' of. In this paper, we demonstrate that active learning approaches to labeling texts reduce the costs of supervised learning in almost every scenario, with the exception of balanced classification problems, which social science researchers are unlikely to encounter (Ertekin, Huang, Bottou, & Giles, 2007; Sun, Wong, & Kamel, 2009). Because the costliness of text labeling has been a barrier to wider adoption of supervised methods, we hope that wider knowledge of active learning will result in expanded use of supervised models, which are flexible, easy to validate, and come with straightforward performance assessments. Active learning approaches to text analysis are being used to accomplish difficult classification and retrieval tasks in recent political science work (Mebane Jr, Klaver, & Miller, 2016; Linder, 2017; Miller, 2016). We hope that this paper can facilitate wider use of these methods.

1.1 Unsupervised vs. Supervised Learning

Automated text analysis methods can be categorized as supervised or unsupervised. Supervised models 'learn' from a subset of data that is annotated by experts, while unsupervised models require no annotation, instead learning from correlations and co-occurrences of text features. While unsupervised models are important research tools, supervised models are usually a better choice for measuring concepts the researcher has defined a priori. Conversely, unsupervised models are a better choice for discovering the latent

[1] Wilkerson & Casas (2017) provide a review of methods and applications for political science, and Grimmer & Stewart (2013) offer practical applications of several automated text analysis methods.

topics within large corpora in the absence of a priori knowledge about the structure of these corpora.

When deciding between supervised and unsupervised models, it is important to consider the promises and pitfalls of each approach. While unsupervised models are good for summarization and exploration of large corpora, they lack agreed-upon model selection procedures (Wallach, Murray, Salakhutdinov, & Mimno, 2009), are highly unstable with respect to text preprocessing and hyperparameter choices (Denny & Spirling, 2018), and require a great deal of human interpretation (Chang, Gerrish, Wang, Boyd-Graber, & Blei, 2009). Conversely, model selection, hyperparameter/preprocessing choice, and model evaluation are all straightforward in supervised models. Despite the many benefits of supervised approaches, the process of supervised learning is cumbersome, expensive, and labor-intensive. To achieve acceptable performance, supervised models require a large amount of human-annotated training data, especially in imbalanced classification problems, where concepts to measure within a corpus are rare. Active learning can help make this process less cumbersome and expensive, allowing researchers to reap the many benefits of supervised models of text.

1.2 Active vs. Passive Learning

In most automated classification procedures, an expert[2] labels the class membership of a fixed set of observations in the data. These observations are usually drawn from a random sample. Expert-labeled data are then used to "train" a learning algorithm to predict the labels of unlabeled observations in the dataset. While this approach to classification works quite well in many cases, it is often intractable when labeling is costly or when certain classes are very rare. The computer science literature has consistently demonstrated that in these cases, active learning can be quite useful (see e.g. Schohn & Cohn, 2000; Dasgupta, Kalai, & Monteleoni, 2005; Tong & Koller, 2001; Roy & McCallum, 2001; Ertekin et al., 2007). In an active learning approach to classification, a learning algorithm suggests which data points the expert should label. This suggestion usually is made according to a quantitative metric of the expected performance improvement that could be realized by each of the unlabeled observations in the data.

In this paper we introduce several active learning approaches to labeling texts and give guidance to practical applications. We aim to provide an introduction to the concept and an accessible starting point for political scientists who may benefit from this approach to labeling texts. We do not provide an exhaustive overview or theoretically rigorous exposition of the large number of varieties of active learning. For those who want to dive deeper into the theory and literature, see Settles (2012). In this paper, we suggest that political scientists could benefit from active learning approaches to supervised learning, especially when quantitatively analyzing texts.

Supervised machine learning, when applied to texts, can help to extend to the entire corpus, or sometimes to documents outside of the reference corpus, the annotations of a sample of documents. In a "passive learning" approach, an expert begins by labeling a sample of documents according to some predefined concept. This subset of documents is then used to train a learning algorithm to automatically predict the label of new documents outside of this "training sample." This approach breaks down when labeling documents is costly or the concept to measure is extremely rare. If a concept appears in only 1 percent of documents, to get a sample of 20 documents, an expert labeler would

[2]Sometimes referred to as an "oracle," "labeler," or "coder."

have to label, in expectation, 2000 documents. Even with this time-intensive labeling effort, a machine learning algorithm trained on only 20 observations is unlikely to perform well. If this expert were to use an "active learning" rather than "passive learning" approach, an analyst would iteratively answer a learning algorithm's queries for documents the model is uncertain about. After labeling a single document or batch of documents, a new learning algorithm will be trained and will execute a new query of unlabeled documents for the expert coder to label. In this paper, we demonstrate that by using active learning, analysts can train high performance text classification models with a fraction of the labeled documents one would need using random sampling. In our simulations, we find that active learning outperforms passive learning approaches in all cases where there is less than perfect balance between classes.

We run a series of simulations of active and passive learning, varying in each simulation the class balance, document length, sample type, and querying strategy. Our simulations use one of two querying strategies: 1) distance to margin and 2) query by committee, which will be explained in detail in Section 2. In simulations, we vary text domain size: small (tweets), medium (Wikipedia talk comments), and large (news articles from Breitbart), preprocessing choices, and class balance. Our simulations can guide researchers in selecting active learning procedures that are most effective for their specific text domain and research goals.

2 Active Learning

2.1 General Principles

Given a set of documents with known labels $\mathbf{y}$ and a set of features (these can be bag-of-words features but also document metadata like time stamps or author information) $\mathbf{X}$, the goal of text classification (or of machine learning in general) is to learn the function $y = f(\mathbf{X})$ that most accurately maps features to the labels beyond the specific set (training set) $\mathbf{y}, \mathbf{X}$ (see e.g. Friedman, Hastie, & Tibshirani, 2001, for an introduction). For simplicity of exposition we refer only to binary classification in this paper (i.e. $y_i \in \{0, 1\}, \forall i$) but all methods described here are applicable (with minor modifications) to multi-class classification ($y_i \in \{0, 1, ..., k\} \forall i$) and continuous outcomes.

It is assumed that the structure of the problem is well defined, that is, it is clear what the labels for the documents are, and how an expert labeler assigns these labels to the documents. In this case, supervised learning can be more useful than unsupervised strategies such as topic models. Consider the example where a researcher is interested to find all social media posts related to a specific protest movement. In this case, the structure of the problem is well defined, that is, the researcher knows a priori what the classes of interest are (posts relevant to the movement and posts not relevant to the movement), and given a post, she would be able to determine the class membership of said post. In contrast, an unsupervised method, such as a topic model, would not be guaranteed to produce a topic that is congruent with the protest movement class. The supervised method would, therefore, be the more natural choice in such a situation.

However—and this is likely a reason for the bigger popularity of topic models in political science—the big disadvantage of supervised learning is the cost of labeling data (Quinn et al., 2010). Huge text corpora are very easily accessible to researchers (e.g. social media data, websites, legislative documents), but categorizing them into relevant and non-relevant categories requires costly manual labeling. In the classical machine learning

approach, data to be labeled are drawn at random from the population of all documents that have to be classified (we refer to this strategy as passive learning throughout the paper). In many situations this is a good strategy; however, we see two situations in which the cost of labeling data 'passively' can be prohibitively large:

1. If the distribution of labels or classes in $\mathbf{y}$ is imbalanced, that is, if one label occurs much more often than the other, it can take an enormous amount of labeled data to get enough information on the minority class for the algorithm to learn to recover it reliably.

2. If there are many very similar (meaning close in the feature space $\mathbf{X}$) documents, drawing data for labeling at random can be very inefficient (a lot of data is needed to learn representations of all relevant areas of the feature space).

Active learning can reduce the cost of labeling data dramatically by reducing the number of labeled data points (or documents) that are required to get a specified performance of the classification model. The basic principle of active learning is that the learning algorithm is involved in the selection of data points to be labeled. In the classical approach (or passive learning), a random sample of data points is labeled. This labeled data (the training data) is then used to train a model that helps to classify (or predict the label/outcome of) a larger population of data points whose label is unknown. With active learning, in contrast, the training data is not selected at random but iteratively in interaction with the model. In each iteration, a model is trained on the labeled data that is available so far, and the model is 'asked' or 'queried' about which yet-to-be-labeled data point would help the model learn best.

There are many different implementations of the active learning principle, which differ in the way the model is queried and what measure is used to determine what data point would be optimal to be labeled from the current model. Before discussing some of these variations in depth in Section 2.3, we give intuition on the logic of active learning using the least complex variant, relying on logistic regression as the classification model and uncertainty sampling as the querying strategy (Lewis & Catlett, 1994).

We denote the population of documents with unknown labels as $\mathbf{X}^* = \{\mathbf{x}_i^* | i = 1, ..., N\}$ where $\mathbf{x}_i$ is a single document represented by a set of features (for example word counts or document metadata). Each document $\mathbf{x}_i$ has a corresponding true label $y_i \in \{0, 1\}$ that can be obtained by asking an expert labeler to label it. We denote the logistic regression model with a 0.5 probability threshold as $f(\mathbf{x}_i, \theta) = \mathbb{I}(1/(1 + e^{-\mathbf{x}\theta}) > 0.5)$, were $\mathbb{I}(.)$ is the indicator function that returns 1 if the condition is met and 0 otherwise. The most basic active learning algorithm is then:

1. Start with an initial set of documents $\mathbf{X}$ with known labels $\mathbf{y}$

2. Train the model $f^*(., \theta^*)$ using the available training data $\mathbf{y}, \mathbf{X}$

3. Produce a predicted probability for each unknown document:
   $\hat{\mathbf{y}}^* = f^*(\mathbf{X}^*, \theta^*)$

4. Use the query function to obtain a new document for labeling:
   $z = q(\hat{\mathbf{y}}^*) = \text{argmin}_i \, |\hat{y}_i^* - 0.5|$

5. Obtain a label y_z for $\mathbf{x}_z$ from the expert labeler

6. Add y_z and $\mathbf{x}_z$ to $\mathbf{y}$ and $\mathbf{X}$, remove $\mathbf{x}_z$ from $\mathbf{X}^*$

7. Repeat Steps 2 - 5 until a stopping criterion is reached

To summarize this algorithm: Starting with a population of documents, each of which is supposed to be classified, and an initial set of labeled data (this can be chosen at random or be produced from the domain knowledge of the researcher), a model is trained to separate the relevant from the non-relevant documents. Each document of the population is then assigned a predicted probability from the model and the document with a probability closest to 0.5—that is, the document the model is least certain about—is selected for annotation. In many cases it makes sense to not only select a single document for labeling, but the batch of least certain documents. The procedure is then repeated by re-training the model with the additional labeled data, querying more documents, labeling, etc. Usually, the procedure is stopped when either the labeling budget is exhausted or a satisfactory classifier performance is reached (the definition of satisfactory, of course, depends on the application).

This strategy intuitively reduces the number of labeled data points that are necessary to achieve a level of performance in the two situations described above. In the case of imbalanced classes the model has initially little information about rare classes, given that the initial $\mathbf{y}^{(s)}$ most likely contains few data points belonging to the rare class. The model will be less certain about this class and therefore produce more samples for labeling from that class (this is assuming that there is some information about the class membership in the provided features $\mathbf{X}$).

In the second case, where many documents in $\mathbf{X}^*$ are very similar, intuition on the potential benefits can be gained from step 3 in the algorithm above. Given three data points $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3$ with the distance in the feature space $d(\mathbf{x}_1, \mathbf{x}_2) \ll d(\mathbf{x}_1, \mathbf{x}_3) \approx d(\mathbf{x}_2, \mathbf{x}_3)$, if $\mathbf{x}_1$ is labeled, the model will produce a predicted probability closer to 0.5 for $\mathbf{x}_3$ as compared to $\mathbf{x}_2$ since it is very different from what the model has 'seen' in previous iterations. Because, $\mathbf{x}_3$ will provide more information than $\mathbf{x}_2$, the expert labeler will be presented with $\mathbf{x}_3$, thereby reducing inefficiency in the labeling.

2.2 Varieties of Active Learning

There is a large variety of ways to implement the active learning principle described in Section 2.1. Implementations can differ in several regards depending on the application: Availability of unlabeled samples, the type of sample that is queried, the model and outcome, and the querying algorithm. These choices produced a very large technical literature and make the choice for practical applications difficult. In this section we will briefly describe the choices a researcher has, without going into technical depth (see Settles, 2012, for an excellent in depth review of the varieties of active learning).

Availability of Samples: Depending on the data source, documents for annotation might either be available as a pool of documents (as described above) from which single documents can be selected for querying and classification, or as a stream of documents, where not all documents are available at any time (for example in stream based APIs such as the Twitter Streaming API). For simplicity, we only discuss the former case, leaving the latter as a special case.

Sample type: In Section 2.1 in Steps 4 and 5, a new labeled document is obtained by selecting the least certain document from the pool of documents. However, some authors have suggested to produce a synthetic sample, that is constructed in such a way that the

model learns best from it after it was labeled. This strategy can lead to multiple problems in practice. Most importantly, a synthetically generated document will be difficult to label for an expert labeler. Since unlabeled data samples are so readily available in almost all applications, we will not investigate this option in more detail.

Classification Model: In principle every classification model can be used in an active learning approach. Some querying methods have specific requirements for the model. For instance, the uncertainty sampling strategy used in step 4 in the illustration above requires the model to produce a measure of uncertainty (for example a predicted probability), not just a discrete class prediction. There are also active learning varieties for continuous outcomes. Uncertainty in this case is usually operationalized through the variance in the prediction on each unlabeled sample, so applications in this case are relatively straight forward.

Querying Strategy: How to measure the potential information gain for the model from an unlabeled instance is probably the component of active learning that received the larges amount of attention. Besides the uncertainty sampling querying strategy described in Section 2.1, there are a variety of methods to obtain unlabeled documents from the model. In this paper, we will discuss in detail (see Section 2.3) the two most commonly applied methods: query by committee and distance to margin sampling (uncertainty sampling with support vector machines). We note, however, that there are a variety of other methodologies and refer the reader to the literature for more in depth discussions of other options.

2.3 Querying Strategies: Distance to Margin and Query by Committee

Distance to Margin:

Uncertainty sampling as described in the last section is the most intuitive way of doing active learning, and in the case of the logistic regression model a measure of uncertainty is immediately available. Logistic regression, however, might not be the model of choice for text classification. SVM is a very commonly applied model for text classification. SVMs, like all linear models, work well for classification of high-dimensional and approximately linearly separable feature spaces.

In this paper, our approach is similar to the simple margin algorithm detailed in Tong & Koller (2001).[3] Essentially, the algorithm, similar to the logistic regression algorithm above, uses an uncertainty measure to query new points for labeling by an expert labeler. Because the objective of support vector machines is to find a hyperplane that optimally separates classes, observations that are closest to that hyperplane represent observations that the support vector machine is most uncertain of. For the simple margin classifier, at each iteration, a SVM model is trained and points closest to the hyperplane are returned to an expert to label.

Below is an overview of the distance to margin algorithm:

1. Let $\mathbf{X}$ represent the set of labeled observations with labels $\mathbf{y}$.

[3]Though Tong & Koller (2001) find that the simple margin algorithm is not optimal in maximally reducing the "version space," and offer several superior alternatives, it still works quite well and is useful as an example due to its simplicity.

2. Train a regularized support vector machine (SVM) with L2 penalty by minimizing the following loss function using gradient descent: $\min \frac{1}{2}\|\mathbf{w}\|^2 + \frac{C}{2}\sum_{i=1}^{p}\xi_i^2$ subject to $y_i(\mathbf{w}'\mathbf{x_i} + b) \geq 1 - \xi_i$ for $i = 1, ..., N$ where C is the regularization hyperparameter that us chosen using a cross-validated randomized hyperparameter search that also chooses text preprocessing methods[4]

3. The class-separating hyperplane learned above $\mathbf{h} \subset \mathbb{R}^p$ is defined by $\mathbf{h} = \{\tilde{\mathbf{x}} : \mathbf{x}'\mathbf{w} + b = 0\}$.

4. With all unlabeled observations $\mathbf{X}^*$, calculate the distance vector $\mathbf{d}$ of each point in $\mathbf{x_i^*}$ to $\mathbf{h}$

5. Return the m (batch size) points closest to the hyperplane (m can be chosen to suit the specific coding task)

6. The expert labels each returned document.

7. Repeat Steps 2 - 6 with new labeled data points until a stopping criterion is reached.

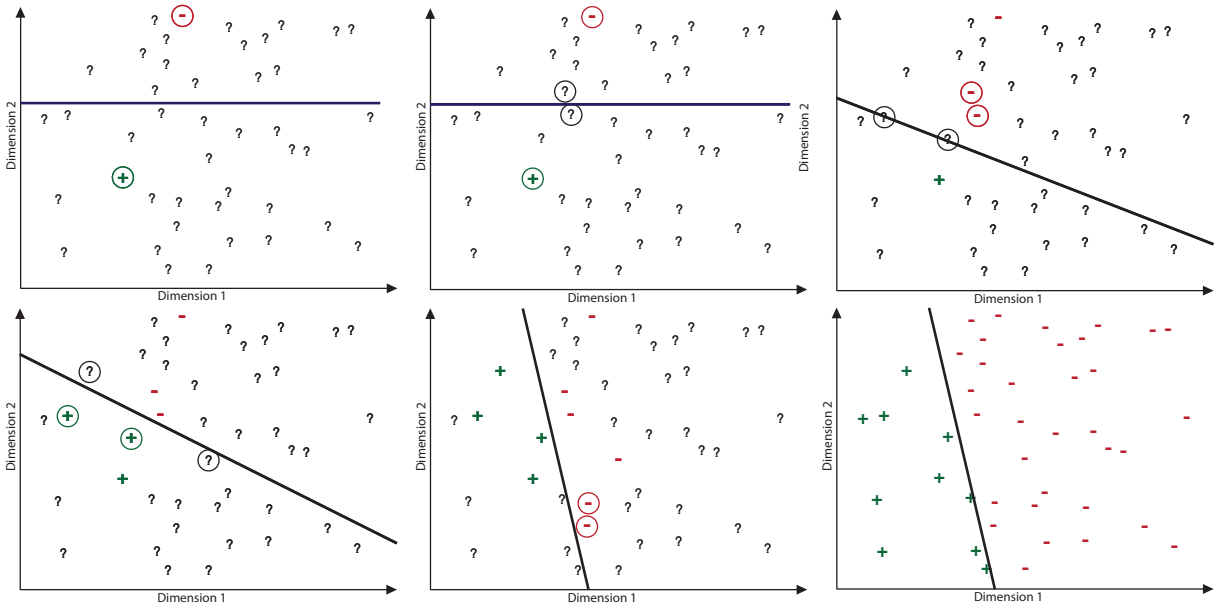

Figure 1: From left to right, top to bottom, the first 5 panels visualize the distance to margin algorithm. Circles represent queries of unlabeled data points. Once an expert has labeled the data, they are represented by pluses for the positive class and minuses for the negative class. The last panel shows the classes of all labeled points.

Query by Committee:

The query by committee approach to active learning functions similar to uncertainty sampling and class boundary sampling but relies on multiple models instead of a single model

[4]In our specific application, we use a randomized hyperparameter search (Bergstra & Bengio, 2012) with all combinations of text preprocessing decisions randomly selected with uniform probability. The hyperparameter C is drawn from a prior distribution that we define as an exponential distribution with $\lambda = 50$. For our randomized hyperparameter search, we train 20 random models and select the model with the highest performance according to its 5-fold cross-validated F1 score.

(Seung, Opper, & Sompolinsky, 1992; Freund, Seung, Shamir, & Tishby, 1997). The principle is to have a 'committee' of classification models all vote on unlabeled documents, and then to select documents to label where there exists most disagreement among the committee members. Disagreement in the committee is measured by entropy in the votes, choosing the document with maximal entropy for labeling. There are multiple ways of implementing such a committee. Multiple models can be produced by using the same type (e.g. SVM, logistic regression or Random Forest) with different hyperparamters. These methods, also referred to as Bayesian query by committee, draw hyperparameters from distributions in order to obtain a variety of 'hypotheses' about each document (Dagan & Engelson, 1995, e.g.). Other options are committees through boosting (Freund & Schapire, 1997) bagging (Abe & Mamitsuka, 1998), or partially trained convolutional neural networks (Ducoffe & Precioso, 2015). For the experiments below, we choose a combination of Bayesian query by committee and bagging of multiple models. We choose logistic regression, SVM and Naive Bayes (all classifiers commonly employed for text classification) and draw their hyperparameters randomly from distribution. See Section 4 for a more detailed description of the approach.

2.4 Biased Training Data, Generalization Error and Intercoder Reliability

The fundamental principle of active learning is to produce training data that is biased towards documents that the classification model(s) are uncertain about. That is, the dataset used to train the classification model is *not* representative of the total corpus. This has consequences for several aspects of the coding process. First, estimates of generalization error (i.e. how well the classification model will perform on the general corpus) obtained from the training data will be biased or inconsistent estimates of the true generalization error. Because the training data has been selected by querying uncertain documents, these documents are especially difficult for the model to classify. Estimating the performance of the classification model for the general corpus that potentially includes many 'easy' documents using this 'hard' training data, will most likely underestimate the true performance of the model (Baram, Yaniv, & Luz, 2004; Ali, Caruana, & Kapoor, 2014). However, as we demonstrate below, the model still performs better or at least as good as a model trained on randomly labeled data.

Second, expert labelers are most likely presented with cases that are more difficult to label. This is a problem that has been, to our knowledge, omitted from the computer science literature (Zhao, Sukthankar, & Sukthankar (2011) studies the implications of labeling errors on the performance of active learning algorithms but not the other way around). This could lead in turn to the complication that the error rate of the expert coders is likely to be higher as compared to randomly selected labeled data. An additional point to consider is, that estimates of inter-coder reliability, analogous to the generalization error, cannot be extrapolated to the general corpus. In planned simulations (not yet included in this draft), we will include varying levels of inter-coder reliability in our experiments presented below to investigate to what extent labeling error influences the performance of the classification process.

3 Data

We use three corpora that vary by text length and text style. By varying the text domain, we hope to demonstrate that active learning works across many of the domains relevant to political scientists. Accordingly we chose a corpus of tweets, a corpus of Wikipedia talk page entries, and a complete corpus of news articles from news website Breitbart. These corpora vary in document size from small, to medium, to large, respectively. The style of writing in each also varies, with social media, specialized/academic writing, and news articles respectively.

1. Twitter: This corpus is comprised of 24,420 tweets collected from a random sample of German Twitter users. The sample of tweets is a subset of a larger random sample of all Tweets authored by about 80,000 users. Each tweet was labeled as being about the refugee topic or not by German speaking CrowdFlower workers. Of the 24 thousand tweets about 700 are labeled as being about the topic. The dataset is used by Linder (2017) to study public reactions to refugee allocation in the German refugee crisis.

2. Wikipedia: This corpus of 159,571 Wikipedia talk page comments includes annotations of different kinds of toxic comments. The database was released as part of a machine learning competition, "Toxic Comment Classification Challenge" on the website Kaggle that is sponsored by ConversationAI, a team organized by Jigsaw and Google to build "tools to help improve online conversation." The goal of this competition is to classify the types of toxic comments using the provided expert annotations. For our simulations, we chose the label "toxic" because it has the most support of all represented classes. "Toxic" comments are aggressive comments, violent comments, personal attacks, etc. that do not contribute to a healthy and productive discussion on talk pages.

3. Breitbart: This corpus of 174,847 news articles represents the population of articles on the Breitbart news website. These articles each come with meta tags that are chosen by Breitbart authors and editors. For this dataset, we use the label "Muslim identity," which indicates whether a specific reference to Muslim identity was made in the article tags. This corpus is used to measure how moral and emotional frames in news media can increase support for violence against out-groups in Javed & Miller (2018).

4 Design

In our simulations, we sample 20 documents at a time and stop our simulation once we have processed 25% of documents in the corpus. Even though all documents are labeled, we treat all documents in the corpus as unlabeled and add each 20 sampled documents to a set of labeled observations. At each iteration of our simulation we train a regularized linear support vector machine with L2 norm using labeled documents. We use gradient descent to train the SVM.[5] We set values of the hyperparameters and decide

[5]For larger datasets, we provide an option in our simulation script for stochastic gradient descent.

on text preprocessing using a cross-validated randomized hyperparameter search.[6] For all conditions in our simulation, we use this SVM model to measure performance by calculating the F1 score on a held-out development set.

We test two sampling conditions: active and random. For random sampling, we sample 20 documents randomly from the set of unlabeled documents at each iteration and refit the SVM model. For active learning we sample documents using one of 2 approaches: 1) distance to margin and 2) query by committee.

For the distance to margin approach, we sample documents based on their proximity to the SVMs class-separating hyperplane. This sampling procedure selects documents that the classifier is most 'uncertain' of. This form of active learning is similar to the simple margin classifier detailed in Tong & Koller (2001). A more detailed description of the distance to margin procedure can be found in Section 2.3.

For query by committee sampling, we fit 5 random logistic regression, SVM, and naive Bayes classifiers. These classifiers are 'random' as they are trained using random hyperparameters drawn from distributions centered on sensible values. These 15 classifiers will serve as our committee. The 20 documents with the highest disagreement among our committee of classifiers are then selected for 'labeling' in the next iteration.

Finally, we simulate these sampling approaches by artificially inducing various degrees of class imbalance. We create new datasets with different levels of class imbalance by over- and undersampling the positive class from the original corpora. In our simulations we simulate active and passive learning with the following positive class proportions: 0.01, 0.05, 0.10, 0.30, and 0.50.

5 Results

[Note: All results are from the uncertainty sampling strategy. The query by committee results are in progress and not available yet.]

In this section, we present the results of the experiments described above. Figure 2 displays the results for all three datasets.[7] Each panel of the figures represents a level of class imbalance. 'Balance: 0.01' means that there are 1% relevant labels and 99% non-relevant labels. The lines are generalized additive model fits across replications of the same conditions. The grey solid line represents the active learning models, the yellow dashed line displays average performance for the models relying on randomly annotated data (or passive learning models). All Figures display the performance of the model (defined as the F1-Score) for different amounts of available training data (ranging from 20 documents to 6000 documents). Note that there are different amounts of training data

[6] Model selection (algorithm and tuning parameters) with active learning is non-trivial. Ali et al. (2014) discuss the problem, that the data points selected with active learning are highly biased, therefore making generalization error estimates obtained from this training data (e.g. via cross validation) invalid. Labeled data points are chosen to maximize the uncertainty of the classifier which leads to the performance of the classifier on this sample being much lower than on a true random sample from the population. In our experiments, we choose to still apply a standard model selection step because we did not intend to make generalization errors using estimates at each step. We also believed that this approach kept as many factors constant as possible between conditions.

[7] **[Note: The difference in amounts of data across datasets is due to the fact that at the time of this draft fewer replications of the experiment with the Breitbart data were completed.]**

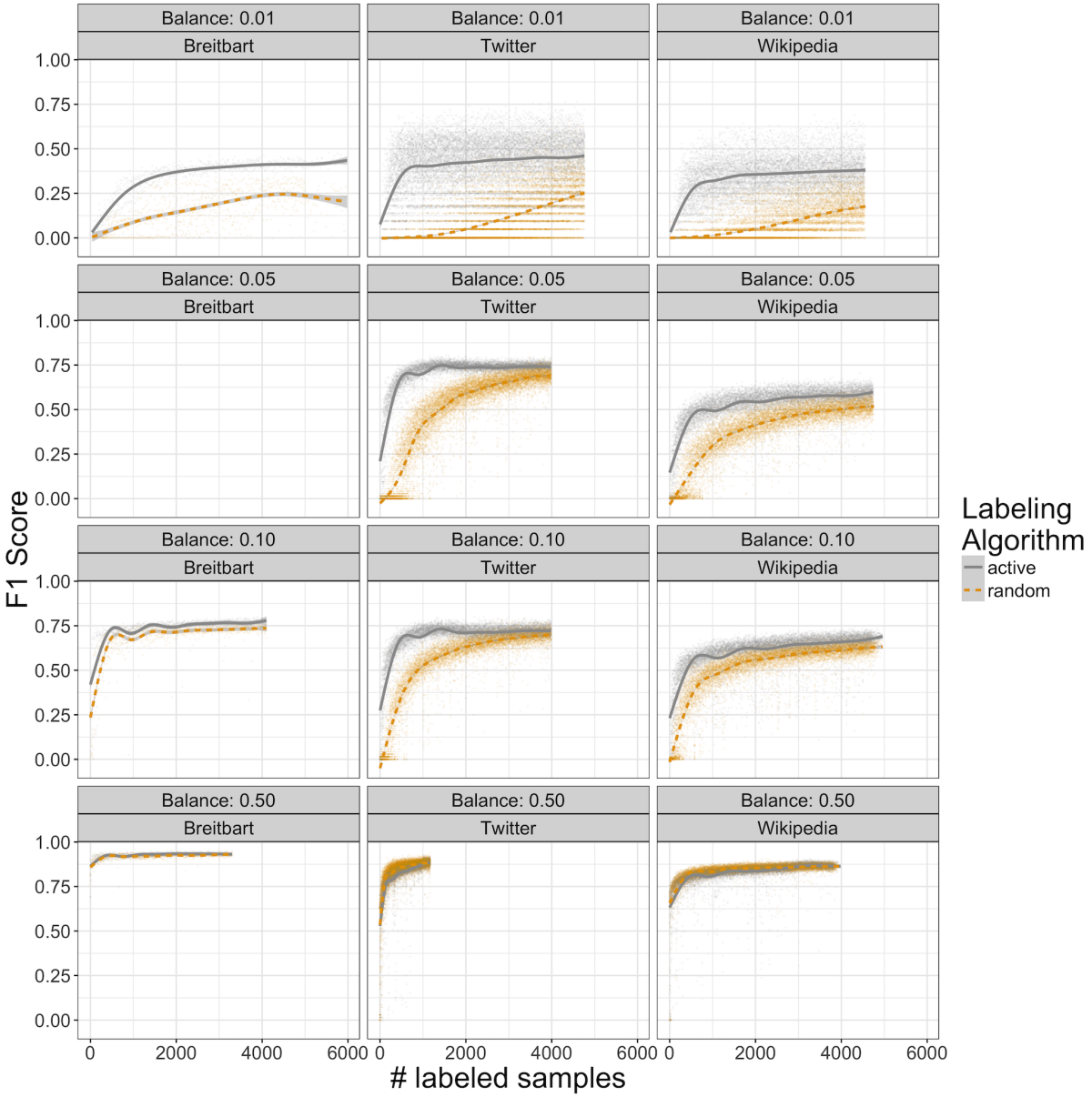

Figure 2: F1 score for experiments. The panel columns correspond to the datasets the rows to the different levels of class imbalance. Dots represent single replications of the experiment, smoothed lines are fits (and standard errors) of a generalized additive model.

in the different balance levels. This is due to the fact that, for example, for a balance of 0.5, the total data set size is constrained to two times the number of relevant documents in the corpus. The original Twitter data has about 3% relevant tweets, which means that creating a balanced dataset reduces the total size of the data considerably.

The gains in performance and reduction in cost resulting from the active learning approach are clearly visible in the imbalanced conditions. The model trained on the randomly labeled data does not achieve the same performance as the active learning model, even with almost one-fourth of the total corpus being labeled. The active learning algorithm, on the other hand, reaches its best performance very quickly—at about 500 labeled documents. The relationship of the performance differential with the balance of the data is striking. With 1% of the data being relevant the difference is dramatic, while for the balanced data, there is virtually no difference in performance.

The length of documents seems to matter as well. The difference between the active and random labeling strategies is less evident for the Breitbart corpus as compared to the other two. The Twitter data, which is comprised of the shortest documents, show the largest differences.

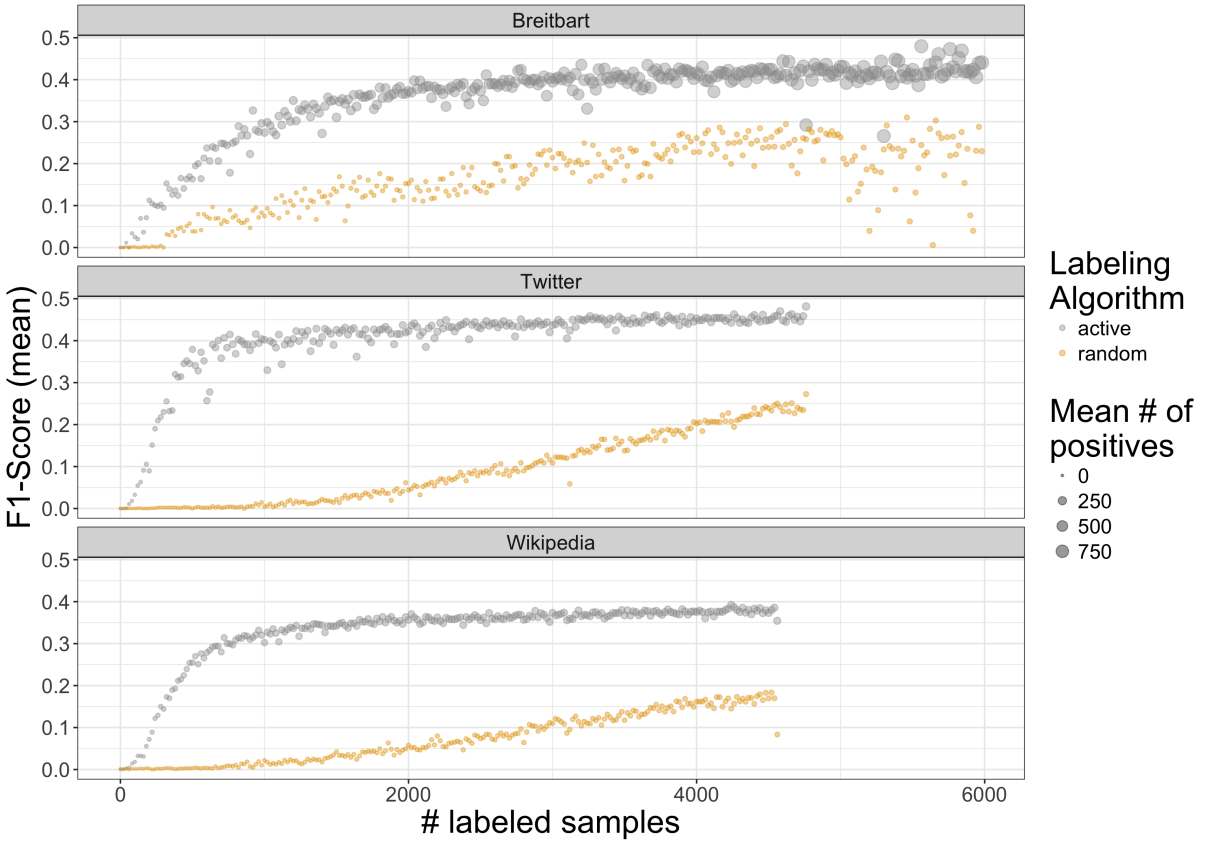

Figure 3: Performance by number of labeled examples for classifiers trained with active and passive learning (with class balance 0.01). Dots represent the average classifier performance across replications. Dot size is proportional to the average number of positively labeled data points in the training sample across replications.

To illuminate further why the active learner performs so much better in the imbalanced conditions, we zoom in on the 1% relevant data condition and additionally display the number of positive (or minority class) documents in the training data. These results are displayed in Figure 3.[8] In this figure, each dot represents the average f1-score across experimental replications. The size of the dots represents the average number of positively labeled documents in the training data. The following pattern can be observed: The active learning algorithm acquires positively labeled documents much faster than the random learner. While the proportion of positive sample increases linearly—proportionally increasing with the size of the training data—for the random learner, the active learner acquires almost all its positive samples within the first 1000 training data samples. This explains the performance difference: The active learner has much more information on

[8] **[Note: The larger variance in the Breitbart results is due to the fact that at the time of this draft fewer replications of the Breitbart experiment were completed.]**

the minority class much earlier, allowing it to perform better more quickly.[9]

6 Conclusion

To illustrate the benefits of active learning approaches to labeling texts for supervised learning, we ran several simulations with three datasets, varying the class balance, document length, sample type, and querying strategy. We found that active learning produces high-performance classification models with fewer labeled documents when compared to passive learning approaches. In all simulations, except for those with perfect class balance, active learning significantly reduced the time spent labeling documents. We hope our findings will serve to promote the use of supervised learning approaches to automated text analysis by making the process less costly and less time-intensive for researchers. In order to facilitate use of these methods for social scientists, we are developing software systems to simplify the process of labeling documents, tracking progress of expert coders, and checking inter-coder reliability.

[9]Figures 4 and 5 in the Appendix display precision and recall separately. From these figures it is evident, intuitively, that the availability of samples of the minority class increases recall on this class dramatically, which causes the much larger f1-scores observed in Figure 2.

References

Abe, H., & Mamitsuka, N. (1998). Query learning strategies using boosting and bagging. In *Machine learning: proceedings of the fifteenth international conference (icml98)* (Vol. 1).

Ali, A., Caruana, R., & Kapoor, A. (2014). Active learning with model selection. In *Aaai* (pp. 1673–1679).

Baram, Y., Yaniv, R. E., & Luz, K. (2004). Online choice of active learning algorithms. *Journal of Machine Learning Research*, *5*(Mar), 255–291.

Bergstra, J., & Bengio, Y. (2012). Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, *13*(Feb), 281–305.

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of machine Learning research*, *3*(Jan), 993–1022.

Ceron, A., Curini, L., Iacus, S. M., & Porro, G. (2014). Every tweet counts? how sentiment analysis of social media can improve our knowledge of citizens political preferences with an application to italy and france. *New Media & Society*, *16*(2), 340–358.

Chang, J., Gerrish, S., Wang, C., Boyd-Graber, J. L., & Blei, D. M. (2009). Reading tea leaves: How humans interpret topic models. In *Advances in neural information processing systems* (pp. 288–296).

Collingwood, L., & Wilkerson, J. (2012). Tradeoffs in accuracy and efficiency in supervised learning methods. *Journal of Information Technology & Politics*, *9*(3), 298–318.

Dagan, I., & Engelson, S. P. (1995). Committee-based sampling for training probabilistic classifiers. In *Machine learning proceedings 1995* (pp. 150–157). Elsevier.

Dasgupta, S., Kalai, A. T., & Monteleoni, C. (2005). Analysis of perceptron-based active learning. In *International conference on computational learning theory* (pp. 249–263).

Denny, M. J., & Spirling, A. (2018). Text preprocessing for unsupervised learning: why it matters, when it misleads, and what to do about it. *Political Analysis*, 1–22.

Drutman, L., & Hopkins, D. J. (2013). The inside view: Using the enron e-mail archive to understand corporate political attention. *Legislative Studies Quarterly*, *38*(1), 5–30.

Ducoffe, M., & Precioso, F. (2015). Qbdc: query by dropout committee for training deep supervised architecture. *arXiv preprint arXiv:1511.06412*.

Ertekin, S., Huang, J., Bottou, L., & Giles, L. (2007). Learning on the border: active learning in imbalanced data classification. In *Proceedings of the sixteenth acm conference on conference on information and knowledge management* (pp. 127–136).

Freund, Y., & Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences*, *55*(1), 119–139.

Freund, Y., Seung, H. S., Shamir, E., & Tishby, N. (1997). Selective sampling using the query by committee algorithm. *Machine learning*, *28*(2-3), 133–168.

Friedman, J., Hastie, T., & Tibshirani, R. (2001). *The elements of statistical learning* (Vol. 1). Springer series in statistics New York.

Grimmer, J., & Stewart, B. M. (2013). Text as data: The promise and pitfalls of automatic content analysis methods for political texts. *Political analysis*, *21*(3), 267–297.

Javed, J., & Miller, B. (2018). Mobilizing hate: Moral-emotional frames, outrage, and violent expression in online media. *unpublished*.

Lewis, D. D., & Catlett, J. (1994). Heterogeneous uncertainty sampling for supervised learning. In *Machine learning proceedings 1994* (pp. 148–156). Elsevier.

Linder, F. (2017). Improved data collection from online sources using query expansion and active learning.

Mebane Jr, W. R., Klaver, J., & Miller, B. (2016). Frauds, strategies and complaints in germany. In *Annual meeting of the midwest political science association, chicago.*

Mebane Jr, W. R., Pineda, A., Woods, L., Klaver, J., Wu, P., & Miller, B. (2017). Using twitter to observe election incidents in the united states. In *Annual meeting of the midwest political science association, chicago.*

Miller, B. (2016). Automated detection of chinese government astroturfers using network and social metadata.

Quinn, K. M., Monroe, B. L., Colaresi, M., Crespin, M. H., & Radev, D. R. (2010). How to analyze political attention with minimal assumptions and costs. *American Journal of Political Science*, *54*(1), 209–228.

Roberts, M. E., Stewart, B. M., Tingley, D., Airoldi, E. M., et al. (2013). The structural topic model and applied social science. In *Advances in neural information processing systems workshop on topic models: Computation, application, and evaluation.*

Roy, N., & McCallum, A. (2001). Toward optimal active learning through monte carlo estimation of error reduction. *ICML, Williamstown*, 441–448.

Schohn, G., & Cohn, D. (2000). Less is more: Active learning with support vector machines. In *Icml* (pp. 839–846).

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

Seung, H. S., Opper, M., & Sompolinsky, H. (1992). Query by committee. In *Proceedings of the fifth annual workshop on computational learning theory* (pp. 287–294).

Sun, Y., Wong, A. K., & Kamel, M. S. (2009). Classification of imbalanced data: A review. *International Journal of Pattern Recognition and Artificial Intelligence*, *23*(04), 687–719.

Tong, S., & Koller, D. (2001). Support vector machine active learning with applications to text classification. *Journal of machine learning research*, *2*(Nov), 45–66.

Wallach, H. M., Murray, I., Salakhutdinov, R., & Mimno, D. (2009). Evaluation methods for topic models. In *Proceedings of the 26th annual international conference on machine learning* (pp. 1105–1112).

Wilkerson, J., & Casas, A. (2017). Large-scale computerized text analysis in political science: Opportunities and challenges. *Annual Review of Political Science*, *20*, 529–544.

Workman, S. (2015). *The dynamics of bureaucracy in the us government: How congress and federal agencies process information and solve problems.* Cambridge University Press.

Zhao, L., Sukthankar, G., & Sukthankar, R. (2011). Incremental relabeling for active learning with noisy crowdsourced annotations. In *Privacy, security, risk and trust (passat) and 2011 ieee third inernational conference on social computing (socialcom), 2011 ieee third international conference on* (pp. 728–733).

7 Appendix

7.1 Precision Results

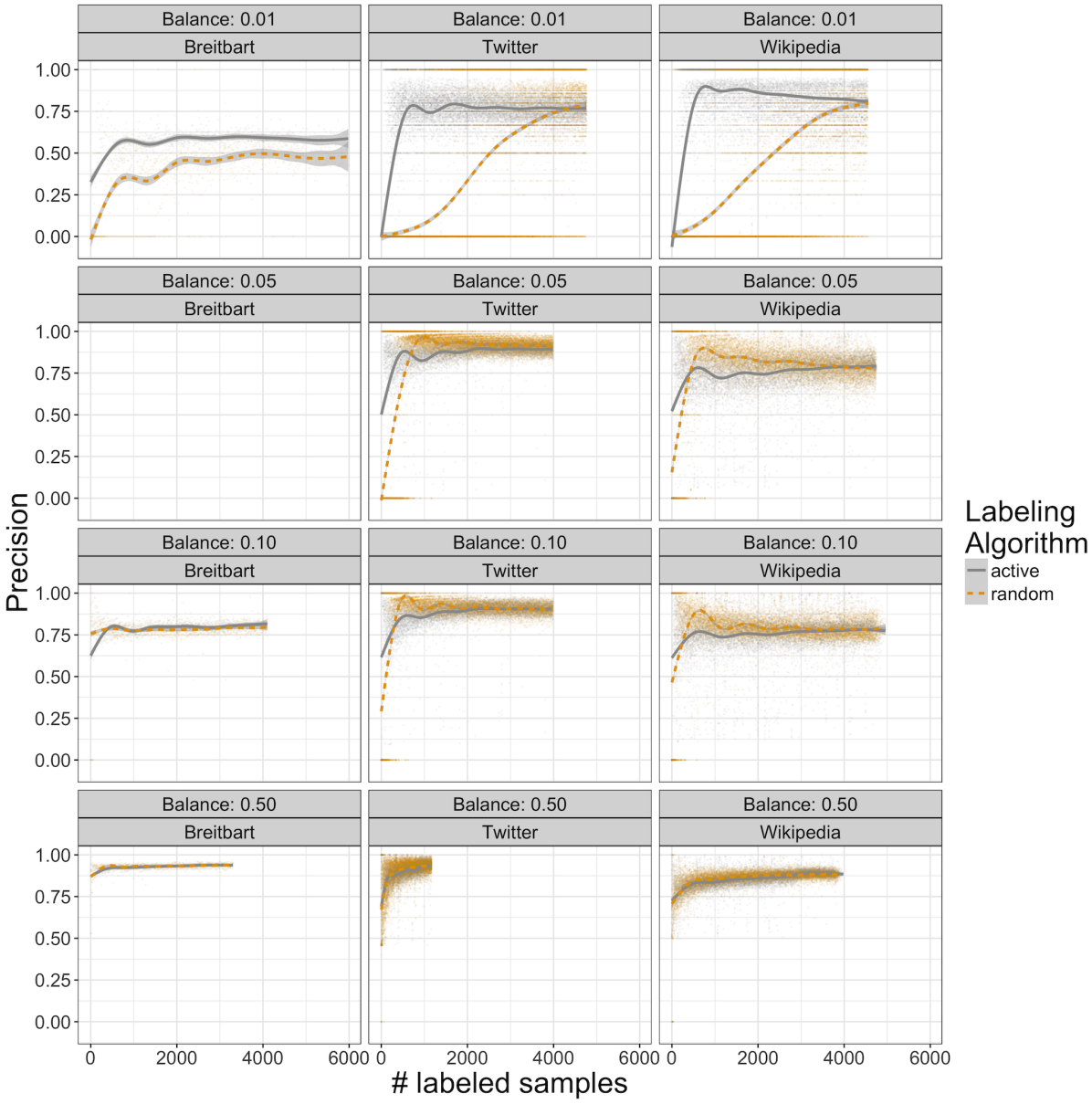

Figure 4: Precision score for experiments. The panel columns correspond to the datasets the rows to the different levels of class imbalance. Dots represent single replications of the experiment, smoothed lines are fits (and standard errors) of a generalized additive model.

7.2 Recall Results

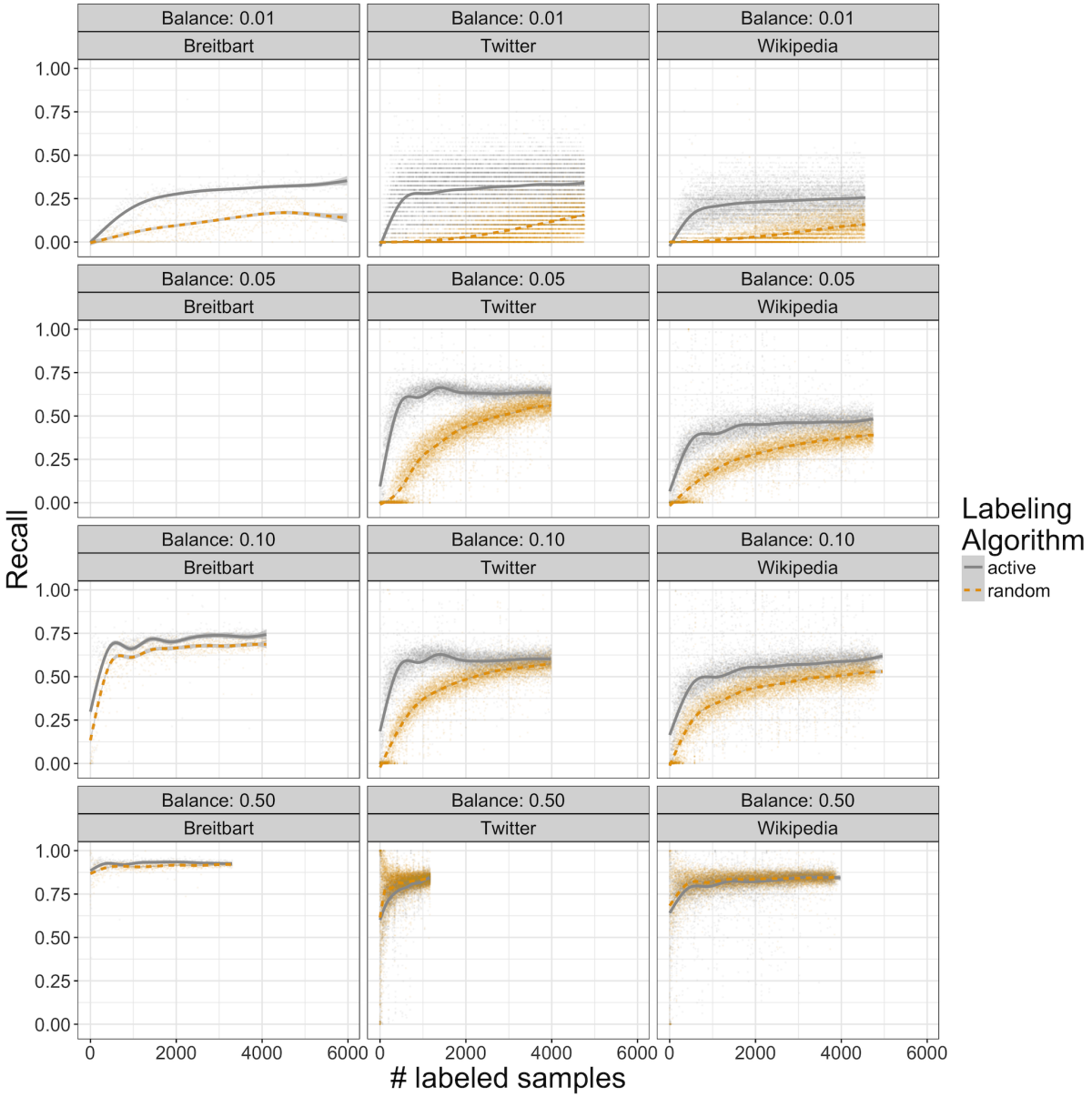

Figure 5: Recall score for experiments. The panel columns correspond to the datasets the rows to the different levels of class imbalance. Dots represent single replications of the experiment, smoothed lines are fits (and standard errors) of a generalized additive model.