Contextual Multi-View Active Learning for Short Text Classification in User-Generated Data

Payam Karisani
Emory University
pkarisa@emory.edu

Negin Karisani
Purdue University
nkarisan@purdue.edu

Li Xiong
Emory University
lxiong@emory.edu

Abstract

Mining user-generated content–e.g., for the early detection of outbreaks or for extracting personal observations–often suffers from the lack of enough training data, short document length, and informal language model. We propose a novel multi-view active learning model, called Context-aware Co-testing with Bagging (COCOBA), to address these issues in the classification tasks tailored for a query word–e.g., detecting illness reports given the disease name. COCOBA employs the context of user postings to construct two views. Then it uses the distribution of the representations in each view to detect the regions that are assigned to the opposite classes. This effectively leads to detecting the contexts that the two base learners disagree on. Our model also employs a query-by-committee model to address the usually noisy language of user postings. The experiments testify that our model is applicable to multiple important representative Twitter tasks and also significantly outperforms the existing baselines. Our code and dataset are available at https://github.com/p-karisani/cocoba.

1 Introduction

Over the last decade, social media data became one of the valuable resources for extracting information about real-world phenomena and activities. Early detecting of outbreaks (Karisani and Karisani, 2020), monitoring natural disasters (Imran et al., 2015), collecting daily individual observations (Mei et al., 2014), and monitoring customer satisfaction (Agnihotri et al., 2016) are a few applications. Being cost efficient and having short implementation cycles are the intriguing attributes of mining this data. However, there are significant technical challenges in automatically distilling such knowledge. User postings in social media are typically short, their language is informal, and their content can be highly ambiguous. Additionally, in many scenarios there is not enough training data available (Karisani and Karisani, 2021).

To address these challenges researchers seek to develop models that require smaller training sets and generalize faster. For instance, expansion methods were used to address short document lengths (Karisani et al., 2015), neural word embeddings were used to address feature sparsity (Karisani and Agichtein, 2018), Transfer Learning was used to address the lack of enough training data (Dirkson and Verberne, 2019), and Active Learning was used to address class imbalance distributions (Burkhardt et al., 2020). We focus on Active Learning in this article. The distinctive characteristics of active learning models make them especially appealing to the researchers in this domain. Being robust towards the initial training set and addressing noisy labels (Ghani et al., 2003), overcoming class imbalance challenge (Choi et al., 2020), and compensating for the lack of training data (Cui et al., 2019) are the well-understood qualities of Active Learning.

In this study, we tackle the classification tasks tailored for query words. The applications of such tasks are abundant. In Online Public Health Monitoring where given the variants of a disease name we want to extract the positive report cases (Paul and Dredze, 2017). In Customer Satisfaction Monitoring where given a product or brand name we want to extract the true mentions of the product and visualize the outcome (Agnihotri et al., 2016). In Observation Extraction where given a real-world phenomenon we want to extract the relevant reported observations (Cui et al., 2019). Or in Entity Filtering where given an entity name we want to filter out non-relevant user postings for the downstream tasks–e.g., for Online Reputation Management (Spina et al., 2015). In this article we exploit this shared quality and propose a novel unified active learning model for a range of tasks.

Our model, which we call COCOBA (Context-

aware Co-testing with Bagging), is based on the idea that the content of user postings can be used in a context sensitive multi-view active learning model to resolve the disagreement over similar use cases. To achieve this, we use the properties of the problem and derive two contextual representations from user postings. Then we modify a multi-view active learning model to effectively use these representations. And finally, we use a query-by-committee model to increase robustness to the noise in user postings. We show that COCOBA is applicable to at least three important representative problems[1]. Namely we focus on: Personal Health Mention detection (PHM) (Karisani and Agichtein, 2018) where given an illness name the goal is to detect the positive reports of the illness; Observation Extraction (OE) (Zahra et al., 2020) where given a real-world event the goal is to extract the relevant reported observations; and Product Consumption Pattern identification (PCP) (Huang et al., 2017) where given a product the goal is to detect the number of usages of the product to calculate its penetration rate. Our experiments testify that our novel unified model consistently outperforms existing models.

The contributions of our study are as follows: **1)** We propose a novel unified multi-view active learning model to address the tasks tailored for a query in user-generated data. **2)** We carry out an extensive set of experiments and show that our model is applicable to at least three representative tasks. **3)** We show that our model consistently outperforms existing active learning models. **4)** We constructed a relatively large dataset of manually annotated tweets for PHM task that is publicly available. Our dataset consists of 18,000 tweets across three topics[2]: Parkinson's, cancer, and diabetes.

We believe our novel model, our detailed experiments, and our new dataset significantly push the state of the art, and also help practitioners to develop better systems with smaller training sets. In the next section, we contrast COCOBA with existing models.

2 Background and Related Work

Background. In a typical active learning classification scenario, there is a small set of labeled data and a large set of unlabeled data available[3]. A predictive model is trained on the set of labeled data, and based on a criterion–either labeling cost or model performance–one data point from the set of unlabeled data is *queried* for annotation[4]. The annotated data point is added to the set of labeled data, and the procedure is iterated. The initial state in which the model has access to a small set of labeled data is called the *cold start* state. The learning algorithm that the model employs to explore the hypothesis space is called the ***base learner***; and the algorithm that the model uses to select the next unlabeled data point is called the *query strategy*. Majority of the active learning models rely on *informativeness*, *representativeness*, and *diversity* metrics to select their candidate data points (Chang et al., 2019). Despite the significant advances in Active Learning over the last decades, the uncertainty-based sampling model (Lewis and Gale, 1994) remains one of the most widely used and studied models (Attenberg and Provost, 2011; Jedoui et al., 2019). There are multiple methods to identify uncertainty in the base learner: the amount of entropy in the model prediction (Settles, 2009), the magnitude of gradients in back propagation (Zhang et al., 2017), or the variance in successive predictions of the model (Gal et al., 2017) are a few examples.

Active Learning for user-generated content. Given the stability and usually satisfactory performance of the uncertainty-based sampling model, the majority of the successful applications of Active Learning in user-generated data rely on this model. (Pohl et al., 2018) proposes a model for Crisis Report monitoring, (Tran et al., 2017) integrates Active Learning with Semi-supervised Learning for entity recognition, and (Spina et al., 2015) proposes to combine the informativeness and representativeness metrics for entity recognition. (Li et al., 2017) experiments with Active Learning for detecting symptoms in Chinese tweets, (Stanovsky et al., 2017) reports the application of Active Learning in Adverse Drug Reaction monitoring (ADR) task, and (Burkhardt et al., 2020) combines Active Learning with crowd sourcing for the ADR task. The authors in (Jiang et al., 2020) propose a query diversity criterion for spam filtering on Twitter, and

[1]Please see the cited articles for the discussion on the challenges of the selected tasks.

[2]Based on published reports (Yin et al., 2019) our dataset is the largest manually annotated dataset on this topic.

[3]The survey by Settles (Settles, 2009) and the article by Lowell et al., (Lowell et al., 2019) provide a complete overview of Active Learning.

[4]Our criterion in this article is the model performance, and we assume that the annotation cost is uniform.

the authors in (Zhao et al., 2020) combine Active Learning with crowd-sourcing to develop a pipeline for detecting job-related posts in social media. All of these studies use the uncertainty-based sampling model.

In this study, we focus on a multi-view contention reduction model (Abe and Mamitsuka, 1998) called co-testing (Muslea et al., 2006). The main idea of co-testing algorithm is to construct two views from input data and train a base learner on each view. Then query a data point from the set of unlabeled points that are assigned to the opposite classes by two base learners–these points are called ***contention*** points. To be able to use multi-view models, we derive two contextual representations from user postings. Then we modify the co-testing query strategy to utilize this contextual information and increase the gain in user annotations. We aim at a category of social media tasks tailored for a query word–or a closely related set of query words. Such tasks have many applications, ranging from Entity Filtering and Disease Mining to Crisis Management and Customer Satisfaction Monitoring. We show that our model, which we call COCOBA, is applicable to at least three representative tasks from different domains. Namely we focus on: Personal Health Mention detection (PHM), Observation Extraction (OE), and Product Consumption Pattern identification (PCP).

In summary, to our knowledge, our study is the first that proposes a unified active learning model for a range of social media tasks. It is also the first study that proposes to use a multi-view model to address these tasks. It is one of the very few works that step beyond applying the traditional uncertainty-based model[5], and to our knowledge, it is the only work that extends an active learning model to effectively exploit the properties of the user-generated data.

3 COCOBA: Model Description

We begin this section by discussing the approach for extracting two contextual representations from user postings. Given two views, we can employ co-testing algorithm, however, the default co-testing algorithm is context independent. Therefore, we will modify the default co-testing query strategy to use the contextual information. Finally, we try to tackle the typically noisy language of user postings via a variance reduction technique.

3.1 Extracting Two Contextual Representations from User Postings

Our approach to construct two views from the user postings is inspired by the research on Word Sense Disambiguation (WSD) and their mainstream solutions, i.e., the contextual word embeddings. The neural contextual word embeddings are proven to encode the information required to effectively characterize the context in which the words occur (Scarlini et al., 2020). To extract two contextual representations from the user postings, we extract one representation on the document level to capture the overall information of the user postings, and extract another representation on the word level to capture the context that the query words are used in. Because by definition the user postings always contain at least one of the query words then this task is always feasible. This approach is a derivation of the algorithm that we proposed in (Karisani et al., 2020).

We demonstrate this by outlining the task of extracting the true reports of diabetes on Twitter. Given the query words "diabetes" and "diabetic", we may observe the hypothetical tweet: *"Right now the only complication I've got with my **diabetes** is neuropathy, which isn't fun"*. Given this tweet, we can extract a feature vector on the tweet level which encodes the overall information of the tweet. Additionally, we can extract another feature vector on the word level to capture the context of the search term[6], i.e., the vector representation of the search term in: *"...my **diabetes** is neuropathy..."*.

Even though the feature vectors of the tweet level and word level views are not fully orthogonal, we argue that they still focus on different aspects of the text to represent the context of the tweet. Local and global feature sets have shown to be effective in other scenarios (Ghani et al., 2003). In the next section, we exploit this motif in an active learning framework.

3.2 Incorporating Context in Co-testing

Having two separate contextual representations for every user posting allows us to employ co-testing algorithm. However, the default co-testing query strategy and its variants (Muslea et al., 2006; Ghani

[5]The study by (Cui et al., 2019) employs the expected error reduction technique along a semi-supervised learning model.

[6]In the case that multiple search terms are used to collect the data, all the occurrences of the search terms in the tweets can be mapped to a single synthesized token.

et al., 2003) are unable to fully utilize the contextual information that is stored in the representations. These variations mostly rely on the confidence of base learners to score the candidate data points, e.g., most confident disagreement between base learners. We argue that the contextual representations that we extract contain enough information to detect similar user postings, and this information can be used to resolve the disagreement over a set of user postings, rather than one single user posting. This can potentially lead to a better annotation choice during the active learning iterations. Based on this argument, we propose the following query strategy.

Let $\vec{d}$ and $\vec{w}$ be the document and word level representations of the user posting t, and given t, let $Conf_D(\vec{d}|t)$ and $Conf_W(\vec{w}|t)$ be the confidence of the base learners for classification in the document level and word level views respectively. We define the score of the contention user posting t as follows:

$$score(t) = P_D(\vec{d}|t) \times Conf_D(\vec{d}|t) + P_W(\vec{w}|t) \times Conf_W(\vec{w}|t) \quad (1)$$

where $P_D(\vec{d}|t)$ and $P_W(\vec{w}|t)$ are the probabilities of the user posting t being generated by the distribution of the contention points in the document and word level views respectively. The terms $Conf_D(\vec{d}|t)$ and $Conf_W(\vec{w}|t)$ can be estimated by the output of the classifiers in the document and word level views respectively. To estimate $P_D(\vec{d}|t)$ and $P_W(\vec{w}|t)$, we first fit two density estimators on the vectors of the contention data points in each view to extract the empirical distribution of the population, and then use these estimators to calculate the probability of observing the data points[7].

Intuitively, Equation 1 assigns a higher score to the user postings that are confidently assigned to the opposite classes in two views, and are also close to the other set of contention points in each view. There are two advantages in employing this scoring function. First, scaling the confidence of the base learners by the probability densities naturally aggregates the benefits of contention reduction and density based query strategies. Second, assuming that the data points that are close to each other in the feature space are similar and likely to have the same label (Chapelle et al., 2003), by promoting the user postings that are close to the cluster of the contention points, we can effectively use the contextual information to resolve the disagreement over a set of similar user postings. This is particularly the case when a candidate data point and its adjacent points are projected into the same regions of the input feature space in both views.

Figure 1 demonstrates our query strategy. Each data point in the document representation space (the left panel) is associated to one data point in the keyword representation space (the right panel). The triangular data points are the set of contention tweets, i.e., the tweets that are assigned to the opposite classes by the classifiers in two views. The regular co-testing algorithm selects the data point with the largest distance from the classifier decision boundary–the dashed lines–i.e., the yellow data point. However, we select the data point which is close to the cluster of contention data points and also has a large distance from the classifier decision boundary, i.e., the black data point.

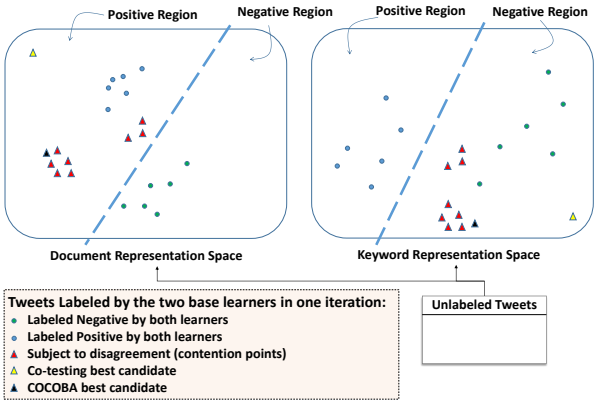


Figure 1: The document and word level views in COCOBA query strategy. The regular co-testing algorithm queries the contention point with the largest distance from the classifier decision boundary in two views (the yellow triangle). COCOBA queries the contention point which is closest to the set of other contention points and also has a large distance from the decision boundary in two views (the black triangle). Figure best viewed in color.

In the next sections, we use Equation 1 as the ranking function in our model.

3.3 Increasing Robustness to Noise in Social Media

As pointed out by (Karisani and Agichtein, 2018), the user postings in social media–particularly on the Twitter website–are highly noisy. They tend to be short, and suffer from inventive lexicons. For instance, in our early example of extracting the reports of diabetes, a user posting may be added to the set of contention points and selected for anno-

[7]For the theoretical discussion regarding the density estimators see (Silverman, 1986).

tation due to its unique figurative language. However, selecting another user posting for annotation might be a better choice to have a more diverse and representative training set. If we assume the relatively uninformative user postings are noise–which due to their unique characteristics may receive a high score by Equation 1–then we may be able to dampen their effect through variance reduction algorithms.

To address this issue we propose to employ bagging technique, which is empirically shown to reduce model variance (Buhlmann and Yu, 2002). In the discussed example, bagging can influence the score of the mentioned user posting, either through affecting the distribution of the contention user postings, or reducing the disagreement rate between two base learners. We use bagging as follows: In each iteration, we sample multiple subsets of user postings from the set of labeled data. On each subset, we train a pair of base learners as described in Section 3.1. For each pair of base learners, we use the model described in Section 3.2 to assign a score to all unlabeled user postings. Finally, the ultimate ranking list is constructed by aggregating the scores of the unlabeled data across the models.

Our approach for employing bagging is slightly different from the regular query-by-committee model (Abe and Mamitsuka, 1998). In the regular query-by-committee model, one estimator is trained on each subset of data, and the best candidate data point is the data point which is subject to the most *disagreement* among the estimators. In our model, the candidate data points, for each subset, are the data points that are assigned to opposite classes by the base learners. Then, each predictive model votes for these contention points, and the best candidate data point is the one that is subject to the most *agreement* among the models.

3.4 Overview of Algorithm

Algorithm 1 summarizes *one iteration* of COCOBA . Lines 10-21 describe the training procedure, and Lines 22-31 describe the labeling procedure. The training stage begins by sampling from the set of labeled tweets; then two base learners are trained on two views of the sampled set. Next, two base learners are used to label the set of unlabeled tweets. The contention tweets are detected, and in each view one density estimator is fitted. The density models are used to approximate the probability mass values of every contention tweet. These steps are repeated for each sub-sample. To rank the set of unlabeled tweets, the prediction confidences and probability mass values are used in Equation 1 to score all the contention tweets. The top tweet is queried and added to the labeled set and all the sampled sets–Line 19 and Line 20. Finally, all the base learners are re-trained on the updated sampled sets. In the labeling stage, each pair of the base learners is used to label the test tweets–Line 25. To predict the final label a majority voting algorithm is employed–Lines 28-31. In the next section, we discuss the implementation details of COCOBA.

Algorithm 1 One Iteration of COCOBA

1: **procedure** COCOBA
2: **Given:**
3: L : Set of labeled tweets
4: U : Set of unlabeled tweets
5: T : Set of test tweets
6: K : Number of estimators
7: **Return:**
8: Labeled set of test tweets, and updated training set
9: **Execute:**
10: **for** $i \leftarrow 1$ to K **do**
11: Sample a subset of L and store in $S[i]$
12: Train two base learners on $S[i]$ and store in $BL[i][0]$ and $BL[i][1]$
13: Use $BL[i][0]$ and $BL[i][1]$ to label the set U
14: Store the contention tweets in $C[i]$, and their prediction confidences in $Conf[i][0]$ and $Conf[i][1]$
15: Fit two density estimation models on two views of $C[i]$ and store them in $DS[i][0]$ and $DS[i][1]$
16: Use $DS[i][0]$ and $DS[i][1]$ to calculate the probability mass values for all the tweets in $C[i]$ and store them in $P[i][0]$ and $P[i][1]$
17: Plug the arrays $Conf$ and P into Equation (1) to calculate the aggregated score for tweets in C
18: Rank all the tweets in C based on their score, and store the top one in W
19: Query the label of W
20: Add W to L and all the tweet sets stored in S
21: Use the updated S to retrain the base learners of BL
22: **for** t in T **do**
23: $PCount \leftarrow 0$
24: **for** $pair$ in BL **do**
25: $label \leftarrow conf_{pair[0]}(t) + conf_{pair[1]}(t)$
26: **if** $label \geq 0$ **then**
27: $PCount \leftarrow PCount + 1$
28: **if** $PCount \geq K/2$ **then**
29: t is Positive
30: **else**
31: t is Negative
32: Return T, L

4 COCOBA: Implementation Details

In this section, first we discuss the feature vectors that we used in COCOBA. Then, we discuss the base learners and the density estimation models that

we implemented. Finally, we explain the details of the bagging step.

Feature vectors (Section 3.1): We used neural contextual word embeddings to represent the two contextual representations discussed in Section 3.1. We used the BERT pre-trained base model (Devlin et al., 2019), to extract the document level and word level views–the size of the vectors in this model is 768. For simplicity, if a task had multiple query words we assumed their contexts is comparable[8]–even though the approach in (Shi and Lin, 2019) could have been leveraged to create a canonical term. Additionally, If a user posting contained more than one search term, we selected the first occurrence to construct the word level view.

Base learners (Section 3.1): We used a one-layer fully connected network as the base learner. To account for the increasing size of the training set during the active learning iterations, we also updated the BERT vectors every few hundred iterations by fine-tuning–see Section 5.3 for detail.

Density estimators (Section 3.2): We used a Parzen density estimator to approximate the density of the contention points (Heidenreich et al., 2013). For simplicity, we opted for a linear kernel model. We set the bandwidth hyper-parameter in the document level view to 30, and in the word level view to 45–these values were determined based on the average distance of the data points in each view which is independent of the labeled data.

Bagging details (Section 3.3): There is no widely accepted number of estimators for the models based on bagging (Settles, 2009). We used 15 estimators in our implementation. For each estimator, we randomly sub-sampled 60% of the labeled set with replacement to be used as the training data.

5 Experimental Setup

We begin this section by describing the datasets, then we discuss the baselines, and finally, explain the experiments.

5.1 Datasets

We show that our model is applicable to three tasks: Personal Health Mention detection (PHM), Observation Extraction (OE), and Product Consumption Pattern identification (PCP). Below we describe the datasets.

[8]Recall that by definition, our tasks are defined for closely related search keywords.

	Training			Test		
Topic	Size	Neg	Pos	Size	Neg	Pos
Parkinson's	4096	84%	16%	2120	85%	15%
Cancer	3915	80%	20%	2091	79%	21%
Diabetes	4318	82%	18%	2097	86%	14%

Table 1: The number of tweets, and the percentage of the positive and negative tweets across the topics in Illness dataset.

Illness dataset: For PHM task, we constructed a dataset of English tweets across three different topics: Parkinson's disease, cancer, and diabetes. To collect the tweets related to diabetes, we used the search terms "diabetes" and "diabetic". We used the Twitter search API and retrieved a set of tweets–excluding retweets and replies–over the span of one year between 2018 and 2019. To create the training sets, we randomly sampled about 4,000 tweets for each topic from the 2018 data. To create the test sets, we randomly sampled about 2,000 tweets per topic from the 2019 data. To annotate the sampled sets, we followed the definition of Personal Health Mention detection problem (PHM), proposed in (Karisani and Agichtein, 2018). That is, the tweets that mention the health condition and contain a health report were labeled positive, otherwise, they were labeled negative. We hired one annotator to annotate the tweets. In order to validate the annotations, we randomly sub-sampled 10% of the labeled tweets, and hired another annotator to re-annotate the set. We found the inter-agreement rate to be 0.81 with Cohen Kappa test, which represents a substantial agreement between the two annotators (Viera and Garrett, 2005). Table 1 summarizes Illness dataset. We see that on average about 18% of the tweets are positive in each topic.

Observation dataset: For OE task, we used the dataset introduced in (Zahra et al., 2020) on reporting flood incidents, which contains 4,000 tweets[9]. Each tweet is categorized as Direct-Observation, Indirect-Observation, or None. We assumed the tweets that make a direct observation are positive–which account for 17% of the dataset. With preserving the original distribution, we sampled 1,000 tweets for the test set. Query keywords used to collect the dataset are "flood", "rain", and "overflow".

Product dataset: For PCP task, we used the dataset introduced in (Huang et al., 2017). This dataset[10] consists of the tweets related to a medi-

[9]Available at https://crisisnlp.qcri.org/

[10]Publicly available via the organizers of SMM4H workshop: https://aclweb.org/portal/content/smm4h

cal product–influenza vaccine. A tweet is labeled positive if it reports receiving the medical product. There are 6,617 tweets in this dataset. We used the tweets posted in 2013 and 2014 in the training set, and the tweets posted in 2015 and 2016 in the test set. In the training set, we found 4,503 tweets for which 31% of them were positive. In the test set, we found 2,114 tweets for which 22% were positive.

5.2 Baselines

In this section, we describe the baseline models that we included in the experiments. We included one naive baseline (random sampling), one classic baseline (uncertainty sampling), one learning-from-data model (LAL), and one self-paced learning model (SPAL). In Section 6.2 we also compare our model with the co-testing algorithm. The input features were identical between all the models–as described in Section 4.

random: This baseline is without Active Learning. In each iteration, we randomly selected one tweet from the set of unlabeled tweets, and added to the labeled set.

uncertainty: We included the most widely used uncertainty-based model described in (Settles, 2009). The output probability of the base learner was used as the confidence score.

lal: We included the model proposed in (Konyushkova et al., 2017)[11]. This model is an error reduction algorithm, which models the query sampling problem as a regression task. We report the *Iterative* variant, which is a stronger baseline and performed better. We used the suggested settings in the reference to set-up the model.

spal: We included the model proposed in (Tang and Huang, 2019)[12]. This model is a self-paced method, which tries to maintain a balance between the informativeness and the easiness of queries through an objective function. We used the settings proposed in the reference to set-up the model.

5.3 Experimental Details

We trained and evaluated all of the models in each topic of Illness, Observation, and Product datasets separately. Following the argument in (Mccreadie et al., 2019), we report the F1 of the models in the positive set. The rest of the experimental setup was identical to what is adopted in the active learning literature (Settles, 2009; Lowell et al., 2019). In the cold start state, we randomly sampled 50 labeled tweets, and assumed that the rest of the labeled data is unlabeled. We report F1 measure in the test set as the training set is augmented with new labeled tweets. We fixed the initial set of labeled tweets across all the experiments, ensuring that all of the models have access to an identical set of tweets in their cold start state. Additionally, we repeated all the experiments 5 times and report the average of the experiments. In order to account for the increasing size of the training sets during the active learning iterations, every 350 iterations we fine-tuned the BERT model–mentioned in Section 4–and updated the entire set of tweet and word representations in all the baseline models.

6 Results and Analysis

In this section we report the main results, and then we provide an empirical analysis.

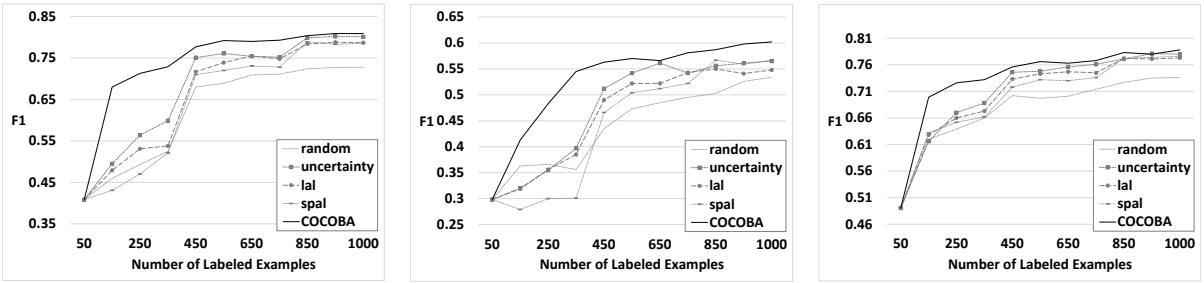
6.1 Results

Figures 2a, 2b, and 2c report the performance of the models in Illness , Observation , and Product datasets respectively. Additionally, Table 2 compares the performances at four different ratios of the training set sizes, i.e., 25%, 50%, 75%, and 100%. The results confirm that–except in a few cases–all the models outperform *random* baseline, confirming that Active Learning is an effective strategy to approach these tasks. The results signify that our model COCOBA is consistently outperforming the baselines. This is particularly the case over the initial iterations. During these iterations our model employs two views to issue the queries, whereas the other models rely on one view. As more training data becomes available, and the pool of unlabeled data shrinks, the models converge–except in Observation dataset. Finally, the experiments show that *uncertainty* model is performing strikingly well, confirming the consistency of this model–discussed in Section 2. The authors in (Attenberg and Provost, 2011) report that under different problem settings state-of-the-art active learning models may be inferior to the uncertainty model.

6.2 Empirical Analysis

In Section 3.2 we argued that the regular co-testing algorithm can be further improved by exploiting the density of the contention points. We also proposed a method to incorporate this information using a

[11] Available at https://github.com/ksenia-konyushkova/LAL
[12] Available at https://github.com/NUAA-AL/ALiPy



(a) Illness (b) Observation (c) Product

Figure 2: F1 of the models at varying training set sizes during the active learning iterations in all three datasets.

	F1 in Illness dataset				F1 in Observation dataset				F1 in Product dataset			
Model	**25%**	**50%**	**75%**	**100%**	**25%**	**50%**	**75%**	**100%**	**25%**	**50%**	**75%**	**100%**
random	0.513	0.688	0.679	0.728	0.360	0.468	0.403	0.533	0.651	0.706	0.715	0.736
uncertainty	0.584	0.757	0.790	0.801	0.359	0.541	0.551	0.565	0.682	0.750	**0.774**	0.780
lal	0.530	0.731	0.778	0.787	0.340	0.514	**0.566**	0.547	0.669	0.734	0.763	0.772
spal	0.503	0.718	0.778	0.786	0.312	0.492	0.506	0.567	0.656	0.722	0.762	0.776
COCOBA	**0.723***	**0.788***	**0.804***	**0.809**	**0.522***	**0.573***	0.559	**0.602***	**0.738***	**0.761**	**0.774**	**0.788**

Table 2: F1 of the models at 25%, 50%, 75%, and 100% of the training set sizes during the active learning iterations. The improvements indicated by * are statistically significant–using paired t-test (adjusted $P < 0.05$).

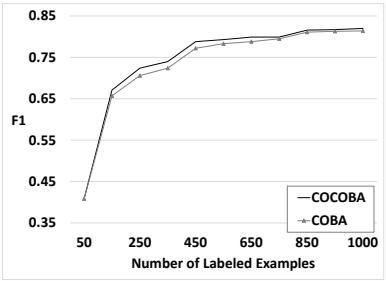


Figure 3: F1 of COCOBA and COBA (COCOBA without context) at varying training set sizes in Illness dataset.

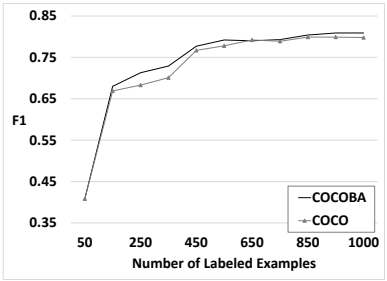


Figure 4: F1 of COCOBA and COCO (COCOBA with no bagging) at varying training set sizes in Illness dataset.

Parzen density estimator. To support our claim and also to evaluate the proposed method, we compare COCOBA with a variant of this method that does not use the context, i.e., this method only uses the classifier confidences–see Equation 1. Figure 3 reports the result of this experiment. We see that the performance of our model is noticeably higher than that of the new model, which we call COBA (Co-testing with Bagging).

In Section 3.3 we argued that a variance reduction technique can mitigate the problem caused by the noisy language model. To support this argument we deactivated the robustness step[13] and evaluated the resulting model which we call COCO (Context-aware Co-testing without bagging). Figure 4 reports the result of this experiment in Illness dataset. We see that the new model COCO is inferior to COCOBA. The figure shows that the new model is particularly outperformed during the early iterations. Our error analysis revealed that due to the small training set during these iterations the base learners are more prone to querying relatively uninformative tweets, which explains why this span is impacted most by the variance reduction technique.

Next, we compare our model with a variant of co-testing (Muslea et al., 2006) which is customized and adapted to event extraction task in (Liao and Grishman, 2011). All the settings were set to be identical between both models for this experiment–we used our idea of constructing two contextual representations in both models. Thus, this experiment focuses on evaluating the ideas of contextu-

[13] In this experiment, bagging is deactivated in both query and labeling stages. Similar results can be achieved if we deactivate bagging only in query stage.

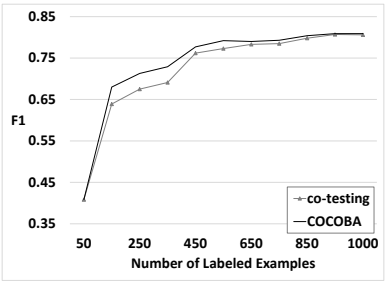


Figure 5: F1 of *co-testing* and COCOBA at varying training set sizes in Illness dataset.

alization and robustness proposed in Sections 3.2 and 3.3. Figure 5 reports the outcome of this experiment in Illness dataset. The results show that the improvement over co-testing model is consistent, however, as the training set grows–and the set of unlabeled data shrinks–both models converge.

A closer look at the graphs in Figure 2 shows the existence of an elbow point in the early iterations. The improvement rate before reaching this point is dramatic and after this point it is slower. Our case by case inspection revealed that during the early iterations our scoring function–described in Section 3.2–can effectively use the density of the contention points. However, as the algorithm proceeds, the set of contention points is exhausted and our model converges to a regular contention reduction algorithm. Thus, we conjecture that in the presence of larger set of unlabeled data COCOBA may yield even better results[14]. One particularly interesting quality of our model is the absence of critical hyper-parameters to tune. Excluding the hyper-parameters of the base learners, which is shared between all the models, in our experiments COCOBA was not sensitive to the number of estimators in the bagging step or the value of the bandwidth in the kernel density estimators[15].

In summary, we showed that our active learning model outperforms the state of the art in multiple settings. The authors in (Attenberg and Provost, 2011) report that active learning models typically show mixed results and fail to generalize to new scenarios. Thus, we selected three datasets and also included two state-of-the-art and two traditional baselines and showed that our model consistently perfroms well. The results suggest that our model can be potentially applied to a broader set of query-based classification tasks. This claim is to be further investigated. Additionally, there is still a set of social media tasks that are not based on queries e.g., sarcasm detection, hate speech detection, and fake news identification. Future work may explore these areas.

[14]In terms of runtime, COCOBA is comparable to *lal*–which is also an ensemble. In the experiments, *spal* performed much slower.

[15]We tried {10,15,20} estimators, the results were consistent.

7 Conclusions

In this paper we proposed a novel active learning model for short text classification tasks in user-generated data. Our model utilizes the contextual information of user postings in a multi-view active learning model, exploits the density of the contention points to increase the gain per query, and employs a query-by-committee step to address the usually noisy language of social media posts. Through an extensive set of experiments we showed that our model, COCOBA, is applicable to multiple tasks. Our code and a relatively large dataset that we constructed along the way are publicly available.

Acknowledgements

We thank the anonymous reviewers for their insightful feedback. Li Xiong is partially supported by the National Science Foundation (NSF) under CNS-1952192 and National Institutes of Health (NIH) under CTSA UL1TR002378.

References

Naoki Abe and Hiroshi Mamitsuka. 1998. Query learning strategies using boosting and bagging. In *Proc of the 5th ICML*, pages 1–9.

Raj Agnihotri, Rebecca Dingus, Michael Y. Hu, and Michael T. Krush. 2016. Social media: Influencing customer satisfaction in b2b sales. *Industrial Marketing Management*, 53:172 – 180.

Josh Attenberg and Foster Provost. 2011. Inactive learning?: Difficulties employing active learning in practice. *KDD Exp. News.*, 12:36–41.

Peter Buhlmann and Bin Yu. 2002. Analyzing bagging. *Ann. Statist.*, 30(4):927–961.

Sophie Burkhardt, Julia Siekiera, Josua Glodde, Miguel A Andrade-Navarro, and Stefan Kramer. 2020. Towards identifying drug side effects from social media using active learning and crowd sourcing. In *Pacific Symposium of Biocomputing (PSB)*, pages 319–330.

Haw-Shiuan Chang, Shankar Vembu, Sunil Mohan, Rheeya Uppaal, and Andrew McCallum. 2019. Overcoming practical issues of deep active learning and its applications on named entity recognition. *arXiv preprint arXiv:1911.07335*.

Olivier Chapelle, Jason Weston, and Bernhard Schölkopf. 2003. Cluster kernels for semi-supervised learning. In *NIPS 15*, pages 601–608. MIT Press.

Jongwon Choi, Kwang Moo Yi, Jihoon Kim, Jincho Choo, Byoungjip Kim, Jin-Yeop Chang, Youngjune Gwon, and Hyung Jin Chang. 2020. Vab-al: Incorporating class imbalance and difficulty with variational bayes for active learning. *arXiv preprint arXiv:2003.11249*.

Hang Cui, Tarek Abdelzaher, and Lance Kaplan. 2019. A semi-supervised active-learning truth estimator for social networks. In *The World Wide Web Conference*, WWW '19, page 296–306, New York, NY, USA. Association for Computing Machinery.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proc of the 2019 NAACL*, pages 4171–4186.

Anne Dirkson and Suzan Verberne. 2019. Transfer learning for health-related twitter data. In *Proc of the Fourth Social Media Mining for Health Applications (#SMM4H) Workshop*, pages 89–92. ACL.

Yarin Gal, Riashat Islam, and Zoubin Ghahramani. 2017. Deep bayesian active learning with image data. In *Proc of the 34th ICML*, pages 1183–1192.

Rayid Ghani, Rosie Jones, Tom Mitchell, and Ellen Riloff. 2003. Active learning for information extraction with multiple view feature sets. In *Proc of the 20th ICML*, pages 26–34.

Nils-Bastian Heidenreich, Anja Schindler, and Stefan Sperlich. 2013. Bandwidth selection for kernel density estimation: a review of fully automatic selectors. *AStA Advances in Statistical Analysis*, 97(4):403–433.

Xiaolei Huang, Michael C Smith, Michael J Paul, Dmytro Ryzhkov, Sandra C Quinn, David A Broniatowski, and Mark Dredze. 2017. Examining patterns of influenza vaccination in social media. In *Workshops at the 31st AAAI*.

Muhammad Imran, Carlos Castillo, Fernando Diaz, and Sarah Vieweg. 2015. Processing social media messages in mass emergency: A survey. *ACM Comput. Surv.*, 47(4):67:1–67:38.

Khaled Jedoui, Ranjay Krishna, Michael Bernstein, and Li Fei-Fei. 2019. Deep bayesian active learning for multiple correct outputs. *arXiv preprint arXiv:1912.01119*.

Zhuoren Jiang, Zhe Gao, Yu Duan, Yangyang Kang, Changlong Sun, Qiong Zhang, and Xiaozhong Liu. 2020. Camouflaged Chinese spam content detection with semi-supervised generative active learning. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3080–3085, Online. Association for Computational Linguistics.

Negin Karisani and Payam Karisani. 2020. Mining coronavirus (covid-19) posts in social media. *arXiv preprint arXiv:2004.06778*.

Payam Karisani and Eugene Agichtein. 2018. Did you just have a heart attack?: Towards robust detection of personal health mentions in social media. In *Proc of the 2018 WWW*, pages 137–146.

Payam Karisani, Joyce C. Ho, and Eugene Agichtein. 2020. Domain-guided task decomposition with self-training for detecting personal events in social media. In *Proceedings of The Web Conference 2020*, WWW '20, page 2411–2420.

Payam Karisani and Negin Karisani. 2021. Semi-supervised text classification via self-pretraining. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*, WSDM '21, page 40–48. Association for Computing Machinery.

Payam Karisani, Farhad Oroumchian, and Maseud Rahgozar. 2015. Tweet expansion method for filtering task in twitter. In *Experimental IR Meets Multilinguality, Multimodality, and Interaction*, pages 55–64.

Ksenia Konyushkova, Raphael Sznitman, and Pascal Fua. 2017. Learning active learning from data. In *Advances in Neural Information Processing Systems 30*, pages 4225–4235. Curran Associates, Inc.

David D. Lewis and William A. Gale. 1994. A sequential algorithm for training text classifiers. In *Proc of the 17th SIGIR*, pages 3–12.

Chao Li, Xin Kang, and Fuji Ren. 2017. Medweb task: Identify multi-symptoms from tweets based on active learning and semantic information. In *Proc of the 13th NTCIR*, pages 5–8.

Shasha Liao and Ralph Grishman. 2011. Using prediction from sentential scope to build a pseudo co-testing learner for event extraction. In *Proc of 5th IJCNLP*, pages 714–722.

David Lowell, Zachary C. Lipton, and Byron C. Wallace. 2019. Practical obstacles to deploying active learning. In *Proc of the 2019 EMNLP*, pages 21–30.

R. Mccreadie, C. Buntain, and I. Soboroff. 2019. Trec incident streams: Actionable information on social media. In *Proc of the 16th ISCRAM*.

S. Mei, H. Li, J. Fan, X. Zhu, and C. R. Dyer. 2014. Inferring air pollution by sniffing social media. In

2014 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2014), pages 534–539.

Ion Muslea, Steven Minton, and Craig A. Knoblock. 2006. Active learning with multiple views. *J. Artif. Int. Res.*, 27(1):203–233.

Michael J. Paul and Mark Dredze. 2017. *Social Monitoring for Public Health*. Synthesis Lectures on Information Concepts, Retrieval, and Services. Morgan & Claypool Publishers.

Daniela Pohl, Abdelhamid Bouchachia, and Hermann Hellwagner. 2018. Batch-based active learning: Application to social media data for crisis management. *Expert Systems with Applications*, 93:232 – 244.

Bianca Scarlini, Tommaso Pasini, and Roberto Navigli. 2020. Sensembert: Context-enhanced sense embeddings for multilingual word sense disambiguation. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 8758–8765. AAAI Press.

Burr Settles. 2009. Active learning literature survey. Computer Sciences Technical Report 1648, University of Wisconsin–Madison.

Peng Shi and Jimmy Lin. 2019. Simple bert models for extraction and semantic role labeling. *arXiv preprint arXiv:1904.05255*.

Bernard W Silverman. 1986. *Density estimation for statistics and data analysis*, volume 26. CRC press.

Damiano Spina, Maria-Hendrike Peetz, and Maarten de Rijke. 2015. Active learning for entity filtering in microblog streams. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '15, page 975–978, New York, NY, USA. Association for Computing Machinery.

Gabriel Stanovsky, Daniel Gruhl, and P Mendes. 2017. Recognizing mentions of adverse drug reaction in social media using knowledge-infused recurrent models. In *Proc of the 15th EACL*, pages 142–151.

Ying-Peng Tang and Sheng-Jun Huang. 2019. Self-paced active learning: Query the right thing at the right time. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 5117–5124. AAAI Press.

Van Cuong Tran, Ngoc Thanh Nguyen, Hamido Fujita, Dinh Tuyen Hoang, and Dosam Hwang. 2017. A combination of active learning and self-learning for named entity recognition on twitter using conditional random fields. *Knowledge-Based Systems*, 132:179 – 187.

A. J. Viera and J. M. Garrett. 2005. Understanding interobserver agreement: the kappa statistic. *Fam Med*, 37(5):360–363.

Zhijun Yin, Lina M Sulieman, and Bradley A Malin. 2019. A systematic literature review of machine learning in online personal health data. *J. of American Medical Informatics Association*, 26:561–576.

Kiran Zahra, Muhammad Imran, and Frank O. Ostermann. 2020. Automatic identification of eyewitness messages on twitter during disasters. *Information Processing & Management*, 57(1):102107.

Ye Zhang, Matthew Lease, and Byron C. Wallace. 2017. Active discriminative text representation learning. In *Proc of the Thirty-First AAAI Conference on Artificial Intelligence*, AAAI'17, pages 3386–3392.

Yunpeng Zhao, Mattia Prosperi, Tianchen Lyu, Yi Guo, and Jing Bian. 2020. Integrating crowdsourcing and active learning for classification of work-life events from tweets. *arXiv preprint arXiv:2003.12139*.