The Use of Unlabeled Data versus Labeled Data for Stopping Active Learning for Text Classification

Garrett Beatty†
Department of Computer Science
The College of New Jersey
Ewing, NJ 08628
Email: beattyg2@tcnj.edu

Ethan Kochis†
Department of Computer Science
The College of New Jersey
Ewing, NJ 08628
Email: kochise1@tcnj.edu

Michael Bloodgood
Department of Computer Science
The College of New Jersey
Ewing, NJ 08628
Email: mbloodgood@tcnj.edu

***Abstract*— Annotation of training data is the major bottleneck in the creation of text classification systems. Active learning is a commonly used technique to reduce the amount of training data one needs to label. A crucial aspect of active learning is determining when to stop labeling data. Three potential sources for informing when to stop active learning are an additional labeled set of data, an unlabeled set of data, and the training data that is labeled during the process of active learning. To date, no one has compared and contrasted the advantages and disadvantages of stopping methods based on these three information sources. We find that stopping methods that use unlabeled data are more effective than methods that use labeled data.**

I. Introduction

The use of active learning to train machine learning models has been used as a way to reduce annotation costs for text and speech processing applications [1], [2], [3], [4], [5]. Active learning has been shown to have a particularly large potential for reducing annotation cost for text classification [6], [7]. Text classification is one of the most important fields in semantic computing and it has been used in many applications [8], [9], [10], [11], [12].

Data annotation is a major bottleneck in developing new text classification systems. Active learning is a method that can be used to reduce this bottleneck whereby the machine actively selects which data to have labeled for training. The careful selection of the data to be labeled enables the machine to learn high performing models from smaller amounts of data than if passive learning were used. The active learning process is shown in Algorithm 1.

An important aspect of the active learning process is the stopping criterion as shown in Algorithm 1. Stopping methods enable the potential benefits of active learning to be achieved in practice. Without stopping methods, the active learning process would continue until the entire unlabeled pool has been annotated, which would defeat the purpose of active learning. Consequently, many stopping methods have been researched to achieve the benefits of active learning in practice [13], [14], [15], [16], [17], [18], [19], [20].

The purpose of active learning is to reduce the data annotation bottleneck by carefully selecting the data to be labeled. To avoid labeling any additional data, active learning stopping methods have been developed that use only unlabeled data to stop the active learning process. It has been suggested that using labeled data would be a straightforward way to stop the active learning process, but stopping methods using labeled data have not been thoroughly explored because of the extra cost of labeling the data. However, the use of labeled data might make stopping methods so much more effective that the extra cost of the labeled data is worthwhile. To date, investigating whether the advantages of using labeled data outweigh the disadvantages of using labeled data for determining when to stop active learning has not been explored. In this paper, we compare stopping methods using unlabeled data with stopping methods using labeled data to see if the additional cost of labeling the data for the purpose of determining when to stop is worthwhile. We find that not only is the extra labeling cost not worthwhile, but stopping methods using unlabeled data actually perform better than stopping methods using labeled data.

```
Input:
        $U$ = large pool of unlabeled data
        $L$ = empty pool of labeled data
        $b$ = batch size
$L \leftarrow$ select $b$ random examples from $U$ and request their labels;
$U = U - L$
Loop until stopping criterion is met
    Train model using $L$;
    $batch \leftarrow$ select $b$ examples from $U$ using selection algorithm and request their labels;
    $U = U - batch$;
    $L = L \cup batch$;
End
```

Algorithm 1: Active Learning Algorithm

Section II explains our methodology. Section III discusses related work. Section IV provides details about our experimental setup. Section V presents the results of our experiments and section VI concludes.

†These students contributed equally to this paper.

This paper was published in the Proceedings of the 2019 IEEE 13th International Conference on Semantic Computing (ICSC), pages 287-294, Newport Beach, CA, USA, 2019. ©2019 IEEE
Link to article abstract in IEEE Xplore: http://dx.doi.org/10.1109/ICOSC.2019.8665546

II. Methodology

A. Stopping Method Information Sources

One could classify the information sources that stopping methods use into three categories:

(i) unlabeled data,
(ii) small labeled data, and
(iii) training data labeled during the active learning process.

The first category is unlabeled data. Stopping methods that use unlabeled data allow for the full potential of active learning to be realized because a stopping point is found without incurring any additional labeling cost.

The second category of data is a small labeled set. Following [19], we will refer to this set as a *validation set* in the rest of this paper. Using a validation set to stop the active learning process would appear to be the most direct way to stop the active learning process. Having a validation set would mean that the performance of the model could be approximated. However, creating a validation set means annotating examples before the active learning process begins. This might defeat the purpose of active learning, since examples are being annotated that may not be requested by the selection algorithm throughout the training process.

The third category of data is created during the active learning process: the training data. Formalized as L in Algorithm 1, this is the data that is labeled in order to train a model. Since the training data is already labeled, one can use it to determine when to stop active learning without incurring additional labeling cost.

Unlabeled data is a potentially large set of unlabeled examples. Since the examples are unlabeled, the data can be made as large as needed to be as representative of the application space as desired. The validation set does not contain artificial sources of bias and does contain labels, but it has to be relatively small due to the extra labeling cost. The training data contains labels and can be of moderate size, but it is systematically biased due to how it is selected. The size of the training set is moderate as it grows over time. It is not clear which information source, or combination of them, is most effective for stopping active learning.

B. Stopping Methods That Use Unlabeled Data

Several stopping methods for active learning have been researched for the field of text classification. Schohn and Cohn created a stopping method, which we denote as SC2000, that will stop the active learning process when the model's confidence values of the unlabeled data are outside of the model's margin [15]. This method can only be used with margin-based learners such as Support Vector Machines (SVMs). Vlachos devised a stopping method, which we denote as V2008, that will stop active learning when the confidence values of the unlabeled data drops consistently for three consecutive models [18]. Laws and Schütze investigated a stopping method, which we denote as LS2008, that will stop active learning when the gradient of model confidence values is less than a user-specified threshold [17]. The gradient is calculated using the medians of the averages of the confidence values of the selected batches of examples for k iterations of active learning, where k is a user-specified parameter. Zhu, Wang, and Hovy created a stopping method, which we denote as ZWH2008, that uses multiple criteria. First, it will check if the accuracy on the next batch of training data exceeds a threshold. Then, it will stop active learning when the classifications of the unlabeled pool did not change from the previous model's predictions [16]. Bloodgood and Vijay-Shanker developed the Stabilizing Predictions (SP) stopping method. We denote this method as BV2009. This method examines the predictions of consecutively trained models on an unlabeled set of data, referred to as a stop set. The method stops active learning when the agreement of consecutively trained models on the stop set is greater than a user-specified threshold [13]. Bloodgood and Grothendieck then improved SP with an added variance check to dynamically adjust the stop set size as needed [14]. We denote this method as BG2013.

In [13], [14], [21], and in our results in section V, SP is shown to be a leading stopping method that uses unlabeled data. Therefore, in the rest of this paper, we use the SP stopping method as representative of the state of the art of stopping methods that use unlabeled data.

C. Stopping Methods That Use Labeled Data

Several stopping methods have been suggested that use labeled data. One such method is the Performance Threshold method that will stop the active learning process after the mean of model performance for a user-defined amount of iterations exceeds a user-defined threshold [22]. This method ensures that the model is reaching a performance level that the user deems acceptable. Another method is the Performance Difference method that will stop the active learning process once the mean of model performance differences for a user-defined amount of iterations is less than a user-defined threshold [13], [18], [19], [22]. This method determines when the performance on the labeled set levels off. These methods can be used with a validation set and with the training data. To our knowledge, these methods have never been implemented or tested.

D. Stopping Methods That Use Multiple Data Sources

Stopping methods that use both unlabeled data and labeled data have not been discussed in previous work. We combine our labeled data stopping methods with Stabilizing Predictions [13] with the variance check described in [14] in four ways:

(i) SP $\wedge$ Performance Threshold
(ii) SP $\wedge$ Performance Difference
(iii) SP $\vee$ Performance Threshold
(iv) SP $\vee$ Performance Difference

The SP $\wedge$ Performance Threshold method and the SP $\wedge$ Performance Difference method stop the active learning process when both SP and the labeled data stopping method indicate to stop. The SP $\vee$ Performance Threshold method and the SP $\vee$ Performance Difference method stop the active learning process when either SP or the labeled data stopping method indicate to stop. Using terminology introduced in [13], SP $\wedge$ Labeled Data stopping methods are more *conservative* and the stopping points are guaranteed to be at least as late as max(SP stopping point, labeled data stopping point). On the other hand, SP $\vee$ Labeled Data stopping methods are more *aggressive* and stop at least as early as min(SP stopping point, labeled data stopping point).

III. Related Work

A. Using Unlabeled Data for Stopping

Past work using unlabeled data for stopping active learning was discussed in section II-B.

B. Using Labeled Data for Stopping

A validation set, or a small labeled set, is one way of stopping the active learning process [15]. Labeling data that might not be used in the training process, however, defeats the purpose of active learning [15]. Determining the size of the validation set is an open question [19]. If the validation set is too small, it might not be representative of what can be learned, resulting in skewed stopping points [18], [19]. However, making a larger validation set would increase the cost, defeating the purpose of active learning [19]. We investigate different validation set sizes in section V-C. Although stopping using a validation set has been discussed as a possibility, to our knowledge, stopping methods using labeled data have never been implemented or tested. We examine the performance of stopping methods that use a validation set in section V-D. Also, although we don't want our separate held-out test set to have any overlap with training examples, it is less clear whether the advantages of allowing the training set to overlap with the validation set outweigh the disadvantages. We explore this in section V-B.

Using cross-validation on the training set has been discussed as an information source for stopping methods. Schohn and Cohn [15] stated that the time needed to re-train an SVM model would make this information source impractical to use. They also stated that the distribution created by the training set might not be representative of the test set distribution [15]. This means that data collected from the cross-validation on the training set could be skewed in relation to the data collected from a test set. It is known that actively sampled data can be quite skewed from randomly sampled data [23]. However, using data labeled for training has the advantage of being able to use relatively large amounts of labeled data without incurring any additional cost. To our knowledge, previous work has not examined using the training data to stop active learning. We examine the performance of stopping methods that use cross-validation on the training data in section V-E.

C. Other Related Work

Small labeled sets have also been used in other areas of active learning. A small labeled set can be used to estimate the ratio of negative to positive examples in an entire corpus to build a cost-weighted SVM [23]. Neural networks can stop training by using the performance score on a small labeled set [24], [25], [26]. Finally, a small labeled set can be used to build a biased SVM when no negative examples are present in the training set [27]. None of these works considered using small labeled sets to stop the active learning process, which we experiment with in section V-D.

IV. Experimental Setup

We use the 20NewsGroups dataset[1], the Reuters dataset[2], the WebKB dataset[3], and the spamassassin corpus[4] for our experiments. For the Reuters dataset, we use the ten largest categories from the Reuters-21578 Distribution 1.0 ModApte split, as in [28] and [29]. Consistent with previous work, we report the results for the four largest categories of the WebKB dataset [30], [19], [16]. Averages for the 20NewsGroups and Reuters datasets were taken across the categories. Averages for the categories of SpamAssassin and WebKB were taken over a 10-fold cross-validation. We use a Support Vector Machine as our classifier and use the closest-to-hyperplane selection algorithm [15], [7], [31]. This selection algorithm was recently found to have better performance than other selection algorithms [32]. We use a batch size that is equivalent to 0.5% of the initial unlabeled pool for each dataset and keep adding this amount of new examples for each iteration of active learning. For text classification, we use binary features for every word that occurs more than three times and remove stop words that appear in the *Long Stopword List* from https://www.ranks.nl/stopwords.

A. Validation Set

We build the validation set by randomly selecting examples from the unlabeled pool. When using a validation set, two important questions arise:

(i) How big should the validation set be?
(ii) Should examples from the validation set be allowed to be selected for training during active learning?

In section V we present results of experiments investigating these questions.

B. Training Data Cross Validation

As mentioned in section II, training data itself could be used by stopping methods. In order to do this, we use 10-fold cross-validation (CV) on the training data, as shown in Algorithm 2.

C. Stopping Method Parameters

The Performance Difference method uses ϵ as its threshold of F-Measure difference between the active learning iterations. A larger value of ϵ would cause the method to be more

[1] Downloaded the "bydate" version from http://qwone.com/~jason/20Newsgroups/ on July 13, 2017. This version does not include duplicate posts and is sorted by date into train and test sets.

[2] Downloaded from http://www.daviddlewis.com/resources/testcollections/reuters21578/ on July 13, 2017.

[3] Downloaded from http://www.cs.cmu.edu/afs/cs.cmu.edu/project/theo-20/www/data/ on July 13, 2017.

[4] Downloaded the latest versions of the 5 distinct sets from http://csmining.org/index.php/spam-assassin-datasets.html?file=tl_files/Project_Datasets/SpamAssassin%20data/ on July 13, 2017.

```
Input:
        $U$ = large pool of unlabeled data
        $L$ = empty pool of labeled data
        $b$ = batch size
        $p$ = empty array of performance scores
        $p\_avg$ = empty array of performance score averages
$L \leftarrow$ select $b$ random examples from $U$ and request their labels;
$U = U - L$
Loop until stopping criterion is met
    $S \leftarrow$ partition $L$ into 10 sets;
    for $i \leftarrow 1$ to $10$ do
        $model_i \leftarrow$ Train model using $S - S[i]$;
        $p[i]$ = Test $model_i$ using $S[i]$;
    end
    $p\_avg \leftarrow$Average $p[1 \ldots 10]$;
    $batch \leftarrow$ select $b$ examples from $U$ using selection algorithm and request their labels;
    $U = U - batch$;
    $L = L \cup batch$;
End
```

Algorithm 2: Active Learning Algorithm Using 10-fold CV on L

aggressive, as it would stop when the performance is still increasing at a faster rate. A smaller value of ϵ would cause the method to be more conservative, as it would only stop when performance changes have become smaller. By default, we use an ϵ value of 0.005: half of a percentage point of F-Measure. Half of a percentage point of F-Measure was chosen as a default value for ϵ since learning will be relatively stable, while still allowing for some fluctuations due to noise and random events.

The Performance Threshold method uses τ as its threshold value. This value is representative of the performance level of the model the user wants to achieve. A larger value of τ would lead to a more conservative method. A smaller value of τ will cause the method to be more aggressive. By default, we set τ to 0.8, or 80% F-Measure. In many cases, a model that has a performance level of 80% F-Measure is considered reasonable. Setting τ is more difficult and dataset-dependent than setting ϵ because the level of performance that is acceptable depends heavily on the task and dataset whereas the level of ϵ that indicates a leveling off in performance is not so heavily dependent on the task and dataset.

Both the Performance Difference and the Performance Threshold method look back w iterations of active learning to determine if the models' performance on the validation set has leveled off or has sustained a user-defined level of performance for w iterations. A relatively small value of w would mean that the models' performance does not have to be stable or above a certain value for many iterations of active learning. If w is too small, the method becomes more aggressive. A larger value of w would mean that the performance needs to be stable or above a certain value over more iterations, which would help avoid the risk of stopping too early. However, using a larger w means one would need more labeled data for the increased number of iterations, causing the method to become more conservative. Following previous work, we set w to three [13], [18]. As [33] advised, if a relatively large batch size is used, a smaller value for w should be used in order to mitigate the degradation in stopping method performance caused when using larger batch sizes.

V. Results

A. Unlabeled Stopping Methods

Table I shows the performance of unlabeled data stopping methods. SP, one of the most widely applicable and easy-to-implement methods, has leading performance, consistent with past findings [13], [14], [21]. Accordingly, we use SP as representative of state of the art unlabeled data stopping methods in the rest of our experiments.

B. Effect of Allowing Validation Set Examples to be Selected for Training

There is a potential validation set performance estimation bias[5] when validation set examples are allowed to be selected for training. We examine the impact of allowing validation set examples to be selected for training on the performance metric F-Measure using a validation set size of 500.

There are two main benefits when validation set examples are allowed to be selected for training. The first benefit is that the overall test set performance is higher, as seen in Figure 1. The reason for this performance increase is because high value examples that were in the validation set were allowed to be used for training. If validation set examples were not allowed to be selected for training, the model's learning efficiency may be hurt because the high value examples can not be used to improve the model. The second benefit is that when a training example is selected from the validation set, it can be used without any extra labeling cost.

As mentioned before, the other option is to not allow validation set examples to be selected for training. The main benefit of this approach is that the validation set estimate of performance will be a better approximation of test set performance, as seen in Figure 2. The reason that it more closely approximates the test set performance than the first approach is because there is no performance estimation bias from examples in the training data also being in the validation set.

Note, however, that the validation set performance curves in Figure 2 qualitatively have the same shape when examples are allowed in the training data as when they're not allowed. The Performance Difference method can use this behavior effectively to determine when to stop. The Performance Threshold method would not be able to use this behavior, but this does not matter because the Performance Threshold method does

[5] Note there is no test set performance estimation bias since the test set is a completely held-out separate set of data with no overlap with any other data.

Datasets	SP (BV2009/BG2013)	SC 2000	V 2008	LS 2008	ZWH 2008	ALL
20NewsGroups	823	**1915**	748	**513**	877	11314
(20-cat AVG)	73.36	74.34	**47.24**	67.09	73.64	74.59
Reuters	691	**1267**	2286	628	739	9655
(10-cat AVG)	77.94	78.12	**58.59**	71.60	78.18	77.70
SpamAssassin-spam	294	**847**	**5441**	1292	**378**	5441
(10-fold AVG)	98.10	**98.78**	**98.91**	96.34	98.47	98.91
WebKB-course	669	**1332**	2314	**370**	**810**	7445
(10-fold AVG)	84.96	86.12	**68.35**	**75.49**	85.83	83.44
WebKB-faculty	728	**1306**	614	**325**	**950**	7445
(10-fold AVG)	86.29	87.22	**68.52**	**78.95**	86.86	85.38
WebKB-project	806	**1335**	1366	**229**	858	7445
(10-fold AVG)	66.29	67.53	**53.24**	**43.28**	66.52	65.57
WebKB-student	1039	**2009**	**4937**	**262**	**1428**	7445
(10-fold AVG)	83.31	84.59	79.16	**72.43**	84.38	83.81
Average	722	**1430**	**2529**	517	**863**	8027
(Macro AVG)	81.46	82.39	**67.72**	**72.17**	81.98	81.35

TABLE I: Unlabeled data methods for stopping AL. For each dataset, the average number of annotations at the automatically determined stopping points and the average F-measure at the automatically determined stopping points are displayed. **Bold** entries are statistically significantly different than SP (and non-bold entries are not). The Average row is simply an unweighted macro-average over all the datasets. The final column (labeled "All") represents standard fully supervised passive learning with the entire set of training data.

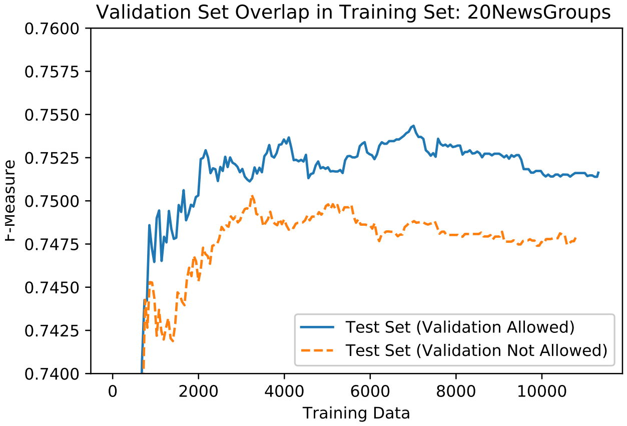


Fig. 1: Test set F-Measure when validation set examples are allowed to be selected for training versus when they are not allowed to be selected for training using a validation set size of 500. There is only one test set, however, two lines are shown because two separate sets of models were trained (one set that is allowed to select validation set examples for training and one set that is not). The dotted gold line stops exactly 500 (size of the validation set) examples earlier than the solid blue line because examples from the validation set were not available to train that set of models.

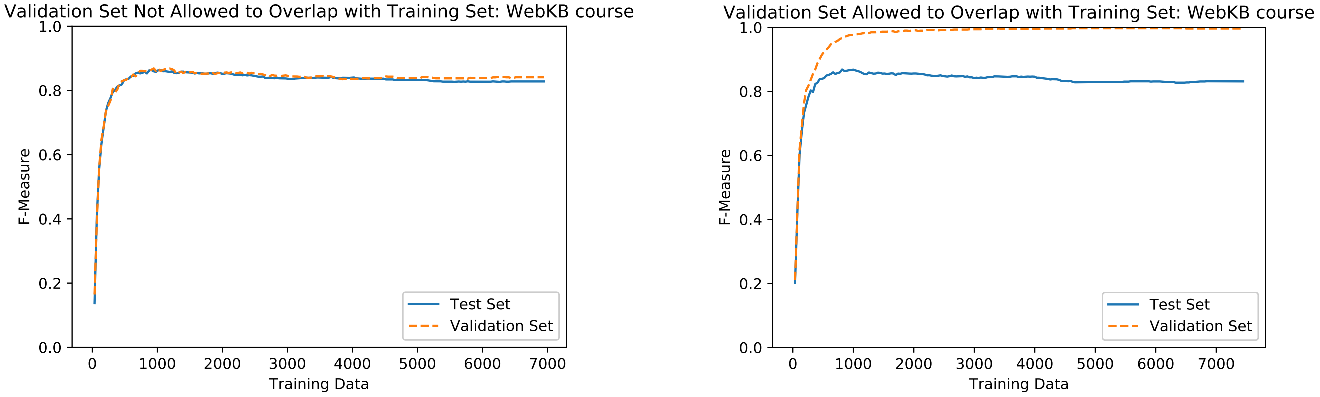
not perform well in our experiments anyway (e.g., see Table II) since it is tough to set the threshold value of τ. Therefore, the benefits of allowing validation set examples to be selected for training outweigh the drawbacks, and we allow examples from the validation set to be selected for training. Note that in all cases all final performance values in all of our experiments are computed using a completely held-out separate test set.

C. Size of Validation Set

The size of the validation set should be large enough to be representative, but small enough to be cost-efficient. To test the effect that size has on validation set stopping methods, we computed validation set performance using validation sets with sizes of 50, 100, 250, 500, and 1000. In Figure 3, we can see that performance estimates using validation set sizes of 50, 100, and 250 are erratic compared to performance estimates using sizes of 500 and 1000. The effect that this erratic behavior has on stopping methods can be seen in Figure 4, where stopping methods that use smaller validation set sizes perform poorly. From Figure 3 one can see that increasing the size of the validation set to be larger than 500 costs more labels, but does not improve performance estimates. In the rest of our experiments, we use a validation set size of 500.

D. Validation Set and Unlabeled Data Stopping Methods

Table II shows the performance of validation set stopping methods and unlabeled data stopping methods. In Table II we can see that validation set stopping methods tend to have worse performance than unlabeled data stopping methods. We can also see that SP $\wedge$ validation set stopping methods stop at a later iteration than unlabeled data stopping methods. This is expected because as mentioned in section II-D, SP $\wedge$ validation set stopping methods are more conservative and are guaranteed to stop later than or at the same point of SP. We can also see



(a) Test set and validation set F-Measure when validation set examples are not allowed to be selected for training using a validation set size of 500.

(b) Test set and validation set F-Measure when validation set examples are allowed to be selected for training using a validation set size of 500.

Fig. 2: Validation Set performance estimation curves when examples from the validation set are allowed to be selected as training data and when examples from the validation set are not allowed to be selected as training data.

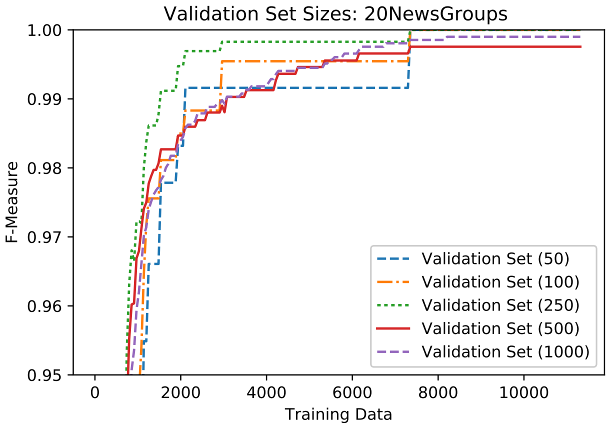


Fig. 3: Validation set F-Measure validation set sizes: 50, 100, 250, 500, 1000. Smaller validation set sizes are shown to be more erratic.

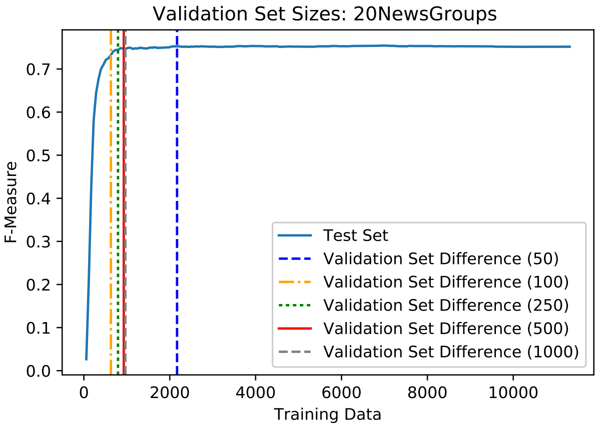


Fig. 4: Test set F-Measure for validation set stopping methods for multiple validation set sizes: 50, 100, 250, 500, 1000. Stopping methods (from left to right): 100, 250, 500, 1000, 50.

that SP ∨ validation set stopping methods stop earlier than or at about the same iteration than unlabeled data stopping methods. Once again, this is expected because SP ∨ validation set stopping methods are more aggressive and are guaranteed to stop earlier than or at the same point as SP.

Overall, unlabeled data stopping methods perform similarly or better than both validation set stopping methods and stopping methods that combine both the validation set and unlabeled data.

E. Training Set CV and Unlabeled Data Stopping Methods

Table III shows the performance of training set CV stopping methods and unlabeled data stopping methods. In Table III we can see that training set CV stopping methods tend to have worse performance than unlabeled data stopping methods. SP ∧ Training Set CV stopping methods stop more conservatively than unlabeled data stopping methods. On the other hand, SP ∨ Training Set CV stopping methods stop more aggressively than unlabeled data stopping methods.

Overall, unlabeled data stopping methods perform similarly or better than both training set CV stopping methods and methods that combine both the training set CV and unlabeled data.

VI. Conclusion

Active learning has the potential to significantly reduce annotation costs for text classification. One of the main considerations in the active learning process is when to stop the iterative process of asking for more labeled data. Previous work has researched stopping methods that use unlabeled data. Using labeled data in the form of a small labeled validation

Datasets	SP (BV2009/BG2013)	Threshold	Difference	SP ∧ Threshold	SP ∧ Difference	SP ∨ Threshold	SP ∨ Difference
20NewsGroups	846	**461**	929	846	957	**461**	817
(20-cat AVG)	74.92	70.29	74.94	74.92	75.16	70.29	74.70
Reuters	662	**355**	628	662	758	**355**	590
(10-cat AVG)	79.47	76.91	78.75	79.47	79.21	76.91	78.66
SpamAssassin-spam	291	**86**	270	291	299	**86**	264
(10-fold AVG)	98.70	**91.03**	98.26	98.70	98.63	**91.03**	98.33
WebKB-course	703	**273**	680	703	**780**	**273**	625
(10-fold AVG)	86.27	**79.89**	84.85	86.27	86.16	**79.89**	84.96
WebKB-faculty	736	**266**	703	736	802	**266**	677
(10-fold AVG)	86.42	**82.59**	86.08	86.42	86.73	**82.59**	85.94
WebKB-project	828	**562**	788	828	**917**	**562**	736
(10-fold AVG)	67.76	64.43	66.53	67.76	67.05	64.43	66.89
WebKB-student	1047	**373**	**817**	1047	1102	**373**	817
(10-fold AVG)	84.55	**79.18**	**82.08**	84.55	84.60	**79.18**	82.08
Average	730	**339**	688	730	802	**339**	647
(Macro AVG)	82.58	77.76	81.64	82.58	82.50	77.76	81.65

TABLE II: SP versus Validation Set Stopping Methods. For each dataset, the average number of annotations at the automatically determined stopping points and the average F-measure at the automatically determined stopping points are displayed. **Bold** entries are statistically significantly different than SP (and non-bold entries are not). The Average row is simply an unweighted macro-average over all the datasets. Performance Threshold has been renamed to "Threshold" and Performance Difference has been renamed to "Difference" to fit the table on the page.

Datasets	SP (BV2009/BG2013)	Threshold	Difference	SP ∧ Threshold	SP ∧ Difference	SP ∨ Threshold	SP ∨ Difference
20NewsGroups	823	864	**1459**	**2694**	**1615**	**316**	803
(20-cat AVG)	73.36	**60.95**	73.96	74.39	74.29	**60.66**	73.27
Reuters	691	**187**	859	734	964	**187**	600
(10-cat AVG)	77.94	**58.01**	77.22	77.97	78.68	**58.01**	76.68
SpamAssassin-spam	294	**89**	**753**	313	**753**	**89**	294
(10-fold AVG)	98.10	**91.58**	**98.78**	98.15	**98.78**	**91.58**	98.10
WebKB-course	669	**303**	**1139**	**1568**	**1139**	**199**	669
(10-fold AVG)	84.96	**70.61**	85.93	85.65	85.93	**70.37**	84.96
WebKB-faculty	728	**299**	**1354**	**1572**	**1417**	**214**	710
(10-fold AVG)	86.29	**73.68**	86.70	87.45	86.85	**73.56**	86.17
WebKB-project	806	**6000**	**1509**	**7445**	**1509**	673	806
(10-fold AVG)	66.29	61.42	67.19	65.57	67.19	61.18	66.29
WebKB-student	1039	1957	1361	**3167**	**1698**	**669**	950
(10-fold AVG)	83.31	78.14	83.60	84.11	84.51	**77.76**	83.03
Average	722	1386	**1205**	**2499**	**1299**	**335**	690
(Macro AVG)	81.46	**70.63**	81.91	81.90	82.32	**70.44**	81.22

TABLE III: SP versus Training Set CV Stopping Methods. For each dataset, the average number of annotations at the automatically determined stopping points and the average F-measure at the automatically determined stopping points are displayed. **Bold** entries are statistically significantly different than SP (and non-bold entries are not). The Average row is simply an unweighted macro-average over all the datasets. Performance Threshold has been renamed to "Threshold" and Performance Difference has been renamed to "Difference" to fit the table on the page.

set or using cross-validation on the training set as information sources for stopping methods was considered but not tested in previous work, leaving an open question of whether labeled data stopping methods would perform sufficiently better than unlabeled data stopping methods in order to justify any additional expenses associated with gathering the labeled data to inform the stopping method. We performed an investigation of stopping methods based on labeled data, unlabeled data, and combinations. We found that unlabeled data stopping methods are convincingly better than labeled data stopping methods. In our experiments, not only is the extra labeling cost not worthwhile, but stopping methods using unlabeled data perform better than stopping methods using labeled data.

Acknowledgment

This work was supported in part by The College of New Jersey Support of Scholarly Activities (SOSA) program and by The College of New Jersey Mentored Undergraduate Summer Experience (MUSE) program. The authors acknowledge use of the ELSA high performance computing cluster at The College of New Jersey for conducting the research reported in this paper. This cluster is funded by the National Science Foundation under grant number OAC-1828163.

References

[1] S. Hantke, Z. Zhang, and B. Schuller, "Towards intelligent crowdsourcing for audio data annotation: Integrating active learning in the real world," *Proc. Interspeech*, pp. 3951–3955, 2017.

[2] M. Bloodgood and C. Callison-Burch, "Bucking the trend: Large-scale cost-focused active learning for statistical machine translation," in *Proceedings of the 48th Annual Meeting of the Association*

for Computational Linguistics. Uppsala, Sweden: Association for Computational Linguistics, July 2010, pp. 854–864. [Online]. Available: http://www.aclweb.org/anthology/P10-1088

[3] S.-W. Lee, D. Zhang, M. Li, M. Zhou, and H.-C. Rim, "Translation model size reduction for hierarchical phrase-based statistical machine translation," in *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Jeju Island, Korea: Association for Computational Linguistics, July 2012, pp. 291–295. [Online]. Available: http://www.aclweb.org/anthology/P12-2057

[4] F. Mairesse, M. Gasic, F. Jurcicek, S. Keizer, B. Thomson, K. Yu, and S. Young, "Phrase-based statistical language generation using graphical models and active learning," in *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*. Uppsala, Sweden: Association for Computational Linguistics, July 2010, pp. 1552–1561. [Online]. Available: http://www.aclweb.org/anthology/P10-1157

[5] A. Miura, G. Neubig, M. Paul, and S. Nakamura, "Selecting syntactic, non-redundant segments in active learning for machine translation," in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. San Diego, California: Association for Computational Linguistics, June 2016, pp. 20–29. [Online]. Available: http://www.aclweb.org/anthology/N16-1003

[6] D. D. Lewis and W. A. Gale, "A sequential algorithm for training text classifiers," in *SIGIR '94: Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*. New York, NY, USA: Springer-Verlag New York, Inc., 1994, pp. 3–12.

[7] S. Tong and D. Koller, "Support vector machine active learning with applications to text classification," *Journal of Machine Learning Research (JMLR)*, vol. 2, pp. 45–66, 2001.

[8] A. Mishler, K. Wonus, W. Chambers, and M. Bloodgood, "Filtering tweets for social unrest," in *Proceedings of the 2017 IEEE 11th International Conference on Semantic Computing (ICSC)*. San Diego, CA, USA: IEEE, January 2017, pp. 17–23. [Online]. Available: https://doi.org/10.1109/ICSC.2017.75

[9] S. C. H. Hoi, R. Jin, and M. R. Lyu, "Large-scale text categorization by batch mode active learning," in *Proceedings of the 15th International Conference on World Wide Web*, ser. WWW '06. New York, NY, USA: ACM, 2006, pp. 633–642. [Online]. Available: http://doi.acm.org/10.1145/1135777.1135870

[10] M. Janik and K. J. Kochut, "Wikipedia in action: Ontological knowledge in text categorization," in *2008 IEEE International Conference on Semantic Computing*. IEEE, 2008, pp. 268–275.

[11] M. Allahyari, K. J. Kochut, and M. Janik, "Ontology-based text classification into dynamically defined topics," in *Semantic Computing (ICSC), 2014 IEEE International Conference on*. IEEE, 2014, pp. 273–278.

[12] M. Kanakaraj and R. M. R. Guddeti, "Performance analysis of ensemble methods on twitter sentiment analysis using nlp techniques," in *Semantic Computing (ICSC), 2015 IEEE International Conference on*. IEEE, 2015, pp. 169–170.

[13] M. Bloodgood and K. Vijay-Shanker, "A method for stopping active learning based on stabilizing predictions and the need for user-adjustable stopping," in *Proceedings of the Thirteenth Conference on Computational Natural Language Learning (CoNLL-2009)*. Boulder, Colorado: Association for Computational Linguistics, June 2009, pp. 39–47. [Online]. Available: http://www.aclweb.org/anthology/W09-1107

[14] M. Bloodgood and J. Grothendieck, "Analysis of stopping active learning based on stabilizing predictions," in *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*. Sofia, Bulgaria: Association for Computational Linguistics, August 2013, pp. 10–19. [Online]. Available: http://www.aclweb.org/anthology/W13-3502

[15] G. Schohn and D. Cohn, "Less is more: Active learning with support vector machines," in *Proceedings of the 17th International Conference on Machine Learning (ICML)*, 2000, pp. 839–846.

[16] J. Zhu, H. Wang, and E. Hovy, "Multi-criteria-based strategy to stop active learning for data annotation," in *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, Manchester, UK, August 2008, pp. 1129–1136. [Online]. Available: http://www.aclweb.org/anthology/C08-1142

[17] F. Laws and H. Schütze, "Stopping criteria for active learning of named entity recognition," in *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, Manchester, UK, August 2008, pp. 465–472. [Online]. Available: http://www.aclweb.org/anthology/C08-1059

[18] A. Vlachos, "A stopping criterion for active learning," *Computer Speech and Language*, vol. 22, no. 3, pp. 295–312, 2008.

[19] J. Zhu, H. Wang, and E. Hovy, "Learning a stopping criterion for active learning for word sense disambiguation and text classification," in *In Proceedings of the International Joint Conference on Natural Language Processing (IJCNLP)*, 2008, pp. 366–372.

[20] M. Altschuler and M. Bloodgood, "Stopping active learning based on predicted change of f measure for text classification," in *Proceedings of the 2019 IEEE 13th International Conference on Semantic Computing (ICSC)*. Newport Beach, CA, USA: IEEE, January 2019.

[21] G. Wiedemann, "Proportional classification revisited: Automatic content analysis of political manifestos using active learning," *Social Science Computer Review*, p. 0894439318758389, 2018.

[22] M. Li and I. K. Sethi, "Confidence-based active learning," *IEEE transactions on pattern analysis and machine intelligence*, vol. 28, no. 8, pp. 1251–1261, 2006.

[23] M. Bloodgood and K. Vijay-Shanker, "Taking into account the differences between actively and passively acquired data: The case of active learning with support vector machines for imbalanced datasets," in *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Companion Volume: Short Papers*. Boulder, Colorado: Association for Computational Linguistics, June 2009, pp. 137–140. [Online]. Available: http://www.aclweb.org/anthology/N/N09/N09-2035.pdf

[24] S. B. Kotsiantis, "Supervised machine learning: A review of classification techniques," in *Proceedings of the 2007 Conference on Emerging Artificial Intelligence Applications in Computer Engineering: Real Word AI Systems with Applications in eHealth, HCI, Information Retrieval and Pervasive Technologies*. Amsterdam, The Netherlands, The Netherlands: IOS Press, 2007, pp. 3–24. [Online]. Available: http://dl.acm.org/citation.cfm?id=1566770.1566773

[25] L. Prechelt, "Automatic early stopping using cross validation: quantifying the criteria," *Neural Networks*, vol. 11, no. 4, pp. 761–767, 1998.

[26] T. Domhan, J. T. Springenberg, and F. Hutter, "Speeding up automatic hyperparameter optimization of deep neural networks by extrapolation of learning curves." in *IJCAI*, vol. 15, 2015, pp. 3460–8.

[27] B. Liu, Y. Dai, X. Li, W. S. Lee, and P. S. Yu, "Building text classifiers using positive and unlabeled examples," in *Data Mining, 2003. ICDM 2003. Third IEEE International Conference on Data Mining*. IEEE, 2003, pp. 179–186.

[28] T. Joachims, "Text categorization with suport vector machines: Learning with many relevant features," in *ECML*, ser. Lecture Notes in Computer Science, C. Nedellec and C. Rouveirol, Eds., vol. 1398. Springer, 1998, pp. 137–142.

[29] S. Dumais, J. Platt, D. Heckerman, and M. Sahami, "Inductive learning algorithms and representations for text categorization," in *CIKM '98: Proceedings of the seventh international conference on Information and knowledge management*, Bethesda, Maryland, United States, 1998, pp. 148–155.

[30] A. McCallum and K. Nigam, "A comparison of event models for naive bayes text classification," in *Proceedings of AAAI-98, Workshop on Learning for Text Categorization*, 1998. [Online]. Available: http://www.kamalnigam.com/papers/multinomial-aaaiws98.pdf

[31] C. Campbell, N. Cristianini, and A. J. Smola, "Query learning with large margin classifiers," in *Proceedings of the 17th International Conference on Machine Learni ng (ICML)*, 2000, pp. 111–118.

[32] M. Bloodgood, "Support vector machine active learning algorithms with query-by-committee versus closest-to-hyperplane selection," in *Proceedings of the 2018 IEEE 12th International Conference on Semantic Computing (ICSC)*. Laguna Hills, CA, USA: IEEE, January 2018, pp. 148–155. [Online]. Available: https://doi.org/10.1109/ICSC.2018.00029

[33] G. Beatty, E. Kochis, and M. Bloodgood, "Impact of batch size on stopping active learning for text classification," in *Proceedings of the 2018 IEEE 12th International Conference on Semantic Computing (ICSC)*. Laguna Hills, CA, USA: IEEE, January 2018, pp. 306–307. [Online]. Available: https://doi.org/10.1109/ICSC.2018.00059