Active Learning with Labeled and Unlabeled Documents in Text Categorization

Saed Jalili
Tarbiat Modares University, Tehran, Iran
sjalili@modares.ac.ir

Parisa Kord Jamshidi
Tarbiat Modares University, Tehran, Iran
kordp@modares.ac.ir

Abstract

In many real-world learning problems, preparing labeled examples for training is very expensive. In this paper a method for designing a classifier has suggested. At first an initial naïve Bayesian classifier is built, with a few labeled examples. Then an active learning method called Uncertainty Sampling with new similarity idea for batch selection has been used to select more informative documents for learning. So that, required labeled examples is reduced noticeably. Then by applying EM method, a large number of unlabeled document are given to the classifier. The classifier estimates unknown labels and uses them for retraining itself. Finally a boosting committee is built based on the derived naive Bayesian. The final class prediction is made through committee voting. The applied boosting method has not examined on text categorization up to now. It improves recall parameter of the classifier.

Keywords: Naïve Bayesian, active learning, unlabeled documents, EM, selective sampling, boosting.

1. Introduction

The automated categorization of texts in to predefined categories has witnessed a booming interest in the last ten years, due to the increased availability of documents in digital form and ensuing need to organize them. In research community the dominant approach to this problem is based on machine learning techniques [1]. One key difficulty with text classification learning algorithms is that they require many hand labeled examples to learn accurately. And it needs the high expenditure of man power. So that in suggested method, active learning is used. An active learner may begin with a very small number of labeled examples, carefully select additional examples for which it requests labels, learn from the result of that request, and using its newly-gained knowledge carefully choose which examples to request next. In this way the active learner aims to reach high performance using as few labeled examples as possible [15]. For doing it, we use the uncertainty sampling within new similarity idea for batch selection of new training samples. This process can be repeated until intended accuracy achieved.
Then we add a very large number of unlabeled examples with EM method, that estimates unknown labels of documents automatically and use them for retraining the classifier [3].

Finally boosting method is applied and naïve Bayesian is used as weak learner. Boosting methods are one of the best methods in text classification, in addition in the boosting used feature removal is applied for making committee [4]. This kind of boosting has never been used in text categorization up to now.

In consider of important characteristic of text, having a large number features and noise, this method improves some effectiveness parameters. Experimental results has shown on reuters-21578 data collection. At the follow of this paper, in section 2 related works, in section 3 proposed classifier and details, in section 4 classifier evaluation, and finally conclusion and future works are explained.

2. Related works

Recently text categorization has been applied to a wide variety of practical applications: cataloging news articles classifying web pages in to symbolic ontology finding a persons home page, automatically learning the reading interests of users, automatically threading and filtering email by content and book recommendation [3].

An early and popular machine learning technique for text classification is naïve Bayesian. Its straightforward probabilistic nature has made it amenable to a variety of extensions. Leverage of class hierarchy can be provided through statistical shrinkage[6] or other more ad-hoc techniques. Another methods such as SVM, Neural Networks, rule base, KNN, Rocchio and various boosting techniques are applied in this field [1,2].

To date, no single technique has emerged as clearly better than the others, though some recent evidence suggests that KNN and SVM perform at least as well as other algorithms when there is a lot of labeled data for each class of interest [7]. Boosting methods even have been better than SVM [1].

Most studies in to text classification use the simple document representation of bags-of-words, tracking the number of times each word occurs in a document or even just or not it occurred. Some efforts to include semantic information have provided too.

Active learning provides the possibility of selecting more informative samples. So that the number of required labeled samples is reduced noticeably. There are two main types of active learning [8]. The first uses membership queries in which the learner constructs examples and asks a teacher to label them. While this approach provides computational advantages, it is not always possible to construct meaningful and informative unlabeled examples for training.

The second type of active learning is selective sampling in which the learner examines many available unlabeled examples and select only the most informative ones for learning. In [24], it is described a statistically optimal
solution to this problem. That method selects the training example that once labeled and added to the training documents, it is expected to result in the lowest error on future test examples. This method is optimal but is not efficient.

Lewis & Gale [25] presented Uncertainty sampling algorithm for choosing the example with the greatest uncertainty in predicted label. In [14] a similar method applied with SVM. Active learning with SVM is also used in [26]. One effective active learning method is query by committee[10]. In [8,9] QBC is used for part of speech tagging. In [11] a committee of winnow learners for text classification is used. In QBC examples selected for labeling that the committee of hypothesis disagree on the predicted label. In [12] QBC through EM is used in text categorization.

Recently some methods based on estimating error reduction are used [15,16]. In these methods any sample with any possible label is added to classifier and the sample that the sample that is expected to result in the lowest error on future classification is selected.

Using unlabeled documents within labeled ones is an old idea in the statistics community that at least as early as 1968 it was suggested [3].

Nigam in [3] has used EM with naïve Bayesian. Generalization of this method with classifiers that haven’t a probabilistic framework is suggested [17].

In TSVM classifier as a fact of matter, after building SVM, TSVM is built through adding unlabeled examples. The effectiveness of TSVM has shown in [19].

Finally, boosting method has a special place in text classification. In [20], Ada boost with C4.5 and nearest neighbor as weak learner is used. These methods have shown good effectiveness on OCR problem and machine learning data repository. In the BoostTexter system, two different boosting algorithms are tested using one level decision tree weak learner [21]. A boosting algorithm based on a committee of classifier sub committees that improves on the effectiveness and the efficiency of Ada boost-MH is presented in[22]. Ada boost is applied with Naïve bayesian and examined on TREC-7, TREC-8 data collections. Naïve bayesian as week learner of boosting is used in [4]. In this method, for making any member of committee a subset of attributes is selected. We apply this last method in this paper that has never been examined in text classification. In comparison with sleeping experts, naïve Bayesian, rocchio, ripper and prtfidf, the Ada Boost has been better [1].

3. Proposed classifier

At first initial classifier with naïve Bayesian and a few labeled examples is built Fig.1. At the second stage, learning process is continued with uncertainty sampling through new similarity idea for batch selection. In proposed method, in place of selecting one uncertain example, once a batch of uncertain examples is selected.
Each of uncertain samples is compared with others and different ones is labeled by an expert and added to classifier at once and the classifier is updated. In this way the selected samples not only are uncertain but also are different, so that the information dimensions of the problem is covered more quickly.

The similarity measure of the selected samples is calculated with cosine similarity.

At the third stage reminder of unlabeled documents are used for retraining the classifier. Their labels are estimated with EM method. The driven classifier from previous stages is the first member of the final boosting committee. Another members is made based on this classifier at forth stage.

3.1. Making initial classifier

Naïve Bayesian is one of the most commonly used learning methods that it has shown a reasonable efficiency. In this method the probability of each possible class \( c = \{c_1, ..., c_n\} \) for any document is calculated and the most probable class is selected as the target class[5].

\[
C_{\text{max}} = \arg \max_{c \in \mathcal{C}} P(c_j \mid d) \tag{1}
\]

with applying bayes rule:

\[
C_{\text{max}} = \arg \max_{c \in \mathcal{C}} \frac{p(c_j) \prod p(a_i \mid c_j)}{p(d)} \tag{2}
\]

\( p(d) \) is constant for all classes and it can be removed. \( d \) can be represented with attributes set, as result:

\[
C_{\text{max}} = \arg \max_{c \in \mathcal{C}} p(c_j), p(a_1, a_2, ..., a_n \mid c_j) \tag{3}
\]

with independence assumption:

\[
C_{\text{max}} = \arg \max_{c \in \mathcal{C}} p(c_j) \prod p(a_i \mid c_j) \tag{4}
\]

although this assumption is unrealistic , Naïve Bayesian has a good effectiveness. The probabilities in above equations is calculated in this manner:

\[
p(c_j) = \frac{\text{doc}_j}{\text{examples}} \tag{5}
\]

\[
p(a_k \mid c_j) = \frac{n_k + 1}{n + |\text{vocabulary}|} \tag{6}
\]

In equations 5,6 \( \text{doc}_j \) is a subset of training documents that belong to \( c_j \), \( \text{examples} \) is the total set of training documents, \( n_k \) is the frequency of each word \( a_k \) in all documents belonging to \( c_j \), \( n \) is the number of distinct words in related documents and vocabulary is the number of distinct words or attributes.

3.2. Updating naïve Bayesian

Naïve Bayesian will be used as the base of active learning at the next section. While updating the classifier is required frequently in active learning, using the classifiers that updating them needs retraining from scratch, is very inefficient.

Fortunately in classifiers such as naïve Bayesian, SVM and neural networks, training can be provided incrementally, so that adding a new example to the classifier is performed more efficiently.

While one example belonging to \( c_j \) is added to the classifier, firstly in Eq.5 the numerator of related class is
incremented by one and also total number of examples in the denominator is incremented.

Secondly the word frequencies of new document is added to the previous statistics, more exactly to \( n_i \) of the related class only. For efficiency of implementation the word counts can be hold in place of probabilities (dominator and numerator of Eq. 5,6) and at the classification phase required process will performed for calculating the probabilities.

3.3. Active learning

At this stage, informative examples is recognized by initial NB classifier (NB1) that had been built with a few examples. In uncertainty sampling [13,25], the classifier classifies any sample and the assigned probability is compared with a threshold, then closed ones are selected for labeling. This method can be implemented with stream base sampling in which any sample is classified and compared with the threshold and a decision about labeling it, is made.

Another way for implementing uncertainty sampling is pool base method in which all unlabeled samples is analyzed and the most uncertain one is selected for labeling. The drawback of this method is ignoring prior distribution. In [12] a way is proposed to interfere the prior distribution through QBC method, in that way in addition to disagreement of the committee, it measures the density weighted of that document calculated with a statistic divergence equation and the product of disagreement and density is used as a criterion for selecting one sample.

A new idea that proposed in this paper is that we select \( n \) uncertain samples in stream based manner, in other

Figure 1. Overall Diagram of proposed classifier[27]

Figure 2. Active learning with uncertainty batch sampling and similarity analysis[27]
words if predicted probability is closed to a threshold, the sample is added to the selected batch. Then in place of adding all members of batch a similarity matrix is performed in the following manner:

\[
\begin{bmatrix}
    \text{sim}(1,1) & \cdots & \text{sim}(1,n) \\
    \vdots & \ddots & \vdots \\
    \text{sim}(n,1) & \cdots & \text{sim}(n,n)
\end{bmatrix}
\]  

(7)

At first, the vectors of samples are normalized:

\[
d'_i = (w'_1, w'_2, \ldots, w'_n)
\]

(8)

\[
w'_i = \frac{w_i}{\sqrt{w'_1 \cdot w'_2 \cdots w'_n}}
\]

(9)

\[
d^i = (w'^1_i, w'^2_i, \ldots, w'^n_i)
\]

(10)

where \( d^i \) is normalized vector of \( d \). then the cosine similarity is used to measure the similarity between them.

\[
\cos(\alpha) = \frac{d_i \cdot d_z}{\|d_i\| \|d_z\|} = d_i \cdot d_z
\]

(11)

After deriving similarity between two pairs, the element that has most total similarity to another ones is selected. The elements which their similarity to selected one is smaller than a threshold, is removed. It means they aren’t good candidates to be added to the classifier due to their similar content to added ones. Exploring among remaining samples is continued until no one is remained. Active learning algorithm has been presented in Fig.2.

In this way available documents is investigated faster. In addition experimental results show that this method improves uncertainty sampling because uncertain samples that have more new information given to classifier at once and faster, and the classifier reaches to its maximum effectiveness quickly. Similarity process also is performed for a few documents (around 5 or 6), so that it has a very low time complexity.

- Inputs: sets \( D^l \) and \( D^u \) of labeled and unlabeled documents.
- Build initial classifier \( \mathcal{H} \), based only on the labeled documents \( D^l \).
- Loop while classifying the unlabeled documents \( D^u \) with the current classifier \( \mathcal{H} \) changes as measured by the class membership of the unlabeled documents \( U^* \).
  - E-step: Use the current classifier \( \mathcal{H} \) to evaluate classification scores for each unlabeled documents \( D^l \) and unlabeled documents \( D^u \) with labels obtained from \( U^* \).
  - M-step: rebuild the classifier \( \mathcal{H} \) based on labeled and unlabeled documents.
- Output: classifier \( \mathcal{H} \) for predicting class labels of unseen unlabeled documents

**Figure 3.** EM algorithm for using unlabeled documents[17]

### 3.4. Using unlabeled documents

At the end of second stage, NB2 is derived. This classifier is become to some extent rich from the training samples point of view. As mentioned in section two it is theoretically and empirically proven that using unlabeled data is almost useful. For doing it, EM method is applied. EM estimates the maximum probability of unknown labels of unlabeled documents and uses them within labeled documents in an iterative process until their labels converge to constant ones Fig.3.

This method is experimented within probabilistic and nonprobabilistic classifiers and it shows almost good results.
3.5. Making committee

A classification committee is determined with two characteristics: 1) selecting classifiers 2) combination of the votes.

Boosting committees have been shown as very effective methods in the literature [1]. In these methods committee members are built with only one learning algorithm called weak learner.

An important difference between these methods and other committee base algorithms is the dependence of committee members. It means they should been built serially and any one is based on the previous member error rate.

The method used in this paper is very different from the other boosting methods such as Ada boost and its extensions, because for making each member a subset of attributes is selected and used. The error rate of each classifier is used as criterion for performing new subset of attributes and building next classifier.

Leave-one-out cross validation is used for measuring the error of each classifier. In this method once a sample is removed from training set and classified with the derived classifier. This task is repeated for a subset of training documents. Then error count and error percent can be calculated.

This method not only calculates the error of classifier more accurately but it also is implemented with naïve Bayesian very efficiently. The algorithm of building naïve Bayesian committee is represented in Fig.4.

After the previous stages and by using unlabeled documents NB3 was derived. In this stage NB3 that is built based on all of attributes, is used as the first member of boosting committee $NB_{base}$. The error of $NB_{base}$ is calculated in the above mentioned manner. This error is used as a criterion for making another members.

A vector of probabilities $P$ is hold for selecting a subset of attributes and half of the attributes selected with sampling based on $P$.

There is no need to any other calculations after creating attribute subset because all of class probabilities and conditional probabilities are available from the first member of committee and determining the related subset is sufficient.

However each produced classifier is evaluated with leave-one-out cross validation and the probabilities of attributes $P$ is updated according to this error measure.

So that if effectiveness of the current classifier was better than the base classifier then the probabilities of selecting current subset of attributes is increased at next trail and they decreased otherwise.

As the result of this method we will have $T$ naïve Bayesian classifiers with different subsets of attributes.
NBC(Att,D_{training},T)
Input: Att: a set of attributes,
D_{training}: a set of training examples described using Att and classes.
T: the number of attributes N, as its default value.
Output: a naïve Bayesian committee.

- Build a naïve baysian classifier using Att and D_{training} called \( NB_{base} \).
- \( \varepsilon_{NB_{base}} = leave-one-out-evaluation(NB_{base}, D_{training}) \)
- Add \( NB_{base} \) in to committee as the first member which uses all attributes, that is, \( NB_0 = NB_{base} \).
- \( MaxT = 10 \times T \)
- Initialize \( P[a] = 0.5 \) for each attribute a in att
- \( l = 1 \), \( t = 1 \)
- While ( \( t < MaxT \))
  - \( At_{navec} \) = Sample attributes from att based on P
  - \( NB_{aux} \) = Build a naïve baysian classifier using \( At_{navec} \)
  - \( \alpha_i = (1 - \varepsilon_{NB_{aux}} + 1)/2 \)
  - \( \beta_i = \alpha_i / (1 - \alpha_i) \)
  - For each attribute a in \( At_{navec} \): \( P[a] = P[a] / \beta_i \)
  - Normalize \( P \) such that \( \sum_{a} P[a] = 0.5 \times N \)
  - if \( \beta_i < 1 \) then
    - \( NB_t \leftarrow NB_{aux} \)
    - \( t \leftarrow t + 1 \)
    - \( l \leftarrow l + 1 \)
  - \( T \leftarrow t + 1 \)
- Return the naïve Bayesian committee containing \( NB_t \).

Figure 4. The Naïve Bayesian classifier committee learning algorithm [4]

10*T trails is performed even no suitable classifier is derived. In this case the committee might have less than T members. \( NB_{base} \) is put as the first member of committee to avoid an empty committee. So that we have at least one classifier. Almost always committee has \( T+1 \) naïve Bayesian classifiers. When a document is classified, committee members generate the probability of each class membership and with some combination formula their votes is combined. For example, the generated probabilities can be added and the class with maximum total sum is selected as the target class.

4. Evaluation of proposed method

The proposed classifier was examined on Reuters-21578 data collection. Ten classes of this documents which have more document number than others were selected. 30 percent of documents per class were hold for test and 70 percent per class were used for training in various stages. The statistics of used documents are shown in first three columns of Table 1.

At first stage the classifier is built by 20 training documents, two documents per class. Recall and precision of initial classifier is shown in second two columns of Table 1.

Then 80 documents are selected by applying uncertainty and similarity based sampling through active learning and added to the classifier in some batches.

As it is clear from Fig. 5,6,7 selecting training documents with uncertainty sampling is grows faster than random selection of training documents. In Fig.7. documents are selected in batch size 5 based on uncertainty and then similarity processing is performed between this batch of documents. In this way less updates of classifier is required and time expenditure of active learning is decreased.

In other hand the similarity processing is performed on a very few number of documents that it is not a time consuming process. As it is clear from Fig.7. the
maximum effectiveness is higher and it is also achieved faster. This is due to not using similar documents for learning that they have not new information for learning.

After covering dimensions of the problem with active learning, the remaining unlabeled documents, around 6000 documents, is labeled with EM and added to the classifier. In forth column of Table1. effectiveness parameters after applying EM, are shown.

Finally by applying boosting method the recall parameter can reach to its maximum possible value (last column of Table1).

In many experiments performed in this work, using this boosting method almost always increases recall parameter. This is due to using leave-one-out cross validation for making committee members. Leave-one out cross validation has very closed relation to calculating 1-recall except its test set.

Various experiments were performed on pairs of 20-newsgroups data classes and same results were observed.

5. Conclusion and future works

Nowadays with increasing availability of documents in digital form, for easy and more useful access to them, it is required to organize them. To date no single technique has emerged as clearly better than the others, though some recent evidence suggests that KNN and SVMs perform at least as well as other algorithms when there is a lot of labeled data for each class of interest. Boosting methods are even better than SVMs.
Table 1. Results on Reuter-21578 data collection

<table>
<thead>
<tr>
<th>Category</th>
<th>trainNo</th>
<th>testNo</th>
<th>First stage (NB1)</th>
<th>Second stage (NB2)</th>
<th>third stage (NB3)</th>
<th>Forth stage (NB4)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>P</td>
<td>R</td>
<td>P</td>
<td>R</td>
<td>P</td>
<td>R</td>
</tr>
<tr>
<td>Acq</td>
<td>0.85</td>
<td>0.53</td>
<td>0.87</td>
<td>0.67</td>
<td>0.89</td>
<td>0.91</td>
</tr>
<tr>
<td>coffee</td>
<td>0.37</td>
<td>0.59</td>
<td>0.57</td>
<td>0.70</td>
<td>0.79</td>
<td>0.81</td>
</tr>
<tr>
<td>crude</td>
<td>0.52</td>
<td>0.78</td>
<td>0.66</td>
<td>0.67</td>
<td>0.83</td>
<td>0.67</td>
</tr>
<tr>
<td>earn</td>
<td>0.82</td>
<td>0.90</td>
<td>0.99</td>
<td>0.46</td>
<td>1.00</td>
<td>0.87</td>
</tr>
<tr>
<td>Grain</td>
<td>0.58</td>
<td>0.23</td>
<td>0.17</td>
<td>0.66</td>
<td>0.59</td>
<td>0.57</td>
</tr>
<tr>
<td>interest</td>
<td>0.23</td>
<td>0.44</td>
<td>0.48</td>
<td>0.42</td>
<td>0.57</td>
<td>0.47</td>
</tr>
<tr>
<td>money-fx</td>
<td>0.34</td>
<td>0.23</td>
<td>0.60</td>
<td>0.85</td>
<td>0.59</td>
<td>0.88</td>
</tr>
<tr>
<td>ship</td>
<td>0.35</td>
<td>0.43</td>
<td>0.31</td>
<td>0.73</td>
<td>0.48</td>
<td>0.70</td>
</tr>
<tr>
<td>Sugar</td>
<td>0.38</td>
<td>0.16</td>
<td>0.45</td>
<td>0.86</td>
<td>0.40</td>
<td>0.88</td>
</tr>
<tr>
<td>Trade</td>
<td>0.38</td>
<td>0.82</td>
<td>0.38</td>
<td>0.87</td>
<td>0.75</td>
<td>0.81</td>
</tr>
<tr>
<td>Average</td>
<td>0.47</td>
<td>0.51</td>
<td>0.56</td>
<td>0.70</td>
<td>0.69</td>
<td>0.76</td>
</tr>
</tbody>
</table>

Naïve Bayesian is also a popular technique due to its straightforward probabilistic nature and its easy updating.

While hand labeled documents are very expensive and providing them needs man power, it is better that in place of random selection of training examples they analyzed and informative ones are selected, therefore active learning is used here and the number of required examples is reduced noticeably. In the proposed method a batch of documents are selected by using uncertainty sampling then the documents of this batch compared with each other. The similar ones is not added to classifier but the different ones is added at once. So that the samples that have new information are labeled and the classifier improves more, with less labeled documents and less classifier updating. While a large number of unlabeled documents are available for free, they are used with EM method for retraining the classifier. Finally a boosting committee of bayesian classifiers is used that has improved recall. This kind of boosting is not examined on text categorization up to now.

Machine learning methods and applying them on text categorization has become a wide research field. Specially minimization of labeled examples and applying active learning methods has become an interest field for research that have many ambiguous problems for future works.

References


