Combining Lightly-Supervised Text Classification Models for Accurate Contextual Advertising

Yiping Jin
Dept. of Mathematics & Computer Science
Chulalongkorn university
Thailand 10300
Yiping.Ji@student.chula.ac.th

Dittaya Wanvarie
Dept. of Mathematics & Computer Science
Chulalongkorn university
Thailand 10300
Dittaya.W@chula.ac.th

Phu T. V. Le
Knorex Pte. Ltd.
2 Science Park Drive,
Singapore 118222
le_phu@knorex.com

Abstract

In this paper we propose a lightly-supervised framework to rapidly build text classifiers for contextual advertising. In contextual advertising, advertisers often want to target to a specific class of webpages most relevant to their product, which may not be covered by a pre-trained classifier. Moreover, the advertisers are only interested in the target class. Therefore, it is more suitable to model as a one-class classification problem, in contrast to traditional classification problems where disjoint classes are defined a priori.

We first apply two state-of-the-art lightly-supervised classification models, generalized expectation (GE) criteria (Druck et al., 2008) and multinomial naïve Bayes (MNB) with priors (Settles, 2011) to one-class classification where the user only provides a small list of labeled words for the target class. We fuse the two models together by using MNB to automatically enrich the constraints for GE training. We also explore ensemble method to further improve the accuracy. On a corpus of real-time bidding requests, the proposed model achieves the highest average F_1 of 0.69 and closes half of the gap between previous state-of-the-art lightly-supervised models to a fully-supervised MaxEnt model.

1 Introduction

Contextual advertising (or contextual targeting) is a technique to maximize the relevance of online advertisements by performing page analysis of the webpages. Contextual advertising is closely related to text classification problem, which is well-known to the NLP community. Underlying, the system classifies webpages into predefined Ads verticals (categories). Based on the classification result, the real-time bidding (RTB) system will decide whether to bid for a particular page to display their Ads. Successful contextual advertising leads to lower advertising cost and higher click-through and conversion rate (Chatterjee et al., 2003; Broder et al., 2007).

Contextual targeting differs from traditional text classification in three ways. Firstly, with thousands of vastly different products coming to market every day, there are large number of categories (usually thousands of categories). Secondly, the advertisers often want to customize an existing category or create new categories to target a set of more relevant webpages. I.e. the classes are not static but evolving. It is therefore unrealistic to ask the advertisers to provide labeled webpages for each new or modified category they want to target. Lastly, while the advertisers can provide prior knowledge for the class they want to target, they cannot accurately specify the irrelevant (or negative) category because it likely covers broad topics in the wild. However, to train a classifier, usually labeled instances for each class are required. Because of these constraints, prominent service providers such as Peer39 [1] and Grapeshot [2] build their contextual targeting systems mainly using hand-crafted keywords instead of learning based approaches.

In this work, we model contextual targeting as lightly-supervised one-class classification problem. The algorithm takes unlabelled documents [3] DOC_U and the user provided keywords S_c for the

[1] https://www.sizmek.com/peer39/

[2] https://www.grapeshot.com/

[3] We classify only based on the text in the webpages. Henceforth we use "document" to refer to "webpage", conforming to the terminology commonly used in NLP.

Proceedings of the The 8th International Joint Conference on Natural Language Processing, pages 545–554,
Taipei, Taiwan, November 27 – December 1, 2017

target class c as input and returns a classifier M_c that can classify documents belonging to class c. It is lightly-supervised because we do not use any document labels , but labeled keywords instead. It is one-class classification because the users need to provide labeled keywords only for the class they want to target, not the negative class.

We apply two state-of-the-art lightly-supervised classification models, generalized expectation (GE) criteria (Druck et al., 2008) and multinomial naïve Bayes (MNB) with priors (Settles, 2011) in the one-class classification setting. Inspired by the relative strength and weakness of the two models, we propose a novel approach to fuse them together where MNB is trained first. The salient words are read off from the posterior class-word distributions and automatically added to form additional constraints for GE. We also employ ensemble method to produce a final classifier that closed more than half of the gap between previous state-of-the-art lightly-supervised models to a fully-supervised MaxEnt model.

The contributions of this paper are:

1. Extended state-of-the-art semi-supervised classifiers for one-class classification problem.

2. Enriched expectation constraints for GE using a MNB model trained using EM algorithm.

3. Successfully employed ensemble method to produce a final classifier that closed more than half of the gap between previous state-of-the-art semi-supervised models to a fully-supervised MaxEnt model.

The paper is organized as follows: Section 2 presents previous works in two related fields: semi-supervised classification and one-class classification. We review the two previous state-of-the-art models we depend heavily on, generalized expectation (GE) criteria and multinomial naïve Bayes (MNB) with priors in Section 3 and 4, followed by applying them to one-class classification problem (Section 5). Section 6 describes two distinct approaches to combine these two models together. In section 7, we present our experimental results. Lastly, we present conclusions and suggest future directions.

2 Related Work

2.1 Semi-Supervised Text Classification

Supervised text classification methods were successfully applied to various tasks, such as sentiment analysis (Pang et al., 2002; Wang and Manning, 2012), information extraction (Jin et al., 2013) and stance recognition (Hasan and Ng, 2014). The main problem of supervised classification technique is that it requires sizeable set of labeled training documents for each predefined class.

Various semi-supervised classification methods have been proposed to address the lack of training documents. We are particularly interested in *lightly-supervised* methods that exploit prior knowledge (usually in the form of labeled words for each class) and eliminate the need of any labeled documents. Two main categories of approaches are often employed to exploit the word labels. The first one is to build an initial weak classifier to obtain soft labels of the documents and then apply expectation maximization (EM) algorithm (Liu et al., 2004; Schapire et al., 2002). More recently, there is growing interest in methods that incorporate labeled word features directly into the classification model, either as constraints in an objective function (Druck et al., 2008; Zhao et al., 2016) or as priors on model parameters (Settles, 2011; Lucas and Downey, 2013).

Liu at al. (2004) proposed to label a set of representative words for each class, which is used to extract a set of documents as the initial training set. EM algorithm was then applied to iteratively refine the labels of the documents and to improve the accuracy of the classifier. Schapire et al. (2002) used hand-crafted rules based on keywords to label documents, and modified AdaBoost to fit both the labeled training data and the soft-labeled data.

The methods above convert domain knowledge into labeled documents. An alternative approach is to use the domain knowledge to provide model constraints. Druck at al. (2008; 2011) proposed generalized expectation (GE) criteria, which use labeled words to constrain the model's predictions on unlabeled data. GE has been successfully applied on different tasks, such as text categorization (Druck et al., 2008), semantic tagging (Druck et al., 2009), information structure analysis of scientific documents (Guo et al., 2015) and language identification in mixed-language documents (King and Abney, 2013). Similarly, Zhao et al. (2016)

made use of word-level statistical constraints to preserve the class distribution on words, so that the classifier will not drift due to the extra (noisy) labels introduced by EM algorithm.

Labeled words can also provide priors for generative models. Settles (2011) extended multinomial naïve Bayes model to allow labels for words by increasing their Dirichlet prior. His method consists of three steps: firstly to estimate the initial parameters using only the priors; secondly to apply the induced classifier on unlabeled documents; lastly to re-estimate the model parameters using both labeled and probabilistically-labeled documents. Using an interactive approach to query document and word labels from the user, the system can achieve 90% of state-of-the-art performance after a few minutes of annotation.

Dermouche et al. (2013) also exploited prior knowledge in a multinomial naïve Bayes model. Instead of modifying the priors, their method artificially modifies the occurrences of the terms in the right and wrong class. This method uses the full set of document labels and a large sentiment lexicon consisting of around eight thousand terms, making it less suitable for the lightly-supervised setting for contextual targeting.

2.2 One-Class Classification

Another area related to our work is one-class classification problem (Moya and Hush, 1996), where no labeled negative instance is available. One-class SVM (Schölkopf et al., 2001) learns only from positive examples. The model classifies new instances as similar or different to the training set. Lee at al. (2003) observed that this approach is highly sensitive to the input representation and it did not perform well for text classification.

We highlight that one-class classification problem is not restricted to using only positive labeled instances. When positive and *unlabelled* data are available, a popular approach is to randomly assign the negative class to unlabelled instances in the beginning and iteratively refine the labels using EM-like algorithms (Yu et al., 2002; Liu et al., 2003; Li et al., 2009).

Our task lies in the intersection of semi-supervised classification and one-class classification, yet it differs from both tasks in principled ways. Aforementioned semi-supervised classification algorithms were applied on corpora with predefined (two or more) classes. When the irrelevant (or negative) class refers to "everything else", we cannot accurately provide prior knowledge for the irrelevant class using expert domain knowledge. On the other hand, previous one-class classification problems assumed the presence of labeled instances of the positive class, without which neither the similarity-based nor the EM-like algorithms can be applied directly.

3 Generalized Expectation (GE) Criteria

Generalized expectation (GE) criteria (Mann and McCallum, 2008) are constraint terms used to train discriminative linear models. GE provides a flexible framework for encoding prior knowledge to provide training signals for parameter estimation. When applied to text classification, constraint functions G_k are expressed as the reference distribution of the feature labels. [4] For example $puck \rightarrow \{baseball : 0.1, hockey : 0.9\}$ means 90% of documents containing the word "puck" should be labeled the class "hockey". Each constraint is translated into a term to add to the objective function to encourage parameters that yield predictions conforming to the reference distribution on unlabeled documents. Formally, the combined objective function can be written as:

$$C = -\sum_{k \in K} D(\hat{p}(y|x_k > 0)||\tilde{p}(y|x_k > 0)) - \Delta$$

where $\hat{p}(y|x_k > 0)$ is the reference distribution, $\tilde{p}(y|x_k > 0)$ is the empirical distribution and D is a distance function. Δ is shorthand for a zero-mean σ^2 -variance Gaussian prior on parameters.

GE does not require one-to-one correspondence between constraint functions G_k and model feature functions. The optimization problem for GE is always under-constrained, meaning the number of parameters to be estimated far exceeds the number of constraints the user provides. To make the optimization tractable, the model updates the gradient for an unlabeled feature j based on how often j co-occurs with a labeled feature k. [5] Hence Druck et al. (2009) commented that GE can also be interpreted as a bootstrapping method that estimates parameters using limited training signals.

[4] The features consist of word unigrams in both Druck et al. (2008) and in this paper.

[5] Please refer to Druck et al. (2008) for the derivation.

4 Multinomial Naïve Bayes (MNB) with Priors

Multinomial Naïve Bayes is a generative classification algorithm making a strong assumption that each word w_k occurs independent of each other when conditioned on the class label c_j. Hence,

$$\hat{y} = \underset{j \in \{1,\ldots,J\}}{\operatorname{argmax}} \; P(c_j) \prod_{k=1}^{n_d} P(w_k|c_j)$$

where $P(c_j)$ is the probability of class c_j and $P(w_k|c_j)$ is the probability of generating word w_k given class c_j. $P(w_k|c_j)$ is estimated using:

$$P(w_k|c_j) = \frac{m_{jk} + \sum_i P(c_j|x^{(i)}) f_k(x^{(i)})}{Z(f_k)}$$

where $f_k(x^{(i)})$ is the count of word w_k in the ith document in the training set and $Z(f_k)$ is a normalization constant summing over all words in the vocabulary. Typically, a uniform Laplacian prior is used (all m_{jk} have the same value 1). To incorporate the word labels, Settles (2011) increased the prior m_{jk} by α if word w_k is labeled with class c_j. To exploit unlabeled documents, Settles (2011) used an initial model estimated with only the priors to probabilistically label the unlabeled documents. The probabilistically labeled documents are combined with the labeled words and documents to estimate the parameters for the final model using a single-iteration EM algorithm.

5 Applying GE and MNB to One-Class Classification

In Druck et al. (2009) and Settles (2011), the authors ran their models using human labeled keywords for each predefined class. Analogous to one-class classification, where the system takes only positive (and unlabelled) documents as input, We now try to train GE and MNB with priors using only *user-provided* positive keywords and unlabelled documents as input. [6]

GE cannot cope with one-class classification by nature. If we only provide labeled words for the "+" class, the "+" label will be "propagated" to all the word features co-occurring with a labeled word. The trained classifier will predict "+" for all the unseen documents. In contrary, MNB with priors can be trained using only the "+" keywords. When we increase the prior for labeled words in class c_+, it decreases $P(w'_k|c_+)$ with respect to $P(w'_k|c_-)$ for an unlabeled feature w'_k (the normalization constant Z_+ is greater that Z_- due to the increased priors in the "+" class). This is desirable because if a document only contains random unlabeled words, the model will predict "-".

5.1 Labeling Words for the Target Class

Liu et al. (2004) observed that it is difficult for the user to come up with a set of representative words for each class independently because they usually can only provide a few words, which are insufficient to train an accurate classifier. Therefore, it is critical that the system can assist the user by suggesting a good set of candidate words to label instead of asking them to come up with all the words by themselves. We use a hybrid candidate word suggestion method that asks the user to input a seed keyword (in most cases, it is merely the class name they want to target) and the system will suggest other words that are closely related to it. We make use of word vectors (Mikolov et al., 2013), pointwise mutual information (PMI) (Church and Hanks, 1990) and Wikipedia.

Word vectors have been widely used to measure word similarities (Tang et al., 2014; Levy et al., 2015). We calculate cosine similarity using pretrained GloVe word vectors [7] (Pennington et al., 2014) to find similar words to the seed word. Word vectors can identify linguistically or semantically related words. E.g. "luxurious" and "lavish" are the nearest neighbours of "luxury".

We use PMI to mine the words that tend to co-occur with the seed word. For example, "resort", "BMW", "Gucci" all receive high PMI scores with "luxury" while none of them are near the word "luxury" in the word vector space.

Lastly, we automatically extract the keywords and keyphrases from the Wikipedia page of the seed word (if the page exists). This method is aimed to cover the technical terms that are confident indicators but are rarely observed in the corpus. E.g. "Somniloquy" is a synonym of "sleep-talking" but it is thirty time less frequent than the latter.

We show the top 50 candidate words suggested by each of the three methods to the user, who will

[6] We use positive/+ to denote the target class and negative/- to denote the irrelevant class.

[7] http://nlp.stanford.edu/projects/glove/

select relevant words to add to the keyword list. Based on user study reported in Settles (2011), labeling words takes on average 3.2 seconds. This suggests the total time for the user to complete labeling is within 10 minutes.

5.2 Special Treatment for GE

As mentioned above, GE also requires labeled words for the "-" class to train. However, it is impractical and time-consuming to ask the users to label a list of words irrelevant to the target class. We form the keyword list for the "-" class by simply performing a multinomial sampling from the vocabulary, where each word is weighted by its log count in the unlabeled corpus. This assumes that a randomly sampled word is unlikely to be a keyword for the target class. Additionally, we use L_2-based penalty instead of the default KL divergence because previous work showed that it is more robust to label noises (Druck, 2011).

The labeled words are translated into GE constraints using Schapire-distributions (Schapire et al., 2002). For the user labeled "+" keywords, We assign P_+ = 0.9 and P_- = 0.1. I.e. if "puck" is a labeled word for the class "hockey", the corresponding constraint will be $puck \rightarrow \{hockey : 0.9, others : 0.1\}$. For the "-" class, we randomly sample 20 times more keywords than the "+" class but use a more even distribution, where we set P_+ = 0.25 and P_- = 0.75. This is in the same spirit with *biased sparsity* in Wang et al. (2016), which says the word distribution of the targeted topic only focuses on a small number of representative words and the word distribution of non-targeted topics contain almost all possible words. We do not use too many negative keywords (constraints) because it significantly increases the computational cost to train GE.

We refer to this system as $GE/Random$ because it uses randomly sampled "-" keywords.

6 Combining GE and MNB

6.1 Using MNB to Enrich GE Constraints

Settles (2011) observed that MNB outperformed GE when the number of labeled words were between five to 20. When the number of labeled keywords increased (with more prior knowledge), GE usually performed better. This inspired us to fuse the two models together, in which MNB's role is to bootstrap more labeled words to train a more accurate GE classifier.

We first use the user labeled "+" words and unlabeled documents to train an MNB model. Then we obtain a list of salient words for the "+" class which are not already covered in the user labeled words. We use a simple rule that word w_k will be added to the set S_{MNB} if $\frac{P(w_k|c_+)}{P(w_k|c_-)} > 10$ (w_k appears 10 times more likely given the "+" class than given the "-" class) . [8] In this way, we exploit the strength of a generative model (MNB) to discover latent structure and topics from unlabeled data to augment a discriminative model (GE), which usually achieves higher accuracy for classification tasks.

Algorithm 1 summarizes the training procedure for the combined model GE/MNB. Line 2-4 are the procedure to train a MNB with priors model with EM algorithm. We obtain the list of salient words for the target class in line 5. All the words in S_{user}, S_{MNB} and S_{rand} are translated into GE constraints to train the final GE/MNB model.

Algorithm 1: Train GE/MNB Classifier

1 train classifier (S_{user},DOC_U);
  Input : User labeled features S_{user} and unlabeled corpus DOC_U
  Output: Trained GE/MNB classifier
2 $M_{MNB/Priors}$ = train(S_{user});
3 DOC_{prob} = $M_{MNB/Priors}$.classify(DOC_U);
4 $M_{MNB/Priors+EM_1}$ = train(S_{user}, DOC_{prob});
5 S_{MNB} = $M_{MNB/Priors+EM_1}$.getSalientWords();
6 S_{rand} = randomlySampleWords();
7 $M_{GE/MNB}$ = train([S_{user}, S_{MNB}, S_{rand}], DOC_U);
8 return $M_{GE/MNB}$;

6.2 Using Ensemble Approach

Another intuitive way to combine GE and MNB is to train the two classifiers independently and use ensemble approach (Dietterich, 2000) to combine the prediction results of the two classifiers. GE and MNB behave quite differently although they make use of the same labeled keywords. The variety they introduce is a critical condition for the success of ensemble approach (Dietterich, 2000).

We first group the aforementioned classifiers into two families based on the algorithm to train their final classifier. The GE family includes GE/Random and GE/MNB. The MNB family includes MNB/Priors and MNB/Priors+EM$_1$. We used a simple ensemble rule: the classifier ensemble outputs "+" if and only if at least one GE **and**

[8] We also tried to replace the randomly sampled "-" keywords with words that have high $\frac{P(w_k|c_-)}{P(w_k|c_+)}$. However, the result was worse in general.

one MNB classifier output "+". We denote this classifier ensemble as $GE_1 \wedge MNB_1$. We used this heuristic rule instead of stacking or other more sophisticated techniques because it does not require any labeled documents to train the ensemble.

7 Evaluation

7.1 Baselines and Datasets

We compared the proposed GE/MNB and $GE_1 \wedge MNB_1$ with various baselines. For GE, we experimented with GE/Random, which uses user provided keywords for the target class and randomly sampled negative keywords to train. For MNB with priors, we ran two configurations following Settles (2011), MNB/Priors+EM_1, the full model using EM algorithm and MNB/Priors, the initial model using only user labeled priors. We also compared with a keyword voting baseline and a fully-supervised MaxEnt model trained using labeled documents. The results of the MaxEnt model is for reference purpose, as our goal is not to beat a supervised model, but to improve from previous lightly-supervised models.

We use the GE implementation in the MALLET toolkit [9] and the implementation of MNB with priors provided by Settles (2011) [10], which also extends from MALLET. All the classifiers share the same standard preprocessing pipeline.

We made use of two datasets for evaluation. The first dataset is sampled from the actual real-time bidding (RTB) requests. The second one is the well-known 20 Newsgroups corpus (Lang, 1995). The results on the first corpus is more relevant and indicative while the results on the 20 Newsgroups allows to benchmark the models beyond the application of RTB.

7.2 Evaluations on RTB Dataset

We created the real-time bidding (RTB) dataset from a database of 30 million historical requests. We used open-source Boilerpipe library (Kohlschütter et al., 2010) to extract the textual content from the webpages and we obtained the category labels for the webpages from a leading Ad Exchange platform (in total 2,200 categories). [11]

[9] http://mallet.cs.umass.edu

[10] https://github.com/burrsettles/dualist

[11] The dataset is available at https://sites.google.com/site/jinyipingnus/research for future works to reproduce our results.

Class	Docs	+/- Ratio
Cold & Flu	1,363	1:50
Cancer	3,234	1:20
Diabetes	1,394	1:50
Sleep Disorder	2,592	1:25
Nutrition	22,176	1:3
Sampled "-" docs	68,626	

Table 1: Number of documents for each class and relevant/irrelevant class ratio.

We randomly selected five categories in the "Health" domain to carry out evaluation. For each experiment, the documents labeled with one of the selected categories were assigned to be the "+" class. We uniformly sampled from all other categories to use as the "-" class. We hide the document labels during training for all models except for MaxEnt.

Table 1 shows the number of documents for each class and the rounded +/- ratio. We can see that the +/- classes are very imbalanced, which is to be expected in real-world RTB requests. In practice, we cannot simply perform up- or down-sampling without the labels of the documents. Therefore, we did not try to modify the class ratio to make it more balanced. For each set of experiment, we used the same 9:1 split of training and testing set.

One author of this paper composed the labeled keywords with the assistance of the hybrid keyword suggestion method described in Section 5.1. The keywords for each class were composed independently. He was not allowed to add words not suggested by the system so that we can validate the utility of the keyword suggestion method. Table 2 shows the keywords for each category he composed. The labeling stopped when either he finished reviewing all the suggested words or the time reached 10 minutes.

Table 3 shows the P/R/F_1 scores of each classifier for each class. We did not use the accuracy measure because the ratio of +/- classes is strongly imbalanced. We used the same set of user labeled keywords for system 0-5. For GE and MNB with priors, we used the parameter setting proposed in the original papers (GE: Gaussian Prior=1; MNB: α=50).

We can make some interesting observations from the result. Firstly, the keyword voting approach lagged behind all of the lightly-supervised models. This shows that learning based approaches improved from a simple rule-based sys-

Cold & Flu	*cough, flu, throat, nasal, sinus, congestion, respiratory, sneezing, influenza, mucus, runny, stuffy, decongestant, phlegm, pandemic, epidemic, measles, typhoid, diphtheria, antihistamines*
Cancer	*cancer, tumor, chemotherapy, radiation, melanoma, leukemia, lymph, malignant, oncology, chemo, biopsy, oncologist, carcinoma, neoplasm, benign, colonoscopy, fibroid, invasive,lumpectomy, nonmelanoma, metastasis, palliative, adjuvant, neoadjuvant, polyp, smear, pathologist, prognosis, colposcopy*
Diab.	*diabetes, insulin, glucose, mellitus, diabetic, ketoacidosis, ketosis, dka, hyperosmolar, hyperglycemic, nonketotic, niddm, polydipsia, polyphagia, polyuria, glucagon, metformin*
Sleep Dis.	*sleep, asleep, awake, bedtime, sleepy, dream, snoring, snooze, nap, pillow, melatonin, circadian, apnoea, somnipathy, polysomnography, actigraphy, dyssomnias, parasomnias, apnea, sleepwalking, catathrenia, hypopnea, hypersomnia, narcolepsy, cataplexy, nocturia, enuresis, somniphobia*
Nutri.	*nutrition, protein, nutrients, livestrong, vitamin, intake, carbohydrates, fiber, myplate, minerals, carb, grain, metabolism, dietary, antioxidants, calcium, nutritional, nutritious, nutritionist*

Table 2: Labeled words for each class

tem using the same prior knowledge.

Secondly, GE/Random and MNB/Priors+EM$_1$ performed comparably, with GE/Random performing slightly better.

The combined GE/MNB classifier improved recall by 4% from GE/Random. This is mainly due to the additional keywords we obtained from the MNB model. On average, 53 new keywords were added for each classifier, doubling the number of the original user labeled words.

The average precision dropped slightly due to the decreased precision of "ColdFlu" and "Cancer" class. After error analysis, we identified that for these two classes, some common words frequently co-occurring with user labeled keywords were introduced by MNB as new constraints. E.g. for "ColdFlu": cold, congestion, swine; for "cancer": breast, prostate. Such words are not real indicators of the target class and likely cause the precision to drop. Table 4 shows the top 10 automatically added keywords for each class.

The performance of the classifier ensemble GE$_1 \wedge$ MNB$_1$ was impressive, improving a further 4% from GE/MNB and reaching macro average F_1 score of 0.69. The system achieved the highest or close to the highest F_1 for all the classes among the lightly-supervised models. Its F_1 score is only 3% lower than the fully-supervised MaxEnt model. This is particularly encouraging because the MaxEnt model was trained using many thousands of labeled training documents.

7.3 Evaluations on 20 Newsgroups Corpus

We also applied the systems on the well-known 20 Newsgroups corpus (Lang, 1995) to facilitate future comparisons. The corpus contains 20 different newsgroups having 1,000 documents each. We used the documents with file name ending with "0" for testing (roughly 10% of the corpus) and the rest for training.

Following Druck at al. (2008), we used mutual information of the candidate words with the oracle document labels to mine the keywords for each class. This simulates the scenario where a domain expert can suggest and label relevant keywords. We further removed the keywords that appear in more than two classes. While keywords that appear in many classes can be "balanced out" in multi-class classification setting. We find that including them will usually harm the performance for one-class classification. We used in total 262 keywords (on average 13.1 per class).

We ran 20 experiments using each newsgroup as the target class at a time. We differentiate from Druck at al. (2008) and Settles (2011) mainly in that for each experiment, we only take the keywords for the target class as input, but not the keywords for other classes. We made this decision in order to be consistent with the one-class classification problem. We can think of this experiment as more lightly-supervised than Druck at al. (2008) and Settles (2011). Table 5 shows the performance of various systems.

We were not surprised that GE performed worse that the MNB counterpart given the small list of labeled words. By bootstrapping labeled words using MNB, the GE/MNB model improved recall by 3% at the sacrifice of 2% precision, which is similar to the result on the RTB dataset.

The classifier ensemble GE$_1 \wedge$ MNB$_1$ still managed to achieve the highest precision and F$_1$ score (tie with MNB/Priors+EM$_1$), showing its robustness despite the lackluster performance of individual classifiers.

We also experimented with MaxEnt varying the amount of the training data. MaxEnt (γ=0.1) used 10% of the corpus (2,000 labelled documents) for training. Its average F$_1$ score is similar to the

System	Cold Flu	Cancer	Diabetes	Sleep Dis.	Nutrition	Macro Avg.
0: keyword voting	.44/.63/.52	.60/.59/.60	.65/.61/.63	.53/.65/.58	.64/.44/.52	.57/.59/.58
1: GE/Random	.56/.63/.59	**.78**/.56/.65	.77/.56/.65	.72/.65/.68	.78/.45/.57	**.72**/.57/.64
2: MNB/Priors +EM_1	**.59**/.56/.57	.66/.53/.59	.62/.56/.59	.72/.60/.66	.58/**.86**/**.67**	.63/.62/.63
3: MNB/Priors	.37/**.81**/.50	.51/**.76**/.61	.56/**.84**/.67	.33/**.79**/.47	.62/.70/.66	.48/**.78**/.59
4: GE/MNB	.50/.65/.57	.67/.67/.67	**.80**/.59/.68	.73/.62/.67	.80/.53/.63	.70/.61/.65
5: $GE_1 \wedge MNB_1$	.54/.72/**.62**	.71/.66/**.68**	.73/.77/**.75**	**.74**/.64/**.69**	**.84**/.55/.66	.71/.67/**.69**
MaxEnt	.75/.65/.70	.71/.66/.68	.70/.70/.70	.76/.71/.73	.83/.76/.79	.75/.69/.72

Table 3: Precision/Recall/F_1 scores on RTB dataset. The best score for each measure is bolded. The last line shows the performance of a fully-supervised MaxEnt model for reference purpose.

Cold & Flu	*lisinopril, colds, congestion, neti, vaporub, sinusitis, swine, nostril, throat, runny*
Cancer	*lymphoma, metastatic, colorectal, humira, cancerous, ovarian, prostate, hpv, metastases, xeloda*
Diab.	*hypoglycemia, diabetics, prediabetes, lisinopril, glycemic, hyperglycemia, ckd, pancreas, catspyjamas, retinopathy*
Sleep Dis.	*zaps, ryu, insomnia, cpap, urara, rem, naps, toranosuke, rls, lucid*
Nutri.	*carbs, ldl, whey, folate, amino, creatine, niacin, potassium, fats, antioxidant*

Table 4: Top 10 automatically added constraints in GE/MNB for each class

System	Macro Avg.
0: keyword voting	.62/.43/.50
1: GE/Random	.62/.50/.55
2: MNB/Priors+EM_1	.63/.64/**.63**
3: MNB/Priors	.39/**.69**/.50
4: GE/MNB	.60/.53/.56
5: $GE_1 \wedge MNB_1$	**.67**/.60/**.63**
MaxEnt ($\gamma = 0.1$)	.86/.42/.57
MaxEnt	.88/.72/.79

Table 5: Macro average Precision/Recall/F_1 scores for each classifier on 20 Newsgroups corpus.

lightly-supervised models. Based on the user experiments in Settles (2011), annotating documents takes 10.8s on average. Therefore the estimated time to annotate the training data for this model is 6 hours, roughly 25 times the time needed to label 262 keywords. The last line shows the result of the MaxEnt model using the full training set. Its average F_1 score is 16% higher than $GE_1 \wedge MNB_1$.

The larger gap is likely because the keywords used on the 20 Newsgroups Corpus are automatically extracted from the corpus, while the keywords used on the RTB dataset exploited external resources (pre-trained word vectors and Wikipedia) and they are curated by a human annotator. In the real-world scenario, the keywords will be composed by non-technical users, instead of researchers in NLP who are familiar with the algorithm. Therefore, we cannot make assumptions of the quality of the keywords the user composes. However, this further confirms the importance of a good keyword suggestion method to assist the user to compose high-quality keywords.

8 Conclusions and Future Works

This paper proposed a framework to build lightly-supervised one-class text classifiers by applying generalized expectation (GE) criteria (Druck et al., 2008) and multinomial naïve Bayes (MNB) with priors (Settles, 2011). The classification methods make use of user-labeled words for the target class as the form of supervision and do not require any labeled documents. Motivated by the relative strengths of the two models, we merged them by using MNB to enrich the set of GE constraints.

We further improved the classification accuracy by combining the output of two families of classifiers through ensemble method. This resulted in a classifier ensemble which achieved an

average of 0.69 F_1 score on a dataset of webpages from real-time bidding requests. It is 5% or 6% higher than previous state-of-the-art lightly-supervised models and 3% lower than a supervised MaxEnt model.

The framework has been deployed into an online advertising platform where advertisers can build customized classifiers to target their Ads to the most relevant webpages.

A direction for future work is to further improve the classification accuracy and to match the performance of not just a MaxEnt model, but a more recent fully-supervised deep neural network models such as Lai et al. (2015). This would likely require a more complex non-linear model and a novel method to train such model in a lightly-supervised manner.

In ongoing research, we are exploring to build multi-modal classifiers by exploiting information in the webpages beyond the textual content, such as URLs and images. We are also exploring transfer learning methods which can use the predictions for existing classes to improve the accuracy for new classes.

References

Andrei Broder, Marcus Fontoura, Vanja Josifovski, and Lance Riedel. 2007. A semantic approach to contextual advertising. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 559–566. ACM.

Patrali Chatterjee, Donna L Hoffman, and Thomas P Novak. 2003. Modeling the clickstream: Implications for web-based advertising efforts. *Marketing Science*, 22(4):520–541.

Kenneth Ward Church and Patrick Hanks. 1990. Word association norms, mutual information, and lexicography. *Computational linguistics*, 16(1):22–29.

Mohamed Dermouche, Leila Khouas, Julien Velcin, and Sabine Loudcher. 2013. Ami&eric: How to learn with naive bayes and prior knowledge: an application to sentiment analysis. *Atlanta, Georgia, USA*, page 364.

Thomas G. Dietterich. 2000. Ensemble Methods in Machine Learning. *Multiple Classifier Systems*, 1857:1–15.

Gregory Druck. 2011. *Generalized expectation criteria for lightly supervised learning*. Ph.D. thesis, University of Massachusetts Amherst.

Gregory Druck, Gideon Mann, and Andrew McCallum. 2008. Learning from labeled features using generalized expectation criteria. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 595–602. ACM.

Gregory Druck, Burr Settles, and Andrew McCallum. 2009. Active learning by labeling features. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1*, pages 81–90. Association for Computational Linguistics.

Yufan Guo, Roi Reichart, and Anna Korhonen. 2015. Unsupervised declarative knowledge induction for constraint-based learning of information structure in scientific documents. *Transactions of the Association for Computational Linguistics*, 3:131–143.

Kazi Saidul Hasan and Vincent Ng. 2014. Why are you taking this stance? identifying and classifying reasons in ideological debates. In *EMNLP*, pages 751–762.

Yiping Jin, Min-Yen Kan, Jun-Ping Ng, and Xiangnan He. 2013. Mining scientific terms and their definitions: A study of the acl anthology. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 780–790.

Ben King and Steven P Abney. 2013. Labeling the languages of words in mixed-language documents using weakly supervised methods. In *HLT-NAACL*, pages 1110–1119.

Christian Kohlschütter, Peter Fankhauser, and Wolfgang Nejdl. 2010. Boilerplate detection using shallow text features. In *Proceedings of the third ACM international conference on Web search and data mining*, pages 441–450. ACM.

Siwei Lai, Liheng Xu, Kang Liu, and Jun Zhao. 2015. Recurrent Convolutional Neural Networks for Text Classification. *Twenty-Ninth AAAI Conference on Artificial Intelligence*, pages 2267–2273.

Ken Lang. 1995. Newsweeder: Learning to filter netnews. In *Proceedings of the 12th international conference on machine learning*, pages 331–339.

Wee Sun Lee and Bing Liu. 2003. Learning with Positive and Unlabeled Examples Using Weighted Logistic Regression. *Algorithmic Learning Theory*, 348(1):71–85.

Omer Levy, Yoav Goldberg, and Ido Dagan. 2015. Improving distributional similarity with lessons learned from word embeddings. *Transactions of the Association for Computational Linguistics*, 3:211–225.

Xiaoli Li, S Yu Philip, Bing Liu, and See-Kiong Ng. 2009. Positive unlabeled learning for data stream classification. In *SDM*, volume 9, pages 257–268. SIAM.

Bing Liu, Yang Dai, Xiaoli Li, Wee Sun Lee, and Philip S Yu. 2003. Building text classifiers using positive and unlabeled examples. In *Data Mining, 2003. ICDM 2003. Third IEEE International Conference on*, pages 179–186. IEEE.

Bing Liu, Xiaoli Li, Wee Sun Lee, and Philip S Yu. 2004. Text classification by labeling words. In *AAAI*, volume 4, pages 425–430.

Michael Lucas and Doug Downey. 2013. Scaling semi-supervised naive bayes with feature marginals. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 343–351.

Gideon S Mann and Andrew McCallum. 2008. Generalized expectation criteria for semi-supervised learning of conditional random fields.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Mary M Moya and Don R Hush. 1996. Network constraints and multi-objective optimization for one-class classification. *Neural Networks*, 9(3):463–474.

Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, pages 79–86. Association for Computational Linguistics.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*, volume 14, pages 1532–1543.

Robert E Schapire, Marie Rochery, Mazin Rahim, and Narendra Gupta. 2002. Incorporating prior knowledge into boosting. In *ICML*, volume 2, pages 538–545.

Bernhard Schölkopf, John C Platt, John Shawe-Taylor, Alex J Smola, and Robert C Williamson. 2001. Estimating the support of a high-dimensional distribution. *Neural computation*, 13(7):1443–1471.

Burr Settles. 2011. Closing the loop: Fast, interactive semi-supervised annotation with queries on features and instances. In *Proceedings of the conference on empirical methods in natural language processing*, pages 1467–1478. Association for Computational Linguistics.

Duyu Tang, Furu Wei, Nan Yang, Ming Zhou, Ting Liu, and Bing Qin. 2014. Learning sentiment-specific word embedding for twitter sentiment classification. In *ACL (1)*, pages 1555–1565.

Shuai Wang, Zhiyuan Chen, Geli Fei, Bing Liu, and Sherry Emery. 2016. Targeted Topic Modeling for Focused Analysis. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '16*, pages 1235–1244.

Sida Wang and Christopher D Manning. 2012. Baselines and bigrams: Simple, good sentiment and topic classification. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers-Volume 2*, pages 90–94. Association for Computational Linguistics.

Hwanjo Yu, Jiawei Han, and Kevin Chen-Chuan Chang. 2002. Pebl: positive example based learning for web page classification using svm. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 239–248. ACM.

Li Zhao, Minlie Huang, Ziyu Yao, Rongwei Su, Yingying Jiang, and Xiaoyan Zhu. 2016. Semi-supervised multinomial naive bayes for text classification by leveraging word-level statistical constraint. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, pages 2877–2883. AAAI Press.