วิธีการเรียนรู้แบบมีผู้สอนเล็กน้อยสำหรับการจัดหมู่ข้อความแบบคลาสเดียว

นายอี้ปิง จิน

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาวิทยาการคอมพิวเตอร์และเทคโนโลยีสารสนเทศ ภาควิชาคณิตศาสตร์และ
วิทยาการคอมพิวเตอร์
คณะวิทยาศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2561

LIGHTLY-SUPERVISED LEARNING METHODS FOR ONE-CLASS TEXT CLASSIFICATION

Mr. Yiping Jin

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Science Program in Computer Science and
Information Technology
Department of Mathematics and Computer Science
Faculty of Science
Chulalongkorn University
Academic Year 2018

Thesis Title	LIGHTLY-SUPERVISED LEARNING METHODS FOR ONE-CLASS TEXT CLASSIFICATION
By	Mr. Yiping Jin
Field of Study	Computer Science and Information Technology
Thesis Advisor	Assistant Professor Dittaya Wanvarie, Ph.D.

Accepted by the Faculty of Science, Chulalongkorn University in Partial Fulfillment of the Requirements for the Master's Degree

Dean of the Faculty of Science

. . . . . .

(Professor Polkit Sangvanich, Ph.D.)

THESIS COMMITTEE

. . . . . Chairman

(Professor Chidchanok Lursinsap, Ph.D.)

. . . . . Thesis Advisor

(Assistant Professor Dittaya Wanvarie, Ph.D.)

. . . . . External Examiner

(Assistant Professor Kritsada Sriphaew, Ph.D.)

อี้ปิง จิน: วิธีการเรียนรู้แบบมีผู้สอนเล็กน้อยสำหรับการจัดหมู่ข้อความแบบคลาสเดียว. (LIGHTLY-SUPERVISED LEARNING METHODS FOR ONE-CLASS TEXT CLASSIFICATION) อ.ที่ปรึกษาวิทยานิพนธ์หลัก : ผศ. ดร. ฑิตยา หวานวารี, 50 หน้า.

วิทยานิพนธ์นี้นำเสนอวิธีการเรียนรู้แบบมีผู้สอนเล็กน้อยเพื่อสร้างตัวจำแนกข้อความโดยอาศัยการกำกับคลาสเพียงเล็กน้อย เราปรับใช้ตัวแบบการเรียนรู้แบบมีผู้สอนเล็กน้อยล่าสุดสองตัวแบบ ได้แก่ เกณฑ์การคาดหวังทั่วไป (generalized expectation criteria: GE criteria) (Druck et al. (2008)) และตัวจำแนกอเนกนามแบบเบส์อย่างง่าย (Multinomial Naive Bayes: MNB) โดยมีความรู้ก่อน (Settles (2011)) กับปัญหาการจำแนกคลาสเดียว ผู้ใช้เพียงต้องป้อนคำสำคัญของคลาสที่ต้องการเท่านั้น เราใช้วิธีทั้งสองที่กล่าวมาโดยให้ MNB ช่วยเพิ่มเติมรายการเงื่อนไขของ GE นอกจากนี้ เรายังรวมผลลัพธ์ของตัวจำแนกทั้งสองเพื่อเพิ่มความแม่นยำอีกด้วย

เราใช้ตัวแบบที่นำเสนอกับการโฆษณาออนไลน์ซึ่งเป็นปัญหาในโลกจริง ตัวแบบที่นำเสนอเมื่อใช้กับคลังข้อความโฆษณาออนไลน์มี F_1 เฉลี่ยรวม 0.69 ซึ่งเพิ่มขึ้น 50% จากความแตกต่างของตัวแบบเดิมที่มีผู้สอนเพียงเล็กน้อยกับตัวจำแนกแบบเอนโทรปีสูงสุด (MaxEnt) ซึ่งใช้ผู้สอนกำกับข้อความทั้งหมด

ภาควิชา	คณิตศาสตร์และวิทยาการคอมพิวเตอร์	ลายมือชื่อนิสิต	.....
สาขาวิชา	วิทยาการคอมพิวเตอร์และเทคโนโลยีสารสนเทศ	ลายมือชื่ออ.ที่ปรึกษาหลัก	.....
ปีการศึกษา	2561		

5972634023: MAJOR COMPUTER SCIENCE AND INFORMATION TECHNOLOGY
KEYWORDS: TEXT CLASSIFICATION / SEMI-SUPERVISED LEARNING METHODS / ONE-CLASS CLASSIFICATION

YIPING JIN : LIGHTLY-SUPERVISED LEARNING METHODS FOR ONE-CLASS TEXT CLASSIFICATION. ADVISOR : ASST. PROF. DITTAYA WANVARIE, Ph.D., 50 pp.

This thesis introduces a lightly-supervised learning method to train text classifiers with very little manual labelling effort. We adapt two previous state-of-the-art lightly-supervised models, generalized expectation (GE) criteria (Druck et al. (2008)) and multinomial naïve Bayes (MNB) with priors (Settles (2011)) to one-class classification problem. Users just need to label a handful of keywords for the target category. We also combine the two aforementioned models by letting MNB automatically augment the list of GE constraints. In addition, we ensemble two families of classifiers to improve the accuracy further. We successfully applied our model to a real-world problem of online advertising. On a corpus of online advertising data, the proposed model achieved the top macro average F_1 of 0.69 and closed 50% gap between previous state-of-the-art lightly-supervised models and a fully-supervised model MaxEnt model.

Department	: Mathematics and Computer Science	Student's Signature	.....
Field of Study	: Computer Science and Information Technology	Advisor's Signature	.....
Academic Year	: 2018		

Acknowledgements

Firstly, I want to express my gratitude to my thesis advisor Asst. Prof. Dr. Dittaya Wanvarie, for her guidance, suggestions, and support.

I appreciate Prof. Dr. Chidchanok Lursinsap, Asst. Prof. Dr. Kritsada Sriphaew, for being my thesis committee and for their useful comments to improve my work.

I greatly appreciate my reporting officer, Phu Le, head of R&D Department at Knorex, who offerd me full support to pursue M.S. study at Chulalongkorn University. I learned a lot from him not only technical experience, but more importantly leadership and inter-personal skills.

I am grateful to my parents for their endless support and understanding. The decision to move to Thailand and to pursue a Masters' study has not been easy. But my parents are always supportive and respect my decision.

Last but not least, I want to thank Department of Mathematics and Computer Science for the excellent program. During my study, I enjoyed the classes and the fruitful discussions with professors and fellow students.

CONTENTS

Page

Page

LIST OF TABLES

Table **Page**

LIST OF FIGURES

Figure **Page**

Chapter I

INTRODUCTION

Text classification is gaining more attention from the research community because of the increasing availability electronic documents from various sources (Aggarwal and Zhai (2012)). Over 80% of the online information is stored as text (Korde and Mahender (2012)), making text classification, and more generally text mining one of the most important and ubiquitous tasks in natural language processing.

Some of the sample applications of text classification are to filter news articles (Lang (1995)); to organise textual documents for easy retrieval and navigation (Chakrabarti et al. (1997)), which is especially applicable to user-generated content and social feeds. Text classification has also been widely used to mine opinion and sentiment (Liu and Zhang (2012)) as well as to classify emails (Carvalho and Cohen (2005)) and to filter spams (Sahami et al. (1998)).

A wide array of methods have been employed to build text classifiers, including but not limited to decision trees (Li and Jain (1998); Weiss et al. (1999)), rule-based approaches (Apté et al. (1994); Johnson et al. (2003)), Naive Bayes classifiers (McCallum et al. (1998)), support vector machines (Zhang and Yang (2003)) as well as neural networks (Kim (2014)). Most of these approaches require either extensive expert knowledge encoded as decision rules or large labelled corpus.

In the case where the set of interested categories/topics are constantly changing, such as social feeds, collecting and labelling a large set of documents for each category will become cost-ineffective or even infeasible. On the other hand, machine learning algorithms usually assume the presence of two or more classes. When users build a classifier for a new class, they typically do not bother with the "irrelevant"' class.

Due to these limitations, we propose a new lightly-supervised one-class classification framework to address the text classification problem. The inputs of this framework are 1) unlabelled documents DOC_U and 2) a handful of user-provided keywords S_c for the target class c. The output is a classifier M_c which can classify documents of class c. Figure 1.1 depicts the flow of the framework. The model belongs to lightly-supervised methods because no labelled document is needed, but only a handful of labelled keywords. the fact that users need to label keywords just for a single class makes it an instance of one-class classification problems.

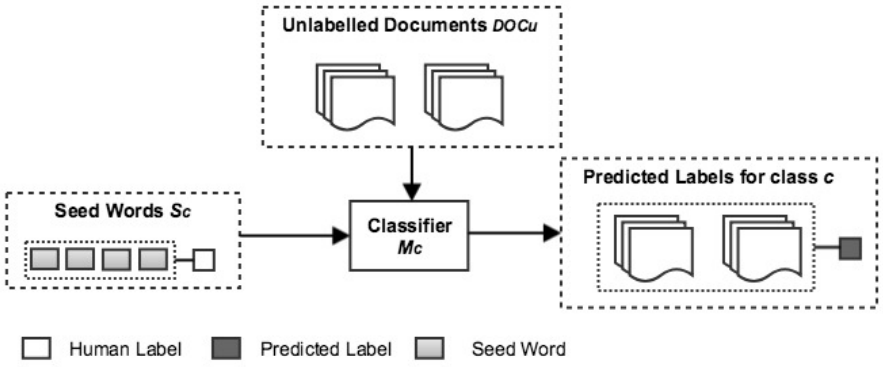

Figure 1.1: Lightly-supervised one-class classification framework

We adapt two previous state-of-the-art lightly-supervised learning methods, generalized expectation (GE) criteria (Druck et al. (2008)) and multinomial naïve Bayes (MNB) with priors (Settles (2011)) to one-class classification. After observing the characteristics of the two models, we propose a novel approach to combine them where we train an MNB model first, then read off the salient features from the posterior class-word distribution to automatically augment the set of GE constraints. We also apply ensemble approach to produce a final classifier which closed more than 50% gap between previous state-of-the-art lightly-supervised models and a MaxEnt model on a corpus of online documents.

1.1 Design Goals for a Lightly-Supervised One-Class Classification Algorithm

Our overall goal is to reduce the user input to the minimum while ensuring similar classification accuracy which will otherwise require thousands of labelled documents to achieve. More specifically:

- we avoid using any labelled documents because labelling document in general takes much longer time than labelling individual words (Settles (2011)).
- the user-input word list is not a complete lexicon of the class. Instead, we use the user-input words as seeds to bootstrap domain knowledge implicitly and explicitly.
- we do not require the user to input or even care about the keywords for the irrelevant class.

1.2 Scope and Assumption

Below is the scope of this thesis:

- This thesis considers classifying text documents belonging to a single class (*a.k.a* not multi-label classification problem). while we can treat individual labels separately and aggregate the result, we do not consider the correlation among labels and method to rank the list of labels for a document.
- The proposed algorithm does not rely on external knowledge base or pre-trained classifiers. While they might provide further improvement to the accuracy, we cannot assume the existence of such resources for new segments and new languages.

Additionally, in this dissertation, we assume the following:

- This dissertation assumes the user-input keywords are sensible, representing the semantics of the target class and not overly general.
- We have access to a sizeable unlabelled corpus (tens of millions of documents) containing at least a few hundred documents belonging to any target class (but we do not know their label).

1.3 Summary of Contributions

The central contribution of this thesis is proposing a framework to model text classification as lightly-supervised one-class classification problem. We extended state-of-the-art lightly-supervised classifiers to one-class classification. We also enriched expectation constraints for GE using an MNB model trained using EM algorithm. Our model achieved competitive result on a well-known dataset for text classification. Furthermore, we applied our model to a real-world problem of contextual advertising and showed very promising results.

1.4 Thesis Structure

The rest of the thesis is organised in the following manner. We firstly review previous works in two related fields: semi-supervised and one-class classification, followed by a more detailed illustration of two previous state-of-the-art classification models which we will build on top of in Section 2.3 and 2.4: generalized expectation (GE) criteria and multinomial naïve Bayes (MNB) with priors. We then introduce our main model in chapter 3. In chapter 4 we introduce the contextual advertising problem and how our proposed model addresses the challenges in contextual advertising seamlessly. In chapter 5, we show the experimental results on two corpora. One is a popular corpus for text classification, another is a corpus we sampled from actual real-time-bidding traffic for online advertising. Lastly, we conclude the thesis and suggest areas for future works.

Chapter II

RELATED WORKS

2.1 Semi-Supervised Classification

In this section, we review the approaches for semi-supervised learning methods, with the emphasise on the works applying on textual data.

Supervised classification models achieved impressive results on different tasks. such as information extraction (Jin et al. (2013)), sentiment analysis (Pang et al. (2002); Wang and Manning (2012)) and stance recognition (Hasan and Ng (2014)). The major issue about supervised classification techniques is that they require sizeable labelled training data for each predefined class.

In contrast, semi-supervised learning requires fewer or no labelled training instances and are usually faster to build (Druck (2011)). We are especially inspired by a particular type of semi-supervised learning, *lightly-supervised* learning methods, which use prior knowledge in the form of labelled keywords for each category and do not require any labelled documents at all. [1] In general, there are two main types of approaches are to make use of the labels for words. The first group of models builds an initial classifier trained using a small set of labelled documents or labelled features. The initial classifier is then used to predict the probabilistic labels of the unlabelled documents. Applying expectation maximisation (EM) algorithm on top of the soft labels (Liu et al. (2004); Schapire et al. (2002)) usually improve the accuracy further. In recent years, researchers began to explore methods which incorporate labelled features in the model learning itself without having to build a separate initial classifier. Such methods encode the labelled features either as additional constraint terms in the cost function (Druck et al. (2008); Zhao et al. (2016)) or directly as priors on model parameters (Settles (2011); Lucas and

[1] In this thesis, we will use the terms "word", "keyword" and "feature" interchangeably because the models we use all use individual words as features.

Downey (2013)).

Liu et al. (2004) proposed a classic approach tapping on EM algorithm to perform semi-supervised learning. Many following works shared the same spirit with this paper. In their work, Liu et al. (2004) first labelled a list of keywords for each category. The keywords were used to extract a set of most confident documents to form the initial training dataset. In the E-step of EM algorithm, they use the existing model to predict the labels of the unlabelled documents. In the M-step, the latest soft-labelled documents are used to reestimate the model parameter. Similarly, Schapire et al. (2002) used hand-crafted keyword rules to soft-label documents, and they modified AdaBoost algorithm to fit the model to both the labelled and the soft-labelled data.

The aforementioned methods requires building an initial model to obtain document labels. Then, traditional supervised or semi-supervised methods are applied to learn the final classification model. An alternative method is to use labelled features to constrain model parameters. Graça et al. (2007) regularised the expectations during EM algorithm with rich constraints on the posteriors of latent variables. They applied their model to the word alignment task. Druck et al. (2008) introduced generalized expectation (GE) criteria, which are additional terms added to the cost function to constrain the predictions on unlabelled instance. GE was successfully applied to various tasks, including semantic tagging (Druck et al. (2009)), text categorisation (Druck et al. (2008); Druck (2011)), structural analysis of research articles (Guo et al. (2015)) and identifying language of words in mixed-language documents (King and Abney (2013)). Zhao et al. (2016) also used word-level statistical constraints to ensure the classifier will not diverge too far from the original word-class distribution because of the noisy labels the EM algorithm generates.

Labelled keywords can also be used to modify priors in a generative model. Settles (2011) modified multinomial naïve Bayes model to increase the Dirichlet priors of the labelled words. The author firstly estimates the initial parameters us-

ing only the priors; then applies the resultant classification model on unlabelled documents; lastly he re-estimates the final parameters with the probabilistically-labelled documents. Settles (2011) also proposed an interactive user interface to prompt the user for document and word labels. The system is able to match 90% of state-of-the-art classification accuracy with only a few minutes of annotation.

Similarly, Dermouche et al. (2013) also built on a multinomial naïve Bayes model. However, they did not modify the priors as in Settles (2011). Instead, they artificially changed the count of term occurrence in the correct and wrong category. Unfortunately, their method requires labelled documents and a manually-curated sentiment dictionary with approximately 8,000 words, making it not applicable for lightly-supervised learning paradigm.

2.2 One-Class Classification

The other related area is one-class classification problem, which was initially proposed by Moya and Hush (1996). In one-class classification, we only have labelled instances belonging to a single class. Schölkopf et al. (2001) proposed One-class SVM. It internally builds kernels (boundaries) surrounding the positive instances. During prediction time, the model predicts whether a new instance belongs to the positive category based on their similarity or distance to the kernels. Despite being simple and elegant, this method is heavily affected by how the input instance is represented and its performance was not good for textual data (Lee and Liu (2003)).

We want to bring to the reader's attention that one-class classification does not mean we can only use instances belonging to a single class when training the model. *Unlabelled* data often provide additional insights and help to improve the decision boundary. Assuming that the majority of unlabelled documents are not instances of the positive category, we can randomly select some documents from the unlabelled data and assign them to the negative category. With the "pseudo-negative" documents, we can now train a normal binary classifier. Then we can

use EM algorithm to iteratively improve the classifier and to refine the labels. This type of approach was popularized in Yu et al. (2002), Liu et al. (2003) and Li et al. (2009).

One focus of research closely related to one-class classification is to reduce open space classification risk. When we train a classifier, it tries to generalise to the hyperspace where no or little training data are observed, which will likely cause false-positive classification. Scheirer et al. (2013) propose to reduce open space risk by bounding the positive region by two parallel hyperplanes. Shu et al. (2017), on the other hand, fits the positive instances with a half-Gaussian distribution and eliminates "outliers" lying below the threshold (three standard deviations below the mean).

Our task lies at the intersection of semi-supervised classification and one-class classification. However, we differ from either problem in important aspects. The semi-supervised classification approaches require that multiple categories are predefined and fixed. Users need to provide labelled features for each category. If we only have knowledge about one category, the models may not work. Furthermore, previous works in one-class classification make use of labelled examples (documents) belonging to the positive category. If we do not have any labelled examples, we cannot apply the similarity-based or the EM algorithm.

2.3 Generalized Expectation (GE) Criteria

Generalized expectation (GE) criteria (Mann and McCallum (2008)) are constraint terms added to the objective function of a MaxEnt model. GE allows us to flexibly encode prior knowledge and reduce or eliminate the requirement of labelled training instances. When applied to text classification, constraint functions G_k are the reference word-class distribution. To illustrate, $OSX \rightarrow \{Windows : 0.1, Mac : 0.9\}$ indicates that 90 per cent of documents where the word "OSX" occurs should be assigned the category "Mac" instead of "Windows". Each constraint is included as an additional term in the cost function to encourage parameter values that sat-

isfy most of the constraints. In practice, the reference probability distribution does not have to be precise. I.e. setting the probability of "Mac" to 0.9 or 0.8 does not make much difference. Therefore, we can usually use a fixed reference distribution instead of having to estimate the precise distribution for each labelled feature. Formally, the combined objective function to maximise is as following:

$$\mathcal{O} = -\sum_{k \in K} D(\hat{p}(y|x_k > 0)||\tilde{p}(y|x_k > 0)) - \Delta \tag{2.1}$$

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

In GE, the number of constraints specified by users is much smaller than the number of vocabulary. Therefore, we say that the optimisation problem is under-constrained. In order to learn the parameters of unlabelled word features, the training algorithm will firstly calculate the co-occurence matrix of each word in the vocabulary. It then updates the gradient of an unlabelled feature j based on whether it appears frequently together with a labelled feature k. [2] Therefore, we can interpret GE as a bootstrapping method that estimates parameter values based on a small number of user-input constraints.

2.4 Multinomial Naïve Bayes (MNB) with Priors

Multinomial Naïve Bayes (MNB) is one of the simplest classifiers. In fact, its training step consists of only simple counting. MNB model makes a strong assumption that word w_k occurs independently of each other conditioned on the class label c_j. This assumption is almost always violated, nevertheless, MNB model still achieves good empirical performance and is widely used as a competitive baseline model due to its simplicity, efficiency and interpretability (Wang and Manning (2012)). To define the MNB model formally,

[2] Please refer to Mann and McCallum (2008) for the derivation.

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

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 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 labelled for category c_j. He further exploited unlabelled documents by using an initial model estimated with only the priors to label the unlabelled documents probabilistically. The probabilistically labelled documents are combined with the labelled words to estimate the final model parameters using one-iteration EM algorithm.

Chapter III

LIGHTLY-SUPERVISED ONE-CLASS CLASSIFICATION MODEL

3.1 Applying Existing Models to One-Class Classification

In Section 2.3 and 2.4, we presented two previous state-of-the-art models: generalized expectation (GE) criteria and multinomial naïve Bayes (MNB) with priors. In previous works, the researchers used labelled keywords for each predefined category. To make the models applicable to one-class classification setting, we first try to adapt the two models so that we can train the classifier without the user having to input the keywords for the negative category.

Adapting GE to One-Class Classification

To recap, GE model's objective function is as follows:

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

The model cannot train a classifier successfully with labelled words from only one class. If all the labelled words x_k are from the positive class, the "+" label will be "propagated" to other word features which co-occur with the positive labelled words. This leads to most words in the vocabulary to have a positive weight. Therefore, no matter what the input document is, the trained classifier will always predict positive.

One straight-forward solution is to request users to provide a list of keywords which is irrelevant to the target category additionally. However, it is time-consuming and drastically degrades the user experience to build new classifiers. On the other hand, the user input keywords may be too specific or rare. In such

case, the GE model is not able to learn negative weights for the vast majority of the vocabulary because they do not co-occur with the rare user-labelled keywords.

We found a simple alternative to come up with the list of negative keywords. We can do a multinomial sampling of the full vocabulary. The assumption is that a random word sampled from the vocabulary is not likely to be related to the target category. The weight of each word is its log frequency. In this way, we are less likely to sample rare words which appear only a couple of times in the whole corpus. We will also not always sample the most frequent words because the log scale brings the difference between frequent and infrequent words much closer. Besides, we use the L_2 penalty instead of Kullback–Leibler divergence in equation 2.1 because Druck (2011) demonstrated its superior robustness to noisy feature labels, which is unavoidable if we randomly sample keywords to form constraints.

We translate the labelled keywords into constraints using the simple heuristic introduced by Schapire et al. (2002). For each user-labelled target keyword, we assign $P_+ = 0.9$ and $P_- = 0.1$. I.e. if "launchpad" is labelled for the category "Mac", it translates to the constraint $launchpad \rightarrow \{Mac : 0.9, others : 0.1\}$. For the negative category, we sample twenty times more keywords than the target category. However, we use a less skewed distribution, setting $P_+ = 0.25$ and $P_- = 0.75$. This is in inspired by *biased sparsity* in Wang et al. (2016), which says the word distribution of the target topic only focuses on a small number of representative words and the word distribution of irrelevant topics contain almost all possible words. We limit the number of negative keywords because adding too many constraints will significantly increase the training time of GE.

Adapting MNB to One-Class Classification

Compared to GE, it is trivial to adapt MNB with Priors model for one-class classification. We recap the formula of $P(w_k|c_j)$ below:

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

If we increase the prior m_{+k} for all labeled words w_k of the target category, it will increase $P(w_k|c_+)$, which will consequently cause $P(w'_k|c_+) < P(w'_k|c_-)$ for unlabelled words w'_k because the probability of all vocabulary sums up to 1 for both positive and negative category. This means if a document only contains some random unlabelled words, the model will assign a higher probability of the class c_- than the class c_+. Therefore we do not need to include negative priors in theory.

We also tried to randomly sample negative keywords and modify their priors similar to what we did for GE. However, it drastically lowered the accuracy of the classifier. We conjecture that MNB with priors model can learn the weights for the negative category relatively well with EM algorithm. EM algorithm is an iterative approach. It works well with a small set of accurate training signals. By including a large set of noisy training signals, it might lead the algorithm to diverge.

Assisting Users to Compose Keywords of the Target Category

Liu et al. (2004) discovered that users often have difficulty coming up with a good list of representative keywords independently. Unless they are the domain expert, they often can only come up with a few words, which might not be sufficient to train the classifier. Besides tapping on lightly-supervised methods to reduce the amount of training signals the users need to provide, we also propose a way to automatically suggest related words based on the seed word the user inputs. This will speed up the process to compose keywords drastically and generate more keywords with higher quality.

We apply a hybrid keyword suggestion method. The user inputs a seed keyword, which can be the category name. The system will suggest keywords related to the seed word. We tap on word embeddings (Mikolov et al. (2013)), pointwise mutual information (PMI) (Church and Hanks (1990)) and Wikipedia hyperlinks to

build the hybrid keyword suggestion model.

Word embeddings are used to represent each word in the vocabulary with a fixed low-dimensional vector. Using unsupervised training such as skip-gram or continuous bag of words, similar words will appear close to each other in the vector space. Word embeddings are widely used to measure word similarities (Tang et al. (2014); Levy et al. (2015)). We use the pre-trained GloVe word embeddings with the most frequent 100,000 words [1](Pennington et al. (2014)). To suggest similar keywords to the seed word, we calculate the cosine similarity of each word to the seed word and return the top words in descending order of the similarity. Word embeddings can suggest both linguistically and semantically related words. E.g. the closest words to "luxury" are "lavish" and "luxurious".

The motivation for using PMI besides word embeddings is that sometimes words that co-occur with each other may not be similar semantically or syntactically. E.g., the words "resort", "Gucci" and "BMW" all have high PMI scores with the word "luxury". However, they are not close to the word "luxury" in the word embedding space at all. To rank related words based on PMI, we make use of a large dataset of web pages which are not included in the training or testing dataset.

Lastly, we also extract the hyperlinks from the Wikipedia page whose title is the seed word. Since disambiguation and Wikification are two other challenging tasks, we do not consider them in this work. Instead, we just use the exact match with Wikipedia URL. We use a simple rule that we will include the hyperlinked keyword w_k if its corresponding page contains a link back to the page of the seed word.

Compared to general text on the Internet, Wikipedia is more technical and contains some academic terminologies. Such words appear less frequently but are reliable signals of the target category. For example, "Somniloquy" is a synonym of

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

"sleep-talking" but it is thirty times less frequent than the latter [2].

We display the top fifty keywords suggested by each of the above methods. The user will identify the words that are relevant and append them to the keyword list. Based on the estimation of a previous user study (Settles (2011)), it takes around 3.2 seconds to label a word. The total time needed for the user to label all the suggested words will be within 10 minutes.

3.2 Combining GE and MNB

The experiments of Settles (2011) showed that MNB performed better than GE when the number of labelled keywords is small (around ten labelled keywords). However, when the number of labelled keywords increases, GE usually gives higher accuracy. Motivated by this observation, we aim to tap on the strengths of the two models and produce a final classifier which has higher accuracy than either of the individual classifier.

MNB belongs to the family of generative classifiers while GE is an instance of discriminative classifiers. For classification problems, discriminative models usually give good generalisation performance when plentiful training data are available. However, generative models can be easily tap on semi-supervised learning paradigm where few training signals are available (Lasserre et al. (2006)).

We therefore propose to train an MNB model first, then extract the learned knowledge from the model to form additional training signals for GE. We describe the full model in detail below.

Firstly, we train an MNB model with user-labelled positive keywords and unlabelled documents. The parameters of the MNB model include the probability of each class $P(c)$ and the probability of generating each word from each class $P(w|c)$. Secondly, we extract a list of representative words for the positive class from the

[2] Based on the number of search results of Google.

trained model. We set a threshold that a word w_k will be extracted if $\frac{P(w_k|c_+)}{P(w_k|c_-)} > 10$ (w_k is 10 times more likely to appear in a positive document than in a negative document). We denote the set of automatically extracted positive keywords as S_{MNB}. Lastly, we combine the user labelled keywords, the additional keywords from the trained MNB model as well as the randomly sampled negative keywords to train the final GE model. We denote the model as GE/MNB.

This approach exploits a generative model (MNB) to learn the underlying topic from unlabelled documents. It provides another discriminative classifier (GE) additional training signals to help it achieve better final accuracy.

Algorithm 1 depicts the training procedure of the combined model GE/MNB.

Algorithm 1 Training of GE/MNB Model

INPUT: User labelled words S_{user} and unlabelled corpus DOC_U
OUTPUT: Trained GE/MNB classifier
$M_{MNB/Priors}$ = train(S_{user})
DOC_{prob} = $M_{MNB/Priors}$.classify(DOC_U)
$M_{MNB/Priors+EM_1}$ = train(S_{user}, DOC_{prob})
S_{MNB} = $M_{MNB/Priors+EM_1}$.getSalientWords()
S_{rand} = randomlySampleWords()
$M_{GE/MNB}$ = train([S_{user}, S_{MNB}, S_{rand}], DOC_U)
return $M_{GE/MNB}$

3.3 Applying Ensemble Method

Although GE and MNB use the same set of labelled keywords and they train on the same unlabelled corpus, the underlying learning methods are very different. This variety gives an opportunity to employ ensemble approach (Dietterich (2000)) further and combine the two families of classifiers. We firstly group the classifiers based on whether their final model is GE or MNB. The GE group consists of GE/ Random and GE/MNB, while the MNB group consists of MNB/Priors and MNB/ Priors+EM_1. We adopt a simple rule-based ensemble approach: we will label a document as positive if at least one classifier from each group predicts positive. We denote the classifier ensemble as $GE_1 \wedge MNB_1$. The prediction is logically equivalent to $(GE/Rand \vee GE/MNB) \wedge (MNB/Priors \vee MNB/Priors+EM_1)$. We use

a rule-based ensemble instead of the more sophisticated stacking approach because, in the lightly-supervised setting, we do not have labelled development set to tune the parameters of the classifier ensemble.

Chapter IV

APPLYING THE MODEL TO CONTEXTUAL ADVERTISING

4.1 Background of Online Advertising and Contextual Advertising

Traditionally, advertisers publicise for their products or services via channels such as billboards, newspapers or television ads. Such forms of advertising have several drawbacks. Firstly, it is not possible for the advertiser to control who will see their ads. The vast majority of people who see the ads may find it irrelevant or even annoying. Secondly, the traditional channels of advertising do not provide the possibility to track the return on investment (ROI). Advertisers cannot know how much additional sales are due to the effect of the advertisement versus seasonal sales fluctuation or the impact of a particular promotion. Lastly, traditional channels of ads are often costly, especially if the advertisers want to display their ads in a prominent spot. For example, a 30-second ad displayed in national channels costs around 123,000 USD, while the highest placement cost for Super Bowl Ads is beyond 4 million USD [1].

The ubiquity of the World Wide Web (WWW) and various portal browsing devices such as mobile phones and tablets give rise to a new form advertising. Online advertising addresses the drawbacks of traditional forms of advertising. Firstly, by analysing the content of the web page and the audience browsing history, advanced machine learning algorithms can display the ads to the right audience at the right moment (Yan et al. (2009)). Therefore, instead of blindly showing the ads to a random audience, online advertising can display ads only to a small group of targetted audience based on their age, gender, interest and browsing history. Secondly, advertisers usually embed tracking pixels in both the advertisements and their website.

[1] `https://fitsmallbusiness.com/tv-advertising/`. Accessed on 2018-04-12

It can observe the "user journey" before they purchase a product and attribute the purchase to one or more ad display in the past. With the help of ad attribution, advertisers can calculate the additional revenue due to the ad placement and the return on investment. Lastly, online advertising has a lower cost compared to traditional forms of advertising [2]. The cost per click for Google Ads ranges from $1 to $2. For social media like Facebook, the cost is even lower [3]. Therefore, small businesses can get started with online advertising with a low budget of $100 per month, which is impossible for traditional forms of advertising.

Contextual advertising is a specific online advertising strategy to display the ads to most relevant web pages. E.g. showing an ad of Amari Hotel [4] on a TripAdvisor page about travelling in Bangkok is likely more appropriate than showing it on a page about US financial news in New York Times. The importance of contextual advertising is two-fold. Firstly, successful contextual advertising will greatly enhance the user perception of the ads and increase the advertising revenue (Chatterjee et al. (2003); Broder et al. (2007)). Secondly, contextual targeting helps real-time bidding agents to drastically reduce the number of requests for down-stream processing. This is crucial to the stability of the bidding agents because it processes a huge amount of RTB traffic and can easily break when the traffic grows beyond its capacity.

To target to the most relevant web pages, we need to analyse the web page where we want to display our ads. Underlying, the system first extracts the textual content of the web page, then classifies the content into predefined categories using either rule-based keyword matching approach or machine-learning methods. The list of categories (content taxonomy) [5] is defined by Interactive Advertising Bureau (IAB) [6], the international governing organisation for digital advertising industry.

[2] https://en.wikipedia.org/wiki/Online_advertising#Benefits_of_online_advertising

[3] https://fitsmallbusiness.com/tv-advertising/. Accessed on 2018-04-12

[4] https://www.amari.com/

[5] https://github.com/InteractiveAdvertisingBureau/taxonomy

[6] https://www.iab.com

The content taxonomy defines two tiers of categories. The first tier consists of broad categories such as “education”, “technology” and “travel”. Tier 2 categories are more fine-grained, such as “early childhood education”, “artificial intelligence” and “Thailand travel”. The taxonomy covers around 30 tier 1 categories and 400 tier 2 categories. It gives advertisers the ease to choose the closest categories to their business and only target to web pages belonging to these categories.

4.2 Applying Lightly-Supervised One-Class Classification to Contextual Advertising

Contextual advertising differentiates itself from traditional text classification in some principled ways. Firstly, there is a large number of categories. Typical text classification benchmark datasets usually contain no more than a dozen of categories (Maas et al. (2011); Lang (1995)). However, we typically have thousands of categories in contextual advertising if we combine predefined and custom categories. Secondly, although the tier 2 predefined categories are relatively specific, they often do not suffice the needs of the advertisers. For example, if you are the owner of a hotel in Krabi, you might find the category “Thailand travel” too broad and prefer web pages specifically about travelling in Krabi. It is not possible to collect sizeable training data for all such fine-grained categories. In fact, even to come up with a complete list of all categories that might be useful for advertisers is infeasible. Therefore, there is a need for advertisers to quickly create a segment without too much manual effort. Lastly, while advertisers are interested in the category of web pages they want to target, we cannot ask them to specify what kind of web pages they do not want to display their ads. However, traditional classification problems usually require the presence of two or more predefined categories.

Based on the analysis above, we argue that contextual advertising is a perfect venue for us to apply the lightly-supervised one-class classification model proposed in this thesis. The model requires the advertisers to input only a handful of keywords related to the categories they want to target. It dramatically decreases the

time and effort needed compared to traditional supervised methods. Differing from rule-based keyword matching model, our model applies semi-supervised learning methods on unlabelled datasets and can potentially yield much better accuracy (we compare the performance of different models in the following chapter). Besides, one-class classification paradigm allows the advertisers to focus on the category they want to target without having to worry about the unspecified category of irrelevant web pages.

Figure 5.1 shows a complete contextual advertising system UI based on this thesis. User inputs keywords in the textbox at the top left corner. Bottom left corner displays the suggested keywords based on the algorithm in section 3.1. The real-time classification result (based on MNB with Priors model because it is much faster to train) is displayed in the "matched" box on the right to give the advertiser a feel of the segment. Once the final ensemble classifier has been trained, it is applied to the RTB traffic to classify incoming requests in the background.

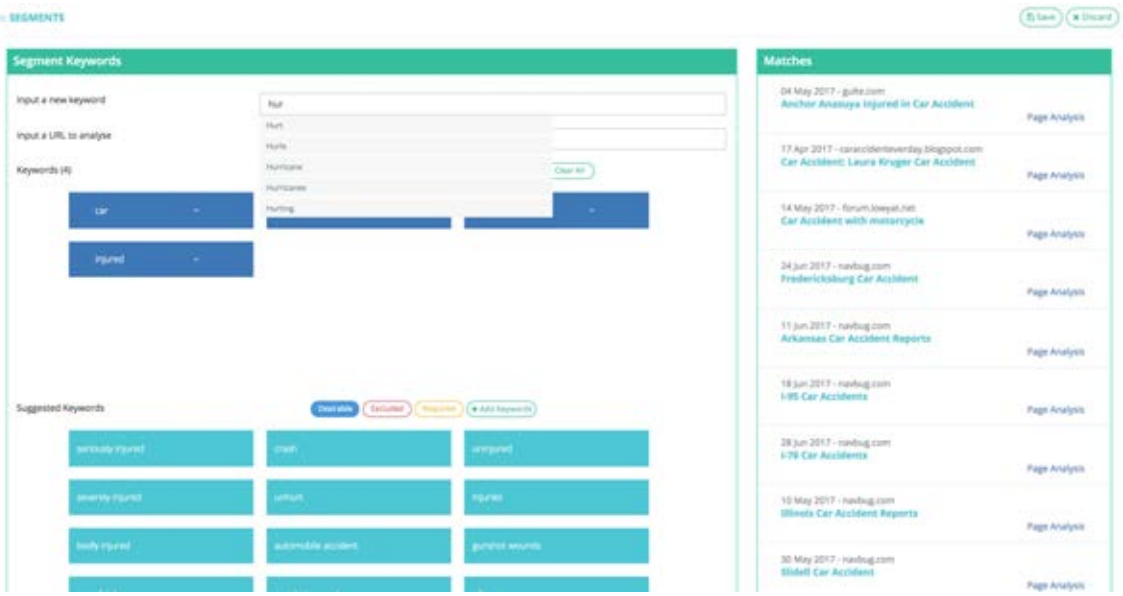

Figure 4.1: Contextual Advertising User Interface

Chapter V

EVALUATION RESULTS

To validate the effectiveness of our proposed models, we compared them against a bag of baseline models (described in section 5.1). We made use of two datasets, the first one, 20 Newsgroups dataset (Lang (1995)), is a widely used dataset to benchmark text classification algorithms. The second dataset, RTB dataset, is sampled from the real-time-bidding (RTB) traffic. It is used to evaluate the performance of our model for contextual advertising. We describe the two datasets in detail in section 5.2.

5.1 Baseline Systems

We compare our proposed model with previous state-of-the-art lightly-supervised models after we apply adaptations to make them work for the one-class classification setting. Specifically, we compare with the following models:

- *GE/Random*: GE model trained with user-input positive keywords and randomly sampled negative keywords.
- *MNB/Priors+EM*$_1$: the full model proposed by Settles (2011).
- *MNB/Priors*: Only increase the priors for the user-input keywords but without running EM algorithm. This is used as a competitive baseline in Settles (2011).

The two models below are the proposed models in this thesis:

- *GE/MNB*: build final GE model with additional constraints provided by a trained MNB model. Proposed in section 3.2.

- $GE_1 \wedge MNB_1$: the ensemble model proposed in section 3.3.

We include the result of a rule-based keyword voting baseline to confirm the hypothesise that semi-supervised learning methods will outperform a rule-based baseline.

Besides, we want to investigate how large the gap between state-of-the-art lightly-supervised models and a fully-supervised model is. To this end, we also compare the results with a fully-supervised MaxEnt model trained using labelled documents of each class.

GE implementation is available in the MALLET toolkit [1] and MNB with priors implementation is available at `https://github.com/burrsettles/dualist`. We built a shared preprocessing pipeline for all models so that the difference in performance is only due to the model.

5.2 Datasets and Evaluation Measure

Lang (1995) collected the 20 News dataset, and it has since become one of the most popular datasets to evaluate text classification algorithms. As its name suggests, the dataset consists of 20 different newsgroups. Each newsgroup contains roughly 1,000 documents. Some of the categories in the dataset are more related to each other, such as *comp.windows.x* and *comp.os.ms-windows.misc*. Some others are not related at all, such as *rec.sport.baseball* and *sci.electronics*. This makes the classification for each category having different levels of difficulty. Figure 5.1 shows the full list of categories in the dataset. For all the experiments, we used the documents whose filenames end with "0" (approximately 10 per cent of the dataset) as the evaluation set and the rest as the training set.

Unlike the 20 newsgroups dataset, the data for online advertising consist of heterogeneous web pages such as online forums, blogs, news and video pages.

[1] `http://mallet.cs.umass.edu`

comp.graphics comp.os.ms-windows.misc comp.sys.ibm.pc.hardware comp.sys.mac.hardware comp.windows.x	rec.autos rec.motorcycles rec.sport.baseball rec.sport.hockey	sci.crypt sci.electronics sci.med sci.space
misc.forsale	talk.politics.misc talk.politics.guns talk.politics.mideast	talk.religion.misc alt.atheism soc.religion.christian

Figure 5.1: Categories in 20 Newsgroups Dataset.

Therefore, we decide to use a dataset consists of actual online advertising traffic to reflect the accuracy of the model on the contextual advertising task. The dataset is provided by Knorex [2], a leading technology provider of online advertising. The company processes tens of thousands of real-time-bidding requests per second and possesses a database of hundreds of millions of unique URLs with their category labels. We applied the open-source Boilerpipe tool (Kohlschütter et al. (2010)) to preprocess the web pages by removing HTML tags and cleaning up the content of the web pages. The dataset contains in total 2,200 categories, which cover a wide variety of topics.

Out of the 2,200 categories, we arbitrarily chose five categories about "Health & Fitness" for evaluation. In each experiment run, one of the chosen categories was used as the positive category. We sampled from all other categories uniformly to form the negative category. For all the experiments, we use the same 0.9/0.1 train & test split as we did for the 20 Newsgroups corpus. The document labels are hidden in the training phase of all models except for the supervised MaxEnt model.

Table 5.1 shows some basic statistics of the RTB dataset. We can easily observe that the positive documents are much fewer than the negative documents for all categories except for "nutrition". This aligns with the actual traffic distribution of the real-world RTB requests because web pages related to a specific topic is always a small proportion of the general browsing traffic.

Because the categories are highly imbalanced, if we use accuracy (the number

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

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 5.1: Count of documents for each category and positive/negative class ratio.

of correct predictions divided by the number of total predictions), the result will be dominated by the accuracy of the majority class. Therefore, we choose to use macro precision/recall/F_1 scores as the evaluation metrics, which give equal weights to each category.

5.3 Evaluations on 20 Newsgroups Dataset

We firstly report the evaluation results on the 20 Newsgroups dataset. We followed Druck et al. (2008) and used mutual information to rank keywords for each category based on the oracle document labels. This simulated a human expert who can come up with meaningful keywords with the knowledge of the domain. Besides, we also removed all keywords which appear in two or more categories. While they might be important words describing the semantics of the category, they often do not help the classification. E.g. if we add the word "sport" to the keyword list for the class *rec.sport.hockey*, it might cause the documents related to other kinds of sports to be falsely labelled as *rec.sport.hockey*. After removing such keywords and keywords which do not have any meaning (likely due to the noise in the corpus), we were left with in total 262 keywords (on average 13 keywords per category). We show the full list of keywords in table 5.2.

We ran twenty independent experiments for each category in the corpus. The main difference between our experiments and Settles (2011) or Druck et al. (2008) is that within each experiment, we only use the keywords for the target category, but not the keywords for other categories. Also, we do not make the closed world

comp.graphics	*gif, image, images, graphics, format, plot*
comp.os.ms-windows.misc	*driver, windows, microsoft*
comp.sys.ibm.pc.hardware	*drive, bios, motherboard, controller, drives, bus, scsi, port, ide, isa*
comp.sys.mac.hardware	*centris, apple, quadra, mac*
comp.windows.x	*xterm, sunos, sun, xlib, window, server, application, widget, motif*
misc.forsale	*sell, offer, shipping, asking, selling, sale, condition*
rec.autos	*car, cars, dealer, ford, engine*
rec.motorcycles	*motorcycle, ama, ride, riding, bike, biker, dod, bmw, bikes*
rec.sport.baseball	*hitter, ball, pitchers, pitcher, pitching, won, hit, braves, baseball, sox, batting, mets*
rec.sport.hockey	*islanders, play, goal, coach, penguins, leafs, playoff, wings, played, goals, boston, goalie, cup, pens, bruins, hockey, puck, stanley, ice, devils, playoffs, pittsburgh, nhl, espn, rangers, detroit, scoring*
talk.politics.misc	*clayton, insurance, cramer, president, gay*
talk.politics.guns	*gun, fire, waco, deaths, compound, firearms, criminal, assault, amendment, weapons, atf, guns*
talk.politics.mideast	*serdar, argic, palestinians, ohanus, appressian, sahak, melkonian, killed, turkish, israel, villages, muslims, muslim, war, troops, extermination, attacks, countries, soviet, killing, population, genocide, peace, arab, israeli, forces, jew, closed, palestine, escape, soldiers, israelis, jewish, jews, territories, civilians, turkey, palestinian, russian, turks, jerusalem, armenians, armenian, army, mountain, arabs, roads, armenia, policy*
sci.crypt	*encrypted, encryption, clipper, communications, secret, des, agencies, security, wiretap, nsa, escrow, sternlight, key, cryptography, pgp, scheme, keys, phones, crypto, secure, privacy, chip, chips, enforcement, algorithm*
sci.electronics	*voltage, circuit, electronics, chip, electric*
sci.med	*skepticism, jxp, chastity, diet, banks, treatment, symptoms, studies, doctor, food, gordon, disease, medical, medicine, patients, heal, sick, cancer*
sci.space	*satellite, sky, earth, nasa, spencer, flight, moon, pat, launch, spacecraft, solar, shuttle, henry, orbit, mission*
talk.religion.misc	*morality, religion, sect, pagan, fundamentalists, liberal*
alt.atheism	*atheist, morality, jon, livesey, atheism, atheists*
soc.religion.christian	*christians, holy, church, eternal, faith, sin, scripture, sins, christianity, clh, heaven, spirit, spiritual, catholic, christ, love, doctrine, god's, homosexuality*

Table 5.2: Keywords for 20 Newsgroups Corpus labelled using mutual information.

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

assumption as mentioned in Shu et al. (2017), meaning that we do not assume the set of categories is known in advance and fixed. We argue that the one-class classification problem is harder than the multi-class classification problem in previous works. In multi-class classification, users can carefully pick keywords for all the categories to achieve the best result in distinguishing the classes. E.g. when classifing *rec.sport.hockey* versus *rec.sport.baseball*, if we observe the keyword "bat", the document is more likely to be related to *baseball* instead of *hockey*. However, if our task is to distinguish *rec.sport.hockey* versus *rest*, we can handpick only the keywords of the target class, but not the "open class".

The evaluation results on the 20 Newsgroups dataset is shown in table 5.3. The same set of user labelled words was used for system 0-5. We followed the parameter settings used in the original papers (GE: Gaussian Prior=1; MNB: α=50).

We observe from the result that most machine learning based models outperformed the rule-based keyword voting approach by a large margin, validating the hypothesise that semi-supervised learning on top of user-input domain knowledge does help the system to achieve more accurate predictions. MNB/Priors model gave similar performance to the keyword matching algorithm because it only increases the priors for the labelled words and it does not involve any learning.

Our experimental results confirmed the observation by Settles (2011) that

MNB usually performs better than GE when the labelled features are few. In this case, MNB/Priors+EM$_1$ had 8% higher F_1 than GE/Random. The difference mainly resulted from the higher recall of the MNB model.

As we expected, GE/MNB model achieved higher recall than GE/Random due to the automatically included keywords for the positive class. However, its lowered precision is likely due to the noise. Since we use mutual information algorithm to mine keywords automatically instead of composing keyword list manually, the noise in the input keyword list cannot be avoided. The noise will be propagated when running EM algorithm with MNB, causing wrong keywords to be included along with the correct ones. The final GE/MNB model marginally outperformed GE/Random in terms of macro F_1 score, but it still lagged behind MNB/Priors+EM$_1$ model.

Despite the lacklustre performance of GE models, the ensemble model achieved roughly equal performance as MNB/Priors+EM$_1$, showing its robustness to the performance of individual models.

We also compared with fully-supervised MaxEnt model varying the number of training documents. MaxEnt (γ=0.1) was trained using 10% of the corpus (2,000 labelled documents in total and 100 labelled documents for each category). Its macro average F_1 roughly matched the lightly-supervised counterparts. Based on the user study conducted in Settles (2011), labelling a document will take around 11 seconds on average. Therefore, the estimated time to label 2,000 documents would be six hours, which is approximately 25 times more than the time to label keywords. The last line shows the performance of MaxEnt model trained using all the available training documents (around 19,000 documents). Its macro average F_1 was 16% higher than the GE$_1 \wedge$ MNB$_1$ model.

5.4 Evaluations on RTB Dataset

As contextual advertising is one of the most critical applications for our model, we also ran experiments on the RTB dataset. Different from the experiments for 20 Newsgroups corpus, We made use of the full-fledged pipeline and composed the list of keywords with the help of the keyword suggestion algorithm proposed in section 3.1. Figure 5.2 shows the list of keywords for each category. We set a time limit of 10 minutes to compose the keyword list for each category. The labelling process stopped whenever the annotator finished labelling the suggested keywords or the time ran out.

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*
Diabetes	*diabetes, insulin, glucose, mellitus, diabetic, ketoacidosis, ketosis, dka, hyperosmolar, hyperglycemic, nonketotic, niddm, polydipsia, polyphagia, polyuria, glucagon, metformin*
Sleep Disorder	*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*
Nutrition	*nutrition, protein, nutrients, livestrong, vitamin, intake, carbohydrates, fiber, myplate, minerals, carb, grain, metabolism, dietary, antioxidants, calcium, nutritional, nutritious, nutritionist*

Figure 5.2: Labeled words for each category.

We summarise the results of each classifier in figure 5.3.

We observe a similar pattern that semi-supervised learning algorithms outperformed keyword voting methods by a large margin. GE models performed considerably better and achieved better results than MNB models because we now have more and higher quality keywords as input. The GE/MNB model had 4% higher

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
MNB/Priors						
2: +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$_1$∧ 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

Figure 5.3: Precision/Recall/F_1 scores on RTB dataset.

recall than GE/Random model. The algorithm included on average 53 new keywords automatically for each category, which is roughly two times the size of the input keywords. This helps the model cover more related documents where the input keyword may not be present. Figure 5.4 reveals the top ten automatically included keywords for each category.

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

Figure 5.4: Top 10 automatically added constraints in GE/MNB for each class.

The macro average precision dropped by 2% because of the lower precision of *Cold & Flu* and *Cancer* category. After performing error analysis, we found out that some words frequently appearing together with the input keywords were added by MNB. E.g. the words "cold" and "congestion" were added for the *Cold & Flu* category. While they do frequently appear in the target documents, they might also appear in other contexts such as "cold weather" and "traffic congestion". These keywords will cause the system to make false positive errors. This highlights an area for future work to investigate how to reduce the risk of introducing noises when automatically including constraints.

The $GE_1 \wedge MNB_1$ model achieved impressive performance. It improved 4% further from GE/MNB and gave 69% macro average F_1. It also obtained best or close to the best F_1 score for individual categories among all lightly-supervised classifiers.

Compared to the experiments on the 20 Newsgroups dataset, the gap between the lightly-supervised models and the fully-supervised MaxEnt model was much smaller. The macro average F_1 of $GE_1 \wedge MNB_1$ was only 3% lower than the MaxEnt model trained using the whole training set. This is most likely due to the higher quality of the input keywords, which results from the novel keyword suggestion algorithm.

Chapter VI

CONCLUSION

6.1 Thesis Summary

In this thesis, we first introduced the problem of supervised text classification methods, namely requiring sizeable labelled dataset to train and assuming a predefined list of categories. When the categories/topics are constantly changing, it will take users lots of effort to label new sets of documents or to fine-tune the models. Such applications include but not limited to online advertising, social media and online forums.

To address these limitations, we introduced the task of *lightly-supervised one-class classification*, which lies at the intersection of two previously studied problems: semi-supervised learning and one-class classification. To our best knowledge, our work is the first to formally define the lightly-supervised one-class classification problem, where we can build the classifier with only a handful of input keywords belonging to a single class.

We firstly adapted/applied two previous state-of-the-art semi-supervised learning methods, generalized expectation (GE) criteria (Druck et al. (2008)) and multinomial naïve Bayes (MNB) with priors (Settles (2011)) in the one-class classification problem. Their distinct characteristics and the underlying difference between generative models and discriminative models motivated us to combine the two models by exploiting the knowledge MNB model learned to form additional training signals for GE. Besides, we also proposed an ensemble-based approach which achieved impressive and robust performance.

We benchmarked our proposed models with a list of semi-supervised, fully-supervised and rule-based methods on two different datasets, one for text classification, another for contextual advertising. We showed that semi-supervised learning

methods outperformed rule-based approach by a large margin in general. The final ensemble model $GE_1 \wedge MNB_1$ achieved a macro average F_1 of 0.69 on the contextual advertising dataset, closing 50% of the gap between previous state-of-the-art semi-supervised models to a fully-supervised MaxEnt model.

The proposed method has been deployed into a production system for online advertising, which predicts the labels of roughly four billion URLs per day.

6.2 Limitations and Future Works

As we mentioned in section 5.4, our proposed method of reading off salient features from MNB model and adding them directly to GE constraints may introduce noise. While additional constraints can improve the recall, it usually lowers the precision. Future works can study how to combine the two models in a better way to avoid the noisy keywords.

Applying more sophistically deep learning models to the lightly-supervised one-class classification problem is also an exciting area for future work. However, most modern deep learning methods are fully-supervised and require a much larger set of labelled documents to train than simple linear models like MaxEnt. Deep reinforcement learning or generative neural networks might provide possibilities to train deep models in a lightly-supervised manner.

In the application of contextual advertising, sometimes it is not possible to detect the category of the web page based on the textual information alone. Some real-world examples we observed are 1) a page where students practice English comprehension. The main content of the page is a story. 2) a picture or video page containing little or no textual information. In the first example, we need to combine with domain classification on website level. In the second example, we need to combine with image/video classification. These cases hint that multi-modal learning might help to improve the accuracy.

REFERENCES

Aggarwal, C. C. and Zhai, C. 2012. A survey of text classification algorithms. In Mining text data, pp. 163–222. : Springer.

Apté, C., Damerau, F., and Weiss, S. M. 1994. Automated learning of decision rules for text categorization. ACM Transactions on Information Systems (TOIS) 12.3 (1994): 233–251.

Broder, A., Fontoura, M., Josifovski, V., and Riedel, L. 2007. A semantic approach to contextual advertising. In Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 559–566. :

Carvalho, V. R. and Cohen, W. W. 2005. On the collective classification of email speech acts. In Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 345–352. :

Chakrabarti, S., Dom, B., Agrawal, R., and Raghavan, P. 1997. Using taxonomy, discriminants, and signatures for navigating in text databases. In VLDB, volume 97, pp. 446–455. :

Chatterjee, P., Hoffman, D. L., and Novak, T. P. 2003. Modeling the clickstream: Implications for web-based advertising efforts. Marketing Science 22.4 (2003): 520–541.

Church, K. W. and Hanks, P. 1990. Word association norms, mutual information, and lexicography. Computational linguistics 16.1 (1990): 22–29.

Dermouche, M., Khouas, L., Velcin, J., and Loudcher, S. 2013. Ami&eric: How to learn with naive bayes and prior knowledge: an application to sentiment analysis. Atlanta, Georgia, USA (2013): 364.

Dietterich, T. G. 2000. Ensemble Methods in Machine Learning. Multiple Classifier Systems 1857 (2000): 1–15.

Druck, G. 2011. Generalized expectation criteria for lightly supervised learning. PhD thesis, University of Massachusetts Amherst.

Druck, G., Mann, G., and McCallum, A. 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, pp. 595–602. :

Druck, G., Settles, B., and McCallum, A. 2009. Active learning by labeling features. In Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1, pp. 81–90. :

Graça, J., Ganchev, K., and Taskar, B. 2007. Expectation maximization and posterior constraints. In NIPS, volume 20, pp. 569–576. :

Guo, Y., Reichart, R., and Korhonen, A. 2015. Unsupervised declarative knowledge induction for constraint-based learning of information structure in scientific documents. Transactions of the Association for Computational Linguistics 3 (2015): 131–143.

Hasan, K. S. and Ng, V. 2014. Why are you taking this stance? identifying and classifying reasons in ideological debates. In EMNLP, pp. 751–762. :

Jin, Y., Kan, M.-Y., Ng, J.-P., and He, X. 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, pp. 780–790. :

Johnson, D. E., Oles, F. J., and Zhang, T. 2003. Decision-tree-based symbolic rule induction system for text categorization.

Kim, Y. 2014. Convolutional neural networks for sentence classification. arXiv preprint arXiv:1408.5882 (2014):

King, B. and Abney, S. P. 2013. Labeling the languages of words in mixed-language documents using weakly supervised methods. In HLT-NAACL, pp. 1110–1119. :

Kohlschütter, C., Fankhauser, P., and Nejdl, W. 2010. Boilerplate detection using shallow text features. In Proceedings of the third ACM international conference on Web search and data mining, pp. 441–450. :

Korde, V. and Mahender, C. N. 2012. Text classification and classifiers: A survey. International Journal of Artificial Intelligence & Applications 3.2 (2012): 85.

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

Lasserre, J. A., Bishop, C. M., and Minka, T. P. 2006. Principled hybrids of generative and discriminative models. In Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on, volume 1, pp. 87–94. :

Lee, W. S. and Liu, B. 2003. Learning with Positive and Unlabeled Examples Using Weighted Logistic Regression. Algorithmic Learning Theory 348.1 (2003): 71–85.

Levy, O., Goldberg, Y., and Dagan, I. 2015. Improving distributional similarity with lessons learned from word embeddings. Transactions of the Association for Computational Linguistics 3 (2015): 211–225.

Li, X., Philip, S. Y., Liu, B., and Ng, S.-K. 2009. Positive unlabeled learning for data stream classification. In SDM, volume 9, pp. 257–268. :

Li, Y. H. and Jain, A. K. 1998. Classification of text documents. The Computer Journal 41.8 (1998): 537–546.

Liu, B. and Zhang, L. 2012. A survey of opinion mining and sentiment analysis. In Mining text data, pp. 415–463. : Springer.

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

Liu, B., Li, X., Lee, W. S., and Yu, P. S. 2004. Text classification by labeling words. In AAAI, volume 4, pp. 425–430. :

Lucas, M. and Downey, D. 2013. Scaling semi-supervised naive bayes with feature marginals. In Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, pp. 343–351. :

Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., and Potts, C. 2011. Learning word vectors for sentiment analysis. In Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies-volume 1, pp. 142–150. :

Mann, G. S. and McCallum, A. 2008. Generalized expectation criteria for semi-supervised learning of conditional random fields. (2008):

McCallum, A., Nigam, K., et al. 1998. A comparison of event models for naive bayes text classification. In AAAI-98 workshop on learning for text categorization, volume 752, pp. 41–48. :

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. 2013. Distributed representations of words and phrases and their compositionality. In Advances in neural information processing systems, pp. 3111–3119. :

Moya, M. M. and Hush, D. R. 1996. Network constraints and multi-objective optimization for one-class classification. Neural Networks 9.3 (1996): 463–474.

Pang, B., Lee, L., and Vaithyanathan, S. 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, pp. 79–86. :

Pennington, J., Socher, R., and Manning, C. D. 2014. Glove: Global vectors for word representation. In EMNLP, volume 14, pp. 1532–1543. :

Sahami, M., Dumais, S., Heckerman, D., and Horvitz, E. 1998. A bayesian approach to filtering junk e-mail. In Learning for Text Categorization: Papers from the 1998 workshop, volume 62, pp. 98–105. :

Schapire, R. E., Rochery, M., Rahim, M., and Gupta, N. 2002. Incorporating prior knowledge into boosting. In ICML, volume 2, pp. 538–545. :

Scheirer, W. J., de Rezende Rocha, A., Sapkota, A., and Boult, T. E. 2013. Toward open set recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence 35.7 (2013): 1757–1772.

Schölkopf, B., Platt, J. C., Shawe-Taylor, J., Smola, A. J., and Williamson, R. C. 2001. Estimating the support of a high-dimensional distribution. Neural computation 13.7 (2001): 1443–1471.

Settles, B. 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, pp. 1467–1478. :

Shu, L., Xu, H., and Liu, B. 2017. Doc: Deep open classification of text documents. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pp. 2911–2916. : Association for Computational Linguistics.

Tang, D., Wei, F., Yang, N., Zhou, M., Liu, T., and Qin, B. 2014. Learning sentiment-specific word embedding for twitter sentiment classification. In ACL (1), pp. 1555–1565. :

Wang, S., Chen, Z., Fei, G., Liu, B., and Emery, S. 2016. Targeted Topic Modeling for Focused Analysis. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '16 (2016): 1235–1244.

Wang, S. and Manning, C. D. 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, pp. 90–94. :

Weiss, S. M., Apte, C., Damerau, F. J., Johnson, D. E., Oles, F. J., Goetz, T., and Hampp, T. 1999. Maximizing text-mining performance. IEEE Intelligent Systems and their applications 14.4 (1999): 63–69.

Yan, J., Liu, N., Wang, G., Zhang, W., Jiang, Y., and Chen, Z. 2009. How much can behavioral targeting help online advertising? In Proceedings of the 18th international conference on World wide web, pp. 261–270. :

Yu, H., Han, J., and Chang, K. C.-C. 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, pp. 239–248. :

Zhang, J. and Yang, Y. 2003. Robustness of regularized linear classification methods in text categorization. In Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval, pp. 190–197. :

Zhao, L., Huang, M., Yao, Z., Su, R., Jiang, Y., and Zhu, X. 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, pp. 2877–2883. :

Biography

Mr. Yiping Jin was born in Beijing, China, on March, 1990. He received B.Comp. in Computer Science, from National University of Singapore, Singapore, in 2013 with First Class Honours Degree. His bachelor degree was supervised by Assoc. Prof. Min-Yen Kan. He received Singapore Ministry of Education full scholarship for undergraduate study as well as German Academic Exchange Service (DAAD) scholarship and Baden-Württemberg scholarship for two summer programmes in Germany.

He currently works as Senior Research Scientist at Knorex Pte. Ltd. (Singapore). His research interests include natural language processing (NLP), machine learning and computational advertising. Prior to joining Knorex, Mr. Yiping Jin worked at National University of Singapore and Institute for Infocomm Research as research assistant and research engineer respectively. He has published three research papers in renowned international NLP conferences as follows.

1. Yiping Jin, Dittaya Wanvarie., Phu Le. "Combining Lightly-Supervised Text Classification Models for Accurate Contextual Advertising". Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers). (2017). pp. 545–554.

2. Yiping Jin, Phu Le. "Selecting Domain-Specific Concepts for Question Generation with Lightly-Supervised Methods". Proceedings of The 9th International Natural Language Generation conference, pages 133–142, Edinburgh, UK, September 5-8 2016.

3. Yiping Jin, Min-Yen Kan, Jun-Ping Ng, Xiangnan He. "Mining Scientific Terms and their Definitions: A Study of the ACL Anthology". Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, pages 780–790, Seattle, Washington, USA, 18-21 October 2013.