Computer Engineering and Intelligent Systems
ISSN 2222-1719 (Paper) ISSN 2222-2863 (Online)
Vol.6, No.3, 2015
www.iiste.org

Positive Unlabeled Learning Algorithm for One Class Classification of Social Text Stream with only very few Positive Training Samples

Abhinandan Vishwakarma
Research Scholar, Technocrats Institute of Technology, Bhopal, Madhya Pradesh, India
abhinandantit@gmail.com

Abstract
Text classification using a small labelled set (positive data set) and large unlabeled data is seen as a promising technique especially in case of text stream classification where it is highly possible that only few positive data and no negative data is available. This paper studies how to devise a positive and unlabeled learning technique for the text stream environment. Our proposed approach works in two steps. Firstly we use the PNLH (Positive example and negative example labelling heuristic) approach for extracting both positive and negative example from unlabeled data. This extraction would enable us to obtain an enriched vector representation for the new test messages. Secondly we construct a one class classifier by using one class SVM classifier. Using the enriched vector representation as the input in one class SVM classifier predicts the importance level of each text message.

Keywords: Positive and unlabeled learning, one class SVM (Support Vector Machine), one class classification, text stream classification.

Introduction

With the rapid growth of the social networking sites, the social text stream data such as weblogs, message boards, mailing lists have become ubiquitous. A collection of text communication that arrives over time is referred as social text stream. Each piece of the text in the stream is associated with some social attributes such as author, reviewer, sender and recipients. Much of the data in the social scenario arises in the context of streaming application, in which the text arrives as a continuous and massive stream of text segments [1]. Such application present challenge, because data arrives continuously which requires continuous updating of model and one cannot store all the data on disk for re- processing.

Social text stream generate large volume of text data from various type of sources. Efficiently organizing and summarizing these streams have become an important issue [2,4] as these streams have rich content of text, social actors and relations and other temporal information. Social text stream is substantially different from general text stream data: 1) Social text stream data contains rich social connections, and 2) Social text stream data is more context sensitive.

The purpose of this work is to use the semi-supervised classification techniques for the classification of social text stream. All the semi-supervised classification techniques are performing quite well on static text and still to be applied on dynamic text. Traditionally, supervised learning techniques are proposed to build accurate classifier that requires a large number of labeled training examples from the predefined classes for learning. There are some famous traditional text classification methods, such as support vector machine, Naïve Bayes, K-Nearest Neighbor and so on which belongs to this category. All these methods needs lots of documents manually labeled from every class to train classifier so they are problematic getting lots of document manually labeled is so time consuming. Another alternative approach of Positive and Unlabeled (PU) learning also known as semi-supervised learning has been investigated in the recent years. Semi-supervised learning reduces the amount of labeled training data by developing the classification algorithm that can learn from a set of labeled positive examples augmented with a set of unlabeled examples. In the other words given a set P of positive examples of a particular class and a set U of unlabeled examples (which contains both the hidden positive

and hidden negative examples), we build a classifier using P and U to classify the data in U as well as future test data. Several PU learning technique [5-8] have recently been proposed to solve the PU learning problem in document classification domain. The dynamic data stream is such environment where it is highly possible that only a small set of positive data and no negative data is available in practice.

In this work, we propose to classify the text stream into one class. One class learning on text stream is a new and challenging research issue. In one class learning only one class of samples is labeled in the training phase [9] and the final goal is to predict whether the new instance fall into the same category as the positive example or not. The labeled class is typically called the *target class,* while all other samples that do not belong to target class are defined as non target class. The purpose of the one class learning is to build a distinctive classifier that decides whether a test instance belongs to that the target class or the non target class. Such one class classification problem often referred to as outlier detection.

In this paper we present a study to measure and analyze the abusive content on online social networks.

Our work is based on a large dataset of “wall” message from facebook. Wall posts are the primary form of communication on facebook where a user can leave messages on the public profile for a friend. Wall messages remain on a user’s profile unless explicitly removed by the owner. As such, wall messages are the intuitive place to look for attempts to spread malicious content on Facebook since the messages are persistent and public, *i.e.* likely to be viewed by the target user and potentially the target’s friends.

So many abusive posts are being posted on the social networking site on regular basis which is harming our society to a great extent, our main purpose is to classify these posts from other. The one class SVM is the best known support vector learning methods [10] for one class classification problems. The one class SVM approaches allows for a solution as it only requires the data of the class to be discovered to learn a decision function. One class SVMs are an extension of the original two class SVM learning algorithms to enable a training of a classifier in the absence of any negative example data. One class SVM determines the hyperplane that separates the target class from the other class with the maximal margin i.e. it defines the boundary around the target class, such that it accepts as much of target object as possible, while it minimizes the chance of accepting outlier or non-target object.

This paper is organized as follows: section II introduces a related work, and our work is presented in section III. In section IV, we introduced our method in detail. Then in section V later we will evaluate experimental analysis. At last conclusion is given in section VI.

Related Work

Sometimes it is very difficult to obtain a set of negative examples for training the classifier. A new approach known as positive unlabeled learning have been studied in the recent year to reduce the human effort of labeling the negative training examples. A number of PU learning methods were proposed [12-14]. All positive and unlabeled learning methods work in two steps (1) First step is to fine out reliable negative examples from set of unlabeled examples (2) Second step is to construct the classifier from positive examples and the extracted reliable negative example from step 1. All the methods differ from each other in the way how they extract the reliable negative examples and the classifier used for training.

Numbers of techniques are available to classify the data stream [11-17]. Classification of data stream is a challenging area of research. There are many problems to be solved, such as handling continuous attribute, concept drift, sample taken question and classification accuracy problem. These challenges are traditionally solved by using either an incremental learning [18-19] or an ensemble learning approach [17,20]. For incremental learning problem is to build a model form the small portion of the data stream and continuously update the model by using newly arrived samples. For ensemble learning, a number of base classifier are build from the different portion of data stream and the final goal is to combine the models to form an ensemble classifier for prediction. All the existing method for the classification of text stream can be categorized on the basis of how they deal with historical data. Some methods [11,13] discards the historical data after certain period of time while other methods choose historical record that matches with the current data to help to train a better model instead of using just a recent data alone [14,17].

For example in [21], a one class SVM that uses only positive data to build the SVM classifier was proposed. One class SVM and support vector data description are representative methods [21-22]. Both methods aims to construct a one class classifier using only the target class. The advantage of these methods is that they can cope with any one class classification problem. Such approaches are different from our method in that they do not use the unlabeled data for training.

Almost all the current PU learning methods have been devised for static data environment. The problem of employing one-class learning for data stream was recently added by [23] in their positive unlabeled learning method, which refines positive samples and includes sample from the most recent data chunk for data stream classification. Our work differs from the previous work in the way that we use different approach for the refinement of the positive training samples and content on training data. The other one class learning methods [23-25], they are developed mainly for document related one class classification problem. Their main task is to extract negative samples from unlabeled data and to construct a binary classifier using target samples and extracted negative examples.

PROBLEM DEFINITION

If size of the positive training set is small we cannot consider it for training the classifier because it may not reflect the true feature distribution of the entire positive example in the domain. Data stream arrives continuously and at very rapid rate, we assume text stream arrives on chunk by chunk basis and initially we are assuming only few positive samples. Suppose text stream constitutes sample $T_{1,}$ $T_{2\ldots,}$ T_m where each T_i (i = 1, 2……m) denotes the sample or chunk arrives between time $t_{\ t-1}$ and $t_{t.}$ Here T_c is called the current sample and T_{c+1} is the next chunk to come is called the target chunk. For the classification we also assume that only the instances in the most recent chunk (T_c) are accessible and once the algorithm moves from chunk T_{c-1} to chunk $T_{c,}$ all instances in the chunk T_{c-1} and predecessor are inaccessible.

The positive and unlabeled classification of text stream can be modeled as follows:

The training set for the classifier includes only few positive text streams samples T_P and lots of unlabeled text streams $T_{U,}$ where T_U constitutes both positive text streams T_P as well as negative text stream $T_{N.}$ All the positive training samples can be grouped together and let termed as $PT_{i.}$ Since initially we are considering only few positive training samples, the size of PT_i is very small. Accuracy of the classifier depends upon size and the content of positive training sample PT_i and the negative training sample $NT_{i,}$ so in our proposed technique we will try to extract both positive training sample T_P and negative training sample T_N and then we will enlarge the size of PT_i and $NT_{i.}$

PROPOSED TECHNIQUE

The accuracy of the classifier largely depends upon the size of the training set i.e. the number of positive and negative labeled example and the content of the training set. Since initially we are starting with only few positive labeled examples T_P and unlabeled data $T_{U,}$ our first job is to extract reliable positive and negative example by using T_P and $T_{U.}$ For this purpose we will use the technique known as *positive example and negative example labeling heuristic (PNLH)* addressed by Yu [30] but with some modification. *PNLH* has been successfully applied with the static text classification context, this is the first time it is being applied to the dynamic text classification. Our work differs from Yu [30] work on one aspect. We are interested in one class classification, which requires only positive data samples for training so we pay our more attention in extracting reliable positive example then negative example. One class classification distinguishes one class of data from the rest of the feature space given only a positive data set. Since we are interested in one class classification, we pay attention in extracting the positive examples and increasing the size of the positive training sample. *PNLH* employs a two stage process extraction and enlargement.

First stage known as extraction and aims at extracting set of negative examples from the unlabeled data based on the concept of core vocabulary. Core vocabulary contains the feature or keyword with the feature strength greater than the threshold. Negative examples can be extracted by comparing feature distribution of given positive example and unlabeled data. Initially we are considering only few positive examples so by comparing feature distribution with the unlabeled data will extract large number of negative examples. These extracted negative examples are not true negative examples and may contain large number of positive examples since initially taken set of positive examples are very small and not representing true feature distribution. In the first step we try to extract negative examples not the positive examples because the size of the T_p is very small if we try extract positive example in the first step, it will be very less in number and will not represent true feature distribution. Although our aim is to extract only positive example since the construction of one class classifier requires only positive examples and no negative examples. In the next stage we will try to extract reliable positive examples and will combine this extracted set with the initially taken sample T_P to increase the size of the positive example.

Second stage is the enlargement. It enlarges the size of the positive example set. Partition based approach in [30] is used for that purpose. Negative examples obtained in stage one is partitioned into k partition $N_{1,}$ N_2 ...... $N_{K.}$ Each partition focuses on a smaller set of more related features. In order to extract more positive example in this stage we compare the similarity of document with unlabeled data with the centroid of $T_{p.}$ Any kind of existing clustering technique can be applied for partitioning of negative examples obtained in stage one [26-28]. We adopt a k-means clustering algorithm as mentioned in [29]. The complete description of *PNLH* approach is shown in fig 1.

For one class classification best known classifier in one class SVM classifier which aims to construct a classifier using only the target class. One class SVM is a special type of support vector machine where learning intend to find a hyper-sphere enclosing all positive sample [28] or hyper planes separating positive examples from the origin with maximum margin [27].

Suppose the training target is $S = \{x_{1,} x_{2,} ...., x_n\}$, these are the n positive samples. One class SVM aims to determine hyperplane in the input space to separate the positive samples from the origin with the maximal margin. Hyperplane is determined by $W^T.X= \rho$. which is described by one class SVM model in [29]. Kernel transformation function *Φ()* are employed to transform an input example from one space to another, which gives the hyperplane denoted by $W^T.\Phi(X)=\rho$. This objective is defined by the convex problem given in Eq. (1), where W is the orthogonal to the determined hyper-plane, v is the fraction of positive samples not separated from the origin by the determined hyper-plane. X_i is the i^{th} positive sample and ξ is the slack variable which defines the penalty if a sample is not separated from the origin.

$$\min \quad \frac{1}{2}\,\|W\|^2 - \rho + \frac{1}{V.N}\sum_{i=0}^{n} \xi_i$$

$$s.t.\ W^T.\Phi(X_i) \geq \rho - \xi_i$$

$$\xi_i \geq 0,\ i=1,2,3.\ .\ .\ .n$$

For a test sample x_t if

$$W^T.\Phi(X_t) \geq \rho$$

Then the new instance x_t is classified into the target class, otherwise it belongs to the non-target class. The kernel function is denoted by $\Phi(x_i)$. A kernel is a function that takes the original data points and several other parameters and increases their dimensionality i.e. data can be separated into a higher dimensional feature space using kernels using kernels. Some commonly used kernels are linear, polynomial, radial basis function or Gaussian kernel, sigmoid. A good choice of the kernel function and the corresponding parameter will allow the data to then be separable by hyper-plane. The Gaussian kernel function can be given as;

$$K(X_i,X_j) = \Phi(X_i).\Phi(X_j) = \exp\left(\frac{\|x_i - x_j\|^2}{2\sigma^2}\right)$$

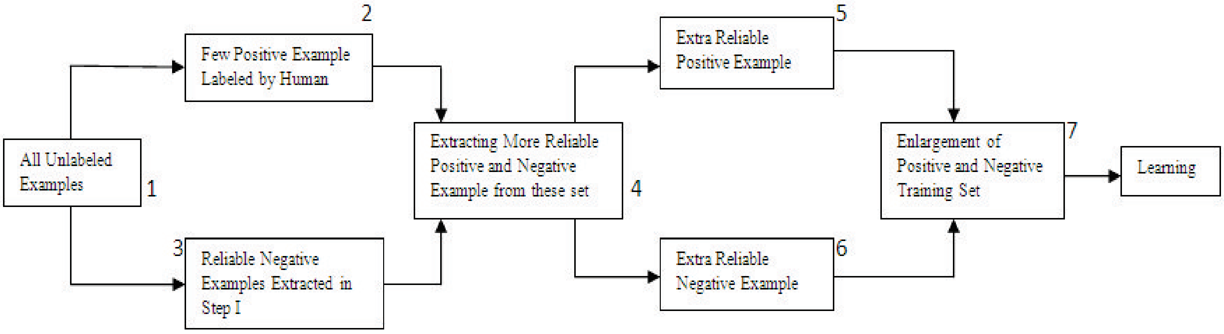
All Unlabeled Examples
1
Few Positive Example Labeled by Human
2
Reliable Negative Examples Extracted in Step I
3
Extracting More Reliable Positive and Negative Example from these set
4
Extra Reliable Positive Example
5
Extra Reliable Negative Example
6
Enlargement of Positive and Negative Training Set
7
Learning

Fig. 1

REFERENCES

[1] J. Kleinberg. Data Stream Management: *Processing High-Speed Data Streams, chapter Temporal Dynamics of On-Line Information Streams*. Springer, 2006.
[2] S. D. Afantenos, *An introduction to the summarization of evolving events: Linear and non-linear evolution*. In LNCS, pages 91–99, 2005.
[4] A. Krause, J. Leskovec, and C. Guestrin. *Data association for topic intensity trackingn.* In ICML, 2006.
[5] B. Liu, W. S. Lee, P. S. Yu, and X. Li, *Partially supervised classification of Text Documents*, ICML, 2002.
[6] H. Yu, J. Han, and K. C. -C. Chang, *PEBL: Positive Example Based Learning for Web Page classification Using SVM*, SIGKDD, 2002.
[7] X. Li and B. Liu, *Learning to Classify Texts Using Positive and Unlabeled Data*, IJCAI, 2003.
[8] B. Liu, Y. Dai, W. S. Lee, P. S. Yu, and X. Li, *Building Text classifier Using Positive and Unlabeled Examples*, ICDM, 2003.
[10] V. Vapnik. *Statistical learning theory*. Springer-Verlag, London, UK, 1998.
[11] C. C. Aggarwal, J. Han, J. Wang, and P. S. Yu, *On Demand classification on Data Streams*, SIGKDD, 2004.
[12] C. C. Aggarwal and P. S. Yu, *LOCUST: An Online Analytical Processing Framework for High Dimensional classification of Data Steams*, ICDE, 2008.
[13] G. Hulten, L. Spencer, and P. Domingos, *Mining time-changing data streams*, SIGKDD, 2001.
[14] W. N. Street and Y. Kim, *A streaming ensemble algorithm (SEA) for large-scale classification*, SIGKDD, 2001.
[15] Y. Chen, G. Dong, J. Han, B. W. Wah, and J. Wang, *Multi-dimensional regression analysis of time-series data streams*, VLDB, 2002.
[16] H. Wang, J. Yin, J. Pei, P. S. Yu, and J. X. Yu, *Suppressing model overfitting in mining concept-drifting data streams*, SIGKDD, 2006.
[17] H. Wang, W. Fan, P. S. Yu, and J. Han, *Mining concept-drifting data streams using ensemble classifiers*,

SIGKDD, 2003.
[18] P. Domingos & G. Hulten, *Mining high-speed data streams*, Proc. of KDD, 2000.
[19] G. Hulten, L. Spencer, & P. Domingos, *Mining time-changing data streams*, Proc. of KDD, 2001.
[20] W. Street & Y. Kim, *A streaming ensemble algorithm (SEA) for large-scale classification,* Proc. of KDD, 2001.
[21] B. SchÄolkopf, J. C. Platt, J. C. Shawe-Taylor, A. J. Smola, and R. C. Williamson, *Estimating the Support of a High-Dimensional Distribution*, Neural Computation, vol. 13, pp. 1443-1471, 2001.
[22] D. M. J. Tax and R. P. W. Duin. *Support vector data description*. Machine Learning, 54(1):45–66, 2004.
[23]. X. Li, P. Yu, B. Liu, & S. Ng, *Positive Unlabeled Learning for Data Stream Classification, in Proc. of SDM*, 2009.
[24]B. Liu, Y. Dai, X. Li, W. S. Lee, and P. S. Yu. *Building text classifiers using positive and unlabeled examples*. ICDM, pages 179–186, 2003.
[25] G. P. C. Fung, J. X. Yu, H. Lu, and P. S. Yu. *Text classification without negative examples revisit*. TKDE, 18(6):6–20, 2006.
[26]P. Bradley and U. Fayyad, *Refining Initial Points for k-Means Clustering*, Proc. 15th Int'l Conf. Machine Learning, 1998.
[27] D.R. Cutting, D.R. Karger, J.O. Pederson, and J.W. Tukey, *Scatter/Gather a Cluster-Based Approach to Browsing Large Document Collections*, Proc. 15th Int'l Conf. Research and Development in Information Retrieval, 1992.
[28] B. SchÄolkopf, J. Platt, J. S. Taylor, A. J. Smola, and R. Williamson. *Estimating the support of a high-dimensional distribution*. Neural Computation, 13:1443–1471, 2001.
[29] B. Larsen and C. Aone, *Fast and Effective Text Mining Using Linear-Time Document Clustering,* Proc. Fifth Int'l Conf. Knowledge Discovery and Data Mining, 1999
[30] G. P. C. Fung, J. X. Yu, H. Lu, and P. S. Yu, *Text classification without Negative Examples Revisit*, IEEE TKDE, vol. 18, pp. 6-20, 2006.