Pattern Recognition Letters 116 (2018) 187–194

Contents lists available at ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

Online Biterm Topic Model based short text stream classification using short text expansion and concept drifting detection

Xuegang Hu[a,b], Haiyan Wang[a], Peipei Li[a,*]

[a] School of Computer and Information, Hefei University of Technology, Hefei, Anhui 230601, PR China
[b] Anhui Province Key Laboratory of Industry Safety and Emergency Technology, Hefei, Anhui 230601, PR China

ARTICLE INFO

Article history:
Received 3 May 2018
Available online 16 October 2018

MSC:
41A05
41A10
65D05
65D17

Keywords:
Short text streams
Classification
Concept drifting
Online BTM

ABSTRACT

Short text stream classification suffers from enormous challenges, due to the sparsity, high dimension and rapid variability of the short text stream. In this paper, we present a short text stream classification approach refined from online Biterm Topic Model (BTM) using short text expansion and concept drifting detection. Specifically, in our method, we firstly extend short text streams from an external resource to make up for the sparsity of data, and use online BTM to select representative topics instead of the word vector to represent the feature of short texts. Secondly, we propose a concept drift detection method based on the topic model to detect the hidden concept drifts in short text streams. Thirdly, we build an ensemble model using several data chunks and update with the newest data chunk and results of the concept drift detection. Finally, extensive experimental results demonstrate that compared to well-known baselines, our approach achieves a better performance in the classification and concept drifting detection.

1. Introduction

With the rapid development of Internet technology, large amounts of short text streams have been produced in the real-world applications, including tweets and instant messages, to name a few. Short text streams contain the following three characteristics. First, each short text is too short to gain enough semantic information and the whole short texts suffer high sparseness. Second, short texts are generated within a short time and it is easy to cause the curve of dimensionality seriously. Third, text concepts produce potential drifts over time. Therefore, it is hard to directly apply the conventional classification methods to the short text streams, such as SVMs [21] and Random Forest[4]. To solve the sparseness of short texts in the classification, there are mainly two directions as follows. The first one is to enrich short texts with external resources. For example, Phan et al. [15] proposed to mine hidden topics from universal datasets for short text expansion using the topic model, such as LDA [2]. The other one is to expand short texts by using rules and statistics hidden in short texts. For instance, Kim et al. [10] proposed a novel kernel for short text classification, called language independent kernel based on semantic annotation. However, all aforementioned short text classification approaches belong to the batch algorithms, and they cannot be applied directly to classify short text streams. To handle the short text streams, a well-known approach called online Biterm Topic Model (BTM) [5] has been proposed. It builds on data chunks with equal time windows, and uses the aggregated word co-occurrence patterns based on biterms[1] in each time slice for topic discovery. Thus, online BTM can discover more prominent and coherent topics than the traditional topic model. However, biterms are more sparse than words in short text streams. Meanwhile, online BTM ignores the problem of concept drifts hidden.

To end this, we propose a refined short text stream classification approach based on online BTM. Our main contributions of this paper are summarized below. 1) Our approach further alleviates the effects of the sparseness of short texts, and achieves the dimensionality reduction in the classification of short text streams. That is, in order to reduce the effects of the sparseness, we use an external corpus containing topics consistent with the distribution of the given short text streams to enrich short texts, and then we select representative topics as feature vectors instead of word vectors to decrease the high dimension. 2) Our approach can effectively detect concept drifts hidden in short text streams. We first divide all data chunks used in building the ensemble model into class clusters according to the label. And then we calculate the

[*] Corresponding author.
E-mail address: peipeili@hfut.edu.cn (P. Li).

[1] A biterm denotes an unordered word-pair co-occurring in a short text.

https://doi.org/10.1016/j.patrec.2018.10.018
0167-8655/© 2018 Elsevier B.V. All rights reserved.

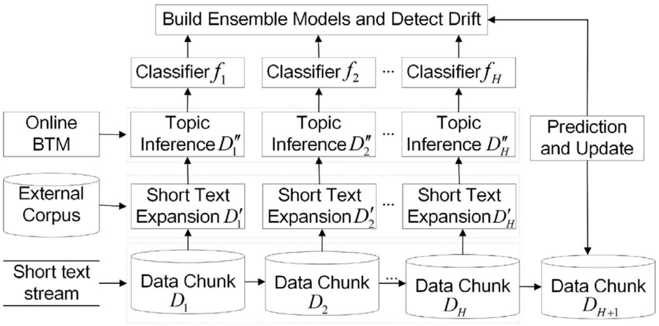


Fig. 1. Framework for short text stream classification.

semantic distance between short texts in the incoming data chunk and each class cluster, and use the minimum semantic distance to detect whether concept drifts happen or not in the incoming data chunk. 3) Experimental study shows that our approach can achieve a higher classification accuracy than classical data stream classification algorithms on two artificial data sets and a real data set. This is because that given a short text stream containing a set of data chunks, each data chunk is first expanded through an external corpus to alleviate the sparseness problem. Then the topic based representation is obtained for short text in a data chunk using online BTM to decrease the dimensions. Finally, an ensemble model is built on multi-data chunks and updated with the newest data chunk.

The rest of the paper is organized as follows. Section 2 presents the details of our approach. Section 3 shows experimental results. In Section 4, we introduce the related work on short text classification. Finally, the conclusions are summarized in Section 5.

2. Online BTM based short text stream classification approach

In this section, we first formalize our online BTM based short text stream classification approach. Then, we will give the details of the techniques used in our approach, including short text expansion based online BTM, concept drifting detection and ensemble model building.

2.1. Problem formalization

We formalize our approach for short text stream classification in this section. Suppose a short text stream is composed of N data chunks, namely $D = \{D_i\}_{i=1}^{N} (N \rightarrow \infty)$. Each data chunk contains a set of short texts, namely $D_i = \{d_j\}_{j=1}^{|D_i|}$, $|D_i|$ indicates the number of short texts in the i^{th} data chunk $(1 \leq i \leq N)$. In general, each document can be represented as a vector space, namely $d_j = \{(R^M, y_j) \mid y_j \in Y\}$, where R denotes the domain of a document space, M is the attribute dimensionality and Y denotes the set of document labels. Regarding the characteristics of sparseness and high dimension in short text streams, we enrich short texts in each data chunk using the external resource to reduce the sparseness, represented by $D_i' = \{d_j'\}_{j=1}^{|D_i|}$. Then we build the online BTM using extended short texts. The topics inferred by online BTM are used to represent the feature vector for dimension reduction, correspondingly each short text is represented as a set of topics, namely $D_i'' = \{d_j''\}_{j=1}^{|D_i|}$ and $d_j'' = \{z_{j,k}\}_{k=1}^{K}$, where $z_{j,\,k}$ denotes the kth topic in the jth short text and K is the number of topics. Finally, our approach aims to build a dynamic classifier $f : F_{\sum_{D_i''}} \rightarrow Y$ that adapts to the unseen streaming short texts and can detect the occurring concept drifts.

Fig. 1 illustrates the processing framework of our approach. To handle the problem of short text stream classification efficiently and effectively, it combines all H base classifiers to construct an ensemble model $E = \{f_1, f_2, \ldots f_H\}$, which is built on H data chunks, denoted as $S = \{D_1'', D_2'', \ldots, D_H''\}$, so that to each yet-to-come short text d_j, which is expanded and represented as d_j'',the ensemble E can predict its class label $y*$ which satisfies Eq. (1).

$$y* = argmax_{y \in Y} P\big(y|d_j'', E\big) = \sum_{h=1}^{H} w_{h,j} P\big(y|d_j'', f_h\big) \tag{1}$$

where $w_{h,\,j}$ indicates the weight of short text d_j''.

2.2. Online Biterm Topic Model based on short text expansion

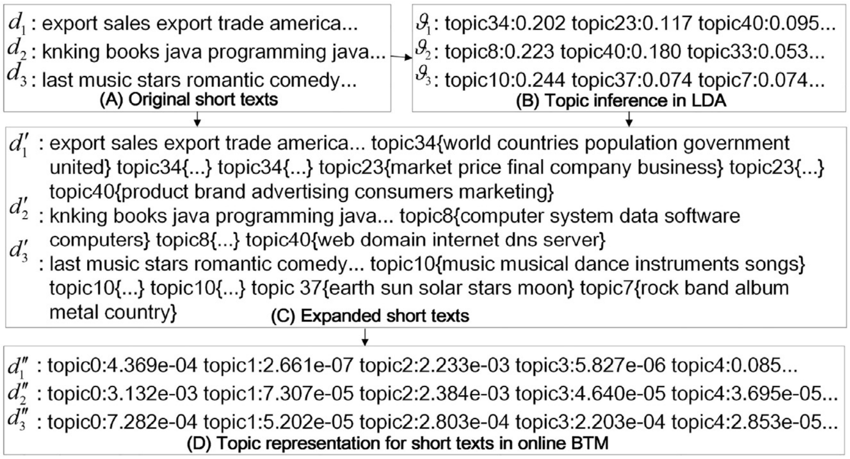
Biterm Topic Model (BTM) was proposed for short texts [5] and it was extended to handle short text streams, called online BTM. It reveals the correlation between words and enhances the semantic information via the word co-occurrence patterns based on biterms. Nevertheless, the word co-occurrence patterns increase the sparsity of the data for short texts. Thus, we propose a refined online BTM based on short text expansion. More precisely, first, we need to acquire an external corpus. Second, we expand short texts in arrived data chunks with the external corpus. Finally the online BTM is built on expanded data chunks. We will give the details of techniques as follows.

External corpus collection: In this subsection, we collect the external corpus collection for short text expansion. The quality of external corpus directly affects the expansion of short texts. Therefore, the external corpus must meet two characteristics [15], first it must be large and rich enough to cover most of contents in short text streams to be classified. Therefore, we select Wikipedia as the external corpus, because it is large enough. Second, it is highly topic consistent with short text streams. Thus, we obtain the correlate data from Wikipedia. More specifically, we first use the JWikiDocs tool[2] to obtain relevant raw pages according to the top 50 keywords in every category from short text streams, and each keyword crawls 100 pages. Then, we obtain the total size of 1.34GB with about 60,600 raw documents. After the preprocessing of deleting duplicate pages, HTML tags and page links, and removing stop words, finally we obtain 20,968 documents with 486,653 different words and an external corpus which is highly topic consistent with given short text streams to be classified.

Short text expansion: In this subsection, we select Latent Dirichlet Allocation (LDA) [2] instead of BTM [5] to mine hidden information for enriching short texts in terms of the external corpus mentioned above. Because the external corpus obtained from Wikipedia belongs to long texts, and LDA has a good effect on topic analysis for long texts, while BTM is suitable for short texts, and it is time-consuming because of the huge word co-occurrence patterns. We conduct the topic analysis for external corpus and obtain a LDA topic model M_{LDA} at first. Then we conduct topic inference for each data chunk according to M_{LDA} and gain the topic distribution of each short text in a data chunk, for more details please refer to the work in [15]. Finally, representative words for topics are added once or several times to each short text according to the topic distribution. To reduce the noise, we select top N_w representative words in top N_t topics to enrich each short text.

For instance, Fig. 2(B) shows the results of topic inference $\{\vartheta_j\}_{j=1}^{3}$ using the topic model M_{LDA} given a data chunk $\{d_j\}_{j=1}^{3}$ in Fig. 2(A), where θ_j denotes the topic distribution of short texts in a data chunk. Short texts in Fig. 2(A) come from the data set of Snippets [15]). To expand short texts, we assume that the higher probability in a topic for representative words indicates the larger

[2] JWikiDocs: http://jwebpro.sourceforge .



d_1 : export sales export trade america...
d_2 : knking books java programming java...
d_3 : last music stars romantic comedy...
(A) Original short texts

ϑ_1: topic34:0.202 topic23:0.117 topic40:0.095...
ϑ_2: topic8:0.223 topic40:0.180 topic33:0.053...
ϑ_3: topic10:0.244 topic37:0.074 topic7:0.074...
(B) Topic inference in LDA

d'_1 : export sales export trade america... topic34{world countries population government united} topic34{...} topic34{...} topic23{market price final company business} topic23{...} topic40{product brand advertising consumers marketing}
d'_2 : knking books java programming java... topic8{computer system data software computers} topic8{...} topic40{web domain internet dns server}
d'_3 : last music stars romantic comedy... topic10{music musical dance instruments songs} topic10{...} topic10{...} topic 37{earth sun solar stars moon} topic7{rock band album metal country}
(C) Expanded short texts

d''_1 : topic0:4.369e-04 topic1:2.661e-07 topic2:2.233e-03 topic3:5.827e-06 topic4:0.085...
d''_2 : topic0:3.132e-03 topic1:7.307e-05 topic2:2.384e-03 topic3:4.640e-05 topic4:3.695e-05...
d''_3 : topic0:7.282e-04 topic1:5.202e-05 topic2:2.803e-04 topic3:2.203e-04 topic4:2.853e-05...
(D) Topic representation for short texts in online BTM

Fig. 2. An example of online BTM based on short text expansion.

frequency occurring in the texts. Thus, we specify that the representative words in a topic with probability in interval [0.5, 1) will appear 4 times, [0.2, 0.5) and [0.1, 0.2) will appear 3 times and twice respectively, and [0.07, 0.1) will appear once. In addition, each short text selects N_w (eg. N_w =5) representative words to expand, and these words are from no more than N_t (eg. N_t=3) topics. Correspondingly, we can get expanded short texts with representative words in topics, accordingly we get the expanded data chunk $\{d'_j\}_{j=1}^3$ as shown in Fig. 2-(C), details of which in the same topic are omitted.

Online Biterm Topic Model: After achieving the expansion of data chunks, we utilize the online BTM to obtain the representation of each short text in the expanded data chunks. Online BTM is a topic model for short text streams based on biterms. For the data in the tth time slice, its generative process is described as follows.

1. Draw a topic distribution in the tth time slice $\theta^{(t)} \sim Dirichlet(\alpha^{(t)})$;
2. For each topic k from K topics:
   (a) Draw a word distribution of topic in the tth time slice $\phi_k^{(t)} \sim Dirichlet(\beta_k^{(t)})$;
3. For each biterm $b_j = \{w_{j,1}, w_{j,2}\} \in B^{(t)}$:
   (a) Draw a topic $z_j \sim Multinomial(\theta^{(t)})$;
   (b) Draw words $w_{j,1}, w_{j,2} \sim Multinomial(\phi_{z_j}^{(t)})$.

In this generative process, $B^{(t)}$ indicates the biterm set in the tth time slice and the size is $|B^{(t)}|$, where $B^{(t)} = \{b_j\}_{j=1}^{|B^{(t)}|}$, $\alpha^{(t)}$ and $\beta_k^{(t)}$ are Dirichlet hyper-parameters for $\theta^{(t)}$ and $\phi_k^{(t)}$, where $\alpha^{(t)} = \{\alpha_k^{(t)}\}_k^K$ is a K-dimensional multinomial distribution and $\beta_k^{(t)} = \{\beta_{k|w}^{(t)}\}_W^W$ is a W-dimensional multinomial distribution. And symmetric Dirichlet distribution is used to set their initial prior $\alpha^{(1)} = (\alpha, \ldots, \alpha)$ and $\beta^{(1)} = (\beta, \ldots, \beta)$. The graphic representation of online BTM in the tth time slice is shown in Fig. 3.

After the above processing in the tth time slice, we can obtain the number of biterms in each topic k (denoted as $n_k^{(t)}$) and the number of times that each word w is assigned to topic k (denoted as $n_{w|k}^{(t)}$). To obtain the hyper-parameters $\alpha^{(t+1)}$ and $\{\beta_k^{(t+1)}\}_{k=1}^K$ for the $(t+1)$th time slice, we utilize the above $n_k^{(t)}$ and $n_{w|k}^{(t)}$ to calculate as $\alpha_k^{(t+1)} = \alpha_k^{(t)} + \lambda n_k^{(t)}$ and $\beta_{k|w}^{(t+1)} = \beta_{k|w}^{(t)} + \lambda n_{w|k}^{(t)}$ respectively. Therefore, for the data in the $(t+1)$th time slice, its generative process is similar with the tth time slice with new the hyper-parameters $\alpha^{(t+1)}$ and $\{\beta_k^{(t+1)}\}_{k=1}^K$.

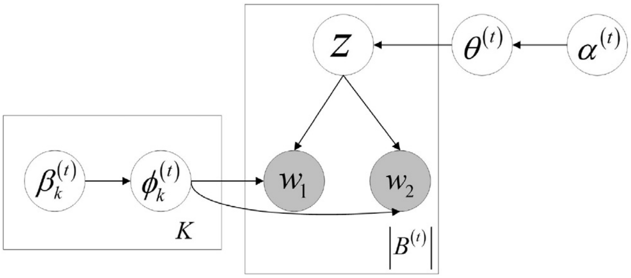


Fig. 3. Graphic representation of online BTM in the tth time slice.

Furthermore, suppose d contains N_b biterms $\{b_j\}_{j=1}^{N_b}$, the topic z for a short text d is calculated as $P(z \mid d) = \sum_{j=1}^{N_b} P(z \mid b_j^{(d)}) P(b_j^{(d)} \mid d)$, for more details please refer to the work in [5]. Therefore, each short text can be represented by the topic distribution using the above calculation.

In our approach, we exploit online BTM to infer the topic distribution for each short text in the expanded data chunk. Thus, each expanded data chunk D'_i can be represented by the topic distribution, denoted as $D''_i = \{d''_j\}_{j=1}^{|D_i|}$ with $d''_j = \{z_{j,k}\}_{k=1}^K$. For instance, expanded short texts $\{d'_j\}_{j=1}^3$ in Fig. 2(C) are represented into a set of topics $\{d''_j\}_{j=1}^3$ in Fig. 2(D).

2.3. Concept drifting detection

In short text streams, topics appear potential drifts over time, called concept drifts here. It seriously affects the classification performance, thus how to detect concept drifts becomes a critical issue. To solve this problem, we propose a concept drifting detection method based on the topic distribution. We determine whether concept drifts occur or not by accumulating the sum of the semantic distance between each short text (denoted as d''_j) in the next data chunk (denoted as D''_{i+1}) and the current data chunk (denoted as D''_i) as shown in Eq. (2).

$$dist\left(D''_{i+1}, D''_i\right) = 1/|D_{i+1}| \sum_{j=1}^{|D_{i+1}|} dist\left(d''_j, D''_i\right) \quad (2)$$

To compute the semantic distance in Eq. (2), we divide the current data chunk D''_i into class clusters $\{I_c\}_{c=1}^C$ according to the label distribution, denoted as $I_c = \{d''_l\}_{l=1}^{|I_c|}$, where $|I_c|$ is the number of texts in cth class label. Then we calculate the semantic distance between the short text d''_j and all class clusters $\{I_c\}_{c=1}^{C}$, and finally use the minimum semantic distance to represent the distance between short text d''_j and the current data chunk D''_i as shown in Eq. (3). To reduce the impact from a class cluster with fewer short texts, we add a weight in the calculation between short texts d''_j and d''_l as shown in Eq. (4).

$$dist\left(d''_j, D''_i\right) = \min dist\left(d''_j, I_c\right) \tag{3}$$

s.t., $dist(d''_j, I_c) = 1/|I_c| \sum_{l=1}^{|I_c|} dist(d''_j, d''_l), d''_l \in I_c$

$$dist\left(d''_j, d''_l\right) = 1 - |I_c|/|D_i| \cos\left(d''_j, d''_l\right) \tag{4}$$

s.t., $\cos(d''_j, d''_l) = (z_{j,1}.z_{l,1} + \ldots + z_{j,K}.z_{l,K})/(\sqrt{\sum_{k=1}^{K}(z_{j,k})^2}.\sqrt{\sum_{k=1}^{K}(z_{l,k})^2})$

In this paper, a threshold μ is used to detect concept drifts. If $dist(D''_{i+1}, D''_i) \in (\mu, 1]$, we consider there is a concept drift hidden in the $(i+1)$th data chunk.

2.4. Ensemble model building and prediction

To predict a yet-to-come short text, we select H expanded data chunks mentioned above to build H classifiers respectively. We use SVM as the base classifier, because it is popularly used in the text classification. When a new data chunk D_e arrives, we first expand D_e with the external corpus as D'_e, and then represent D'_e as a group of topics $D''_e = \{d''_j\}_{j=1}^{|D_e|}$ in the above method. Furthermore, to obtain the weight of short text d''_j namely $w_{h,j}$, relative to base classifier f_h built using the data chunk D''_h, we calculate the semantic distance between short text d''_j in the D''_e and data chunk D''_h, namely $dist(d''_j, D''_h)$, and the semantic distance $dist(D''_e, D''_h)$. Therefore, the weight $w_{h,j}$ can be calculated by Eq. (5).

$$w_{h,j} = \left(1 - dist\left(d''_j, D''_h\right)\right) * \left(1 - dist\left(D''_e, D''_h\right)\right) \tag{5}$$

where $1 - dist(d''_j, D''_h)$ indicates the semantic similarity between short text d''_j in D_e and data chunk D''_h. And $1 - dist(D''_e, D''_h)$ indicates the semantic similarity between the new data chunk D_e and data chunk D''_h, which is defined to reduce the impact in the classification using those classifiers with hidden concept drifts. Finally, in terms of ensemble model E, we can predict each short text in D_e using Eq. (1).

To update the ensemble model E, we compute the semantic distance between the new data chunk D''_e and each data chunk used in E to detect concept drifts, while the new data chunk is used to build classifier f. If there is a concept drift between D''_e and each data chunk in E, and the number of classifiers on E is less than H, the classifier f will be added into the ensemble model E. If the number of classifiers on E is equal to H, the classifier f is used to replace the oldest classifier in E. Otherwise, the classifier f is used to replace the classifier built the data chunk where semantic distance is the smallest. Algorithm 1 summarizes the process of the proposed method.[3]

Algorithm 1 Our approach.

```
Input:
  String data $D$, the LDA topic model $M_{LDA}$, the ensemble model
  $E$, the set of $H$ data chunks used for the ensemble model $S$, the
  threshold $\mu$;
Output:
  updated $E$, predicted labels in data chunk $D_e$;
for each data chunk $D_e$ in $D$ do
    Expand $D_e$ with the topic model $M_{LDA}$ as $D'_e$;
    Represent expanded $D'_e$ as $D''_e$ using online BTM;
    for $D''_i$ in $S$ do
        for $d''_j$ in $D''_e$ do
            Calculate semantic distance between $d''_j$ and $D''_i$ using
            Eq. (3);
        end for
        Calculate semantic distance between $D''_e$ and $D''_i$ using Eq.
        (2);
    end for
    for $d''_j$ in $D''_e$ do
        Calculate weights using Eq. (5);
        Predict $d''_j$ according to ensemble model $E$ using Eq. (1);
    end for
    Detect concept drifts between $D''_e$ and each data chunk in $S$
    according to the threshold $\mu$;
    Build a new classifier $f$ on $D''_e$ and update ensemble model
    $E$ according to results of concept drifts;
end for
```

3. Experiments

We first introduce benchmark data sets of short texts used in our experiments, and then give details of baseline methods and evaluation measures.

Data sets: 1) Snippets[4] [15]: Snippets was selected from results of web search transaction using predefined phrases of different domains. And it contains eight categories with 12,340 snippets. 2) News: News comes from TagMyNews Dataset.[5] And it has seven categories. Each new contains a short title, a short description, a link, and so on. In our experiment, we only extract short titles as the data set with 32,597 records. 3) Tweets [24]: It contains four categories required during Nov. and Dec. in 2012 via Twitters keywords tracking API.[6] In our experiment, we only select tweets on December 2012 as the data set with 803,613 records. To simulate the concept drift, we reorganise two datasets of Snippets and News. Each dataset has random concept drifts whose drifting periods are set to 100 to 700 instances. Moreover, we set the r% noisy for the data set in each topic, in which the value of r% is set to 5% here.

Baseline approaches: In our experiments, we evaluate the effectiveness of our algorithm in two ways. First, we compare our approach with four classical data stream classification algorithms in the classification performance. We call our approach with short text expansion and concept drifting detection as OurE.Drift. To validate the effectiveness of the short text expansion, we also use our approach with only concept drifting detection as the baseline one, called Our.Drift. Second, we compare our concept drift detection method with five excellent ones in the performance of concept drifting detection. Details of all comparison algorithms are listed in Table 1. All base classifiers and concept drifting detection algorithms are from the open source platform of MOA [1].

Evaluation measures: We introduce incremental accuracy to evaluate the data streams classification algorithm. Because it is more suitable for data streams classification [11]. Moreover, we discuss the impact of the number of base classifiers on the classification performance. In the concept drift detection, we use the prequential evaluation using fading factor (e.g.,0.995) to investigate the effectiveness of our concept drifts in short text stream classification [8]. Meanwhile, we introduce three evaluation measures for

[3] Our source code is available at https://github.com/HayaWang https://github.com/HayaWang/OurE.Drift.

[4] http://jwebpro.sourceforge.net/data-web-snippets.tar.gz.

[5] http://acube.di.unipi.it/tmn-dataset.

[6] https://dev.twitter.com/docs/api/1.1/post/statuses/filter.

Table 1
Competing methods.

Category	Approach	Description
Data Stream Classifiers	Naive Bayes	A simple classifier known for its simplicity and low computational cost.
	Spegasos [18]	Implements the stochastic variant of the Pegasos method.
	KNN+PAW+ADWIN	K Nearest Neighbor adaptive with Probabilistic Approximate Window and Adaptive sliding window.
	HoeffdingOptionTree [14]	Decision option tree for streaming data.
Topic Drifting Detectors	DDM [7]	Drift detection method.
	CusumDM [17]	Drift detection based in Cusum.
	PageHinkleyDM [8]	Drift detection method based in Page Hinkley Test
	HDDM_A_Test [6]	Online drift detection method based on Hoeffding's bounds.
	HDDM_W_Test [6]	Online drift detection method based on McDiarmid's bounds.

statistics, including (1) False Alarm: The rate that false alarms occur in the concept drifting detection; (2) Missing: The rate of concepts missed in the drifting detection; (3) Delay: The mean count of short texts required to detect a concept drift after the occurrence of a concept drift.

All experiments are performed on an Intel Core i5 2.90 GHz CPU and 8G physical memory. According to the experience, hyperparameters for two topic models α and β are set to 0.5 and 0.01 respectively, the number of topics for two topic models is set to 50, decay weight λ is set to 0.5, and we run 1000 iterations. In addition, the chunk size is set to 50 on two artificial data sets, and it is set to 1000 for Tweets, because it is too large. For concept drift detection, we set the threshold μ to $1 - 0.5/C$, where C indicates the number of class labels. Finally, we select SVM from libsvm[7] as the base classifier, and the corresponding parameter settings are set to C-SVC as the type and linear kernel as the kernel function while the rest ones are set to default values.

3.1. Experimental results

We first evaluate the performance in the short text stream classification, and then we illustrate the effectiveness of concept drift detection.

Classification performance: Fig. 4 reports curves of incremental accuracy on three data sets.[8] We can get the following observations. 1) OurE.Drift presents the higher prediction accuracy than Our.Drift by 8% on average for Snippets and News, and 4% for Tweets. These data show the short text expansion used in OurE.Drift is effective. 2) OurE.Drift almost beats all competing algorithms, and OurE.Drift is more stable than other competing ones. Because all competing ones directly build an incremental classifier model over short texts and do not consider the sparseness of short texts. But OurE.Drift presents a new online BTM based on the short text expansion to produce the lower sparsity and higher semantic information and meanwhile to alleviate dimensions on short text streams. Moreover, to adapt to the short text streams with concept drifts, we use semantic distances between the current data chunk and data chunks in ensemble model to assign weights for each base classifier, and predict the current data chunk. And the ensemble model is always updated with the latest data chunk.

Fig. 5 reports the impact of the number of base classifiers varying from 2 to 15 on the incremental accuracy. From the experimental results we can see that as the number of base classifiers increases, the incremental accuracy on Snippets and Tweets presents increasing and then maintains stable after the number of base classifiers is equal to or larger than 10, while the experimental result on News presents decreasing. This is because the larger the number of base classifiers, the more diverse the prediction results, however, the feature space of News is more sparse than other data sets and there are some overlapping features occurring in the different categories, which causes to the lower incremental accuracy.

In addition, Fig. 6 compares the efficiency of OurE.Drift and other competing ones. From these experimental results, we can observe the followings. Firstly, OurE.Drift is faster than KNN+PAW+ADWIN on News and Tweets, but it performs worse on Snippets. The reason is analyzed below. KNN+PAW+ADWIN introduces the mechanisms of probabilistic approximate window and the adaptive sliding window for adapting to concept drifts. When the concept drift occurs, it requires updating K nearest neighbors frequently. In our approach, online BTM costs time $O(N_{iter}K|B^{(i)}|)$ with $\left|B^{(i)}\right| \approx \left|M'\right|\bar{l}(\bar{l}-1)/2$, where N_{iter} indicates the number of iterations, $|M'|$ denotes the number of documents, and $\bar{l}$ denotes the average length of documents. The average length of short texts on Snippets is more twice than that in News and Tweets, thus, our approach spends more time on online BTM compared to KNN+PAW+ADWIN. Second, our approach is lower than the competing ones. This is because in our approach, we reduce data dimensions to improve the time to build the ensemble model, but we spend too much time on expanding short texts with LDA and obtaining topic representation with online BTM, while other competing ones only require building classifiers using the original short texts.

Concept drift detection: In this subsection, we first give the prequential error evaluation in OurE.Drift and KNN+PAW+ADWIN with the best performance in the baselines. And then we report the statistics of our concept drift detection approach and all baselines.

Fig. 7 provides the classification results with prequential error evaluation. According to the experimented results, we can observe that 1) if concept drifts occur, our approach can achieve lower prequential error than KNN+PAW+ADWIN. This is because OurE.Drift is built on multiple base classifiers using different data chunks. It contains some concepts occurring which is beneficial to adapt to concept drifts. 2) Our approach can recover from each concept drift earlier than KNN+PAW+ADWIN. The reason is that we can quickly update the ensemble model with the new classifier built by the newest data chunk if detecting concept drifts.

Table 2 shows the statistics of our concept drift detection approach and competing ones. All competing approaches based on the classification error rates select best results on different classifiers mentioned above. And the number in right parentheses is used to represent the ranking of the corresponding approach. Furthermore, the best result is in bold. And we can draw the following conclusions. 1) For News and Tweets, our approach can beat all competing methods on evaluation measures of Delay and Missing. Because we enrich each short text to alleviate sparseness and detect concept drifts based on the topic distribution, thus, our approach is more sensitive for concept drift. HDDM_A/W_Test perform better than other approaches on the evaluation measure of Missing in Snippets. This is because the non-weighted or weighted statistics are used as the estimator in the Hoeffding

[7] https://www.csie.ntu.edu.tw/~cjlin/libsvm/ .

[8] HoeffdingOptionTree cannot handle Tweets due to the huge memory consumption.

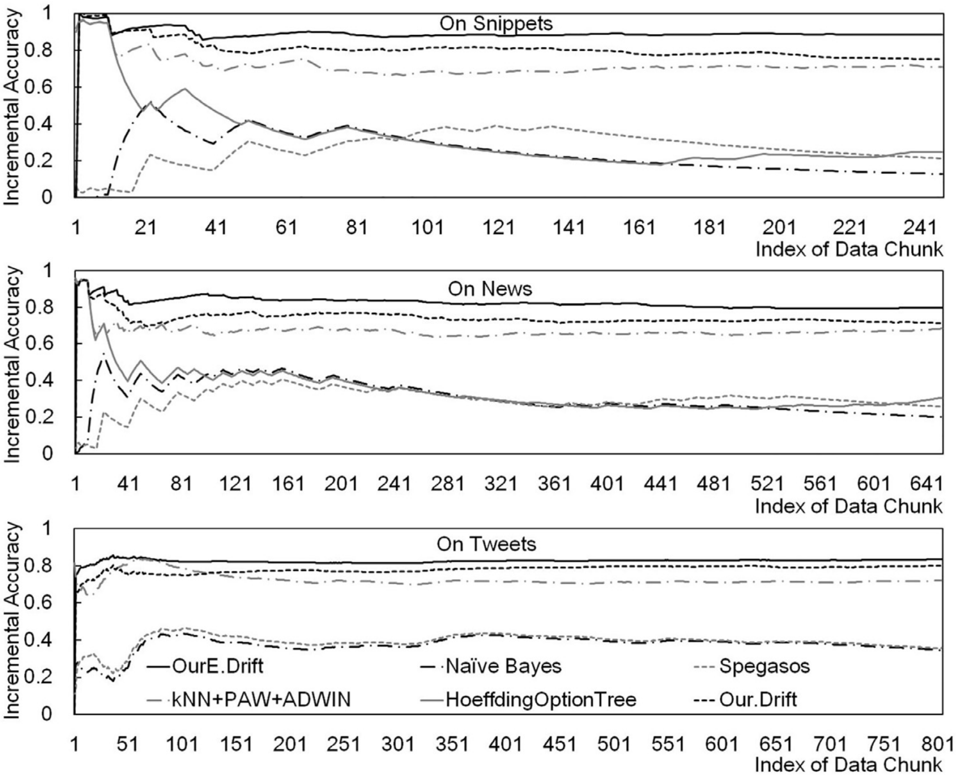


Fig. 4. Accuracy comparison between data stream classification methods and ours.

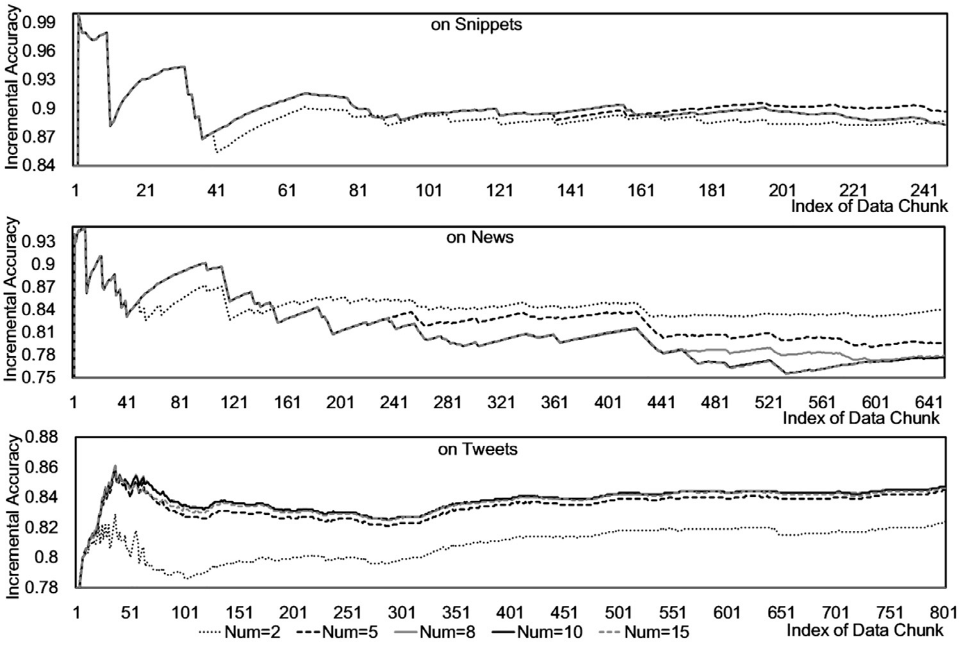


Fig. 5. Impact of the number of base classifiers on incremental accuracy.

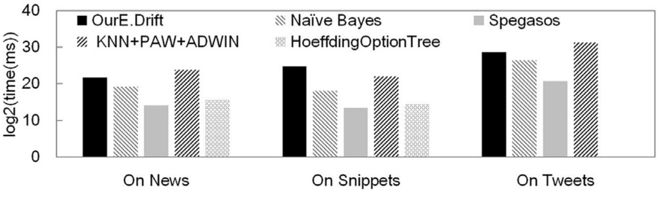


Fig. 6. Execution time of all approaches.

bounds computation, it demands more statistical information of instances while each short text on Snippets has more semantic information. 2) PageHinkleyDM always holds the lowest error rate, but it is built on the premise that at least 90% of topic drifts are missing in the detection. This is because the sparsity and the high dimension of short texts make it difficult to distinguish the difference between observed values and their means.

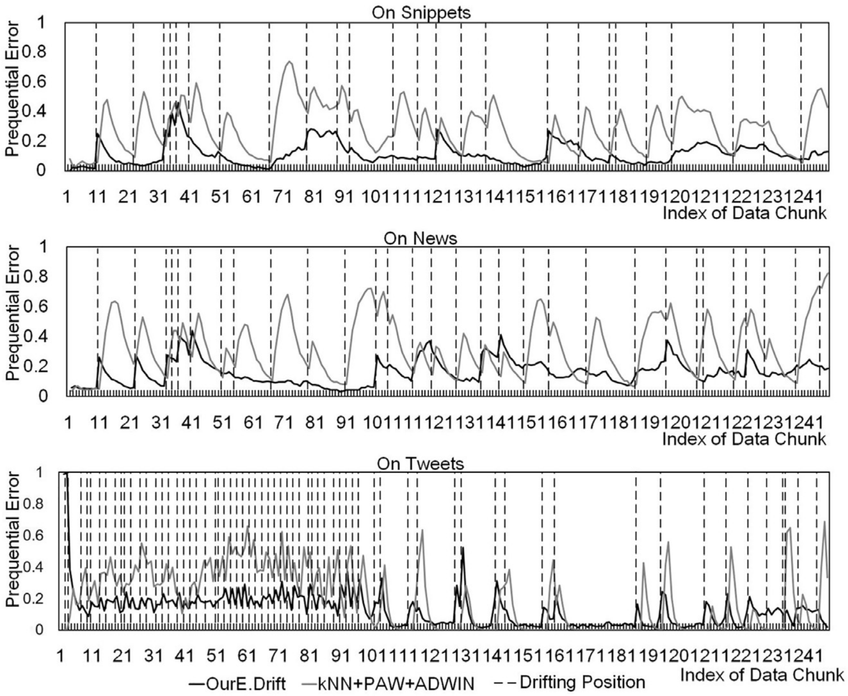


Fig. 7. Drift detection between OurE.Drift and KNN+PAW+ADWIN.

Table 2
Drifting detection statistics (FA: False Alarm, Miss: Missing).

Measures	DDM	Cusum-DM	PageHinkleyDM	HDDM_A_Test	HDDM_W_Test	**Ours**
			Snippets			
FA(%)	5.26(3)	6.25(4)	**0(1)**	7.69(5)	4.17(2)	**0(1)**
Miss(%)	28(4)	37.5(5)	96.0(6)	**4.0(1)**	8.0(2)	12(3)
Delay	32.89(4)	64.47(5)	122.0(6)	9.625(2)	11.43(3)	**0(1)**
			News			
FA(%)	**0(1)**	**0(1)**	**0(1)**	1.79(2)	1.79(2)	8.75(3)
Miss(%)	64.18(4)	62.69(3)	94.03(5)	17.91(2)	17.91(2)	**6.49(1)**
Delay	37.5(4)	82.32(5)	243.0(6)	7.93(2)	12(3)	**0(1)**
			Tweets			
FA(%)	3.79(3)	1.98(2)	**1.12(1)**	5.20(5)	7.88(6)	5.13(4)
Miss(%)	56.60(3)	62.57(5)	96.54(6)	57.96(4)	52.15(2)	**16.94(1)**
Delay	133.72(3)	131.23(2)	468.92(6)	166.59(4)	158.82(5)	**1.92(1)**

4. Related work

In this section, we briefly summarize the related work on short text classification in the following two dimensions.

One is to expand the short text with external corpora. Representative works are below. Phan et al. [15] introduced a hidden topic based framework for enriching short texts. With the help of LDA, [3] combined data enrichment with the introduction of semantics in Random Forest to improve short text classification. Vo and Ock [22] presented a new method for combining external texts from adapted topics to improve features in short texts. Zhang and Zhong [25] viewed topics of words generated by the topic model as new words and integrated into texts for data enriching. Li et al. [12] proposed a novel topic classification based on entity knowledge base and topic enhanced word embedding. However, the above methods need to define the number of topics in advance. If the number of selected topics is not appropriate, it leads to the poor classification. Furthermore, there are other ways to extend the short texts. For example, Li et al. [11] introduced more semantic contexts based on the senses of terms in short texts to alleviate the data sparsity. Wang et al. [23] proposed a novel method to expand short texts based on word embedding cluster and convolutional neural network. Liu et al. [13] built the semantic relevant concept sets of Wikipedia firstly, and then used Word2Vec to measure the semantic similarity between concepts, and disambiguated terms by their semantics to reduce the noise impact. Sun and Zhao [19] presented a novel feature extension method based on Topical N-Grams model to solve the feature sparseness problem. Tang et al. [20] proposed a novel end-to-end deep memory network approach which was used to automatically find relevant information from long documents and reformulate the short text through a gating mechanism for short text expansion. These ways alleviate the sparseness of short text and improve the classification accuracy. But it is extremely difficult to obtain the appropriate external corpus which covers all dataset domains to be classified.

The other dimension is to construct the classification model by using hidden rules and statistics information. Representative works are summarized below. To alleviate the sparseness of short texts, Rao et al. [16] proposed a topic-level maximum entropy (TME) model for social emotion classification on short texts. Kim et al. [10] proposed a novel kernel, called language independent semantic kernel, which was designed to effectively classify short-text documents without using grammatical tags and lexical databases. Gao et al. [9] introduced the structured sparse representation to short text classification and proposed an effective method called convex hull vertices selection to reduce the data correlation and redundancy.

In sum, the aforementioned classification approaches almost belong to batch algorithms, which hardly deal with the data stream classification. Meanwhile, they also miss the facts of concept drifts hidden in short text streams.

5. Conclusion

We proposed a new short text stream classification approach based on online BTM with short text expansion and concept drifting detection. We first enrich each short text with an external corpus to alleviate the sparseness and utilize online BTM to gain the topic representation of each expanded short text to decrease the high dimension. Second, to detect concept drifts, we presented a new topic distribution-based detection method. Meanwhile, we build an ensemble model using multiple base classifiers, and calculate the semantic distances between the new data chunk and data chunks in the ensemble model as the weights for each base classifier to predict the new data chunk. We update this ensemble model using the newest data chunk in terms of concept drifting results. Finally, extensive results showed that as compared to the classic approaches, our approach can achieve a better performance in the short text stream classification and concept drifting detection. However, there is still some room to improve the time consumption in our work.

Acknowledgments

This work is supported in part by the National Key Research and Development Program of China under grant 2016YFC0801406, the Program for Changjiang Scholas and Innovative Research Team in University (PCSIRT) of the Ministry of Education under grant IRT17R32, the Natural Science Foundation of China under grants (61503112, 61673152, 91746209), and the Natural Science Foundation of Anhui province under grant 1708085QF142.

References

[1] A. Bifet, G. Holmes, R. Kirkby, B. Pfahringer, Moa: massive online analysis, J. Mach. Learn. Res. 11 (2) (2010) 1601–1604.

[2] D.M. Blei, A.Y. Ng, M.I. Jordan, Latent dirichlet allocation, J Mach. Learn. Res. Arch. 3 (2003) 993–1022.

[3] A. Bouaziz, C. Dartigues-Pallez, C.D.C. Pereira, F. Precioso, P. Lloret, Short Text Classification Using Semantic Random Forest, Springer International Publishing, 2014.

[4] L. Breiman, Random forests, Mach. Learn. 45 (1) (2001) 5–32.

[5] X. Cheng, X. Yan, Y. Lan, J. Guo, Btm: topic modeling over short texts, IEEE Trans. Knowl. Data Eng. 26 (12) (2014) 2928–2941.

[6] I. Frias-Blanco, J.D. Campo-Avila, G. Ramos-Jimenez, R. Morales-Bueno, Online and non-parametric drift detection methods based on hoeffdings bounds, Knowl. Data Eng. IEEE Trans. 27 (3) (2015) 810–823.

[7] J. Gama, P. Medas, G. Castillo, P. Rodrigues, Learning with drift detection, in: Advances in Artificial Intelligence - Sbia 2004, Brazilian Symposium on Artificial Intelligence, So Luis, Maranho, Brazil, September 29 - October 1, 2004, Proceedings, 2004, pp. 286–295.

[8] J. Gama, R. Sebastio, P.P. Rodrigues, On evaluating stream learning algorithms, Mach. Learn. 90 (3) (2013) 317–346.

[9] L. Gao, S. Zhou, J. Guan, Effectively classifying short texts by structured sparse representation with dictionary filtering, Inf. Sci. 323 (2015) 130–142.

[10] K. Kim, B.S. Chung, Y. Choi, S. Lee, J.Y. Jung, J. Park, Language independent semantic kernels for short-text classification, Expert Syst. Appl. 41 (2) (2014) 735–743.

[11] P. Li, L. He, X. Hu, Y. Zhang, L. Li, X. Wu, Concept based short text stream classification with topic drifting detection, in: IEEE International Conference on Data Mining, 2016, pp. 1009–1014.

[12] Q. Li, S. Shah, X. Liu, A. Nourbakhsh, R. Fang, Tweetsift: Tweet topic classification based on entity knowledge base and topic enhanced word embedding, in: ACM International on Conference on Information and Knowledge Management, 2016, pp. 2429–2432.

[13] W. Liu, Z. Cao, J. Wang, X. Wang, Short text classification based on wikipedia and word2vec, in: IEEE International Conference on Computer and Communications, 2017, pp. 1195–1200.

[14] B. Pfahringer, G. Holmes, R. Kirkby, New Options for Hoeffding Trees, Springer Berlin Heidelberg, 2007.

[15] X.H. Phan, C.T. Nguyen, D.T. Le, L.M. Nguyen, S. Horiguchi, Q.T. Ha, A hidden topic-based framework toward building applications with short web documents, IEEE Trans. Knowl. Data Eng. 23 (7) (2011) 961–976.

[16] Y. Rao, H. Xie, J. Li, F.L. Wang, F.L. Wang, Q. Li, Social emotion classification of short text via topic-level maximum entropy model, Information & Management 53 (8) (2016) 978–986.

[17] M. Severo, Change detection with kalman filter and cusum, in: International Conference on Discovery Science, 2006, pp. 243–254.

[18] S. Shalev-Shwartz, Y. Singer, N. Srebro, A. Cotter, Pegasos: primal estimated sub-gradient solver for svm, Math. Program. 127 (1) (2011) 3–30.

[19] B. Sun, P. Zhao, Feature extension for chinese short text classification based on topical n-grams, in: IEEE/ACIS International Conference on Computer and Information Science, 2017.

[20] J. Tang, Y. Wang, K. Zheng, Q. Mei, End-to-end learning for short text expansion, in: The ACM SIGKDD International Conference, 2017, pp. 1105–1113.

[21] V.N. Vapnik, An overview of statistical learning theory, IEEE Trans. Neural Netw. 10 (5) (1999) 988–999.

[22] D.T. Vo, C.Y. Ock, Learning to classify short text from scientific documents using topic models with various types of knowledge, Pergamon Press, Inc., 2015.

[23] P. Wang, B. Xu, J. Xu, G. Tian, C.L. Liu, H. Hao, Semantic expansion using word embedding clustering and convolutional neural network for improving short text classification, Neurocomputing 174 (PB) (2016) 806–814.

[24] Z. Wang, L. Shou, K. Chen, G. Chen, S. Mehrotra, On summarization and timeline generation for evolutionary tweet streams, IEEE Trans. Knowl. Data Eng. 27 (5) (2015) 1301–1315.

[25] H. Zhang, G. Zhong, Improving short text classification by learning vector representations of both words and hidden topics, Knowl. Based Syst. 102 (2016) 76–86.