Applied Soft Computing 108 (2021) 107440

Contents lists available at ScienceDirect

Applied Soft Computing

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

Topic-level sentiment analysis of social media data using deep learning

Ajeet Ram Pathak *, Manjusha Pandey, Siddharth Rautaray

School of Computer Engineering, Kalinga Institute of Industrial Technology (KIIT) University, Bhubaneswar, India

ARTICLE INFO

Article history:
Received 4 May 2020
Received in revised form 28 March 2021
Accepted 14 April 2021
Available online 24 April 2021

Keywords:
Deep learning
Topic detection
Social media data
Topic modeling
Topic-level sentiment analysis
Attention mechanism
Streaming data
Out-of-domain sentiment analysis

ABSTRACT

Due to the inception of Web 2.0 and freedom to facilitate the dissemination of information, sharing views, expressing opinions with regards to current world level events, services, products, etc. social media platforms have been mainly contributing to user-generated content. Such social media data consist of various themes discussed online and are associated with sentiments of the users. To catch up with the speed of streaming data at which it generates on social media platforms, it is crucial to detect the topics being discussed on social media platforms and analyze the sentiments of users towards those topics in an online manner to make timely decisions. Motivated by the same, this paper proposes a deep learning based topic-level sentiment analysis model. The novelty of the proposed approach is that it works at the sentence level to extract the topic using online latent semantic indexing with regularization constraint and then applies topic-level attention mechanism in long short-term memory network to perform sentiment analysis. The proposed model is unique in the sense that it supports scalable and dynamic topic modeling over streaming short text data and performs sentiment analysis at topic-level. For SemEval-2017 Task 4 Subtask B dataset as a case of in-domain topic-level sentiment analysis, average recall of 0.879 has been achieved, whereas, for out-of-domain data, average recall of 0.846, 0.824 and 0.794 has been achieved for newly developed datasets collected under the hashtags #ethereum, #bitcoin and #facebook from Twitter. To assess the performance of the model for scalability, we analyzed the model in terms of average time in milliseconds for creation of feature vectors, throughput in terms of topics detected per second and average response time in seconds to handle the sentiment analysis queries. The experimental results are significant enough to enable large scale topic modeling over streaming data and perform topic-level sentiment analysis.

1. Introduction

Due to the ubiquitous nature of social media platforms and the continuous availability of network connection to access the Internet, more than 2 billion users, i.e., about 35% of the world's population, are now social media users [1]. Social networking sites (Facebook, LinkedIn, Google+), microblogging sites (Twitter, Tumblr), photo sharing platforms (Snapchat, Pinterest, Instagram), and video sharing platforms (YouTube, Vimeo, Facebook Live) are serving as the primary contributors and valuable sources of big data. For instance, everyday Facebook handles more than 500 TB of data [2]. On average, about 6000 tweets are tweeted on Twitter. This means around 350,000 tweets hit Twitter per minute generating 500 million tweets per day and approximately 200 billion tweets per year [3].

A large amount of social data contains tweets, blogs, and reviews from multiple domains, which pose numerous challenges and opportunities to the researchers of natural language processing (NLP) to discover useful information. Moreover, such data enable us to interpret the aspects of people on a specific topic and exhibit information that can be leveraged to perform prediction in the domain of product sales, stock market, political elections, etc. For taking the decisions in a timely manner based on people's reviews, sentiment analysis is considered as one of the crucial tasks from business perspectives, and it is highly sought-after research domains. Sentiment analysis is the study of analyzing people's sentiment expressed towards services, products, mandates, organizations, etc. [4]. Many domains such as gaming, business sectors, retail, and advertising industries, healthcare organizations have started to adopt sentiment analysis systems to monitor their social image, and therefore, it is anticipated that the market of sentiment analysis and emotion recognition would reach $3.8 Billion till 2025 [5].

Research works till date focused on performing sentiment analysis at various levels of granularities such as document [6], sentence [7], aspect [8], topic [9], etc. to mention a few. Approaches proposed in [6–9] focuses only on single modality, i.e., textual data for inferring the sentiment.

* Corresponding author.
E-mail addresses: ajeet.pathak44@gmail.com (A.R. Pathak), manjushapandey82@gmail.com (M. Pandey), sr.rgpv@gmail.com (S. Rautaray).

https://doi.org/10.1016/j.asoc.2021.107440
1568-4946/

Recently few approaches based on cross-modal sentiment analysis have also been proposed [10–12]. Xu et al. [10] handled multi-modal data (texts, images, and social links) through hierarchical LSTMs. Similarly. Zhang et al. [11] considered cross-modal approach for capturing semantic connection between images and captions. Agarwal et al. [13] also proposed to apply RNN variants for multimodal sentiment analysis incorporating text, video and audio data. As an advancement to traditional sentiment analysis, there is also a trend of detecting emotions from videos i.e., affective video content analysis [14].

Though there exists substantial research in the domain of cross-modal sentiment analysis, this paper focuses on single modality (i.e., text), especially sentiments with respect to the topics detected.

Document-level sentiment analysis focuses on giving the overall polarity of a complete document or a paragraph. This level assumes that the document contains an opinionated text about the single entity only. Sentence level focuses on fine-grained analysis by extracting the sentiments from sentences but assumes the occurrence of a single entity only for sentiment analysis. It involves the tasks of subjectivity classification and polarity classification.

Aspect-based sentiment analysis (ABSA) is divided into four subtasks [15] *viz.* aspect term extraction, aspect term polarity, aspect category detection, and aspect category polarity. Aspect term extraction focuses on identifying the aspect terms present in the sentence. As an outcome of performing this task, a distinct list of aspect terms is returned. The next subtask in ABSA is determining the aspect term polarity which deals with finding out the polarity of all aspect terms within a sentence as positive, negative, neutral or conflict. Aspect category detection focuses on identifying the aspect categories from a predefined set of aspect categories. Such aspect categories are coarser than the aspect terms as mentioned in the first subtask. Finally, fourth subtask, namely, aspect category polarity, is used for determining the polarity of each aspect category as positive, negative, neutral or conflict. The application of aspect term extraction has been well justified by Guerrero et al. [16]. They proposed a solution for improved features to be considered by modeling the user behavior to personalize the recommendations and quantify the contribution of each aspect over the ratings of a set of reviews.

Few differences between aspect-based sentiment analysis and topic level sentiment analysis have been demystified here. In case of aspect term extraction, more than one aspect may be identified. On the contrary, we assume that a sentence possesses a single aspect (which we call as a topic). In the first phase of topic-level sentiment analysis i.e., topic detection, a topic from a sentence streaming from the social media platform [17–22] is identified. Once the topic is detected, a topic and a sentence containing that topic is fed as input to the model for finding the polarity of the topic as positive and negative [23–27]. In this way, topic-based sentiment analysis aims to give sentiments towards the topics mentioned in a sentence or the document. Basically, the proposed model of topic-level sentiment analysis merges research methodologies from 2 research domains, i.e., topic detection from streaming media and sentiment analysis. This paper focuses on topic-level sentiment analysis in which the topic is extracted from a sentence and then sentiment analysis with reference to the extracted topic and corresponding sentence is performed.

Motivation.

Need of maintaining the timeliness of the topics: Social media platforms enable users to express and discuss various worldwide events, services, products, cultural trends, government mandates, etc. Due to the vast number of topics discussed on social media platforms, it is also important to know what people are currently talking about and what are the sentiments of the people towards these topics. This is beneficial to both the end-users as well as government, industries, or firms. For example, reading the reviews of specific services used by one user would help other users whether to avail of the service or not. Considering the citizens' reaction towards a new mandate of government would help the government to make the modification or cancel the mandate or design a new mandate. Based on customers' reviews towards some product, the industry would be prompted either to increase the sales or improve the product and set the marketing strategies. However, after the occurrence of some event, due to the ubiquitous nature of social media platforms, information spreads rapidly and people start discussing the events. Due to this, social media platforms get flooded with streaming data. In case of extreme negative sentiments towards some events, the users may get violent. Therefore, it is very crucial to maintain the timeliness of the topics being discussed on social media platforms and know the sentiments of the users in a faster way.

Need of handing the short text: With advancements in Internet technology, the behavioral habits of the Internet users have drastically changed, and people started expressing their opinions and emotions on micro-blogging sites like Twitter, Facebook, etc. The salient features of such micro-blogging sites like maintaining the brevity (limitation on character length, for example, 280 characters in Twitter), ease of use, rapid dissemination leads to generation of large amounts of short texts. Therefore, analysis of such short text is of significant importance for understanding the sentiments of people. However, extracting the sentiments from such data is challenging. This is because data emanating from social media platforms is of streaming nature, and it is characterized by short text. For instance, the length of Tweets supported by Twitter is 280 characters. This brevity feature of Twitter allows the users to express the opinions using short text. Therefore, the issue of data sparsity arises due to occurrence of short text. Data sparseness and streaming data are the two main challenges while performing sentiment analysis. To handle streaming and short text data, the model should be scalable enough to perform sentiment analysis. Moreover, user-generated data on social media platforms are characterized by spelling and grammatical errors, slangs, or jargon and perhaps present in an unstructured format. These factors often pose a problem to the approaches relying on dictionaries, manually constructed lexicons, and knowledge bases, and due to this, an important aspect may miss out. On the contrary, unsupervised approaches are not affected by lexical forms and able to capture frequently occurred unknown phrases or words. Thus, they ensure that the salient topic mentioned in the data will be captured by the model. Moreover, due to the large variety of domains, services, and products reviewed by people, it is not practical to use supervised methods. Therefore, this paper uses an unsupervised learning approach for topic modeling and then performs sentiment analysis with reference to the topic being extracted from a sentence.

Motivated by these challenges, this paper proposes a deep learning model for sentiment-analysis. Specifically, the aim is to detect the topics in an online manner to handle streaming data in scalable manner and then perform topic-level sentiment analysis. For detection of topics from streaming data, online version of latent semantic indexing constrained by regularization has been proposed. Unlike traditional topic models like Latent Dirichlet Allocation or Probabilistic Latent Semantic Analysis which apply the generative technique on whole document, we perform topic modeling at sentence level. To find the sentiments of the detected topic, topic-level attention mechanism has been incorporated in long short-term memory network. Our assumption is that each sentence contains single topic since most of the text being commented is of short length.

The detailed description of scalability and online learning method of the proposed model is given below.

– Scalability of the proposed model: In real world, data generated by social media platforms contains the large number of terms and therefore, the scalability issue of topic modeling arises due to the requirement of simultaneously updating the term-topic matrix to fulfill the probability distribution assumption. For instance, the traditional latent semantic indexing model applies orthogonality constraint in its formulation and solves the problem using singular value decomposition (SVD). In case of large data, it is hard to parallelize the SVD. Therefore, to support scalability, we use regularization constraint (instead of orthogonality constraint) to apply deep learning based topic modeling approach based on online version of latent semantic indexing. We approximate the term-document matrix as a product of the term-topic matrix and topic-document matrix in which loss function is constrained by $\ell 1$ and $\ell 2$ norm.
– Online nature of the proposed model: For online working of the proposed model, sentence(s) emanating from social media platforms are processed in serial fashion. Deep learning based online latent semantic indexing model incrementally builds the topic model according to streaming data. Given a new sentence(s), topic vector of new sentence is predicted given the previously learned term-topic matrix. After this, the term-topic matrix is updated based on new sentence(s) and predicted topic vector. This way by applying online learning technique, the proposed model supports large dataset and support scalable topic modeling.

The contributions of the paper are as follows.

1. The proposed approach works adaptively for detection of topics at Sentence level from large scale data using online latent semantic indexing designed with the help of long-short term memory network
2. The model performs sentiment analysis of the detected topics using LSTM with topic-level attention mechanism
3. We put forth three datasets for topic-level sentiment analysis collected from Twitter under the hashtags #ethereum, #bitcoin and #facebook
4. Online learning makes the proposed approach scalable to support streaming data and supports out-of-domain sentiment analysis from sentences at topic-level

Novelty of the proposed methodology is stated as follows. Based on the contribution and the existing research work in this domain, the novelty of the paper can be stated as follows. Considering streaming nature of social media platforms which is characterized by short text, it is important to capture each sentence and immediately extract the topic from it instead of capturing the whole document in certain time span to infer topics from whole document. Concerning the literature survey, very less research has been conducted to address the task of detecting the topics from sentences in an online and dynamic manner. With reference to the proposed work, to perform sentiment analysis, we first aim to extract the topics from sentences and then perform sentiment analysis at topic-level. The proposed methodology is novel in the sense that we extract the topics from streaming sentences using online version of latent semantic indexing constrained by regularization and incrementally builds the topic model according to streaming data. We then find sentiments associated with the detected topics. Due to its online nature, the proposed approach is scalable and can get deployed in real-time environment. After extracting the topic, a sentence and a corresponding topic is fed as input to the sentiment analysis model. The model applies topic-level attention mechanism in LSTM network to get the sentiments associated with the topic in a sentence. The proposed model supports topic-level sentiment analysis with both in-domain and out-domain data.

The remaining portion of the paper is organized as follows. Section 2 deals with related work based on joint sentiment/topic models and the approaches performing topic modeling and sentiment analysis in a pipelined manner. Section 3 deals with proposed methodology for topic detection using online latent semantic indexing approach and topic-level sentiment analysis using attention-based long short-term memory network. Section 4 deals with experimentation details in terms of dataset developed, training and regularization strategies applied for deep learning models to perform topic detection and sentiment analysis. Results have been discussed is Section 5. Comparison with state-of-the-art approaches has been performed in Section 6. Section 7 gives the conclusion and pointers to future directions.

2. Related work

In this section, topic-level sentiment analysis methods have been thoroughly discussed. Topic-level sentiment analysis methods can be categorized into 3 methods based on the way how a topic is detected from the opinionated text and how sentiment analysis is performed. These categories can be mentioned as joint sentiment/topic model based methods, pipelined model based methods and sentiment analysis methods with the topic as input. Fig. 1 shows the flow of these different topic-level sentiment analysis methods. We synthesized the discussion of the state-of-the-art approaches along with the different aspects like short text topic modeling, online and dynamic topic modeling for large scale data, neural network based approaches for topic modeling to give background since the proposed approach supports online and dynamic topic modeling for short text data using deep learning based topic modeling and sentiment analysis.

2.1. Approaches based on joint sentiment/topic model

This category of joint sentiment/topic model focuses on jointly modeling topics and sentiments from documents [9,15,28,29]. Brody and Elhadad [29] proposed an aspect-sentiment model working in unsupervised mode. This model considers the influence of the aspect on sentiment and works at aspect extraction at the sentence level. For sentiment detection, it extracts unsupervised seed words of positive and negative adjectives replacing the method of manual construction of seed words. The issue of the sparseness of short text on social media has been handled in the work by Xu and Qi [30]. They leverage the structural information such as time slice, users, and hashtags for alleviating the context sparsity problem of short text. The supervised approach proposed by Fatemi and Safayani [31] modifies Restricted Boltzmann Machine (RBM) by appending it with the sentiment layer for joint sentiment/topic modeling. This generative modeling based neural network approach uses the Contrastive Divergence algorithm for training.

Dynamic topic-based sentiment analysis model [32] performs simultaneous dynamic detection of topics and associated sentiments by incorporating multiple timescale models in probabilistic generative model. This model updates sequentially and draws online inference using stochastic expectation–maximization algorithm. It alleviates errors caused by user interaction and adjusts model parameters in real-time. Dahal et al. [33] proposed to use Latent Dirichlet allocation for detecting different topics, whereas technique based on Valence Aware Dictionary and Sentiment Reasoner has been used for getting an overall sentiment of the dataset. They used it to check the sentiments of the people towards change in climate-related tweets. Fu et al. [34] applied

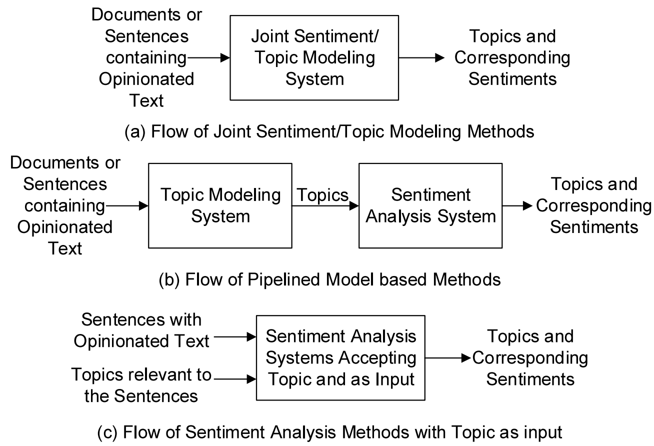


Fig. 1. Flow of topic-level sentiment approaches.

a weakly supervised learning model for joint modeling of sentiments and topics. For topic identification, they used a mixture of Dirichlet multinomial component and a word embeddings component. For sentiment recognition, this approach utilizes HowNet lexicon to find sentiment polarity of a word and then initializes sentiment value associated with the word. Chen and Parsons [35] proposed to model sentiments and topics of an opinionated text at the sentence level by recursively defining the topic distribution of a sentence. This approach follows the Sparse Gamma model, which uses the inference network for integrating language features and representing the sentiments.

Almars et al. [36] incorporated user attitudes to enhance the performance of sentiment analysis using a probabilistic generative model. Their Hierarchical User Sentiment Topic Model (HUSTM) hierarchically models opinions of the users with different sentiment and topic information. The Word-pair sentiment-topic model [37] is a joint topic-sentiment model. Unlike topic modeling approaches which models generative process of documents, their approach directly models the generation of word-pair set from large corpus. This model uses sentiment lexicon as a prior knowledge and applies LDA and Gibbs sampling for joint sentiment-topic modeling of product reviews. Nirmala and Jebakumar [38] handled the data sparsity problem of short text by generating pseudo document using biterms or word pair-set. Amplayo et al. [39] proposed an improvement to existing aspect term extraction models based on topic modeling. They presented 2 models that extend from Aspect Sentiment Unification Model (ASUM) differently to leverage on the information found in the product description. Fu et al. [40] proposed a weakly supervised topic sentiment joint model incorporating word embeddings and HowNet for improving topic identification and sentiment analysis. This approach works well for short text documents and sparse textual features. The problem of sparse text in short text reviews has been handled in [41]. In this approach, it is assumed that all words in a sentence possess sentiment polarity, and two words in a word-pair exhibit the same topic. This approach is applied to Chinese product reviews for joint topic detection and sentiment analysis. Garcia and Berton [42] performed sentiment analysis for both English and Portuguese Tweets generated in the pandemic period of COVID-19. Their approach is inspired by topic modeling and combination of word embedding vectors obtained from SBert, mUSE, FastText.

2.2. Pipelined model based methods

Pipelined model based methods focus on detecting the topics from documents followed by sentiment analysis. Considering the nature of pipelined model based methods, we first shed light on fundamental topic modeling methods and then the methods which first perform topic modeling and then sentiment analysis.

2.2.1. Topic modeling based methods

Fundamental topic models.
Due to its wide range of applicability in multitude of text mining and natural language processing tasks, significant research work has been done in topic modeling. Probabilistic topic models such as LDA [43], LSI [44], pLSA [45], etc. are used for analyzing and extracting latent topics from large collection of text dataset which work on the assumption that each document is posit as mixture of topics. Every topic is represented using probability distribution over words. Latent topic parameters in these graphical models have direct connections with observed parameters, which represent words in a document.

LDA is a widely preferred probabilistic generative model. Due to the intractable nature of exact inference method, LDA uses a convexity-based variational method based on approximate technique. LDA based topic models can be easily deployed in complex architectures due to their modular nature. LSI, on the other hand, does not support this property. LSI is based on SVD method and extracts implicit semantic structure in the document collection. PLSA is a modified version of latent semantic analysis model. It is based on statistical latent class model. However, these models suffer from the following major drawbacks. (1) Exact inference in such models is intractable and therefore, slow or inaccurate approximation techniques are required for computing the posterior distributions over topics. (2) These models cannot capture the essence of distributed representations.

Scalable topic models.
Online LDA proposed in [46] follows non-Markovian Gibbs sampling and maintains weight-matrix history in the generative process based on the homogeneity of domain. However, inter-topic differences are not handled by this model. Topic model put forth in [47] performs dynamic prediction of future trends for temporally sequenced data and supports scalable topic modeling. The conventional inference process requires multiple passes over the document for topic modeling. This inference process proves to be inefficient when topic modeling is applied over large scale data. Additionally, documents may contain large number of short text documents and it raises ambiguity in merely using words of documents for analyzing the data. To address these issues, Hennig et al. [48] patented a method for topic modeling that trains a model by sequential processing and incorporates to use features of documents (author of the document, document metadata, author location, etc.) for topic modeling. To support scalable topic modeling, the method of regularized latent semantic indexing has been patented in [49]. This method allows an equation involving approximation of the term-document matrix to be executed in parallel. For memory-limited environments, Kim et al. [50] proposed disk-based topic modeling approach, namely, BlockLDA, which applies space reduction technique and local scheduling technique for minimizing the disk I/O and supports scalable topic modeling even if data and model do not fit into the memory. Deep probabilistic autoencoder for topic modeling proposed in [51] supports scalable Bayesian inference on big data. This model is based on stochastic gradient Markov chain Monte Carlo (MCMC).

Topics models dealing with short text.
The text available on social media platforms is characterized by short length, and it becomes unreliable to extract topics from short texts using conventional topic models such as LDA and its variants due to severe data sparsity problem. An approach proposed in [52] utilizes global semantic correlation to improve the coherence among the topics and local semantic correlation for

word sense disambiguation and introduces conditional random field model for short text topic modeling. Performance of short text topic modeling may hinder due to data sparsity and presence of noisy words. The issue of noisy words has been alleviated in [53] by introducing common semantic topic model to capture semantics and noisy words. This topic model is based on mixture of unigram models. For extracting hierarchical latent representation from big collection of corpus, an inference method based on Weibull hybrid autoencoder for deep LDA has been proposed in [54]. This method factorizes high-dimensional count vectors using Poisson likelihood and uses Gamma likelihood for modeling latent representation. Sridhar [55] patented unsupervised topic model for short text. This method handles a collection of documents as a Gaussian mixture model in which topics are represented by Gaussian components. For extracting the topics from short text, posterior distribution over document topics is found with the help of the Gaussian mixture model. Ozyurt and Akcayol [56] proposed an aspect extraction method based on adaptation of LDA for short texts. They evaluated the performance of the model on SemEval-2016 Turkish dataset. Traditional topic models usually underperform while discovering topics from short text due to limitations like sparsity of features and lack of word co-occurrence patterns. Zhao et al. [57] addressed these limitations using variational Auto-encoder topic model. For enhancing the effectiveness in terms of performance, this model makes use of pre-trained word vectors and vector representations of entities in the knowledge graph. Data sparsity problem in short text is handled using fuzzy topic modeling approach in [58]. Specifically, bag-of-words model is used for computing local and global term frequencies whereas Principal component analysis is performed for ameliorating the impact of high dimensionality on global term weighing. Finally, fuzzy c-means clustering is used for extracting the semantically relevant topics. The issue of data sparsity in short text has also been handled in [59] using topic distribution quantization approach. Moreover, to avoid the retrieval of repetitive topics, negative sampling decoder based method is also proposed. To mitigate the effect of overfitting on probabilistic models like LDA, Ha et al. [60] proposed to use dropout regularization technique. For short text, dropout method worked effectively however, for long text, no significant benefits have been achieved.

Neural network based topic models.

Various deep learning based language models and dedicated topic models have been designed till date in the domain of text mining for efficient analysis. Salakhutdinov and Hinton [61] put forth the undirected graphical model, namely, replicated softmax, which works as a generative model of word counts. This model handles documents with varying lengths and it is used for computation of posterior distribution over latent topics and represents document as binary distribution. Replicated softmax is generalized form of RBM and uses bipartite graph for organizing binary observed and latent variables. This model achieves log-based probability over unseen documents with good accuracy. The drawback of Replicated softmax is that its learning rate of updates scales linearly with number of words encountered in an observed document i.e. size of the vocabulary.

Inspired from RBM, Neural Autoregressive Distribution Estimator (NADE) [62] works like autoencoder neural network, which accepts input as a vector of observations and outputs vector with same size. It works as a generative model by assigning valid probabilities to observations. NADE converts RBM into a tractable distribution estimator. It does not need symmetric connectivity as required in RBM i.e. weights incoming to and outgoing from hidden units are different. DocNADE [63] is an extension of NADE model [62]. This model has also been originated from RBM and uses conditional mean-field recursive equations of Replicated softmax to design feedforward Neural Network (NN) for predicting the probability of observing a new word given the prior probability of previously observed words. This model alleviates the need of computationally expensive softmax distribution by introducing hierarchical distributions in a binary tree of words. DocNADE considers the words present before the word to be predicted and does not consider the forward reference of the words for predicting a current word. Document informed topic model (iDocNADE) put forth in [64] is an extension of neural autoregressive model and considers the contextual information of the words for topic modeling using forward and backward language models.

Deep learning model based on convolutions has been patented in [65], which detects signals of interestingness in click transitions between source and target documents. In knowledge based question answering system [66], convolution neural network model is used for extracting the topics from questions and enable to get local context of the words. For modeling the distribution over the word count vectors in topic modeling, Srivastava and Salakhutdinov proposed to use two layers of deep Boltzmann Machine with replicated softmax function [67]. Shamanta et al. [68] handled the problem of finding the distributed representations of topic models in the same space as documents and words with the help of joint use of neural network and distributed representation generation technique. Yan et al. [69] formulated the problem of semantic indexing as multi-label classification problem and proposed hierarchical classification scheme based on convolution neural networks for semantic indexing. Li et al. [70] adopted an attention mechanism of neural network for learning the representation of microblog posts. They proposed topical co-attention network to model both content attention and topic attention for achieving the goal of hashtag recommendation. Ali et al. [71] proposed an ontology and latent Dirichlet allocation (OLDA)-based topic modeling and word embedding approach for sentiment classification in transportation domain. Wasserstein autoencoders have been applied for topic modeling in [72] by directly constraining Dirichlet prior on the latent document-topic vectors.

Yadav and Vishwakarma [73] scrupulously studied the performance and key aspects of various deep learning models for sentiment analysis. Their survey of sentiment analysis is applicable to different levels such as document, sentence, aspect, multi-domain, multimodal level and also demystified how the deep learning models like CNN, LSTM, RNN are suitable to address the task of sentiment analysis at different levels.

Due to ubiquitous nature of social media platforms and increasing tendency to express emotions and opinions via images, visual sentiment analysis i.e., sentiment analysis from images [74–76] has become popular research domain as an extension to traditional way of performing sentiment analysis at text level.

For alleviating the problem of cumbersome process of annotating the images for the task of visual sentiment analysis, She et al. [74] proposed to use weakly supervised learning technique in which images are labeled globally with manual annotation by human personnel. WSCNet is an end-to-end coupled network which jointly optimizes the task of localizing the affective images and classifying the sentiments using two branches.

Yadav and Vishwakarma [75] proposed residual attention-based deep learning network for classification of visual sentiments from imagJes. They have extensively analyzed CNN-oriented deep learning architectures like VGG-16, VGG-19, InceptionResnet-V2, Inception-V3, ResNet-50, Xception, and NAS-Net demonstrating the effect of fine-tuning the models for the task of visual sentiment analysis.

Therefore, based on the works related to sentiment analysis using deep learning [73], and previous work [77], it can be noted that different deep learning architectures have been successfully applied to address the task of sentiment analysis at various levels.

Dynamic topic models.

To analyze the evolution of topics over time on large corpus, a dynamic topic model (DTM) is proposed [78]. DTM works by dividing data by time slice, for instance, by day, month, or year. Then documents of each slice are modeled using component topic model in which topics at time slot *t* are assumed to be associated with topics at time slice *t-1*. This approach works on sequential data and uses the metadata of documents to get the notion of time and describes the word-topic distribution. In 2013, Ankan et al. [79] issued a patent for a method and a software program for extracting the emerging and evolving topics from streaming data. This method is based on dynamic non-matrix factorization with temporal regularization. For extraction of trending topics from streaming data, Rekik and Jamoussi [80] used stacked autoencoder coupled with data stream clustering algorithm. Their approach uses vanilla LSTM network and topic level attention-based LSTM for detection and topic-level sentiment analysis, respectively. The LDA-based generative topic model put forth in [81] incorporates time zones and location regions. In this model, parameters are incrementally updated between consecutive windows. For faster processing, adaptive window length has been put forth. For adaptive topic modeling in online scenario, Tang and Xia [82] have proposed probabilistic pseudo feedback mechanism. This mechanism refines the term-weights and relevance threshold for adjusting the topic model for change in the environment. Dynamic Online Hierarchical Dirichlet Process model put forth by Fu et al. [83] gives evolution of topics at two-levels, namely, inter-epoch and intra-epoch. At inter-epoch level, topics are evolved sequentially over time by modeling short distance dependencies in the same epoch whereas intra-epoch level gives more emphasis on modeling long term dependencies in continuous epochs modeled using exponential decay function. Tu et al. [84] in 2018 proposed a hierarchical online non-negative matrix factorization (NMF) method for dynamically adjusting topic hierarchy to adaptively manage the emerging, evolving and fading process of the topics. Yin et al. [85] proposed a user-temporal mixture topic model based on regularization framework to perform the detection of both stable and temporal topics from social texts. They applied a burst-weighted smoothing scheme for improving the performance of temporal topic detection. To preserve the time while extracting topics from data that emerge, evolve and fade over time, method of nonnegative CANDECOMP/PARAPAC (CP) tensor decomposition (NNCPD) has been proposed in [86] in which data tensor is decomposed into a minimal sum of outer products of nonnegative vectors. This approach supports both real and synthetic data for dynamic topic modeling.

2.2.2. Pipelined model based methods performing topic detection followed by sentiment analysis

Zhao et al. [87] first performs hot topic detection and then topic based sentiment analysis by aggregating sentiments for the detected hot topic. Fu et al. [88] applied LDA method for detecting theme from text and then applied lexicon based method for finding the sentiment. However, this method works at the paragraph level.

Deep learning architectures have been used for topic modeling tasks [62,63,89,90] due to remarkable results achieved in various research problems. Ren et al. [91] demonstrated the use of topic-enhanced word embedding obtained using recursive autoencoder framework. They first generated topic-related information using LDA and then performed sentiment classification using topic-enhanced word embedding. Kalarani and Brunda [92] focused on selecting the best features for sentiment analysis by combining POS features and joint sentiment topics. They used hidden Markov Model (HMM) and nonparametric hierarchical Dirichlet process for extraction of features. For sentiment classification, both SVM and artificial neural network has been used. The sequential process of topic-level sentiment analysis using LDA and SVM has been adopted by Montenegro et al. in [93]. In this, LDA is used to generate the topics from large number of tweets and then clustered by finding the probability of each tweet belonging to the cluster. Once the topic clusters are formed, supervised machine learning algorithm, namely, SVM is applied to get the sentiment class of each topic cluster. Kaddouri and Mataoui [94] proposed a domain-level topic modeling approach based on supervised learning for improving the sentiment analysis of Arabic text. The approach follows the evolutionary prediction model. During training, a graph based on "terms-categories" is constructed for prediction. During testing phase, documents are classified based on one or more categories and the results are verified using manually annotated labels. Many neural network approaches have proposed to use attention mechanism for sentiment analysis. Pergola et al. [95] proposed hierarchical neural architecture namely topic dependent attention model (TDAM) based on the bidirectional Gated Recurrent Unit. Specifically, hierarchical architecture is used to encode shared topics among words and sentences, and internal attention mechanism is used for getting local topic representations of words and sentences. In majority voting approach proposed by Carmo et al. [96] text is represented as topics in low-dimensional space. To deal with data sparsity problem of short texts, text expansion technique has been proposed. They applied LDA, BTM and MedLDA topic models to detect the topics and used Linear SVM, Random Forest, and Logistic Regression for classification. As per their claim, MedLDA and Random Forest classifier gives the best results for topic-based sentiment analysis.

2.3. Sentiment analysis methods with topic as input

BB_twtr model [23] pre-trains word embeddings using the large scale of unlabeled data and fine-tunes by using small subset of unlabeled data and SemEval 2017 dataset. Moreover, ensemble of CNNs and LSTMs have been used for boosting the performance of topic-level sentiment analysis. DataStories model [24] applies Siamese Bidirectional LSTM having context-aware attention mechanisms for topic-level attention mechanisms. DataStories [24] model focuses on task specific data preprocessing related to text tokenization, word normalization, word segmentation and spell correction to prepare data for Siamese bidirectional LSTM network with context-attention layer. They adopt Bayesian optimization approach to find good hyperparameter values.

Tweeter model proposed by Kolovou et al. [25] uses multiple independent models based on neural networks, topic modeling, affective models, semantic similarity features and word embeddings and then combined them using a late fusion scheme for topic-level sentiment analysis. Tweester model assumes that hashtags associated with some topic or social media event help in expressing the user's sentiment. Therefore, this model incorporates absolute and relative frequencies of hashtags as features in the data preprocessing step. This model uses multiple independent classifiers based on semantic-affective system, a single and a two-step CNN, word embedding and stacking based system. Finally, late fusion scheme is applied which averages the output probabilities from each independent classifier.

Nourbakhsh et al. [26] handled the task of topic-levels sentiment analysis by capturing left and right context of a topic (target). They used both general embeddings and sentiment specific embeddings learned from 200 million and 10 million tweets and incorporated a weighted text feature model. Two types of word embeddings as general and sentiment specific word embeddings have been used in the classifiers of funSentiment [26]

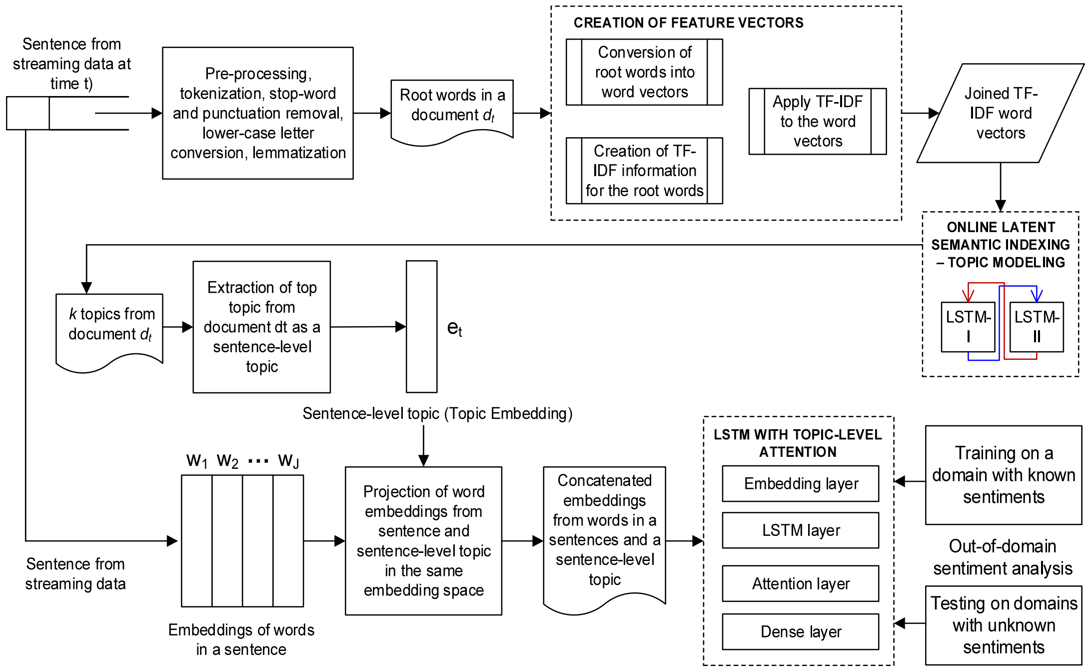


Fig. 2. Overview of proposed methodology.

model. Moreover, this model performs feature extraction using negation handling, TF-IDF weighting scheme and Rocchio text classifier. CrystalNest model [27] is a cascade of two classifiers. This model combines features from sarcasm detection model and other features from Alchemy, NRC lexicon [97], n-grams, word vectors and part-of-speech features. Initially, they build sarcasm detection model based on affect, cognition and socio-linguistic features and then train SVM to detect sarcastic text. After this, the cascaded classification system combines sarcasm score obtained from sarcasm classifier and sentiment score obtained from n-gram features.

Baly et al.'s [98] approach is based on topic-based sentiment analysis in which four different models based on unsupervised topic classification, supervised topic classification, supervised domain classification, and direct sentiment classification have been proposed. Amobee model [99] encompasses 2 parts. First part focuses on supervised training of recursive-neural-tensor-network (RNTN) model, whereas, the second part makes use of Logistic regression for sentiment classification. Since Amobee is an official Twitter partner, global stream of data was available to them for training purpose. Moreover, a custom dictionary has been created considering different aspects of pre-processing. ELiRF-UPV [100] at SemEval-2017 used CNNs and RNNs and both general and specific word embeddings having polarity lexicons to handle the task of topic-level sentiment analysis. Their model works for both English and Arabic languages for sentiment analysis. EICA model [101] used simple CNN for topic-level sentiment classification. In this model, Word2Vec tool has been used to obtain pretrained word embeddings trained on 238 million tweets from Sentiment140 dataset. YNU-HPCC [102] model handles local information within short text of Twitter messages and also long-distance dependency across tweets with the help of multi-channel CNN and LSTM. Specifically, multi-channel strategy allows to extract local n-gram features and such features are then sequentially composed using LSTM.

Based on the literature survey related to topic-level sentiment analysis, very few papers [29,35] addressed the task of detecting the topics from sentences. The research papers [23–27,98] from SemEval 2017 competition [15], assume that topic with reference to the given sentence is already given and calculate the polarity associated with the given topic from each sentence. With reference to the proposed work, in order to perform sentiment analysis, we first aim to extract the topics from sentences and then perform sentiment analysis. The proposed methodology is novel in the sense that we extract the topics from streaming sentences in an online manner and then find sentiments associated with the detected topics. Due to its online nature, the proposed approach is scalable and can be deployed in real-time environment. After extracting the topic, a sentence and a corresponding topic is fed as input to the sentiment analysis model. The model applies topic-level attention mechanism in LSTM network to get the sentiments associated with the topic in a sentence. The proposed model supports topic-level sentiment analysis with both in-domain and out-domain data.

3. Proposed methodology

The proposed approach captures the influence of topic on sentiment by concatenating the topic embeddings with the word embeddings of a sentence and then applies topic-level attention mechanism at sentence level to find the sentiment score. Table 1 gives meaning of notations used in the methodology. Fig. 2 gives an overview of the proposed approach. As per our assumption, each incoming sentence from social media platform is treated as a document. The procedure followed for pre-processing and the creation of feature vectors is depicted in Fig. 3. The approach is divided into 3 phases as creation of feature vectors, topic detection using online latent semantic indexing and topic-level sentiment analysis using LSTM network with topic-level attention. Initially, the tokenization is performed on each sentence from the streaming data to get a list of words using spaCy library [103]. After this, Gensim library [104] is used to remove the stop words, punctuation marks and conversion of words to lowercase. After applying the lemmatization, we get the root words. To improve the accuracy of topic detection, we carefully checked the suitability of word embedding vector models like Word2Vec [105] and Glove [106].

Word2Vec tries to capture the co-occurrence one window at a time but Glove is based on co-occurrence of words in the whole

Table 1
Meaning of the notations used in the proposed methodology.

Notation	Meaning
$\Gamma \in \mathbb{R}^M$	Terms, $\Gamma \equiv [\Gamma_1, \Gamma_2, ..\Gamma_M]$ where M is number of terms
$u_i \in \mathbb{R}^{K\times 1}$	Topics, $u_i \equiv \{u_1, u_2, \ldots, u_K\}$ where $i = \{1, 2, \ldots, K\}$ and K is number of topics
$D \in \mathbb{R}^{M\times N}$	Term-document matrix, $D \equiv [d_1, d_2, \ldots, d_N]$ where M is number of terms, and N is number of documents
$\mathrm{U} \in \mathbb{R}^{M\times K}$	Term-topic matrix, U where M is number of terms and K is number of topics. At sentence level, the term, Γ gives the name of the aspect if term belongs to the topic i.e. corresponding weight of the term u_{mk} is higher for the given topic u.
$V \in \mathbb{R}^{K\times N}$	Topic-document matrix, $V \equiv [v_1, v_2, \ldots, v_N]$ where K is number of topics, N is number of documents. Entries in the matrix V are denoted by v_{kn} which denotes weight of kth topic in document d_N.

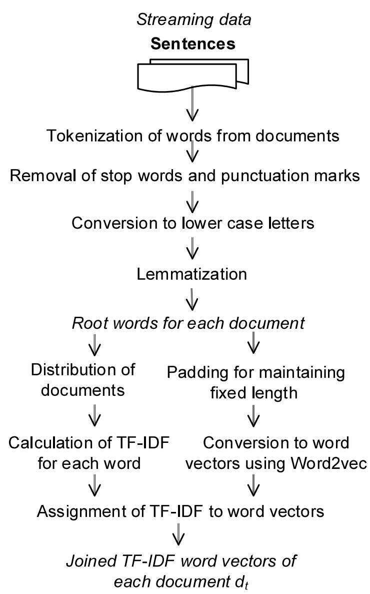


Fig. 3. Pre-processing and creation of feature vectors.

corpus. In addition, Word2Vec does incremental training of a neural network, by repeatedly iterating over a training corpus. Whereas GloVe works to fit vectors to model a giant word co-occurrence matrix built from the corpus. Incremental training suits with the proposed approach of online learning for topic detection from streaming data and therefore Word2Vec chosen [107,108]. We aim to combine the TF-IDF features with word embeddings obtained from Word2Vec model. We used vectors which are pre-trained on Google News dataset having about 100 billion words. The dimensionality of the vector is 300. There are 2 main reasons of combining these exclusive features. These are (1) TF-IDF works on bag-of-words (BoW) model and gives most relevant topics by capturing the semantic relationship. (2) Word2Vec model helps to maintain the syntactic and semantic relations among words. Therefore, we first create word vectors using Word2Vec model available in Gensim library [104] for each of the cleaned words with 300 dimensions. Then, Scikit library [109] is used for creating the TF-IDF information for each document. The TF-IDF is calculated using Eq. (1).

$$TF\text{–}IDF = \frac{c(\Gamma, d)}{|d|} \times \log \frac{|D|}{|\{d \in D \colon \Gamma \in d\}|} \tag{1}$$

In Eq. (1), $c(\Gamma, d)$ is the count that the term Γ occurs in document d, $|d|$ is the length of document d, $|D|$ is total count of documents in document collection and $|\{d \in D \colon \Gamma \in d\}|$ denotes the total count of documents in which term Γ occurs. TF-IDF weighting is applied to each word in a document. Therefore, as an output of the pre-processing step, we get the word vectors with TF-IDF information applied for each document d_t and collectively we get word vectors with TF-IDF information applied for the documents $D = \{d_1, d_2, \ldots, d_n\}$.

For each incoming document d_t, an online latent semantic indexing model designed using 2 LSTM networks approximates document and incrementally updates the term-topic document matrix and topic-document matrix. The top topic from topic-document matrix V is then declared as a sentence-level topic. For topic-level sentiment analysis, embeddings of words in a sentence and sentence-level are found by projecting them into the same embedding space. Then topic embedding is concatenated with each word embedding from a sentence. After this, attention-based LSTM with topic embeddings is used to get the feature representation of a sentence given an input topic. Applying sigmoid activation on the dense layer yields probability values for each sentiment class.

3.1. Online learning for topic detection from streaming data

Latent semantic indexing is a non-probabilistic approach of topic modeling in which each document d_n is approximated as the product of term-topic matrix U and topic-document matrix V. We aim to handle the problem of topic discovery at sentence-level, assuming sentence as a document using online latent semantic indexing. A document d_t from a collection of n documents $D = [d_1, d_2, \ldots, d_N]$ is represented by a vector d with M dimensions in which mth entry denotes the weight of the mth term calculated using TF-IDF score and actual term is represented by its word vector obtained from Word2Vec. Therefore, documents from this collection can be given as $M \times N$ term-document matrix which is represented as $D = [d_1, \ldots, d_N]$ in which rows represent the terms in the form of word vectors and columns represent the document. As topic modeling discovers latent topics and models the document as a mixture of topics, document d_n can be represented as $d_n \approx \sum_{k=1}^{K} v_{kn} u_k = U v_n$ where $u_1, u_2, \ldots, u_K$ denote the topics, and v_{kn} is the weight of kth topic. Term-document matrix $V = [v_1, \ldots, v_N]$ is used for representing the document d_n in the topic space where each column v_n gives latent topic space representation of the document d_n. An algorithm 1 shows steps for sentence-level topic detection.

For extracting the latent topics from the documents, latent semantic indexing constrained by regularization works in the following manner. We perform the stochastic approximation of squared $\ell 2$ norm of the difference between term-document matrix and a product of term-topic matrix and topic-document matrix as $\|d_n - U v_n\|_2^2$. The pictorial representation of the matrices is given in Fig. 4.

The constraints are imposed on both the topics and the latent space representation of the topics using $\ell 1$ and $\ell 2$ regularization. With reference to previous work [110,111], we impose $\ell 1$ and $\ell 2$ constraints on term-topic matrix and topic-document matrix respectively to optimize an Eq. (2) for stochastic approximation as follows.

$$\left\{ \frac{1}{N} \sum_{n=1}^{N} \left[\|d_n - U v_n\|_2^2 + \lambda \, \|v_n\|_2^2 \right] + \delta \sum_{k=1}^{K} \|u_k\|_1 \right\}_{\min U,\ \{v_n\}} \tag{2}$$

Algorithm 1: Algorithm for sentence-level topic detection
Input: *Streaming Sentence*
Output: *Topic from sentence*
Assumption: *Assume each sentence emanating from social media platform as document*

1. **for** each sentence
2. Perform pre-processing
3. Apply tokenization
4. Remove stop-words and punctuation marks
5. Perform lower-case letter conversion
6. Apply lemmatization
7. Extract root words from a sentence
8. **for** each word in a sentence
9. Calculate TF-IDF information for each root word
10. Transform word to word vector by Word2Vec
11. Apply TF-IDF information to the word vector
12. **end**
13. **end**
14. Build and update term-document matrix D as a collection of streaming sentences such that $D \in \mathbb{R}^{M\times N}$
15. Represent document d_t with row as word vector for each term and corresponding TF-IDF score
16. Repeat for steps from 17 to 22 for extracting the topic from each streaming sentence
17. In first LSTM network
18. repeat
    a. Provide document d_t at time and previous timestamp's U_{t-1} matrix (or randomly initialized U_{t-1}) as input
    b. Project the document d_t to get the projection v_t of document d_t keeping U fixed using equation (3)
    c. Retrieve v_t
19. until convergence
20. In second LSTM network
21. repeat
    a. Provide document d_t at time and document representation v_t as input obtained from first LSTM
    b. Keeping v_t fixed, update the term-topic matrix U_t using equation (4)
22. until convergence
23. Lookup the topic-document matrix V to retrieve the topics from document d_t
24. For document d_t, select the top topic from V as a topic in the sentence

In Eq. (2), N denotes total count of documents in a collection, variables δ and λ are used for controlling the $\ell1$ and $\ell2$ norm respectively. As seen from Eq. (2), both U and $\{v_n\}$ must be minimized. However, this minimization is not jointly convex and therefore, the equation needs to be alternatively minimized by keeping either of the variable fixed and updating the other variable at a time.

For alternate updating of U and V matrices, we implemented 2 LSTM models, as shown in Fig. 5. For online latent semantic analysis with regularization, we assume that documents are assumed to be independent of each other. Initially, when new streaming data d_t arrives at time t, data cleaning and representation process are performed, and TF-IDF word vectors are fed to LSTM network - I. We start with any random matrix U_0 and input them as joined TF-IDF word vectors to LSTM network - I for getting the projection v_1 for the input document. LSTM network - I uses Eq. (3) for calculating this projection.

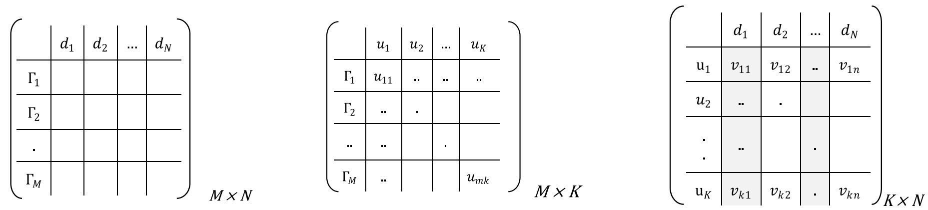
$$\{\|d_t - U_{t-1}v\|_2^2 + \lambda\,\|v\|_2^2\}_{\min v} \tag{3}$$

The projected latent vector v_1 and input document d_t are then provided as input to LSTM network - II, which applies equation (4) to get the term-topic matrix U_1.

$$\left\{\frac{1}{t}\sum_{t=1}^{t}\left[\|d_i - Uv_i\|_2^2 + \lambda\,\|v_n\|_2^2\right] + \delta\sum_{k=1}^{K}\|u_k\|_1\right\}_{\min U} \tag{4}$$

For the next streaming sentence, LSTM network - I use term-topic matrix U_{t-1} from the previous iteration and finds latent space representation v_t. LSTM network - II accepts v_t and input document representation d_t and finds term-topic matrix U_t. This way, by alternatively modifying U_i and v_i and keeping only one LSTM model active at a time, the proposed model works in an online manner. This process is repeated for each streaming sentence.

We are approximating each streaming sentence incrementally as explained above which shows the timeliness of topic modeling. Suppose, for example, at time t, we receive tweet1 and topic detected from tweet1 are obtained as k top topics. The topic having the highest weight is considered as a final topic since in our case the number of topics needed to be chosen for sentiment analysis is 1. Following that principle, we evaluate the sentiment of the topic from sentence. As described in the methodology, a model receives another tweet2 at time t + 1 and so on. This way the topics are detected and finalized. This process is repeated for each streaming sentence and thus, timeliness of topics is maintained.



(a) Term-document matrix, D (b) Term-topic matrix, U (c) Topic-document matrix, V

Fig. 4. Representation of the matrices.

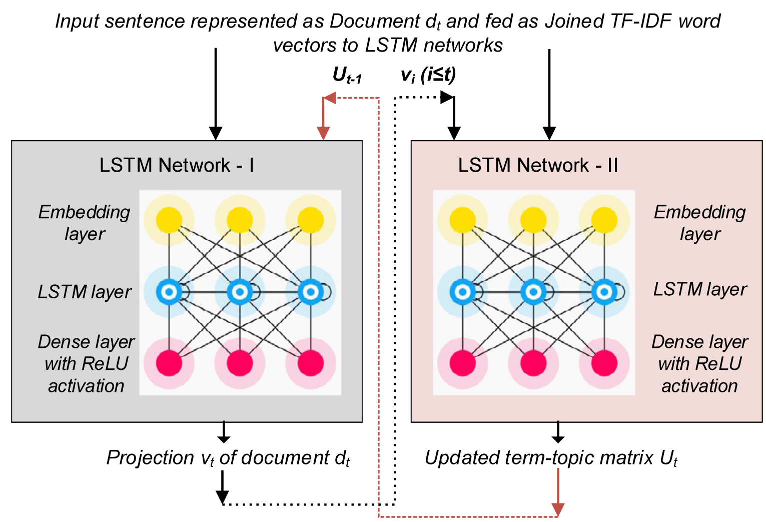


Fig. 5. Approximation of a document using two LSTM networks.

3.2. Topic-level attention LSTM for sentiment analysis

Let $\mathbb{W} \in \mathbb{R}^{d\times|V|}$ be an embedding lookup generated by Word2Vec model where d denotes the dimension of word embeddings and $|V|$ denotes vocabulary size. The word embeddings of the input sentence are denoted as $\{w_1, w_2, \ldots, w_J\}$ where $w_J \in \mathbb{R}^d$. Topic detected by online LSI for the given sentence is represented by its embedded representation e_t and the dimension of topic embedding is given by d_{et}. Word embeddings and topic embeddings vectors can be added together in the embedding space by multiplication, summation and concatenation. As experimented in [112], concatenation operator yields the better results and therefore, for capturing the influence of topic on sentiment, we concatenated the topic embedding with the word embeddings of a sentence. These concatenated vectors mapped in the same embedding space are then fed as input to the attention-based LSTM network. The attention-based LSTM network with topic embedding is shown in Fig. 6. We follow the strategy of attention mechanism mentioned in [113,114]. An algorithm 2 shows steps for topic-level sentiment analysis. The concatenated word embeddings and topic embedding are fed as input to the LSTM. The output of LSTM is represented by $H \in \mathbb{R}^{d\times J}$ where $[h_1, h_2, \ldots, h_J]$ denote hidden vectors, d denotes size of hidden layer and J denotes length of input sentence. Moreover, $o_J \in \mathbb{R}^J$ represents column vector with J number of 1 s. The attention mechanism produces attention vector α, weighted hidden representation γ using Eqs. (6) and (7) respectively.

$$G = \tanh\left(\begin{bmatrix} W_h H \\ W_e e_t \otimes o_J \end{bmatrix}\right) \tag{5}$$

where, $G \in \mathbb{R}^{(d+d_{et})\times J}$, and projection parameters are given as $W_h \in \mathbb{R}^{d\times d}$, $W_e \in \mathbb{R}^{d_{et}\times d_{et}}$, $w \in \mathbb{R}^{d+d_{et}}$. An expression $e_t \otimes o_J$ stands for repeatedly concatenating e_t with column vector o_J J number of times.

$$\alpha = softmax(w^T G) \tag{6}$$

$$\gamma = H\alpha^T \tag{7}$$

The final representation of input sentence with the topic is calculated using Eq. (8).

$$z^* = \tanh\left(W_{p1} + W_{p2} h_J\right) \tag{8}$$

where z^* is final representation of input sentence with the topic, W_{p1} and W_{p2} are the projection parameters. We apply linear layer over z^* to get real-valued vector e with its length equal to number of classes (positive and negative). Passing this real valued vector to a sigmoid layer yields probability score over sentiment classes.

4. Experimentation details

The proposed model is implemented using Python [115]. We used Keras API [116] with TensorFlow [117] as backend and libraries such as spaCy [103], Scikit-learn [109] and Gensim [104]. The experimentations are conducted on Google cloud Platform [118] with 2 vCPUs, 7.5 GB of memory and persistent disk size of 64 TB.

4.1. Datasets used

For topic detection from streaming sentences, we developed our own dataset under three hashtags #facebook, #bitcoin, and #ethereum. The dataset is published on Mendeley research [119]. We followed "random walk training" for fine-tuning the model for topic detection for 3 datasets. We used 1 week of Twitter sentences from 25 March to 31 March 2018 each from 3 datasets

Algorithm 2: Algorithm for topic-level sentiment analysis
Input: *Sentence and topic*
Output: *Sentiment score of a topic in a sentence*

1. Represent words in a sentence as word embeddings $w_1, w_2, \ldots, w_J$ and topic as embedded vector et by Word2Vec lookup by maintaining the same embedding space
2. Concatenate embedded topic with each word embeddings $w_1, w_2, \ldots, w_J$ and pass to LSTM network
3. Calculate hidden representation as $H = h_1, h_2, \ldots, h_J$
4. Find topic-level attention vector α consisting of attention weights
5. Draw γ weighted representation of a sentence
6. Represent input sentence with topic as $z*$ and convert it into real-valued vector e
7. Pass vector e to dense layer with sigmoid activation to get conditional probability distribution for the sentiment classes

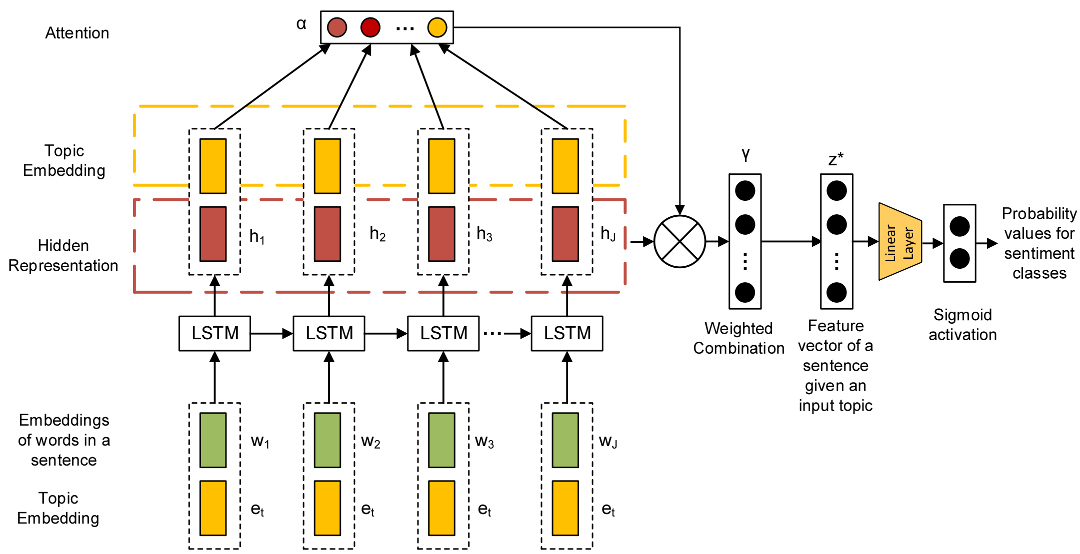


Fig. 6. Topic-level attention LSTM network for sentiment analysis.

for initial fine-tuning. Table 2 shows the statistics of dataset for fine-tuning. For in-domain topic-level sentiment analysis, we trained the topic-level attention LSTM model on SemEval-2017 Task 4 Subtask B dataset [15] and tested on it. For checking out-domain sentiment analysis performance of the model, we trained the model on SemEval-2017 Task 4 Subtask B dataset and tested it on 3 datasets, namely, Facebook, Ethereum and Bitcoin. We repeated the following steps for testing the performance on 3 datasets. We labeled data manually using the service of Amazon Mechanical Turk to form a ground truth in order to quantify the performance of the proposed model on the 3 datasets (Facebook, Ethereum and Bitcoin).

During testing, we fed the streaming sentences one after another from a dataset to our topic detection module for extraction of topic. The extracted topic and corresponding sentence are then passed to proposed topic-level sentiment analysis model to find the sentiment score. The details of the training and testing dataset are given in Table 3.

4.2. Training and regularization

In this section, we give training and regularization strategies for LSTM network used for topic detection and attention-based LSTM network used for topic-level sentiment analysis.

Table 2
Details of dataset for initial fine-tuning for topic detection.

Name of dataset	Approximate No. of Tweets used for initial fine-tuning (Tweets belonging to 1 week)
Facebook	48 861
Ethereum	45 861
Bitcoin	46 494

Table 3
Details of dataset for topic-level sentiment analysis.

Phase	Dataset	Positive	Negative	Total
Training	SemEval-2017 Task 4 Subtask B	14 897	3997	18 894
Testing	SemEval-2017 Task 4 Subtask B	2463	3722	6185
	Ethereum	1067	933	2000
	Bitcoin	1129	879	2000
	Facebook	756	1244	2000

4.2.1. For LSTM network used for topic detection

For word vectors obtained from Word2Vec, we set size of 300 dimensions for each embedded vector. Due to high cost associated with single parameter update in backpropagation through time (BPTT), we applied truncated BPTT (TBPTT) for both the LSTM networks in Fig. 5 for topic detection. Following the conventions mentioned in work by Williams and Peng [120], we applied common configuration as TBPTT (n1, n2) where n1 = n2 = 100 s in which fixed number of time steps are used for forward and backward-pass timesteps. We set initial state of LSTM as noisy one for getting improved performance [121]. For remembering more, the biases of LSTM's forget gate has been set to 1. We kept length of latent dimension as 1024. We used dropout rate of 0.2 for regularizing the network. As mentioned in Eq. (4), we varied the model's parameters in the range of 10 to 50 for K number of topics and λ in the range of 0.01 to 1 and δ in the range of 0.01 to 1. After fine-tuning, we chose to keep parameters as K = 20, $\lambda = 0.5$ and $\delta = 1$. We kept learning rate as 0.1 and used Adam optimizer. As dataset is large and we need to find topics in online manner. We applied "random walk training" approach so that model gets fine-tuned in some iterations.

4.2.2. For attention-based LSTM network used for topic-level sentiment analysis

Attention-based LSTM network shown in Fig. 6 has been trained in an end-to-end manner using backpropagation. For LSTM model, we set the size of embedded vectors to 300, latent dimension size as 1024, the number of epochs as 30 and batch size as 64. For the last layer, we used binary cross-entropy between actual labels and predicted labels for sentiment analysis. For optimization, Adam optimizer has been used. For activity regularizer, $\ell 2$ norm of penalty 0.01 has been set.

4.2.3. Baseline model

We applied Gaussian Naïve Bayes classifier for sentiment analysis as a base model. As part of pre-processing, URLS and usernames have been replaced with keywords, punctuation marks have been removed and text converted to lowercase letters. To train with GaussianNB classifier, a pipeline class has been used to work in the sequence as vectorizer, transformer and classifier. Hyperparameters like n-grams range, IDF usage, TF-IDF normalization type were tuned using grid search and finally best parameters have been obtained as stop_words = 'english', ngram_range = (1,2), tfidf__norm = $\ell 2$, tfidf__use_idf = True. We set default parameter as var_smoothing = 1e−9 (.000000001). We achieved average accuracy, F_1^{PN} score, and accuracy as 0.511, 0.528, and 0.542, respectively.

5. Results

The results have been discussed in terms of quantitative and qualitative analysis. For quantitative analysis, the proposed model is evaluated based on performance metrics given by recall, average recall, macro-average F1 score and accuracy. For qualitative analysis, the performance is measured in terms of scalability to support streaming data from social media platforms.

5.1. Quantitative analysis

The performance of the proposed model has been checked for both in-domain and out-of-domain sentiment analysis. For in-domain sentiment analysis, we trained and tested the model on the dataset of SemEval-2017 Task 4 Subtask B. For out-of-domain sentiment analysis, we tested the model on three datasets — Ethereum, Bitcoin, and Facebook collected via Twitter API with #ethereum, #bitcoin and #facebook as the hashtag.

For checking the performance of the model on these four datasets, metrics such as recall with respect to positive class R^P, recall with respect to negative class R^N, average recall (*AvgRec*), macro-average F_1 score with respect to positive and negative classes (F_1^{PN}) and accuracy (*Acc*) have been calculated. These metrics are mentioned in Eqs. (9), (10), and (11) respectively.

$$AvgRec = \frac{1}{2}\left(R^P + R^N\right) \tag{9}$$

Table 4
Results of applying proposed model on test datasets for topic-level sentiment analysis.

Dataset	AvgRec	R^P	R^N	F_1^{PN}	Acc
Ethereum	0.846	0.862	0.831	0.842	0.844
Bitcoin	0.824	0.841	0.807	0.814	0.817
Facebook	0.794	0.853	0.735	0.787	0.79
SemEval-2017 Task 4 Subtask B	0.879	0.832	0.926	0.879	0.889

where R^P, and R^N denote recall with respect to the positive and the negative class respectively. The value of *AvgRec* ranges in [0, 1].

$$F_1^{PN} = \frac{1}{2}\left(F_1^P + F_1^N\right) \tag{10}$$

where F_1^P, and F_1^N denote F_1 with respect to the positive and the negative class respectively.

$$Acc = \frac{TP + TN}{TP + TN + FP + FN} \tag{11}$$

where *TP, TN, FP* and *FN* denote true positive, true negative, false positive and false negative, respectively.

Table 4 shows the results obtained for topic-level sentiment analysis. As mentioned in the paper [15], we focused on average recall as the primary evaluation measure. Macro-average F_1 measure and accuracy are considered as secondary measures. Average recall measure is robust enough to class imbalance [122]. As mentioned in Table 3, SemEval dataset for training has imbalanced number of positive and negative classes, and therefore, it is more important to emphasize on average recall measure than standard accuracy. Fig. 7 shows the Precision–Recall curves and corresponding Area Under PR Curve values obtained by proposed model on the datasets Ethereum, Bitcoin, Facebook, and SemEval-2017 Task 4 Subtask B.

5.2. Qualitative analysis

The effectiveness of the proposed model has been demonstrated qualitatively in terms of scalability, and the metrics considered for assessing the scalable performance of the model are average time in milliseconds for creation of feature vectors, throughput in terms of topics detected per second and average response time in seconds to handle the sentiment analysis queries. SemEval-2017 Task 4 Subtask B dataset has been used and fed in streaming manner to the model. Initially, we measure the time elapsed between arrival of the tweet text and creation of feature vector of input text, i.e., average time required by the model to create feature vectors. This is referred to as average time in milliseconds for creation of feature vectors.

We also measure the throughput of the model in terms of topics detected per second with reference to arrival rate of tweets (tweets/s). Finally, we calculate the time required by the model to process a sentiment analysis query provided a sentence and a topic in the sentence. For calculating the average response time of sentiment analysis query, arrival rate of different sentiment analysis queries is varied.

As shown in Fig. 8, average time in milliseconds for creation of feature vectors is stable up to 500 tweets/s. Till this rate, tweet text is immediately processed. After this, tweet text needs to wait for processing. As shown in Fig. 9, throughput increases until arrival rate of tweets reaches 700 tweets/s. After this rate, throughput gets flattened showing that processing capacity of the model has reached. This justifies that the processed model can detect topics from streaming data at the rate of 700 tweets/s. Fig. 10 shows variation in response time with reference to increase in the arrival rate of sentiment analysis queries. We varied

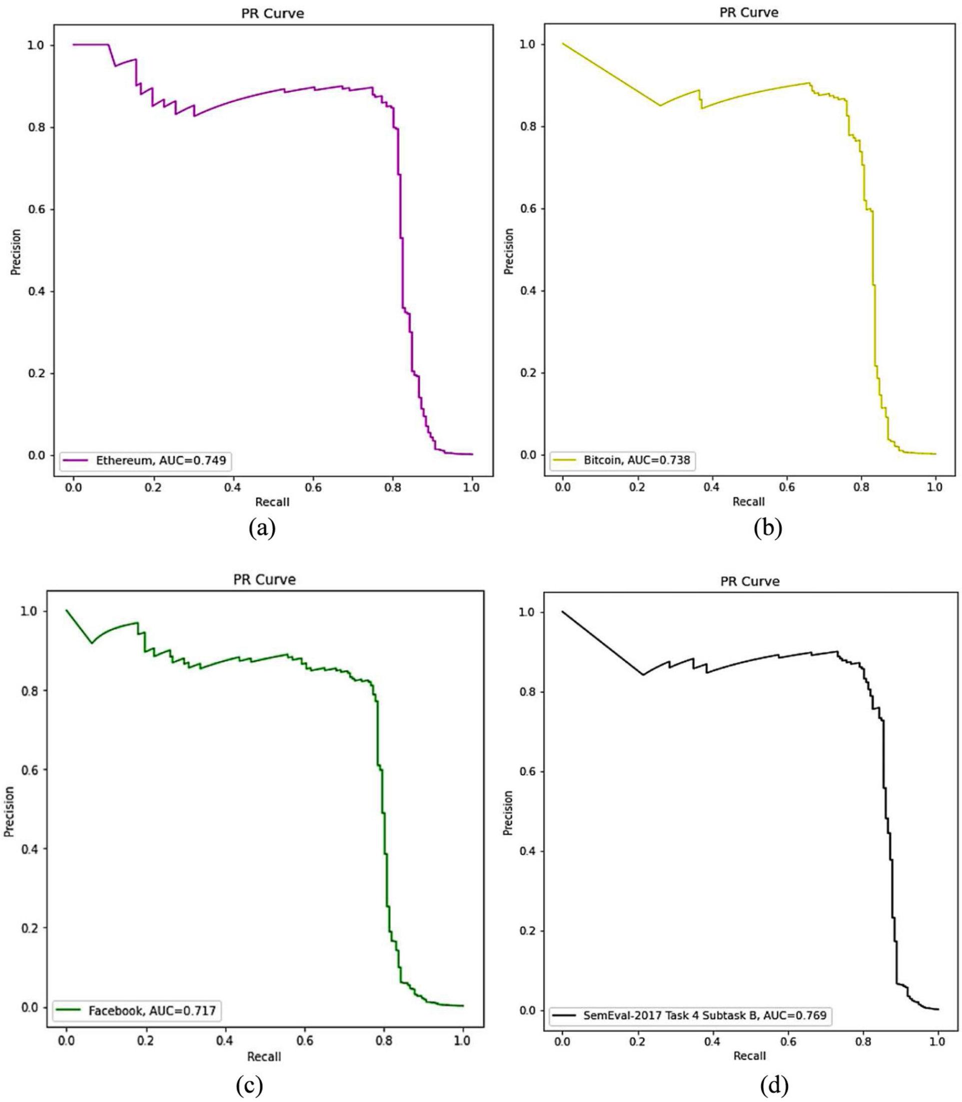


Fig. 7. PR curve and area under PR curve obtained for (a) Ethereum (b) Bitcoin (c) Facebook and (d) SemEval-2017 Task 4 Subtask B dataset.

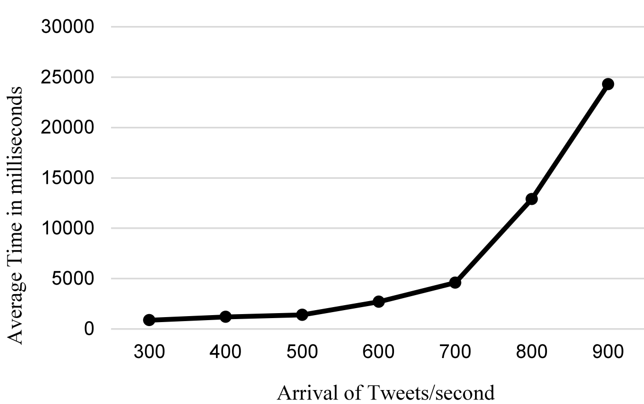


Fig. 8. Average time in milliseconds for creation of feature vectors.

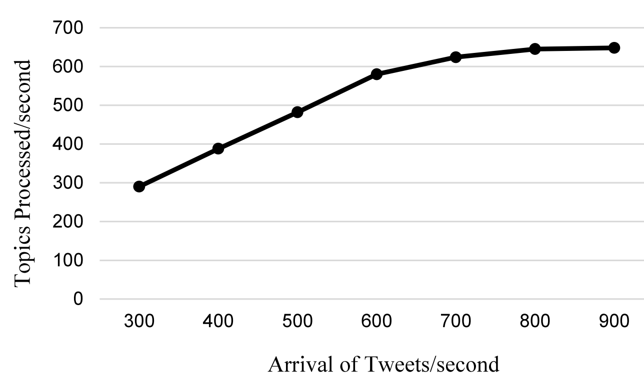


Fig. 9. Throughput in seconds.

the sentiment analysis queries in the range of 5 to 30 queries per second and measured the average response time in seconds for the arrival rate of 200 tweets/s, 400 tweets/s and 600 tweets/s.

For arrival rate of 600 tweets/s, average response time of sentiment analysis queries for the load of 30 queries per second is about 15 s. Based on these results the proposed model qualifies to support online response and scales with streaming data and achieved significant performance for topic-level sentiment analysis.

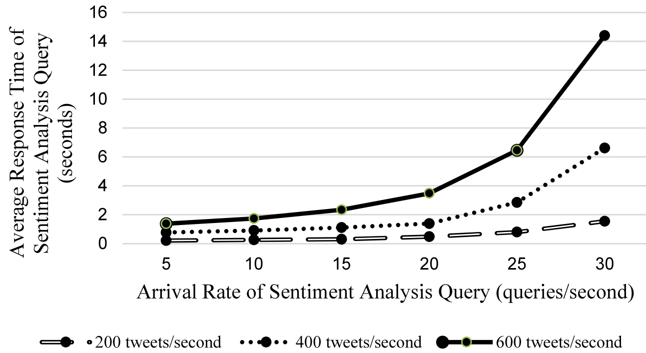


Fig. 10. Average response time of sentiment analysis query.

6. Comparison with state-of-the-art methods

For topic-level sentiment analysis, we applied Gaussian Naïve Bayes as a baseline model and achieved average accuracy, F_1^{PN} score, and accuracy as 0.511, 0.528, and 0.542, respectively. The performance of the proposed model has been compared with state-of-the-art models from SemEval-2017 competition's sub-task B of task 4 [15] as shown in Table 5. Fig. 11 gives a graphical representation for comparative analysis. From the comparative analysis against the outperforming model, we observed that BB_twtr model [23] is heavily dependent on unlabeled dataset other than the dataset made available. In addition to dataset offered by SemEval competition, BB_twtr model uses more 100 million unlabeled tweets in training. They developed new dataset, namely, distant dataset consisting of 5 million positive tweets and 5 million negative tweets. Moreover, BB_twtr model which is an ensemble of CNN and LSTM, applies task specific tricks at data preprocessing and also performs concatenation of word vectors from 2 different embedding spaces of varying dimensions. However, the model proposed in this paper supports scalable, and dynamic topic modeling in addition to the basic task of sentiment analysis as given in SemEval competition. Moreover, the proposed model only used single embedding space and standard dataset from SemEval data and achieved significant performance and outperforms the approaches mentioned in papers [24–27].

Table 5
Comparison with state-of-the-art topic-level sentiment analysis approaches.

Model	AvgRec	F_1^{PN}	Acc
BB_twtr [23]	0.882	0.890	0.897
DataStories [24]	0.856	0.861	0.869
Tweester [25]	0.854	0.856	0.863
funSentiment [26]	0.834	0.824	0.827
CrystalNest [27]	0.827	0.822	0.827
Gaussian Naïve Bayes (Baseline)	0.511	0.528	0.542
Topic-level attention LSTM network (Proposed)	0.879	0.879	0.889

7. Conclusion and future research

Among various tasks in natural language processing, sentiment analysis is one of the crucial ways of understanding the sentiments of the users towards political issues, products, services, worldwide events, etc. Topic modeling, when coupled with sentiment analysis, helps to find the ongoing topics being discussed on social media platforms and helps to understand the sentiments. Based on the user's sentiment towards the detected topic would help make appropriate decisions, devise the strategies for improvement of the service or product. We have proposed the topic-level sentiment analysis approach based on topic modeling and deep learning. As the model works in an online manner for topic detection, it is scalable.

The model achieved an average accuracy of 0.879 on SemEval dataset 2017 dataset for the task of in-domain sentiment analysis and area under PR curve as 0.769. For the task of out-of-domain sentiment analysis at the topic level for the datasets Ethereum, Bitcoin, and Facebook, an average accuracy of 0.846, 0.824, 0.794 has been achieved respectively. Moreover, we achieved an area under PR curve as 0.749, 0.738, 0.717 for datasets Ethereum,

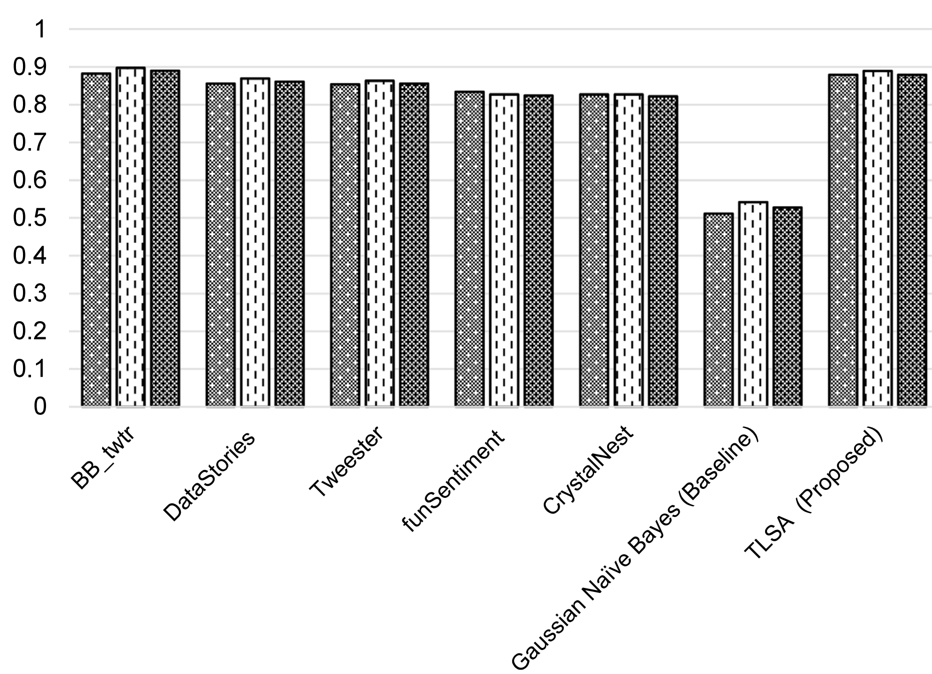


Fig. 11. Pictorial representation for comparison with state-of-the-art approaches.

Bitcoin, and Facebook, respectively for the task of out-of-domain sentiment analysis at the topic level.

The novelty of the proposed approach is that it works at the sentence level to extract the topic using online latent semantic indexing with regularization constraint and then applies topic level attention mechanism in long short-term memory network. Specifically, For online working of the proposed model, short text (sentences) emanating from social media platforms are processed in serial fashion. Deep learning based online latent semantic indexing model incrementally builds the topic model according to streaming data. Given a new sentence(s), topic vector of new sentence is predicted given the previously learned term-topic matrix. After this, the term-topic matrix is updated based on new sentence(s) and predicted topic vector. This way by applying online learning technique, the proposed model supports large dataset and support scalable topic modeling.

To assess the performance of the model for scalability, we analyzed the model in terms of average time in milliseconds for creation of feature vectors, throughput in terms of topics detected per second and average response time in seconds to handle the sentiment analysis queries.

The model took 25 000 average time in milliseconds for creation of feature vectors for arrival of tweets at the rate of 500 tweets/s. The model can detect topics from streaming data at arrival rate of 700 tweets/s, achieving the throughput of processing 624 topics per second. We varied the sentiment analysis queries in the range of 5 to 30 queries per second and measured the average response time in seconds for the arrival rate of 200 tweets/s, 400 tweets/s and 600 tweets/s. For arrival rate of 600 tweets/s, average response time of sentiment analysis queries for the load of 30 queries per second is about 15 s. Based on these results the proposed model qualifies to support online response and scales with streaming short text data and achieved significant performance for topic-level sentiment analysis.

Assuming the proposed model as state-of-the-art sentiment analysis model, pointers to possible future directions have been enlisted. The current approach assumes that each streaming sentence exhibits a single topic being detected. The model can be upgraded to detect multiple topics and associated sentiments from streaming sentence at a time (multiple-topic detection with multiple sentiment classes). Moreover, special symbols, emoticons can be incorporated to perform topic-level sentiment analysis. As future work, the model can be designed to work in the parallel mode so that topic detection and sentiment classification would be performed simultaneously using the joint model.

CRediT authorship contribution statement

Ajeet Ram Pathak: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Data curation, Writing - original draft, Visualization, Project administration. **Manjusha Pandey:** Validation, Formal analysis, Investigation, Resources, Writing - review & editing, Supervision, Project administration. **Siddharth Rautaray:** Validation, Formal analysis, Investigation, Resources, Writing - review & editing, Supervision, Project administration.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] Search Engine Journal, https://www.searchenginejournal.com/growth-social-media-v-3-0-infographic/155115.
[2] Data generated by Facebook, https://techcrunch.com/2012/08/22/how-big-is-facebooks-data-2-5-billion-pieces-of-content-and-500-terabytes-ingested-every-day/.
[3] Twitter statistics, https://www.internetlivestats.com/twitter-statistics/.
[4] B. Liu, Sentiment analysis and opinion mining, Synth. Lect. Hum. Lang. Technol. 5 (2012) 1–167.
[5] Emotion Recognition and Sentiment Analysis Market, https://www.tractica.com/newsroom/press-releases/emotion-recognition-and-sentiment-analysis-market-to-reach-3-8-billion-by-2025/.
[6] B. Pang, L. Lee, S. Vaithyanathan, Thumbs up?: sentiment classification using machine learning techniques, in: Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10, 2002, pp. 79–86.
[7] R. Narayanan, B. Liu, A. Choudhary, Sentiment analysis of conditional sentences, in: Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1, 2009, pp. 180–189.
[8] T.T. Thet, J.-C. Na, C.S.G. Khoo, Aspect-based sentiment analysis of movie reviews on discussion boards, J. Inf. Sci. 36 (2010) 823–848.
[9] C. Lin, Y. He, Joint sentiment/topic model for sentiment analysis, in: Proceedings of the 18th ACM conference on Information and knowledge management, 2009, pp. 375–384.
[10] J. Xu, et al., Sentiment analysis of social images via hierarchical deep fusion of content and links, Appl. Soft Comput. 80 (2019) 387–399.
[11] K. Zhang, Y. Zhu, W. Zhang, W. Zhang, Y. Zhu, Transfer correlation between textual content to images for sentiment analysis, IEEE Access 8 (2020) 35276–35289.
[12] H. Zhang, et al., Multidimensional extra evidence mining for image sentiment analysis, IEEE Access 8 (2020) 103619–103634.
[13] A. Agarwal, A. Yadav, D.K. Vishwakarma, Multimodal sentiment analysis via RNN variants, in: 2019 IEEE Int. Conf. Big Data, Cloud Comput. Data Sci. & Eng., 2019, pp. 19–23.
[14] A. Yadav, D.K. Vishwakarma, A unified framework of deep networks for genre classification using movie trailer, Appl. Soft Comput. 96 (2020) 106624.
[15] S. Rosenthal, N. Farra, P. Nakov, SemEval-2017 task 4: Sentiment analysis in Twitter, in: Proceedings of the 11th international workshop on semantic evaluation (SemEval-2017), 2017, pp. 502–518.
[16] J. Serrano-Guerrero, J.A. Olivas, F.P. Romero, A T1OWA and aspect-based model for customizing recommendations on ecommerce, Appl. Soft Comput. 97 (2020) 106768.
[17] X. Yan, J. Guo, Y. Lan, J. Xu, X. Cheng, A probabilistic model for bursty topic discovery in microblogs, in: Proceedings of the AAAI Conference on Artificial Intelligence 29 (2015).
[18] L. Hong, B.D. Davison, Empirical study of topic modeling in twitter, in: Proceedings of the first workshop on social media analytics, 2010, pp. 80–88.
[19] X. Cheng, X. Yan, Y. Lan, J. Guo, Btm: Topic modeling over short texts, IEEE Trans. Knowl. Data Eng. 1 (2014).
[20] J. Qiang, P. Chen, T. Wang, X. Wu, Topic modeling over short texts by incorporating word embeddings, in: Pacific-Asia Conference on Knowledge Discovery and Data Mining, 2017, pp. 363–374.
[21] J. Pang, X. Li, H. Xie, Y. Rao, SBTM: Topic modeling over short texts, in: International Conference on Database Systems for Advanced Applications, 2016, pp. 43–56.
[22] V.K.R. Sridhar, Unsupervised topic modeling for short texts using distributed representations of words, in: Proceedings of the 1st workshop on vector space modeling for natural language processing, 2015, pp. 192–200.
[23] M. Cliche, Bb_twtr at semeval-2017 task 4: Twitter sentiment analysis with cnns and lstms, 2017, arXiv Prepr. arXiv:1704.06125.
[24] C. Baziotis, N. Pelekis, C. Doulkeridis, Datastories at semeval-2017 task 4: Deep lstm with attention for message-level and topic-based sentiment analysis, in: Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017), 2017, pp. 747–754.
[25] A. Kolovou, et al., Tweester at SemEval-2017 Task 4: Fusion of Semantic-Affective and pairwise classification models for sentiment analysis in Twitter, in: Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017), 2017, pp. 675–682.
[26] Q. Li, A. Nourbakhsh, X. Liu, R. Fang, S. Shah, funSentiment at SemEval-2017 Task 4: Topic-Based Message Sentiment Classification by Exploiting Word Embeddings, Text Features and Target Contexts, in: Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017), 2017, pp. 741–746.
[27] R.K. Gupta, Y. Yang, Crystalnest at semeval-2017 task 4: Using sarcasm detection for enhancing sentiment classification and quantification, in: Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017), 2017, pp. 626–633.

[28] H. Wang, et al., Jointly discovering fine-grained and coarse-grained sentiments via topic modeling, in: Proceedings of the 22nd ACM international conference on Multimedia, 2014, pp. 913–916.
[29] S. Brody, N. Elhadad, An unsupervised aspect-sentiment model for online reviews, in: Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Association for Computational Linguistics, 2010, pp. 804–812.
[30] K. Xu, J. Huang, G. Qi, A new sentiment and topic model for short texts on social media, in: Joint International Semantic Technology Conference, 2017, pp. 183–198.
[31] M. Fatemi, M. Safayani, Joint sentiment/topic modeling on text data using a boosted restricted Boltzmann machine, Multimedia Tools Appl. (2019) 1–17.
[32] P. Liu, J.A. Gulla, L. Zhang, Dynamic topic-based sentiment analysis of large-scale online news, in: International Conference on Web Information Systems Engineering, 2016, pp. 3–18.
[33] B. Dahal, S.A.P. Kumar, Z. Li, Topic modeling and sentiment analysis of global climate change tweets, Soc. Netw. Anal. Min. 9 (2019) 24.
[34] X. Fu, X. Sun, H. Wu, L. Cui, J.Z. Huang, Weakly supervised topic sentiment joint model with word embeddings, Knowledge-Based Syst. 147 (2018) 43–54.
[35] T. Chen, J. Parsons, A sentence-level sparse gamma topic model for sentiment analysis, in: Canadian Conference on Artificial Intelligence, 2018, pp. 316–321.
[36] A. Almars, X. Li, X. Zhao, Modelling user attitudes using hierarchical sentiment-topic model, Data Knowl. Eng. 119 (2019) 139–149.
[37] S. Xiong, K. Wang, D. Ji, B. Wang, A short text sentiment-topic model for product reviews, Neurocomputing 297 (2018) 94–102.
[38] K. Nimala, R. Jebakumar, A robust user sentiment biterm topic mixture model based on user aggregation strategy to avoid data sparsity for short text, J. Med. Syst. 43 (2019) 93.
[39] R.K. Amplayo, S. Lee, M. Song, Incorporating product description to sentiment topic models for improved aspect-based sentiment analysis, Inf. Sci. (Ny) 454 (2018) 200–215.
[40] X. Fu, X. Sun, H. Wu, L. Cui, J.Z. Huang, Weakly supervised topic sentiment joint model with word embeddings, Knowl. Based Syst. 147 (2018) 43–54.
[41] S. Xiong, K. Wang, D. Ji, B. Wang, A short text sentiment-topic model for product reviews, Neurocomputing 297 (2018) 94–102.
[42] K. Garcia, L. Berton, Topic detection and sentiment analysis in Twitter content related to COVID-19 from Brazil and the USA. Appl. Soft Comput. 101, 107057..
[43] D.M. Blei, A.Y. Ng, M.I. Jordan, Latent dirichlet allocation, J. Mach. Learn. Res. 3 (2003) 993–1022.
[44] S. Deerwester, S.T. Dumais, G.W. Furnas, T.K. Landauer, R. Harshman, Indexing by latent semantic analysis, J. Am. Soc. Inf. Sci. 41 (1990) 391–407.
[45] T. Hofmann, Probabilistic latent semantic analysis, in: Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence, 1999, pp. 289–296.
[46] L. AlSumait, D. Barbará, C. Domeniconi, On-line lda: Adaptive topic models for mining text streams with applications to topic detection and tracking, in: Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on, 2008, pp. 3–12.
[47] Y. Wang, E. Agichtein, M. Benzi, TM-LDA: efficient online modeling of latent topic transitions in social media, in: Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining, 2012, pp. 123–131.
[48] P. Hennig, D. Stern, T. Graepel, Herbrich. R, Topic models. U. S. Patent, US 2012/0101965 A1, 2012.
[49] J. Xu, H. Li, N. Craswell, Regularized latent semantic indexing for topic modeling. U. S. Patent, US 2012/0330958 A1, 2012.
[50] B. Kim, D. Lee, J. Oh, H. Yu, Scalable disk-based topic modeling for memory limited devices, Inf. Sci. (Ny) 516 (2020) 353–369.
[51] H. Zhang, et al., Deep autoencoding topic model with scalable hybrid Bayesian inference, IEEE Trans. Pattern Anal. Mach. Intell. (2020).
[52] W. Gao, et al., Incorporating word embeddings into topic modeling of short text, Knowl. Inf. Syst. (2018) 1–23.
[53] X. Li, et al., Filtering out the noise in short text topic modeling, Inf. Sci. (Ny) 456 (2018) 83–96.
[54] H. Zhang, B. Chen, D. Guo, M. Zhou, {Whai}: Weibull hybrid autoencoding inference for deep topic modeling, in: International Conference on Learning Representations, 2018.
[55] V. Sridhar, Unsupervised topic modeling for short texts, U. S. Patent, US 2016/0110343 A1, 2016.
[56] B. Ozyurt, M.A. Akcayol, A new topic modeling based approach for aspect extraction in aspect based sentiment analysis: SS-LDA, Expert Syst. Appl. (2020) 114231.
[57] X. Zhao, et al., A neural topic model with word vectors and entity vectors for short texts, Inf. Process. Manag. 58 (2021) 102455.
[58] J. Rashid, S.M.A. Shah, A. Irtaza, Fuzzy topic modeling approach for text mining over short text, Inf. Process. Manag. 56 (2019) 102060.
[59] X. Wu, C. Li, Y. Zhu, Y. Miao, Short Text Topic Modeling with Topic Distribution Quantization and Negative Sampling Decoder, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2020, pp. 1772–1782.
[60] C. Ha, V.-D. Tran, L.N. Van, K. Than, Eliminating overfitting of probabilistic topic models on short and noisy text: The role of dropout, Internat. J. Approx. Reason. 112 (2019) 85–104.
[61] G.E. Hinton, R.R. Salakhutdinov, Replicated softmax: an undirected topic model, in: Advances in Neural Information Processing Systems, 2009, pp. 1607–1614.
[62] H. Larochelle, I. Murray, The neural autoregressive distribution estimator, in: Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, 2011, pp. 29–37.
[63] H. Larochelle, S. Lauly, A neural autoregressive topic model, in: Advances in Neural Information Processing Systems, 2012, pp. 2708–2716.
[64] P. Gupta, F. Buettner, H. Schütze, Document informed neural autoregressive topic models, 2018, arXiv Prepr. arXiv:1808.03793.
[65] J. Gao, L. Deng, M. Gamon, X. He, P. Pantel, Modeling interestingness with deep neural networks. U. S. Patent, US 2015/0363688 A1, 2015.
[66] Z. Xie, Z. Zeng, G. Zhou, W. Wang, Topic enhanced deep structured semantic models for knowledge base question answering, Sci. China Inf. Sci. 60 (2017) 110103.
[67] N. Srivastava, R.R. Salakhutdinov, Multimodal learning with deep boltzmann machines, in: Advances in Neural Information Processing Systems, 2012, pp. 2222–2230.
[68] D. Shamanta, S.M. Naim, P. Saraf, N. Ramakrishnan, M.S. Hossain, Concurrent inference of topic models and distributed vector representations, in: Joint European Conference on Machine Learning and Knowledge Discovery in Databases, 2015, pp. 441–457.
[69] Y. Yan, X.-C. Yin, B.-W. Zhang, C. Yang, H.-W. Hao, Semantic indexing with deep learning: a case study, Big Data Anal. 1 (2016) 7.
[70] Y. Li, T. Liu, J. Hu, J. Jiang, Topical co-attention networks for hashtag recommendation on microblogs, Neurocomputing 331 (2019) 356–365.
[71] F. Ali, et al., Transportation sentiment analysis using word embedding and ontology-based topic modeling, Knowledge-Based Syst. 174 (2019) 27–42.
[72] F. Nan, R. Ding, R. Nallapati, B. Xiang, Topic modeling with wasserstein autoencoders, in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, 2019, pp. 6345–6381, http://dx.doi.org/10.18653/v1/P19-1640.
[73] A. Yadav, D.K. Vishwakarma, Sentiment analysis using deep learning architectures: a review, Artif. Intell. Rev. 53 (2020) 4335–4385.
[74] D. She, J. Yang, M.-M. Cheng, Y.-K. Lai, P.L. Rosin, L. Wang, Wscnet: Weakly supervised coupled networks for visual sentiment classification and detection, IEEE Trans. Multimed. 22 (2019) 1358–1371.
[75] A. Yadav, D.K. Vishwakarma, A deep learning architecture of RA-DLNet for visual sentiment analysis, Multimed. Syst. 26 (2020) 431–451.
[76] A. Yadav, A. Agarwal, D.K, Vishwakarma, XRA-net framework for visual sentiments analysis, in: 2019 IEEE Fifth Int. Conf. Multimed. Big Data, 2019, pp. 219–224.
[77] A.R. Pathak, B. Agarwal, M. Pandey, S. Rautaray, Application of deep learning approaches for sentiment analysis, in: Deep Learning-Based Approaches for Sentiment Analysis, Springer, 2020, pp. 1–31.
[78] D.M. Blei, J.D. Lafferty, Dynamic topic models, in: Proceedings of the 23rd international conference on Machine learning, 2006, pp. 113–120.
[79] S. Ankan, A. Banerjee, S. Kasiviswanathan, R. Lawrence, P. Melville, V. Sindhwani, E. Ting, Inferring emerging and evolving topics in streaming text. U. S. Patent, US 2013/0151525 A1, 2013.
[80] A. Rekik, S. Jamoussi, Deep learning for hot topic extraction from social streams, in: International Conference on Hybrid Intelligent Systems, 2016, pp. 186–197.
[81] K. Giannakopoulos, L. Chen, Incremental and adaptive topic detection over social media, in: International Conference on Database Systems for Advanced Applications, 2018, pp. 460–473.
[82] G. Tang, Y. Xia, Adaptive topic modeling with probabilistic pseudo feedback in online topic detection, in: International Conference on Application of Natural Language to Information Systems, 2010, pp. 100–108.
[83] X. Fu, J. Li, K. Yang, L. Cui, L. Yang, Dynamic online HDP model for discovering evolutionary topics from chinese social texts, Neurocomputing 171 (2016) 412–424.
[84] D. Tu, L. Chen, M. Lv, H. Shi, G. Chen, Hierarchical online NMF for detecting and tracking topic hierarchies in a text stream, Pattern Recognit. 76 (2018) 203–214.
[85] H. Yin, B. Cui, H. Lu, Y. Huang, J. Yao, A unified model for stable and temporal topic detection from social media data, in: 2013 IEEE 29th International Conference on Data Engineering (ICDE), 2013, pp. 661–672.
[86] M. Ahn, et al., On large-scale dynamic topic modeling with nonnegative cp tensor decomposition, 2020, arXiv Prepr. arXiv:2001.00631.

[87] Y. Zhao, B. Qin, T. Liu, D. Tang, Social sentiment sensor: a visualization system for topic detection and topic sentiment analysis on microblog, Multimedia Tools Appl. 75 (2016) 8843–8860.
[88] X. Fu, G. Liu, Y. Guo, W. Guo, Multi-aspect blog sentiment analysis based on LDA topic model and hownet lexicon, in: International Conference on Web Information Systems and Mining, 2011, pp. 131–138.
[89] X. Chen, Y. Zhang, J. Xu, C. Xing, H. Chen, Deep learning based topic identification and categorization: mining diabetes-related topics on chinese health websites, in: International Conference on Database Systems for Advanced Applications, 2016, pp. 481–500.
[90] Y. Zhang, et al., Does deep learning help topic extraction? A kernel k-means clustering method with word embedding, J. Informetr. 12 (2018) 1099–1117.
[91] Y. Ren, R. Wang, D. Ji, A topic-enhanced word embedding for Twitter sentiment classification, Inf. Sci. (Ny) 369 (2016) 188–198.
[92] P. Kalarani, S.S. Brunda, Sentiment analysis by POS and joint sentiment topic features using SVM and ANN, Soft Comput. 23 (2019) 7067–7079.
[93] C. Montenegro, C. Ligutom III, J.V. Orio, D.A.M. Ramacho, Using Latent Dirichlet Allocation for Topic Modeling and Document Clustering of Dumaguete City Twitter Dataset, in: Proceedings of the 2018 International Conference on Computing and Data Engineering, 2018, pp. 1–5.
[94] B. Kaddouri, M. Mataoui, Domain-level topic detection approach for improving sentiment analysis in arabic content, in: International Conference on Computer Science and its Applications, 2018, pp. 135–146.
[95] G. Pergola, L. Gui, Y. He, TDAM: A topic-dependent attention model for sentiment analysis, Inf. Process. Manag. 56 (2019) 102084.
[96] R. do Carmo, A.M. Lacerda, D.H. Dalip, A Majority Voting Approach for Sentiment Analysis in Short Texts using Topic Models, in: Proceedings of the 23rd Brazillian Symposium on Multimedia and the Web, 2017, pp. 449–455.
[97] NRC Word-Emotion Association Lexicon, http://saifmohammad.com/WebPages/NRC-Emotion-Lexicon.htm.
[98] R. Baly, et al., Omam at semeval-2017 task 4: Evaluation of english state-of-the-art sentiment analysis models for arabic and a new topic-based model, in: Proceedings of the 11th international workshop on semantic evaluation (SEMEVAL-2017), 2017, pp. 603–610.
[99] A. Rozental, D. Fleischer, Amobee at SemEval-2017 task 4: Deep learning system for sentiment detection on Twitter, 2017, arXiv Prepr. arXiv:1705.01306.
[100] J.-A. González, F. Pla, L.-F. Hurtado, ELiRF-UPV at SemEval-2017 task 4: sentiment analysis using deep learning, in: Proceedings of the 11th international workshop on semantic evaluation (SEMEVAL-2017), 2017, pp. 723–727.
[101] H. Zhang, et al., EICA at SemEval-2017 Task 4: A Simple Convolutional Neural Network for Topic-based Sentiment Classification, in: Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017), 2017, pp. 723–727.
[102] H. Zhang, J. Wang, J. Zhang, X. Zhang, Ynu-hpcc at semeval 2017 task 4: Using a multi-channel cnn-lstm model for sentiment classification, in: Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017), 2017, pp. 796–801.
[103] spaCy library, https://spacy.io.
[104] Gensim, https://radimrehurek.com/gensim/.
[105] T. Mikolov, I. Sutskever, K. Chen, G.S. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, in: Advances in Neural Information Processing Systems, 2013, pp. 3111–3119.
[106] J. Pennington, R. Socher, C. Manning, Glove: Global vectors for word representation, in: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), 2014, pp. 1532–1543.
[107] L.-C. Yu, J. Wang, K.R. Lai, X. Zhang, Refining word embeddings for sentiment analysis, in: Proceedings of the 2017 conference on empirical methods in natural language processing (2017) 534–539.
[108] S.M. Rezaeinia, R. Rahmani, A. Ghodsi, H. Veisi, Sentiment analysis based on improved pre-trained word embeddings, Expert Syst. Appl. 117 (2019) 139–147.
[109] Scikit-learn, https://scikit-learn.org.
[110] A.R. Pathak, M. Pandey, S.S. Rautaray, Adaptive framework for deep learning based dynamic and temporal topic modeling from big data, Recent Patents Eng. 13 (2019) 1, http://dx.doi.org/10.2174/1872212113666190329234812.
[111] A.R. Pathak, M. Pandey, S.S. Rautaray, Adaptive model for dynamic and temporal topic modeling from big data using deep learning architecture, Int. J. Intell. Syst. Appl. 11 (6) (2019) 13–27, http://dx.doi.org/10.5815/ijisa.2019.06.02.
[112] S. Ruder, P. Ghaffari, J.G. Breslin, Insight-1 at semeval-2016 task 5: Deep learning for multilingual aspect-based sentiment analysis, 2016, arXiv Prepr. arXiv:1609.02748.
[113] Y. Wang, M. Huang, L. Zhao, Others, Attention-based lstm for aspect-level sentiment classification, in: Proceedings of the 2016 conference on empirical methods in natural language processing, 2016, pp. 606–615.
[114] T. Rocktäschel, E. Grefenstette, K.M. Hermann, T. Kočiskỳ, P. Blunsom, Reasoning about entailment with neural attention, 2015, arXiv Prepr. arXiv:1509.06664.
[115] Python, https://www.python.org.
[116] Keras, https://keras.io.
[117] TensorFlow, https://www.tensorflow.org.
[118] Google Cloud Platform, https://cloud.google.com.
[119] A.R. Pathak, Social media data #bitcoin #ethereum #facebook, Mendeley Data, 2019, http://dx.doi.org/10.17632/chx9mdyydb.1.
[120] R.J. Williams, J. Peng, An efficient gradient-based algorithm for on-line training of recurrent network trajectories, Neural Comput. 2 (1990) 490–501.
[121] H.-G. Zimmermann, C. Tietz, R. Grothmann, Forecasting with recurrent neural networks: 12 tricks, in: Neural Networks: Tricks of the Trade, Springer, 2012, pp. 687–707.
[122] F. Sebastiani, An axiomatically derived measure for the evaluation of classification algorithms, in: Proceedings of the 2015 International Conference on The Theory of Information Retrieval, 2015, pp. 11–20.