Twitter Flu Trend: A Hybrid Deep Neural Network for Tweet Analysis

Mahsa Abazari Kia[1][0000−0003−1842−9270] and Fatemeh Ebrahimi Khaksefidi[2]

[1] School of Computer Science and Electronic Engineering, University of Essex, United Kingdom ma19194@essex.ac.uk
[2] University of Isfahan, Isfahan, Iran
ebrahimi.f72@gmail.com

Abstract. Popular social networks such as Twitter have been proposed as a data source for public health monitoring because they have the potential to show infection disease surveillance like Influenza-Like Illnesses (ILI). However, shortness, data sparsity, informality, incorrect sentence structure, and the humorous are some challenges for tweet analysis and classification. In order to overcome these challenges and implement an accurate flu surveillance system, we propose a hybrid 1d-CNN-BiLSTM framework for semantic enrichment and tweet classification. Different embedding algorithms are compared for producing semantic representations of tweets to assist unrelated tweet filtering in the classification stage. We find that fine-tuning pre-trained Word2Vec enhances the model capability for representing the meaning of flu-related tweets than other embedding models. Our approach has been evaluated on a flu tweet dataset and compared with several baselines for tweet processing and classification. Experimental results show that: (1) the proposed hybrid deep neural networks can improve tweet classification due to considering their semantic information;(2) the proposed flu surveillance system achieves a state-of-the-art correlation coefficient with ILI rate published by CDC [3].

Keywords: Tweet analysis · Hybrid deep Neural Network · CNN-BiLSTM · Flu-trend .

1 Introduction

Public health condition monitoring is important for health organizations and governments to respond to emerging epidemics [5]. U.S. Centers for Disease Control and Prevention (CDC) aggregate data from a network of a huge number of outpatient providers across the united states to provide the count of Influenza-Like Illnesses (ILI) rates and infection level monitoring for each previous week [3]. Delays and costs for identifying the start of an infection epidemic in these systems of disease surveillance systems cause big damage to society [12]. Strong interest in reducing these delays and costs and a way to develop an accurate

[3] https://www.cdc.gov/

surveillance system brings us to use real-time and low-cost data sources. With the growth and popularity of social networks in the world, such as Twitter, we are going to use them as valuable data sources that can fulfill the implementation of on-time and low-cost ILI surveillance systems. But the analysis of tweets faces some challenges, such as shortness and informality, humor, incorrect sentence structure, ambiguity, and so on, which poses difficulties in using tweets and causes some drawbacks in analysis and classification [13]. These tweets' characteristics make it challenging to detect influenza infection tweets. For example, tweets like "I feel so sick I got Bieber fever" and "I had a dream that I had a terrible flu last night. I woke up and clearing my throat extra" are classified as ILI infection tweets because of the existence of ILI related words and complexity in understanding the meaning [2]. In this way, humorous and ambiguous tweets, tweets that look like ILI infection tweets but their meaning is unrelated to ILI infection, are easily counted as ILI infection tweets which cause incorrect results and overestimating or underestimating of ILI rate. Adding semantics to tweet representation may be needed to deal with these kinds of tweets. Most of the researchers that have tried to add semantics to tweets have used entity extraction and knowledge source linking, which is appropriate for limited fields such as sentiment analysis, topic classification, incident detection, etc. For this kind of enrichment, there should be some named entities that improvement of analysis depends on their accurate extraction and linking to their related concepts in knowledge sources. This paper proposes a framework for implementing a flu surveillance system using deep learning approaches. We investigate embedding algorithms and deep neural networks for semantic enrichment and analyzing flu tweets. We studied deep learning approaches for semantic classification and designed a framework to track flu trends in the U.S. We used pre-trained Word2Vec and GloVe for generating tweet representation. We used a hybrid 1d-CNN and BiLSTM network for tweet classification, which uses convolution as a feature extractor and then BiLSTM as a sequence model to treat extracted features as a sequence and classifies the tweets. We evaluate the proposed framework on a dataset that is prepared especially for the case of flu and is collected using a boolean query containing standard flu symptoms introduced by [25]. We compare our results with previous works which aim to calculate ILI rates. Also, we use MNB and SVM with n-grams, 1d-CNN, BiLSTM, and LSTM as baselines. Experimental results show that the proposed framework achieves state-of-the-art correlation with the ground truth data (CDC ILI rate). This shows that embedding algorithms and hybrid deep neural networks can improve tweet classification by considering the meaning of tweets. The main contribution of our work is summarized below:

- While previous flu tweet classification researches used conventional methods, we studied the deep learning algorithms' performance for tweet semantic enrichment and classification.
- We proposed a novel framework, "Twitter Flu Trend", for exploring flu trends in the U.S. by fine-tuning the pre-trained Word2Vec model for learn-

ing word vectors that contain both syntactic and semantic information and a combinational 1d-CNN-BiLSTM classifier.
– We have created a Flu-related tweet dataset by collecting old tweets utilizing a boolean query containing standard flu symptoms. We have labeled the tweets to Flu-infectious and Non-Flu-infectious.

2 Related Work

We classify the related works into three main sections: Researches about ILI surveillance using Twitter, semantic enrichment via knowledge sources, and semantic representation with deep learning.

2.1 ILI Surveillance Using Twitter

In the area of social media data mining, Twitter data provided lots of applications and valuable insights into various fields, but there is limited study on the public health information system, specially influenza-like illnesses which is a significant infectious disease. The most famous study is [8], which designed the Google Flu Trend surveillance system tried to estimate the level of flu activity by analyzing the queries that were searched on Google. Flu trend estimated by computing the search frequency of 45 selected queries all around the world which is no longer publishing. Other researches are based on Twitter; some of them are done manually and have not used classification for detecting flu infection tweets automatically like [12] that presented an approach for differentiating between flu infection, concerns, and awareness tweets by relying on a set of features and templates which is defined manually. Doan et al. [5] developed a novel filtering method for ILI-related tweets, which filters tweets based on syndrome keywords from BioCaster ontology and semantic features such as negation, hashtags, emoticons, humor, and geography. Broniatowski et al. [3] created a two-phased supervised classification model to separate flu-related tweets from health-related tweets in the first phase and flu infection tweets in the second. Allen et al. [1] Proposed an approach that uses n-gram for generating features and an SVM classifier for classifying tweets that contain flu and influenza terms. Jain et al. [10] introduced a novel approach that produces dynamic keywords for collecting flu-related tweets and then classifies tweets with SVM for recognizing flu infection tweets. Velardi et al. [25] developed an approach that produces slang synonym clusters of flu symptoms and proves that using flu symptoms is more helpful in retrieving flu infection messages than using flu names. The synonyms are used to extend the standard symptom-based query for influenza-related disease defined by European CDC to collect tweets indicating infection.

2.2 Semantic Enrichment with Knowledge Sources

Knowledge sources can be used at various stages in data mining processes for various purposes, such as creating additional variables [21]. Schulz et al. [23]

proposed a model for semantic enrichment of tweets using DBpedia to improve incident tweets identification. Saif et al. [22] presented a semantic sentiment classification method that extracts entities from tweets and their semantic concepts from DBpedia as additional features. Varga et al. [24] used semantic concepts of knowledge sources related to the extracted entities from short texts to enhance classification or clustering tasks. All these researches used semantic concepts from DBpedia or other knowledge sources to fix the ambiguity of those entities that have several names, and they have to be homogenized to improve data mining tasks such as sentiment analysis, topic classification, or car crash classification which is dependent on correct diagnosis of named entities. Using the type and attributes of semantic concepts creates a semantic set of features which is useful for completing lexical features for these tasks.

2.3 Semantic Representation with Deep Learning

Deep learning approaches can automatically capture the text's syntactic and semantic features simultaneously without feature engineering, which is laborious and time-consuming. They have drawn much attention in natural language processing (NLP) and achieved state-of-the-art performances [4]. Wang et al. [26] proposed a framework to expand short texts based on word embedding clustering and convolutional neural network to overcome the sparsity and semantic sensitivity to context in short texts. Gatti et al. [6] explored the richness of word embeddings produced by unsupervised pre-training, a deep convolutional neural network proposed to exploits character to sentence-level information and perform sentiment analysis of short texts. Edouard et al. [7] exploited information acquired from external knowledge bases to enrich Named Entities (NEs) mentioned in the tweets. Then enriched content is used for building word-embedding vectors, which serve as feature models for training supervised models for event classification, Naïve Bayes, SVM, and LSTM classification algorithms have been compared. Wang et al. [27] developed a CNN-LSTM model consisting of two parts, regional CNN and LSTM, for dimensional sentiment analysis. Unlike a conventional CNN, the proposed regional CNN divides an input text into several regions, and then regional information is integrated using LSTM for prediction.

3 Our Method

In this section, we present our approach for improving tweet classification in case of ILI infection via hybrid deep neural networks. An overview of the proposed hybrid deep neural network is shown in Figure 2 . We first introduce word embedding as a class of techniques for generating distributional representations of words and their models in section 3.1 and then we describe the 1d-CNN model, which produces features of a tweet via convolution layers (section 3.2). The BiLSTM model is described in section 3.3, which predicts tweet labels in our hybrid network.

3.1 Word Embedding

Word embeddings is a class of techniques in which words are represented as vectors with real values in a predefined space. Each word is mapped into a vector, and vector values represent different aspects of a word mainly learned through context [16]. Word vectors capture general syntactical and semantic information [14], and the main advantage of distributional vectors is that they capture the similarity between words [9]. Thus it has been proven that embeddings are efficient in capturing context similarity and analogies [18]. In this research, we use Word2Vec and GloVe algorithms for producing word vectors to evaluate them for producing the semantic representation of words through their context to filter ambiguous tweets (containing flu-related terms, but their meaning is unrelated to flu).

Word2Vec Skip-gram and CBOW are the two models of Word2Vec, which are proposed by Mikolov et al. [17]. Skip-gram computes the conditional probability of the context words surrounding the target word in both directions across a window with size k. On the other hand, the CBOW model aims to maximize the Formula 1 while the skip-gram tries to maximize Formula 2. In the Formula 1 and 2, v corresponds to vocabulary size, m_t refers to the target word, m_j refers to context words, and c is the window size. Moreover, negative sampling and hierarchical Softmax are the two algorithms for learning the output vectors of CBOW and skip-gram.

$$\frac{1}{v} \sum_{t=1}^{v} \log p\left(m_t \mid m_{t-\frac{c}{2}} \ldots m_{t+\frac{c}{2}}\right) \tag{1}$$

$$\frac{1}{v} \sum_{t=1}^{v} \sum_{j=t-c, j=t}^{t+c} \log p\left(m_j \mid m_t\right) \tag{2}$$

GloVe GloVe is another famous word embedding method which is based on word occurrences in textual corpus and proposed by Pennington et al. [20]. This method is based on two main steps, the first is constructing matrix X from training corpus as a co-occurrence matrix where X_{ij} is the frequency of the word i co-occurring with the word j, second is factorization of X in order to get the embedding vectors. Using Ratios instead of raw probabilities helps to reduce noise by identifying relevant words from irrelevant which is shown in Equation 3.

$$F\left(\mathcal{W}_i - w_j, \varpi_k\right) = \frac{p_{ik}}{p_{jk}} \tag{3}$$

$p_{ik} = X_{ik}/X_i$ is the occurrence probability of word k in the context of the word i. $\mathcal{W}$ are word vectors and ϖ_k are context word vectors. They used vector differences and the dot product of the arguments for preventing mixing dimensions and preserving linearity as depicted in Equation 4.

$$F\left(\left(w_i - w_j\right)^T \varpi_k\right) = \frac{p_{ik}}{p_{jk}} \tag{4}$$

F is assumed as a function and for resolving symmetry, the Equation 4 can be re-written in a different way:

$$w_i^T \varpi_k + b_i + b_k = \log\left(x_{ik}\right) \tag{5}$$

Finally an objective function is proposed that should be minimized by Equation 6 where $f(x)$ is weighting function.

$$J = \sum_{i,j=1}^{v} f\left(x_{ij}\right)\left(w_i^T \varpi_j + b_i + b_j - \log\left(x_{ij}\right)\right)^2 \tag{6}$$

3.2 Convolutional Neural Network (CNN)

1d-CNN, firstly proposed by Kim et al. [11], takes sentences of varying length as input which is the concatenation of word vectors. If $w_i \in R^d$ refers to word embedding of the i^{th} word in the sentence, where d is the dimension of word embedding and $w_{i:i+j}$ refers to the concatenation of vectors $w_i, w_{i+1}, \ldots, w_j$. A number of filters, also called kernel, with different window size move on the word embeddings to perform one-dimensional convolution and create feature maps. Each filter extracts a specific pattern of n-gram. For example a filter $k \in R^{hd}$ produces a feature c_i with using the window of words $w_{i:i+h-1}$:

$$c_i = f\left(w_{i:i+h-1} \cdot k^T + b\right) \tag{7}$$

Here, b is the bias term and f is a non-linear activation function. The filter k is applied to all possible windows using the same weights to create the feature map [28].

$$c = [c_1, c_2, ..., c_{n-h+1}] \tag{8}$$

The next layer usually is max-pooling layer. In this layer max pooling operation, $\hat{c} = max[c]$, captures the most useful local features from feature maps. The outputs of multiple filters which is operated by max-pooling layer are concatenated in the next layer to form a single feature vector. Finally, a fully connected Softmax layer generates the probability distribution over labels.

3.3 Bidirectional Long Short Term Memory (BiLSTM)

Bidirectional Long Short Term Memory (BiLSTM) is putting two independent LSTM together and LSTM [28] is a modified Recurrent neural networks (RNN) architecture. RNN [29] use the idea of processing sequential information. The term recurrent means that a same task is performed over each instance of the sequence such that the output is dependent on previous computations and results [28]. Simple RNNs are consisting of a memory known as hidden state s which maintain previous computations and an optional output y. At each time step t the hidden state s_t is computed based on the previous hidden state s_{t-1} and the input at current time step x_t:

$$S_t = f\left(ux_t + ws_{t-1}\right) \tag{9}$$

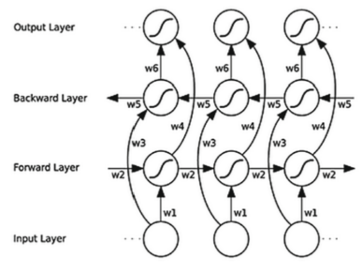

Fig. 1. Bidirectional LSTM architecture [29]

f is taken to be a nonlinear activation function such as $tanh$, $ReLU$ and u, w account for weights that are shared across time. The output at time step t is computed as $y_t = softmax(vh_t)$ where v is another shared weight parameter of network and is an activation function often implemented at the final layer of a network. In practice the simple RNNs networks suffer from vanishing gradient problem that LSTM, a RNN variant, designed to deal with this and overcome the limitations of simple RNNs such as processing long sequences with long-term temporal dependencies [15]. LSTM has three gates: input, forget and output gates and hidden state is calculated based on the combination of these three gates as per Equations below:

$$x = \begin{bmatrix} h_{t-1} \\ x_t \end{bmatrix} \tag{10}$$

$$f_t = \sigma\left(w_f.x + b_f\right) \tag{11}$$

$$i_t = \sigma\left(w_i.x + b_i\right) \tag{12}$$

$$o_t = \sigma\left(w_o.x + b_o\right) \tag{13}$$

$$c_t = f_t \otimes c_{t-1} + i_t \otimes \tanh\left(W_c \cdot X + b_c\right) \tag{14}$$

$$h_t = o_t \oplus \tanh\left(c_t\right) \tag{15}$$

BiLSTM uses two LSTMs to learn each token of sequence based on both the right and the left context of the token. As shown in Figure 1 one LSTM process the sequence from left to right and the other one from right to left [19]. Output W at each time step is dependent on both forward and backward layers. Forward layer processes the left context of the input and backward layer processes the right context.

3.4 Hybrid Neural Network

We combine 1d-CNN and BiLSTM in a way that features are extracted by one-dimensional convolution. Multiple filters with different sizes and max-pooling have been applied at the first stage and then passed to the BiLSTM as a sequence input. BiLSTM network does the classification based on generated feature maps which indicate the features of the input. Finally, a fully connected layer and Softmax regression are used to output the probability for each class.

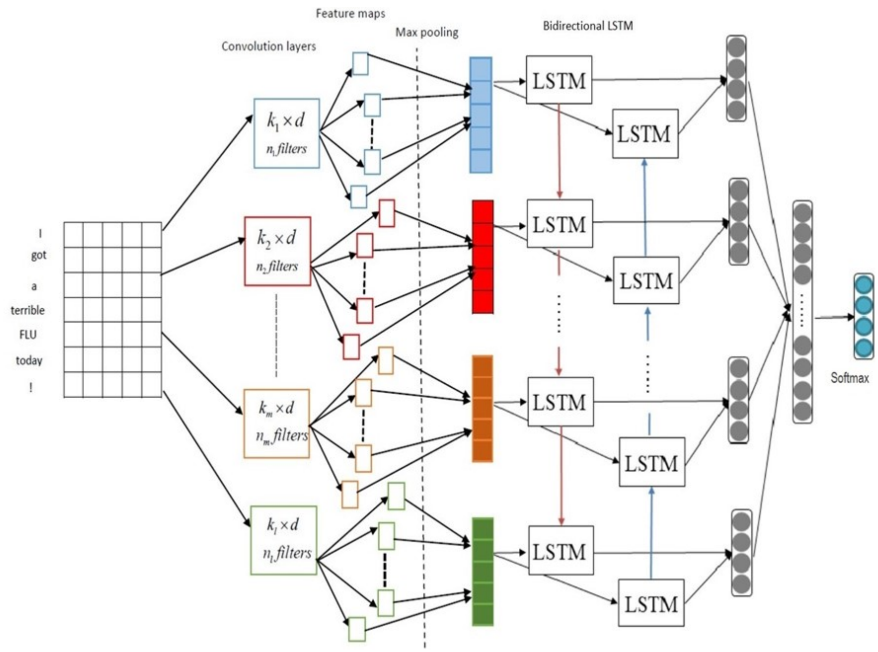

Fig. 2. The proposed hybrid 1d-CNN BiLSTM for generating tweet semantic representation and classification

4 Experiments

4.1 Experimental Setup

For training 1d-CNN-BiLSTM, we prepared a dataset using a java program that collects tweets by connecting to Twitter search and applying a time span and a Boolean query. The prepared dataset is filtered by location to build a training dataset with only a U.S. location. Accessing a great number of tweets is one of the limitations in using standard Twitter APIs that the java program tries to overcome. We use a symptom-based query for fetching flu-related tweets because using symptoms is more preferred by people than using disease names. The query is based on standard flu definition, which is symptom-based and defined by CDC. The query is expanded by using slang synonyms for symptoms introduced in [25]. The expanded query contains both technical and slang terms, which are used for collecting tweets from September 2019 to March 2020 (influenza peak season). We give a positive label to tweets that indicate infection and a negative label to other tweets. Other tweets contain awareness, ambiguous, satirical tweets in which the Boolean query is true for them, but they do not declare flu infection. Also, two different states are considered for embedding production.

– Pre-trained Word2Vec: initialize word vectors with the Google's pre-trained Word2Vec model and adjust the values through the training process. The

pre-trained word embeddings were learned on the part of the Google News dataset, which contains 300-dimensional vectors for 3 million words and phrases.
- Pre-trained GloVe: initialize word vectors with twitter's pre-trained GloVe model and adjust the values through the training process. The pre-trained word embeddings were trained over 2 billion tweets with an embedding size of 200.

At the first stage for 1d-CNN, we use filter windows of 2,3,4,5 with 256 feature maps each, ReLU as an activation function, and a general max pooling. At the next stage for training BiLSTM, the input sequence is features extracted from input tweets through the convolution layer, followed by the max-pooling layer. u, v, w and are initialized to a random vector of small values. Finally, a fully connected layer and Softmax regression are used to output the probability for each class. A back-propagation algorithm with the Adam optimization method is used to train the network through time. After each training epoch, the network is tested on validation data.

4.2 Baseline Methods

To make strong comparisons, nine popular methods which are commonly used for tweet classification are utilized as baselines. In experiments, we evaluate all methods on our benchmarks and separate them into deep neural network and conventional models. Some brief introductions of baseline methods are given below:
MNB-uni: multinomial Naïve Bayes with unigrams as features and TF-IDF weighting.
MNB-uni-bi: multinomial Naïve Bayes with uni-bigrams as features and TF-IDF weighting.
MNB-uni-bi-tri: multinomial Naïve Bayes with uni-bi-trigrams as features and TF-IDF weighting.
SVM-uni: SVM with unigrams as features and TF-IDF weighting.
SVM-uni-bi: SVM with uni-bigrams as features and TF-IDF weighting.
SVM-uni-bi-trigrams: SVM with uni-bi-trigrams as features and TF-IDF weighting.
1d-CNN: one dimensional CNN with two different embeddings and parameters, given in the previous section.
BiLSTM: Bidirectional LSTM with two different embeddings and parameters, given in the previous section.
LSTM: LSTM with two different embeddings and the same parameters given for BiLSTM.

5 Results

All results are obtained under the same distribution of experimental data, 90% for training and 10% for testing. In Table 1 it can be observed that the best

Table 1. Results comparison of deep neural network baseline methods and conventional tweet classification baselines.

Model	Accuracy				
	Word2Vec Pre-trained	GloVe Pre-trained	Unigrams	Uni-bigrams	Uni-bi-trigrams
1d-CNN	0.91	0.86			
BiLSTM	0.86	0.84			
LSTM	0.89	0.87			
1d-CNN-BiLSTM(Ours)	0.92	0.87			
MNB			0.80	0.82	0.83
SVM			0.81	0.80	0.79

performance of all models obtained with Word2Vec embedding. It can be seen that our approach achieved the highest accuracy in all embedding states in comparison to other baseline models and the second-best model is 1d-CNN. It is obvious that single BiLSTM performed poorly but when it is combined with 1d-CNN it enhanced the accuracy. Using only the left context information of a sequence in LSTM, hinder accurate processing for text sequences, because both right and left contexts are important for the sequences and this is the reason that we didn't use LSTM in a combinational model as a baseline. MNB has better performance with combinational n-grams than SVM and its best result is achieved by uni-bi-trigrams which is 0.02% better than the best of SVM. The salient difference, about 0.09%, can be seen between the accuracy of our approach with Word2Vec and MNB with un-bi-trigrams which demonstrates flu tweets classification is a challenging task for conventional classification models. Table 2 shows four examples of tweets that cause conventional models to perform poorly because of negation, ambiguity, sense of humor, and complexity in awareness tweets detection. As it is clear our approach can classify these kinds of tweets more accurately due to fine-tuning pre-trained Word2Vec and utilizing a hybrid classifier benefiting both BiLSTM and 1d-CNN properties.

Table 2. Four examples of tweets that conventional models cannot classify them accurately.

Negation	Homesick for day 2. I was awake at 4 am this morning and still can't sleep. Not the flu but an ugly cough and... https://fb.me/74P04sXKL
Ambiguity	I had a dream that I had a terrible flu last night. I woke up and clearing my throat extra
Humorous	I feel so sick, I have Bieber fever
Awareness	Dry, chesty cough, sore throat? How to get rid of the Aussie flu symptom - and if cough medicine doesn't work go http://newspaper-report.today/2018/01/dry-chesty-cough-how-to-get-rid-of-the-aussie-flu-symptom-and-if-cough-medicine-works/ . . .

5.1 Correlation Evaluation

For evaluating the proposed approach in the real world, a set of new tweets is collected during the 2020-2021, and 2021-2022 influenza peak season for analyzing and classification to estimate the Pearson correlation coefficient between Twitter Infection rate (obtained by our approach) and ILINET (flu infection rate published by CDC). The Pearson correlation coefficient between ILINET and Twitter infection rate (Twitter Flu Trend) is 0.96 for 2020-2021 and 0.97 for 2021-2022 that are shown in Figure 3. Vertical axis at the right-hand shows weighted ILINET and the left hand axis indicates the number of tweets expressing infection. The horizontal axis shows the number of week in the year, and as reported by CDC the influenza season peak is usually between the week 48 and week 13. We can see the same peak points, ascending and descending trends in Figure 3 for both rates which shows the proposed approach estimation is higly correlated with the real ILI rate. Comparison of previous flu-related works and our approach is shown in Table 3. It can be observed that the highest correlation belongs to [5] and [25] which used manual classification and they are not proper for producing an automated ILI surveillance system. Google Flu Trend [8] is the only flu surveillance system that is no longer published and its mean correlation is 0.90. In comparison with other works that used a classifier for estimating Twitter flu rate, we can say our approach achieved state-of-the-art performance in both accuracy and correlation, which conduct us to implement an ILI surveillance system using Twitter with the name Twitter Flu Trend. This ILI surveillance system is publishing U.S. ILI daily rate and shows the U.S. ILI trend from the beginning of 2023.

Table 3. Correlation coefficient and accuracy comparison between the proposed approach and flu-related methods.

Model	Correlation coefficient	Classification algorithm	Accuracy
Lamb et al.[12]	0.79	manual	
Broniatowski et al.[3]	0.93	SVM	
Allen et al. 2016[1]	0.70	SVM	0.78
Ginsberg et al.[8]	0.90		
Jain et al.[10]		SVM	0.77
Aramaki et al.[2]	0.89	SVM	0.76
Doan et al.[5]	0.98	manual	
Velardi et al.[25]	0.98	manual	
Proposed approach	**0.97**	**1d-CNN-BiLSTM**	**0.92**

6 Conclusion

Twitter offers unique challenges and opportunities for monitoring and surveillance of public health. We have presented a method for tracking the flu epidemic

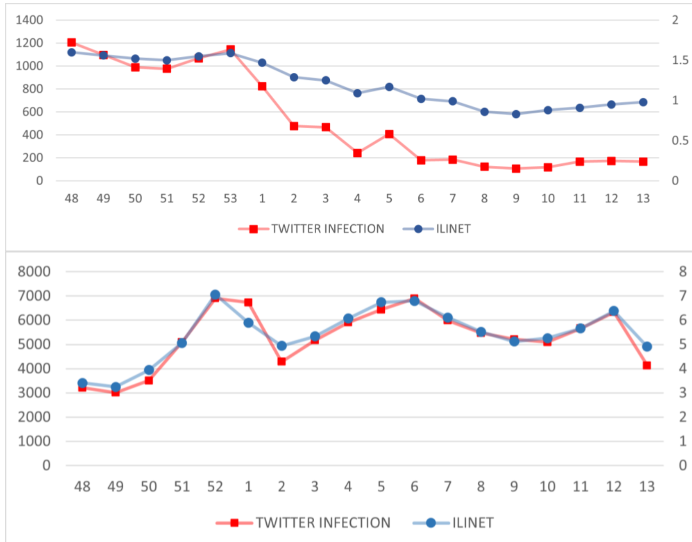

Fig. 3. The upper plot:Pearson correlation coefficient between ILINET and Twitter infection rate between week 48-2021 and week 13-2022. The lower plot:Pearson correlation coefficient between ILINET and Twitter infection rate between week 48-2020 and week 13-2021.

in the U.S. using a hybrid deep neural network for analyzing tweets. Our model is capable of differentiating between reports of actual infection and Twitter chatter by utilizing the advantages of combining 1d-CNN and BiLSTM and semantic embedding vectors. Our Twitter infection rate correlates strongly with CDC ILI data. In addition, we have demonstrated the ability to use this technique for designing an ILI surveillance system called "Twitter Flu Trend," which calculates the U.S. ILI daily rate. Also, ILI intensity calculation for each U.S. state is our future work for detailed infection level monitoring.

References

1. Allen, C., Tsou, M.H., Aslam, A., Nagel, A., Gawron, J.M.: Applying gis and machine learning methods to twitter data for multiscale surveillance of influenza. PloS one **11**(7), e0157734 (2016)
2. Aramaki, E., Maskawa, S., Morita, M.: Twitter catches the flu: detecting influenza epidemics using twitter. In: Proceedings of the 2011 Conference on empirical methods in natural language processing. pp. 1568–1576 (2011)
3. Broniatowski, D.A., Paul, M.J., Dredze, M.: National and local influenza surveillance through twitter: an analysis of the 2012-2013 influenza epidemic. PloS one **8**(12), e83672 (2013)
4. Chen, T., Xu, R., He, Y., Wang, X.: Improving sentiment analysis via sentence type classification using bilstm-crf and cnn. Expert Systems with Applications **72**, 221–230 (2017)

5. Doan, S., Ohno-Machado, L., Collier, N.: Enhancing twitter data analysis with simple semantic filtering: Example in tracking influenza-like illnesses. In: 2012 iEEE second international conference on healthcare informatics, imaging and systems biology. pp. 62–71. IEEE (2012)
6. Dos Santos, C., Gatti, M.: Deep convolutional neural networks for sentiment analysis of short texts. In: Proceedings of COLING 2014, the 25th international conference on computational linguistics: technical papers. pp. 69–78 (2014)
7. Edouard, A., Cabrio, E., Tonelli, S., Le Thanh, N.: Semantic linking for event-based classification of tweets. International Journal of Computational Linguistics and Applications p. 12 (2017)
8. Ginsberg, J., Mohebbi, M.H., Patel, R.S., Brammer, L., Smolinski, M.S., Brilliant, L.: Detecting influenza epidemics using search engine query data. Nature **457**(7232), 1012–1014 (2009)
9. Goldberg, Y.: Neural network methods for natural language processing. Synthesis lectures on human language technologies **10**(1), 1–309 (2017)
10. Jain, V.K., Kumar, S.: An effective approach to track levels of influenza-a (h1n1) pandemic in india using twitter. Procedia Computer Science **70**, 801–807 (2015)
11. Kim, Y.: Convolutional neural networks for sentence classification. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). pp. 1746–1751 (2014)
12. Lamb, A., Paul, M., Dredze, M.: Separating fact from fear: Tracking flu infections on twitter. In: Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. pp. 789–795 (2013)
13. Li, H., Ji, H., Zhao, L.: Social event extraction: Task, challenges and techniques. In: Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015. pp. 526–532 (2015)
14. Li, Y., Yang, T.: Word embedding for understanding natural language: a survey. In: Guide to big data applications, pp. 83–104. Springer (2018)
15. Manaswi, N.K., Manaswi, N.K., John, S.: Deep learning with applications using python. Springer (2018)
16. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781 (2013)
17. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. Advances in neural information processing systems **26** (2013)
18. Mikolov, T., Yih, W.t., Zweig, G.: Linguistic regularities in continuous space word representations. In: Proceedings of the 2013 conference of the north american chapter of the association for computational linguistics: Human language technologies. pp. 746–751 (2013)
19. Nowak, J., Taspinar, A., Scherer, R.: Lstm recurrent neural networks for short text and sentiment classification. In: International Conference on Artificial Intelligence and Soft Computing. pp. 553–562. Springer (2017)
20. Pennington, J., Socher, R., Manning, C.D.: Glove: Global vectors for word representation. In: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). pp. 1532–1543 (2014)
21. Ristoski, P., Paulheim, H.: Semantic web in data mining and knowledge discovery: A comprehensive survey. Journal of Web Semantics **36**, 1–22 (2016)
22. Saif, H., He, Y., Alani, H.: Semantic sentiment analysis of twitter. In: International semantic web conference. pp. 508–524. Springer (2012)

23. Schulz, A., Ristoski, P., Paulheim, H.: I see a car crash: Real-time detection of small scale incidents in microblogs. In: Extended semantic web conference. pp. 22–33. Springer (2013)
24. Varga, A., Basave, A.E.C., Rowe, M., Ciravegna, F., He, Y.: Linked knowledge sources for topic classification of microposts: A semantic graph-based approach. Journal of web semantics **26**, 36–57 (2014)
25. Velardi, P., Stilo, G., Tozzi, A.E., Gesualdo, F.: Twitter mining for fine-grained syndromic surveillance. Artificial intelligence in medicine **61**(3), 153–163 (2014)
26. Wang, J., Yu, L.C., Lai, K.R., Zhang, X.: Dimensional sentiment analysis using a regional cnn-lstm model. In: Proceedings of the 54th annual meeting of the association for computational linguistics (volume 2: Short papers). pp. 225–230 (2016)
27. Wang, P., Xu, B., Xu, J., Tian, G., Liu, C.L., Hao, H.: Semantic expansion using word embedding clustering and convolutional neural network for improving short text classification. Neurocomputing **174**, 806–814 (2016)
28. Young, T., Hazarika, D., Poria, S., Cambria, E.: Recent trends in deep learning based natural language processing. ieee Computational intelligenCe magazine **13**(3), 55–75 (2018)
29. Yu, Y., Si, X., Hu, C., Zhang, J.: A review of recurrent neural networks: Lstm cells and network architectures. Neural computation **31**(7), 1235–1270 (2019)