Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020), pages 5046–5054
Marseille, 11–16 May 2020

Multi-domain Tweet Corpora for Sentiment Analysis: Resource Creation and Evaluation

Mamta[1], Asif Ekbal[1], Pushpak Bhattacharyya[1], Shikha Srivastava[2], Alka Kumar [2], Tista Saha[2]
[1] Department of Computer Science and Engineering
Indian Institute of Technology Patna, India
[2] Centre for Development of Telematics (C-DOT, India)
{mamta_1921cs11, asif, pb}@iitp.ac.in, {shikha, alkakm, tista}@cdot.in

Abstract

Due to the phenomenal growth of online content in recent time, sentiment analysis has attracted attention of the researchers and developers. A number of benchmark annotated corpora are available for domains like movie reviews, product reviews, hotel reviews, etc. The pervasiveness of social media has also lead to a huge amount of content posted by users who are misusing the power of social media to spread false beliefs and to negatively influence others. This type of content is coming from the domains like terrorism, cyber security, technology, social issues, etc. Mining of opinions from these domains is important to create a socially intelligent system to provide security to the public and to maintain the law and order situations. To the best of our knowledge, there is no publicly available tweet corpora for such pervasive domains. Hence, we firstly create a multi-domain tweet sentiment corpora and then establish a deep neural network based baseline framework to address the above mentioned issues. Annotated corpus has Cohen's Kappa measurement for annotation quality of 0.770, which shows that the data is of acceptable quality. We are able to achieve 84.65% accuracy for sentiment analysis by using an ensemble of Convolutional Neural Network (CNN), Long Short Term Memory (LSTM), and Gated Recurrent Unit (GRU).

Keywords: Sentiment, Multi-domain Corpus, Deep Learning

1. Introduction

With the tremendous increase in the number of users on social media, a huge amount of content is being generated every day. Twitter generates on average 6000 tweets per second, 500 million per day, and around 200 billion tweets a year [1]. Mining opinions expressed in these tweets can provide interesting insights for the construction of a socially intelligent system. Human creativity and perversity continuously generate a large volume of a variety of texts. But, it is also true that many users misuse power of social media to accentuate the sectarian conflict by spreading false beliefs. Many negative groups use social media platforms to strategize, to gain supporters via tweets, and to negatively influence others. Global politics has been the new victim of social media, impacting the election result. Domains which are contributing to generate such type of contents include terrorism, cyber security, technology, and many other social issues. Mining of opinions from the above-mentioned domains can help the government and security agencies to monitor the content generated everyday. It can help to monitor terrorist groups, domestic threats, and crime activities to provide security to public and to maintain the law and order. The very first step in building such type of intelligent system is the mining of user sentiments.

Deep learning has evolved as a popular technique over the years to solve many Natural Language Processing (NLP) problems including sentiment analysis. Annotated corpora is certainly the foremost requirement. Many annotated corpora are available for sentiment analysis. But most of the prior efforts have been in the domains such as movie, product, and hotel reviews (Pang et al., 2002; Pang and Lee, 2004; Blitzer et al., 2007). It can help to provide them satisfactory service or recommendation before buying a product etc. Apart from this, a small body of research has also been dedicated to sentiment analysis in the financial domain (Malo et al., 2013; Takala et al., 2014) and medical domain (Yadav et al., 2018).

To the best of our knowledge, there is no publicly available multi-domain tweet corpus, dedicated towards sentiment analysis for such pervasive domains. In this paper, we introduce a multi-domain tweet corpus for sentiment analysis, and then develop a deep neural network based baseline model to tag each tweet into three affect classes, namely positive, negative, and neutral. This is the very first attempt towards creating a benchmark setup for sentiment analysis in the above mentioned socially relevant domains. The corpus is manually annotated by three expert annotators. The inter-annotator agreement score comes out to be 0.770. We obtain the overall accuracy of 84.65%, and the precision, recall and F-measure values of 84.57%, 85.01% and 84.71%, respectively.

The remainder of the paper is organized as follows. Section 2 describes the related works. Section 3 describes the detailed processes involved in our corpus development along with the challenges. Section 4 describes the methodology that we adopted for our task. Section 5 presents the details of the experiments performed, evaluation results, and the necessary analysis. Section 6 concludes the paper and future plans for research.

2. Related work

Sentiment analysis is one of the most important research areas in the domain of Natural Language Processing (NLP).

[1] https://www.internetlivestats.com/twitter-statistics/

Several resources over the years have been created for sentiment analysis. But, most of these efforts have been put in the domain of movie reviews, product reviews, hotel reviews, etc. Below we present a survey on the various resources created for sentiment analysis.

(Pang et al., 2002) created a sentiment corpora from internet movie database which contains only those reviews where the author rating was expressed either with stars or some numerical value. Based on the rating, sentiment polarity was automatically decided from 2 categories, positive and negative. This contains 752 negative and 1301 positive reviews. (Pang and Lee, 2004) published another movie review polarity dataset for 2 classes, positive and negative. Dataset contains 1000 positive and 1000 negative movie reviews. (Blitzer et al., 2007) created a Multi-Domain Sentiment (MDS) Dataset consisting of four different types of product reviews taken from Amazon.com including Books, DVDs, Electronics, and Kitchen appliances. Dataset comprises of 1000 positive and 1000 negative reviews for each domain. (Shamma et al., 2009) constructed Obama-McCain Debate dataset by crawling the first U.S. presidential TV debate tweets in September 2008. They annotated the tweets for positive, negative, mixed, or other classes. The authors have shown an inter-annotator agreement of 0.655.

(Thelwall et al., 2012) created a dataset consisting of 4,424 tweets. Tweets were manually annotated with positive (1 to 5) and negative strength (-1 to -5). (Socher et al., 2013) introduced a Sentiment Treebank (STB) dataset constructed from the movie reviews domain. This dataset contains 215,154 phrases in the parse trees of 11,855 sentences annotated at the fine-grained level. This dataset was annotated with 5 classes, *viz.,* positive, negative, neutral, very positive and very negative. (Maas et al., 2011) introduced a larger IMDB dataset containing 50000 movie reviews for binary classification. Only highly polarized reviews were considered by them. For example, a negative review had score ≤ 4 out of 10 and a positive review had a score ≥ 7 out of 10. (Go et al., 2009) used distant supervision to create Stanford Twitter Sentiment (STS) corpora containing 160,000 tweets. The data was crawled by using positive and negative emoticons from Twitter using Twitter Search API. Tweets with positive emoticons were considered as positive and tweets with negative emoticons were considered as negative. It also contains a test set which contains 498 tweets, manually annotated for 3 classes, *viz.,* positive, negative, and neutral. Test set was crawled using names of products, companies, and people. SemEval 2017 dataset was constructed as a part of Task 4 (Rosenthal et al., 2017). Dataset was annotated on points of 2 (positive and negative), 3 (positive, negative, and neutral), and 5 (strongly positive, weakly positive, neutral, weakly negative, and strongly negative) scales. All the annotations were performed using CrowdFlower. SemEval 2017 dataset was built by merging all previous year's SemEval datasets, consisting of 50,333 tweets related to twitter trends Donald Trump, iPhone, etc.

Researchers have also put their efforts towards building sentiment corpora for the financial domain. As an example, (O'Hare et al., 2009) created financial blog corpus. They collected data of 500 specific companies and annotated it at document and paragraph-level at 3 point (positive, negative, and neutral) and 2 point (positive and negative) polarity scale of the sentiment. The whole corpus contains 1,691 annotated documents. (Malo et al., 2013) collected data from a number of financial news sources and then manually annotated 5000 sentences for positive, negative, and neutral class. This data was annotated by three annotators and the Kappa score was in the range of 0.611 to 0.886. They also trained a classifier to conduct sentence-level analysis of financial news sentiments. (Takala et al., 2014) annotated data of Thomson Reuters newswire related to 10 different topics having a significant financial impact. They annotated 297 documents and 9000 sentences of those documents for 7 classes (very positive, positive, slightly positive, neutral, slightly negative, negative, and very negative) and for three-point scale polarity (positive, negative, and neutra)l.

In recent times, researchers have also started exploring the importance of sentiment analysis in the medical domain. (Yadav et al., 2018) created a medical corpus to analyze the sentiment with respect to the user's medical condition. Table 1 shows the overview of datasets available with respect to their domain, classes, and size. Review shows that there is no existing corpus on the target domains that we consider.

3. Sentiment-Annotated Corpora Development

We collect data related to the following socially relevant domains: *terrorism, cyber security, technology, and social issues like terrorism, crime, alcoholism etc*. After the collection of raw data, we apply certain filters to obtain the relevant tweets, and then assign it to the human experts for the annotation. The resources can be obtained from here, [2]. In the subsequent subsections, we discuss these steps:

3.1. Data Collection

We collect the data from Twitter using the Streaming API [3] and Twitter Search API [4]. Streaming API collects real-time streaming data, whereas Search API crawls tweets published in the past 7 days. The crawler was designed to extract the data by searching with the following set of keywords *casteism, terrorism, counter-terrorism, cyber security, earthquake, cyber crime, cyclone, naxalism, communal dispute, human trafficking, narcotics, technology, weapons, crime, and elections.*. Data is crawled in several weeks between January 03, 2019 to April 01, 2019.

3.2. Data Pre-processing

We pre-process the raw data collected from Twitter, and convert it into the desirable form.

- Raw data contains many irrelevant tweets. To reduce annotation efforts for such irrelevant tweets, we designed a filter to extract only relevant tweets based on the following criteria:

[2]The corpus is publicly available at `https://www.iitp.ac.in/~ai-nlp-ml/resources.html#sentimentM`

[3]https://developer.twitter.com/en/docs/tweets/filter-realtime/overview

[4]https://developer.twitter.com/en/docs/tweets/search/overview

Data	Domain	Class	Instance
(Pang et al., 2002)	Movie Reviews	2	2053
(Pang and Lee, 2004)	Movie Reviews	2	2000
multi-domain Dataset (Blitzer et al., 2007)	Amazon product reviews for Books, DVDs, Electronics and Kitchen Appliances	2	2000 for each domain
Sentiment Treebank (Socher et al., 2013)	Movie Reviews	5	11855
IMDB (Maas et al., 2011)	Movie Reviews	2	50000
STS (Go et al., 2009)	Tweets based on Emoticons	2	160,000
SemEval 2017 (Rosenthal et al., 2017)	Twitter trends Donald Trump, iPhone etc.	3	50333
(O'Hare et al., 2009)	Financial Blog	2,3	1,691 documents
(Malo et al., 2013)	Financial News	3	5000
(Takala et al., 2014)	Financial	3,7	297 documents

Table 1: Short Review of Available Sentiment Datasets

  - If a tweet is duplicate.
  - If a tweet only contains URL.
  - If a tweet is written in non-English language or mixture of English and non-English.
  - If tweet length is less than 10 characters.
  - If a tweet only contains some user mentions.
- All URLs in tweet were replaced by word url.
- All mention @username and retweet symbols rt@username were removed from tweet.
- All emojis were replaced by their meaning using emoji library[5].

Irrelevant tweets detected by the designed filters were eliminated, and the corpus was prepared for manual annotation. Other steps mentioned above were performed after annotations to make the data suitable for performing experiments. Some examples of the tweets along with their irrelevant categories are shown in Table 2.

3.3. Data Annotation

After data extraction and pre-processing, we conduct manual annotation of the dataset. Three annotators with postgraduate level knowledge in English are employed for annotation. Annotators are asked to write the overall polarity of the tweet for 3 classes, *viz.,* neutral, negative and positive, if any opinion expression is found. For annotation, we follow the guidelines used in the SemEval task (Rosenthal et al., 2015; Mohammad, 2016). We provided some tweets to the annotators with gold labels to create understanding of the class labels. Annotators were also instructed to annotate the tweet without being biased towards any specific demographic area, religion, etc.

3.4. Challenges

During annotation, we faced the following challenges:

- In some of the tweets, writer provides information about a negative/positive situation without holding his/her own opinion. For example: *Far-right extremism poses the biggest security threat to northern England, a counter-terrorism expert has warned. https://t.co/UETyNzMbee*. In this tweet, no opinion is expressed, so this can be annotated with either neutral as no opinion is expressed, or negative because of negative situation or event. So, we decided to annotate such type of tweets based on the situation, the writer describes. The above tweet is annotated as negative for our case.
- If a tweet is of mixed nature having both positive as well as negative content, then overall polarity is decided based on the majority of positive or negative content.
- If a writer makes a request to do something positive in the context of a negative situation, then we assumed the sentiment to be positive. For example, *we should unite together to remove crime from our country.* In this example tweet, the writer is requesting everyone with positive attitude, so we considered the polarity of such types of tweets as positive.
- If a writer asks a question. For example: *@Partisangirl What the hell are they bombing? What is left of Syria to bomb at this point?*. In this tweet, writer expresses a kind of frustration by asking a question. We assume an overall polarity of this tweet as negative.

Some examples of annotations are described in Table 3.

3.5. Dataset Statistics

The corpus that we created contains 12,737 tweets across various domains containing a total of 4036 positive, 4299 negative, and 4402 neutral tweets. Some statistics of this dataset are reported in Figure 1.

3.6. Quality Test

We engaged three annotators to label each tweet instance with its associated sentiment. We measure inter-rater agreement in order to check the goodness of annotations given

[5]https://pypi.org/project/emoji

Tweet	Irrelevant Category
Kashmir is..	Less than 10 characters
@Joseph @mksahni	User mentions
https://www.google.com	Only URL
@JasonLeopold https://www.google.com	URL and user mentions
Terrorism bohot increase ho rha hai.	Code mixed (English and Hindi Language)

Table 2: Samples of Irrelevant Tweets

Tweet	Annotation
We are enjoying the chance to see some new prototype designs for adaptive technology made specifically for individuals in our community at the PRAC Meeting in the National Electronics Museum tonight!	positive
Pakistan government has given a green signal to opening Sharda Peeth Corridor, a long -standing demand of Kashmiri Pandits. Finally! After Kartarpur Corridor, it's an another positive signal by Pakistan. Hope! They will strictly work to end terrorism as well	positive
@realDonaldTrump has just made a decision that threatens the lives of every Israeli and puts the future of Israel in doubt. There is NO WAY that giving nuclear technology to Saudi Arabia is pro-Israel. https://t.co/jFOKpEx0KW	negative
@tarutorikka We burn more than 50 percent of our budget on weapons and give billionaires huge tax breaks. Fact is the politicians here could care less about the people	negative
Every Indian is proud of the fact that India is a nuclear weapons power. It makes us secure, strong. https://t.co/lpYupvcoEf	positive
4 Simple Tips to Protecting Your Business From Cyber Attacks https://t.co/IH3hbGEcEi security	neutral

Table 3: Samples from annotated corpora

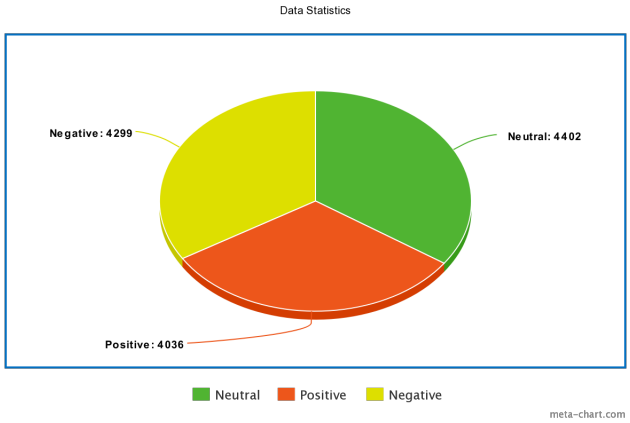

Figure 1: Data statistics of multi-domain annotated corpora

by different annotators. Cohen's Kappa coefficient (Cohen, 1960) is a statistical measure to analyze the inter-rater agreement. It is thought to be a more robust measure than simple percent agreement calculation and is defined as:

$$K = \frac{P(o) - P(e)}{1 - P(e)} \quad (1)$$

where P(o) P(e) are the observed and by chance agreement among annotators. Inter annotator agreement Kappa comes out to be 0.77 with confidence percentile of 95%. Kappa score shows that data is of acceptable quality. To merge three annotated versions of corpus, majority voting based technique was used.

4. Methodology for Sentiment Analysis

In this section, we describe the models we develop to perform sentiment analysis. We choose deep learning techniques to build sentiment analyzer because of the effectiveness of deep learning in solving a variety of Natural Language (NLP) tasks including sentiment analysis. We also compare our proposed deep learning based model with Support Vector Machine (SVM) based model that make use of a set of handcrafted features.

4.1. Models

We develop an ensemble model that utilizes the effectiveness of various deep learning models. The overall architecture of our model is shown in Figure 2. First, the tokenized sentence is passed through the embedding layer to obtain the embedding vector for each word. Then it is passed through individual deep learning-based models such as Convolution Neural Network (CNN), Long Short Term Memory (LSTM), and Gated Recurrent Unit (GRU). Finally, representation of all these models are concatenated and is passed through 2 layers of MLP to produce the final output class. We explain our ensemble architecture in the subsequent sections.

4.1.1. Word Embeddings

A bag-of-words (BoW) is created from all the unique words present in the tweet to create word representation. Then for each word w present in the tweet, a lookup matrix L is created to obtain its embedding $e(w) \in R^D$. Lookup matrix can be initialized using pre-trained word embedding vectors ((Mikolov et al., 2013; Pennington et al., 2014; Joulin et al., 2016)). For our work, we use the pre-trained word

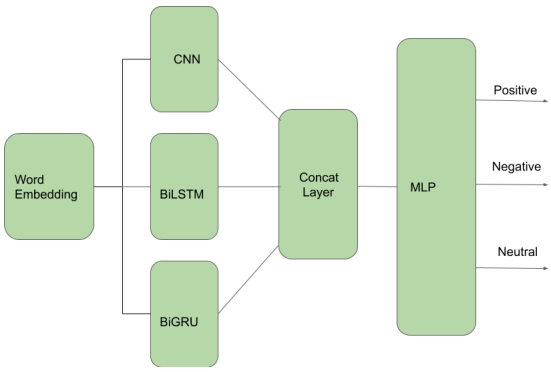

Figure 2: The proposed Ensemble architecture

embeddings from fastText ((Joulin et al., 2016)). The fastText uses subword information to generate embedding for a word, and hence it is able to handle the out-of-vocabulary (OOV) problem. The embedding of each word is then used as an input to the individual deep learning model to learn the representation of tweet.

4.1.2. Convolutional Neural Network (CNN)

CNN architecture has been widely used in variety of NLP task (Kumar and Singh, 2019; Kim, 2014) and it has been successfully applied to solve sentiment classification at various levels (Akhtar et al., 2016). Convolutional neural network (CNN) consists of convolutional layers. Convolutional process on sentence is used to conserve n-gram information of sentence. Convolutional layers are followed by non-linear layer, Relu, followed by the pooling layers. We use 2 convolutional layers. The first convolutioanl layer contains 128 filters of sizes 2, 3, and 4 each, and second convolutional layer contains 128 filters of size 3. It is then followed by max pooling layer, dense layer and output layer. Filters of sizes 2, 3, and 4 correspond that the filter can slide over 2, 3, or 4 words at a time.

4.1.3. Long Short Term Memory Network (LSTM)

We use LSTM network (Hochreiter and Schmidhuber, 1997) to learn sequential features from text using its gating mechanism. LSTMs are special type of recurrent neural network (RNN) which are capable of learning long-term dependencies by handling the vanishing or exploding gradient problem. LSTM has three gates, *viz.* input gate, forget gate, and output gate. Collectively, these three gates determine how much information should be lost and how much information should be added to memory.
We use two layers of Bidirectional LSTM on top of each other with 128 units in each LSTM layer.

4.1.4. Gated Recurrent unit (GRU)

Similar to LSTM unit, the GRU (Chung et al., 2014) has gating units that modulate the flow of information inside the unit without having a separate memory cell which makes it simpler than LSTM. In the LSTM unit, output gate controls the amount of cell content seen or used by other units in the network, whereas GRU's recurrent state is fully exposed without any control. GRU has lesser parameters to learn hence it takes less time to train than LSTMs. We use two layers of bidirectional GRU on top of each other with 128 units in each GRU layer.

4.1.5. Ensemble using Multilayer Perceptron

Ensemble combines several models to produce one optimal predictive model. In our work, we combine the performance of CNN, LSTM, and GRU using multilayer perceptron to improve the prediction power of system. We use 2 layers of MLP which receives input representation from CNN, LSTM, and GRU. Finally, the output of last MLP layer is given to the output layer.
We use ReLU activation function in hidden layers of MLP and Softmax in the output layer. We use Categorical Cross-entropy as the loss function, and Adam optimizer as an optimization function to optimize the weights. To avoid overfitting (Hawkins, 2004), we use dropout of 0.25 (Srivastava et al., 2014). Dropout means that some randomly selected neurons were deactivated during training. Output of this MLP network is a vector representing the class probability values. From this vector, we found the final class by choosing the class with the highest probability value.
All the three individual models are separately trained and optimized using Adam optimizer. Ensemble shows the increased performance level compared to the individual models.

5. Experimental Results and Analysis

We divide the dataset into 3 parts: train data, validation data, and test data. Training data contains 8659 samples, validation data consists of 1530 samples, and test data consists of 2548 samples. Detailed class-wise distribution of train, validation, and test are shown in the Table 4. We implement our model using python based Keras library with TensorFlow as back-end [6]. All the computations are performed on Nvidia GeForce GTX 1080 GPU with 12GB memory. We use 300 dimensional word vectors produced by pre-trained word embedding model fastText.

Type	Positive	Negative	Neutral	Total
Train	2755	2906	2998	8659
Validation	467	511	552	1530
Test	815	882	851	2548

Table 4: Statistics of the dataset used in the experiment

At first we implement the following four baseline models: Support Vector Machine (SVM) based model using lexicon features, Convolutional Neural Network (CNN), Long Short Term Memory (LSTM), and Gated Reccurrent Unit (GRU).
To train SVM, we use the following set of features:

- **N-grams**: We use Tf-Idf word n-grams (n = 1, 2, 3, 4) and character n-grams (n = 2, 3, 4).
- **Lexicon Features**: To extract lexicon features. we use BingLiu Lexicon [7], SentiWordNet (Baccianella et al.,

[6] https://keras.io/
[7] https://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html#lexicon

SNo.	Model	Precision	Recall	F1 measure	Accuracy
1	SVM	67.00	66.00	66.00	66.49
2	CNN	82.69	83.64	82.74	82.77
3	LSTM	83.17	84.11	83.63	84.00
4	GRU	83.59	84.36	84.00	84.07
5	**Ensemble**	**84.57**	**85.01**	**84.71**	**84.65**

Table 5: Evaluation results: Individual models and the ensemble based model

2010), and NRC Lexicons (Mohammad et al., 2013; Svetlana Kiritchenko and Mohammad, ; Zhu et al., 2014; Mohammad and Turney, 2013). BingLiu lexicon contains list of words with their polarity. SentiWordNet and NRC Lexicons contain sentiment intensity of words. NRC lexicon contains two separate files for unigrams and bigrams. We extract the following set of features (for unigrams as well as bigrams) from these lexicons:

  - Number of positive words in tweet (num_pos).
  - Number of negative words in tweet (num_neg).
  - Difference between num_pos and num_neg.
  - 1 if num_pos is greater than num_neg otherwise -1.
  - Log value of ratio of num_pos and num_neg.
  - Maximum of positive score among all words of a tweet (max_pos).
  - Sum of all positive scores in a tweet (sum_pos).
  - Mean of all positive scores in a tweet (mean_pos).
  - Find the maximum of negative score among all words of a tweet (max_neg).
  - Sum of all negative scores in a tweet (sum_neg).
  - Mean of all negative scores in a tweet (mean_neg).
  - Sum of max_pos and max_neg
  - Sum of sum_pos and sum_neg

- **Average Embeddings**: We scale the 300-dimensional embedding vector of words according to their tf-idf weights. Then we compute the average of these weighted embeddings for all the words in a tweet and used it as a feature vector.

All these features are concatenated together and passed to a SVM classifier for learning. Table 5 shows the performance of our proposed multi-domain corpora. The baseline classifier, SVM yields an accuracy of 66.49% with precision, recall, and F1 measure of 67%, 66%, and 66%, respectively. With CNN, we are able to achieve an accuracy of 82.77% with precision, recall, and F1 measure of 82.69%, 83.64%, and 82.74%, respectively. For LSTM, we achieve an accuracy of 84.00%, and precision, recall, and F1 measure of 83.17%, 84.11%, and 83.63%, respectively. The GRU model shows 84.07% accuracy with precision, recall, and F1 measure of 83.59%, 84.36%, and 84.00%, respectively.

Further, we construct an MLP based ensemble which yields the accuracy of **84.65%** with precision, recall, and F1 measure of **84.57%**, **85.01%**, and **84.71%**, respectively. This shows that our proposed ensemble achieves superior performance compared to the participating models.

5.1. Error Analysis

In this section, we present the detailed error analysis, both quantitatively and qualitatively.

class	negative	neutral	positive
negative	772	93	17
neutral	104	702	46
positive	65	66	683

Table 6: Confusion matrix for MLP based ensemble model

Table 7 shows two types of cases: i). The deep learning based models, CNN, LSTM, and GRU correctly classify the instances, which are wrongly predicted by SVM. ii). Although the performance of the deep learning models are quite similar quantitatively, qualitatively they are contrasting in nature. There are significantly a large number of instances, where one model predicts correctly, while other fail and the vice versa. Motivated by this, we combine the individual deep learning models through an ensemble of multilayer perceptron network.

Further, we analyze the performance of our ensemble model. We show the quantitative analysis through the confusion matrix, shown in Table 6. From the confusion matrix, it is clear that the instances from the negative and positive classes are confused with the neutral class. Instances from the neutral are confused with negative as well as positive. We perform a very closer analysis to the output of the individual models as well as the ensemble model. In Table 8, we show some of the cases (Example 1 and 2), where all the individual deep learning models perform misclassifications, but the ensemble model succeeds. We also show some example cases, where the proposed ensemble performs misclassification. Example 3 and 4 are the cases where the ensemble classifier misclassifies. It misclassifies the negative into neutral class and neutral into the negative class. The possible reason could be that for example 3, the classifier got confused due to the presence of both positive (effective) and negative (terrorism) words, and also failed to identify the negation term (no), which reverse the overall polarity of positive word. In example 4, speaker gave information by using negative words (e.g. killing, crime etc). Hence, the classifier is again confused with the presence of negative words, and so it misclassifies the neutral class into

Sr.	Tweet	SVM	CNN	LSTM	GRU	Actual	Comments
1	Sir @DelhiPolice @CPDelhi pls register a FIR against @quizzicalguy who is circulating such videos to disturb the peace and harmony of the country during election time	neutral	negative	negative	negative	negative	SVM is wrong
2	Women rising up in India. This is a good start! Sabarimala temple: Indian women form '620km human chain' for equality.	neutral	positive	positive	positive	positive	SVM is wrong
3	Operation has helped in eliminating the sleeper cells of terrorism throughout country.	negative	positive	negative	negative	positive	CNN is correct
4	@derasachasauda Volunteers remain 24*7 365 Ready To Serve Humanity...	neutral	neutral	positive	neutral	positive	LSTM is correct
5	Liberation of the last remaining territory held by Daesh in Syria is a huge achievement in the joint fight against terrorism. Many challenges still persist and Slovenia remains a committed member of The Global .	neutral	negative	negative	positive	positive	GRU is correct

Table 7: Examples showing contrasting nature of models

Sr.	Tweet	CNN	LSTM	GRU	Ensemble	Actual
1	@RogersHistory Sat in a doctors waiting room, I had just started writing my dissertation on terrorism	negative	negative	negative	neutral	neutral
2	The Somali Govts counter insurgency campaign in 1988 was in response to Ethiopian backed Somali rebels (SNM)	negative	negative	negative	neutral	neutral
3	@ANI No 'effective' action on security and terrorism.	neutral	neutral	negative	neutral	negative
4	Anybody with information about the killing is asked to call Suffolk Constabulary on 101 quo--ting crime reference	negative	negative	negative	negative	neutral

Table 8: Qualitative analysis

negative.

6. Conclusion

In this paper, we have created a benchmark corpus for multi-domain sentiment analysis. We have crawled tweets belonging to multiple domains from Twitter, applied several filters to clean the data, and annotated the corpus with three sentiment classes, namely positive, negative and neutral. The dataset comprises of tweets crawled from Twitter across multiple domains. Based on annotated corpora we built deep learning-based supervised classifiers for sentiment classification. Evaluation results show the overall accuracy, precision, recall, and F-measure values of 84.62, 84.57, 85.01, and 84.71 respectively for sentiment classification. In future, we would like to explore contextual embeddings and mechanism for negation handling to improve the system performance.

7. Acknowledgement

Authors would like to thank Centre for Development of Telematics, India (C-DOT) for funding this research. We would also like to extend special thanks to the linguists Saroj Jha (IIT Patna), Akash Bhagat (IIT Patna), and Swati Srivastava (IIT Patna) for their support in annotation of tweets and Harshada Sorte (C-DOT) for her contribution in data collection based on domain coherent keywords.

8. Bibliographical References

Akhtar, M. S., Kumar, A., Ekbal, A., and Bhattacharyya, P. (2016). A hybrid deep learning architecture for sentiment analysis. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 482–493.

Baccianella, S., Esuli, A., and Sebastiani, F. (2010). Sentiwordnet 3.0: an enhanced lexical resource for sentiment analysis and opinion mining. In *Lrec*, volume 10, pages 2200–2204.

Blitzer, J., Dredze, M., and Pereira, F. (2007). Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *Proceedings of the 45th annual meeting of the association of computational linguistics*, pages 440–447.

Chung, J., Gulcehre, C., Cho, K., and Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*.

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46.

Go, A., Bhayani, R., and Huang, L. (2009). Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford*, 1(12):2009.

Hawkins, D. M. (2004). The problem of overfitting. *Journal of chemical information and computer sciences*, 44(1):1–12.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.

Joulin, A., Grave, E., Bojanowski, P., and Mikolov, T. (2016). Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759*.

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

Kumar, A. and Singh, J. P. (2019). Location reference identification from tweets during emergencies: A deep learning approach. *International journal of disaster risk reduction*, 33:365–375.

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

Malo, P., Sinha, A., Takala, P., Ahlgren, O., and Lappalainen, I. (2013). Learning the roles of directional expressions and domain concepts in financial news analysis. In *2013 IEEE 13th International Conference on Data Mining Workshops*, pages 945–954. IEEE.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Mohammad, S. M. and Turney, P. D. (2013). Crowdsourcing a word-emotion association lexicon. 29(3):436–465.

Mohammad, S. M., Kiritchenko, S., and Zhu, X. (2013). Nrc-canada: Building the state-of-the-art in sentiment analysis of tweets. In *Proceedings of the seventh international workshop on Semantic Evaluation Exercises (SemEval-2013)*, Atlanta, Georgia, USA, June.

Mohammad, S. (2016). A practical guide to sentiment annotation: Challenges and solutions. In *Proceedings of the 7th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 174–179.

O'Hare, N., Davy, M., Bermingham, A., Ferguson, P., Sheridan, P., Gurrin, C., and Smeaton, A. F. (2009). Topic-dependent sentiment analysis of financial blogs. In *Proceedings of the 1st international CIKM workshop on Topic-sentiment analysis for mass opinion*, pages 9–16. ACM.

Pang, B. and Lee, L. (2004). A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42nd annual meeting on Association for Computational Linguistics*, page 271. Association for Computational Linguistics.

Pang, B., Lee, L., and Vaithyanathan, S. (2002). Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, pages 79–86. Association for Computational Linguistics.

Pennington, J., Socher, R., and Manning, C. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Rosenthal, S., Nakov, P., Kiritchenko, S., Mohammad, S., Ritter, A., and Stoyanov, V. (2015). Semeval-2015 task 10: Sentiment analysis in twitter. In *Proceedings of the 9th international workshop on semantic evaluation (SemEval 2015)*, pages 451–463.

Rosenthal, S., Farra, N., and Nakov, P. (2017). Semeval-2017 task 4: Sentiment analysis in twitter. In *Proceedings of the 11th international workshop on semantic evaluation (SemEval-2017)*, pages 502–518.

Shamma, D. A., Kennedy, L., and Churchill, E. F. (2009). Tweet the debates: understanding community annotation of uncollected sources. In *Proceedings of the first SIGMM workshop on Social media*, pages 3–10. ACM.

Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C. D., Ng, A., and Potts, C. (2013). Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1631–1642.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958.

Svetlana Kiritchenko, X. Z. and Mohammad, S. M.). Sentiment analysis of short informal texts. 50:723–762.

Takala, P., Malo, P., Sinha, A., and Ahlgren, O. (2014). Gold-standard for topic-specific sentiment analysis of economic texts. In *LREC*, volume 2014, pages 2152–2157.

Thelwall, M., Buckley, K., and Paltoglou, G. (2012). Sentiment strength detection for the social web. *Journal of the American Society for Information Science and Technology*, 63(1):163–173.

Yadav, S., Ekbal, A., Saha, S., and Bhattacharyya, P. (2018). Medical sentiment analysis using social media: towards building a patient assisted system. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

Zhu, X., Kiritchenko, S., and Mohammad, S. (2014). Nrc-canada-2014: Recent improvements in the sentiment

analysis of tweets. In *Proceedings of the 8th international workshop on semantic evaluation (SemEval 2014)*, pages 443–447.