Application Notes

Md Shad Akhtar
Department of Computer Science and Engineering, Indian Institute of Technology Patna, INDIA

Asif Ekbal
Department of Computer Science and Engineering, Indian Institute of Technology Patna, INDIA

Erik Cambria
School of Computer Science and Engineering, Nanyang Technological University, SINGAPORE

How Intense Are You? Predicting Intensities of Emotions and Sentiments Using Stacked Ensemble

Abstract

Emotions and sentiments are subjective in nature. They differ on a case-to-case basis. However, predicting only the emotion and sentiment does not always convey complete information. The degree or level of emotions and sentiments often plays a crucial role in understanding the exact feeling within a single class (e.g., '*good*' versus '*awesome*'). In this paper, we propose a stacked ensemble method for predicting the degree of intensity for emotion and sentiment by combining the outputs obtained from several deep learning and classical feature-based models using a multi-layer perceptron network. We develop three deep learning models based on convolutional neural network, long short-term memory and gated recurrent unit and one classical supervised model based on support vector regression. We evaluate our proposed technique for two problems, i.e., emotion analysis in the generic domain and sentiment analysis in the financial domain. The proposed model shows impressive results for both the problems. Comparisons show that our proposed model achieves improved performance over the existing state-of-the-art systems.

Digital Object Identifier 10.1109/MCI.2019.2954667
Date of current version: 10 January 2020

©ISTOCKPHOTO.COM/MA_RISH

I. Introduction

We live in a time where access to information has never been so free. Online platforms like Twitter, Facebook, etc. give a sense of power where a user can express his/her views, vent opinions and get to know about others' ideas and thought processes. All this is possible in mere 140 characters that Twitter limits per tweet (recently, Twitter raised the characters limit to 280; however, all the tweets in our datasets are of 140 characters or less). This short piece of text has the potential to shape peoples' outlook toward any situation or product. Companies and service providers can utilize dynamic textual information, and infer the public opinions about a newly launched product or any service or market conditions.

Corresponding Author: Erik Cambria (cambria@ntu.edu.sg).

1556-603X/20©2020IEEE

Emotion analysis [1] in natural language processing (NLP) targets to automatically extract the emotional state of a user through his/her writing (tweets, post, blogs, etc.). Ekman [2] studied the human emotion behavior in details and categorized them into six basic human emotions. According to him, basic emotions are *anger*, *fear*, *surprise*, *sadness*, *joy* and *disgust*. Comparatively, the aim of sentiment analysis is to predict the polarity orientation (e.g., *positive*, *negative*, *neutral* or *conflict*) in the user-written texts [3]. Coarse-grained sentiment analysis (document or sentence level) usually ignores critical information toward a target. In fine-grained sentiment analysis [4], we can emphasize on a target without losing any critical information.

Sentiment and emotions are closely related. Emotions are usually shorter in duration, whereas sentiments are more stable and valid for a longer period of time [5]. Sentiments are also normally expressed toward a target entity, whereas emotions are not always target-centric [6]. Table I depicts example scenarios for both the problems. In the first example, emotion *'joy'* is derived from the phrase '*died from laughter*' which is also very intense. However, the emotion associated with the second example which contains similar phrase '*died from cancer*' is *'sadness'*. In such scenario predicting the correct emotion is often very challenging and non-trivial. The third and fourth examples reflect emotion classes *'fear'* and *'anger'* derived from the respective phrases '*Still salty*' and '*revenge*'. In sentiment analysis problem, the first example expresses *'positive'* sentiment whereas the second example has *'negative'* sentiment for their respective targets, i.e., *WTS* and *Lloyds*.

In general, emotion analysis and sentiment analysis classify a text into one of the predefined classes (e.g., *joy*, *fear*, etc. for emotion and *positive*, *negative*, etc. for sentiment). However, predicted opinion or sentiment class of a text does not carry the finer information such as the exact state of mood or opinion of a user. Level or intensity of the expressed emotions or sentiments often differs on a case-to-case basis within a single class. For example, some emotions are comparatively gentler than the others (e.g., '*not good*' versus '*terrible*'). Emotion expressed by both the phrases is *anger*, however, the phrase '*not good*' expresses relatively mild emotion, whereas the phrase '*terrible*' is much severe.

Similarly, both phrases '*its fine*' and '*its awesome*' carry positive sentiment but express different level of sentiments. Sentiment of the latter case is strong, whereas the sentiment of the earlier case is comparatively weak. Thus, measuring the degree of emotion is of paramount importance in analyzing the finer-level details of the expressed emotions and sentiments. Such analysis has wide real-world applications such as big social data analysis for business intelligence [9], stock market prediction [10], healthcare [11], recommendation systems [12], etc.

In this paper, we propose a multi-layer perceptron (MLP) based ensemble technique for solving two different problems, i.e., emotion analysis and fine-grained sentiment analysis. We aim to identify the intensities of emotions and sentiments, respectively for the two tasks. For emotion analysis, we employ generic tweets, whereas for sentiment analysis our target domain is financial text. At first, we develop a support vector regression (SVR) [13] based feature-driven system and three deep learning systems, namely a convolutional neural network (CNN) [14], a long short-term memory (LSTM) network [15] and a gated recurrent unit (GRU) network [16] for the intensity prediction. In the second step, we combine the outputs of these systems via the MLP network. The final output obtained from this combined model is better as compared to the individual models. We further perform a series of normalization heuristics to minimize the noise. The normalized text has a higher degree of readability than un-normalized text, thus making it a better candidate to find more representative word embeddings.

The current work follows one of our previous works [17] on intensity prediction. However, our current research significantly differs from earlier work *w.r.t.* the following points: a) Our previous work [17] addressed only financial sentiment analysis task, whereas in the current work we also focus on emotion intensity prediction for the four emotions, i.e., '*anger*', '*fear*', '*joy*' and '*sadness*'. Please note that intensity prediction of emotion is completely a different task; b) We include several features for training and testing of the classifier; c) We incorporate various normalization heuristics to address the noisy text; d) We present a detailed analysis of the obtained results *w.r.t.* various state-of-the-art and traditional techniques;

TABLE I Examples of Emotion and Sentiment analysis. Intensity values reflect the degree of emotion/sentiment in the respective text. Examples are taken from the respective datasets [7], [8].

EMOTION ANALYSIS (0: *NO EMOTION*, 1: *HIGH EMOTION*)

TEXT	DOMAIN	EMOTION	INTENSITY
JUST DIED FROM LAUGHTER AFTER SEEING THAT.	TWITTER	JOY	0.92
MY UNCLE DIED FROM CANCER TODAY…		SADNESS	0.87
STILL SALTY ABOUT THAT FIRE ALARM AT 2AM THIS MORNING.		FEAR	0.50
HAPPINESS IS THE BEST REVENGE		ANGER	0.25

SENTIMENT ANALYSIS (−1: *EXTREMELY NEGATIVE*, +1: *EXTREMELY POSITIVE*)

TEXT	DOMAIN	TARGET	SENTIMENT	INTENSITY
BEST STOCK: $WTS +15%	MICROBLOGS	WTS	POSITIVE	0.857
UK GOVERNMENT CUTS STAKE IN LLOYDS TO BELOW 11 PCT	NEWS HEADLINE	LLOYDS	NEGATIVE	−0.596

and e) We also presented a detailed qualitative analysis on the errors encountered by our proposed method. In another work, Ghosal et al. [18] developed a deep ensemble model that utilizes the character, word and lexicon level fusion for the sentiment and emotion prediction.

The main contributions of our proposed work are highlighted below: a) We effectively combine deep learning and feature driven traditional model via an ensemble framework; b) We develop a stacked denoising autoencoder based technique for an enhanced word representation by leveraging the syntactic and semantic richness of the two distributed word representations; c) We perform normalization of tweets by utilizing various heuristics; and d) We build a state-of-the-art model that effectively solves both the problems of emotion analysis and sentiment analysis.

The remainder of the paper is organized as follows: Section II briefly discusses existing techniques; Section III defines the overall problem; Section IV describes our proposed method in detail; experimental results along with detailed analysis on the results are presented in Section V; finally, Section VI concludes this paper.

II. Related Work

A survey of the literature [19]–[21] suggests mainly three groups of approaches for detecting the emotion from text, i.e., keyword-based methods, learning-based methods and hybrid methods. A linguistic resource WordNet-Affect was developed in [22] for the lexical representation of affective words. Applications of support vector machine (SVM) and conditional random field (CRF) for emotion detection are proposed in [23] and [24], respectively. Dung et al. [20] exploited human mental states *w.r.t.* an emotion for training a hidden Markov model (HMM). In contrast, Wu et al. [19] proposed a rule-based approach to extract emotion-specific semantics, which is then utilized for learning through various separable mixture models.

Recently, there is a growing trend to perform sentiment analysis involving financial texts [25]. The model proposed in [26] makes use of lexical cohesion to create a commutable metric for identifying the sentiment polarity of the financial news. O'Hare et al. [27] used the word-based approach on financial blogs to train a sentiment classifier for automatically determining the sentiment toward companies and their stocks. The analysis of financial news is an important component in predicting the stock market behavior as shown by [28]. The authors use the bag-of-words (BoW) and named entities (NEs) with SVM for predicting the stock prices. This goes to show that the stock market behavior is based on the opinions. Among the other notable works, a topic-centric Twitter sentiment analysis for stock prediction is proposed by Si et al. [29]. They employed Dirichlet process model for learning the topic and then utilized lexicons for predicting the sentiment score toward the topic. A fine-grained sentiment annotation scheme was incorporated by [30] for predicting the explicit and implicit sentiment in the financial text. An application of multiple regression model was developed by [31]. In another work, a multitask representational learning approach has been proposed in [32]. The authors evaluated four combination of tasks involving sentiment and emotion prediction, and showed that the multitask learning framework attained improved performance over the single-task learning framework.

III. Problem Definition

In this article, we focus on the problems of emotion analysis and sentiment analysis for different domains. For both the problems at hand, we aim to find the intensity score of a given emotion or sentiment. By nature, both the problems are of regression types, where we have to predict a continuous value representing the intensity of emotion or sentiment.

For the first problem, an instance of a tweet and an associated emotion are given. We aim to predict the intensity of emotion felt by the user - a score on a continuous scale of 0 to 1 is to be determined. Intensity values close to 1 reflect high-degree of emotions, whereas intensity values close to 0 reflect low-degree of emotions of the users at the time of writing the text. In this article, we target four different emotions, i.e., *'anger'*, *'fear'*, *'joy'* and *'sadness'*. Table I depicts one example scenario along with their intensity values for each emotion. For the second problem, financial short texts for two different domains, namely *'micro-blog messages'* and *'news headlines'*, having one or multiple company stock symbols (cashtags) are given. The objective is to predict the sentiment score for each of the company or stock mentioned in the range of −1 (bearish) to 1 (bullish), with 0 implying neutral sentiment.

IV. Proposed Methodology

We propose an MLP based ensemble approach to leverage the goodness of various supervised systems. We develop one feature-driven supervised model and three deep neural network architecture based models, *viz.* LSTM, CNN and GRU. The classical feature-based system utilizes a diverse set of features (c.f. Section IV-A) to train an SVR. The three deep architectures are trained on top of distributed word representations. In this article, we employ GloVe [33] and Word2Vec [34] models to learn our word embeddings. Although the underlying techniques of these two distributed models are different (GloVe is a count-based model that works on the principle of word co-occurrence matrix, whereas Word2Vec is a contextual model that aims to predict a word based on its context or vice-versa), literature suggests that both the techniques are efficient at capturing the syntactic and semantic properties of a word in the embedding space. However, some applications perform better on GloVe while other applications adapt well to Word2Vec. We, therefore, aim to leverage the goodness of these two models through a stacked denoising auto-encoder network. Finally, we ensemble the outputs of all four individual models through a three-layered MLP network. The output of the MLP network serves as the final intensity value for the respective problems. We furnish the details of our system in subsequent subsections. Figure 1 summarizes our proposed system.

A. Feature-based Model

In addition to the three deep learning based frameworks, the fourth model that we employ is a classic feature-driven model. We define and employ a diverse set of features for training and evaluation of an SVR. The SVR model predicts the intensity values on the continuous scale of [0, 1] and [−1, +1] for emotion analysis and sentiment analysis, respectively. The following set of features was used for this SVR model:

1) **Word and Character Tf-Idf:** The Tf-Idf measures the importance of word *w.r.t.* to a document in a corpus. We use Tf-Idf values of continuous sequences of 2, 3, 4, 5 words and characters at a time as the features.
2) **Tf-Idf Weighted Word Vector:** Every word in input is not equally significant for some specific problems. We, therefore, scale the word embedding of each word (E_w) corresponding to their Tf-Idf weights. The resultant vector (E'_w) is used as a feature for the experiments.
3) **Lexicon Features:** Lexicons are the list of words along with their polar information. Following are the list of lexicon features that we employ for each tweet:
   - **MPQA [35] and Bing Liu [36]:** For each sentence, we extract two features, i.e., positive word count and negative word count per lexicon and utilize them as feature values for the classifier.
   - **NRC Hashtag Sentiment and NRC Sentiment140 [37]:** We extract positive, negative and aggregate scores of each word in a sentence and use as feature values.
   - **SentiWordNet** [38]**:** We compute the sum of the positive, negative and aggregate scores of each word in a sentence and use them as feature values for building the model.

In addition to these lexicon features, we also extract the following features for emotion intensity prediction.

- **NRC Word-Emotion Association [39]:** We count the number of words matching respective emotion in the lexicon and use it as the feature value. We also use NRC Word-Emotion Association Expanded [40] lexicon for the feature extraction.
- **NRC Hashtag Emotion [41]:** We extract the sum of association scores of the words in a tweet for the emotions and use as the feature values in the model.
- **AFINN Sentiment/Emotion score** [42]**:** We use the aggregate of positive and negative word scores as the feature values. We, also, compute the aggregate scores for each emoticon present in a tweet and use it as the feature value for training.

4) **VADER Sentiment:** We employ VADER (Valence Aware Dictionary for sEntiment Reasoning) sentiment [43] score, which makes use of various grammatical and syntactical heuristics. For each sentence, VADER returns a compound sentiment score on a continuous scale of −1 (extremely negative) to +1 (extremely positive). For example, the compound sentiment score of the sentence ('*The book was good.*') is 0.4404, whereas for another sentence ('*The book was kind of good.*') score is 0.3832. It also reports three scores corresponding to ratios of positive, neutral and negative tokens in a sentence. We use the compound score and the three ratios as features in our feature-based model.

B. Multi-Layer Perceptron Based Ensemble

Ensemble is an efficient technique that tries to improve the overall performance of the system by combining the outputs of various candidate systems. The basic idea is to leverage the correctness of several systems for improving the overall performance.

Literature [44]–[46] suggests that the traditional approaches to ensemble are Boosting [47], Bagging [48], Voting (Weighted, Majority) [49]. However, our proposed approach differs from these existing works on the basis of the class of problem that we solve. Most of the systems solves a classification problem, while, in current work, we aim to predict the regression problem. Another notable difference is the underlying problem domain that these systems solves.

Our proposed ensemble technique is based on MLP. The MLP network is stacked on top of the candidate systems' predictions, i.e., predictions of CNN, LSTM, GRU & Feature-driven SVR systems. We use a three-layered ensemble network in our proposed system. The MLP network has $4 \rightarrow 4 \rightarrow 1$ neurons corresponding to the three layers of the network. As activation function, we employ *'Relu'* at the two hidden layers, while for the prediction, we use *'sigmoid'* and *'tanh'* activations at the output layer for emotion analysis and sentiment analysis, respectively. We choose *'Adam'* as our optimizer and introduced 20% Dropout at the intermediate layers.

All four candidate models are separately trained and tuned for both the problems. Although the performances of the individual systems are quite encouraging, a qualitative analysis suggests that the predictions of the individual systems are often complementary in nature, i.e., there are instances where one model fails, but another model succeeds in correctly predicting the intensity. This heterogeneous characteristic leads us to build an ensemble model that effectively combines the outputs of all the component models and further improves the performance.

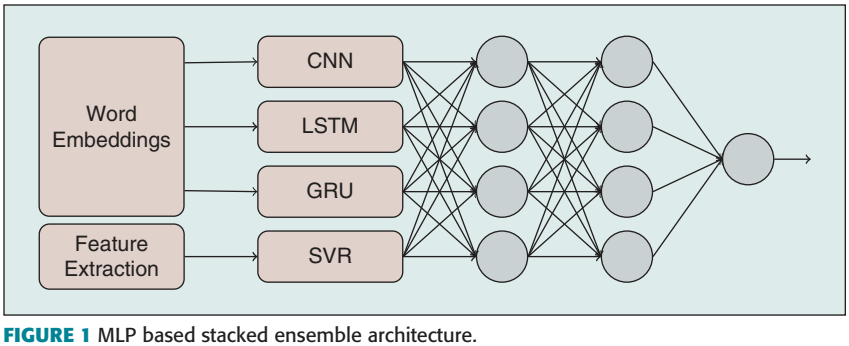

FIGURE 1 MLP based stacked ensemble architecture.

C. Word Embeddings

Any neural network architecture requires a vector representation of a word or sentence to work. Distributed representation models such as GloVe and Word2Vec have been proved to be effective for a wide range of NLP applications. The effectiveness of any neural network architecture depends on the quality of word embeddings which inherently depends upon two important entities: i) amount of training corpus and ii) in-domain corpus. The pre-trained word embedding models of GloVe (PWE-GLV) and Word2Vec (PWE-W2V) are trained on top of general purpose *Common Crawl* and Google News corpus. In general, they capture the syntactic and semantic properties of a word pretty well. However, to capture the domain-specific properties of a word, it is always recommended to use in-domain word embeddings. For example, the word *'hot'* in the sentences *'They serve hot foods.'* and *'The charger gets hot pretty quickly.'* conveys the opposite semantic, respectively, for the restaurant and laptop domains. Since one of the problems that we address here belongs to the financial domain, we train and use separate word embeddings utilizing the financial text corpus (FWE). We started with crawling Google & Yahoo News and collected 126 K financial news articles consist of approx. 92 million tokens. Subsequently, we train a GloVe (FWE-GLV) and a Word2Vec (FWE-W2V) model for the financial text. In comparison with the GloVe and Word2Vec pretrained word embedding corpus, our financial corpus size is relatively small; however, FWE performs reasonably well for the problem at hand (c.f. Table IV). Since the financial word embedding is specific to financial sentiment analysis task, we do not employ it for the generic emotion analysis task.

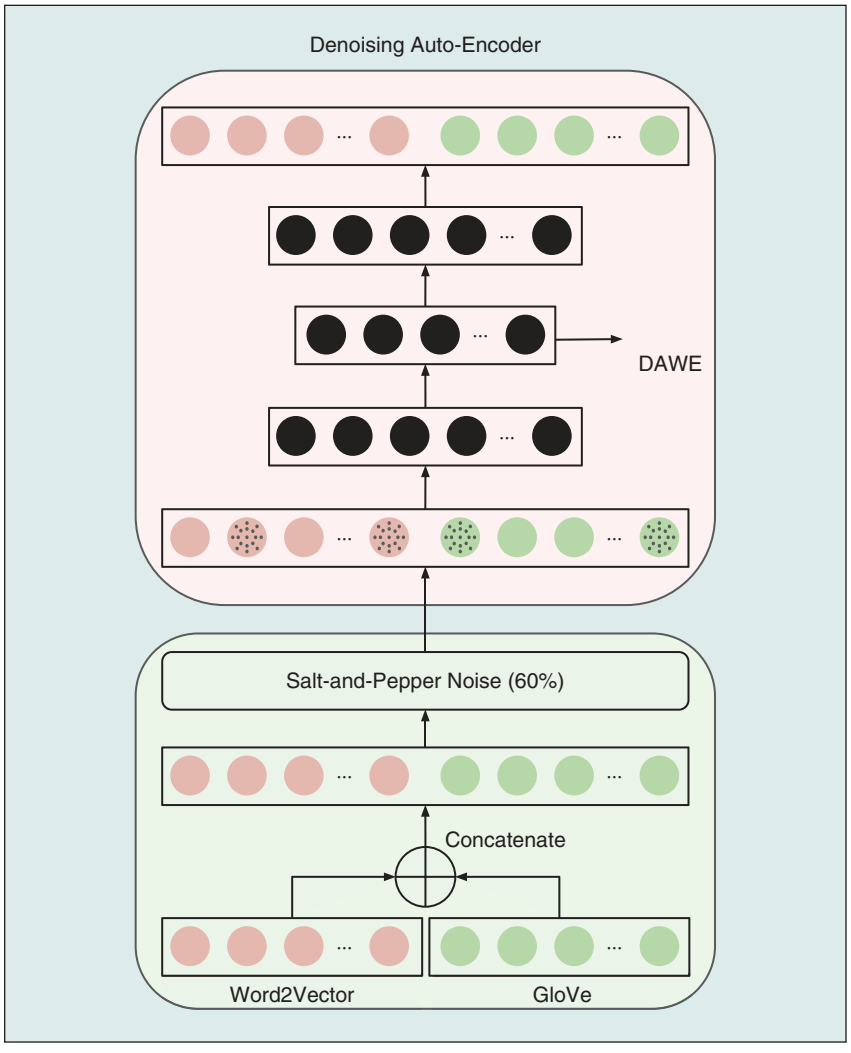

FIGURE 2 Scheme of the denoising auto-encoder based word embeddings (DAWE).

The performance of GloVe and Word2Vec embeddings are often competitive in nature. For some tasks, GloVe performs better whereas for other tasks Word2Vec has the advantage. To break the tie, we adopt a hybrid word embedding model that takes pretrained GloVe and Word2Vec word representations as input and produces a new representation that combines the best of both the pre-trained models. The hybrid model follows the work of [50] that comprises of stacked denoising auto-encoders. A denoising autoencoder is a neural network which is trained to reconstruct a clean repaired input from a corrupted version of the input. We concatenate the word embeddings of GloVe & Word2Vec into a single vector of dimension 600 (GloVe:300 and Word2Vec:300). Subsequently, we add *salt-and-pepper* noise to make the input corrupted. We experimented with the varying amount of noises ranging from 20 to 70% and observed that 60% noise is the optimal amount for our case. The denoising auto-encoder takes the concatenated noisy representation as input and tries to predict the original concatenated representation. The auto-encoder network comprises of three hidden layers having 400, 300 & 400 neurons, respectively. We take activation values of the middle hidden layer (i.e., 300-dimensional layer) as our new denoising auto-encoder word embeddings (DAWE). We employ *Adam* [51] optimizer with *mean-squared-error* loss and train the model for 90 epochs. The batch size is set to 16. Figure 2 summarizes the process for computing denoising auto-encoder based word embeddings.

In total we employ five different word embedding models for financial sentiment analysis (i–v) and three models for generic emotion analysis (i, ii and v): i) PWE-W2V; ii) PWE-GLV; iii) FWE-W2V; iv) FWE-GLV; and v) DAWE. We keep word embedding dimension of all these models as 300. Also, we train our proposed DL models in dynamic mode,

which allows word embeddings to be fine-tuned during the training.

V. Experiments, Results and Analysis

A. Dataset

We evaluate our model on the datasets of eighth Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis shared task on emotion intensity (EmoInt-2017) [7] for emotion analysis. The datasets of SemEval-2017 shared task on 'Fine-Grained Sentiment Analysis on Financial Microblogs and News' [8], are used for sentiment analysis. The EmoInt-2017 datasets [7] contain generic tweets representing four emotions, i.e., *joy*, *fear*, *anger* and *sadness*.

Datasets of SemEval-2017 [8] comprise of financial texts from microblogs (Twitter and StockTwits) and news (Yahoo finance). For the experimental purpose, we perform five-fold cross validation for model tuning and hyperparameter selection. According to the respective description papers, SemEval-2017 dataset [8] was manually annotated by three human financial experts while Emotion Intensity [7] dataset was created using the Best-Worst Scaling technique [52]. Detailed statistics of both the datasets are presented in Table II.

TABLE II Dataset statistics.

DATASETS	DOMAIN	TRAIN	DEV	TEST
EMOTION ANALYSIS (WASSA-2017) [7]	ANGER	857	84	760
	FEAR	1,147	110	995
	JOY	823	79	714
	SADNESS	786	74	673
SENTIMENT ANALYSIS (SEMEVAL-2017) [8]	MICROBLOGS	1,700	–	800
	NEWS	1,142	–	491

B. Preprocessing

We use NLTK [53] for tokenization. Since the contents were derived from the Internet, pre-processing is of paramount importance due to lack of proper grammar and structures. Since URLs, user names and numbers usually do not carry any polar sentiments, we replace these with the tags: <*url*>, <*user*> and <*number*>, respectively. For example, we replace '*www.twitter.com*' by <*url*> and '*@JonSnow*' by <*user*>. After stripping off excess white spaces, all the characters were converted to lowercase. Additionally, all HTML entities were converted to their corresponding unicode characters such as *'&'* was replaced by *'and'*. Hashtags carry meaningful information and are relevant to extract underlying emotions and sentiments. We first strip-off # symbol from the hashtags and then split the resulting token into constituent words. For example, *'#GreatDayEver'* is converted to *'Great Day Ever'*. We employ python-based *WordSegment*[1] module for the segmentation of hashtags. Finally, we perform normalization of noisy text by employing the following set of heuristics in line with [54].

- **Elongation of a valid word:** To convey their state of emotions or sentiments, users tend to express through elongation of a valid word, e.g., '*joooyy*', '*gooood*', etc. We define a heuristics that identifies all such elongated words and process them into valid dictionary words by iteratively dropping the consecutive sequence of characters. For example '*joooyy*' and '*gooood*' are converted to '*joy*' and '*good*', respectively.
- **Frequent noisy term:** Due to the character limit in Twitter, usage of abbreviations and slang terms in tweets are in common practice among users, e.g., *'grt'*, *'g8'* for *'great'*. To handle such cases we created a dictionary of commonly used abbreviations and slang terms along with their expanded valid forms. We then perform a lookup in the dictionary for each token in a tweet and on the success we use its expanded valid form for further processing. We compile the dictionary of the frequent noisy term by consulting the datasets of WNUT-2015 shared task on Twitter Lexical Normalization [55].
- **Verb present participle:** By careful inspection of tweets, it is observed that users have a common practice to skip the characters '*g*' or '*i*' from the present participle form of a verb, i.e., '*ing*' form of the verb. For example, '*enjoying*' is written as '*enjoyin*' or '*enjoyng*'. We correct these cases by applying a heuristics that considers all the verbs that end with either '*in*' or '*ng*' and convert them into valid present participle form of the verb.
- **Expand contraction:** Twitter users generally tend to merge two words by introducing an apostrophe (') symbol in place of few in between character sequences, e.g., the contraction *"i've"* belongs to valid words *"i have"*. Such practice saves a few crucial characters in a tweet and can be utilized for extra words. We create a list of such contractions and their expanded forms by consulting the WNUT-2015 datasets [55]. We apply a heuristic that identifies contracted tokens in a tweet and converts them to its normalized form.

[1]https://github.com/grantjenks/wordsegment

C. Experiments

For evaluation of the proposed models, we employ the Pearson correlation coefficient and cosine similarity score for the problems of emotion intensity and sentiment score, respectively. The choice of evaluation metrics was derived from the guidelines of the shared tasks on EmoInt-2017 [7] and SemEval-2017 [8]. Pearson correlation coefficient measures the linear correlation between the actual and predicted scores, whereas the cosine similarity score measures the degree of agreement between the actual and predicted values.

We separately train and tune all the deep learning systems (CNN, LSTM and GRU) over different word embeddings—pretrained, financial and the DAWE. As mentioned earlier, we do not employ financial word embedding for the generic emotion analysis task. The network architecture (dimensions, layers,

TABLE III Choice of Hyper-parameters.

MODELS	PARAMETERS	VALUES
CNN	CONVOLUTION	1×1D CONV
	FILTERS	300 (2, 3 & 4-GRAM)
	POOLING	1× MAX-POOLING (STRIDE: 2)
	FULLY-CONNECTED	3 LAYERS (50→10→1)
LSTM	LAYERS	2×100
	FULLY-CONNECTED	3 LAYERS (50→10→1)
GRU	LAYERS	2×100
	FULLY-CONNECTED	3 LAYERS (50→10→1)
SVR	C	2
	γ	0.03
MLP	**FULLY-CONNECTED**	**3 LAYERS (4→4→1)**
COMMON PARAMETERS	OUTPUT	EMOTION → *SIGMOID*, SENTIMENT → *TANH*
	OPTIMIZER	ADAM
	DROPOUT	20%
	ACTIVATIONS	RELU
	WE DIMENSION	300

etc.) of all these models have been fixed through cross-validation. In classical feature-based models, we use SVR to predict a regression value on a continuous scale. We perform a grid search for hyper-parameters tuning for both SVR and neural network based models. We also ensure common architecture and hyper-parameters reasonably suited to all the models. A summary of the choice of parameters used in the experiments is mentioned in Table III.

Table IV shows Pearson coefficient and cosine similarity scores of our various models. For emotion analysis in '*anger*' class, the CNN, LSTM, GRU and SVR report best Pearson scores of 0.664, 0.664, 0.652 & 0.701, respectively. In '*joy*' we observe 0.647, 0.625, 0.631 & 0.697 Pearson scores for the four models. For '*sadness*' and '*fear*', we obtain Pearson

TABLE IV Cosine similarity (Financial Sentiment) and Pearson correlation (Emotion Analysis) scores of various models on the test data. PWE: pretrained word embeddings; FWE: financial word embeddings; DAWE: denoising autoencoder word embeddings; W2V: Word2Vec embeddings; GLV: GloVe embeddings; PWE-W2V CNN: CNN model trained on pretrained Word2Vec.

		FINANCIAL SENTIMENT		EMOTION ANALYSIS				
	MODELS	MICROBLOGS	NEWS	ANGER	JOY	SADNESS	FEAR	AVERAGE
CNN								
CNN1	PWE-W2V CNN	0.705	**0.722**	**0.664**	0.626	0.701	0.697	0.672
CNN2	PWE-GLV CNN	0.721	0.697	0.662	**0.647**	**0.709**	**0.706**	0.681
CNN3	FWE-W2V CNN	0.710	0.705	–	–	–	–	–
CNN4	FWE-GLV CNN	**0.724**	0.714	–	–	–	–	–
CNN5	DAWE CNN	0.697	0.698	0.598	0.532	0.639	0.583	0.588
LSTM								
LSTM1	PWE-W2V LSTM	0.700	0.704	0.659	0.620	0.668	**0.704**	0.662
LSTM2	PWE-GLV LSTM	0.715	0.683	**0.664**	**0.625**	**0.679**	0.702	0.667
LSTM3	FWE-W2V LSTM	**0.727**	0.680	–	–	–	–	–
LSTM4	FWE-GLV LSTM	0.717	0.691	–	–	–	–	–
LSTM5	DAWE LSTM	0.722	**0.720**	0.579	0.527	0.579	0.597	0.570
GRU								
GRU1	PWE-W2V GRU	0.689	**0.721**	0.635	0.582	0.658	0.691	0.641
GRU2	PWE-GLV GRU	0.713	0.705	**0.652**	**0.631**	**0.674**	**0.701**	0.664
GRU3	FWE-W2V GRU	0.715	0.687	–	–	–	–	–
GRU4	FWE-GLV GRU	0.713	0.703	–	–	–	–	–
GRU5	DAWE GRU	**0.721**	0.712	0.567	0.481	0.567	0.583	0.550
FEATURE - SVR								
SVR1	TF-IDF + LEXICON + VADER	0.752	0.749	0.686	0.661	0.705	0.707	0.690
SVR2	SVR1 + PWE-W2V	0.740	0.731	**0.701**	0.678	**0.707**	0.742	0.707
SVR3	SVR1 + PWE-GLV	0.758	0.745	0.691	**0.697**	0.706	**0.748**	0.710
SVR4	SVR1 + FWE-W2V	0.709	0.702	–	–	–	–	–
SVR5	SVR1 + FWE-GLV	0.732	0.725	–	–	–	–	–
SVR6	SVR1 + DAWE	**0.765**	**0.760**	0.695	0.661	0.702	0.721	0.696

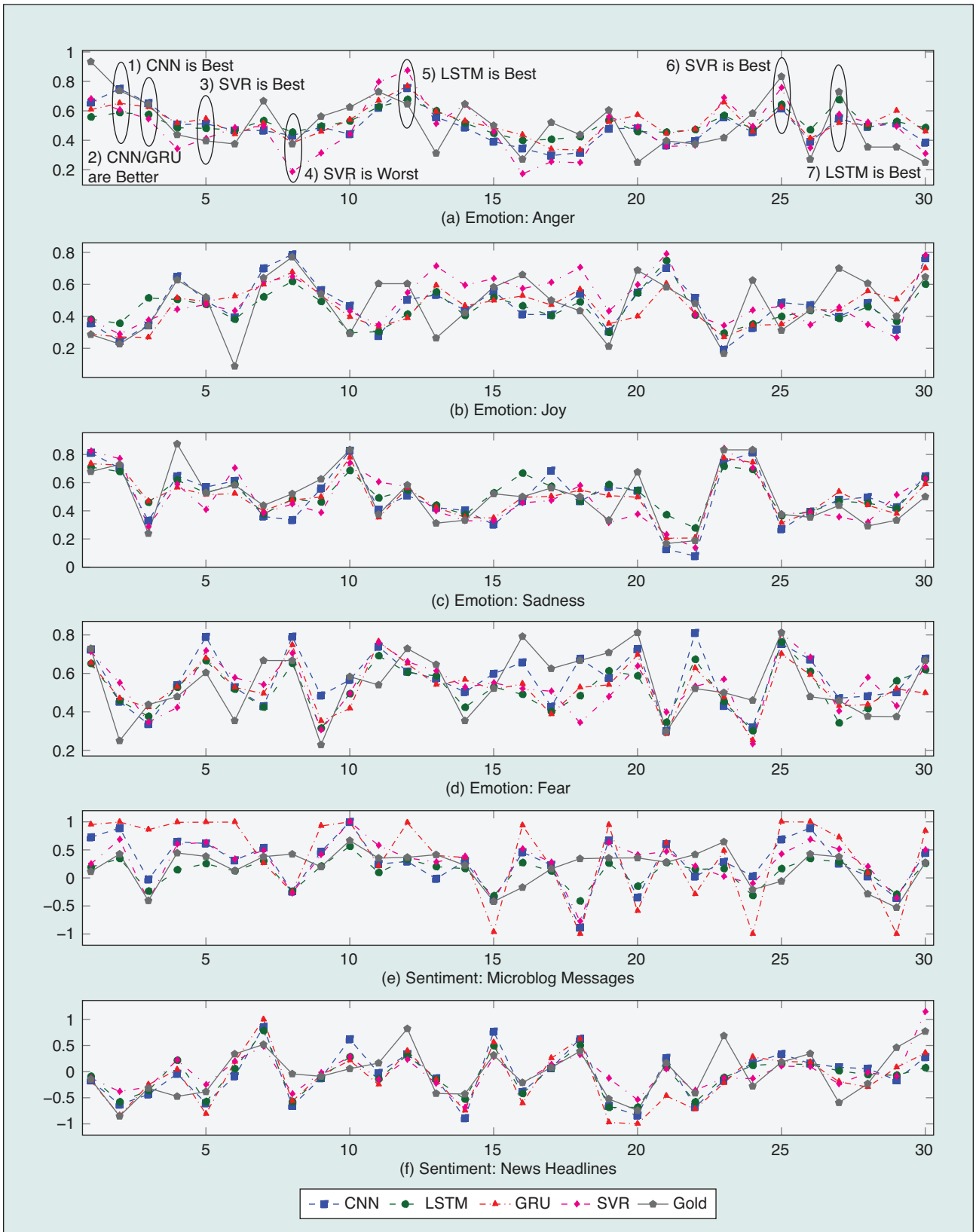

FIGURE 3 Contrasting nature of different models *w.r.t.* the gold standard values which motivate us to build an ensemble system; Y-axis: Intensities; X-axis: Samples; Sample size –30. (a) Highlighted region 1: CNN is best; 2: CNN & GRU are better; 3 and 6: SVR is best; 4: SVR is worst; 5 and 7: LSTM is best.

scores of 0.709, 0.679, 0.674 & 0.707 and 0.706, 0.704, 0.701 & 0.748, respectively. In *microblog* dataset four individual models, i.e., CNN, LSTM, GRU and feature-based systems obtain cosine similarity of 0.724, 0.727, 0.721 and 0.765, respectively. Similarly, in *headline* dataset the four models report 0.722, 0.720, 0.721 and 0.760 cosine similarities, respectively.

We analyze the predictions of all the four individual models (i.e., CNN, LSTM, GRU & SVR) as reported in Table IV and observe that the performances of these systems are numerically quite similar. However, when we qualitatively analyze the predictions, we observe the contrasting nature of these individual models. In most of the case, the predictions of each individual model are non-overlapping to each other. For some examples, one system (say *A*) obtains relatively correct predictions (i.e., prediction closer to the desired or gold intensity score) than the competing systems (*B, C & D*), while for some other examples system *A* reports less accurate prediction than the other systems. We depict the contrasting behavior of these competing systems with respect to the gold score in Figure 3. Further, we highlight a few scenarios in Figure 3a to make the differences more apparent. In the first highlighted region, CNN performs better than other systems, whereas, in the second case both CNN and GRU report closer values to the gold score. Subsequently, in the third region, SVR is the best among all, however, in the fourth region, it has the least performance. Similarly, SVR has the best and least performances for the sixth and seventh highlighted regions, respectively. In contrast, LSTM and GRU both have better performances for the fifth highlighted region. Such contrasting behavior of these four systems motivates us to combine the predictions for overall better performance.

Consequently, we train an MLP based stacked ensemble on top of the best performing individual models, i.e., one each for CNN, LSTM, GRU and Feature-SVR. In response, the MLP ensemble network reports enhanced scores for each of the datasets as reported in Table V. We obtain Pearson scores of 0.747, 0.712, 0.755 & 0.779, for '*anger*', '*joy*', '*sadness*' and '*fear*', respectively. The ensemble network improves the performance of individual systems by a significant margin of 4, 2, 5 & 5 Pearson scores, respectively. Similarly, the proposed ensemble approach aids in improving the performance of the individual systems by 3 & 2 cosine similarity points at 0.797 and 0.786, respectively for the *microblog messages* and *news headlines*.

We compare our proposed system with state-of-the-art systems for both

TABLE V Results of the ensemble model for financial sentiment analysis and emotion analysis tasks. Ensemble models (*CNN#, LSTM#, GRU#* & *SVR#*) refer to the best models of *CNN, LSTM, GRU* & *Feature-SVR* based models of Table IV.

		FINANCIAL SENTIMENT		EMOTION ANALYSIS				
	ENSEMBLE MODELS	MICROBLOGS	NEWS	ANGER	JOY	SADNESS	FEAR	AVERAGE
E1	CNN4 + LSTM3 + GRU5 + SVR6	**0.797**	0.765	–	–	–	–	–
E2	CNN1 + LSTM5 + GRU1 + SVR6	0.779	**0.786**	–	–	–	–	–
E3	CNN1 + LSTM2 + GRU2 + SVR2	–	–	**0.747**	0.705	0.744	0.769	0.741
E4	CNN2 + LSTM2 + GRU2 + SVR3	–	–	0.731	**0.712**	0.745	0.772	0.740
E5	CNN2 + LSTM2 + GRU2 + SVR2	–	–	0.738	0.702	**0.755**	0.768	0.740
E6	CNN2 + LSTM1 + GRU2 + SVR3	–	–	0.732	0.707	0.748	**0.779**	0.741

TABLE VI Comparison with the state-of-the-art systems. Emotion Analysis: Prayas, IMS & IITP were the ranked first, second & fifth systems at EmoInt-2017 [7]. Systems [56]–[58] are the recent works evaluated on the EmoInt-2017 datasets. Sentiment Analysis: ECNU & Fortia-FBK were the top performing systems at SemEval-2017 task 5 [8] for microblogs and news headlines, respectively. System [59]$^+$ 10-fold CV.

	FINANCIAL SENTIMENT		EMOTION ANALYSIS				
SYSTEMS	MICROBLOGS	NEWS	ANGER	JOY	SADNESS	FEAR	AVERAGE
SYSTEM [59]$^+$	0.726	0.655	–	–	–	–	–
ECNU [60]	0.777	0.710	–	–	–	–	–
FORTIA-FBK [61]	–	0.745	–	–	–	–	–
BASELINE [7]	–	–	0.625	0.635	0.706	0.620	0.647
IITP [62]	–	–	0.649	0.657	0.709	0.713	0.682
SYSTEM [56]	–	–	0.723	0.671	0.735	0.725	0.713
SYSTEM [57]	–	–	0.716	0.692	0.733	0.728	0.717
IMS [63]	–	–	0.705	0.690	0.767	0.726	0.722
SYSTEM [58]	–	–	0.718	0.717	**0.771**	0.729	0.734
PRAYAS [64]	–	–	0.732	**0.732**	0.765	0.762	0.747
PROPOSED SYSTEM	**0.797**	**0.786**	**0.747**	0.712	0.755	**0.779**	**0.748**

the problems. Table VI shows the comparative results on test datasets. For emotion analysis task, Prayas [64] and IMS [63] are the two best-performing systems with the average Pearson scores of 0.747 and 0.722 as compared to the average Pearson score 0.748 of our proposed model. Prayas [64] used an ensemble of five different neural network models (a feed-forward model, a multi-tasking feed-forward model and three joint CNN-LSTM models). The final predictions were generated by a weighted average of the base models. IMS [63] employed a random forest regression model on concatenated lexicon features and CNN-LSTM features. IMS used an external lexicon source (ACVH-Lexicons) containing unmodified ratings for arousal, concreteness, valency and happiness (ACVH), which was not part of the original baseline model [7]. They also use a 2016 Twitter corpus containing 50 million tweets with 800 million tokens containing emotion hashtags and popular general hashtags to train their word embeddings. Our proposed system performs better than all these existing best systems without using such external resources. Recently, Xie et al. [57] proposed a CNN based model that associates attention weights for each convolution windows. They defined a new activation function (inspired by *ReLu* activations) for predicting the intensities in the range 0 to 1. In another work, Khosla et al. [56] proposed affect-enriched distributional word representation (Aff2Vec) model to effectively encode the affective and emotional word semantics.

For financial sentiment analysis task, ECNU [60] reported cosine similarities of 0.777 and 0.710, respectively in *microblog messages* and *news headlines* domains against the cosine similarities of 0.797 and 0.786 of the proposed system. The underlying approach of ECNU utilized various regressors (e.g., SVR, XGBoost regressor, AdaBoost regressor, etc.) as the base models and then averaged the predictions of these regressors for the final scores. These regressors were trained on an optimized set of features obtained using the application of *hill climbing*. In comparison, Fortia-FBK [61] utilized a CNN architecture and obtained a cosine similarity of 0.745 for the *news headlines* domain. The employed CNN architecture was assisted by various sentiment lexicons. In another work, Atzeni et al. [59] proposed a feature-driven approach to study the effect of lexical (n-grams) and semantic (BabelNet, Semantic frames) features for predicting the intensity of sentiment. They experimented with seven different feature combinations and five different regressors for the study.

The proposed approaches of these systems (ECNU [60], Fortia-FBK [61] and Atzeni et al. [59]) have a major limitation, i.e., their proposed method does not perform well across domains. The proposed system of ECNU performed reasonably well for *microblog messages* domain but performed below par for *news headlines* domain. We observe similar trends for Atzeni et al. [59] as well, where the 10-fold CV performance in news headline is not at par with the microblog messages. In comparison, Fortia-FBK obtained decent performance for *news headlines* but did not report the results for *microblog messages*. However, in comparison, our proposed system reports better performance than both the existing best systems (i.e., ECNU and Fortia-FBK) for microblogs and news headlines. It suggests that our proposed system is more generic and robust in predicting the sentiment scores.

To further show the efficacy of our proposed approach we perform a statistical significance test on the obtained outputs. We observe that the predicted outputs are statistically significant (*t-test*) with *p*-values 0.004 & 0.037 for *microblog messages* and *news headlines*, respectively. For the emotion analysis, we compute *p-value* for the overall Pearson score as 0.017 which is significantly lesser than the threshold 0.05.

D. Discussion

The evaluation shows that feature-based model performs better compared to the deep learning models. A possible reason for such behavior would be the lack of training samples used in the deep learning model which, in general, requires a good amount of data instances to learn. In our case, the number of training samples in all the six datasets (2 sentiments + 4 emotions) is in the range of 1000–1700 samples only. We believe that with more training samples the performance gap between the deep learning and feature-based models would be even lesser and the ensemble model would be able to further improve upon that.

We also tried combining various feature-based models (different combinations of feature-SVR models (SVR1, SVR2, …, SVR6) of Table IV) but the resultant ensemble predictions were not at par. A possible reason is that the candidate models were not diverse enough. Stacking algorithm usually requires a diverse set of candidates to improve the performance and to prevent overfitting. We ensure diversity by combining 3 deep learning models and a feature-based model after analyzing their predictions.

E. Comparison with Other Ensemble Techniques

We also compare our proposed system with other ensemble techniques. For comparison, we utilized three standard ensemble techniques, e.g., Gradient Boosting [65], AdaBoost [66] and Bagging Regressor [48]. Shallow decision trees are used as meta estimators for Gradient Boosting model. For AdaBoost and Bagging Regressor, we use Nearest Neighbor Regressor as the base estimator. Results are reported in Table VIII. Although the results reported by these models were quite encouraging, the MLP based ensemble improves at least 1—2 cosine/Pearson point over the best performing non-MLP based ensemble for all the cases. It suggests that, indeed, the proposed MLP is a better choice for the ensemble.

F. Error Analysis

We perform qualitative analysis and observe that the proposed system faces difficulties in the following scenarios.

1) **Implicit sentiment:** The presence of implicit sentiments often causes the model to mispredict the intensity. For example, *"I'm such a shy person, oh my lord."*, the gold intensity is 0.833, but our model predicts 0.494. For *"Tesco breaks its downward slide by*

TABLE VII Qualitative analysis of the error cases.

DOMAIN	TEXT	ACTUAL	PREDICTED	REMARKS
EMOTION-FEAR	*I'M SUCH A SHY PERSON, OH MY LORD.*	0.833	0.494	**IMPLICIT SENTIMENT/ EMOTION**
SENTIMENT	*TESCO BREAKS ITS DOWNWARD SLIDE BY CUTTING SALES DECLINE IN HALF.*	0.172	−0.694	
SENTIMENT	*IS $FB A BUY? TOPEKA CAPITAL MARKETS THINKS SO.*	−0.373	0.363	**NUMBERS AND SYMBOLS**
	BEST STOCK: $WTS +15%	0.857	0.106	
EMOTION-JOY	*CANNOT WAIT TO SEE YOU HONEY!*	0.770	0.160	**IMPLICIT EMOTION WITH NEGATION**
	I SEE THINGS IN THE CLOUDS THAT OTHERS CANNOT SEE SO I CAN BE LATE.	0.620	0.220	
EMOTION-ANGER	*ALWAYS SO HAPPY TO SUPPORT YOU BROTHER, KEEP THAT FIRE BURNING.*	0.132	0.418	**METAPHORIC SENTENCES**

TABLE VIII Comparison with the other ensemble techniques.

	FINANCIAL SENTIMENT		EMOTION ANALYSIS
ENSEMBLE METHOD	**MICROBLOGS**	**NEWS**	**AVERAGE**
PROPOSED SYSTEM-MLP	**0.797**	**0.786**	**0.748**
GRADIENT BOOST	0.773	0.762	0.737
ADABOOST	0.780	0.771	0.729
BAGGING	0.779	0.768	0.735

cutting sales decline in half", the gold sentiment is 0.172 and our model predicts −0.694.

2) **Numbers and Symbols:** The presence of numeric entities and special symbols often confuses the model into predicting sentiments with higher error. We observe that for, *"Is $FB a BUY? Topeka Capital Markets thinks so."* and *"best stock: $WTS +15%"*, predicted sentiment scores are 0.363 and 0.106 but the gold sentiment scores are −0.373 and 0.857, respectively.
3) **Implicit emotion with negation:** The intensity of tweets containing emotions derived from various negation phrases like '*no wonder*', '*cannot wait*', etc. are often predicted incorrectly in the 'joy' datasets. For instance, gold intensities of *"Cannot wait to see you honey!"* and *"I see things in the clouds that others cannot see so i can be late"* are 0.77 and 0.62, but the model predicted relatively lower intensities of 0.16 and 0.22, respectively. On in-depth analysis, we observe that for '*joy*' datasets implicit negation is one of the prime factors in our system's relatively below-par performance. We tried to incorporate specific negation features into account, but the performance did not improve for the '*joy*' dataset and we also observed performance degradation for the other datasets.
4) **Metaphoric sentences:** Intensities are often wrongly predicted for the tweets containing metaphors. For example, expressed emotion for the sentence *"Always so happy to support you brother, keep that fire burning"* is anger with an intensity of 0.132, whereas the predicted intensity was 0.418.

Table VII presents a summary of the frequent error cases.

VI. Conclusion

In this paper, we have presented an ensemble framework for intensity prediction of sentiment and emotion. We develop three deep learning models based on LSTM, CNN and GRU and one feature-driven classical supervised model based on SVR. These are combined using an MLP classifier. With the help of our experiments, we tried to establish that the proposed method is applicable to different domains of problems. In total, we evaluate our proposed technique for three different problem domains, i.e., sentiment intensity prediction in financial *microblog messages*, sentiment intensity prediction in financial *news headlines* and emotion intensity prediction in *generic tweets*. The proposed model shows impressive results for all the problem domains. We have implemented a series of linguistic and semantic heuristics for our analysis of the noisy text in tweets and news headlines. We have evaluated our proposed system on the benchmark setup of EmoInt-2017 and SemEval-2017 for emotion and sentiment analysis, respectively. Comparisons suggest that our proposed model performs significantly better than the state-of-the-art systems with the improvement of 2.0 and 4.1 points on the tasks of sentiment prediction of financial *microblog messages* and *news headlines*. For emotion analysis, our proposed model also performs comparatively better than the state-of-the-art models.

As future work, we would like to build an end-to-end stock market prediction system, which should be able to forecast the stock prices of a given company based on public sentiments. For emotion analysis, we would like to investigate other emotion classes along with the tweets with mixed-emotions.

VII. Acknowledgment

Asif Ekbal acknowledges the Young Faculty Research Fellowship (YFRF), supported by Visvesvaraya PhD scheme for Electronics and IT, Ministry of Electronics and Information Technology (MeitY), Government of India, being implemented by Digital India Corporation (formerly Media Lab Asia).

References

[1] S. Poria, E. Cambria, R. Bajpai, and A. Hussain, "A review of affective computing: From unimodal analysis to multimodal fusion," *Inform. Fusion*, vol. 37, pp. 98–125, Sept. 2017.
[2] P. Ekman, "An argument for basic emotions," *Cogn. Emot.*, vol. 6, no. 3–4, pp. 169–200, 1992.
[3] E. Cambria, S. Poria, A. Gelbukh, and M. Thelwall, "Sentiment analysis is a big suitcase," *IEEE Intell. Syst.*, vol. 32, no. 6, pp. 74–80, Nov. 2017.
[4] Y. Ma, H. Peng, and E. Cambria, "Targeted aspect-based sentiment analysis via embedding commonsense knowledge into an attentive LSTM," in *Proc. 32nd AAAI*

Conf. Artificial Intelligence, New Orleans, LA, 2018, pp. 5876–5883.
[5] R. J. Davidson, K. R. Sherer, and H. H. Goldsmith, Eds., *Handbook of Affective Sciences*. New York: Oxford Univ. Press, 2009.
[6] J. A. Russell and L. F. Barrett, "Core affect, prototypical emotional episodes, and other things called emotion: Dissecting the elephant," *J. Pers. Soc. Psychol.*, vol. 76, no. 5, pp. 805–819, May 1999.
[7] S. Mohammad and F. Bravo-Marquez, "WASSA-2017 shared task on emotion intensity," in *Proc. Workshop Computational Approaches to Subjectivity, Sentiment and Social Media Analysis (WASSA)*, Copenhagen, Denmark, 2017, pp. 34–49.
[8] K. Cortis, A. Freitas, T. Daudert, M. Huerlimann, M. Zarrouk, and B. Davis, "SemEval-2017 Task 5: Fine-grained sentiment analysis on financial microblogs and news," in *Proc. 11th Int. Workshop SemEval*, Vancouver, Canada, 2017, pp. 519–535.
[9] E. Cambria, D. Rajagopal, D. Olsher, and D. Das, "Big social data analysis," in *Big Data Computing*, vol. 13, R. Akerkar, Ed. Chapman and Hall/CRC, 2013, pp. 401–414.
[10] F. Z. Xing, E. Cambria, and R. E. Welsch, "Intelligent asset allocation via market sentiment views," *IEEE Comput. Intell. Mag.*, vol. 13, no. 4, pp. 25–34, 2018.
[11] E. Cambria, A. Hussain, T. Durrani, C. Havasi, C. Eckl, and J. Munro, "Sentic computing for patient centered applications," in *Proc. IEEE Int. Conf. Signal Processing (ICSP)*, 2010, pp. 1279–1282.
[12] E. Cambria, "Learning binary codes with neural collaborative filtering for efficient recommendation systems," *Knowl.-Based Syst.*, vol. 172, pp. 64–75, 2019.
[13] A. J. Smola and B. Schölkopf, "A tutorial on support vector regression," *Stat. Comput.*, vol. 14, no. 3, pp. 199–222, Aug. 2004.
[14] Y. Kim, "Convolutional neural networks for sentence classification," in *Proc. Conf. Empirical Methods in Natural Language Processing*, Doha, Qatar, Oct. 2014, pp. 1746–1751.
[15] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.
[16] K. Cho, B. van Merrienboer, D. Bahdanau, and Y. Bengio, "On the properties of neural machine translation: Encoder–decoder approaches," in *Proc. 8th Workshop Syntax, Semantics and Structure in Statistical Translation (SSST)*, Doha, Qatar, 2014, pp. 103–111.
[17] M. S. Akhtar, A. Kumar, D. Ghosal, A. Ekbal, and P. Bhattacharyya, "A multilayer perceptron based ensemble technique for fine-grained financial sentiment analysis," in *Proc. Conf. Empirical Methods in Natural Language Processing*, Copenhagen, Denmark, 2017, pp. 540–546.
[18] D. Ghosal, M. S. Akhtar, A. Ekbal, and P. Bhattacharyya, "Deep ensemble model with the fusion of character, word and lexicon level information for emotion and sentiment prediction," in *Neural Information Processing*. Cham: Springer-Verlag, 2018, pp. 162–174.
[19] C.-H. Wu, Z.-J. Chuang, and Y.-C. Lin, "Emotion recognition from text using semantic labels and separable mixture models," *ACM Trans. Asian Language Inform. Process.*, vol. 5, no. 2, pp. 165–183, 2006.
[20] D. T. Ho and T. H. Cao, "A high-order hidden Markov model for emotion detection from textual data," in *Proc. 12th Pacific Rim Conf. Knowledge Management and Acquisition for Intelligent Systems*, Kuching, Malaysia, 2012, pp. 94–105.
[21] E. Cambria, "Affective computing and sentiment analysis," *IEEE Intell. Syst.*, vol. 31, no. 2, pp. 102–107, Mar. 2016.
[22] C. Strapparava and A. Valitutti, "WordNet-Affect: An affective extension of WordNet," in *Proc. 4th Int. Conf. Language Resources and Evaluation*, Lisbon, Portugal, 2004, pp. 1083–1086.
[23] Z. Teng, F. Ren, and S. Kuroiwa, "Recognition of emotion with SVMs," in *Proc. ICIC 2006: Computational Intelligence*, Kunming, China, 2006, pp. 701–710.
[24] C. Yang, K. H. Y. Lin, and H. H. Chen, "Emotion classification using web blog corpora," in *IEEE/WIC/ACM Int. Conf. Web Intelligence*, Silicon Valley, CA, 2007, pp. 275–278.
[25] F. Xing, E. Cambria, and R. Welsch, "Natural language based financial forecasting: A survey," *Artif. Intell. Rev.*, vol. 50, no. 1, pp. 49–73, June 2018.
[26] A. Devitt and K. Ahmad, "Sentiment polarity identification in financial news: A cohesion-based approach," in *Proc. 45th Annu. Meeting Association of Computational Linguistics*, Prague, Czech Republic, 2007, pp. 984–991.
[27] N. O'Hare et al., "Topic-dependent sentiment analysis of financial blogs," in *Proc. Workshop Topic-sentiment Analysis for Mass Opinion*, Hong Kong, China, 2009, pp. 9–16.
[28] R. Schumaker and H. Chen, "Textual analysis of stock market prediction using breaking financial news: The AZFinText system," *ACM Trans. Off. Int. Syst.*, vol. 27, no. 2, Feb. 2009.
[29] J. Si, A. Mukherjee, B. Liu, Q. Li, H. Li, and X. Deng, "Exploiting topic based twitter sentiment for stock prediction," in *Proc. 51st Annu. Meeting Association of Computational Linguistics (Volume 2: Short Papers)*, Sofia, Bulgaria, 2013, pp. 24–29.
[30] M. V. de Kauter, D. Breesch, and V. Hoste, "Fine-grained analysis of explicit and implicit sentiment in financial news articles," *Expert Syst. Appl.*, vol. 42, no. 11, pp. 4999–5010, July 2015.
[31] N. Oliveira, P. Cortez, and N. Areal, "On the predictability of stock market behavior using stocktwits sentiment and posting volume," in *Proc. 16th Portuguese Conf. Artificial Intelligence*, Angra do Heroísmo, Portugal, 2013, pp. 355–365.
[32] M. S. Akhtar, D. Ghosal, A. Ekbal, P. Bhattacharyya, and S. Kurohashi, "All-in-one: Emotion, sentiment and intensity prediction using a multi-task ensemble framework," *IEEE Trans. Affect. Comput.*, p. 1, 2019.
[33] J. Pennington, R. Socher, and C. D. Manning, "GloVe: Global vectors for word representation," in *Proc. Empirical Methods in Natural Language Processing*, Doha, Qatar, 2014, pp. 1532–1543.
[34] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proc. Advances in Neural Information Processing Systems*, Lake Tahoe, NV, 2013, pp. 3111–3119.
[35] J. Wiebe and R. Mihalcea, "Word sense and subjectivity," in *Proc. Int. Conf. Computational Linguistics/Association of Computational Linguistics*, Sydney, Australia, 2006, pp. 1065–1072.
[36] X. Ding, B. Liu, and P. S. Yu, "A holistic lexicon-based approach to opinion mining," in *Proc. Int. Conf. Web Search and Data Mining*, Palo Alto, CA, 2008, pp. 231–240.
[37] S. Mohammad, S. Kiritchenko, and X. Zhu, "NRC-Canada: Building the state-of-the-art in sentiment analysis of tweets," in *Proc. 2nd Joint Conf. Lexical and Computational Semantics (*SEM), Volume 2: Proc. 7th Int. Workshop on SemEval*, Atlanta, GA, USA, 2013, pp. 321–327.
[38] S. Baccianella, A. Esuli, and F. Sebastiani, "SentiWordNet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining," in *Proc. 7th Int. Conf. Language Resources and Evaluation*, Valletta, Malta, 2010, pp. 2200–2204.
[39] S. M. Mohammad and P. D. Turney, "Crowdsourcing a word-emotion association lexicon," *Comput. Intell.*, vol. 29, no. 3, pp. 436–465, Aug. 2013.
[40] F. Bravo-Marquez, E. Frank, S. M. Mohammad, and B. Pfahringer, "Determining word–emotion associations from tweets by multi-label classification," in *Proc. IEEE/WIC/ACM Int. Conf. Web Intelligence*, Omaha, NE, 2016, pp. 536–539.
[41] S. Mohammad, "#Emotional tweets," in *Proc. 1st Joint Conf. Lexical and Computational Semantics*, Montréal, Canada, 2012, pp. 246–255.
[42] F. Å. Nielsen, A new ANEW: Evaluation of a word list for sentiment analysis in microblogs. 2011. [Online]. Available: arXiv:1103.2903
[43] C. H. E. Gilbert, "VADER: A parsimonious rule-based model for sentiment analysis of social media text," in *Proc. 8th Int. Conf. Weblogs and Social Media*, Ann Arbor, MI, 2014.
[44] T. Xiao, J. Zhu, and T. Liu, "Bagging and boosting statistical machine translation systems," *Artif. Intell.*, vol. 195, pp. 496–527, Feb. 2013.
[45] A. Ekbal and S. Saha, "Weighted vote-based classifier ensemble for named entity recognition: A genetic algorithm-based approach," *ACM Trans. Asian Language Inf. Process.*, vol. 10, no. 2, pp. 1:9–9:37, June 2011.
[46] M. S. Akhtar, D. Gupta, A. Ekbal, and P. Bhattacharyya, "Feature selection and ensemble construction: A two-step method for aspect based sentiment analysis," *Knowl.-Based Syst.*, vol. 125, pp. 116–135, June 2017.
[47] Y. Freund and R. E. Schapire, "Experiments with a new boosting algorithm," in *Proc. 13th Int. Conf. Machine Learning (ICML)*, Bari, Italy, 1996, pp. 148–156.
[48] L. Breiman, "Bagging predictors," *Mach. Learn.*, vol. 24, no. 2, pp. 123–140, Aug. 1996.
[49] J. Kittler, M. Hatef, R. P. W. Duin, and J. Matas, "On combining classifiers," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 3, pp. 226–239, Mar. 1998.
[50] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," *J. Mach. Learn. Res.*, vol. 11, pp. 3371–3408, Dec. 2010.
[51] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2014.
[52] T. N. Flynn and A. A. J. Marley, *Best-Worst Scaling: Theory and Methods*. Cheltenham, U.K.: Edward Elgar Publishing, 2014.
[53] S. Bird, E. Klein, and E. Loper, *Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit*. O'Reilly Media, 2009.
[54] M. S. Akhtar, U. K. Sikdar, and A. Ekbal, "IITP: Hybrid approach for text normalization in twitter," in *Proc. Workshop Noisy User-generated Text (WNUT)*, Beijing, China, 2015, pp. 106–110.
[55] T. Baldwin, M. C. de Marneffe, B. Han, Y.-B. Kim, A. Ritter, and W. Xu, "Shared tasks of the 2015 workshop on noisy user-generated text: Twitter lexical normalization and named entity recognition," in *Proc. Workshop Noisy User-generated Text (WNUT)*, Beijing, China, 2015, pp. 126–135.
[56] S. Khosla, N. Chhaya, and K. Chawla, "Aff2Vec: Affect–enriched distributional word representations," in *Proc. 27th Int. Conf. Computational Linguistics*, Santa Fe, NM, 2018, pp. 2204–2218.
[57] H. Xie, S. Feng, D. Wang, and Y. Zhang, "A novel attention based CNN model for emotion intensity prediction," in *Proc. Natural Language Processing and Chinese Computing*, Hohhot, China, 2018, pp. 365–377.
[58] S. Madisetty and M. S. Desarkar, "An ensemble based method for predicting emotion intensity of tweets," in *Proc. 5th Int. Conf. Mining Intelligence and Knowledge Exploration (MIKE)*, Hyderabad, India, 2017, pp. 359–370.
[59] M. Atzeni, A. Dridi, and D. R. Recupero, "Using frame-based resources for sentiment analysis within the financial domain," *Prog. Artif. Intell.*, vol. 7, no. 4, pp. 273–294, Dec. 2018.
[60] M. Jiang, M. Lan, and Y. Wu, "ECNU at SemEval-2017 Task 5: An ensemble of regression algorithms with effective features for fine-grained sentiment analysis in financial domain," in *Proc. 11th Int. Workshop SemEval*, Vancouver, Canada, 2017, pp. 888–893.
[61] Y. Mansar, L. Gatti, S. Ferradans, M. Guerini, and J. Staiano, "Fortia-FBK at SemEval-2017 Task 5: Bullish or bearish? Inferring sentiment towards brands from financial news headlines," in *Proc. 11th Int. Workshop SemEval*, Vancouver, Canada, 2017, pp. 817–822.
[62] M. S. Akhtar, P. Sawant, A. Ekbal, J. Pawar, and P. Bhattacharyya, "IITP at EmoInt-2017: Measuring intensity of emotions using sentence embeddings and optimized features," in *Proc. Workshop Computational Approaches to Subjectivity, Sentiment and Social Media Analysis (WASSA)*, Copenhagen, Denmark, 2017, pp. 212–218.
[63] M. Köper, E. Kim, and R. Klinger, "IMS at EmoInt-2017: Emotion intensity prediction with affective norms, automatically extended resources and deep learning," in *Proc. Workshop Computational Approaches to Subjectivity, Sentiment and Social Media Analysis (WASSA)*, Copenhagen, Denmark, 2017, pp. 50–57.
[64] P. Jain, P. Goel, D. Kulshreshtha, and K. K. Shukla, "Prayas at EmoInt 2017: An ensemble of deep neural architectures for emotion intensity prediction in tweets," in *Proc. Workshop Computational Approaches to Subjectivity, Sentiment and Social Media Analysis (WASSA)*, Copenhagen, Denmark, 2017, pp. 58–65.
[65] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," *Ann. Stat.*, vol. 29, no. 5, pp. 1189–1232, Oct. 2001.
[66] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," in *Proc. 2nd European Conf. Computational Learning Theory (COLT)*, Barcelona, Spain, 1995, pp. 23–37.