Predicting Information Diffusion on Twitter a Deep Learning Neural Network Model Using Custom Weighted Word Features

Amit Kumar Kushwaha, Arpan Kumar Kar, P. Ilavarasan

Predicting Information Diffusion on Twitter A Deep Learning Neural Network Model using Custom Weighted Word Features

Amit Kumar Kushwaha, Arpan Kumar Kar and P. Vigneswara Ilavarasan

Department of Management Studies, Indian Institute of Technology Delhi, New Delhi 110016, India

Kushwaha.amitkumar@gmail.com, arpan_kar@yahoo.co.in, vignes@iitd.ac.in

Abstract Researchers have been experimenting with various drivers of the diffusion rate like sentiment analysis which only considers the presence of certain words in a tweet. We theorize that the diffusion of particular content on Twitter can be driven by a sequence of nouns, adjectives, adverbs forming a sentence. We exhibit that the proposed approach is coherent with the intrinsic disposition of tweets to a common choice of words while constructing a sentence to express an opinion or sentiment. Through this paper, we propose a Custom Weighted Word Embedding (CWWE) to study the degree of diffusion of content (retweet on Twitter). Our framework first extracts the words, create a matrix of these words using the sequences in the tweet text. To this sequence matrix we further multiply custom weights basis the presence index in a sentence wherein higher weights are given if the impactful class of tokens/words like nouns, adjectives are used at the beginning of the sentence than at last. We then try to predict the possibility of diffusion of information using Long-Short Term Memory Deep Neural Network architecture, which in turn is further optimized on the accuracy and training execution time by a Convolutional Neural Network architecture. The results of the proposed CWWE are compared to a pre-trained glove word embedding. For experimentation, we created a corpus of size 230,000 tweets posted by more than 45,000 users in 6 months. Research experimentations reveal that using the proposed framework of Custom Weighted Word Embedding (CWWE) from the tweet there is a significant improvement in the overall accuracy of Deep Learning framework model in predicting information diffusion through tweets.

Keywords: Information Diffusion, Twitter analytics, Deep Learning, Convolutional Neural Network, Linguistic.

1 Introduction

Microblogging website Twitter has emerged as one of the primary online social media platforms across the globe for sharing opinions, interests, and points of view on events and issues varying from education, sports, online content to politics. Users on Twitter can associate their share of opinions related to a subject matter by writing a simple

tweet and posting it in real-time. The selection of right expletives and positioning the same in a tweet sentence plays a critical role in attracting the attention of right receivers (users of Twitter) of this tweet with similar interests inside the Twitter platform. Sharples [1] has claimed that readers on any platform of communication should treat writing as a creative act of writers using which they deliberately select phrases and position them to impose a certain interpretation through lexical and linguistic preferences. This hypothesis was first developed by Maun [2], showing how diverse set of writers from various backgrounds, are extremely cognizant of articulated aspects of writing, positioning the expletives with opinions that we should hypothesize as writing practice and as an act of thoughts representation. The idea of sentence writing includes the design of the factual or non-factual information of a topic, person, event or domain, followed by opinion generation through experience or knowledge expressed in words, within a framework requiring the expertise of linguistics as well as the nature of genres and domains in which the communication is taking place. The sentence then reaches the receiver (also referred to diffusion in modern world) to whom the original writer/sender was planning to convey the message to. The receiver in turn again uses the knowledge of linguistics to interpret the meaning which original sender was trying to convey. Hence linguistic competence is central to understand the degree of diffusion of information in any communication channel.

Social media application Twitter bring users with similar opinion across the globe on the same platform to communicated, cascade the information to a wider set of users beyond their group referred to as information diffusion (also referred to as retweet on Twitter) in a real-time basis. This, in turn, creates mass information on Twitter. The ability of Twitter to elevate the exposure of any information through a simple tweet is contained in various aspects of the same tweet. Like the number of users connected to the account from which the tweet got generated, how active the user has been in the past, how active the followers in the network of the user are on Twitter.

To achieve effective diffusion of information, other than the previously listed factors, one more aspect that plays a key role is: the sentence used in a tweet should be conveying the appropriate information with a factual point of view. Prior research appears a little apprehensive about the accurate enriching relationship between the selection of words, placing in a certain order, and about how the thoughts get reconstructed when we alter the sequence of words. The area of linguistics has well-described literature on how a well-formed sentence structure can guide the communication of inference and opinion expressed and at the same time increase the penetration of the sentence at the user level. Aligning the sentences for an appropriate theme and rheme and the idea of linearization of the words in which ‘what an author places first will drive the interpretation and penetration of rest that has to follow’[3] are known ideas to linguist researchers. Also is the goal of concluding remarks in a sentence [4]: because the initial vocable or noun in a phrase gains more importance, and professional writers will change the order of the words to engage the reader’s attention to the factual information using the active mode of writing a sentence. These are design alternatives which the writer of a sentence or a tweet on Twitter has to learn and there are options within this learning that require attention, be it direct or indirect, that a verbally spoken sentence is not the same as a written one. Using the linguistic aspect of feature formation is not

entirely explored in the area of research of feature extraction from verbatim of tweet[5,6-8].

Through this paper, we are presenting and contribute to the literature an original and creative approach combined with the deep learning neural network architecture and the branch of linguistics for feature extraction. We assign custom weights to the word vector matrix calculated by the index of these words in the neighborhood of the sequence of the rest of the words in the same sentence. We are further defining this feature engineering framework as Custom Weighted Word Embedding (CWWE). We then try to combine this feature engineering framework with the state of the art Long Short Term Memory(LSTM) to train an initial model that is further optimized by adding convolutional layer 1D on top of embedding to make it more time-efficient while training and improve the accuracy as well. For benchmarking the results of classification using the proposed architecture, we compare the overall accuracy with a CNN model running on the same architecture using a pre-trained glove word embedding instead of the experimented Custom Weighted Word Embedding (CWWE). In the scope of current research, we hypothesize that the proposed framework works better in classifying a tweet as retweetable or not if we take into consideration the linguistic art of sentence formation. The ideology behind the research in the current paper is to create the features in such a way that these are not only geography independent but also independent of any theme of any event which makes the proposed framework easily generalizable in any situation.

The rest of the research paper is structured as: in Section 2, prior literature is discussed for defining information diffusion, social media analytics and deep leaning architecture, in section 3, we describe the proposed CWWE(Custom Weight Word Embedding). To begin the experimentation phase, we first train a base LSTM model, record the results and define as “baseline results1”. We then introduce our novel framework with Convolutional Neural Network and record the results as “proposed framework results”. With the same CNN architecture used to test the proposed features but predict the results using a pre-trained glove model of 10,000-word embedding instead of CWWE and define the results as “baseline results2”. Finally, we discuss the findings by comparing “baseline results1” vs. “baseline results2” vs. “proposed framework results” and conclude with proposing future scope of research in Section 4.

2 Prior Literature

The literature review has been designed to follow the following flow: we start with defining the information diffusion, how social media analytics has helped the researchers’ community to gain insights about the domain, followed by discussing the established vectorization process as one of the techniques from the social media analytics toolkit and concluding with deep learning architectures used for unstructured data like tweet text.

2.1 Information Diffusion and Social Media Analytics

With the internet opening the barriers of communication among people across the globe, there is a need to perform research on effective ways of communication. When there is more communication there is more information diffusion too [9-11]. We define information diffusion as the travel process of a piece of knowledge having factual or non-factual details about a person, event, domain or place getting initiated from a sender and reaching a set of receivers through a channel or carrier. In the case of Twitter [12,13], the carrier is the tweet posted on the channel Twitter, the sender is the user posting the tweet and receivers are followers of the user posting the tweet. There are three aspects of information diffusion: innovation in communication practice, factual information and right audience(user-network) which plays an important role in predicting the degree of information diffusion.

With the increase of internet penetration, social media platforms have started to gain importance as a channel for message penetration and propagation. Twitter is constantly explored by organizations and individuals for multiple communications which creates an abundance of information and hence researchers have started exploring various analysis frameworks to gain insights and one such framework is Social Media Analytics (SMA). SMA as an evolving cross-domain research area for gaining insights through social media has made the usage of Twitter for information communication and diffusion across multiple domains possible. Many researchers from their work have proved the potential of SMA across different academic disciplines and business domains. Prior work strongly shows that SMA can provide important insights around disciplines such as marketing [14], in deliberation on political [15] and social concerns [16], in emergency [17], for building public relation [18] or even ruling out possibility of polarization of any major event [19]. Social media platforms like Twitter can also help us to gain insights into any specific domain [20] or technology [21]. However, in the domain of social media, penetration and diffusion of communication has three strong pillars, firstly managing and controlling misinformation [22], secondly use machine learning to automatically predicting the information(tweet) diffusion speed, scale, range and trend [23] and thirdly use frameworks to evaluate the credibility of a tweet, i.e. whether to trust the tweet and the person writing the tweet.

2.2 Word Features and Deep Learning Architecture

The most popular model in SMA for predicting the information diffusion (retweet on Twitter) is to convert sentences to words and then to vectors and finally use these vectors as features in classifiers to predict the probability of reweet. Most of the prior experimentation performed in the retweet prediction is based on the opinion similarity analysis of the words used in the tweet or at the user account and network level. Xu and Yang [24] have proposed term frequency-inverse document frequency hinged BOW(bag of words) for every tweet in a content-based model for retweet prediction. The research by [25] presented the work of information-sharing approach and trend of users on social media platforms. The assessment of the importance of these variables in the same research presented that the term frequency–inverse document frequency

weighted BOW homogeneity of retweeted tweets performing as the highest significant variable in predicting user retweet trend. Brief research also exists around computational linguistic research [26] to predict the impact of any information, but there certainly exist a gap for thorough research which has motivated us to take up this research.

Many people try to reword the original sentence in their own words with the intent of communicating the original message in a much better way. However, as such, any literature or verbatim synopsis is already abstractive in nature and has very little possibility of reproduction of the original sentence. With all deep learning architectures getting noticed by researchers in the field of unstructured data, it has become a more practical option for any NLP analysis, researchers have begun to consider these frameworks as a fully data-driven alternative to abstractive text summarization. The possibility of storing the meaning of a sentence with abstract features extracted is not completely explored. In the latest research, Konstantin Lopyrev [27] relates an application of an encoder-decoder RNN(recurrent neural network) with LSTM units and notices that the framework starts creating headlines from the actual text of news articles. In a slightly different but related work, Alexander M. Rush [28] presented a fully data-driven framework for sentence summarization. His approach makes use of an attention driven algorithm that creates each word from the input sentence as summary transformed. Sumit Chopra [29] proposes a constraint-based RNN (recurrent neural network) which can easily create an abstract of any input phrase. The transformation has been done by a unique approach in which a convolutional encoder is implemented to make sure that the corresponding decoder centers around the proper input words at each phase of generation. In an alternate paper that sharply resonates with our proposed framework, Ramesh Nallapati [30] has proposed a summary-based text abstraction using Attentional Encoder-Decoder RNN (Recurrent Neural Network), and show that these models achieve state-of-the-art performance on two different corpora [31,32,33]. Our work starts with a similar-looking framework but with the focus of extracting and storing the original sense of a sentence (tweets on Twitter) in very abstract features(CWWE) and assigning custom weights which will be further used to predict the diffusion(retweet) of the same sentences (tweets on Twitter).

3 CWWE-based LSTM-CNN model experimentation

3.1 Data collection

The first step in developing a classifier is to create a training corpus. In our study, we are focusing on collecting data from Twitter. The reason behind selecting Twitter is its flexible architecture that allows one-to-many communication and is less restricted when compared to other social networking sites like Facebook.

We specifically targeted to collect tweets for “Game of Thrones season8” because there will not be any sponsored posts and this series (particularly season8) was liked, watched and hated by viewers across the globe from different ethnicity. Hence the process of data extraction, gave us tweets originated at different parts of the world with varied writing styles. To achieve this, we used the following tags: [#GOTS08, #Game

of Thrones Season8, #Game of Thrones Final Season, # Game of Thrones Last Season, #GOT Season8]. Through streaming API of Twitter, we can scrap (free) one percent of tweets posted every day. Using Python as a tool, we used streaming API to collect tweets during the period: [April 14 to May 19, 2019]. This helped us to create a corpus of 300,000 tweets relevant to our analysis. Out of 300,000 tweets, a corpus of 234,884 tweets was created after data cleaning and removing the tweets which were posted by major official accounts. Finally, 234,884 tweets were labeled as "Viral" vs "Non-Viral" basis if the number of retweets of the tweet was greater than '10' referred to as "Viral" and the tweets for which the retweets was less than '10' was referred to as "Non-Viral" tweets. Table1 below represents the exploratory description of the tweet corpus.

	Tweet corpus size	Distinct users	Viral tweets	Non-Viral tweets	Average age of the twitter account in the corpus	Average number of tweets per user per day	Average number of followers per user
#	234,884	46,302	56,936	177,948	6 years	9	7200

Table 1. Descriptive Statistics of Data Set

3.2 Data cleaning

The pretreatment of the tweet text is a mandated step as it cleans and converts the actual tweet to structured features matrix, ready for analysis, i.e., it becomes easier to extract rich and factual information from the tweet and apply machine learning algorithms to it. If we bypass this data preparation step, then there is a highly likely a chance that we are dealing with noisy and incompatible data. The goal of this data cleaning exercise is to not only smoothen out the noise, the ones that are much less relevant to uncover the sentiment of tweets together with punctuation, special characters, numbers, and phrases that do not convey a lot of weight in context to the text. In the next steps of data cleaning, we are converting unstructured data of text to more structured data by generating the matrix of words extracted from the original tweet. If the initial data cleansing of tweets is done diligently, then the resulting feature space will also be of good quality.

Initial data cleaning makes every tweet to go through the following phases: Firstly, we know that owing to security concerns, Twitter never reveals the actual Twitter handle posting the tweet. Hence as part of the data cleaning process, we clear the special character sign of "@" which masks the Twitter handle. Secondly, we also know from the linguistic literature that special characters, alphanumeric, numeric and lingos do not convey any specific factual information hence, we cleaned any other special characters, alphanumeric, numeric and lingos from the actual tweets. Thirdly we have replaced most of the shorter words with the root words like: "haven't":" have not", "doesn't": "does not". Once every tweet loops through the above 3 steps we extract words from all the tweets and create a matrix with all the words from the corpus as columns. However, we have a huge corpus of tweets and hence the size of the matrix of words also

becomes large. To reduce the dimension, we reduced same versions of the words to its original root word, like: {like, likable}: {like}.

3.3 Build a preliminary LSTM model with standard word embedding

For simplicity, we start with the retweet column which represents how many re-tweets a tweet has received. Tweets that have more than ten (‘10’) as the retweet value are considered as the tweets which got diffused easily and strongly and will be labeled as “Retweeted” tweets for our research. The tweets for which the retweet column has less than ten (‘10’) as value will be labelled as the tweets which were not able to diffuse (were not retweeted) on twitter. We define this new column as “Retweet Identifier” which will have binary values and will be treated as the class label in our research analysis. Because we have a class label classifier (retweet identifier) to label each tweet, the research problem can now be defined as supervised learning and the theories of any classifier algorithm can be implemented to train, validate and test our model. We start our feature engineering with first tokenizing the text and convert them to word vectors. For feature dimension control and reduction, we are placing a constraint of 50 words on each tweet. This means that tweets having texts shorter than 50 words are padded with zeros, and longer texts are truncated. We now start with the model training.

The architecture of the neural network starts with an embedding layer wherein each word from a lower dimension is projected to a higher dimension, which in turn helps the network to learn every word vector in a more descriptive manner. The layer takes 20000 as the first argument, which is the size of our vocabulary, and 100 as the second input parameter, which is the dimension of the embedding. The third parameter has been constrained at 50 for the maximum size of every tweet sentence. We have also divided the entire tweet corpus into three sections: training (on which we let the model to train and optimize the parameters by minimizing the loss function), validation (on which the model validates the new parameters selected after every iteration) and test data (on which the trained model is tested for its performance on an unseen data). Model results at the end of training at this step are recorded and represented in table2.

Test Data Set Accuracy			
LSTM	0.52		
Confusion Matrix			
Model		Retweet	
		Yes	No
	Yes	10482	15539
	No	2561	9000
Precision = 0.8 Recall = 0.4			

Table 2. Model results for a preliminary LSTM model

3.4 Build a custom weight word embedding matrix

With the hypothesis of establishing information diffusion as a function of the weights derived from position based linguistic features [16], we pursue with the first step of our proposed CWWE framework. To establish the relative position index, we extract each word of a tweet as an independent feature and locate the relative presence of this word feature in the presence of other words in the respective tweet. We also try to see the context in which the word is used in the tweet while conveying a message. We try to explain the above concept with a basic illustration below:

James played **brilliantly** [manner] in the **match** [place] on **Saturday** [time] evening.

vs.

The **match** [place] played on **Saturday** [time], James performed comparatively good from his last game **good** [manner].

In the first sentence, the writer gives more emphasis by using the word **brilliantly** at the beginning of the sentence and the reader might not read the entire sentence to interpret that James performed brilliantly in the match. On the contrary in the second sentence, how the player played is being placed at last, which might leave less emphasis on the player James and some of the readers might even skip that part. Considering the tweet was about player James, we hypothesize that if the word **brilliantly** is used at the beginning there is highly likely chance that this tweet will diffuse (will be retweeted) in the Twitter network faster as compared to the second sentence. With this hypothesis, we will be testing the proposed framework by assigning the custom weights depending upon the relative position of the words. The details of the framework are explained in the following steps below.

Step1: After the cleaning the tweet text, performing the text to Word2Vector matrix

Tweet text	James	played	brilliantly	match	Saturday	good
James played brilliantly in the match on Saturday evening	1	1	1	1	1	0
The match played on Saturday, James performed compara-tively good from his last game	1	1	0	1	1	1

Table 3. Step1 of building a custom weight word embedding matrix

Step 2: After generating the Word2Vector matrix, we now try to find the length of each tweet text, mid-point index, first quartile and third quartile basis the length of text

Tweet text	Length	Mid-point index	First quartile	Third quartile
James played brilliantly in the match on Saturday evening	9	5	3	6
The match played on Saturday, James performed comparatively good from his last game	13	7	4	9

Table 4. Step2 of building a custom weight word embedding matrix

Step 3: We now look at the position index of each word from the Word2Vector matrix in the tweet text and check if the index falls in the first quartile, close to mid-point, in the third quartile or beyond the third quartile.

Tweet text	James	played	brilliantly	match	Saturday	good
James played brilliantly in the match on Saturday evening	First	First	Third	Third	Third	0
The match played on Saturday, James performed comparatively good from his last game	Mid	First	0	First	First	Third

Table 5. Step3 of building a custom weight word embedding matrix

Step 4: In the last step we assign a weight of 0.75 to the word which appears close to the first quartile, nothing to the words which are close to the middle part of the sentence and 0.25 to the words which are close to the third quartile of the tweet text. We further multiply these weights for the respective words in the respective tweets to the matrix from step1. The resulting matrix for the illustration looks like below.

Tweet text	James	played	brilliantly	match	Saturday	good
James played brilliantly in the match on Saturday evening	0.75	0.75	0.75	0.25	0.25	0
The match played on Saturday, James performed comparatively good from his last game	0	0.75	0	0.75	0.75	0.25

Table 6. Final step of building a custom weight word embedding matrix

3.5 Build a CNN model with CWWE

The LSTM model worked well with the initial Word2Vec matrix. However, it took forever to train one epoch. CNN is known to work well in image processing domains but the potential usage of CNN architecture in unstructured data like tweet analysis is still a fairly new area for research. Literature suggests that to increase the response time of the models while training on the data, we can add a convolutional layer, by doing so, uses a “filter” over the input feature matrix and calculates a higher-level representation. We then perform max pooling through a kernel of the input dimensions which in turn reduces the parameters required and this, in turn, reduces the training time. While doing the same we also introduce the model to the new CWWE features to test if the new derived features help to increase the accuracy of the classification. Model results at the end of training at this step are captured and represented in table7.

Test Data Set Accuracy			
CNN	0.80		
Confusion Matrix			
Model		Retweet	
		Yes	No
	Yes	19798	6223
	No	1603	9958
Precision = 0.9			
Recall = 0.76			

Table 7. Model results for the CNN model with CWWE

3.6 Build a CNN model with a pre-trained glove word embedding

In this step, we use the same network architecture from *3.5* above, but instead of using CWWE, we now use the pre-trained glove 100-dimension word embedding as input features and consider the results as a benchmark to compare against the results recorded from step *3.5*. If the results from step *3.5*, outperform the results of the current step then our original hypothesis is correct. The glove model used in this sub-section is pre-trained on four hundred thousand (400,000) words over a billion tokens. The glove has embedding vector sizes, including 50, 100, 200 and 300 dimensions. We chose the 100-dimensional version. We also want to see the model behavior in case the learned word weights do not get updated. We, therefore, set the trainable attribute for the model to be False. The results of this step are recorded and represented in table8 below.

Test Data Set Accuracy			
CNN+Glove	0.70		
Confusion Matrix			
Model		Retweet	
		Yes	No
	Yes	17004	9017
	No	2665	8896
Precision = 0.8			
Recall = 0.65			

Table 8.

3.7 Summarization of the results

A summarization of the results reveal that using the proposed framework of Custom Weighted Word Embedding (CWWE), if the word features extracted from the tweet are assigned custom weights basis the relative position of these words in the sentence, we record a significant improvement in the overall accuracy of CNN model from 53%

to 80%. We also see that precision and recall of the model with the proposed framework outperforms the preliminary model at 0.90 and 0.76 as compared to 0.80 and 0.40 respectively for the concerned class which is - will a tweet retweet (retweet >10). The CNN model with the proposed framework has also outperformed pre-trained established glove word-based CNN model experimentation with the same architecture with the latter's values as (accuracy: 0.70, precision:0.8, recall:0.65). Hence we can state that our initial hypothesis of a writer's deliberate choice of placing the more influential words (like nouns, adjectives) at the first half of the tweet puts more emphasis on the intent interpretation by the readers and helps to cascade the same at a faster rate on social media platform as compared to sentences/tweets where these words are used at the latter half is correct.

4 Discussion

The primary contribution of our research paper is CWWE with a combination of CNN in using the linguistic features. With the outcome of our experimentation set-up we confirm our hypothesis that degree and rate of any information diffusion on twitter (tweet getting retweeted) can be established as a function of how a user has designed the sentence with the constraint limitation of the size of the tweet (140 characters). Intentionally placing more impactful words related to the context of discussion can increase of potential retweet faster and hence for it is important to also understand the linguistic aspect of writing English literature. We were also able to bring down the training time of the model knowing that unstructured data will have higher dimension of features set as compared to other regular analysis. We have proved through our experimentation and research that the proposed CWWE can outperform any other established word-embedding framework.

5 Conclusion

Every tweet contains both micro and macro-level information about the information conveyed that can be extracted and then can be further used to represent a tweet. With this motivation we chose the semantics which represents the micro level information of a tweet and developed a framework of embedding matrix representing a writer's habit of placing the part of speech (Nouns and adjectives). We further used the rules of semantic information with the help of deep neural network framework to predict if a tweet will be retweeted on not. We proved that the diffusion of a particular content on Twitter can be driven by a sequence of nouns, adjectives, adverbs forming a sentence. We exhibited that the proposed approach in this paper is coherent with the intrinsic disposition of tweets to a common choice of words while constructing a sentence to express an opinion or sentiment. With this experimentation we have established that if a writer uses an adjective or a noun at the beginning of the sentence, there is highly likely a chance that the sentence/text/information is going to diffuse at a faster rate.

Motivated by the positive results of the current experiment, as next steps for future research work, we plan to extend the CWWE along with the other features like size of

the network structure, the user is part of and understand the relation with social influence on social media platform Twitter. The framework should be able to predict the exact social influence in terms of number of retweets any user will get while posting any tweet.

6 References

1. Sharples, M.: How We Write: Writing as Creative Design. Routledge, London (2016).
2. Maun, I. and Myhill, D.: Text as design, writers as designers. English in Education 39 (2), pp. 5–21, (2005).
3. Brown, G. and Yule, G.: Discourse Analysis. Cambridge University Press. Cambridge, (1983).
4. Leech, G.N. and Svartvik, J.: A Communicative Grammar of English. London (1975).
5. Danyluk, A.P., Bottou, L., Littman, M.L. (Eds.), ICML, ACM International Conference Proceeding Series, ACM, Vol. 382, pp.140, (2009).
6. Shi, Q., Petterson, J., Dror, G., Langford, J., Smola, A., Vishwanathan, S.: Hash kernels for structured data, Journal of Machine Learning Research 10, pp. 2615–2637, (2016).
7. Ganchev, K., Dredze, M.: Small statistical models by random feature mixing, Proceedings of the ACL-2008, Workshop on Mobile Language Processing, Association for Computational Linguistics, (2008).
8. Colmenares, C.A., Litvak, M., Mantrach, A.: HEADS: headline generation as sequence prediction using an abstract feature-rich space[C]. HLT-NAACL, pp. 133–142, (2015).
9. Goel, S., Anderson, A., Hofman, J., Watts, D.J.: The structural virality of online diffusion, Management Science, 62, pp.180–196, (2016).
10. Myers, S.A., Leskovec J.: Clash of the contagions: Cooperation and competition in information diffusion. In Proceedings of the IEEE 12th International Conference on Data Mining. Brussels: IEEE, 2012, pp. 539–548, (2012).
11. Romero, D.M., Meeder, B., Kleinberg, J.: Differences in the mechanics of information diffusion across topics: Idioms, political hashtags, and complex contagion on Twitter. In Proceedings of the 20th International Conference on World Wide Web. New York, NY: Association for Computing Machinery, pp. 695–704, (2011).
12. Stieglitz, S., Dang-Xuan, L.: Emotions and information diffusion in social media Sentiment of microblogs and sharing behavior, Journal of Management Information Systems, 29, 4, pp. 217–248, (2013).
13. Yoo, E., Rand, W., Eftekhar, M., Rabinovich, E.: Evaluating information diffusion speed and its determinants in social media networks during humanitarian crises, Journal of Operations Management, 45, pp. 123–133, (2016).
14. Aswani, R., Kar, A.K., Ilavarasan, P.V. & Dwivedi, Y.: Search Engine Marketing is not all gold: Insights from Twitter and SEOClerk. International Journal of Information Management, 38(1), pp. 107-116, (2018).
15. Grover, P., Kar, A.K., Dwivedi, Y.K., & Janssen, M.: The untold story of USA presidential elections in 2016 – Insights from Twitter Analytics. Lecture Notes in Computer Science, 105.95, pp. 339-350, (2017).
16. Mohan, R., & Kar, A.K.: Demonetization and its Impact on the Indian Economy – Insights from Social Media Analytics. Forthcoming in Lecture Notes in Computer Science, 105.95, pp. 363-374, (2017).

17. Starbird K., Palen L.: Pass it on? Retweeting in mass emergency. Proceedings of the 7th Int Conf of Inf Sys for Crisis Response and Management: 1–10, (2010).
18. Grover, P., Kar, A.K. & Ilavarasan, P.V.: Impact of Corporate Social Responsibility on Reputation – Insights from Tweets on Sustainable Development Goals by CEOs. International Journal of Information Management, (2019).
19. Grover, P., Kar, A.K., Dwivedi, Y.K. & Janssen, M.: Polarization and Acculturation in US Election 2016 outcomes – Can Twitter Analytics predict changes in Voting Preferences? Technology Forecasting and Social Change, (2018).
20. Grover, P., Kar, A.K., Davies, G.H.: "Technology enabled Health" – Insights from Twitter Analytics with a Socio-Technical Perspective. International Journal of Information Management, 43, pp. 1-13, (2018).
21. Grover, P., Kar, A.K., Janssen, M & Ilavarasan, P.V.: Perceived usefulness, ease of use and user acceptance of blockchain technology for digital transactions – insights from user-generated content on Twitter. Enterprise Information Systems, 13(6), pp. 1-30, (2019).
22. Aswani, R., Kar, A.K. & Ilavarasan, P.V.: Experience: Managing Misinformation in Social Media – Insights for policy makers from the Twitter Analytics. Journal of Data and Information Quality, (2019).
23. Yang, J., Counts, S., Predicting the Speed, Scale, and Range of Information Diffusion in Twitter. Proceedings of the Fourth International AAAI Conference on Weblogs and Social Media, (2010).
24. Xu, Z., Yang, Q.: Analyzing user retweet behavior on Twitter. In: IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), pp. 46-50. IEEE, (2016).
25. Nguyen, D.A., Tan, S., Ramanathan, R., Yan, X.: Analyzing information sharing strategies of users in online social networks. In: Proceedings of the IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, pp. 247–254. IEEE Press, (2016).
26. Lerman, K., Gilder, A., Dredze, M., Pereira, F.: Reading the markets: forecasting public opinion of political candidates by news analysis. In: Proceedings of the 22nd international conference on computational linguistics 1, pp. 473–480, (2008).
27. Riedhammer, K., Favre, B., Hakkani-Tür, D.: Long story short–global unsupervised models for key phrase based meeting summarization[J]. Speech Comm 52(10), pp. 801–815, (2010).
28. Zhang, Y., Shen, D., Wang, G., et al: DE convolutional paragraph representation learning[C]. Advances in Neural Information Processing Systems, pp. 4172–4182 (2017).
29. Li, J., Luong, M.T., Jurafsky, D.: A hierarchical neural auto encoder for paragraphs and documents[J]. arXiv preprint arXiv:1506.01057, (2015).
30. Rush, A.M., Chopra, S., Weston, J., A neural attention model for abstractive sentence summarization[J]. arXiv preprint arXiv:1509.00685, (2015).
31. Gu, J., Lu, Z., Li, H., et al: Incorporating copying mechanism in sequence-to-sequence learning[J]. arXivpreprint arXiv:1603.06393, (2016).
32. Zhong B., Xing X., Love P., Wang X., Luo H.: Convolutional neural network: Deep learning-based classification of building quality problems, pp. 46-57, Volume 40, April 2019, Advanced Engineering Informatics, ScienceDirect, Elsevier.
33. Yu M., Huang Q., Qin H., Scheele C., Yang C.: Deep learning for real-time social media text classification for situation awareness – using Hurricanes Sandy, Harvey, and Irma as case studies, Volume 12, 2019 - Issue 11: Social Sensing and Big Data Computing for Disaster Management, International Journal of Digital Earth.