

# **Stance Classification of Rumours**

A MAJOR II REPORT

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS  
FOR THE AWARD OF DEGREE OF

**MASTER OF TECHNOLOGY  
IN  
COMPUTER SCIENCE & ENGINEERING**

Submitted By  
**Meghna Upadhyay**  
**2K19/CSE/12**

Under the supervision of

**Dr. AKSHI KUMAR**  
**(Assistant Professor)**



**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**

DELHI TECHNOLOGICAL UNIVERSITY  
(Formerly Delhi College of Engineering)  
Bawana Road, Delhi-110042

**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Bawana Road, Delhi-10042

**DECLARATION**

I, Meghna Upadhyay, Roll No. 2K19/CSE/12 student of M.Tech (Computer Science & Engineering), hereby declare that the Project Dissertation titled “**Stance Classification of Rumours**” which is submitted by me to the Department of Computer Science & Engineering, Delhi Technological University, Delhi in partial fulfillment of the requirement for the award of the degree of Master of Technology, is original and not copied from any source without proper citation.



Place: DTU, Delhi

Meghna Upadhyay

2K19/CSE/12

**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Bawana Road, Delhi-10042

**CERTIFICATE**

I, hereby certify that the Project Dissertation titled “**Stance Classification of Rumours**” which is submitted by Meghna Upadhyay, Roll No. 2K19/CSE/12, Department of Computer Science & Engineering, Delhi Technological University, Delhi in partial fulfillment of the required for the award of the degree of Master of Technology, is a record of the project work carried out by the students under my supervision. To the best of my knowledge this work has not been submitted in part or full for any Degree or Diploma to this University or elsewhere.

Place: DTU. Delhi



Dr. Akshi Kumar

Supervisor

Assistant Professor

Department of Computer Science & Engineering

Delhi Technological University

## **ACKNOWLEDGEMENT**

I am profoundly thankful to Dr. Akshi Kumar Assistant Professor, Department of Computer Science Engineering, Delhi Technological University, Delhi for providing invaluable direction and being supportive round the clock. I will always appreciate her thorough support and encouragement she provided.

I am also thankful to all the research scholars of the web Research Group at Delhi Technological University, for their great suggestions and helpful discussions over the work of this project.

A rectangular box containing a handwritten signature in black ink. The signature is written in a cursive style and appears to read 'Meghna'.

Meghna Upadhyay  
2K19/CSE/12

## **ABSTRACT**

The accessibility and comfort of using social media have provided an optimal environment for people to expeditiously spread the information they have and sometimes without any knowledge of the authenticity of the information. Consequently, people inspect the stances reflected in the corresponding responses. To discover the certainty of rumour, stances are generally classified into 4 classes: support, deny, query and comment. The work presented brings forward a model for the Stance Classification of Rumours on a Twitter dataset which utilizes the newly introduced Capsule Network along with Multilayer Perceptron. The rule-based strategy is used to merge the output of both the networks in a way that utilizes the strength of the two networks.

The performance of the proposed model is compared with the state-of-the-art Turing model and two baseline CNN models, one with parallel layer architecture and one without parallel layer architecture. The hybrid of Capsule Network and Muti-Layer Perceptron model surpasses the Turing model with regard to the macro average F1-score indicating better results across different sets of classes. Furthermore, the proposed capsule network model also outperforms both the CNN model in terms of both accuracy and macro average F1-score.

## CONTENTS

<b>CHAPTER</b>	<b>CONTENTS</b>	<b>PAGE NO.</b>
<b>1.</b>	<b>INTRODUCTION</b>	<b>ix</b>
1.1	OVERVIEW	ix
1.2	MOTIVATION	xi
<b>2.</b>	<b>LITERATURE REVIEW</b>	<b>xii</b>
2.1	CAPSULE NETWORK	xiv
<b>3.</b>	<b>PROPOSED ARCHTECTURE</b>	<b>xvi</b>
<b>4.</b>	<b>IMPLEMENTATION</b>	<b>xx</b>
4.1	DATASET	xx
4.2	DATA INCONSISTENCY	xx
4.3	DATA PRE-PROCESSING	xxi
4.4	EXPERIMENT	xxii
<b>5.</b>	<b>RESULT</b>	<b>xxiv</b>
<b>6.</b>	<b>CONCLUSION</b>	<b>xxvii</b>
<b>7.</b>	<b>REFERENCES</b>	<b>xxviii</b>

## LIST OF FIGURES

<b>FIGURE</b>	<b>CONTENT</b>	<b>PAGE NO.</b>
1.	Architecture of the proposed model	xvi
2.	Layered Architecture of Capsule Network	xvii
3.	Architecture of Multi-Layer Perceptron Neural Network	xviii
4.	A different model architecture	xxii
5.	Baseline CNN model with parallel convolutional layers	xxv

## LIST OF TABLE

<b>TABLE</b>	<b>CONTENT</b>	<b>PAGE NO.</b>
1.	Examples of rumours along with their responses	x
2.	Distribution of tweets in the dataset into individual sets	xx
3.	Examples showing inconsistency in the dataset	xxi
4.	Comparison with Turing model: Accuracy, Macro Average F1-score and F1 score of individual classes.	xxiv
5.	Precision and Recall Value of each class	xxiv
6.	Confusion Matrix for the testing set.	xxv
7.	Comparison with CNN model: Accuracy and Macro Average F1-Score	xxvi



# **1. INTROUCTION**

## **1.1 Overview**

Misdirecting and deceiving individuals for personal or other reasons such as financial or political has become quite effortless with the advancement in social media which for the most part brings about hurting the people. Apart from sharing the information, social media has facilitated the creation of false information and manipulation of the information shared/exchanged by individuals. Rumour can be described as a fragment of data about a certain event/incident whose truth value is undetermined. It's circulating news without adequate information and proof to help it, in this way putting a question on its credibility. Therefore spreading rumours by the means of social media has become inevitable. Consequently, there is a growing requirement to analyze and work on rumours circulating expeditiously through social media.

Conventional newsgathering sources such as newspapers, news channels, or radio are still used by people and are reliable. But the boost in internet access speeds and smartphones has increased the use of social media and days of viewing social networking as a meaningless activity is a distant memory. In 2020, there are 3.96 billion people actively using social media, and this is an increase of 10.9% year-on-year from 3.48 billion in 2019. In case of a catastrophic event, the general public, media, and emergency responders make use of social networking platforms to search and circulate information related to that uncertain event. Correspondingly, this catastrophe-related information flow often results in the origination of rumours.

For the most part, it is not easy for an individual to precisely determine the authenticity of rumours by only going through the text contents of a post. Thus, people have started to focus on the stances given by other users in the responses to the particular post. Stance classification can be defined as the matter of recognizing the attitude captured by a user in a brief response text within a rumour microblog [1]. Commonly, stances reflected in the responses display support or deny with the source information. Although, for an uncertain circumstance people tend to express more than just support or deny. The RumorEval dataset [2] labels users' stance into four groups- support, deny, question, and

comment (SDQC) [3]. Table I. shows example of rumours from the RumorEval dataset along with the responses.

- *Support*: The response text precisely reflects support towards the source rumour.
- *Deny*: The response text precisely denies the source rumour
- *Query*: The response text doesn't reflect either support or deny but demands some supplementary information concerning the rumour.
- *Comment*: The response text is a comment without any precise contribution of data which may lead to the determination of the veracity of rumour.

TABLE I: EXAMPLE OF RUMOURS ALONG WITH THEIR RESPONSES

<i>Source Tweet 1</i> : Coup in #Russia? Good article by @[user1]. [link1] #RamzanKadyrov #Putin #putindead [link2]"
Response 1:#RamzanKadyrov- A Sunni Muslim Russian/Chechen Ultranationalist w/ ties to the \"United Russia\" party(Putinism); #Russia's FSB(current KGB).
Response 2:So if #RamzanKadyrov had #BorisNemtsov assassinated was it at the direction of Putin or is/was Putin at the direction of FSB/Kadyrov? Hmm
<i>Source Tweet 2</i> : Latest on #Germanwings crash: Pilots signaled 911 before dropping out of midair; airline CEO calls this a \"dark day.\"[link1]
Response 1:"@[user1] Signalled 911? Called 'Mayday' would be more appropriate, factual reporting...
Response 2: @[user1] signaled 911?
Response 3: @[user1] you might want to change the use of 911 in this context."

This work presents an approach for the advancement of the stance classification system which is critical to therefore deciding the truthfulness of rumours. The task of determining the stance of rumours can be viewed as a text classification problem where a rumour is considered as a plain text which needs to be classified into four target classes (SDQC). Consistently, text classification problems are accomplished by recurrent neural networks (RNNs) and convolutional neural networks (CNNs). The state-of-art utilizes LSTM based approach to determine the stance of the rumours. The work presented brings forward a new model for determining the stance of the rumour. The model comprises Capsule

Neural Network and Multilayer Perceptron (MLP) and the result achieved by this model is comparable to the state-of-the-art methods.

## **1.2 Motivation**

Rumour Stance Classification is an intricate problem. The basis of this task is to figure out the intention displayed by a person in his/her text content of the post. Additionally, with the hike in the internet and social media, a piece of information can be easily manipulated and disseminated. The creation of the false information can have some bad consequences. There are some studies which have discovered some strong association between heavy usages of social media and self-harm.

However, it is extensively difficult task to manually identify the true attitude of every user who has responded to a particular rumour. Therefore, it is vitally important to examine the information provided by the social media.

This provides the inspiration to look into the existing models tackling this essential issue and present a new approach for the improvement of the stance classification system.

## **2. LITERATURE REVIEW**

Researchers have since quite a while ago taken a look at the characteristics of rumours to comprehend their propagation style and to recognize them from different sorts of data that are frequently shared by individuals. Two main factors of rumour diffusion claimed by Allport and Postman [4] are that people are always looking for some significance in matters and, when encountered with some sort of vagueness, people attempt to seek meaning by saying some convincing tale [5]. This claim supports why rumours with time generally change by turning out to be more limited, sharper, and more reasonable. However, Rosnow [7] declared that there are four salient factors responsible for the propagation of rumour. It essentially should be related to the end result of the audience, required to surge the anxiety of the person, be acceptable to a certain degree, and be undetermined. On the contrary, Guerin and Miyazaki [6] proclaim that a rumour is a way of talking and interacting to build up a good relationship with people. Accordingly, it fulfils the purpose of creating and conserving relationships. Therefore, the task of rumour stance classification turns out to be significantly more relevant with the rampant increase of social media users.

The problem of rumour stance classification started to be extensively investigated in RumourEval task at SemEval 2017 [1]. Team Turing's paper describes a sequential approach to deal with the classification problem. After performing pre-processing steps which were removing non-alphabetic characters, converting them into lowercase, and tokenizing, they used word embeddings from Google News [8]. Along with this, additional features such as count of negative and swear words, punctuation marks, URLs, and pictures were also utilized to describe the rumour. The conversational structure of rumour was modeled by applying a LSTM-based sequential model and an accuracy of 0.784 on RumourEval test set was achieved [1] and thereby sets the state-of-art for the same.

One of the earliest works in the field of analyzing stances of rumours is done by Qazvinian in 2011 [9]. The work done was categorized into two parts: (1) Extracting rumours. (2) Belief Classification classifies the users who consider the rumour to be true versus those who do not believe the rumours to be true or question the rumour. Hence, it was

considered as a binary classification task (support/ deny) as the deny class and the questioning class was consolidated into one class [1]. The approach used was based on building different Bayes classifiers as high-level features and then learning a linear function of these classifications for the retrieval in the first task and classification in the second task [9].

Since 2015, there has been a sudden increase in enthusiasm for this task. Work done by Xiaomo Lie [10] proved to be better than Qazvinian. He implemented a rule-based method using an extensive list of positive, negative, and negation keywords with a set of language rules such as “negative words not preceded by a negation word imply that the user denies the event” [10]. Further, Hamidian and Diab [11] employed a Tweet Latent Vector (TLV) feature and created a 100-d vector representation of each tweet [11] to access the capability of carrying out the task of binary classification of stances. Thereafter, a three-way classification that separates the denying and questioning classes was carried out by Lukasik [5]. He developed an automated, supervised classifier based on Gaussian Processes that use multi-task learning to classify the stance expressed in each individual tweet as either supporting, denying, or questioning the rumour [5]. In 2016, the set of labels considered in previous work were expanded to incorporate a new label commenting and thus making it a four-way classification. This was done by Michal Lukasik [12]. Along with this, he introduced the Hawkes process for the stance classification and demonstrated the importance of using temporal information of tweets along with textual content [12].

Moving away from the two-way classification, Zubiaga and Kochkina [13] adopted the four-way stance classification and the approach evaluated the tree structure of Twitter conversation instead of determining the stance on the basis of a single tweet alone and gave a considerably better performance in terms of macro-averaged F1-score. Similar sequential approaches exploiting the conversational threads have been analyzed and concluded that LSTM provides the best performance [14]. Eventually, deep neural network-based work started to rise to tackle the stance classification task. Lozano, Lilja, Tjörnhammar†, and Karasalo [15] combined CNNs with automatic rule mining and manually written rules. Jing Ma, Gao, and Wong [16] used Gated Recurrent Unit (GRU) for representing hidden units rather than LSTM to improve efficiency. In SemEval 2019 competition, Fajcik, Burget, and Smrz [17] built the pre-trained end-to-end Bidirectional

Encoder Representations from Transformers (BERT) architecture and finished second in the competition. In 2020, A Stochastic Attention Convolutional Neural Network (SACNN) was introduced by Na Bai, Zhixiao Wang, and Fanrong Meng [18]. The model captures the different habits of the public for the rumour stance classification task [18] and the outcome of the model was higher than the state-of-art results. In addition to the deep learning neural networks, some scholars have utilized machine learning models for the stance classification task. Kaizhou Xuan and Rui Xia [3] implemented classical machine learning classifiers such as Logistic Regression (LR), Decision Tree (DT), Random Forest (RF), Linear Support Vector Machine (SVC), and Naive Bayes (NB). These classifiers were combined with 18 defined features including text features, user features, and propagation features. Logistic Regression classifier gave the best performance.

The work presented has an objective towards improving the stance classification system by taking advantage of Capsule Networks first introduced by Geoffrey E. Hinton [19].

## **2.1 Capsule Network**

In spite of the fact that various models built on deep neural networks have been offered for the problem of stance classification, there exists demand development in this project. This research is focused on using a Capsule Network to upgrade the accuracy of the Rumour Stance Classification system. In the introductory paper of Capsule Network, image classification was implemented on the MNSIT dataset and it revealed that the proposed model works better than Convolutional Neural Networks. A Convolutional Neural Network recognizes objects by detecting image features. Initial layers are responsible for detecting simple features such as edges and more complicated features such as eyes or nose (in case of face detection) are discovered by layers that are deep into the architecture. Therefore, the combination of all these features makes the ultimate prediction. It can be concluded that CNN does not make use of any spatial information and also the connection of layers is done by the use of the pooling function. *“The pooling function used in the Convolution Neural Networks is a big mistake and the fact that it works so well is a disaster”*- Geoffrey E. Hinton.

A Capsule Neural Network aims to perform inverse graphics. Given that, the method ventures to reverse engineer a technique that is responsible for the production of the

desired image [20]. Equivariance is known to be one of the crucial features of the Capsule Network. Its main objective is to preserve the comprehensive information of the location and pose of the object all over the network. For example, considering the event of marginally rotating the target, there is also a marginal change in the activation vectors. As defined by Geoffrey E. Hinton, a capsule is a group of neurons whose activity vector represents the instantiation parameters of a specific type of entity such as an object part [19]. The probability of the existence of an entity is defined by an activity vector's length. Also, the instantiation parameters are depicted by the orientation of the activity vector. The Capsule Network consists of many capsules at different levels. The particular level of capsules that are activated generates predictions by utilizing transformation matrices and are responsible for the initiation of the higher level capsule's parameters. A higher-level capsule gets activated as soon as multiple predictions agree. A repetitive procedure accountable for routing that decides the credit attribution among nodes of low and high level was also introduced [21].

Lately, Capsule Networks have gained substantial consideration and have been successful even in the field of NLP inclusive of text classification [22], sentimental analysis [23], fake news detection [20], and identifying toxic comments [24].

### 3. PROPOSED ARCHITECTURE

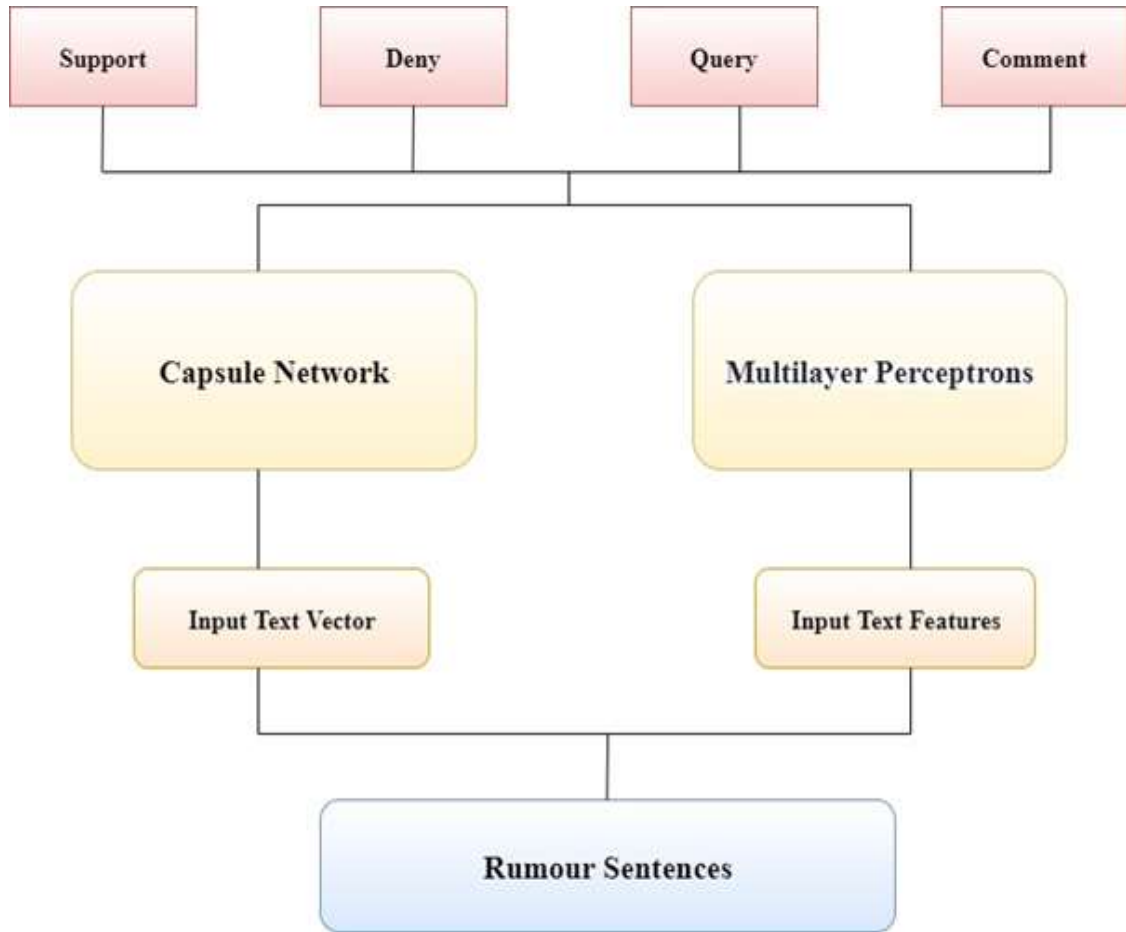


FIGURE 1: ARCHITECTURE OF THE PROPOSED MODEL

The model put forward by the research done for the Rumour Stance Classification problem is given by Figure 1. The model consists of two neural networks, Capsule Neural Network and Multilayer Perceptrons Neural Network. The Capsule Network takes sentences converted into vectors of a fixed length as input while the input to the Multilayer Perceptrons is the features extracted from the text sentences. The combined result from both the neural networks produces the final class label for the rumour sentences. The layered architecture of the Capsule Network is shown in the Figure 2.

#### ***Embedding Layer***

This is the first layer of the model. The layer is responsible for converting the integer encoded data into fixed-size vectors. The integer encoded data is an individual word represented by unique integers. The layer is loaded with arbitrary weights and it learns



embedding for all the words present in the vocabulary. The output of the layer is fixed size dense vector embedding for each word.

### ***Convolutional Layer***

This layer is responsible for obtaining various features from the input sentence that are placed at different positions in the sentence. Convolution filters are intended for the same.

### ***Primary Capsule Layer***

The aim of this first capsule layer in the architecture is to reform the end product of the convolution layer into a capsule vector representation. This maintains the semantic meaning of the words in the sentence. 8 dimension capsules are present in this layer with 32 channels. Moving a layer up in the network depends upon an algorithm called routing by agreement. Said algorithm activates the capsule present in the above layer when multiple capsules of the current layer vote for the specific capsule of the above layer.

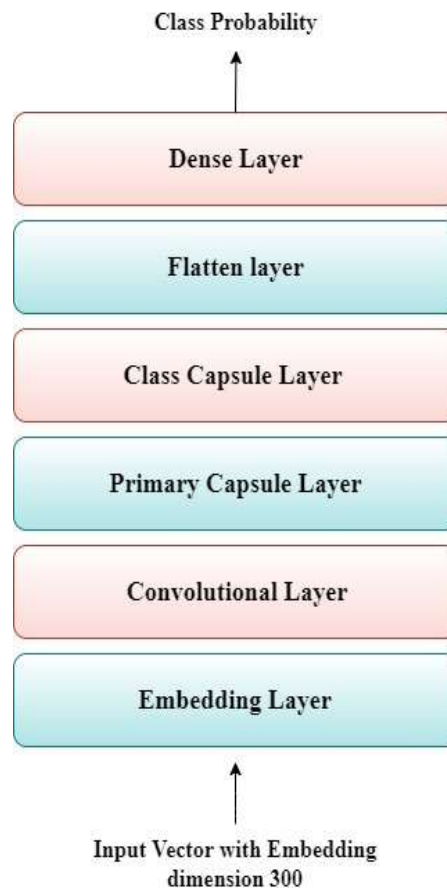


FIGURE 2: LAYERED ARCHITECTURE OF CAPSULE NETWORK

### ***D. Class Capsule Layer***

The intake of the layer is the output from the primary capsule layer. Every capsule present in the previously mentioned layer is in connection with the local region of the layer below.

The connection between this layer and the previous layer is provided by the mechanism of dynamic routing. The capsules are of dimension 16 in this layer and the number of routing used in the iterative dynamic routing algorithm is 3.

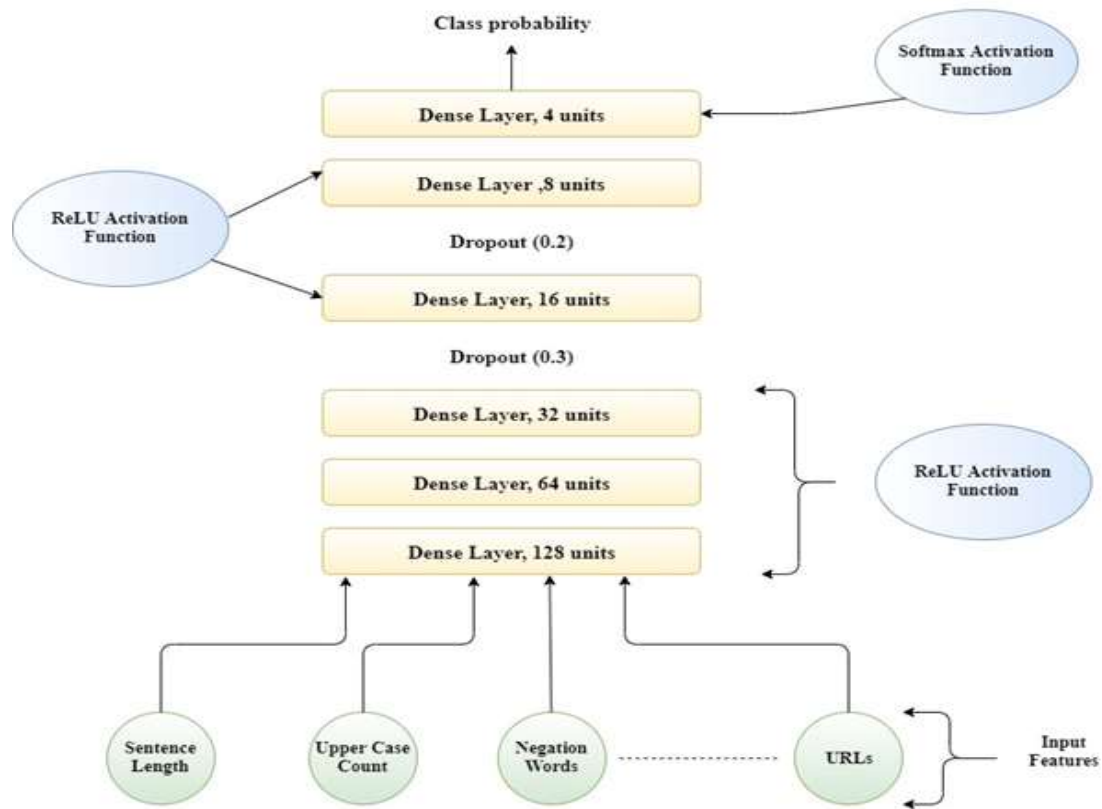


FIGURE 3: ARCHITECTURE OF MULTILAYER PERCEPTRONS NEURAL NETWORK

Figure 3 depicts the architecture of Multilayer Perceptrons Neural Network. The input to the network is a normalized text feature vector with a size equal to the number of features. Feature Extraction is an essential process and the context-based features used in the training of the network are explained below.

- *Sentence Length*: This counts the number of words present in the rumour post excluding all URLs and special characters.
- *Upper Case Count*: To express strong feelings, some people use all capital letters in the word, and therefore the count of such words is used as a feature.
- *Negation words*: It is the count of the presence of negation in the rumour post. A vocabulary is built containing all the common words that can be utilized to express deny and words that represent a contradiction. Some of the words in the vocabulary are ‘never’, ‘isn’t’, ‘barely’, ‘no’, ‘shouldn’t’.

- *Abusive words:* Most often people use dirty or abusive words to convey some strong feelings. A collection of such words is created which include words like ‘morons’, ‘fuck’, ‘stupid’, ‘bastard’, ‘bitch’. This collection also takes into account the slang representation of these words like ‘stfu’, ‘wtf’. Count of these words in the rumour post is used as a feature. These words appear more in deny or comment responses rather than query or support.
- *Question words:* A small collection of question words such as ‘what’, ‘where’, ‘why’ is used and the count of occurrence of these words is used as a feature.
- *Denial words:* The existence of words like ‘liar’, ‘unconfirmed’, ‘contradicting’ can be used to express the feeling of denial. These words are rarely present in support or query responses. Count of these words as a feature is used to distinguish deny class from other classes.
- *Belief words:* A group of positive words such as ‘agree’, ‘believe’, ‘sure’, ‘absolutely’ is generated and the count of such words in a rumour post is used as a feature. These words appear more frequently in support responses.
- *Sentiment Score:* This feature is constructed using VADER Sentiment Analysis tool. A compound score of rumour posts is used as a feature. The compound score ranges from -1 to +1. -1 demonstrates the most extreme negative score and +1 demonstrates the most extreme positive score. A chance of a low compound score is more in deny class.
- *Question Mark:* For this feature, the presence of a question mark in the rumour post is taken as ‘1’ and its absence as ‘0’. It plays an important role as the chances of the presence of a question mark in query response is higher than the other three classes.
- *Exclamation Mark:* People tend to use exclamation marks to express emotions. The presence of an exclamation mark is assigned ‘1’ and its absence ‘0’.
- *Presence of URLs:* The rumour post containing URL links has a high possibility to show support towards the rumour as it can provide extra information in the favour of the rumour post. Therefore, the presence of URL links is significant to the support responses.

## 4. IMPLEMENTATION

### 4.1 Dataset

To assess the efficiency of the suggested model for the Rumour Stance Classification task, the experiment was conducted on the RumorEval dataset formally came from real Twitter data. Twitter is a liberal wellspring of reports of breaking news. The dataset is composed of Tweets from eight separate incidents which include Ebola Essien, Ottawa shooting, Charlie Hebdo, Sydney Seige, Prince Toronto, Putin missing, Germanwings, and Ferguson riots [25].

Every rumour in each incident in the dataset is tagged as SDQC. The dataset incorporates a total of 5568 tagged rumours. Out of 5568 total tweets, 4238 tweets are used in the training set and 281 tweets are used in the development set. And for testing, 1049 tweets were used. The per-class distribution of tweets in the training, development, and testing is shown in table 2. It can be precisely observed that there is a big class imbalance in the dataset. More than 60% are tagged as ‘comment’. At the same time, ‘deny’ and ‘query’ combined make less than 16% of the total rumours. Because of this skewed nature of the dataset, class weights are used to compensate for the imbalance of the data. The calculated class weights are  $[S, D, Q, C] = [0.157, 0.396, 0.399, 0.048]$ .

TABLE II: DISTRIBUTION OF THE TWEETS IN THE DATASET INTO INDIVIDUAL SETS

	<b>S</b>	<b>D</b>	<b>Q</b>	<b>C</b>
<b>Training</b>	841	333	330	2734
<b>Development</b>	69	11	28	173
<b>Testing</b>	94	71	106	778
<b>Total</b>	1004	415	464	3685

### 4.2 Data Inconsistency

Upon careful examination of the dataset, it has been found that for some few tweets the labeling of the class is slightly inconsistent. Consider the tweets presented in the Table 3. The first and the second tweet convey somewhat similar meanings and are in response to the same source tweet but the first one is labeled as a comment and the second one is labeled as support. Similarly, there are few tweets that are labeled as query but there is no question asked in the tweet. To remove this data inconsistency, some of the tweets are relabeled after careful examination of tweet context and the source tweet. Training and

development set combined consists of 4519 tweets. Out of these 4519 tweets, 260 tweets are changed which means only 5.75% of the tweets are relabeled. No change is made in the testing data set.

TABLE III. EXAMPLES SHOWING INCONSISTENCY IN THE TWITTER DATASET

I just feel sick RT @[user]: At least 12 dead in Paris shooting. Updated story:[link]	Comment
Awful. RT @[user]: At least 12 dead in the Paris shooting. [link]	Support
I'm in London right now, feel free to make an appointment @[user] @[user] @[user]	Query
@[user] @[user] good thats a start	Query
@[user] @[user] @[user] @[user] How would you know the pilot was Muslim?	Deny

### **4.3 Data Pre-Processing**

Before feeding the data into the neural network, there is a need to apply some transformation and this process is known as data pre-processing. The structure of the data must be in a proper form to get a better outcome. The pre-processing steps performed on the dataset are as follow:

- All the sentences in the dataset are tokenized. The purpose of tokenization is to split the sentences into individual units or words.
- Using all the tokens, a vocabulary is created which consists of all the distinct words in the dataset.
- The input is a sequence of words. So, a tokenizer is built to modify them into an integer sequence. Hence, every word is mapped to a unique integer.
- The dataset contains sentences of variable length. Therefore, all the sentences were converted into fixed-length sentences. Sentences of shorter size as compared to the maximum size sentence are packed with trailing zeros. These sentences are then fed to Capsule Neural Network.
- For Multilayer Perceptron Neural Network, all the feature data is normalized by using MinMaxScaler which scales each feature to a given range with a default range of 0 to 1.

#### **4.4 Experiment**

The proposed architecture for Rumour Stance Classification is given in the Figure 1. It consists of a single convolution layer before the capsule layer. Other architectures which include two convolution layers, three convolution layers, and parallel convolution layers were also tested. One of the alternative approach is shown in Figure 4. It was observed that these architecture models were not able to perform any better than the proposed one. The results were comparable and more layers only added to the complexity of the architecture with no noticeable improvement in the result. Similarly, an attempt with parallel capsule layers architecture was also made but didn't show any observable improvement. Hence the architecture with the least complexity was preferred over others as they all generated comparable results.

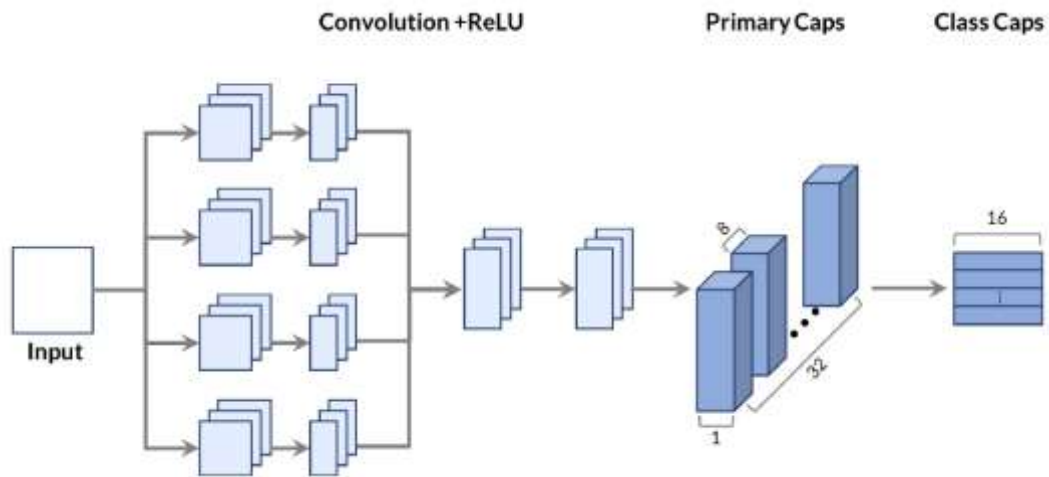


FIGURE 4: A DIFFERENT MODEL ARCHITECTURE

To apprehend the semantic meaning from the words in the sentences, word embeddings are used. It's a technique to represent separate words as a predefined vector. To initialize word embedding vectors, Keras Embedding Layer with a vector dimension of 300 is used. Further, these vectors are fed into the neural network. Consecutive to the embedding layer, there is an N-gram convolutional layer with N values as 9 and with a filter of size 256. The filter is set to move one unit at a time and therefore the stride is set as 1. Following all layers are capsule layers with capsule dimensions set as 8 in the first layer and 16 in the next layer. The first one is the primary layer followed by the class capsule layer. The connection of these capsule layers is achieved by using the transformation matrices and is multiplied by the coefficient generated by dynamic routing. The output generated by the class capsule layer is then flattened to a dense layer with softmax

activation which brings out the class probabilities. Similarly, in the Multilayer Perceptron network, extracted input features go through a number of hidden dense layers and produce the class probabilities. To get the final class label, the per-class performance of both networks is carefully considered and merged in a way that brings out the best from both networks. The Multilayer Perceptron network provided better results for the query class in comparison to the capsule layer network. On the other hand, both the networks were not able to detect the deny class as well as the other classes. For the improvement of the same, few rules are developed after examining the training and development dataset and these were used to merge the result from the neural networks. It has been observed that phrases like ‘stop spreading these lie’, ‘I believe otherwise’, ‘there is no truth’, are commonly used in tweets which are labelled as deny. So the occurrence of phrases with similar meaning implies that the probability of ‘deny’ class is greater than the other classes. Eventually, the class with maximum probability from both the networks is assigned as the final class label.

## 5. RESULT

The achievement of the proposed model on the testing dataset is displayed in Table 4. As the dataset is skewed, the average macro F1-score and per class F1-score are presented in the result table. Macro average F1-score shows how the model has performed overall across various classes present within the dataset. The model discussed in this paper has slightly less accuracy than Turing which has set the state of the art for the Rumour Stance Classification problem but the average macro F1-score of the proposed model is more than the Turing. The model also achieves better results in predicting the support, deny, and query classes. Table 5 shows the per class Precision and Recall values.

TABLE IV: COMPARISON WITH TURING MODEL: ACCURACY, MACRO AVERAGE F1-SCORE, AND F1-SCORE OF INDIVIDUAL CLASSES

	Accuracy	Average Macro F1-Score	S	D	Q	C
<b>Turing</b>	78.4	0.43	0.40	0.00	0.46	0.87
<b>Proposed Model</b>	77.6	<b>0.55</b>	<b>0.49</b>	<b>0.27</b>	<b>0.57</b>	0.86

The proposed model anticipates the comment class as the majority class. Table 6 demonstrates the confusion matrix for the model. Although, the performance of the model in picking out the deny class is not as good as compared to the other classes. Most of the deny class data are wrongly predicted and are labelled as comment. This can be improved with the addition of more data labelled as deny. However, the majority of ‘query’ class data got the right prediction label and the Multilayer Perceptron Network plays a crucial role in this.

TABLE V: PRECISION AND RECALL VALUES OF EACH CLASS

	S	D	Q	C
<b>Precision</b>	0.52	1.00	0.57	0.83
<b>Recall</b>	0.46	0.15	0.58	0.90



TABLE VI: CONFUSION MATRIX FOR TESTING SET

	<b>S</b>	<b>C</b>	<b>Q</b>	<b>D</b>
<b>S</b>	43	48	3	0
<b>C</b>	37	700	41	0
<b>Q</b>	1	44	61	0
<b>D</b>	2	56	2	11

CNN baseline models are tested on the same dataset to demonstrate the comparison between the Capsule Network model and CNN based models. The first CNN model is a parallel architecture which includes three parallel convolutional layers with max pooling followed by a fully connected dense layer. The architecture of this model is shown in Figure 5.

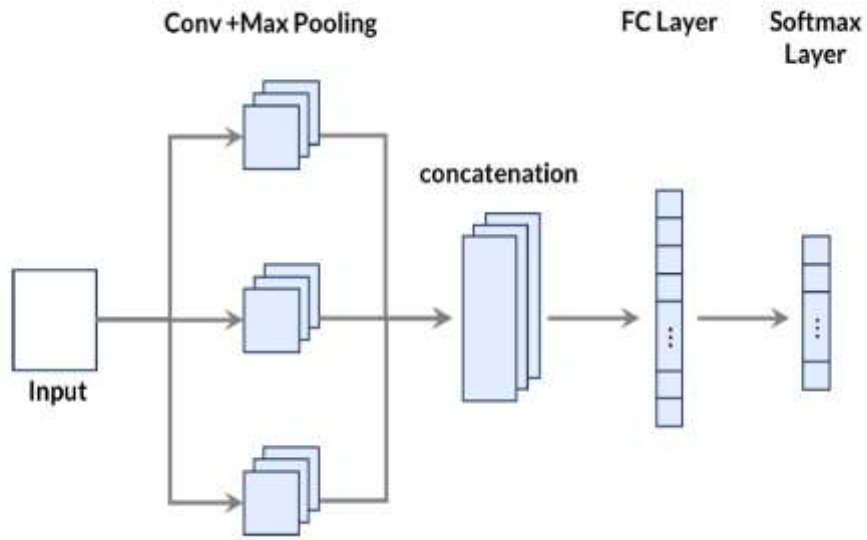


FIGURE 5: BASELINE CNN MODEL WITH PARALLEL CONVOLUTIONAL LAYERS

An alternative model without any parallel layers is also considered for the comparison. This model architecture contains only one convolution layer with max pooling followed by a fully connected dense layer with ReLU activation function and a softmax layer for the final class prediction values. Both the CNN models are used along with Multi-Layer Perceptron Neural Network with the same feature set as used in the proposed model. Additionally, the same rules are applied to the hybrid of CNN and Multi-Layer Perceptron

to merge the output of both the neural network. Table 7 demonstrates the result achieved by the CNN models and compares it with the proposed model.

TABLE VII: COMPARISON WITH CNN MODELS: ACCURACY AND MACRO AVERAGE F1-SCORE

	<b>Accuracy</b>	<b>Macro Average F1-Score</b>
CNN without parallel layers	68.4	0.26
CNN with parallel layers	50.5	0.21
Capsule Network	72.5	0.27
CNN without parallel layers with MLP	76.9	0.51
CNN with Parallel Layers with MLP	73.8	0.51
<b>Proposed Model</b>	<b>77.6</b>	<b>0.55</b>

CNN model without parallel layers outperforms the CNN model with parallel layers in terms of both accuracy and Macro-Average F1-Score. A parallel layer architecture makes it complex to implement and does not provide better result. Therefore, making a simple CNN architecture a better choice. On the other hand, Capsule Network performance surpasses both the CNN models. Additionally, a hybrid of Capsule Network and Multi-Layer Perceptron provides better accuracy and F1-score than a hybrid of CNN and Multi-Layer Perceptron given the same set of features.

## **6. CONCLUSION**

The work done presents a technique to address the matter of Rumour Stance Classification by exploiting Capsule Network along with Multilayer Perceptron Network. For the training of Multilayer Perceptron Network, we studied the impact of various features extracted from the dataset and some of the selected features proved to be effective, chiefly for the ‘query’ class. The final class label is assigned after combining the class probabilities of both the networks in a way that utilizes each network’s strength. To improve the performance of the ‘deny’ class, few rules were selected after the careful examination of the training and development dataset and applied before merging the output from the two networks.

For future work, more data can be added for the weaker classes such as the ‘deny’ class and some new features can be explored that focus on distinguishing the class better.

## REFERENCES

- [1] Elena Kochkina, Maria Liakata, Isabelle Augenstein,, Turing at SemEval-2017 Task 8: Sequential Approach to Rumour Stance Classification with Branch-LSTM, Association for Computational Linguistics, Proceedings of the 11<sup>th</sup> International Workshop on Semantic Evaluation (2017) 475-480.
- [2] A. Zubiaga, M. Liakata, R. Procter, G. W. S. Hoi, and P. Tolmie, Analysing how people orient to and spread rumours in social media by looking at conversational threads, arXiv:1511.07487v3 [cs.SI] (2016)
- [3] Kaizhou Xuan, Rui Xia, Rumour Stance Classification via Machine Learning with Text, User and Propagation Features, International Conference on Data Mining Workshops (ICDMW) (2019)
- [4] G. W. Allport and L. Postman. 1947, The psychology of rumor, Journal of Clinical Psychology (1947).
- [5] Michal Lukasik, Kalina Bontcheva, Trevor Cohn, Arkaitz Zubiaga, Maria Liakata, Rob Procter, Using Gaussian Processes for Rumour Stance Classification in Social Media, arXiv: 1609.01962v1 [cs.CL] (2016).
- [6] Bernard Guerin and Yoshihiko Miyazaki, Analyzing Rumors, Gossip, and Urban Legends Through Their Conversational Properties, The Psychological Record (v56) (2006).
- [7] Ralph L. Rosnow, The psychology of rumor, American Psychologist, Vol 46(5) (1991), 484–496.
- [8] T. Mikolov, K. Chen, G. Corrado, and J. Dean, Efficient estimation of word representations in vector space, arXiv preprint arXiv: 1301.3781, 2013
- [9] Vahed Qazvinian, Emily Rosengren, Dragomir R. Radev, Qiaozhu Mei, Rumor has it: Identifying Misinformation in Microblogs, Association for Computational Linguistics, Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing (D11-1147) (2011), 1589-1599.
- [10] Xiaomo Lie, Armineh Nourbakhsh, Quanzhi Li, Rui Fang, Real-time Rumor Debunking on Twitter, The 24<sup>th</sup> ACM International Conference on Information and Knowledge Management (CIKM) (2015).

- [11] Sardar Hamidian and Mona T Diab, Rumor Identification and Belief Investigation on Twitter, Association for Computational Linguistics, Proceedings of the 7<sup>th</sup> Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis (W16-0403) (2016), 3-8.
- [12] Michal Lukasik, P.K. Srijith, Duy Vu, Hawkes Processes for Continuous Time Sequence Classification: an Application to Rumour Stance Classification in Twitter, Association for Computational Linguistics, Proceedings of the 54<sup>th</sup> Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Pages) (2016), 393-398
- [13] Arkaitz Zubiaga, Elena Kochkina, Mariaw Liakata, Rob Procter, Michal Lukasik, Stance Classification in Rumours as a Sequential Task Exploiting the Tree Structure of Social Media Conversations, The COLING 2016 Organizing Committee, Proceedings of COLING 2016, the 26<sup>th</sup> International Conference on Computational Linguistics: Technical Papers (C16-1230) (2016), 2438-2448.
- [14] Zubiaga, Kochkina, Lukasik, Discourse-aware rumour stance classification in social media using sequential classifiers, arXiv: 1712.02223v1 [cs.CL] (2017).
- [15] Marianela Garcia Lozano, Hanna Lilja, Edward and Maja Karasalo, Mama Edha at SemEval-2017 Task8: Stance classification with CNN and Rules, Association for Computational Linguistics, Proceedings of the 11<sup>th</sup> International Workshop on Semantic Evaluation (SemEval-2017) (2017), 481-485.
- [16] Jing Ma, Wei Gao, Kam-Fai Wong, Detect Rumor Stance Jointly by Neural Network Multi-task Learning, WWW '18: Companion Proceedings of the The Web Conference (2018), 585-593.
- [17] Martin Fajcik, Lukas Burget, Pavel Smrz, BUT-FIT at SemEval-2019 Task 7: Determing the Rumour Stance with Pre-Trained Deep Bidirectional Transformers, Association for Computational Linguistics, Proceedings of the 13<sup>th</sup> International Workshop on Semantic Evaluation (S19-2192) (2019), 1097-1104.
- [18] Na Bai, Zhixiao Wang, Farrong Meng, A Stochastic Attention CNN Model for Rumor Stance Classification, IEEE Access (2020).
- [19] Sara sabour, Nicholas Frost, Geoffrey E. Hinton, Dynamic Routing Between Capsules, 31<sup>st</sup> Conference on Neural Information Processing Systems (NIPS 2017), (2017).

- [20] Mohammad Hadi Goldani, Saeedeh Momtazi, Reza Safabakhsh, Detecting Fake News with Capsule Neural Networks, arXiv: 2002:01030v1 [cs.CL] (2020).
- [21] Wei Zhao, Jianbo Ye, Min Yang, Zeyang Lei, Soufei Zhang, Zhou Zhao, Investigating Capsule Networks with Dynamic Routing for Text Classification, Association for Computational Linguistics, Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (D18-1350) (2018), 3110-3119.
- [22] Kim, J., Jang, S., Park, E., & Choi, S., Text classification using capsules, Neurocomputing, (Volume 376) (2020), 214-221.
- [23] Jingjing Gong, Xipeng Qiu, Shaojing Wang, Xuanjing Huang, Information Aggregation via Dynamic Routing for Sequence Encoding, Association for Computational Linguistics, Proceedings of the 27<sup>th</sup> International Conference on Computational Linguistics (C18-1232) (2018), 2742-2752.
- [24] Saurabh Srivastava, Prerna Khurana, Vartika Tewari, Identifying aggression and toxicity in comments using capsule network, Association for Computational Linguistics, Proceedings of the First Workshop on Trolling, Aggression and Cyberbullying (TRAC-2018) (2018), 98-105.
- [25] L. Derczynski, K. Bontcheva, M. Liakata, R. Procter, G. W. S. Hoi, and A. Zubiaga, Semeval-2017 task 8: Rumoureal: Determining rumour veracity and support for rumours, Association for Computer Linguistics, Proceedings of the 11<sup>th</sup> International Workshop on Semantic Evaluation (SemEval-2017) (S17-2006) (2017), 69-76.