

Rumor Mill: Tracking Viral Rumors through Textual Analysis

Shaik Rahima, Rupa Pujitha, Shaik Subhani

Abstract:

Nowadays, Rumors are Controversial and fact checkable statement. Rumors are powerful, pervasive and persistent force that affects people and group. It is a statement that is unverified at the time of circulation and either unverified or verified to be false after some time. Those unverified statements spread widely, have a big influence on how we see the world. They can shape what we believe and how we act. This abstract looks at rumors in depth, highlighting how controversial they are and why fact-checking is so important in figuring out what's true and what's not. By breaking down how they spread and stick around, we see how important it is to question information and check the facts. Recognizing rumors as powerful but also something to be careful about stresses the need for critical thinking and relying on evidence when we deal with information.

Keywords: Text classification , Naïve Bayes Classifier , Random Forest, Decision Tree classifier ,Dataset.

1 Introduction

What is a rumor?

A rumor is a piece of information or a story that is circulated widely, often informally and through word of mouth, but has not been verified as true. Rumors can range from harmless gossip to more serious claims, and they have the potential to influence people's beliefs, perceptions, and behaviour, even if they are later proven to be false. Rumors can spread rapidly, especially in today's digital age with the use of social media platforms, and they can have significant social, cultural, and political consequences.

How it can be spread?

In today's digital age, rumors can spread rapidly and widely through the internet, primarily through social media platforms, messaging apps, and other online forums. Social Media Platforms like Facebook, Twitter, Instagram, and TikTok provide instant communication and sharing capabilities, making it easy for rumors to spread quickly. Users can repost or retweet rumors without verifying their accuracy, leading to their amplification. Online Forums and Discussion Boards: Websites like Reddit, Quora, and various forums host discussions on a wide range of topics. Rumors can gain traction as users share them and engage in discussions, sometimes without verifying the information. Misinformation Campaigns: Individuals or groups may intentionally spread rumors or false information online to manipulate public opinion, sow discord, or achieve specific political or social goals. These misinformation campaigns can be coordinated and strategically targeted to reach a wide audience. Algorithmic Amplification: Social media algorithms are designed to prioritize content that generates engagement, such as likes, shares, and comments. This can inadvertently amplify

rumors and misinformation, as sensational or controversial content often garners more attention.

The paper compares and contrasts the approaches for identifying rumors using textual analysis by applying models to the acquired dataset. We remove stopwords from the text and preprocess it to store all the keywords into an array. After that using Countvectorizer we can create a feature matrix, later split the data into train and test sets. After that we get the score of training and testing. MultinomialNB from navie_bayes helps to fit the large amount of data and assign numerical value to different texts. By using matplotlib package we can visualize the heat map between true and predicted labels in confusion matrix. And, to check the most accurate model we tested the data with random forest, Decision tree.

This section discusses the research done in the field of text classification. The various authors have used machine learning and deep learning techniques such as

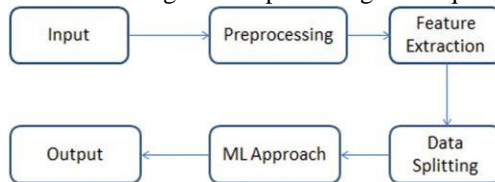


Fig. 1. Proposed Model for identifying Emotions

Decision trees are a supervised learning algorithm used for classification and regression tasks. They work by recursively partitioning the input space into regions, based on the features of the data. In text classification, decision trees can be used to create a tree structure where each node represents a feature or attribute of the text data, and each branch represents a possible value or decision based on that feature. For text classification, decision trees can consider various features of the text data such as word frequencies, presence or absence of certain words, syntactic structures, etc., to make decisions about the class labels. Decision trees are relatively interpretable, allowing users to understand the decision-making process behind the classification.

Random forest is an ensemble learning method that operates by constructing a multitude of decision trees during training and outputting the mode of the classes (classification) or the mean prediction (regression) of the individual trees.

Each decision tree in the random forest is trained on a random subset of the training data and a random subset of the features. In text classification, random forest can be applied similarly to decision trees, but it typically offers better performance due to its ability to reduce overfitting and improve generalization by combining multiple decision trees. Random forest can handle large feature spaces effectively, which is common in text classification tasks where each word or n-gram can be considered a feature. Random forest is less interpretable compared to a single decision tree, but it often provides better accuracy and robustness in classification tasks.

Proposed Work

This section represents the Proposed Model, in which our App takes the input from the user later it will be converted to lower case. And proceeded as the input or variable to the class which exhibits processing tasks. Later, the text which is processed will be checked with the array of stop words and remove unwanted words which will save time for model to search for text. Finally, this text will be converted to its base form and appended into the list.

The PHEME Dataset for Rumor Detection Database contains total of 61471 rows of data in which the columns of the data are as follows

Table 1. Details of the PHEME Dataset for Rumor Detection

Dataset	text	Is_rumor	User handle	Topic
PHEME Dataset for Rumor Detection	61471	2	29160	3

1.1 Data Pre-processing

Initially, data must be collected from diverse sources like social media, news platforms, and forums, with accompanying metadata such as timestamps and sources, along with labels denoting rumor presence. Subsequently, the text undergoes meticulous cleaning, which involves the removal of irrelevant elements like HTML tags, URLs, and special characters, followed by converting text to lowercase for uniformity. Tokenization then splits the text into individual words or characters, while eliminating stopwords to minimize noise. Normalization techniques like stemming or lemmatization are applied to reduce words to their base forms, thereby streamlining the feature space. Handling outliers, such as excessively long or short texts, ensures data integrity. Next, relevant features are extracted using methods like Bag-of-Words, TF-IDF, or embeddings to capture semantic nuances effectively. The preprocessed data is then partitioned into training, validation, and test sets for model development and evaluation. Addressing class imbalance, if present, is crucial, possibly through techniques like oversampling or weighted loss functions. Data augmentation may be employed to diversify the dataset further. Finally, categorical labels are encoded numerically to facilitate model training. Through these systematic preprocessing steps, the text data becomes refined, standardized, and suitably formatted for subsequent analysis and classification tasks within the project's scope.

1.2 Feature Selection and Extraction

1.3 In addition to linguistic analysis and machine learning techniques like decision trees and random forests, incorporating text, rumor status, topic, and user handle as features can further enhance the accuracy and effectiveness of rumor tracking systems. The textual content itself serves as a primary source of information for feature extraction. Techniques such as natural language processing (NLP) can be employed to preprocess and analyze the text data, extracting relevant features such as word frequencies, grammatical patterns, named entities, and sentiment scores. Alongside text, the binary classification of whether a piece of information is a rumor or genuine news can be a crucial feature for training machine learning models. This label serves as the ground truth for supervised learning algorithms, allowing them to learn to distinguish between rumors and non-rumors based on other features present in the data. Furthermore, the topic or subject matter of the text can also be an informative feature for rumor tracking. Certain topics or domains may be more susceptible to the spread of rumors than others, and understanding the context in which the information is shared can help in discerning its credibility. Topic modeling techniques, such as Latent Dirichlet Allocation (LDA), can be used to extract topics from the text data, which can then be represented as features for further analysis. Additionally, the identity of the user or source sharing the information can provide valuable contextual information for rumor tracking. Incorporating user handles as features can help in assessing the reliability of the information and identifying potential sources of misinformation. By integrating text, rumor status, topic, and user handle as features in the feature extraction process, the rumor tracking system can leverage a diverse range of information sources to make more accurate predictions about the credibility of the shared information. This holistic approach allows for a comprehensive analysis of the text data and its surrounding context, ultimately improving the effectiveness of rumor detection and mitigation efforts in online environments.

1.4 Classification

The Data is split into training and testing in which for testing 25% of the data is used and the remaining 75% data is trained.

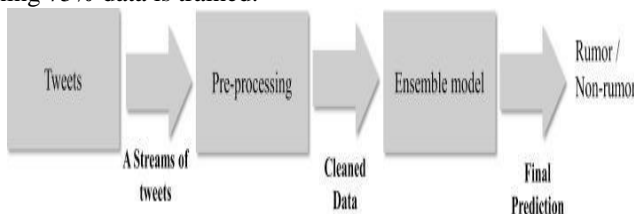


Fig. 2. Classification Architecture

Classification is done.

Rumor Mill : Tracking Viral Rumors through Textual Analysis
Table 2. Performance evaluation on Testing Data

Algorithm	Accuracy
Naivebayes	91 %
Randomforest Classifier	87 %

2 Results and Discussion

The experimentation evaluations Provide case studies or examples of rumors tracked by the system, highlighting the characteristics of the text data and the reasoning behind the classification decisions. Discussed many challenges encountered in identifying certain types of rumors or misinformation. Reflect on the ethical considerations surrounding rumor tracking and misinformation mitigation. Discussed the potential impact of the project on societal trust, privacy, and freedom of expression.

Conclusion and Future Work

In this paper, the proposed approach mainly focuses on Enhancements in machine learning algorithms and text analysis techniques could lead to the development of real-time rumor detection systems. These systems could monitor social media platforms, news websites, and online forums continuously, providing early detection and mitigation of viral rumors. With Naïve Bayes we get accuracy of 91% in training and 91% in testing. In future or present this rumor tracking will prevent so many modern combats between countries, groups and religions as well.

The images or other third party material in this chapter are included in the chapter’s Creative Commons license, unless indicated otherwise in a credit line to the material.