The entire team worked together to realise this project, helping out one another wherever doubts and opportunities arose.

Contributions	
Name	Contributions
Anmol Jha	Data Annotation, Abstract, Introduction, Dataset Description and Experimental Design
Anoushka Dixit	Web Scraping URLs, Dataset Preparation & Preprocessing, Methodology and Experimental Design in documentation
Mehak Aggarwal	Data Annotation, Related Work, Future work and Conclusion, Machine Learning Models Description
Parul Mann	Data Annotation, Experimental Design, Result and Observations
Saumya	Data Annotation, Abstract, Introduction, ML and DL Model description
Sohali Baisla	Data Annotation, Related Work, Future Work and Conclusion, DL Model preparation and description
Vidhi Bansal	Web Scraping of fact check articles, Dataset Preparation, Machine learning and Deep learning model preparation

Hindi Fake News Dataset and Analysis
ACM Internship 2021

Anmol Jha, Anoushka Dixit, Mehak Aggarwal
Parul Mann, Saumya, Sohali Baisla, Vidhi Bansal

Indira Gandhi Delhi Technical University For Women, Kashmere Gate, Delhi, India

October 20, 2021

Abstract

The last decade has experienced an amazing and drastic advancement in technology, especially in the field of communication. However, with the advancement in communication technology and the ease of spreading information, there has also been a rise in the spread of fake news. Fake news is defined as "news articles that are intentionally and verifiably false"[6]. Social media has been playing a very important role in the spread of fake news especially during the current situation of the Covid pandemic. Fake news has no language barriers and deciding whether the information present before us is reliable or not is not an easy task. The problem of fake news is serious, and many communities and research teams are working towards finding an effective solution to it. For building any AI/ML model, data annotation is important. The task of data annotation is very tedious yet crucial for an AI/ML/DL model as it plays a major role in determining the accuracy of the model. Most of the existing models for annotation of data work on English data. We have tried to extend the automation of data annotation to Hindi. This research concentrates on automating the task of identifying the links that lead to fake news and annotating it accordingly using a machine learning and a deep learning model. A manually annotated Hindi Fake News links dataset was used for training the models which works on algorithms such as Gaussian Naive Bayes, k-Nearest Neighbours, Support Vector Machine, LSTM and others. On a benchmark dataset consisting of 932 fake and 1274 not-fake news links, the model has been successful in identifying most of the fake news links and we achieved an accuracy of 82.35% for the Random Forest model implemented on 10% test data and an accuracy of 60.42% for the LSTM model implemented on 30% test data.

KEYWORDS: Data Annotation, Fake News Detection, Machine Learning, Deep Learning, Fact Check, Social Media

1 Introduction

The past decade has seen unprecedented growth in the field of technology. There has been a revolution in several industries and their old manual ways of functioning have been replaced by new modern technology. While some industries have been negatively affected by technological advancement, several industries have flourished and have achieved greater heights. The communication industry is one of the many industries which has flourished and improved with technological advancement. Mass communication has become easier with the developments in technology. Now it is simple to spread the news to a large population at once within no time. However, with each good change comes a bad one. While it has become easy to spread news anywhere, anytime, to anyone it has also become easier to spread fake news. Fake news has become a very serious problem since the digitization of mass communication media [25]. Our world has been gripped by the contagion of fake news. No country is spared, right from the US to rising economies like India [19]. Over the past few years, we can see a sharp rise in the number of fake news being reported. Genuine information is increasingly getting buried in an abundance of false information, creating estrangements between various communities, castes, and religions. It has also led to polarization and violent crimes in many countries like India[33]. Fake news can manipulate people and change their perceptions. People form their opinions

based on what they read, what they hear, and what they see in the news[22]. Exposure to content that is misleading can affect the opinions and mindsets of a lot of people. Overlong exposure to such content can even lead to several other bigger problems.

Social media has been playing a very key role in the spread of fake news [20]. It has been acting as the main key platform where fake news starts spreading. It has been observed that most of the fake news starts emerging from social media platforms [21]. Here news travels fast and reaches a large audience within no time. Since most people are active on social media and depend on them for information, it is easy to start a thread of fake news on any such social media platform like Facebook or Twitter and reach a large audience instantly. In fact, in a report by the Jumpshot Tech Blog[7], it was found that Facebook referrals accounted for 50% of the total traffic to fake news sites and 20% total traffic to reputable websites. Also, once any fake news is spread on a social media platform it is difficult to completely remove it from there because by the time it is put down, a lot of people have already seen the fake news and re-shared it through their account. Hence a quite large thread of fake news is formed involving many users. It is difficult and at times nearly impossible to reach the end of such chains of fake news since they have been re-shared so many times, hence much fake news remains unmarked on such platforms. However, not all the people on such platforms start the chain of fake news. Most of the people are only found re-sharing the fake news. It is only a few selected groups of people who start the chain. Several small-scale analyses have observed that there are often groups of users that heavily publicize fake news, particularly just after its publication [4] [27]. A lot of people or groups of people or organizations are working towards spreading fake news. They even spend money on spreading fake news and especially employ people to spread fake news and rumors on social media. The New York Times cited examples of people profiting from publishing fake stories online. It is observed that the more provoking the news is, the greater is the response, and the larger is the profit [24]. If fake news continues to spread like this people will lose their trust in social media. Hence, a lot of social media platforms are working towards detecting and deleting fake news and banning users who spread such news. However, regardless of a couple of late drives by some online media suppliers like Facebook, there is no precise system of detecting fake news [17].

To deal with the problem of fake news a lot of fact-checking websites have emerged over the years. These fact-checking websites look into the details of suspicious fake news, look into their sources, check the facts, and verify the claims made in the news to check whether the news is fake or not. However, all the fake news reported is not entirely false. At times false statements are mixed with true ones making the differentiation difficult. The use of facts makes the news more believable and it becomes difficult to classify them as fake news [14]. Usually, there are patterns in fake news that these fact-checkers observe to check whether the news is fake or not, like the structure of the URL, the credibility of the media source, to the profile of the journalist who authored it [34]. One of the most important patterns is the keywords in the URLs. While most of the URLs contain links to social media platforms like Facebook and Twitter, they also contain certain keywords which help in deciding whether the news is fake or not [26]. However, the people spreading fake news are aware of these detection mechanisms and have come up with new ways to disguise their fake news links. Usually, they will omit the keywords or misspell them so that they go undetected when the link is checked. However, fact-checkers are not the best and reasonable way to curb the problem of fake news. This is because fact checks report about news only on specific domains of interest like politics. Also, a lot of human work and expertise is involved in verifying facts for fact checks. It is even difficult to obtain datasets and provide a degree of generalization in case of fact checks.

Fake news has no linguistic barriers and it can be spread even in other languages like Hindi. The Hindi language is a member of the Indo-Aryan group within the Indo-Iranian branch of the Indo-European language family. It is also the preferred official language of India. India has a very large population. It is the second-most populous country in the world. Almost 4.43% of people in the entire world speak Hindi [13]. Most of the residents in India speak and understand Hindi. The National Crime Records Bureau (NCRB) has provided statistics on fake news for the first time in 2017 due to the increase in instances of fake news.[1] The statistics presented by the NCRB designate 'fake news' as a crime. The problem is aggravated by India's huge growth of internet connectivity [1].

Undoubtedly social media has become a platform for making connections and sharing information quickly but according to the BBC. It might be assumed that consumers are accessing these social media platforms and falling prey to false news because of the low data tariff. While most of the people in India are literate (74.04%) [2], many of them are still uneducated. Fake news on social media platforms reaches such users also and impacts them greatly. Being incapable of making wise decisions due to lack of education these people become easy targets to fake news. They take the fake news seriously, spread it vigorously in their community, and shape their opinions accordingly. This leads to poor decisions which make them suffer in the future. What occurs is that someone posts a fabricated incident on WhatsApp, which is then forwarded to a large number of individuals without anyone evaluating the news for validity. This aids the spread of a few people's "propaganda." The 'fake news' problem is only likely to get worse and more problematic in the coming years, according to CISCO's VNI predictions for India.

In this project, we aim to automate the process of data annotation for fake news in the Hindi language. Data annotation is a very important task and essential for any AI/ML/DL project. Annotated data is used for training AI/ML/DL models, models learn for this training data and then work on the test data. Although the task of data annotation when done manually is very tedious and time-consuming it plays a huge role in determining the accuracy of the model. Data annotation is the key to building a successful AI model with high accuracy. The higher the accuracy is the better are the results of the model. The accuracy of the model depends greatly on the quality of the annotated data. If there is even a slight inaccuracy in data annotation the overall accuracy of the entire model is greatly affected. The industry has realized the importance of data annotation and has started investing greatly in companies that provide quality and accurately annotated data. Many companies have come up with data annotation models of very high accuracy and many researchers are still working towards improving the currently existing models and developing new ones.

Many models are available for annotating English data. However, there is still a lack of models which annotate data available in languages other than English. On searching, we came across the research paper by Singhal et al.[28], which focussed on fake news detection in regional languages. In terms of data, our research is entirely based on the data available on the internet. Data Annotation has been in existence for a long time but the data is available mostly in English hence, due to the lack of datasets we collected data from several fact-checking websites using the web scraping technique. These factcheck websites verify Hindi fake news and give evidence to support their arguments. We collected 2206 news links from over 200 different Hindi fact-check articles from numerous websites to create our data set. This dataset consists of both fake (932) and not-fake (1274) news links, along with the sentences of the article where the links are present and the heading. After collecting the data we exhaustively explored every site manually and annotated whether the links point to a piece of fake (1) or not-fake (0) news. After collecting the data and annotating it manually our main task was to create a model which will automate the process of annotation.

We propose a machine-learning and a deep learning based framework to automate the process of data annotation. Our main contributions are:

- First collected data from various fact check websites.
- After extraction, the next step was pre-processing of data. For pre-processing we removed the punctuations and stopwords from the dataset followed by stemming and lemmatizing. Finally we vectorized the entire dataset using th TF/IDF Vectorizer.
- Finally we applied baseline Machine Learning and Deep Learning Models: Gaussian Naive Bayes, Linear Regression, K-Nearest Neighbors, Support Vector Machines and Random Forest Search and Long Short-Term Memory.
- The proposed models are tested on 10%, 20%, 30% and 40% test data of the dataset prepared.

Our model has shown very promising results with high accuracy of 82.35% for the Random Forest model implemented on 10% test data. The highest accuracy for the LSTM model having 100 epochs and a batch size of 64 implemented on 30% test data was 60.42%. There are not many existing models for the Hindi language but the results we achieved are satisfactory and better than many other existing models which

work for other languages.

The rest of the paper is organized in the following manner. In Section 2, we present the related works. Section 3 describes the proposed approach for the detection of links leading to fake news. Section 4 reports the experimental design through the definition of the models used. Section 5 evaluates the proposed method and presents the experimental results and discussions. Section 6 concludes the paper and sheds light on the future directions.

2 Related Work

Fact-checking is a claim that made by public figures which is a task of assessing the truthfulness such as pundits, politicians etc. As we saw, many researchers do not distinguish fact-checking and fake news detection and since both of them are to assess the truthfulness of claims. Generally, fact-checking is broader while the fake news detection usually focuses on news events Thorne and Vlachos 2018 and Elhadad et al (2019) this will give us a comprehensive and brief review of this topic.

Thorne and Vlachos [30] listed the resources and methods available to automate by reviewed the fact-checking in journalism such a related task as well as the tasks that could benefit from them. [20] Elhadad et al. Machete and Turpin 2020 it differentiated the fake news from other forms of misinformation, malinformation, disseminating and disinformation such as hoaxes, propaganda, satire/parody, rumors, clickbait and junk news. The malinformation is added to the classical categories of misinformation and disinformation. Malinformation was defined as the genuine information with the intent to cause harm was sharing by the,. However, junk news and fabricated, which contain the genuine information which cannot be considered, were considered as a possible malinformation realization, which seems contradictory. there was not mentioned the Sentiment analysi in either [20] or Thorne and Vlachos.

The task of Sentiment analysis is extracting emotions, such as customers' unfavorable or favorable impression of a restaurant. Different from fake news detection and rumor detection, sentiment analysis is to do analyze personal emotions but not to do an objective verification of claim.

Bondielli and Marcelloni it was provided an analysis on the various techniques and had described the features that have been considered in fake news approaches, and the techniques is used to perform these tasks, the collection of relevant data is highlighted for performing them is problematic. the one of the most relevant semantic features of fake news texts is used to obtain by that They considered that the information that provided by the sentiment analysis techniques. In recent years, the interest of the research community in disinformation and misinformation has increase with the number of publications on rumors as a steady growth from since 2006 [8]. In the case of fake news, before 2016 there were few publications , but in a 2017 rapid growth started that become the most important research subject which lead by fake news these issues since 2018, surpassing rumors [8]. 1 presents a comparative analysis of hindi article scarping datasets.

The work presents the first publicly available Hindi Fake news dataset with manual annotation and exploration for automation. Throughout our literature review, we found that most of the works introduce a dataset suitable for their research approach and we work on annotation part of dataset and there is some dataset only focused on particular research topics like hindi news dataset Since fake news related research for the Hindi are still in its early stage, we design our dataset in a diverse way so that it can be used in multiple lines of research. So we enrich our dataset with fake and true news with their headline, article, domain and other metadata i.e our dataset description which is explained briefly in the next section.

Dataset	Language	Investigative Steps	Sources
Thorne and Vlachos	English	NLP, automated fact checking, dataset comparison	Journalism- fullfact, politifact
Bondielli and Marcelloni	English	Approaches to detect fake news	Other papers
Shivangi Singhal et al.	Multilingual	Web scraping, annotation, genre study	IFCN rated Indian fact-checking websites
Mohit Bhardwaj et al.	Hindi	Data collection, hostility annotation, benchmarking	BoomLive, Dainik Bhaskar, facebook, twitter
Our Project	**Hindi**	**Web scraping, annotation, automated annotation using ML/DL**	Fact checkers- AajTak, AltNews, Vishwas News, Amar Ujala, News Nation, Webqoof

Table 1: Past work vis a vis our project for fact checking

3 Methodology

Figure 3 illustrates the different steps of our project.

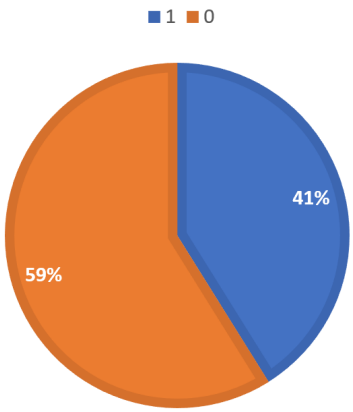

Figure 1: The dataset used to train model has 2206 data points, out of which 41% are fake

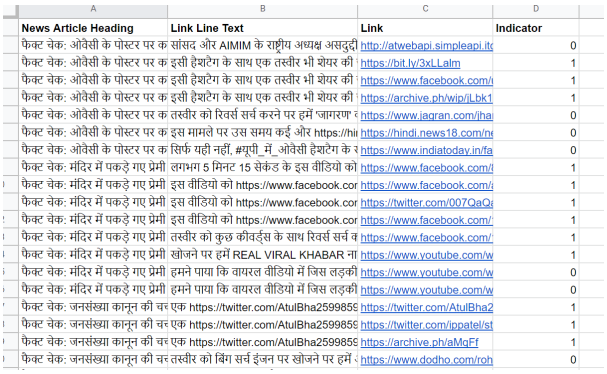

Figure 2: A sample of what the data looks like: each article has multiple links, and we extract all for annotation and model training

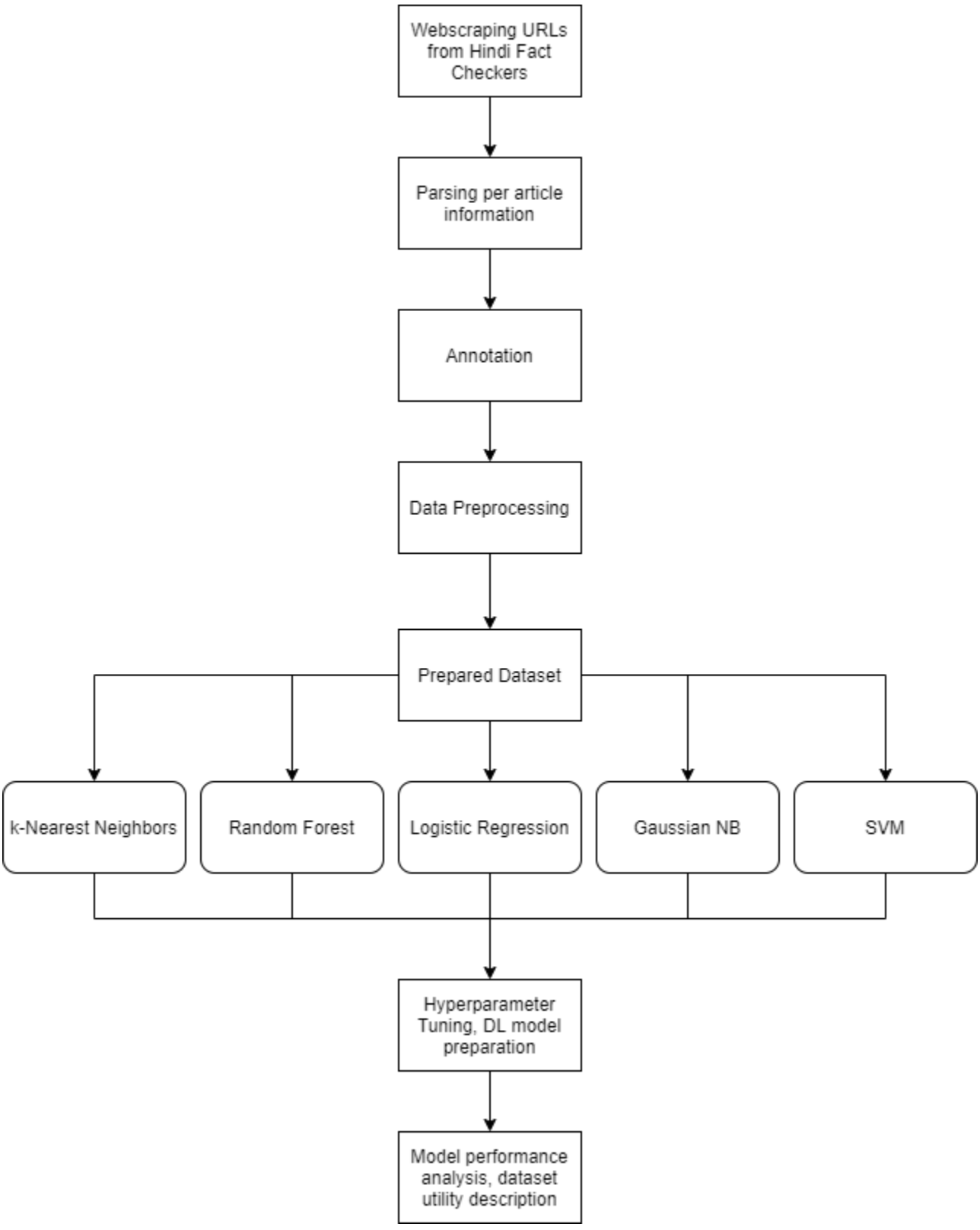

Figure 3: Proposed Workflow

Figure 4: A sample screenshot of the website from which article URLs are extracted by considering only those with are internal pages, and those on banner are manually removed [11]

Figure 5: A sample screenshot of the website from which article URLs are extracted by considering only those with are internal pages, and those on banner are manually removed

3.1 Dataset Preparation

1. **Collecting URLs**
   As of July 19, 2021, we accessed articles including the ones dated back from today from multiple Hindi fact checking websites- Aaj Tak[11] (as shown in Figure 4), AltNews [16], Amar Ujala [31], News Nation [10], Vishwas News [18] and Webqoof [15], and shifted the ones that contained fact check related news from the set. We thus scraped links for **1308 articles**, out of which, we annotated a dataset of 2206 points from Vishwas News and AajTak fact checkers and used them for model training. The next step entailed each article's information extraction for dataset preparation.

2. **Parsing articles**
   The desired information from the article is illustrated in Figure 5. We open each URL in the extracted URL list, and further extract the **heading** of the article, **links** present in the article and **link text**, that is, the text where link is placed in within the article to get a dataset like Figure 2.

3. **Annotation**
   We manually checked each link and annotated for news indicator to identify fake and true news articles in the dataset for model training, which is the **Indicator** column in Figure 2.

4. **Model Preparation**
   For **dataset preprocessing**, html tags were removed, text was cleaned of special characters, irrelevant columns of heading and contained URL were dropped and this cleaned dataset was used for model training. We prepared news tag prediction models to demonstrate dataset utility and automated annotation. The models have been further detailed in

3.2 Exploring Data Preprocessing

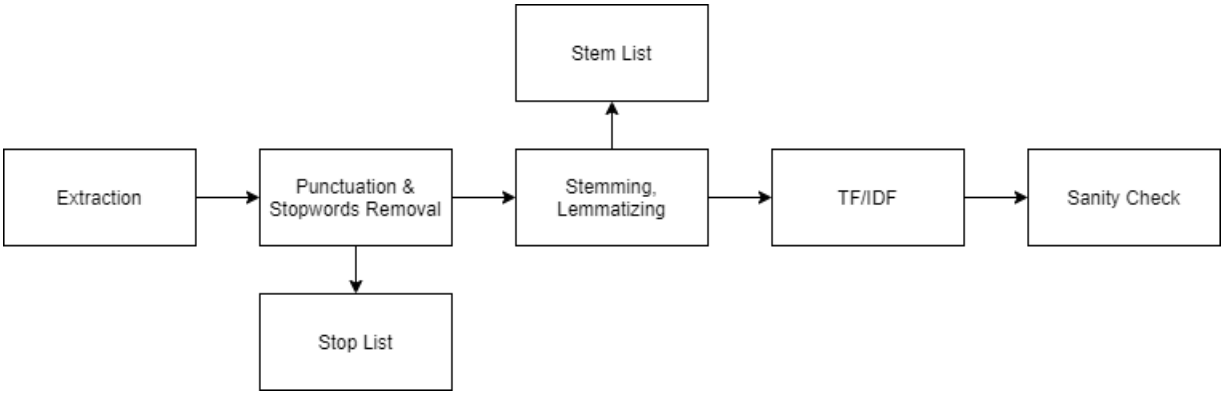

Figure 6: Text Preprocessing- we found that while this is useful for article text preprocessing, parsing phrases and links has much less utility

- The first step for data preprocessing, for us, was to clean the scraped data, removing any duplication and html tags. Further, our data collection was in three steps- 1) collecting articles from the website, 2) parsing article information to get text and links, 3) adding annotation to the prepared dataset- the resulting dataset was further preprocessed to prepare an input to an ML model for exploring automatic annotation. The initial preprocessing of removing tags and extraneous/duplicate data was manual.
- Typical data preprocessing for ML models may involve each/all of these- encoding, feature scaling, standardization, normalization, removing outliers. But for text/natural language datasets, in order to retain only the context and meanignful information, possible preprocessing steps differ- removing punctuation, tokenization, removing stopwords, stemming, lemmatixing, vectorizing, converting text to numbers, and then possibly feature engineering. There are multiple libraries in Python like nltk already available, but since we were working on a Hindi dataset, we simulated preprocessing from a scratch to explore the most relevant method, as shown in Figure 6.
- We used Term Frequency-Inverse Document Frequency (TF-IDF) vectorization to determine the frequency of appearance of a particular word in a given article, and also its frequency of appearance in other articles. We used this to determine the semantic meaning in of our articles and also the importance of certain relevant words being repeated. Terminal Frequency defines the number of times a particular word appears in a particular article with consideration to the total number of words in that article and Inverse Document Frequency gives us an idea about the frequency of appearance of a word across all articles we used in our dataset. TF-IDF is represented as a score that we calculated by multiplying Terminal Frequency and Inverse Document Frequency.
- For preprocessing from a scratch, first step of removing punctuation was the same, an additional character "|" was added. Further, for some systems, encoding utf-08 was required before the system could process the text. First, the space words were removed and then added to a list containing "tokens". Stem words were generated using a list of suffixes used in Hindi, and a dictionary of stem words with there suffixes was constructed. Finally, the stopwords (imported from *stopwords-hi* library) were removed. The title of the article was a less valuable feature and was dropped for model input.
- There are libraries for Hindi NLP that are specific to Hindi's semantics and syntax which we found relevant to our purpose- iNLTK, Indic NLP Library and StanfordNLP. But since the input to our model was not the article text, rather the phrases and links, increasing preprocessing complexity was uncalled for.
- We made word cloud to get the key words in the fake and true articles. After preprocessing, the insignificant and preposition words were removed and the word clouds showed the words that were more relevent in all true news and in all fake news.

Figure 7: Word cloud for link text dataset without preprocessing is preposition-heavy

Figure 8: Word cloud after basic preprocessing brings out more relevant words

Figure 9: Word cloud for link text for fake articles has words like "galat" and social media heavy

Figure 10: Word cloud for link text for true articles has verification words like publish and website

4 Dataset Description

We prepared our dataset using the web-crawling technique. The data that we are using for this project was collected and prepared from various Hindi fact-checking websites like the Aajtak FactCheck, Alt news FactCheck, etc. Our dataset contains 2206 news links from over 200 different news articles. While compiling the collected data we observed that a few news articles were being repeated so we manually removed all the repeated news occurrences. Along with the names of the articles, the dataset contains the news paragraphs which contain those links and the extracted news links. The links present in this dataset were manually annotated as (1) and (0). (1) was used for annotating the links that lead to fake news and (0) was used to annotate the links that do not lead to fake news. Our dataset contains 932 fake news links and 1274 not-fake news links.

One of the major challenges we faced during the project was the pre-processing of data. Numerous pre-processing tools are available for cleaning an English dataset. However, since our entire dataset is in the Hindi language, the pre-processing task was very difficult due to the lack of available resources required for data pre-processing. We took a very simple and direct approach to the problem. We simply removed the stop words and punctuation marks followed by stemming and lemmatizing and vectorized the entire dataset before using the dataset for training the model.

A unique feature of our dataset despite its small size is that it contains a diverse variety of links which helps in training the model well. For example, our data set contains Twitter links which lead to both fake news and supporting material. On carefully observing the dataset we noticed that all the links leading to fake news are present in the same paragraph. Also, generally, there is only one such paragraph in an article. The other paragraphs contain links that lead to proofs and other related material which prove that the news reported is fake. Most of the links which were leading to fake news originated from social media platforms like Facebook, Twitter, etc. We observed that misleading information was being spread not from a single account but various accounts on the same platform. One of the most shocking observations was that even verified accounts on these social media platforms were sources of fake news. Most of the URLs of Facebook archives were leading to fake news. On visiting these links we also observed that most of the misleading videos or articles were already taken down. However, some videos which are spreading misleading information are still unreported. On analyzing these videos we observed that the videos are half fake and half relevant hence making it difficult for the viewers to distinguish and identify them as misleading posts and reporting them hence, as a result, they were still unreported. However, one relieving observation was that none of the sources of fake news were news articles and websites.

Overall, our dataset is unique, diverse and original which helps in training the model well and increases the overall accuracy of the model.

5 Experimental Design

System Configuration: The experiments were run on a computer with Intel Core i5 processor, with 8GB of RAM and 256GB-1TB hard drive, running on Windows 10 operating system.
All the programming has been done in Python.
Train-Test Split: It is a procedure used to get the measure of the performance of a machine learning model. It is used to evaluate the performance of a particular model on new data, and not the data used in training the model [5]. During the train-test split, we take a data set and split it into two subsets, the train data-set and the test data-set. The train data-set is used to train the algorithm and fit the machine learning model and the test data-set, using the input element from the train data, is used to make predictions [29]. The split percentage depends on factors like cost of training and testing the model, size of the data set, etc.

5.1 Exploring Machine Learning logics

We used five ML models to explore automated annotation.

- **Logistic Regression**
  Logistic regression is a supervised learning classification algorithm used to estimate discrete values and predict the probability of a target variable. the output of prediction of occurance of event The nature of target or dependent variable is dichotomous and use to make category or true false decisions from the data .logistic regression by considering a logistic model with given parameters, then seeing how the data can be estimated from coefficients.
  Consider a model with one binary (Bernoulli) response variable $Y$and two predictors, x_1 and x_2, which we denote $p = P(Y = 1)$. In between the predictor variables we assume a linear relationship and the log-odds (also called logit) of the event that $Y = 1$ the following mathematical form of this linear

relationship (where b is the base of the logarithm, ℓ is the log-odds,and $beta_i$ are parameters of the model):

$$\ell = \log_b \frac{p}{1-p} = \beta_0 + \beta_1 x_1 + \beta_2 x_2$$

The highest accuracy (76.01%) was obtained for 10% test data. However, for this project, the accuracy obtained from Logistic Regression implementation is the not best in comparison to accuracy obtained from all the other models for all the different testing datasets.

- **GaussianNB**
  Gaussian Naive Bayes is a variant of Naive Bayes that follows Gaussian normal distribution and supports continuous data.When working with continuous data, an assumption often taken is that the continuous values associated with each class are distributed according to a normal (or Gaussian) distribution.
  An approach to create a simple model is to assume that the data is described by a Gaussian distribution with no co-variance (independent dimensions) between dimensions. This model can be fit by simply finding the mean and standard deviation of the points within each label, which is all what is needed to define such a distribution.

- **k-NN**
  k-NN stands for k Nearest Neighbours. This supervised machine learning algorithm is easy to implement and can be used for both classification and regression. It assumes that similar things exist in proximity [3] hence, it compares the similarity between the data and uses that similarity to predict new data points. The new data point is assigned values based on its similarity to the training dataset. The working of this algorithm is very simple. First, we load the training and test data, and then we choose a value of k. We calculate the distance (Euclidean) between training data and test data and based on this distance we sort them in ascending order. It will choose top k rows from the sorted array and will assign a class to the test point based on the most frequent class of these rows [35].

  For this project, we used a dataset containing 2206 manually annotated Hindi fact-check links along with the sentence of the fact-check which contains the link [Fake : 932 & Not-Fake : 1274]. We performed the train-test split for 10%, 20%, 30% and 40% of the data. The value of k that we have chosen for this project is 5. The highest accuracy (70.13%) was obtained for 10% test data. However, for this project, the accuracy obtained from k-NN implementation is the lowest in comparison to accuracy obtained from all the other models for all the different testing datasets.

- **Random Forest**
  Random forests is a method for supervised learning. It builds multiple decision tree and merge them together and give more accurate and stable prediction. It has the flexibility to be applied both to classification and regression. We used this algorithm for building the model because due to being based on the bagging algorithm and using Ensemble Learning technique it reduces overfitting problem in decision trees and also reduces the variance and therefore improves the accuracy.In our research.
  The highest accuracy (82.3%) was obtained for 10% test data. However, for this project, the accuracy obtained from Random Forest implementation is the highest in comparison to accuracy obtained from all the other models for all the different testing datasets. There are two primarily apparent reasons that make this model ideal for our dataset:

  – Since the fake news dataset is qualitative, an ensemble of models outperform any single classifier. Also, since the training points may or may not be of the same class as that of testing, having a model robust against outliers and invariant to monotonic transformations of the input variables betters performance, which is also why it works well for unsupervised learning.

  – The ensemble is less prone to overfitting and can work with a smaller size dataset, which is not true for DL. Since our dataset is manually annotated with multiple dimensions, Random Forest works well for it.

- **SVM**
  Support Vector Machines refer to supervised learning algorithms used for classification and regression analysis [32]. Ideally suited to our purpose, it is a robust classification mechanism that can perform a linear classification based on choosing a boundary that separates data points correctly with maximum gap, allowing for least error as depicted in Figure **??**, and also allowing for non linear classification by incorporating mapping of data points to a higher dimension. Some notable standard applications of SVM are text, image and biological classification.

- **Gradient descent classifier**

  Stochastic Gradient Descent Classifier is a linear classifier optimized by SDG. SDG is an optimization method used to minimize a cost function. Linear classifiers like Logistic Regression or linear Support Vector Machine are machine learning algorithms/models. The machine learning model defines a loss function, and the optimization method minimizes/maximizes it[12].
  Gradient descent classifier is suited when parameters cannot be computed analytically.

  While the accuracy of a base learner can be increased by boosting, such as linear regression or decision tree, it sacrifices interpretability and intelligibility.[23] Furthermore, its implementation may be more difficult due to the higher computational demand. Gradient boosting models will cause overfitting and can overemphasize because they will continue improving to minimize all errors. It's computationally expensive as it will require many trees above 1000 which can be memory exhaustive. It's easily addressed with various tools as it's less interpretative in nature and the high flexibility results in so many parameters that influence and interact heavily with the behavior of the approach (number of tree depth, regularization, iterations, parameters, etc) and this will require a large grid search during tuning.

5.2 Deep Learning Model

We have also tested our dataset using Long Short Term Memory (LSTM) network in our project, it is a type of recurrent neural network capable of learning order dependence in sequence prediction problems.
We have added four Keras layers:

- **LSTM** has a chain structure that contains four neural networks and different memory blocks called cells, it is a variation of repetitive neural network (RNN) that's pretty successful in anticipating the long arrangements of information like sentences and stock costs over a while. It contrasts with an ordinary feedforward network since there's a feedback loop in its engineering. It moreover incorporates an extraordinary unit known as a memory cell to withhold the past data for a longer time for making a successful forecast.

- **The Embedding Layer** is one of the layers in Keras which is commonly used in Natural Language Processing associated purposes such as language modeling. However, it can additionally be used with different tasks that contain neural networks. While working with NLP problems, we can use pre-trained word embeddings such as GloVe. Alternatively, we can also teach our embeddings the use of the Keras embedding layer.

- **The Dropout Layer** is a critical layer for decreasing over-fitting in neural network models. Intuitively, the main reason for the dropout layer is to remove the clamor that will be present within the input of neurons. This subsequently avoids the over-fitting of the model.

- **The Dense Layer** may be a broadly utilized Keras layer for making a profoundly associated layer within the neural network where each of the neurons of the dense layers gets input from all neurons of the previous layer. At its core, it performs dot item of all the input values together with the weights for getting the yield.

The network has a visible layer with 1 input, 3 hidden layers of Embedding, Dropout, and LSTM, it contains 100 LSTM blocks or neurons and an output layer that makes a single value prediction. The network is

trained for 10, 25, 50, and 100 epochs, and a batch size of 64 is used. Once the model fits, we estimated the performance of the model on the 10%, 20%, 30%, and 40% tested datasets.

6 Results and Observations

6.1 Machine Learning Models

Accuracy	**10% Test Data**	**20% Test Data**	**30% Test Data**	**40% Test Data**
L Regression	76.01%	73.30%	74.47%	72.81%
GaussianNB	61.53%	61.31%	65.10%	60.02%
k-NN	70.13%	65.84%	68.12%	62.97%
Ran Forest	82.35%	76.47%	78.09%	75.76%
SVM	75.56%	73.30%	76.73%	71.34%
Gradient Boosting	78.28%	76.24%	77.79%	75.19%

Table 2: ML model training accuracy variation

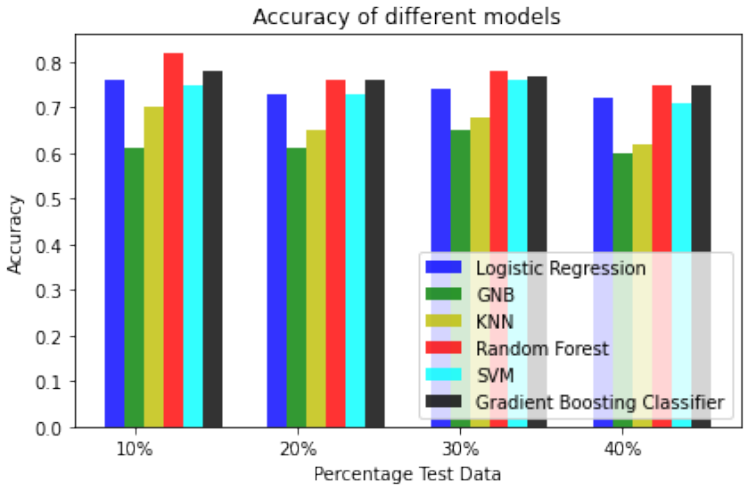

Figure 11: Accuracy Variation with Different Models

We report our results in Table 2. Automated annotation was observed to be not completely fine tuned **due to less features selected for training and small size of the prepared dataset**, but we saw that this particular committee of models did learn from the data.

- When we implemented the models on 10% test data, we got an accuracy of 76.01% for the Logistic Regression model, 61.53% for the Gaussian Naïve Bayes model, 70.13% for the K Nearest Neighbours model, 82.35% for the Random Forest model and 75.56% for the Support Vector Machine model and 78.28% for Gradient Boosting as shown in Table 2.
  For 10% test data, highest accuracy is from the Random Forest model and lowest accuracy is from the Gaussian Naïve Bayes model.

- When we implemented the models on 20% test data, we got an accuracy of 73.30% for the Logistic Regression model, 61.31% for the Gaussian Naïve Bayes model, 65.84% for the K Nearest Neighbours model, 76.47% for the Random Forest model and 73.30% for the Support Vector Machine model and 76.24% for Gradient Boosting as shown in Table 2.
  For 20% test data, highest accuracy is from the Random Forest model and lowest accuracy is from the Gaussian Naïve Bayes model.

- When we implemented the models on 30% test data, we got an accuracy of 74.47% for the Logistic Regression model, 65.10% for the Gaussian Naïve Bayes model, 68.13% for the K Nearest Neighbours model, 78.09% for the Random Forest model and 76.73% for the Support Vector Machine model and 77.79% for Gradient Boosting as shown in Table 2.

  For 30% test data, highest accuracy is from the Random Forest model and lowest accuracy is from the Gaussian Naïve Bayes model.

- When we implemented the models on 40% test data, we got an accuracy of 72.81% for the Logistic Regression model, 60.02% for the Gaussian Naïve Bayes model, 62.97% for the K Nearest Neighbours model, 75.76% for the Random Forest model and 71.34% for the Support Vector Machine model and 75.19% for Gradient Boosting as shown in Table 2.

  For 40% test data, highest accuracy is from the Random Forest model and lowest accuracy is from the Gaussian Naïve Bayes model.

Gaussian Naïve Bayes model has provided low accuracy all throughout. For 40% test data, it provided the lowest accuracy of 60.02% because of which we conclude that this model is not suited for our dataset, which supports intuitive understanding. Support Vector Machine model and Random Forest model consistently give high accuracy but Random Forest has the highest accuracy of 82.35% for 10% test data.

We conclude that Random Forest model is the most suited to our data set as it has lower generalization error and more accuracy.

6.2 Deep Learning Models

Accuracy	**10% Test Data**	**20% Test Data**	**30% Test Data**	**40% Test Data**
Epochs=10	54.10%	56.90%	56.60%	59.04%
Epochs=25	60.50%	59.70%	57.20%	57.20%
Epochs=50	61.10%	58.50%	58.90%	58.30%
Epochs=100	58.60%	59.10%	60.20%	60.10%

Table 3: DL model training accuracy variation

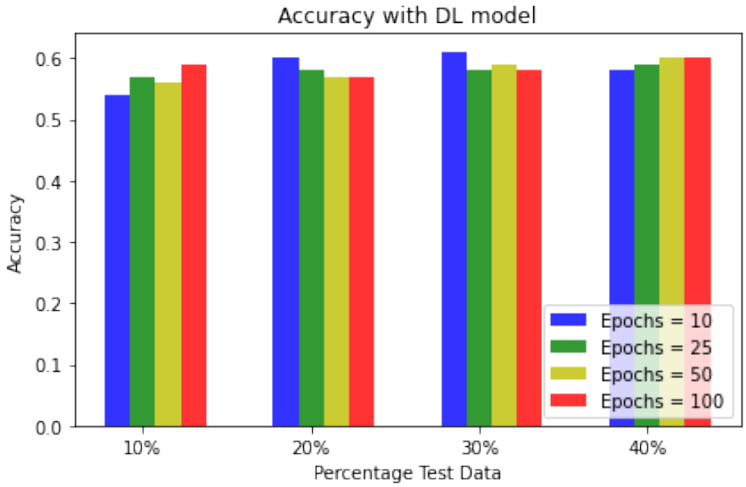

Figure 12: Accuracy Variation with Deep Learning Model

For the DL model training, we see that as the number of epochs increased, more times the weight was changed in the neural network and we acquired a varying accuracy for each subsequent tested dataset. We report the results in Table 3

- When we implemented the DL model with 10 epochs we acquired an accuracy of 54.10% for 10%, 56.90% for 20%, 56.60% for 30%, and 59.04% for 40% tested datasets.
- When we implemented the DL model with 25 epochs we acquired an accuracy of 60.50% for 10%, 59.70% for 20%, 57.20% for 30%, and 57.20% for 40% tested datasets.
- When we implemented the DL model with 50 epochs we acquired an accuracy of 61.10% for 10%, 58.50% for 20%, 58.90% for 30%, and 58.30% for 40% tested datasets.
- When we implemented the DL model with 100 epochs we acquired an accuracy of 58.60% for 10%, 59.10% for 20%, 60.20% for 30%, and 60.10% for 40% tested datasets.

Thus, the highest accuracy is provided by the DL model having 100 epochs and a batch size of 64, on a 30% tested dataset which is 60.20%.

1. Through the research, we extracted information from online fact checkers, manually annotated the information and made a **first of its kind Hindi fact checking dataset**. This dataset can be used as it is, and the preparation methodology is scaleable and relevant to multiple use cases.
2. We also experimented **automatic annotation** by using our dataset as an input to a committee of ML models and a DL model, both of which gave us good results. We were able to identify optimum methods of automatic annotation for Hindi dataset.

7 Future Work and Conclusion

Fake news is a serious problem in today's socio-political context, and it's becoming increasingly hard to distinguish between false and true information. The purpose of the study was to learn more about the problem of misinformation. In this work, we proposed a machine-learning-based framework to automate the process of data annotation on Hindi Fake News Dataset. Our primary focus was on data annotation and automation, after collecting the data, we manually investigated each site to categorize fake (1) and not-fake (0) news.

Our machine learning model has an accuracy of 82.35% for the Random Forest model implemented on

10% test data and the deep learning model has an accuracy of 60.20% when implemented on a batch size of 64 with 100 epochs. We have trained our model in such a way that on providing it with a sentence containing the URL, it generates the required output and classifies whether the news is fake or not. Our best-performing models achieved accuracies that are comparable to the human ability to spot fake content. In the future, we intend to work on Neural Networks and collect a larger dataset for better results. Further, we intend to use more information from the article for automated annotation. By preprocessing the article text, we can 1) extract sentiment and then use that as an input to ML model- our hypothesis is that negative sentiment should yield false indicator for article, 2) make a positive words corpus like "sach, sahi, correct", make negative words corpus like "jhooth, na, nahi", and then add their count, then use that as an input to the model- our hypothesis is that a higher relative count of positive corpus should yield true indicator for the news.

References

[1] https://www.boomlive.in/257-cases-of-fake-news-across-states-170-on-social-media-ncrb 2017-data/amp/. "ncrb". In: (2017).

[2] https://censusofindia2021.com/literacy-rate-of-india 2021/. "censusofindia2021". In: (2021).

[3] https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm 6a6e71d01761. "knn". In: ().

[4] Lawrence Alexander. "Social network analysis reveals full scale of Kremlin's Twitter bot campaign". In: *Global Voices* 2 (2015).

[5] https://machinelearningmastery.com/train-test-split-for-evaluating-machine-learning algorithms/. "train-test". In: (2020).

[6] Hunt Allcott and Matthew Gentzkow. "Social media and fake news in the 2016 election". In: *Journal of economic perspectives* 31.2 (2017), pp. 211–36.

[7] Gary D Bond and Adrienne Y Lee. "Language of lies in prison: Linguistic classification of prisoners' truthful and deceptive natural language". In: *Applied Cognitive Psychology* 19.3 (2005), pp. 313–329.

[8] Alessandro Bondielli and Francesco Marcelloni. "A survey on fake news and rumour detection techniques". In: *Information Sciences* 497 (2019), pp. 38–55.

[9] Leo Breiman. "Random forests". In: *UC Berkeley TR567* (1999).

[10] https://www.newsnationtv.com/fact check. "newsnation fact check". In: (2019).

[11] www.aajtak.in/fact check. "aajtak fact check". In: (2020).

[12] https://michael-fuchs-python.netlify.app/2019/11/11/introduction-to-sgd classifier/introduction. "Gradient Boosting 1". In: ().

[13] en.wikipedia.org/wiki/List_of_languages_by_number_of_native_speakers. "Languages, Wikipedia". In: ().

[14] https://www.cits.ucsb.edu/fake-news/why-we fall. "edufake". In: (2021).

[15] https://hindi.thequint.com/news/webqoof. "webkoof fact check". In: (2019).

[16] https://www.altnews.in/hindi/. "altnews fact check". In: (2019).

[17] https://www.researchgate.net/publication/318981549$_{F}ake_{N}ews_{D}etection_{o}n_{S}ocial_{M}edia_{A}Data_{M}ining_{P}erspective$. "researchgate". In: (2010).

[18] https://www.vishvasnews.com/. "vishvasnews fact check". In: (2020).

[19] https://www.statista.com/topics/2157/internet-usage-in india/. "statista". In: ().

[20] Paul Machete and Marita Turpin. "The use of critical thinking to identify fake news: a systematic literature review". In: *Responsible Design, Implementation and Use of Information and Communication Technology* 12067 (2020), p. 235.

[21] https://cits.ucsb.edu/fake news/spread. "ucsb". In: ().

[22] https://courses.lumenlearning.com/boundless-politicalscience/chapter/forming-public opinion/. "lumenlearning". In: (2010).

[23] Data Analytics in Asset Management: Cost-Effective Prediction of the Pavement Condition Index. "Gradient Boosting 3". In: ().

[24] Verónica Pérez-Rosas et al. "Automatic detection of fake news". In: *arXiv preprint arXiv:1708.07104* (2017).

[25] https://journalistsresource.org/politics-and-government/fake-news-conspiracy-theories-journalism research/. "journalistresource". In: (2017).

[26] Victoria L Rubin, Yimin Chen, and Nadia K Conroy. "Deception detection for news: three types of fakes". In: *Proceedings of the Association for Information Science and Technology* 52.1 (2015), pp. 1–4.

[27] Natali Ruchansky, Sungyong Seo, and Yan Liu. "Csi: A hybrid deep model for fake news detection". In: *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*. 2017, pp. 797–806.

[28] Shivangi Singhal, Rajiv Ratn Shah, and Ponnurangam Kumaraguru. "Factorization of Fact-Checks for Low Resource Indian Languages". In: *arXiv preprint arXiv:2102.11276* (2021).

[29] https://www.jigsawacademy.com/blogs/ai-ml/train-test split. "split". In: (2021).

[30] James Thorne and Andreas Vlachos. "Automated fact checking: Task formulations, methods and future directions". In: *arXiv preprint arXiv:1806.07687* (2018).

[31] https://www.amarujala.com/tags/fact-check-amar ujala. "amarujala fact check". In: (2019).

[32] en.wikipedia.org/wiki/Support vector_machine. "SVM, Wikipedia". In: ().

[33] https://dangerousspeech.org/2016122the-dangerous-side-of-fake-news-rumors-that-inspire violence/. "violencefakenews". In: ().

[34] Nguyen Vo and Kyumin Lee. "Where are the facts? searching for fact-checked information to alleviate the spread of fake news". In: *arXiv preprint arXiv:2010.03159* (2020).

[35] www.tutorialspoint.com. "mlknn". In: (2018).