**REPORT ON FAKE DATA DETECTION SYSTEM**

As a Project Work for Course

**MACHINE LEARNING FOUNDATION (INT 247)**

**NAME:** ABHISHEK SAINI **NAME:** AVEEK SINHA

**REG. NO:** 11902879 **REG. NO:** 11904710

**PROGRAME:** B. Tech CSE

Logo

Description automatically generated with medium confidence**SEMESTER:** SIXTH

**SCHOOL:** SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

**LOVELY PROFESSIONAL UNIVERSITY**

**PHAGWARA, PUNJAB**

**SUBMIT TO:** Prof. UPINDER KAUR

**ACKNOWLEDGEMENT:**

We would like to thank our mentor - Prof. UPINDER KAUR for her advice and inputs on this project. Many thanks to our friends as well, who spent countless hours to listen and provide feedbacks.

**INTRODUCTION:** FAKE DATA DETECTION SYSTEM

Internet is one of the important inventions and many persons are its users. These persons use this for different purposes. There are different social media platforms that are accessible to these users. Any user can make a post or spread the news through these online platforms. These platforms do not verify the users or their posts. So, some of the users try to spread fake news through these platforms. These fake news can be a propaganda against an individual, society, organization, or political party. A human being is unable to detect all these fake news. So, there is a need for machine learning classifiers that can detect this fake news automatically.

World is changing rapidly. No doubt we have a number of advantages of this digital world, but it also has its disadvantages as well. There are different issues in this digital world. One of them is fake news. Someone can easily spread a fake news. Fake news is spread to harm the reputation of a person or an organization. It can be a propaganda against someone that can be a political party or an organization. There are different online platforms where the person can spread the fake news. This includes the Facebook, Twitter etc. Machine learning is the part of artificial intelligence that helps in making the systems that can learn and perform different actions. A variety of machine learning algorithms are available that include the supervised, unsupervised, reinforcement machine learning algorithms. The algorithms first must be trained with a data set called train data set. After the training, these algorithms can be used to perform different tasks. Machine learning is using in different sectors to perform different tasks. Most of the time machine learning algorithms are used for prediction purpose or to detect something that is hidden.

Online platforms are helpful for the users because they can easily access a news. But the problem is this gives the opportunity to the cyber criminals to spread a fake news through these platforms. This news can be proved harmful to a person or society. Readers read the news and start believing it without its verification. Detecting the fake news is a big challenge because it is not an easy task. If the fake news is not detected early, then the people can spread it to others and all the people will start believing it. Individuals, organizations, or political parties can be affected through the fake news.

**Role of Machine Learning in Fake Data Detection:** Increasing use of internet has made it easy to spread the false news. Different social media platforms can be used to spread fake news to several persons. With the share option of these platforms, the news spread in a fast way. Fake news just not only affects an individual, but it can also affect an organization or business. So, controlling the fake news is mandatory. A person can know the news is fake only when he knows the complete story of that topic. It is a difficult task because most of the people do not know about the complete story, and they just start believing in the fake news without any verification. The question arises here how to control fake news because a person cannot control the fake news. The answer is machine learning. Machine learning can help in detecting the fake news. Through the use of machine learning this fake news can be detected

easily and automatically. Once someone will post the fake news, machine learning algorithms will check the contents of the post and will detect it as a fake news. Different researchers are trying to find the best machine learning classifier to detect the fake news.

Accuracy of the classifier must be considered because if it failed in detecting the fake news then it can be harmful to different persons. The accuracy of the classifier depends on the training of this classifier. A model that is trained in a good way can give more accuracy.

Detecting the fake news is one of the most difficult tasks for a human being. The fake news can easily be detected through the use of machine learning. There are different machine learning classifiers that can help in detecting the news is true or false. Nowadays, the dataset can easily be collected to train these classifiers. A classifier with more accuracy is considered as a better classifier. The major thing is the accuracy that is provided by any classifier. Classifier with more accuracy will help in detecting more fake news. Detection of false news is necessary because many persons spread the fake news of social media to mislead the people. To safe the individuals or organizations from losing their reputation because of false news it is necessary to detect it. Many research have said that the machine learning is very helpful in this regard. They used the different machine-learning algorithms and they also found that the Logistic regression is a better classifier because it gives more accuracy.

**Logistic Regression:** This classifier is used when the value to be predicted is categorical. For example, it can predict or give the result in true or false.

Training of the classifiers of machine learning is an important task. This plays an important role for the accuracy of results of these classifiers. A classifier must have to be trained in a proper way with proper data set. Different researchers have trained the machine learning classifiers to detect the fake news. The main problem that occurs while training these classifiers is that mostly the training data set in an imbalanced form.

**OBJECTIVE:**

The main objective behind the development and upgradation of existing projects are the

following smart approaches:

• Be Aware of such article while forwarding to others

• Reveal True stories

• Prevent from false crisis events

• Be Informative

• Be Aware of such article while forwarding to others

• Reveal True stories

• Prevent from false crisis events

• Be Informative

• Be Aware of such article while forwarding to others

• Reveal True stories

• Prevent from false crisis events

• Be Informative

**• Be Aware of such articles/messages while forwarding to others**

**• Reveal True stories**

**• Prevent from false crisis events**

**• Be Informative**

Machine learning (ML) is a type of artificial intelligence (AI) that allows software applications

to become more accurate at predicting outcomes without being explicitly programmed to do so.

Machine learning algorithms use historical data as input to predict new output values.

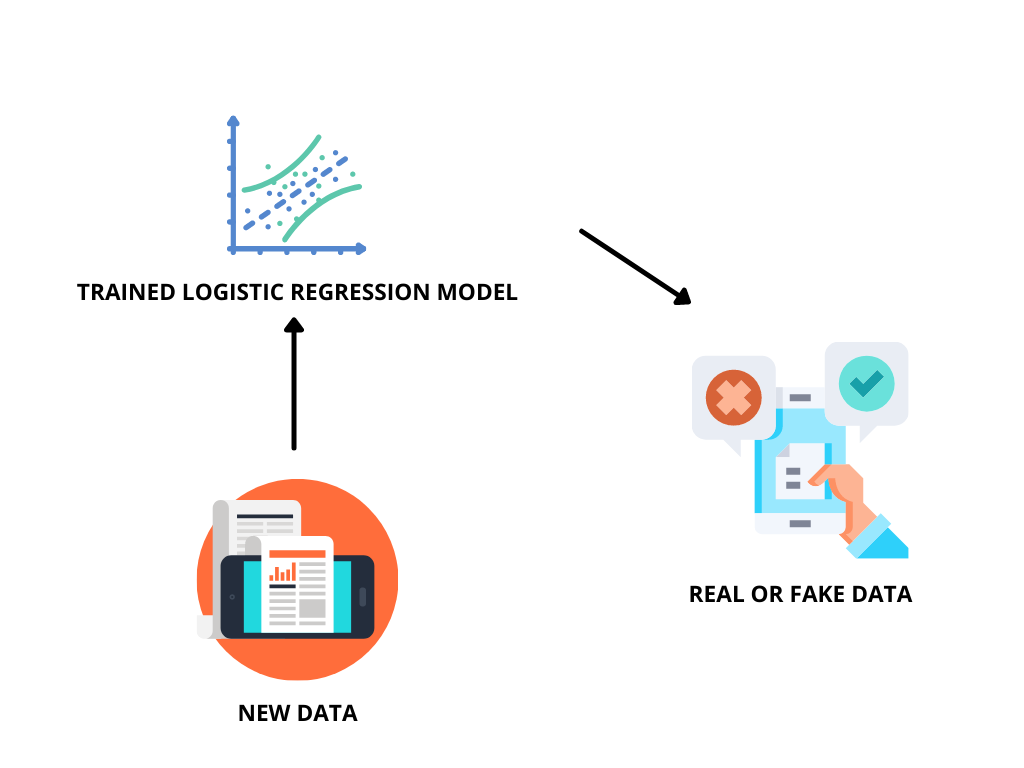
The extensive spread of faux news can have a significant negative impact on individuals and

society. First, fake news can shatter the authenticity equilibrium of the news ecosystem for

instance.

Understanding the truth of new and message with news detection can create positive impact on

the society.

**Diagram

Description automatically generatedWORKFLOW**

**WORKING OF PROJECT**

Firstly, we collected the dataset named as “train” and “spam” dataset. These files we found on the very useful platform named Kaggle.com. We searched for the dataset on various platforms and finally the labelled dataset is required for our project that we found on the Kaggle platform. These files consist of several thousand datasets inside them, and they will be labelled as either it is real news or fake news. The “train” dataset contains many details such as the author of the news, the title of news, text, and labels etc. The “spam” dataset contains two objects V1 and V2 which contain labelled and text data inside the file. After the collection of data, the pre-processing is done, to sort the data for the training of ML model. As we all know that computers can’t understand the direct textual data, so we processed the data into numerical form that is easily understandable by the computer. After pre-processing the data or converting the data into meaningful numbers than we did the splitting of the data into training and testing that is the basic thing a machine learning model requires. We trained the model with training data and to evaluate the results we used testing data. Then after splitting the data, the data goes to Logistic Regression model. We used logistic regression model because this is the binary classification project as we are going to classify the result into two types, its either real or fake. After training this model we got the trained logistic regression model, we did evaluation on this model as we found the accuracy of this model using test data after that we got the trained logistic regression model. After this we are providing the testing data that is new data to the trained model which will predict whether the data is fake or real.

**COMPUTER LANGUAGE AND LIBERARIES USED**

We have used the very famous and most familiar Language to machine learning Python for our project. Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built-in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms and can be freely distributed.

Often, programmers fall in love with Python because of the increased productivity it provides. Since there is no compilation step, the edit-test-debug cycle is incredibly fast. Debugging Python programs is easy: a bug or bad input will never cause a segmentation fault. Instead, when the interpreter discovers an error, it raises an exception. When the program doesn't catch the exception, the interpreter prints a stack trace. A source level debugger allows inspection of local and global variables, evaluation of arbitrary expressions, setting breakpoints, stepping through the code a line at a time, and so on. The debugger is written in Python itself, testifying to Python's introspective power. On the other hand, often the quickest way to debug a program is to add a few print statements to the source: the fast edit-test-debug cycle makes this simple approach very effective.

We required various libraries for this project, these are as follows:

* **NUMPY:** NumPy is the fundamental package for scientific computing in Python. It is a Python library that provides a multidimensional array object, various derived objects (such as masked arrays and matrices), and an assortment of routines for fast operations on arrays, including mathematical, logical, shape manipulation, sorting, selecting, I/O, discrete Fourier transforms, basic linear algebra, basic statistical operations, random simulation and much more.
* **PANDAS:** Pandas is a [Python](https://www.python.org/) package providing fast, flexible, and expressive data structures designed to make working with “relational” or “labelled” data both easy and intuitive. It aims to be the fundamental high-level building block for doing practical, **real-world**data analysis in Python. Additionally, it has the broader goal of becoming **the most powerful and flexible open-source data analysis/manipulation tool available in any language.**

Features of Pandas are:

* Easy handling of **missing data** (represented as Nan) in floating point as well as non-floating-point data
* Size mutability: columns can be **inserted and deleted** from Data Frame and higher dimensional objects
* Automatic and explicit **data alignment**: objects can be explicitly aligned to a set of labels, or the user can simply ignore the labels and let Series, Data Frame, etc. automatically align the data for you in computations
* Powerful, flexible **group by** functionality to perform split-apply-combine operations on data sets, for both aggregating and transforming data
* Make it **easy to convert** ragged, differently indexed data in other Python and NumPy data structures into Data Frame objects
* Intelligent label-based **slicing**, **fancy indexing**, and **subsetting** of large data sets
* Intuitive **merging** and **joining** data sets
* Flexible **reshaping** and pivoting of data sets
* **Hierarchical** labelling of axes (possible to have multiple labels per tick)
* Robust IO tools for loading data from **flat files** (CSV and delimited), Excel files, databases, and saving / loading data from the ultrafast **HDF5 format**
* **Time series**-specific functionality: date range generation and frequency conversion, moving window statistics, date shifting, and lagging.
* **RE:** A RegEx, or Regular Expression, is a sequence of characters that forms a search pattern. RegEx can be used to check if a string contains the specified search pattern.

Function of RegEx:

findall : Returns a list containing all matches

search : Returns a Match object if there is a match anywhere in the string

split : Returns a list where the string has been split at each match

sub: Replaces one or many matches with a string

* **STOPWORDS:** Stopwords are the English words which does not add much meaning to a sentence. They can safely be ignored without sacrificing the meaning of the sentence. For example, the words like the, he, have etc. Such words are already captured this in corpus named corpus. We first download it to our python environment.
* **PORTERSTEMMER:** Stemming is the process of producing morphological variants of a root/base word. Stemming programs are commonly referred to as stemming algorithms or stemmers. A stemming algorithm reduces the words “chocolates”, “chocolatey”, “Choco” to the root word, “chocolate” and “retrieval”, “retrieved”, “retrieves” reduce to the stem “retrieve”.
* **SKLEARN:** Scikit-learn (Sklearn) is the most useful and robust library for machine learning in Python. It provides a selection of efficient tools for machine learning and statistical modelling including classification, regression, clustering and dimensionality reduction via a consistence interface in Python. This library, which is largely written in Python, is built upon NumPy, SciPy and Matplotlib.
* **TFIDFVECTORIZER:** The term **tf–idf** stands for **term frequency–inverse document frequency**, it is a mathematical statistic that is planned to reflect how significant a word is to a record in a collection or corpus. The **tf–idf** esteem builds proportionally to the number of times a word shows up in the document. It is offset by the quantity of documents in the corpus that contain the word, which helps to adjust for the fact that a few words show up more often when all is said in done. **tf–idf** is one of the most well-known term-weighting plans today. An overview led in 2015 demonstrated that 83% of text-based recommender frameworks in advanced libraries use **tf–idf**. It would be difficult to understand **tf–idf** together. So, let's understand each separately.

**Term Frequency (tf) -**It gives us the recurrence of the word in each report in the corpus. It is the proportion of the number of times the word shows up in a report contrasted with the all-out the number of words in that record. It increments as the quantity of events of that word inside the record increments.

**Inverse Data Frequency (idf) -**It is used to figure the heaviness of uncommon words over all reports in the corpus. The words that happen seldom in the corpus have a high IDF score.

* **TRAIN TEST SPLIT:** train\_test\_split is a function in Sklearn model selection for splitting data arrays into two subsets: for training data and for testing data. With this function, you don't need to divide the dataset manually. By default, Sklearn train\_test\_split will make random partitions for the two subsets.
* **LOGISTIC REGRESSION:** [**LogisticRegression**](https://en.wikipedia.org/wiki/Logistic_regression) is a Machine Learning classification algorithm that is used to predict the probability of a categorical dependent variable. In logistic regression, the dependent variable is a binary variable that contains data coded as 1 (yes, success, etc.) or 0 (no, failure, etc.).
* **ACCURACY SCORE:** The accuracy\_score method is used to calculate the accuracy of either the faction or count of correct prediction in Python Scikit learn. Mathematically it represents the ratio of the sum of true positives and true negatives out of all the predictions. Accuracy Score = (TP+TN)/ (TP+FN+TN+FP).
* **NAÏVE BAYES:** Naïve Bayes Classifier is one of the simple and most effective Classification algorithms which helps in building the fast machine learning models that can make quick predictions. It is a probabilistic classifier, which means it predicts on the basis of the probability of an object.
* **SVM:** “Support Vector Machine” (SVM) is a supervised machine learning algorithm that can be used for both classification or regression challenges. However, it is mostly used in classification problems.
* **ENSEMBLE:** We proposed using existing ensemble techniques along with textual characteristics as feature input to improve the overall accuracy for the purpose of classification between a truthful and a false article. Ensemble learners tend to have higher accuracies, as more than one model is trained using a particular technique to reduce the overall error rate and improve the performance of the model. The intuition behind the ensemble modelling is synonymous to the one we are already used to in our daily life such as requesting opinions of multiple experts before taking a particular decision to minimize the chance of a bad decision or an undesirable outcome. For example, a classification algorithm can be trained on a particular dataset with a unique set of parameters that can produce a decision boundary which fits the data to some extent. The outcome of that particular algorithm depends not only on the parameters that were provided to train the model, but also on the type of training data. If the training data contains less variance or uniform data, then the model might overfit and produce biased results over unseen data. Therefore, approaches like cross validation are used to minimize the risk of overfitting. Several models can be trained on different set of parameters to create multiple decision boundaries on randomly chosen data points as training data. Hence, using ensemble learning techniques, these problems can be addressed and mitigated by training multiple algorithms, and their results can be combined for near optimum outcome. One such technique is using voting classifiers where the final classification depends on the major votes provided by all algorithms.
* **RANDOM FOREST:** Random Forest (RF) is an advanced form of decision trees (DT) which is also a supervised learning model. RF consists of large number of decision trees working individually to predict an outcome of a class where the final prediction is based on a class that received majority votes. The error rate is low in random forest as compared to other models, due to low correlation among trees. Our random forest model was trained using different parameters, i.e., different numbers of estimators were used in a grid search to produce the best model that can predict the outcome with high accuracy. There are multiple algorithms to decide a split in a decision tree based on the problem of regression or classification. For the classification problem, we have used the Gini index as a cost function to estimate a split in the dataset. The Gini index is calculated by subtracting the sum of the squared probabilities of each class from one.

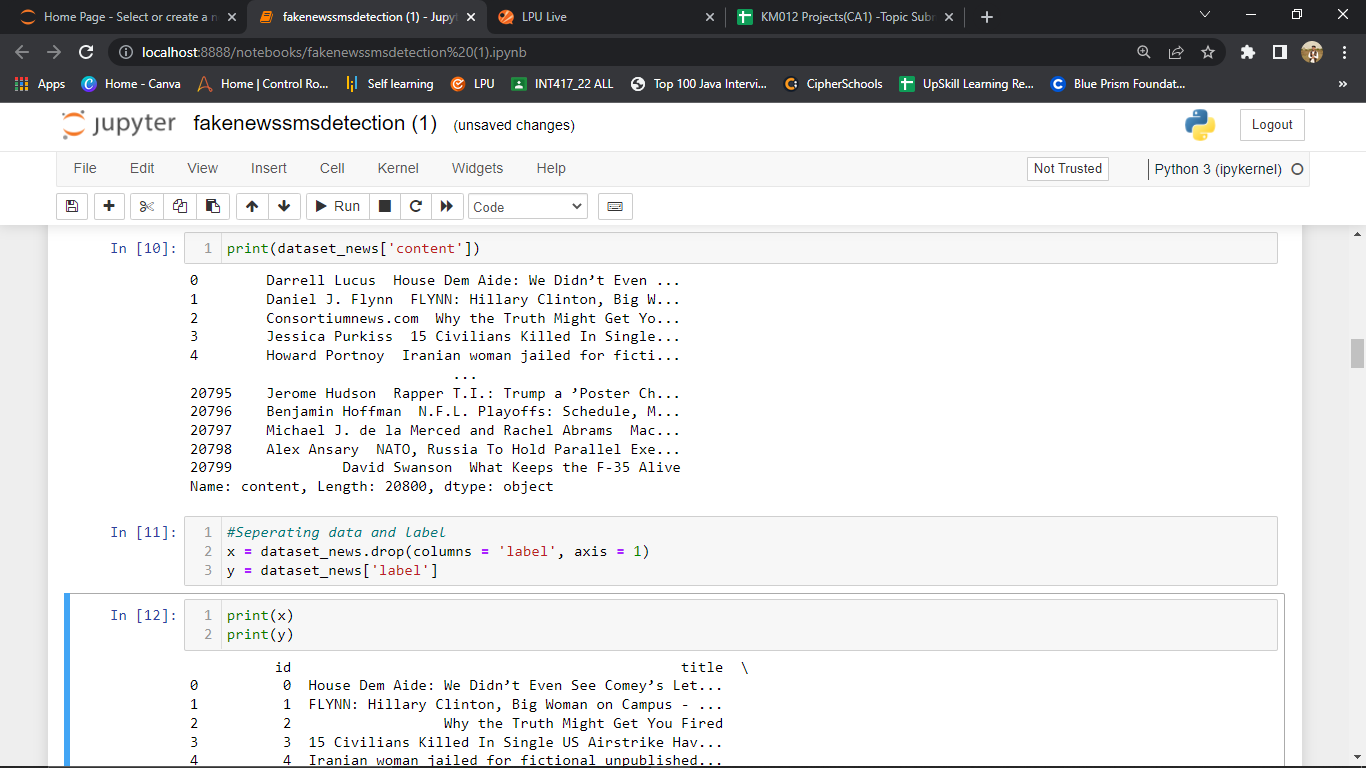
**PROJCT VISUALS**

**Graphical user interface, text, application, email

Description automatically generated**

**Text

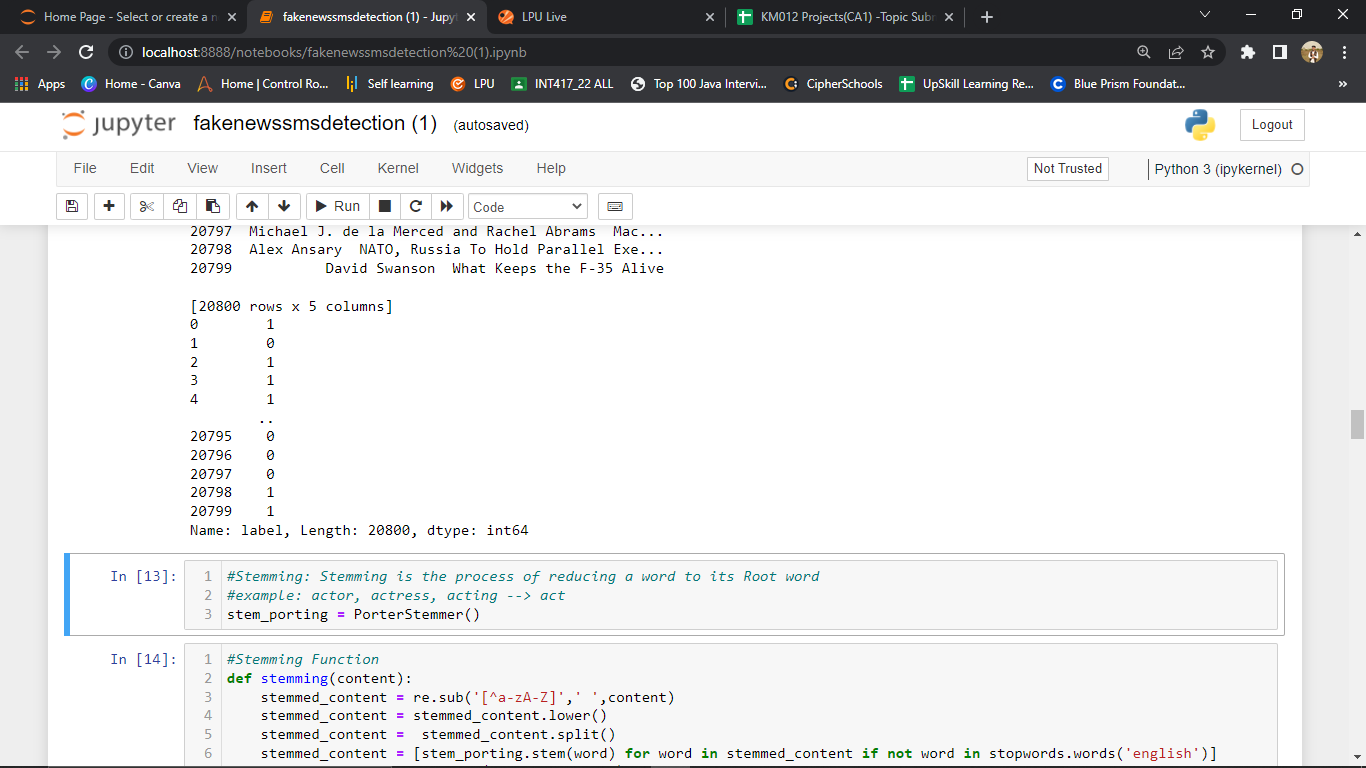
Description automatically generated with medium confidence**

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generatedGraphical user interface, text, application, email

Description automatically generatedGraphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

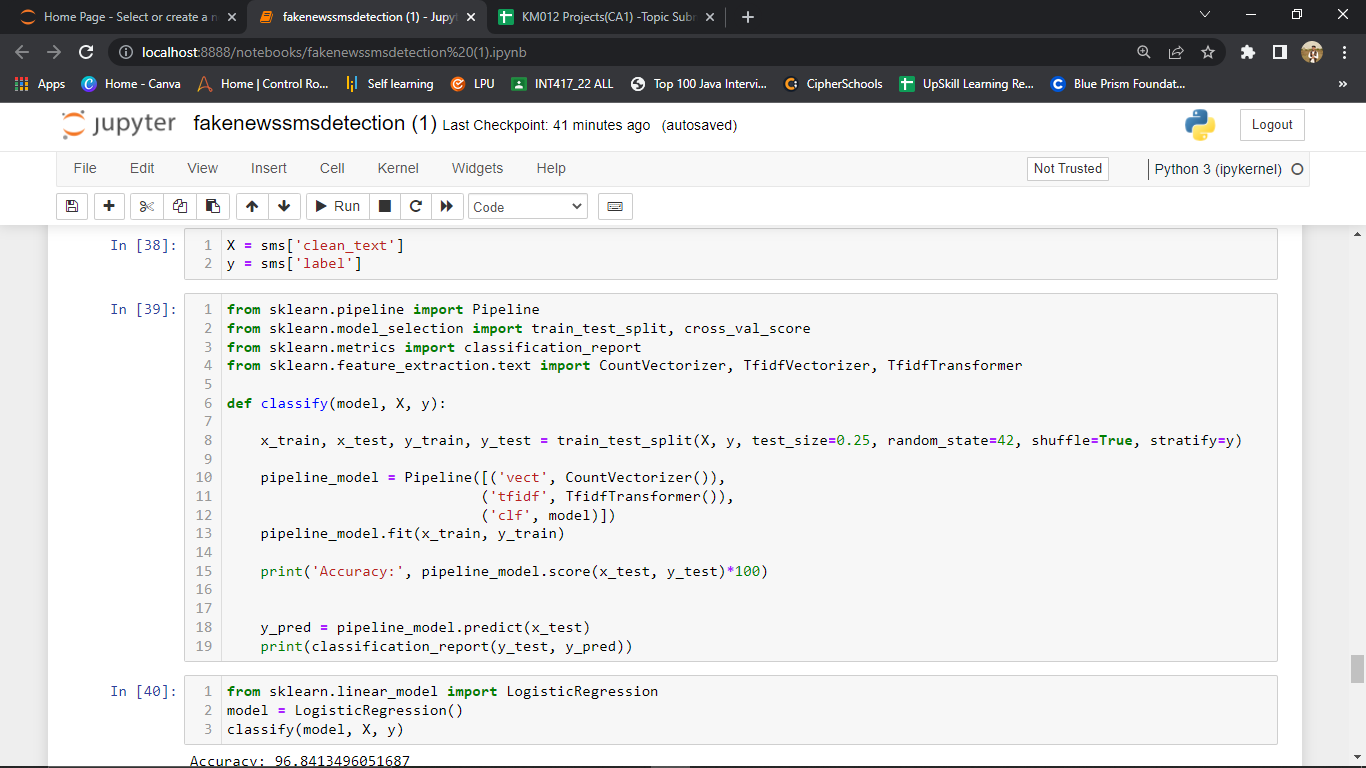
Description automatically generatedGraphical user interface, application

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application

Description automatically generated

A picture containing application

Description automatically generatedA picture containing application

Description automatically generated

Calendar

Description automatically generated with medium confidence

**WORK DONE BY INDIVIDUAL**

**AVEEK SINHA:**

* Collected Dataset (spam.csv)
* Designed Fake SMS Detection System
* Helped in Report Making

**ABHISHEK SAINI:**

* Collected Dataset (train.csv)
* Designed Fake NEWS Detection System
* Report

**CONCLUSION AND SCOPE FOR FUTRE WORK**

The fake news challenge is perilous and is spreading rapidly like a wildfire as it becomes easier for information to reach the mass in various flavours. Reports have shown that, just like in the last US presidential elections, fake news can have a huge impact in politics and thereafter on the people like a domino effect. With the help of artificial intelligence, we can control and limit the spread of such misinformation more quickly and efficiently as compared to manual efforts. The work in this project proposes a stacked model which fine tunes the informational insight gained from the data at each step and then tries to make a prediction. Although many attempts have been made to solve the problem of fake news, any significant success is yet to be seen. With huge amounts of data collected from social media websites like Facebook, Twitter, etc., the best models improve every day. With the use of deep neural networks, the future work in this field seems a lot more promising.

**REFERENCES**

* **Notes:** We learnt various concepts from the lecture/notes of our professor Ms. Upinder Kaur.
* **Dataset1:** <https://www.kaggle.com/c/fake-news/data?select=train.csv>
* **Dataset2:** <https://www.kaggle.com/c/fake-news/data?select=spam.csv>
* **Ideas:** Video Channels (YouTube), Research Papers on Internet.
* **Documentation Content:** NumPy Doc, **Pandas** Doc, Python Doc etc.