**Illicit message on human trafficking in tweets**

**Abstract:** Human trafficking tweets has become a common problem in today’s social media life. Recognition of illicit message in twitter using Computer Vision is very challenging due to its increasing complexity and high interclass variations. In this paper, SVM are used to recognise the tweets. We detecting different human trafficking tweets using SVM classifier, encoder.

Keywords: NB algorithm, Dataset.

**1.Introduction**

Human trafficking, also called trafficking in persons, form of modern-day slavery involving the illegal transport of individuals by force or deception for the purpose of labour, sexual exploitation, or activities in which others benefit financially. Human trafficking is a global problem affecting people of all ages. It is estimated that approximately 1,000,000 people are trafficked each year globally and that between 20,000 and 50,000 are trafficked into the United States, which is one of the largest destinations for victims of the sex-trafficking trade.

Although human trafficking is recognized as a growing international phenomenon, one with a long history (see the story of St. Josephine Bakhita, the patron saint of Sudan and of victims of human trafficking), a uniform definition has yet to be internationally adopted. The United Nations (UN) divides human trafficking into three categories—sex trafficking, labour trafficking, and the removal of organs—and defines human trafficking as the induction by force, fraud, or coercion of a person to engage in the sex trade, or the harbouring, transportation, or obtaining of a person for labour service or organ removal. Though the United States does not acknowledge the removal of organs in its definition, it does recognize sex and labour trafficking and describes human trafficking as the purposeful transportation of an individual for exploitation.

In this work, we use machine learning algorithms for detecting tweets that advertise sexual services of possible victims of trafficking, orienting to the segment of girls under 18 that are being promoted to ‘dates’ and to perform sexual acts. The age of the victims is an indicator of human trafficking.

In the U.S. and other countries, youths constitute the most vulnerable group for becoming victims of this crime. Most women in prostitution entered as minors. The victims of trafficking and violence protection Act (VTVPA) of the year 2000,

Among other definitions, stated that “a commercial sex act induced by force, fraud, or coercion, or in which the person induced to perform such an action has not attained 18 years of age is sex trafficking’’. It is important to note that with these definitions, “any minor, including a U.S. citizen, under the age of 18 who is used in a commercial sex act is a trafficking victim’’.

To detect tweets related to the promotions of sex-trafficking, we will mine messages using characteristics that are related to a description of underage individuals that are used to offer these illegal services. We are going to test the strength of these features to categorize tweets as suspicious to compare results against classification as suspicious or not-suspicious made by experts in the fight against this crime.

**About Dataset:**

**Content:**

The dataset provided is the **Sentiment140 Dataset**which consists of **1,600,000 tweets** that have been extracted using the Twitter API. The various columns present in this Twitter data are:

* **target:**the polarity of the tweet (positive or negative)
* **ids:**Unique id of the tweet
* **date:**the date of the tweet
* **flag:**It refers to the query. If no such query exists, then it is NO QUERY.
* **user:** It refers to the name of the user that tweeted
* **text:** It refers to the text of the tweet

**Twitter Sentiment Analysis: Project Pipeline:**

The various steps involved in the **Machine Learning Pipeline** are:

* Import Necessary Dependencies
* Read and Load the Dataset
* Exploratory Data Analysis
* Data Visualization of Target Variables
* Data Pre processing
* Splitting our data into Train and Test sets.
* Transforming Dataset using TF-IDF Vectorizer
* Function for Model Evaluation
* Model Building
* Model Evaluation

**1.1 Aim and Objective**

**Aim:**

To predict the Human trafficking tweets using Dataset.

**Objective:**

We are developing a ML project to predict the different Human trafficking tweets.

**1.2Existing System**

The system has two phases to recognize the positive and negative tweet. In the first one, it captures Twitter messages that are suspicious of being related to the crime through specific normalized hashtags. In the second phase, the system recognizes positive and negative tweets using SVM algorithm.

**1.3Problem Statement**

Illicit message on Human trafficking tweets in twitter

**1.4Proposed System/Solution**

In the proposed system we are developing a machine learning project. We are detecting the Human trafficking tweets. Using SVM classifier we are predicting whether the tweets are positive or negative. We are predicting Sentimental Analysis using NLP for the detection of the Human trafficking tweets.

**Advantages of SVM**

1. It is highly accurate.
2. It can handle many features.

**Advantages of Sentimental Analysis using NLP**

1. Dictionary is unnecessary.
2. Excellent classification accuracy.
3. High precision ana adaptability.

**2.LITERATURE SURVEY**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SL  NO. | REFERANCE PAPER | LITERATURE REVIEW | MERITS | DEMERITS |
| 1. | Trafficking for sexual exploitation  16 May 2022 – Human trafficking | This form of trafficking affects every region in the world. Victims, most of them women and children, are lured in through different methods, and transported or harbored to or in the place of exploitation where they are often deprived of their travel and identification documents. | The right to life, liberty, and security of person.  . | Human trafficking has a devastating impact on individual victims, who often suffer **physical and emotional abuse, rape**, threats against self and family, passport theft, and even death. |
| 2. | Trafficking for forced labour2014. In P. Kotiswaran | Victims of this widespread form of trafficking come primarily from developing countries. They can be forced to work in labour-intensive jobs such as agriculture, mining, fisheries or construction work, or kept in domestic servitude. | Women and girls account for **4.9 million**of those in forced commercial sexual exploitation, and for **6 million**of those in forced labour in other economic sectors. | The United Nations defines human trafficking as: “The recruitment, transportation, transfer, harboring or receipt of persons, by means of the threat of the use of force or other forms of coercion, of abduction, of fraud, of deception, of the abuse of power or of in a position of vulnerability or of the giving or receiving of payments or benefits to achieve the consent of a person having control of another person. |
| 3. | Trafficking for forced criminal activities 50 per cent of detected victims in 2018 | Victims are forced to carry out a range of illegal activities, which in turn generate income for criminal organizations. Activities can include theft, drug cultivation, selling counterfeit goods, and recently also fraud which is often conducted through the misuse of technology. | Human trafficking is the act of moving an individual with force, coercion, abuse of power through means of transportation and recruitment; harbouring a ‘receipt’ for their purchase. | Many disadvantages faced by these victims include lack of good resources, education and quality of life. |
| 4. | Trafficking for organ removal.[Timothy Caulfield](https://pubmed.ncbi.nlm.nih.gov/?term=Caulfield%20T%5BAuthor%5D), LLB, LLM,1 [Wilma Duijst](https://pubmed.ncbi.nlm.nih.gov/?term=Duijst%20W%5BAuthor%5D), PhD,2 [Mike Bos](https://pubmed.ncbi.nlm.nih.gov/?term=Bos%20M%5BAuthor%5D), MA, MSc,3 | In many countries, waiting lists for transplants are very long, and criminals have seized this opportunity to exploit the desperation of patients and potential victim-donors. After the transplants take place, victim-donors often see little to no compensation for the transaction, facing great health issues and financial challenges to have proper medical follow-up. | Trafficking in human beings for the purpose of organ removal (THBOR) and human organ trade are universally condemned. | Given this interaction, physicians and other health care professionals seem well placed to play a role in the monitoring and, perhaps, the reduction of organ trafficking practices. |
| 5. | Migrant smuggling – key challenges Dec 1, 2021 · 7 takeaways on Emerging Challenges in the Area of Migrant Smuggling | In the last decade, the process of globalization, and the multiple crises impacting different regions of the world through, economic hardships, armed conflicts, terrorism and climate change, have pushed the most vulnerable people to migrate in the look for safer living conditions. | In a crisis context, resorting to smuggling might be the only way for migrants who urgently need to flee their home and seek safety elsewhere. | The key challenge and objective in responding to smuggling situations in the context of conflict and crisis is to identify measures that address. |
| 6. | Organized criminal networks Apr. 19, 2023  [Fatal Hawaii](https://www.britannica.com/news/432090/7cfeda7617dabf9e1e1907d3b76cf132) | Migrant smuggling syndicates are run like businesses, drawn by the high profit margins and low risks. They benefit from weak legislation and a relatively low risk of detection, prosecution and arrest compared to other activities of transnational organized crime. | large sums of [money](https://www.britannica.com/topic/money) have been expended by [syndicate](https://www.britannica.com/topic/syndicate) bosses in an attempt to gain political influence on both local and national levels of government. | The ability of organized crime to flourish in the United States has traditionally rested upon several factors. |

**Python Machine Learning:**

**Machine Learning (ML)** is an automated learning with little or no human intervention. It involves programming computers so that they learn from the available inputs. The main purpose of machine learning is to explore and construct algorithms that can learn from the previous data and make predictions on new input data.

The **input** to a learning algorithm is training data, representing experience, and the **output** is any expertise, which usually takes the form of another algorithm that can perform a task. The input data to a machine learning system can be numerical, textual, audio, visual, or multimedia. The corresponding output data of the system can be a floating-point number, for instance, the velocity of a rocket, an integer representing a category or a class, for example, a pigeon or a sunflower from image recognition.

In this chapter, we will learn about the training data our programs will access and how learning process is automated and how the success and performance of such machine learning algorithms is evaluated.

**Concepts of Learning**

Learning is the process of converting experience into expertise or knowledge.

Learning can be broadly classified into three categories, as mentioned below, based on the nature of the learning data and interaction between the learner and the environment.

* Supervised Learning.
* Unsupervised Learning.
* Semi-supervised learning.

Similarly, there are four categories of machine learning algorithms as shown below:

* Supervised learning algorithm.
* Unsupervised learning algorithm.
* Semi-supervised learning algorithm.
* Reinforcement learning algorithm.

However, the most commonly used ones are **supervised** and **unsupervised learning**.

**Supervised Learning**

Supervised learning is commonly used in real world applications, such as face and speech recognition, products or movie recommendations, and sales forecasting. Supervised learning can be further classified into two types: **Regression** and **Classification**.

**Regression** trains on and predicts a continuous-valued response, for example predicting real estate prices.

**Python Machine Learning – Types of Learning**

**Classification** attempts to find the appropriate class label, such as analyzing positive/negative sentiment, male and female persons, benign and malignant tumors, secure and unsecure loans etc.

In supervised learning, learning data comes with description, labels, targets or desired outputs and the objective is to find a general rule that maps inputs to outputs. This kind of learning data is called **labeled data**. The learned rule is then used to label new data with unknown outputs.

Supervised learning involves building a machine learning model that is based on **labeled samples**. For example, if we build a system to estimate the price of a plot of land or a house based on various features, such as size, location, and so on, we first need to create a database and label it. We need to teach the algorithm what features correspond to what prices. Based on this data, the algorithm will learn how to calculate the price of real estate using the values of the input features.

Supervised learning deals with learning a function from available training data. Here, a learning algorithm analyses the training data and produces a derived function that can be used for mapping new examples. There are many **supervised learning algorithms** such as Logistic Regression, Neural networks, Support Vector Machines (SVMs), and Naive Bayes classifiers.

Common **examples** of supervised learning include classifying e-mails into spam and not-spam categories, labeling webpages based on their content, and voice recognition.

**Unsupervised Learning**

Unsupervised learning is used to detect anomalies, outliers, such as fraud or defective equipment, or to group customers with similar behaviors for a sales campaign. It is the opposite of supervised learning. There is no labeled data here.

When learning data contains only some indications without any description or labels, it is up to the coder or to the algorithm to find the structure of the underlying data, to discover hidden patterns, or to determine how to describe the data. This kind of learning data is called **unlabeled data**.

Suppose that we have a number of data points, and we want to classify them into several groups. We may not exactly know what the criteria of classification would be. So, an unsupervised learning algorithm tries to classify the given dataset into a certain number of groups in an optimum way. Unsupervised learning algorithms are extremely powerful tools for analyzing data and for identifying patterns and trends. They are most commonly used for clustering similar input into logical groups. Unsupervised learning algorithms include K-means, Random Forests, Hierarchical clustering and so on. If some learning samples are labeled, but some other are not labeled, then it is semi-supervised learning. It makes use of a large amount of **unlabeled data for training** and a small amount of **labeled data for testing**. Semi-supervised learning is applied in cases where it is expensive to acquire a fully labeled dataset while more practical to label a small subset. For example, it often requires skilled experts to label certain remote sensing images, and lots of field experiments to locate oil at a particular location, while acquiring unlabeled data is relatively easy.

**Reinforcement Learning**

Here learning data gives feedback so that the system adjusts to dynamic conditions in order to achieve a certain objective. The system evaluates its performance based on the feedback responses and reacts accordingly. The best known of instances include self-driving cars and chess master algorithm AlphaGo.

**Purpose of Machine Learning**

Machine learning can be seen as a branch of AI or Artificial Intelligence, since, the ability to change experience into expertise or to detect patterns in complex data is a mark of human or animal intelligence.

As a field of science, machine learning shares common concepts with other disciplines such as statistics, information theory, game theory, and optimization.

As a subfield of information technology, its objective is to program machines so that they will learn.

Python Machine Learning – Environment Setup

Similarly, we can download and install necessary libraries like numpy, matplotlib etc. individually using installers like pip. For this purpose, you can use the commands shown here:

pip install regex

pip install numpy

pip install matplotlib

pip install pandas

pip install seaborn

pip install wordcloud

pip install nltk

pip install scikit-learn

**Python libraries**

A Python library is a collection of related modules. It contains bundles of code that can be used repeatedly in different programs. It makes Python Programming simpler and convenient for the programmer. As we don’t need to write the same code again and again for different programs. Python libraries play a very vital role in fields of Machine Learning, Data Science, Data Visualization, etc.

**Working of Python Library**

As is stated above, a Python library is simply a collection of codes or modules of codes that we can use in a program for specific operations. We use libraries so that we don’t need to write the code again in our program that is already available. But how it works. Actually, in the MS Windows environment, the library files have a DLL extension (Dynamic Load Libraries). When we link a library with our program and run that program, the linker automatically searches for that library. It extracts the functionalities of that library and interprets the program accordingly. That’s how we use the methods of a library in our program. We will see further, how we bring in the libraries in our Python programs.

**Python standard library**

The Python Standard Library contains the exact syntax, semantics, and tokens of Python. It contains built-in modules that provide access to basic system functionality like I/O and some other core modules. Most of the Python Libraries are written in the C programming language. The Python standard library consists of more than 200 core modules. All this work together to make Python a high-level programming language. Python Standard Library plays a very important role. Without it, the programmers can’t have access to the functionalities of Python. But other than this, there are several other libraries in Python that make a programmer’s life easier. Let’s have a look at some of the commonly used libraries:

1. **Re (regular expression):** Re library in python holds the key to deal with all the problems relating to**textual data analysis**. This library provides a range of methods that can help you build patterns and extract or substitute the desired string.
2. **Matplotlib:**This library is responsible for plotting numerical data. And that’s why it is used in data analysis. It is also an open-source library and plots high-defined figures like pie charts, histograms, scatterplots, graphs, etc.
3. **Pandas:**Pandas are an important library for data scientists. It is an open-source machine learning library that provides flexible high-level data structures and a variety of analysis tools. It eases data analysis, data manipulation, and cleaning of data. Pandas support operations like Sorting, Re-indexing, Iteration, Concatenation, Conversion of data, Visualizations, Aggregations, etc.
4. **Numpy:**The name “Numpy” stands for “Numerical Python”. It is the commonly used library. It is a popular machine learning library that supports large matrices and multi-dimensional data. It consists of in-built mathematical functions for easy computations. Even libraries like TensorFlow use Numpy internally to perform several operations on tensors. Array Interface is one of the key features of this library.
5. **Seaborn:** Seaborn is a library for making statistical graphics in Python. It builds on top of matplotlib and integrates closely with pandas data structures. Seaborn helps you explore and understand your data.
6. **Wordcloud:** Wordcloud is a visualization technique to represent the frequency of words in a text where the size of the word represents its frequency.
7. **NLTK:** NLTK stands for Natural Language Toolkit, is a Python package that you can use for NLP. A lot of the data that you could be analysing is unstructured data and contains human-readable text.
8. **Scikit-learn:** Scikit-learn is an opensource data analysis library, and the gold standard for Machine Learning (ML) in the Python ecosystem.

**Data Analysis:**

Data analysis is a process of inspecting, [cleansing](https://en.wikipedia.org/wiki/Data_cleansing), [transforming](https://en.wikipedia.org/wiki/Data_transformation), and [modelling](https://en.wikipedia.org/wiki/Data_modeling) [data](https://en.wikipedia.org/wiki/Data) with the goal of discovering useful information, informing conclusions, and supporting decision-making. Data analysis has multiple facets and approaches, encompassing diverse techniques under a variety of names, and is used in different business, science, and social science domains. In today's business world, data analysis plays a role in making decisions more scientific and helping businesses operate more effectively.

**Data Visualization of Target Variables**

Data visualization refers to the process of representing data or information in a visual format, such as a graph, chart, or map. The goal of data visualization is to make it easier to understand and analyse large amounts of data by creating visual representations that highlight patterns, trends, and insights.

This is where data visualization comes in — it provides a visual representation of the results of machine learning models, making it easier for non-technical stakeholders to understand and use the insights generated by these models.

In addition, data visualization can also help with the evaluation and debugging of machine learning models. By visualizing the results of a model, it is possible to identify any issues or problems with the model, such as overfitting or bias, and make necessary adjustments to improve the model’s performance.

**The Role of Data Visualization in Machine Learning**

1. **Exploring and Understanding Data**
2. **Evaluating Model Performance**
3. **Communicating Results**
4. **Debugging Machine Learning Models.**

**Types of Data Visualizations**

## **Scatter plot.**

* Bar graph.
* Line graph.
* Heat map.
* Box graph.

**Text pre-processing**

Why do we need to clean the text? Unlike humans, machines like an understanding of the unstructured text, so cleaning the text data is necessary before feeding it to any machine learning algorithm

**Lowercase the tweets**

The first step is transforming the tweets into lower case to maintain a consistent flow during NLP tasks and test mining.

**Remove hyper-links**

Hyperlink are very common in tweets and don’t add any additional information. For any other problem statement, we may need to preserve the hyperlinks. It depends up on the need for the problem statement

**Remove punctuations**

For most NLP problems, punctuation does not provide additional language information. So we generally drop it. Similarly, punctuation symbols are not crucial for sentimental analysis. They are redundant and the removal of punctuation before text modeling is highly recommended

**Remove stopwords**

Stopwords are english words that do not add much meaning to sentence. They can be safely removed without sacrificing the meaning of the sentence.for instance,the words like the, he, have, etc. If we notice, stopwords are some of the most frequently appearing words in any paragraph and do not contribute much meaning to sentences.

**Tokenization**

Tokenization is breaking down the sentence into words and paragraphs into sentences. These broken pieces are called tokens, which help understand the context and create a vocabulary. It works by separating the words by spaces or punctuations

**Stemming**

Stemming works by slicing the end of the word using a list of common prefixes and suffixes like (-ing, -ed, -es). this slicing can be successful on most occasions, but not always.

**Lemmatization**

Lemmatization takes help of the linguistic analysis of the words.it is necessary to have detailed dictionaries that the algorithm can look through to link the form to its lemma. It takes the help of various linguistic insights of that particular word, and due to this very reason, lemmatization is preferred over stemming.

**Vectorization**

Vectorization is the process of transforming a scalar operation acting on individual data elements (Single Instruction Single Data—SISD) to an operation where a single instruction operates concurrently on multiple data elements (SIMD).

Vectorization offers potential speedups in codes with significant array-based computations—speedups that amplify the improved performance obtained through higher-level, parallel computations using threads and distributed execution on clusters.

**TF-IDF vectorization**

TF-IDF vectorization involves calculating the TF-IDF score for every word in your corpus relative to that document and then putting that information into a vector (see image below using example documents “A” and “B”).

TF-IDF is one of the most popular text vectorizers, the calculation is very simple and easy to understand. It gives the rare term high weight and gives the common term low weight.

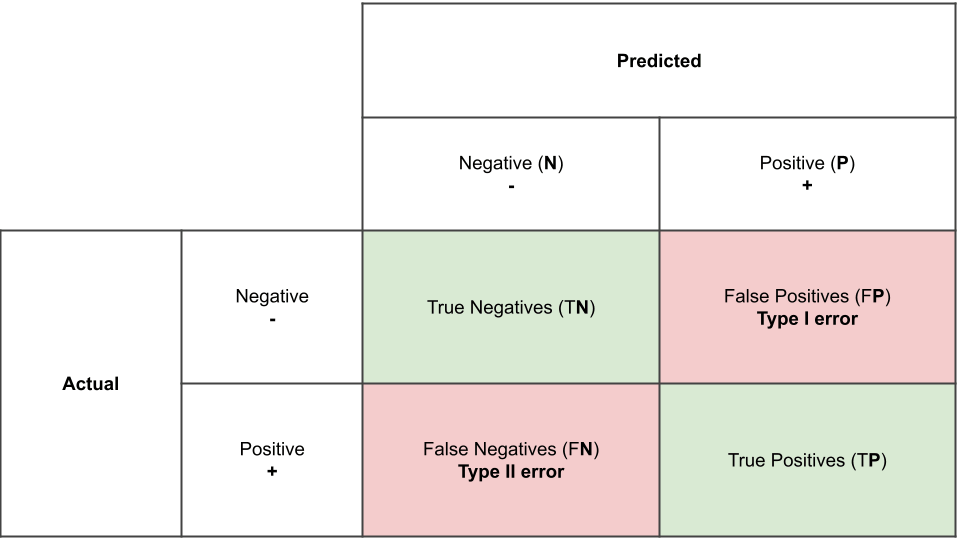
**Data evaluation:**

**Accuracy score**

An Accuracy score (or simply Accuracy) is a Classification measure in Machine Learning that represents a percentage of correct predictions made by a model. Due to its simplicity in calculation and interpretation, the measure has found widespread use. Additionally, the performance of the model is quantified by a single number.

**Confusion matrix**

A Confusion matrix is an N x N matrix used for evaluating the performance of a classification model, where N is the number of target classes. The matrix compares the actual target values with those predicted by the machine learning model.



**ROC-UC curve**

An **ROC curve** (**receiver operating characteristic curve**) is a graph showing the performance of a classification model at all classification thresholds. This curve plots two parameters:

* True Positive Rate
* False Positive Rate

**Pickle:**

Pickle in Python is primarily used in serializing and deserializing a Python object structure. In other words, it's the process of converting a Python object into a byte stream to store it in a file/database, maintain program state across sessions, or transport data over the network.

“Pickling” is the process whereby a Python object hierarchy is converted into a byte stream, and “unpickling” is the inverse operation, whereby a byte stream (from a binary file or bytes-like object) is converted back into an object hierarchy.

**3.Training**

Model training is at the heart of the [**data science development lifecycle**](https://blog.dominodatalab.com/adopting-the-4-step-data-science-lifecycle-for-data-science-projects) where the data science team works to fit the best weights and biases to an algorithm to minimize the loss function over prediction range. Loss functions define how to optimize the ML algorithms. A data science team may use different types of loss functions depending on the project objectives, the type of data used and the type of algorithm.

When a supervised learning technique is used, model training creates a mathematical representation of the relationship between the data features and a target label. In unsupervised learning, it creates a mathematical representation among the data features themselves.

### Importance of Model Training

Model training is the primary step in machine learning, resulting in a working model that can then be validated, tested and deployed. The model’s performance during training will eventually determine how well it will work when it is eventually put into an application for the end-users.

Both the quality of the training data and the choice of the algorithm are central to the model training phase. In most cases, training data is split into two sets for training and then validation and testing.

The selection of the algorithm is primarily determined by the end-use case. However, there are always additional factors that need to be considered, such as algorithm-model complexity, performance, interpretability, computer resource requirements, and speed. Balancing out these various requirements can make selecting algorithms an involved and complicated process.

## **How To Train a Machine Learning Model**

Training a model requires a systematic, repeatable process that maximizes your utilization of your available training data and the time of your data science team. Before you begin the training phase, you need to first determine your problem statement, access your data set and clean the data to be presented to the model.

In addition to this, you need to determine which algorithms you will use and what parameters (hyperparameters) they will run with. With all of this done, you can split your dataset into a training set and a testing set, then prepare your model algorithms for training.

### Split the Dataset

Your initial training data is a limited resource that needs to be allocated carefully. Some of it can be used to train your model, and some of it can be used to test your model – but you can’t use the same data for each step. You can’t properly test a model unless you have given it a new data set that it hasn’t encountered before. Splitting the training data into two or more sets allows you to train and then validate the model using a single source of data. This allows you to see if the model is overfit, meaning that it performs well with the training data but poorly with the test data.

A common way of splitting the training data is to use cross-validation. In [**10-fold cross-validation**](https://www.dominodatalab.com/blog/guide-to-building-models-with-cross-validation), for example, the data is split into ten sets, allowing you to train and test the data ten times. To do this:

1. Split the data into ten equal parts or folds.
2. Designate onefold as the hold-out fold.
3. Train the model on the other nine folds.
4. Test the model on the hold-out fold.

Repeat this process ten times, each time selecting a different fold to be the hold-out fold. The average performance across the ten hold-out folds is your performance **estimate, called the cross-validated score.**

### Select Algorithms to Test

In machine learning, there are thousands of algorithms to choose from, and there is no sure way to determine which will be the best for any specific model. In most cases, you will likely try dozens, if not hundreds, of algorithms in order to find the one that results in an accurate working model. Selecting candidate algorithms will often depend on:

* Size of the training data.
* Accuracy and interpretability of the required output.
* Speed of training time required, which is inversely proportional to accuracy.
* Linearity of the training data.
* Number of features in the data set.

### Tune the Hyperparameters

Hyperparameters are the high-level attributes set by the data science team before the model is assembled and trained. While many attributes can be learned from the training data, they cannot learn their own hyperparameters.

As an example, if you are using a [**regression algorithm**](http://www.sthda.com/english/articles/37-model-selection-essentials-in-r/153-penalized-regression-essentials-ridge-lasso-elastic-net/), the model can determine the regression coefficients itself by analysing the data. However, it cannot dictate the strength of the penalty it should use to regularize an overabundance of variables. As another example, a model using the random forest technique can determine where decision trees will be split, but the number of trees to be used needs to be tuned beforehand.

### Fit and Tune Models

Now that the data is prepared and the model’s hyperparameters have been determined, it’s time to start training the models. The process is essentially to loop through the different algorithms using each set of hyperparameter values you’ve decided to explore. To do this:

1. Split the data.
2. Select an algorithm.
3. Tune the hyperparameter values.
4. Train the model.
5. Select another algorithm and repeat steps 3 and 4.

Next, select another set of hyperparameter values you want to try for the same algorithm, cross-validate it again and calculate the new score. Once you have tried each hyperparameter value, you can repeat these same steps for additional algorithms.

Think of these trials as track and field heats. Each algorithm has demonstrated what it can do with the different hyperparameter values. Now you can select the best version from each algorithm and send them on to the final competition.

### Choose the Best Model

Now it’s time to test the best versions of each algorithm to determine which gives you the best model overall.

1. Make predictions on your test data.
2. Determine the ground truth for your target variable during the training of that model.
3. Determine the performance metrics from your predictions and the ground truth target variable.
4. Run each finalist model with the test data.

Once the testing is done, you can compare their performance to determine which are the better models. The overall winner should have performed well (if not the best) in training as well as in testing. It should also perform well on your other performance metrics (like speed and [**empirical loss**](https://developers.google.com/machine-learning/crash-course/descending-into-ml/training-and-loss)), and – ultimately – it should adequately solve or answer the question posed in your problem statement.

## **Systematic Approach to Model Training**

Using a systematic and repeatable model training process is of paramount importance for any organization planning to build successful [**machine learning model**](https://www.dominodatalab.com/blog/a-guide-to-machine-learning-models) at scale. Central to this is having all of your resources, [**tools**](https://www.dominodatalab.com/blog/data-science-tools), libraries and documentation in a single enterprise platform that will foster collaboration instead of hindering it.

**4.Algorithm**

# **1.Support Vector Machine Algorithm**

Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. However, primarily, it is used for Classification problems in Machine Learning.

The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane.

SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called as support vectors, and hence algorithm is termed as Support Vector Machine. Consider the below diagram in which there are two different categories that are classified using a decision boundary or hyperplane:



**SVM can be of two types:**

* **Linear SVM:** Linear SVM is used for linearly separable data, which means if a dataset can be classified into two classes by using a single straight line, then such data is termed as linearly separable data, and classifier is used called as Linear SVM classifier.
* **Non-linear SVM:** Non-Linear SVM is used for non-linearly separated data, which means if a dataset cannot be classified by using a straight line, then such data is termed as non-linear data and classifier used is called as Non-linear SVM classifier.

**Hyperplane and Support Vectors in the SVM algorithm**:

**Hyperplane:** There can be multiple lines/decision boundaries to segregate the classes in n-dimensional space, but we need to find out the best decision boundary that helps to classify the data points. This best boundary is known as the hyperplane of SVM.

The dimensions of the hyperplane depend on the features present in the dataset, which means if there are 2 features (as shown in image), then hyperplane will be a straight line. And if there are 3 features, then hyperplane will be a 2-dimension plane.

We always create a hyperplane that has a maximum margin, which means the maximum distance between the data points.

**Support Vectors:**

The data points or vectors that are the closest to the hyperplane and which affect the position of the hyperplane are termed as Support Vector. Since these vectors support the hyperplane, hence called a Support vector.

**How does SVM works?**

**Linear SVM:**

The working of the SVM algorithm can be understood by using an example. Suppose we have a dataset that has two tags (green and blue), and the dataset has two features x1 and x2. We want a classifier that can classify the pair(x1, x2) of coordinates in either green or blue. Consider the below image:



Hence, the SVM algorithm helps to find the best line or decision boundary; this best boundary or region is called as a **hyperplane**. SVM algorithm finds the closest point of the lines from both the classes. These points are called support vectors. The distance between the vectors and the hyperplane is called as **margin**. And the goal of SVM is to maximize this margin. The **hyperplane** with maximum margin is called the **optimal hyperplane**.



**Non-Linear SVM:**

If data is linearly arranged, then we can separate it by using a straight line, but for non-linear data, we cannot draw a single straight line. Consider the below image:



So to separate these data points, we need to add one more dimension. For linear data, we have used two dimensions x and y, so for non-linear data, we will add a third-dimension z.

**Advantages of support vector machine:**

* Support vector machine works comparably well when there is an understandable margin of dissociation between classes.
* It is more productive in high-dimensional spaces.

**Disadvantages of support vector machine:**

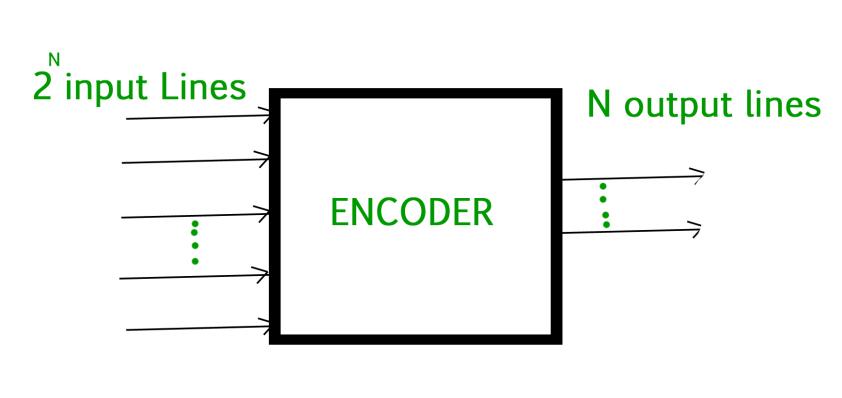
* Support vector machine algorithm is not acceptable for large data sets.
* It does not execute very well when the data set has more sound i.e. target classes are overlapping.

**2.Encoder**

An encoder is a digital circuit that converts a set of binary inputs into a unique binary code. The binary code represents the position of the input and is used to identify the specific input that is active. Encoders are commonly used in digital systems to convert a parallel set of inputs into a serial code.

1. The basic principle of an encoder is to assign a unique binary code to each possible input. For example, a 2-to-4line encoder has 2 input lines and 4 output lines, and assigns a unique 4-bit binary code to each of the 2^2 = 4 possible input combinations.
2. The output of an encoder is usually active low, meaning that only one output is active (low) at any given time, and the remaining outputs are inactive (high). The active low output is selected based on the binary code assigned to the active input.
3. There are different types of encoders, including priority encoders, which assign a priority to each input, and binary weighted encoders, which use a binary weighting system to assign binary codes to inputs.
4. In summary, an encoder is a digital circuit that converts a set of binary inputs into a unique binary code that represents the position of the input. Encoders are widely used in digital systems to convert parallel inputs into serial codes.

An Encoder is a **combinational circuit** that performs the reverse operation of Decoder. It has maximum of **2^n input lines** and **‘n’ output lines**, hence it encodes the information from 2^n inputs into an n-bit code. It will produce a binary code equivalent to the input, which is active High. Therefore, the encoder encodes 2^n input lines with ‘n’ bit

. 

## **Advantages of using Encoders in Digital Logic:**

* Reduction in the number of lines: Encoders reduce the number of lines required to transmit information from multiple inputs to a single output, which can simplify the design of the system and reduce the cost of components.
* Improved reliability: By converting multiple inputs into a single serial code, encoders can reduce the possibility of errors in the transmission of information.
* Improved performance: Encoders can improve the performance of a digital system by reducing the amount of time required to transmit information from multiple inputs to a single output.

## **Disadvantages of using Encoders in Digital Logic:**

* Increased complexity: Encoders are typically more complex circuits compared to multiplexers, and require additional components to implement.
* Limited to specific applications: Encoders are only suitable for applications where a parallel set of inputs must be converted into a serial code.
* Limited flexibility: Encoders are limited in their flexibility, as they can only encode a fixed number of inputs into a fixed number of outputs.
* In conclusion, encoders are useful digital circuits that have their advantages and disadvantages. The choice of whether to use an encoder or not depends on the specific requirements of the system and the trade-offs between complexity, reliability, performance, and cost.

**5. Software Requirements Specifications**

# **5.1 H/W System Configuration:**

|  |  |
| --- | --- |
| Processor | **Dual Core.** |
| Speed | **1.1 G Hz.** |
| RAM | **8 GB (min).** |
| Hard Disk | **B.** |

# **5.2 S/W System Configuration:**

|  |  |
| --- | --- |
| Operating System | **Windows 11.** |
| Technology | **Machine Learning.** |
| IDLE | **Python 3.7 or higher.** |

**Methodology:**

1. We are using dataset.

2. We are Training the dataset.

3. Next we apply the SVM algorithm.

4. We are Testing the dataset.

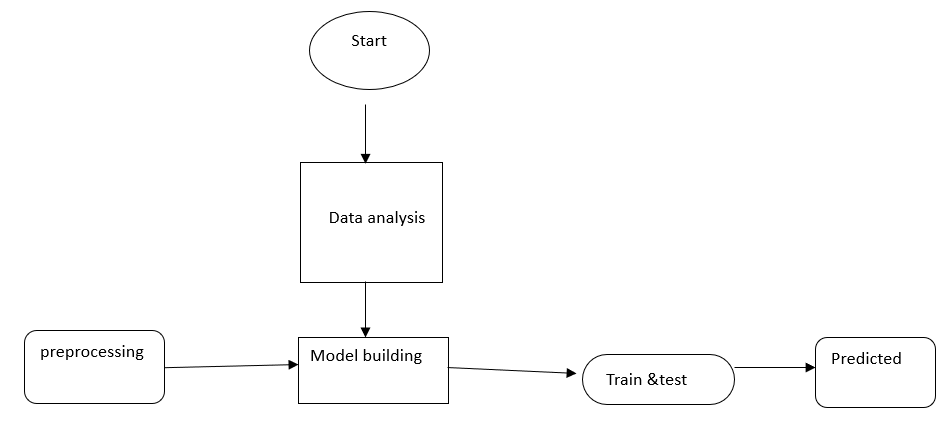
5.Finally detect the positive and negative tweets and prediction is done.

**6. System Design and Architecture**

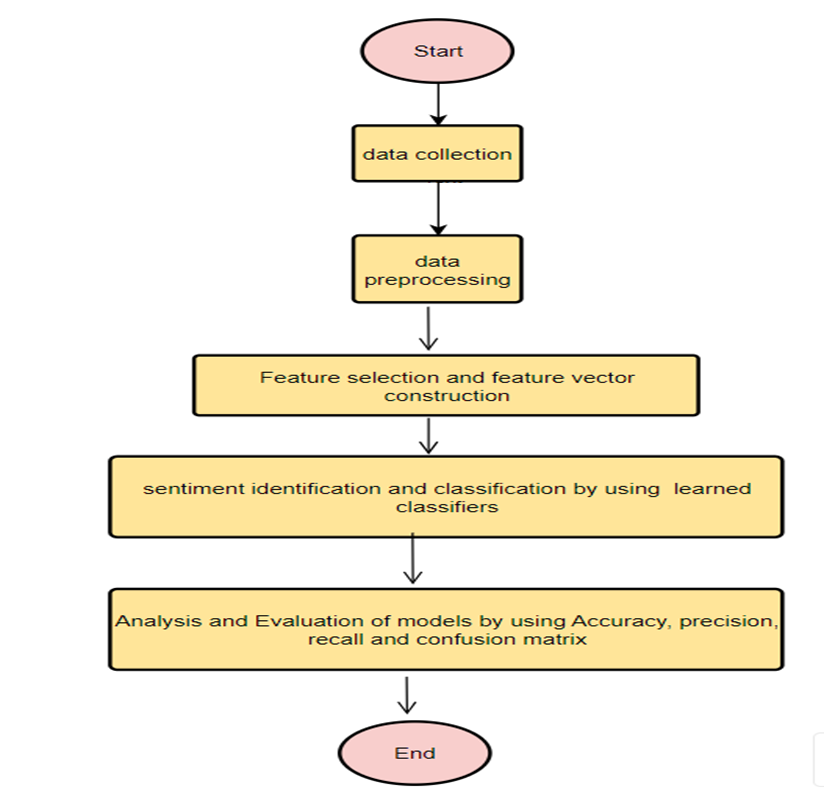
**6.1 Design**

**6.1.1 Architecture Design/ System Architecture**

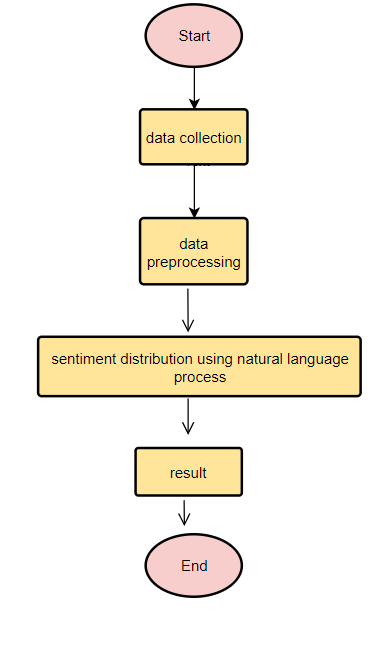
Algorithm



**Fig1:** Architecture of the proposed model



**Fig2.1:** Flow chart for training.

****

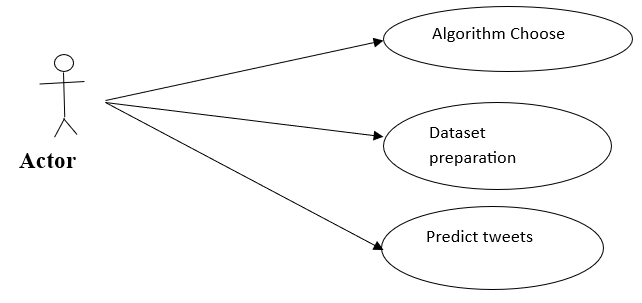
**Fig2.2:** Flow chart for testing.

**UML Diagrams**

Unified Modelling Language (UML) is a standardized general-purpose modelling language in the field of object-oriented software engineering. The standard is managed, and was created, by the Object Management Group.

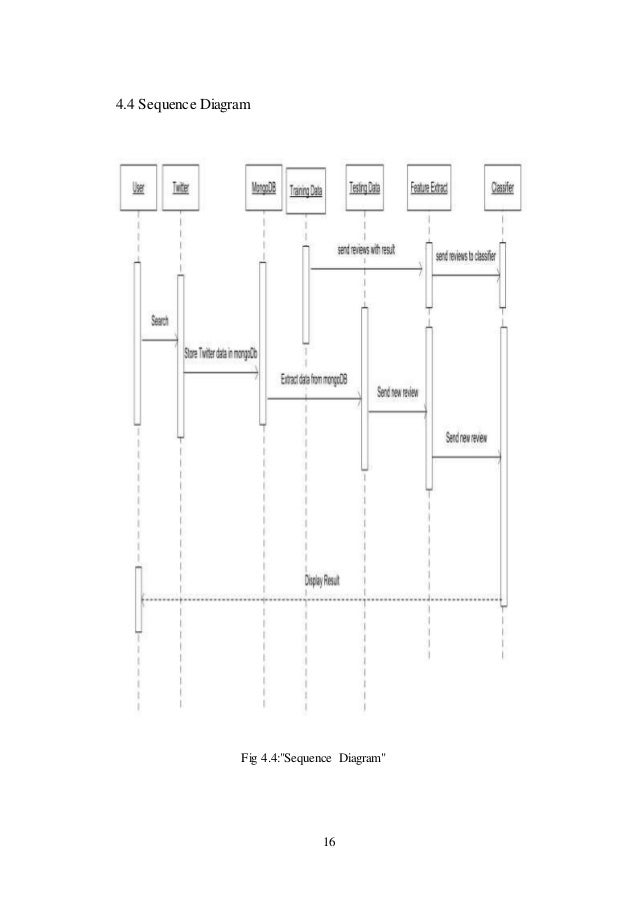
**Use Case Diagrams**

A use case diagram at its simplest is a graphical representation of a user's interaction with the system and depicting the specifications of a use case. A use case diagram can portray the different types of users of a system and the various ways that they interact with the system.

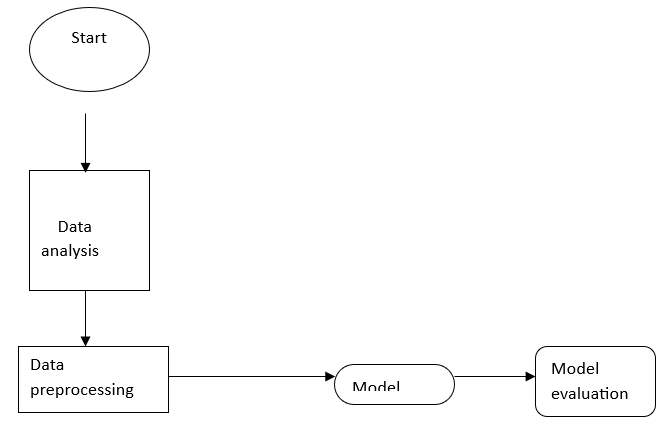


**Sequence Diagram:**

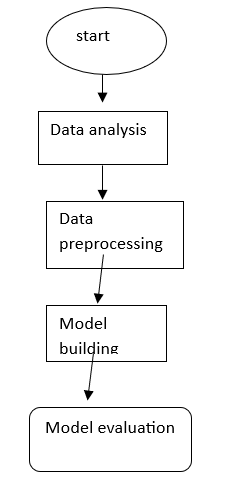
A sequence diagram is a type of interaction diagram because it describes how—and in what order—a group of objects works together. These diagrams are used by software developers and business professionals to understand requirements for a new system or to document an existing process.



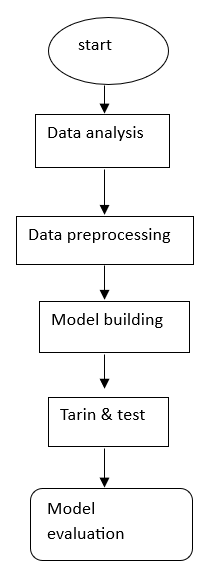
# **Activity diagram:**



# **Object diagram:**



# **Components Diagram:**



**Output Image:**

**Add some output images….**

**7.TESTING**

Testing is a critical element which assures quality and effectiveness of the proposed system in (satisfying) meeting its objectives. Testing is done at various stages in the System designing and implementation process with an objective of developing an transparent, flexible and secured system. Testing is an integral part of software development. Testing process, in a way certifies, whether the product, that is developed, complies with the standards, that it was designed to. Testing process involves building of test cases, against which, the product has to be tested.

**Test objectives**

* Testing is a process of executing a program with the intent of finding an error.
* A good case is one that has a high probability of finding an undiscovered error.
* A successful test is one that uncovers a yet undiscovered error. If testing is conducted successfully (according to the objectives) it will uncover errors in the software. Testing can't show the absences of defects are present. It can only show that software defects are present.

**Testing principles**

Before applying methods to design effective test cases, a software engineer must understand the basic principle that guides software testing. All the tests should be traceable to customer requirements.

**Testing design**

Any engineering product can be tested in one of two ways:

**White box Testing**

This testing is also called as glass box testing. Inthis testing, by knowing the specified function that a product has been designed to perform test can be conducted that demonstrates each function is fully operation at the same time searching for errors in each function. it is a test case design method that uses the control structure of the procedural design to derive test cases.

**Black box Testing**

Inthis testing by knowing the internal operation of a product, tests can be conducted to ensure that "all gears mesh", that is the internal operation performs according to specification and all internal components have been adequately exercised. It fundamentally focuses on the functional requirements of the software.

The steps involved in black box test case design are:

* Graph based testing methods
* Equivalence partitioning
* Boundary value analysis
* Comparison testing

**Testing strategies**

A software testing strategy provides a road map for the software developer. Testing is a set of activities that can be planned in advanced and conducted systematically. For this reason a template for software testing a set of steps into which we can place specific test case design methods should be defined for software engineering process.

**Any software testing strategy should have the following characteristics:**

* 1. Testing begins at the module level and works outward toward the integration of the entire computer-based system.
  2. Different testing techniques are appropriate at different points in time.
  3. The developer of the software and an independent test group conducts testing.
  4. Testing and debugging are different activities but debugging must be accommodated in any testing strategy.

**Levels of Testing**

Testing can be done in different levels of SDLC. They are:

**Unit Testing**

The first level of testing is called unit testing. Unit testing verifies on the smallest unit of software designs-the module. The unit test is always white box oriented. In this, different modules are tested against the specifications produced during design for the modules. Unit testing is essentially for verification of the code produced during the coding phase, and hence the goal is to test the internal logic of the modules. It is typically done by the programmer of the module. Due to its close association with coding, the coding phase is frequently called “coding and unit testing.” The unit test can be conducted in parallel for multiple modules.

**Integration Testing**

The second level of testing is called integration testing. Integration testing is a systematic technique for constructing the program structure while conducting tests to uncover errors associated with interfacing. In this, many tested modules are combined into subsystems, which are then tested. The goal here is to see if all the modules can be integrated properly.

There are three types of integration testing:

* + - *Top-Down Integration*: Top-down integration is an incremental approach to construction of program structures. Modules are integrated by moving downwards throw the control hierarchy beginning with the main control module.
    - *Bottom-Up Integration*: Bottom-up integration as its name implies, begins Construction and testing with automatic modules.
    - *Regression Testing*: In this contest of an integration test strategy, regression testing is the re execution of some subset of test that have already been conducted to ensure that changes have not propagated unintended side effects.

**Functional test**

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Table X: Functional Testing items

|  |  |
| --- | --- |
| Input Images | Input must be accepted. |
| Train | Model will analyse the tweets and train model then  Save model |
| Testing | Yes, pass. |
| Output | Yes, the model predicted, and model analyses the positive or negative tweets. |

**Systems/Procedures:** Interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

**Validation testing**

At the culmination of integration testing, software is completely assembled as a package; interfacing errors have been covered and corrected, and final series of software tests-validating testing may begin. Validation can be defined in many ways, but a simple definition is that validation succeeds when software functions in a manner that can be reasonably expected by customers. Reasonable expectation is defined in the software requirement specification- a document that describes all user visible attributes of the software. The specification contains a section title “validation criteria”. Information contained in that section forms the basis for validation testing approach.

**Alpha testing**

It is virtually impossible for a software developer to forsee how the customer will really use a program. Instructions for use may be misinterpreted; strange combination of data may be regularly used and output that seemed clear to the tester may be unintelligible to a user in field.

When custom software is built for one customer, a series of acceptance tests are conducted to enable the customer to validate all requirements by the end user rather than system developer and acceptable test can range from an informal “test drive” to a planned and systematically executed series of tests. In fact, acceptance testing can be conducted over a period of weeks or months, thereby uncovering cumulative errors that might degrade the system over time. If software is developed as a product to be used by many customers, it is impractical to perform formal acceptance test with each one. Most software product builders use a process called alpha and beta testing to uncover errors that only the end user seems able to find.

A customer conducts the alpha test at the developer’s site. The software is used in a natural setting with the developer “Looking over the shoulder” of the user and recording errors and usage problems. Alpha tests are conducted in controlled environment.

**Beta testing**

The beta test is conducted at one or more customer sites by the end user of the software. Unlike alpha testing, the developer is generally not present. Therefore, the beta test is a “live” application of the software in an environment that cannot be controlled by the developer. The customer records all problems that are encountered during beta testing and reports these to the developer at regular intervals. As a result of problems reported during beta test, the software developer makes modification and then prepares for release of the software product to the entire customer base.

**System Testing and Acceptance Testing**

System testing is actually a series of different tests whose primary purpose is to fully exercise the computer-based system. Include recovery testing during crashes, security testing for unauthorized user, etc.

**Test case**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Case Description** | **Test Data** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| It will predict if the tweet is positive when you enter | Just like #EpsteinClientList | Positive | Positive | pass |
| It will predict if the tweet is negative when you enter | https://t.co/IMWpn3S1E4 | Negative | Negative | Pass |

**Conclusion:**

This paper focused on obtaining twitter messages that are suspected minors for sexual services, and therefore these messages are related to trafficking of persons.

Our system has two phases. in the first one, we used hashtags that have been proven to have to do with this crime. Tweets are proposed to eliminate noise and

Normalize them. With this information, we extract features to input to an SVM algorithm to classify messages as suspicious and not suspicious to build a black list that is going to be further processed.

**References:**

1. [**https://www.britannica.com/topic/human-trafficking**](https://www.britannica.com/topic/human-trafficking)
2. [**https://www.researchgate.net/publication/338451205\_Detection\_of\_Possible\_Human\_Trafficking\_in\_Twitter**](https://www.researchgate.net/publication/338451205_Detection_of_Possible_Human_Trafficking_in_Twitter)
3. [**https://thesidvizstudio.medium.com/why-data-visualization-is-important-in-machine-learning-e534a6c8d5cb#:~:text=Data%20visualization%20can%20help%20identify%20trends%2C%20patterns%2C%20and,variables%20to%20include%20in%20the%20machine%20learning%20model**](https://thesidvizstudio.medium.com/why-data-visualization-is-important-in-machine-learning-e534a6c8d5cb#:~:text=Data%20visualization%20can%20help%20identify%20trends%2C%20patterns%2C%20and,variables%20to%20include%20in%20the%20machine%20learning%20model)
4. [**https://en.wikipedia.org/wiki/Data\_analysis#:~:text=Data%20analysis%20is%20a%20process%20of%20inspecting%2C%20cleansing%2C,useful%20information%2C%20informing%20conclusions%2C%20and%20supporting%20decision-making.%20**](https://en.wikipedia.org/wiki/Data_analysis#:~:text=Data%20analysis%20is%20a%20process%20of%20inspecting%2C%20cleansing%2C,useful%20information%2C%20informing%20conclusions%2C%20and%20supporting%20decision-making.%20)
5. [**https://medium.com/analytics-vidhya/what-is-a-confusion-matrix-d1c0f8feda5**](https://medium.com/analytics-vidhya/what-is-a-confusion-matrix-d1c0f8feda5)
6. [**https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc**](https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc)
7. [**https://www.researchgate.net/publication/342691738\_Detection\_of\_Human\_Trafficking\_Ads\_in\_Twitter\_Using\_Natural\_Language\_Processing\_and\_Image\_Processing**](https://www.researchgate.net/publication/342691738_Detection_of_Human_Trafficking_Ads_in_Twitter_Using_Natural_Language_Processing_and_Image_Processing)
8. [**https://www.guru99.com/test-case.html**](https://www.guru99.com/test-case.html)
9. [**https://www.enjoyalgorithms.com/blog/text-data-pre-processing-techniques-in-ml**](https://www.enjoyalgorithms.com/blog/text-data-pre-processing-techniques-in-ml)