# SMS Spam Detection Using IBM Watson

## 1. INTRODUCTION

### a. Overview

Users are provided with a certain allowance of free SMS messages to any number and it proves to be a great service in areas were internet connection is very limited and the costs of mobile data are very high.This of course attracted the attention of scammers and fraudsters that saw the opportunity to very cheaply lure many eyes to their questionable businesses, which lead to many attempts to abuse the platform.This model and the associated procedures were then developed as an answer to this situation in order to maintain the quality of the user experience within the app.

### b. Purpose

In this project, the goal is to apply different machine learning algorithms to SMS spam classification problem, compare their performance to gain insight and further explore the problem, and design an application based on one of these algorithms that can filter SMS spams with high accuracy.

## 2. LITERATURE SURVEY

### a. Existing problem

Over recent years, as the popularity of mobile phone devices has increased, Short Message Service (SMS) has grown into a multi-billion dollar industry. At the same time, reduction in the cost of messaging services has resulted in growth in unsolicited commercial advertisements (spams) being sent to mobile phones. Due to Spam SMS, Mobile service providers suffer from some sort of financial problems as well as it reduces calling time for users. Unfortunately, if the user accesses such Spam SMS they may face the problem of virus or malware. When SMS arrives at mobile it will disturb mobile user privacy and concentration. It may lead to frustration for the user. So Spam SMS is one of the major issues in the wireless communication world and it grows day by day.

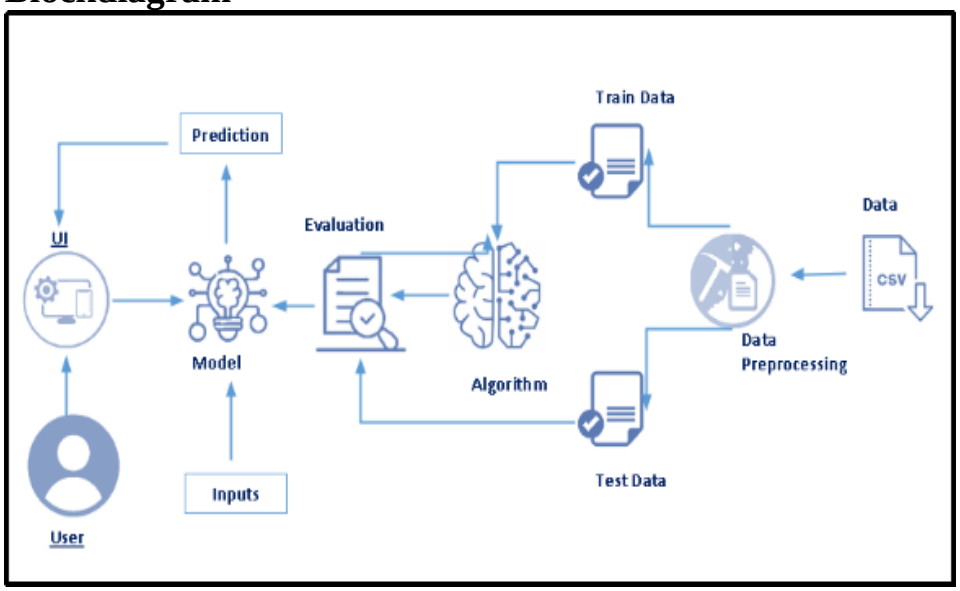
### b. Proposed solution

To avoid such Spam SMS people use white and black list of numbers. But this technique is not adequate to completely avoid Spam SMS. To tackle this problem it is needful to use a smarter technique which correctly identifies Spam SMS. Natural language processing technique is useful for

Spam SMS identification. It analyses text content and finds patterns which are used to identify Spam and Non-Spam SMS.

## 3. THEORITICAL ANALYSIS

### a. Blockdiagram




### b. Hardware/software design

**Software**

- Anaconda Navigator : Anaconda Navigator is a free and open-source distribution of the Python and R programming languages for data science and machine learning-related applications. It can be installed on Windows, Linux, and macOS.Conda is an open-source, cross-platform, package management system. Anaconda comes with so very nice tools like JupyterLab, Jupyter Notebook, QtConsole, Spyder, Glueviz, Orange, Rstudio, Visual Studio Code. For this project, we will be using a Jupyter notebook and Spyder.

**Python packages:**

- NumPy : NumPy is a Python package that stands for 'Numerical Python. It is the core library for scientific computing, which contains a powerful n-dimensional array of objects.
- Pandas : pandas is a fast, powerful, flexible, and easy-to-use open-source data analysis and manipulation tool, built on top of the Python programming language.
- Matplotlib: It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits

- NLTK: is a toolkit build for working with NLP in Python. It provides us various text processing libraries with a lot of test datasets. A variety of tasks can be performed using NLTK such as tokenizing, parse tree visualization, etc…

- The Porter stemming algorithm (or 'Porter stemmer') is a process for removing the commoner morphological and inflexional endings from words in English. Its main use is as part of a term normalisation process that is usually done when setting up Information Retrieval systems.

- Scikit-learn is an open source data analysis library, and the gold standard for Machine Learning (ML) in the Python ecosystem. Key concepts and features include: Algorithmic decision-making methods, including: Classification: identifying and categorizing data based on patterns.

- Flask: Web framework used for building Web applications

**Hardware**

**Device name :** LAPTOP-DERTB7AO

**Processor :** Intel(R) Core(TM) i5-8265U CPU @ 1.60GHz 1.80 GHz

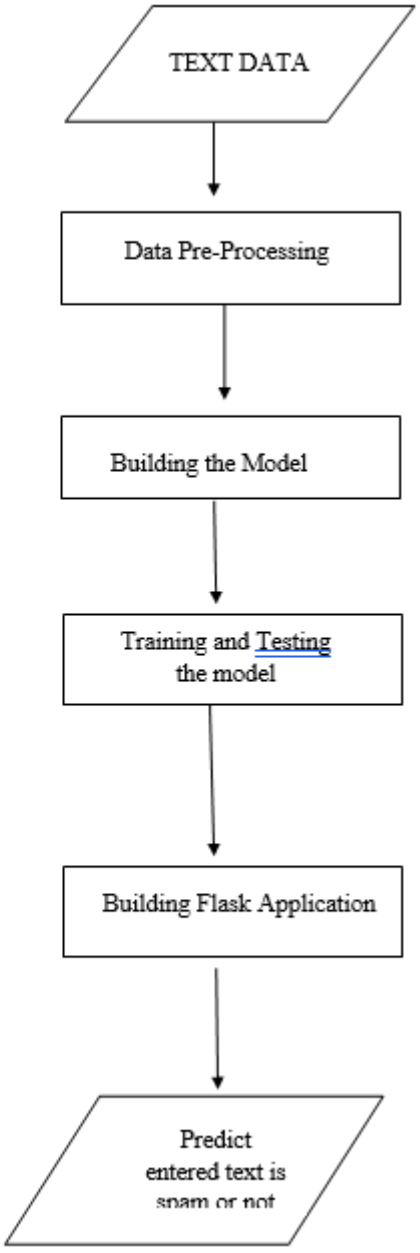
**System type** : 64-bit operating system, x64-based processor

## 4. EXPERIMENTAL INVESTIGATIONS

The Text need to be organized before proceeding with the project. Collect the dataset or create the dataset. Import the required libraries for the model to run. Here, we are reading the dataset(.csv) from the system using pandas and storing it in a variable 'df'. It's time to begin building your text classifier! The data has been loaded into a DataFrame called df. Sometimes you may find some data missing in the dataset. We need to be equipped to handle the problem when we come across them. Obviously you could remove the entire line of data but what if you are unknowingly removing crucial information? Of course we would not want to do that. One of the most common ideas to handle the problem is to take a mean of all the values for continuous and for categorical we make use of mode values and replace the missing data The most important step is to clean the text. Cleaning text means removing all the punctuation, removing stopwords, performing stemming, lemmatization, and converting the text into vectors.. When you are working on a model and you want to train it, you obviously have a dataset. But after training, we have to test the model on some test dataset. For this, you will have a dataset which is different from the training set you used earlier. But it might not always be possible to have so much data during the development phase. In such cases, the solution is to split the dataset into two sets, one for training and the other for testing. You will need to train the datasets to run smoothly and see an incremental improvement in the prediction rate. Finally, we need to check to see how well our model is performing on the test data. There are many evaluation techniques. For this, we evaluate the score produced by the model. Pickle

is used for serializing and de-serializing Python object structures, also called marshalling or flattening. Serialization refers to the process of converting an object in memory to a byte stream that can be stored on disk or sent over a network. Later on, this character stream can then be retrieved and de-serialized back to a Python object. Save our model by importing pickle files.

## 5. **FLOWCHART**



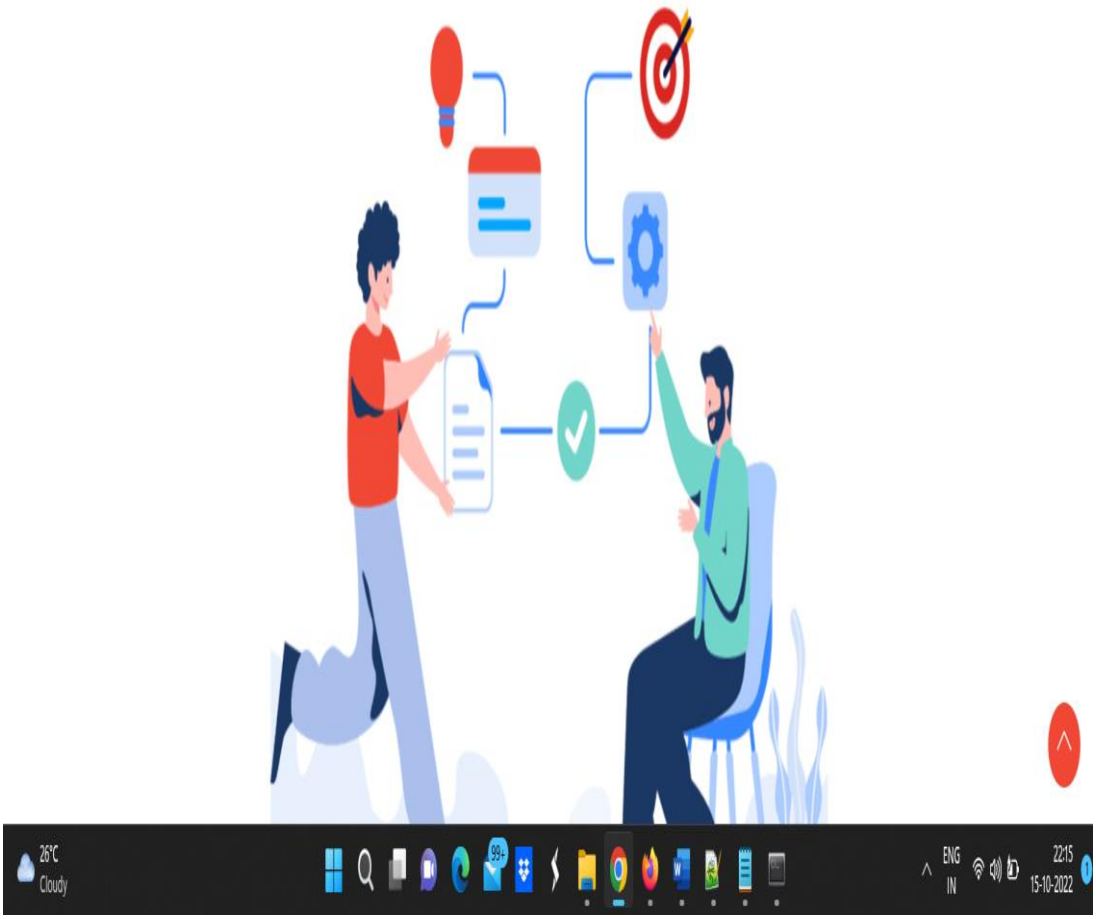

## 6.RESULT

The output of this project is if we are giving a text message then it will predict which category it belongs to whether it is spam or not.

**screenshots:**

SPAM DETECTION

Spam Detector for Short Message Service (SMS).



# Spam Detection.

Your Message

abasements darer prudently fortuitous undergone

lighthearted charm orinoco taster

railroad affluent pornographic cuvier

irvin parkhouse blameworthy chlorophyll

robed diagrammatic fogarty clears bayda

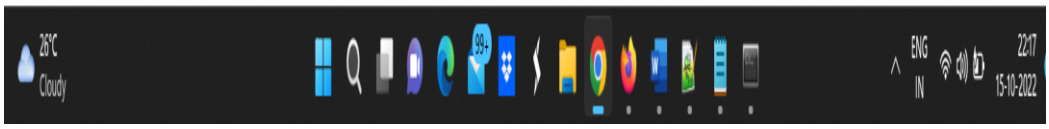
inconveniencing managing represented smartness hashish

academies shareholders unload badness

danielson pure caffein

spaniard chargeable levin

Predict

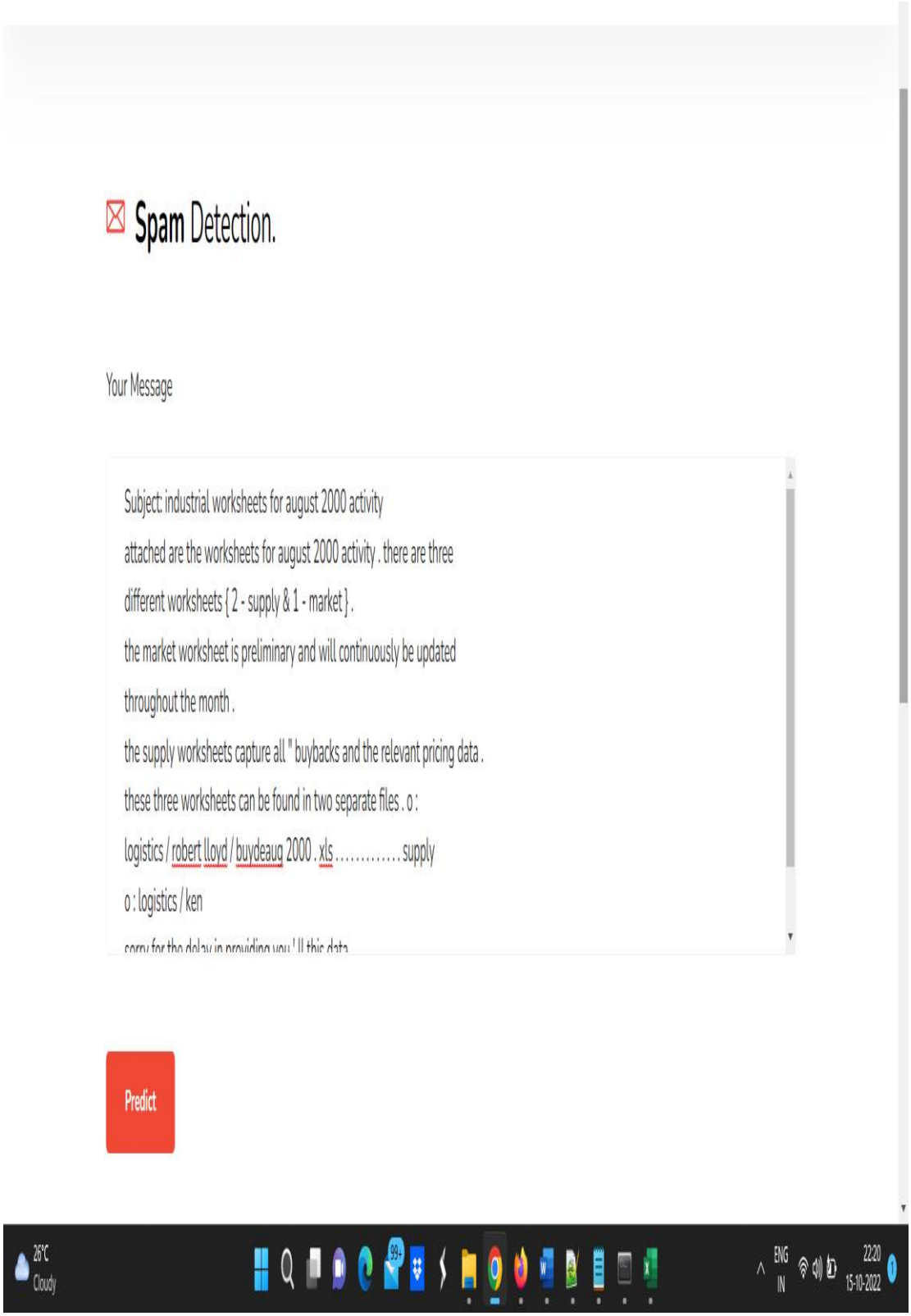


## Spam Detector for Short Message Service (SMS)

*A Machine Learning Web App, Built with Flask.*

## Prediction: Gotcha! This is a SPAM message.


Made with ♥ by SmartBridge.


Spam Detection.
Your Message
Subject: industrial worksheets for august 2000 activity
attached are the worksheets for august 2000 activity . there are three
different worksheets { 2 - supply & 1 - market } .
the market worksheet is preliminary and will continuously be updated
throughout the month .
the supply worksheets capture all " buybacks and the relevant pricing data .
these three worksheets can be found in two separate files . o :
logistics / robert lloyd / buydeaug 2000 . xls ..... supply
o : logistics / ken
Predict
26°C
Cloudy
ENG
IN
22:20
15-10-2022

Spam Detector for Short Message Service (SMS)

*A Machine Learning Web App. Built with Flask.*

Prediction: Great! This is NOT a spam message.

Made with ♥ by SmartBridge.

## 7.ADVANTAGES

- **Easy to use**
- **Cost efficient**
- **Time efficient**

## 8.CONCLUSION

This project was about Automatic labeling based on frequency allows for a reasonable creation of a labeled dataset.Spam detection can be re-framed as a regression problem and the added structure of message spam probability provides a more subtle classification.

The model successfully detects new spam patterns not seen on the training set.

Results improve considerably from better labels.Spam probabilities should not be interpreted in a vacuum but rather in the context of the problem and the analyzed dataset.

Running the model on the cloud is very cheap due to its speed and low memory usage.

## 9. FUTURE SCOPE

In the future, we plan to deal with more challenging problems such as the analysis and management of report in spam SMS filters storing. Solution for this problem is another focus of work in the future.

## 10.BIBILOGRAPHY


- https://towardsdatascience.com/spam-detection-in-sms-messages-3322e03300f5
- https://smartinternz.com/Student/guided_project_info/319022#


## 11. APPENDIX

**source code:**

https://github.com/smartinternz02/SI-GuidedProject-319022-1664614261