Project Synopsis

On

***SPAM DETECTION..***

**Index and Tables**

1. **Project Description**
2. **Modules**
3. **Data flow diagram**
4. **ER-Diagram**
5. **Language/Libraries/Tools**

**1.Description Of Project:**

Text mining (deriving information from text) is a wide field which has gained popularity with the huge text data being generated. Automation of a number of applications like sentiment analysis, document classification, topic classification, text summarization, machine translation, etc has been done using machine learning models.

Spam filtering is a beginner’s example of document classification task which involves classifying a message as spam or non-spam

( ham) mail. Spam box in your Gmail account is the best example of this.

Before starting we must preprocess the messages. First of all, we shall make all the character lowercase. This is because ‘free’ and ‘FREE’ mean the same and we do not want to treat them as two different words.

Then we tokenize each message in the dataset. Tokenization is the task of splitting up a message into pieces and throwing away the punctuation characters. The words like ‘go’, ‘goes’, ‘going’ indicate the same activity. We can replace all these words by a single word ‘go’.

We then move on to remove the stop words. Stop words are those words which occur extremely frequently in any text. For example words like ‘the’, ‘a’, ‘an’, ‘is’, ‘to’ etc. These words do not give us any information about the content of the text. Thus it should not matter if we remove these words for the text.

 You can also use n-grams to improve the accuracy. As of now, we only dealt with 1 word. But when two words are together the meaning totally changes.

**2.Modules:**

1. **Home:**
2. Through this HR can takes the user to the starting point of the site.

**2.login:**

In this module HR enter the User id and password is checked andonly valid user id and password will get entry into search zone. This is a security feature to avoid entry of unauthorized users.

**3.RESET:**

Through this HR can reset all the fields.

**4.Predict:**

In this module HR can predict whether the email given is spam or ham.

**3.Data Flow Diagram:**

**DFD**

The Data flow Diagram shows the flow of data. It is generally made of symbols given below:

(1) A **square** shows the Entity: -

(2) A **Circle** shows the Process: -

(3) An **open Ended Rectangle** shows the data store: --

(4) An **arrow** shows the data flow:-

**Context Level DFD**

filter

spam

inbox

spam

User

**4.E-R Diagram:**

**Definition:**

An entity-relationship (ER) diagram is a specialized graphic that illustrates the interrelationships between entities in a database. ER diagrams often use symbols to represent three different types of information. Boxes are commonly used to represent entities. Diamonds are normally used to represent relationships and ovals are used to represent attributes.

**Entity Relationship (ER) diagram:**

This diagramming technique is used to visually present a database schema or data model and was original proposed by Chen in the 1970s. There are many different data modeling notations; some are very similar to UML class diagrams (with the exception of operations). However, the notation the used here is slightly different, as proposed by Elmasri, et al.

The database schema for this system is shown in figure. The table object has been left out of the diagram because the table management feature set had been dropped from the requirements before this stage of the design process.

Some important database design decisions are as follows:

\_ To store the total price of an order with the order rather than calculating it on the fly when looking at past orders. This is because the price of menu items could change at any time, so the total price at the time of ordering must be stored so that the total price is not incorrectly calculated in future.

\_ Similar to the previous point, the order receipt is stored as a hard-copy and not regenerated when reviewing past orders because things such as the restaurant name or VAT percentage are subject to change. Receipts stored need to be exactly the same as the customer copy in case of dispute.

**Note:** In this project we have not used any database table since it is related to analysis and we read data from docx files which are unstructured so there is no need of ERD .

1. **Language/Libraries/Tools:**

**Front End :**

Python tkinter

**Back End** **:**

Python and DataScience

**Libraries** **:**

* + - * + Tkinter
        + String
        + Sklearn
        + PIL
        + ImageTk
        + MultinomailNB
        + Naïve bayes
        + Panda
        + Regular expression
        + docxpy
        + os
        + pyinstaller

**Other S/W** **:**

* + - * + Python3.x
        + IDLE
        + Anaconda(jupyter lab)

.