Project Synopsis On

Spam Detection

INDEX & TABLES

DESCRIPTION OF PROJECT

Introduction:

This project is aimed at detection of spam e-mails. Spam e-mails are messages randomly sent to multiple addresses by all sorts of groups, mostly lazy advertisers and criminals who wish to lead you to phishing sites.

Problem Statement:

Due to spam emails, critical/important e-mails are missed/delayed and millions of networks are compromised. Spam e-mails crash mail servers and fill up hard drives. Spam e-mails are usually theft related.

Objective:

The main objective of this project is to acknowledge user about the fake/spam e-mails and relevant e-mails. This project increases security and control.

Method:

Naïve bayes classifier is used that calculates a set of probabilities by counting the frequency and combination of values in a given dataset. It is very useful to classify emails properly. The precision and recall of this method is known to be very effective.

2. MODULES:

1. **Login**:

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

2. Search:

Through this user can input a message which is to be checked whether it is a spam or ham.

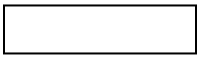
3. **Predict**:

Through this user can get knowledge whether the entered message is spam or relevant/ham.

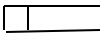
3. DATA FLOW DIAGRAM

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 :- 

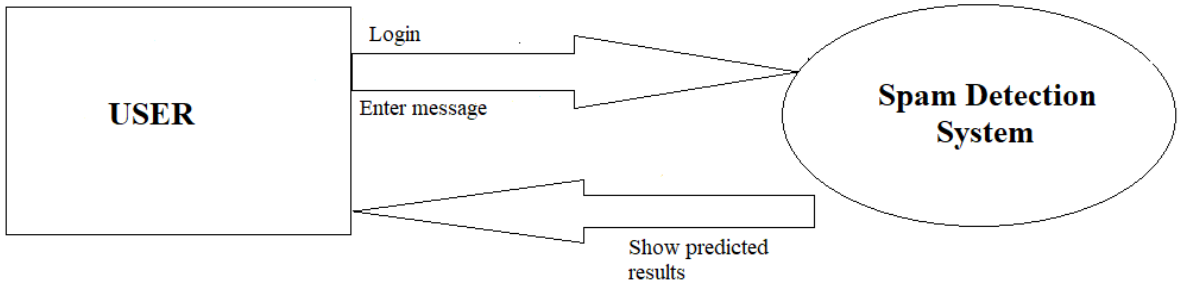
(4) An **arrow** shows the data flow :- 

The DFD can be up to several levels. The 0 level DFD states the flow of data in the system as seen from the outward in each module.

The first level DFD show more detail, about the single process of the 0 level DFD

The second level DFD can show even more details and so on.

Context Level DFD



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 .

5. Language/Libraries/Tools

Front End :

Python tkinter

Back End :

Python and Data Science

Libraries :

- ✓ tkinter
- ✓ docxpy
- ✓ os
- ✓ pyinstaller

Other S/W :

- ✓ Python3.x
- ✓ IDLE
- ✓ Anaconda (jupyter lab)