Report

Team members

Arsalan Munir (20K-1718)

Ferdeen Bakht(20K-0219)

We implemented tf-idf from scratch using just numpy and pandas and it is also running on the website using flask ,js, jquery and html. We have implemented also content filtering on cmd. We have implemented neural network on cmd using keras.

Tf-idf algorithms gives more importance to less occurred words(features) and less importance to words that have occurred more often. Words that have never occurred at all is not given any importance. This is the reason why it is so famous and widely used in software industry. Even google uses tf-idf of course along with the other algorithms.

The / route handles the login functionality. It checks if the request method is POST, retrieves the entered name and password, and stores them in the session. If successful, it redirects the user to the /home route. If the request method is GET, it renders the "login.html" template.

The /logout route logs out the user by clearing the name and password from the session.

Overall, the code performs the following steps:

Reads the necessary data from CSV files.

Computes the TF-IDF vector for item descriptions.

Calculates cosine similarities between the user's purchased items and all other items.

Recommends the top similar items based on cosine similarities.

Extracts relevant information about the recommended items.

Renders the template with the recommended item data.

The website is running on file1.py on which backend is implemented on flask. In front end if you need to evaluate the code go to templates folder and then home.html which decribes how data is send from python to js.

To run the website

first set up flask

run pre\_tf.py first it will generate small\_vector.csv

we didnot upload small\_vector.csv because its size is too large

run file1.py

the website is ready you will see login page

enter any username buyer0 ,buyer1,buyer2 ..till buyer19999

enter any password buyer0 ,buyer1,buyer2 ..till buyer19999