Brief Report

1. Introduction

The objective of this assignment is to classify tables from financial statements into categories: Income Statements, Balance Sheets, Cash Flows, Notes, and Others. This involves extracting and preprocessing data from HTML files, selecting and training a classification model, and evaluating its performance.

2. Data Extraction

I used BeautifulSoup to parse HTML files and extract tables. Each table was then converted into a pandas DataFrame for easier manipulation.

3. Data Preprocessing

The extracted tables were converted into textual format by flattening the table content into a single string. TF-IDF was used for feature extraction to convert text data into numerical features suitable for machine learning.

4. Model Selection and Training

I selected a RandomForestClassifier due to its robustness and ability to handle complex data. The data was split into training and test sets, and the model was trained on the training set.

5. Model Evaluation

The model was evaluated on the test set using accuracy and classification report metrics. The results showed a high accuracy, indicating the model's effectiveness in classifying the financial tables.

6. Conclusion

The approach demonstrated effective classification of financial statement tables with high accuracy. Future improvements could include fine-tuning the model and exploring additional features for better performance.

Explanation of the Streamlit App

1.Title and Description

The app's title and description are set using st.header and st.markdown.

2. File Uploader

The st.file_uploader widget allows users to upload an HTML file.

3. File Handling

The uploaded file is saved temporarily, and tables are extracted from it.

- **Table Extraction**: Extracted tables are converted to text and transformed using the trained TF-IDF vectorizer.
- **Prediction**: The model predicts the category of each table, and the results are displayed on the app.
- **Clean-up**: The temporary file is deleted after processing.

With this Streamlit app, users can upload HTML files containing financial tables, and the app will classify each table into one of the specified categories. This interactive approach makes it easy to demonstrate your model and its capabilities.