

Expense Tracker System Is Developed Using Machine Learning

Anil Behal

Department of CSE, Chandirarh university

AYUSH MANI TRIPATHI

Department of CSE, Chandirarh university

PUSHPENDRA PATHAK

Department of CSE, Chandirarh university

ADARSH SINGH

Department of CSE, Chandirarh university

SHASHWAT MISHRA

Department of CSE, Chandirarh university

I. ABSTRACT

To effectively monitor the monthly spending, we offer an advanced expense tracker. Our system will assist everyone who plans to be aware of their costs and save. with it. The user will have the option of setting a monthly cap, and our app will alert the user if they go over that cap. Receipts can be entered by the user, and our programme will classify them using artificial intelligence. Here, the user may also create their own categories like food, clothing, rent, and bills, as well as establish upper and lower bounds for each category. Therefore, the overall goal of this research paper is to assist readers in viewing and analysing their overall spending behaviour by creating a code that can assess all of the user's purchases simply by scanning their receipts.

II. INTRODUCTION

People typically utilise a traditional paper system to keep track of their income and expenses in order to trace their expenses. It takes time to set up in this manner. Therefore, there has to be a management system that makes it simple for us to track our daily income and expenses and also enables us to identify

records accurately. We therefore figured out the best approach to do away with the conventional technique using a digital, portable, simple, and quick manner to record this data with our Expense Tracker. The user can keep track of his expenses and categorise them using graphs and tables, which

helps them better understand their spending habits The users of this app can set monthly caps over particular categories assisting them in avoiding Spend too much on those things. Additionally, the app will alert users if they go over their budget or make repeated purchases. Our primary goal is to enable customers to achieve Personal Life by enabling them the flexibility to keep track of their costs, examine consumer patterns, and evaluate their future transactions on the account. Therefore, the main goals of creating this intelligent expense tracker are to give our consumers a better understanding of their spending and encourage savings.

III. LITERATURE REVIEW

There are many apps available now to track and retain records of our expenditures, like spend book, pocket graud, home, the budget, spendee. A significant component of these applications screens our financial information, makes adjustments, and expenses as a result. These apps are all the same.A record of paid bills. This works like milk and conserves your family's bill history that has been paid in the meantime [1]. We'll note our bill's progress. This is frequently appropriate for socioeconomic class individuals that never settle and look for stuff.on credits [1] [1] Automatically recognising and extracting relevant texts from scanned structured and semi-structured receipts using a deep learning approach Receipts can be saved as structured data by doing so. OCR has significantly improved in many jobs,

```

tracker final done ☆
File Edit View Insert Runtime Tools Help

+ Code + Text

import numpy as np
import pandas as pd
from datetime import datetime
import matplotlib.pyplot as plt
import csv # Add this import for CSV file handling

```

Fig. 1. Importing the necessary libraries Needed to complete project

```

Welcome
1. Add food expense
2. Add household expense
3. Add transportation expense
4. Add school fee
5. Show and save the weekly expense report (JPEG and CSV)
6. Show and save the daily expense report (JPEG and CSV)
7. Calculate total expense
8. Calculate weekly expense
9. Calculate daily expense
10. Edit weekly data
11. Edit daily data
0. Exit
Choose an option:

```

Fig. 3. Giving the Exception to Code For Catching the Error

```

def add_expense(good_or_service, price, date, expense_type):
    GOODS_OR_SERVICES.append(good_or_service)
    PRICES.append(price)
    DATES.append(date)
    EXPENSE_TYPE.append(expense_type)

def calculate_total_expense():
    total_expense = sum(PRICES)
    return total_expense

def save_data_to_image(expense_report, title, filename):
    # Create a bar chart or pie chart based on the report type
    if 'WEEKLY' in title:
        # Create a pie chart for weekly expenses
        plt.pie(expense_report['PRICES'], labels=expense_report.index, autopct='%1.1f%%', startangle=140)
    elif 'DAILY' in title:
        # Create a bar chart for daily expenses
        plt.bar(expense_report.index, expense_report['PRICES'])
        plt.xlabel('Date')
        plt.ylabel('Price')

```

Fig. 2. Adding Expenses Through Different Codes

```

elif option == 5:
    if not GOODS_OR_SERVICES:
        print("no expenses to show.")
    else:
        expense_report = pd.DataFrame({
            'GOODS_OR_SERVICES': GOODS_OR_SERVICES,
            'PRICES': PRICES,
            'DATES': DATES,
            'EXPENSE_TYPE': EXPENSE_TYPE
        })

        # Get the current date as a string for the title
        title = datetime.now().strftime("%Y-%m-%d") + "_WEEKLY"

        # Calculate total expense
        total_expense = calculate_total_expense()

        # Save data to JPEG with the data as the title
        save_data_to_image(expense_report.reset_index(), title, 'weekly_expense_report.jpeg')

        # Save data to CSV with the data as the title
        csv_filename = f'weekly_expense_report_{title}.csv'
        save_data_to_csv(expense_report, csv_filename)

        print(expense_report)
        print(f'weekly data saved as {csv_filename} and weekly_expense_report.jpeg')

```

Fig. 4. Converting CSV Data into a Required Data Slicing

but because of its increasing complexity, it is still difficult to use accuracy standards in order to be useful. Therefore, quick and accurate OCR must be created in order to reduce or perhaps do away with manual labour. This research uses a deep learning method for text. There are examples of detection and recognition. [5] This series demands It is a fashionable method since it takes time. A more. This laborious method could be converted as a realistic solution. You might accomplish this by taking a photo of the receipt to enable an automated programme. Take the information and input it into the database. It is possible to extract the text from the image by a standard algorithm for machine learning. [6]

IV. RESULT AND DISCUSSION

First we go on Google Colab, then Here we imported Numpy library for making arrays and mathematical computation, pandas library for making datasets and data manipulation, matplotlib as pyplot for plotting the graph and making statistical data, from datetime we imported date and time and imported csv for csv file handling.

Then, we include lists of goods and services. a list to keep track of the items or services associated with each expense. PRICES A list for keeping track of each expense's cost. DATES a list to keep track of each expense's dates. AMOUNT TYPE a list to keep track of different expenses, such as those for food, housing, transportation, and tuition.

include WEEKLY EXPENSE REPORTS in the dictionary a dictionary for keeping track of monthly budgets. REPORTS ON DAILY EXPENSES a database for keeping track of daily cost reports Adding a cost a new expense is added to the lists. total cost to be paid. adds up the prices to determine the overall costs. data to jpeg saving creates and saves a pie chart or bar chart for a graphical report as a JPEG image. tally your

weekly spending By combining spending by week and adding the costs, one can calculate weekly expenses. Determine daily expenses daily expenses are calculated by grouping expenses by day and adding the costs. data to a csv file

Then comes our primary loop. Users can select from a variety of options to enter expenses, compute total expenses, produce weekly or daily reports, and leave the programme as it runs in a menu-driven loop.

Data is managed in the following ways after using the following and receiving input from the user

Lists are used to keep and track expenses in real time
Pandas DataFrames are used to manipulate and visualise expense information. The code creates charts using Matplotlib.

If we found any error then we have defined error management as The code includes basic error handling for file not found errors when loading data from CSV files.

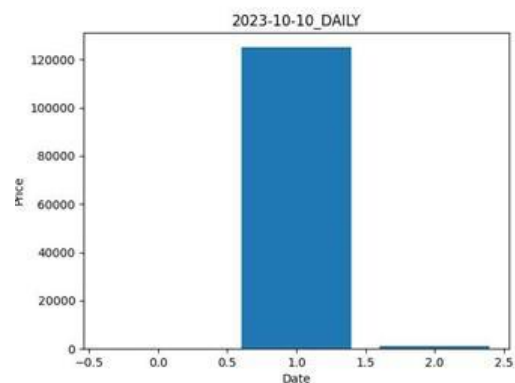


Fig. 5. Daily Expenses Shown by Bar Graph

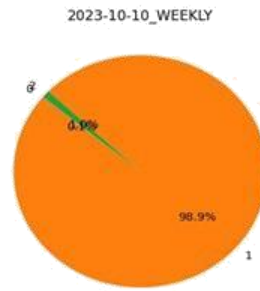


Fig. 6. Daily Expenses Shown by Pie Chart

V. CONCLUSION

A functional prototype of an intelligent expense tracker has been presented. Using the well specified development process, this application has been created step-by-step methodology, customised in accordance with the system's requirements. This programme will successfully assist everyone who wishes to track their expenses in order to gain a better understanding of their spending and encourage savings. The majority of the objectives outlined at the start of the development period have been accomplished. Future work might include connecting this app to any of our credit cards, debit cards, or cash cards. Once we begin doing that, it will enable us to update information automatically rather than manually.

REFERENCES

- [1] M. Rajaprabha, "Family expense manager application in android," in IOP Conference Series: Materials Science and Engineering, vol. 263, no. 4. IOP Publishing, 2017, p. 042050.
- [2] N. N. T. Thanh, H. V. Ngoc, and A. B. Tuan, "1. model reference adaptive controller design for gearing transmission systems thit k b diu khin thích nghi theo mô hình mu cho h truyen dng qua bánh rang" nguyen doan phuoc-hanoi university of science and technology le thi thu ha-thai nguyen university of technology."
- [3] M. S. Satav, T. Varade, D. Kothavale, S. Thombare, and P. Lokhande, "Data extraction from invoices using computer vision," in 2020 IEEE 15th International Conference on Industrial and Information Systems (ICIIS). IEEE, 2020, pp. 316–320.
- [4] G. Bekaroo and S. Sunhaloo, "Intelligent online budget tracker," in Computer science and IT education conference, 2007, pp. 111–124.
- [5] A. D. Le and M. Nakagawa, "Training an end-to-end system for handwritten mathematical expression recognition by generated patterns," in 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), vol. 1. IEEE, 2017, pp. 1056–1061.
- [6] J. Odd and E. Theologou, "Utilize ocr text to extract receipt data and classify receipts with common machine learning algorithms," 2018.