Extraction and Classification of Receipt Data with Machine Learning

Albert Iradukunda
Özyeğin University

***Abstract*—Receipt data processing is traditionally a labor-intensive task requiring significant human effort to extract and classify relevant information. This paper presents a novel pipeline for automating the extraction, classification, and clustering of receipt data, utilizing cutting-edge machine learning techniques. The pipeline extracts key information from receipt images using a pretrained LLM model, such as item names, prices, purchase dates, and locations. The extracted items are then classified into predefined categories such as groceries, electronics, and household items using K-Nearest Neighbors (KNN) and Logistic Regression. Furthermore, the system performs clustering of items and receipts using the K-Means algorithm to identify similar items and receipts. This automated approach aims to reduce manual effort, increase processing speed, and improve accuracy in receipt data management.**

***Index Terms*—KNN(K-Nearest Neighbors), DONUT (Document Understanding Transformers), Logistic Regression, Fine-tuning, K-Means, Clustering**

I. Introduction

Receipt data processing is an essential yet time-consuming task, particularly when dealing with large volumes of receipts. This often requires significant human intervention for tasks such as extracting item details, categorizing purchases, and identifying patterns across receipts. Manual processing not only consumes time but is prone to errors, making it inefficient for modern applications where scalability and accuracy are crucial.

In this paper, we propose an automated pipeline that leverages machine learning and natural language processing (NLP) to streamline the extraction and categorization of receipt data. Using a pretrained LLM model, we extract essential information such as items, prices, purchase dates, and locations from receipt images. To further enhance the process, the extracted items are classified into categories such as groceries, household items, electronics, and more, using K-Nearest Neighbors (KNN) and Logistic Regression models. Additionally, we employ the K-Means clustering algorithm to group similar items and receipts, facilitating the identification of patterns and trends across multiple receipts. The system aims to significantly reduce the need for manual labor, improve processing efficiency, and enable deeper insights into consumer spending behavior.

II. Literature review

Recent advancements in computer vision and machine learning have significantly expanded the potential of automated systems to analyze and interpret visual data. In particular, a number of studies have explored the application of these technologies for document analysis, leading to innovations in text extraction, object detection, and image classification. Techniques such as Optical Character Recognition (OCR) have shown promise in extracting text from various document types, while deep learning models have achieved notable success in identifying objects and classifying image content. Despite these developments, the specific task of classifying expenses based on receipt and invoice images remains an emerging field with unique challenges. Research in this area is still relatively sparse, with most work focusing on general document classification or text extraction rather than targeted financial classification based on transaction documents. By building on these foundational studies, this project seeks to fill a gap in the current research, offering a specialized approach to the categorization of receipts and invoices for expense tracking purposes.

III. Data Collection

To develop a reliable and effective classification system, it is crucial to gather a diverse dataset of receipt images. The assembled dataset includes 234 receipt images: 34 grocery receipts and 200 restaurant receipts. Among the grocery receipts, 20 are personal receipts collected from my own shopping, all written in Turkish and obtained from Migros Grocery Store. I included these to evaluate whether the system could effectively process non-English receipts.

The remaining receipts were sourced from publicly available receipt image datasets.

IV. Data Processing

Once the dataset was complete, the next step was to process it and extract key details such as item names, prices, dates, and shop information. This extracted data would serve as the foundation for fine-tuning the DONUT LLM model [1], generating structured JSON output from the receipts, building a classification model, and clustering the data into relevant groups.

A. Data Annotation and Data Labeling

The images were annotated in the FUNSD [2] format using an open-source image annotation tool, Banksy [3]. The Banksy annotation tool generates outputs for Named Entity Recognition (NER), Named Entity Linking (NEL), and bounding box regions on the image. The NER labels each text region

as “Question”, “Answer”, or “Other”. The NEL task involves linking the text labeled as “Question” to the corresponding “Answer” text. Finally, bounding boxes are drawn around the text regions to define their spatial locations.

Once the annotations were completed, the next step was to process the annotated data into a structured JSON format containing key information such as shop name, date, time, and purchased items. Additionally, all extracted items were compiled into a single CSV file. In the end, the process generated: 1. A processed JSON file for each receipt. 2. A CSV file with columns for image path and the extracted receipt data (shop name, date, time, and purchased items). 3. A CSV file listing all extracted items with columns for item name, quantity, unit price, and total price.

During the generation of the item CSV, each item was categorized into one of the following categories: Produce, Dairy (e.g., milk, cheese), Beverages (non-alcoholic), Health and Personal Care (e.g., soaps, shampoo), Condiments and Spices, Snacks and Sweets, Grains and Bakery, Chicken and Meat, Alcoholic Beverages, Miscellaneous.

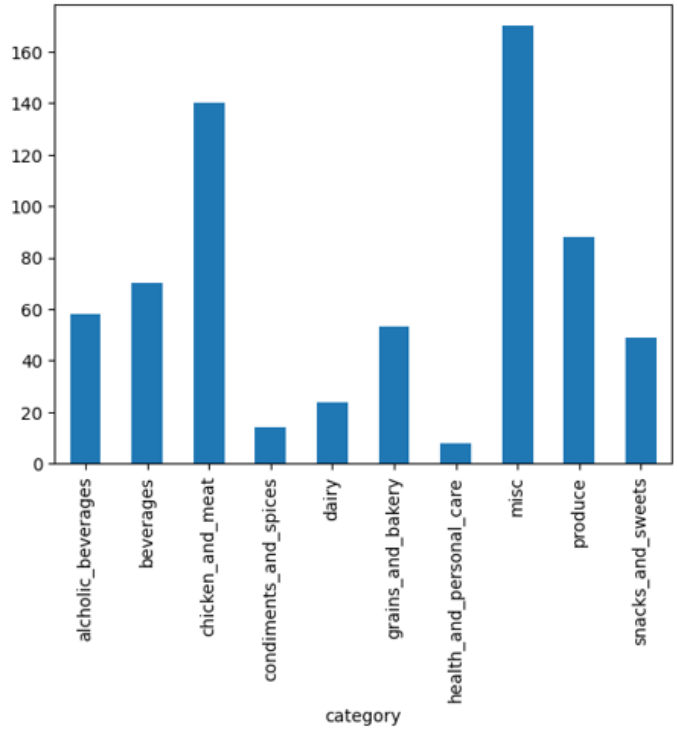

Fig. 1: Receipts Items classified into different categories.

B. Text Extraction with OCR

Plain text extraction from the receipt images was performed using pytesseract [4], an open-source OCR engine. Pytesseract processed each image to extract raw text, which was then added to the corresponding receipt entry in the CSV file. This plain text data would later serve as input for clustering receipts into different groups based on textual similarity.

V. Fine-tuning the DONUT model for Data Extraction

To enhance the performance of receipt data extraction, the DONUT [1] (Document Understanding Transformer) model was fine-tuned using a set of annotated receipt images. The model was fine-tuned using the Hugging Face Transformers library [5]. The fine-tuning process involved feeding the model pairs of receipt images and the corresponding desired JSON outputs. This approach enabled the model to learn the specific structure of receipts and the relationships between key fields such as shop name, date, time, and purchased items.

The goal of fine-tuning was to enable the model to generate structured JSON data directly from an input image. After the model was fine-tuned, it was evaluated by calculating the accuracy of its predictions for each key in the JSON output.

VI. Classification of Items

The classification of receipt items was conducted using two machine learning models: Logistic Regression and K-Nearest Neighbors (KNN). Both models were evaluated using key performance metrics, including accuracy, precision, recall, and F1-score. The features used were: vectorized item name, unit price, quantity and total price.

A. Logistic Regression

The Logistic Regression model was trained for 50 epochs with a learning rate of 0.01 and 2 hidden layers. Despite its simplicity [6], the model provided a baseline for classification performance. The results are summarized below:

Number of Epochs: 50 Learning Rate: 0.01 Number of Hidden Layers: 2 Accuracy: 48.2

B. K-Nearest Neighbors

A KNN model was also employed for item classification. The hyperparameter K was tuned by cross-validation, and the best results were achieved when

K=5, as shown in Figure 2. The classification results for KNN are as follows:

Accuracy: 55.6Precision: 59.6Recall: 55.6F1-Score: 54.24

VII. Clustering Data Into Groups

Clustering was performed to identify patterns [6] and group similar items as well as receipts based on their textual content. The K-Means algorithm [7] was chosen due to its efficiency in partitioning datasets into clusters. The optimal number of clusters was determined using the elbow method.

A. Clustering Items

All extracted items were clustered using K-Means, aiming to group similar products together. To find the optimal number of clusters, the elbow method was applied, which involves plotting the total within-cluster sum of squares (WCSS) against the number of clusters. The elbow point was observed at $K = 40$ clusters, as shown in Figure 4.

Once the clusters were formed, a method for finding similar items was developed by calculating the cluster to which an

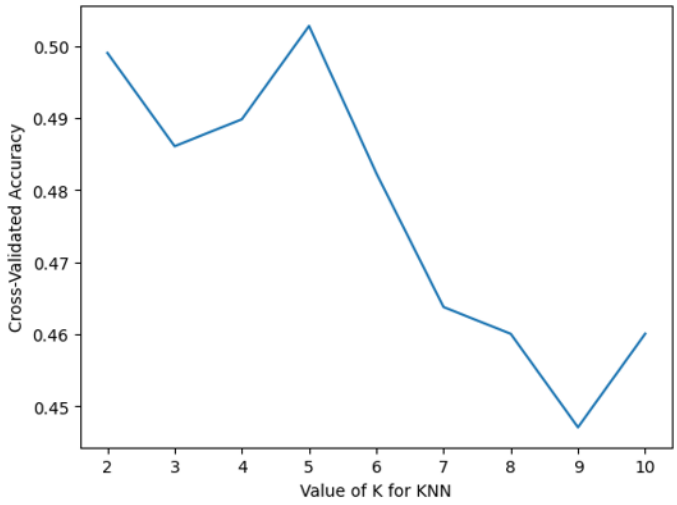

Fig. 2: Cross-validated accuracy for different values of K in KNN.

item belongs and retrieving other items from the same cluster. This approach enables the identification of products that are often purchased together or have similar characteristics [6].

Chosen K: 40 clusters

Method:

- Calculate the cluster of the item using its vectorized representation.
- Retrieve other items within the same cluster to find similar products.

B. Clustering Receipts

The receipt images were also clustered based on the textual information extracted from them. The process involved extracting text from each receipt using PyTesseract, vectorizing the text with Sklearn [8] library, and applying the K-Means algorithm to group similar receipts. The elbow method was again used to find the optimal number of clusters, which was determined to be $K = 11$.

This clustering of receipts allows the system to identify receipts with similar content, enabling functionality such as finding receipts from the same store or with similar purchases.

Chosen K: 11 clusters

Steps:

- Extract text from receipts using OCR.
- Vectorize the extracted text using Sklearn library.
- Apply K-Means clustering on the vectorized texts.
- For a given receipt (image or text), retrieve similar receipts by finding the cluster it belongs to and fetching other receipts from that cluster.

VIII. Results

The results of the pipeline show successful extraction, classification, and clustering of receipt data. The fine-tuned DONUT model accurately extracted key details such as items, prices, and dates. The Logistic Regression model achieved an accuracy of 48.2

Item clustering, using the K-Means algorithm, achieved an optimal number of 40 clusters, grouping similar items effectively. Receipt clustering with K-Means determined the optimal number of clusters to be 11, grouping receipts by similar content such as store names and types of purchases.

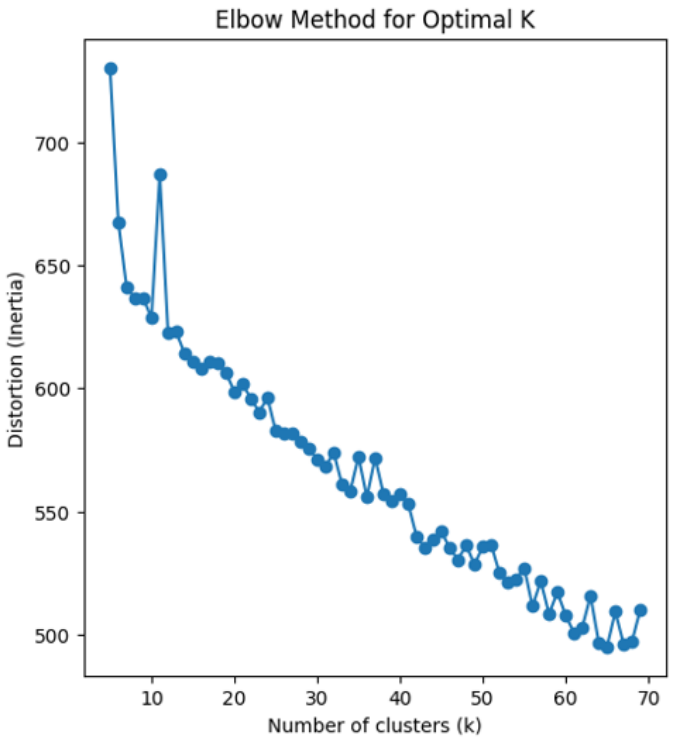

Fig. 3: Elbow method for determining the optimal number of clusters for items.

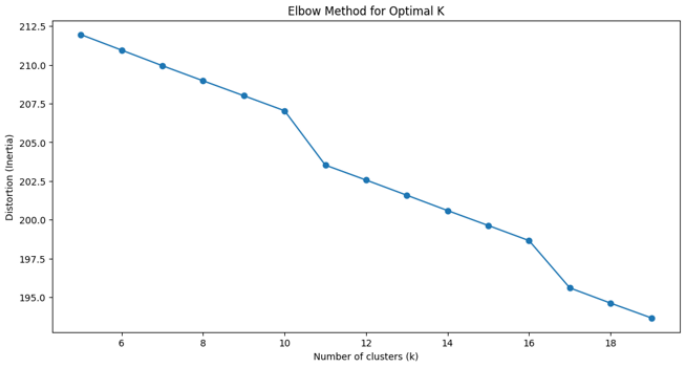

Fig. 4: Elbow method for determining the optimal number of clusters for Receipts.

IX. Conclusion

This paper presents an automated pipeline for receipt data extraction, classification, and clustering. The system demonstrates efficient processing with the DONUT model, KNN, and K-Means clustering, significantly reducing manual effort.

While classification accuracy can be improved, the approach offers a scalable solution for managing receipt data, with potential applications in expense tracking and retail analytics. Future work will focus on refining models and exploring advanced clustering methods to enhance performance.

REFERENCES

[1] G. Kim, Y. Kim, M. Seo, W. I. Cho, T. Lee, and J. Kang, "Donut: Document understanding transformer without ocr," *arXiv preprint arXiv:2111.15664*, 2022.

[2] T. J. Jaume, G. Ekenel H. K., "Funsd: A dataset for form understanding in noisy scanned documents." *Int. Conf. on Document Analysis and Recognition Workshops, pp.*, 2019.

[3] "Banksy annotation tool," *https://github.com/AboutGoods/Banksy-annotation-tool*.

[4] "pytesseract," https://pypi.org/project/pytesseract/.

[5] "Transformers, hugging face," https://huggingface.co/docs/transformers/en/index.

[6] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.

[7] A. K. Jain, "Data clustering: 50 years beyond k-means," *Pattern recognition letters*, vol. 31, no. 8, pp. 651–666, 2010.

[8] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in python," *Journal of machine learning research*, vol. 12, no. Oct, pp. 2825–2830, 2011.