W266 - Merchant Code Categorization using Machine Learning

Gajarajan Nagarajan

Fall 2020

Abstract:

Financial transaction data is extremely valuable, but it is underutilized as it is messy due to inconsistent, incomplete, and mixed with irrelevant information, making it difficult to find context and use the data effectively. Transaction data contains cryptic information which is highly abbreviated and contains lots of numbers and text that lack context, making it difficult to interpret. It also lacks categorization information as there is no clear grouping or sorting of the data, making it hard for businesses to organize, analyze, and effectively use.

The goal of this project is to use ML models to clean and categorize data and unlock insights from transaction data to make better business decisions and improve handling of future transaction data. We will make use of level neural networks and natural language processing and build an ML model to be able to clean the messy transaction data by category, name, and location to get insight into trends over time, consumer spending analysis.

1. Introduction

According to Gartner's report, 40% of businesses fail to achieve their business targets because of poor data quality issues. The importance of utilizing high-quality data for data analysis is realized by many data scientists, and so it is reported that they spend about 80% of their time on data cleaning and preparation. This means that they spend more time on pre-analysis processes, rather than focusing on extracting meaningful insights.

Fintech companies want to get a better understanding of their customers behavior to provide customers more engaging experience and drive long term growth. In the transaction records the merchant names and categories are not standardized. So, the clients cannot analyze the spend or issues by Category or Merchant name. Further they need to clean up the data themselves to provide a more engaging experience to their customers.

2. Problem Statement:

Transaction Data is messy and inconsistently labeled and it is hard to draw insights. FinTech and large global corporations spend significant efforts within their data analytics team to perform 3-way match and various other data cleansing techniques to manage and cleanse/categorize spend categories to meaningful, usable information.

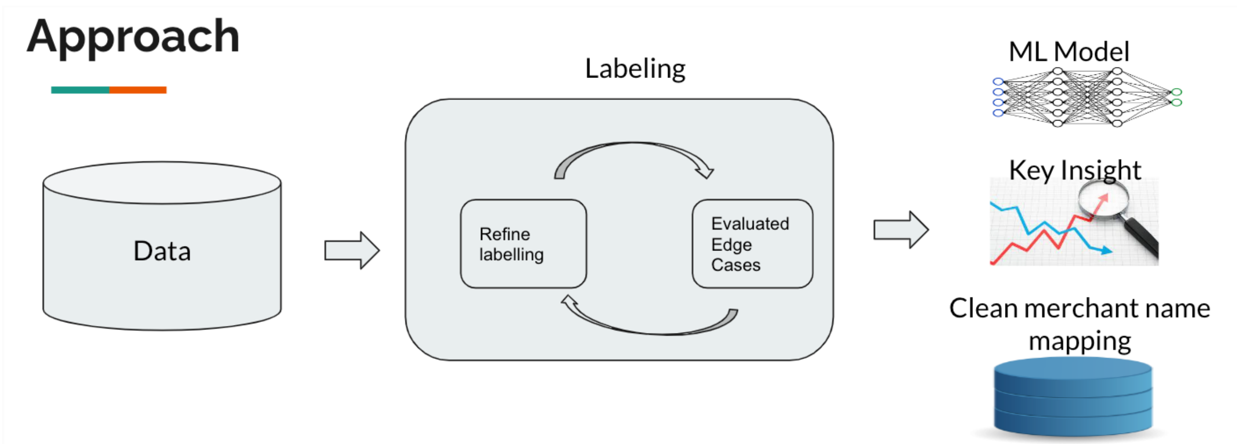
Goals: by leveraging right Machine Learning model, we would like to unlock insights from transaction data by finding clean merchant name, spend category and logo to

- Make better business decisions by understanding trends in data.
- Improve handling of future transaction data

3. Target Audience

- Credit Card Companies - Wells Fargo, Citi, Capital One etc.
- FinTech Companies - Sofi, Greenlight, chime etc.
- Corporate Spend Analytics for FP&A Departments for large global Companies
- Procurement and Internal Audit teams for indirect spend analysis & 3-way match for merchant categorization.
- SaaS Vendors for segment market data analysis

4. Approach



- Split using ALL data in transaction table to categorize 'Category' field in data.
- Create the basic token pattern and obtain text & numeric data.
- Instantiate nested pipeline.
- Use GridSearchCV to find parameters which result in highest accuracy score.
- Experiment with various models including NLP Spacy
- Fit to the training data then compute accuracy.
- Execute with test data using predictions from model in specified folder.

5. Methodology

1. Business Problem Assessment
2. Mapping to ML Problem
3. Exploratory Data Analysis
4. Baseline Model
5. Designing Advance Feature
6. Model Building
7. Deployment
8. Conclusion
9. Future Works

6. Datasets

Merchant Category Data:

- 1K Merchant Category public Data from MasterCard and Visa Networks
- Link:https://www.investopedia.com/terms/m/merchant-category-codes-mcc.asp

Merchant Transactional Data:

- Training Dataset :
- Source: Kaggle
- Size : 5k records
- Link : https://www.kaggle.com/datasets/kaggleay99/transdata

Test Dataset

- Source : Kaggle and OpenData
- Size : 100k Records
- Link : https://www.kaggle.com/datasets/kaggleay99/transdata

Merchant Category Data

- 1K Merchant Category public Data from MasterCard and Visa Networks

- mcc : Merchant Category code
- mcc_name: Merchant category name
- Gen_name: General category

7. Exploratory Data Analysis

a. Merchant Category information is standard across the globe and is used as reference to classify and group merchant category and service category.

	mcc	mcc_name	gen_name
0	11	COMMERCE BANK ODP P	agricultural_services
1	701	POSTAGE TRANSACTION CHARGE A	agricultural_services
2	742	Veterinary Services V, M	agricultural_services
3	763	Agricultural Cooperatives V, M	agricultural_services
4	780	Horticultural and Landscaping Services V, M	agricultural_services
5	1520	General Contractors–Residential and Commercia...	contracted_services
6	1711	Air Conditioning, Heating and Plumbing Contra...	contracted_services
7	1731	Electrical Contractors V, M	contracted_services
8	1740	Insulation, Masonry, Plastering, Stonework an...	contracted_services
9	1750	Carpentry Contractors V, M	contracted_services
10	1761	Roofing and Siding, Sheet Metal Work Contract...	contracted_services
11	1771	Concrete Work Contractors V, M	contracted_services
12	1799	Contractors, Special Trade Contractors–not el...	contracted_services
13	2741	Miscellaneous Publishing and Printing V, M	contracted_services
14	2791	Typesetting, Plate Making and Related Service...	contracted_services
15	2842	Sanitation, Polishing and Specialty Cleaning ...	contracted_services
16	3001	American Airlines V, M AMERICAN AIR (V) AMERI...	airlines
17	3003	Eurofly V, M EUROFLY AIR (V) EUROFLY (M)	airlines
18	3005	British Airways V, M BRITISH AWYS (V) BRITISH...	airlines
19	3007	Air France V, M AIR FRANCE (V) AIR FRAN (M)	airlines
20	3009	Air Canada V, M AIR CANADA (V) AIR CAN (M)	airlines
21	3011	Aeroflot V, M AEROFLOT	airlines
22	3013	Alitalia V, M ALITALIA	airlines
23	3015	Swiss International Air Lines V, M SWISSINTAI...	airlines
24	3017	South African Airways V, M SAA AIRWAYS (V) SA...	airlines

b. Transactional Data provides information on transaction dataset

mcc : Merchant Category code

- mid: Merchant ID
- auth_merch_name: String that identifies the merchant's name.
- auth_amt : Transaction amount

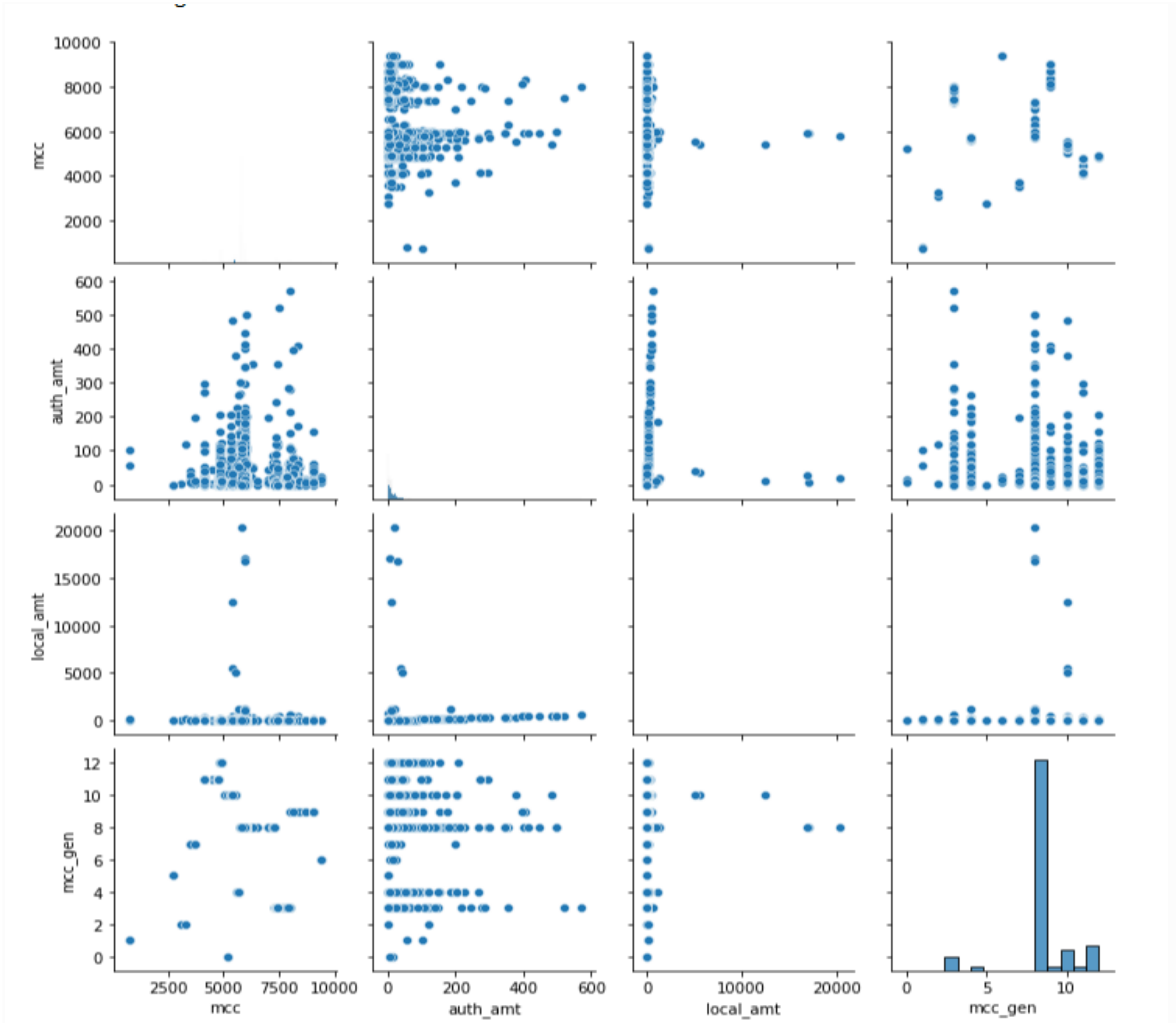
	auth_ts	mcc	mid	auth_merch_name	auth_amt	local_amt
0	2021-08-03 05:15:59.000	5812	4445028928044	TST* THE BLUEBERRY MUF PLYMOUTH MA	3.41	3.41
1	2021-08-03 05:15:59.000	5818	160146000762203	Blink amzn.com/bill WA	3.00	3.00
2	2021-08-03 05:16:00.000	5942	235251000762203	AMZN Mktp US Amzn.com/bill WA	31.97	31.97
3	2021-08-03 05:16:00.000	5814	385106000000000	MCDONALD'S F103 ANNAPOLIS MD	8.88	8.88
4	2021-08-03 05:16:00.000	5945	527021000203861	Oculus Menlo Park CA	0.00	0.00

Combined merchant category name and general category data with transaction data to explore the amount spent per category.

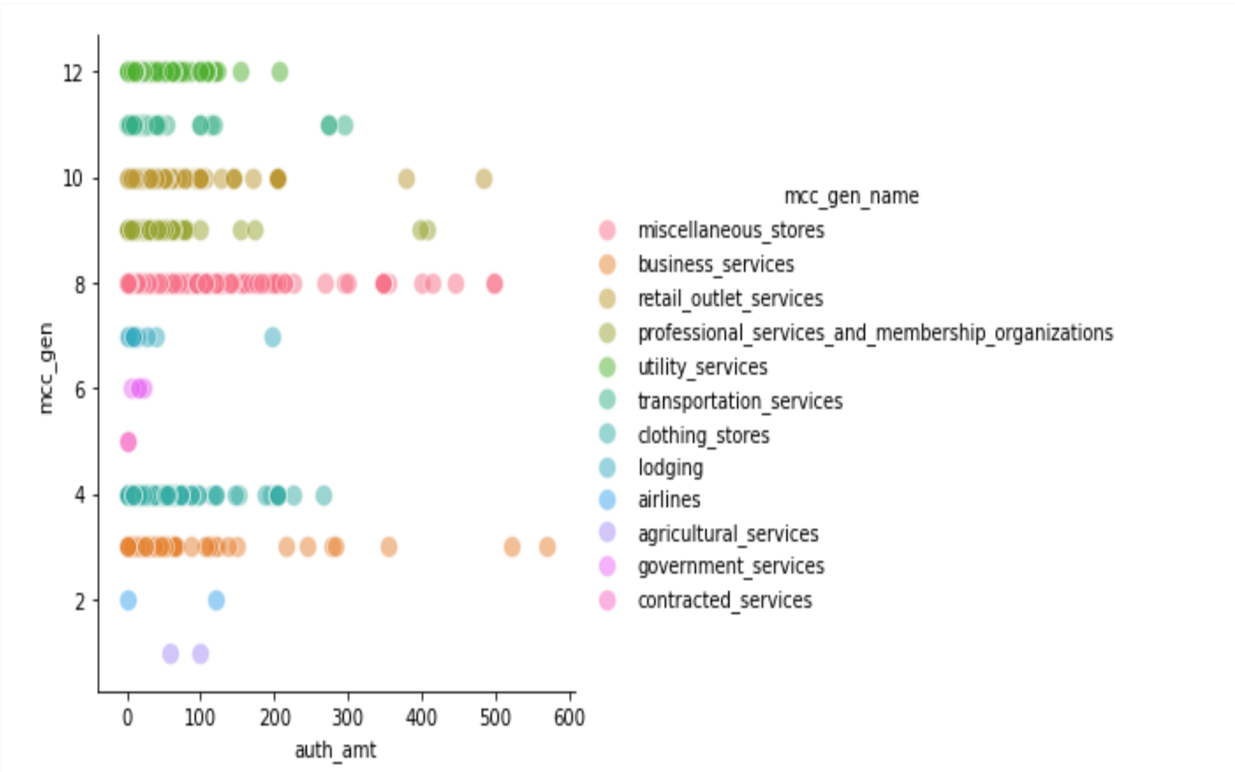
	auth_ts	mcc	mid	auth_merch_name	auth_amt	local_amt	mcc_gen	mcc_gen_name	mcc_name
0	2021-08-03 05:15:59.000	5812	4445028928044	TST* THE BLUEBERRY MUF PLYMOUTH MA	3.41	3.41	8	miscellaneous_stores	Eating Places and Restaurants V, M
1	2021-08-03 05:15:59.000	5818	160146000762203	Blink amzn.com/bill WA	3.00	3.00	8	miscellaneous_stores	Digital Goods: Large Digital Goods Merchant (...
2	2021-08-03 05:16:00.000	5942	235251000762203	AMZN Mktp US Amzn.com/bill WA	31.97	31.97	8	miscellaneous_stores	Book Stores V, M
3	2021-08-03 05:16:00.000	5814	385106000000000	MCDONALD'S F103 ANNAPOLIS MD	8.88	8.88	8	miscellaneous_stores	Fast Food Restaurants V, M
4	2021-08-03 05:16:00.000	5945	527021000203861	Oculus Menlo Park CA	0.00	0.00	8	miscellaneous_stores	Game, Toy and Hobby Shops V, M

Following activities were performed on the dataset to help get adequate pre-processing steps completed to prepare data for ML training.

a. Pre-processing
b. Target Variable Distribution Assessment
c. Categorical Features Analysis
d. Merchant Category Mapping
e. Transaction Mapping
f. Plotting Merchant category on Train and Test datasets
g. Drop Columns that are not relevant.
h. Split Train and Test Data
i. Clean Merchant Data - remove punctuations, special characters etc.
j. Breakdown merchant name by Country, State, City



Graph1: Amount spent by general category & Amount spend by mcc



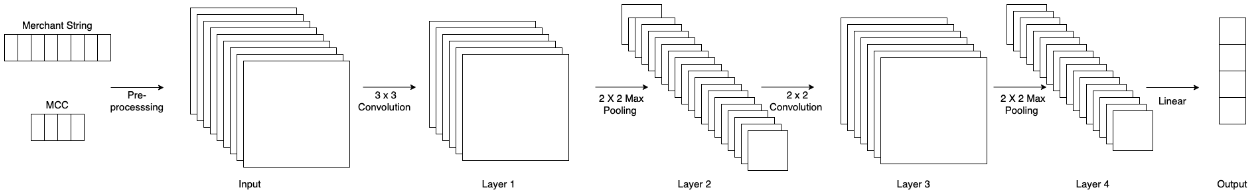
There is more spending in Misc stores/ business services. Government and contracted services show less spending .

```
[ ]: LGB_BO = BayesianOptimization(LGBM_CV, {
         'min_split_gain': (0, 1),
         'subsample': (0, 1) ,
         'min_child_samples': (10, 200),
         'colsample_bytree': (0, 1),
         'reg_alpha': (0, 1),
         'reg_lambda': (0, 1),
         'max_depth': (4, 10),
         'num_leaves': (5, 200),
         'n_estimators' : (10, 750)
         })
```

```
[ ]: start_time = time.time()
     with warnings.catch_warnings():
         warnings.filterwarnings('ignore')
         LGB_BO.maximize(init_points=2, n_iter=30, acq='ei', xi=0.0)

     print("Time taken", time.time()-start_time)
     print('-'*130)
     print('Final Results')
     print('Maximum  value: %f' % LGB_BO.max['target'])
     print('Best  parameters: ', LGB_BO.max['params'])
```

```
|   iter    |  target   | colsam... | max_depth | min_ch... | min_sp... | n_esti... | num_le... | reg_alpha | reg_la... | subsample |
-----------------------------------------------------------------------------------------------------------------------------------
Fold: 0
Training until validation scores don't improve for 200 rounds.
Did not meet early stopping. Best iteration is:
[2]     training's binary_logloss: 0.0567351    valid_1's binary_logloss: 0.0554363
Fold: 1
Training until validation scores don't improve for 200 rounds.
Did not meet early stopping. Best iteration is:
[2]     training's binary_logloss: 0.0565075    valid_1's binary_logloss: 0.0558807
Fold: 2
Training until validation scores don't improve for 200 rounds.
Did not meet early stopping. Best iteration is:
[2]     training's binary_logloss: 0.0550725    valid_1's binary_logloss: 0.0577161
|  1        | -0.1239   |  0.125    |  4.001    |  101.8    |  0.9643   |  163.5    |  155.4    |  0.7511   |  0.3077   |  0.2372   |
Fold: 0
Training until validation scores don't improve for 200 rounds.
Early stopping, best iteration is:
[1]     training's binary_logloss: 0.0597671    valid_1's binary_logloss: 0.0582452
Fold: 1
Training until validation scores don't improve for 200 rounds.
Early stopping, best iteration is:
[1]     training's binary_logloss: 0.0594563    valid_1's binary_logloss: 0.0588356
Fold: 2
Training until validation scores don't improve for 200 rounds.
Early stopping, best iteration is:
[1]     training's binary_logloss: 0.0580696    valid_1's binary_logloss: 0.060845
|  2        | -0.1263   |  0.02276  |  4.623    |  80.59    |  0.7206   |  543.5    |  84.03    |  0.738    |  0.7371   |  0.6018   |
Fold: 0
```



80/20 Train-Test Split
Number of Epochs: 6
Batch Size: 64

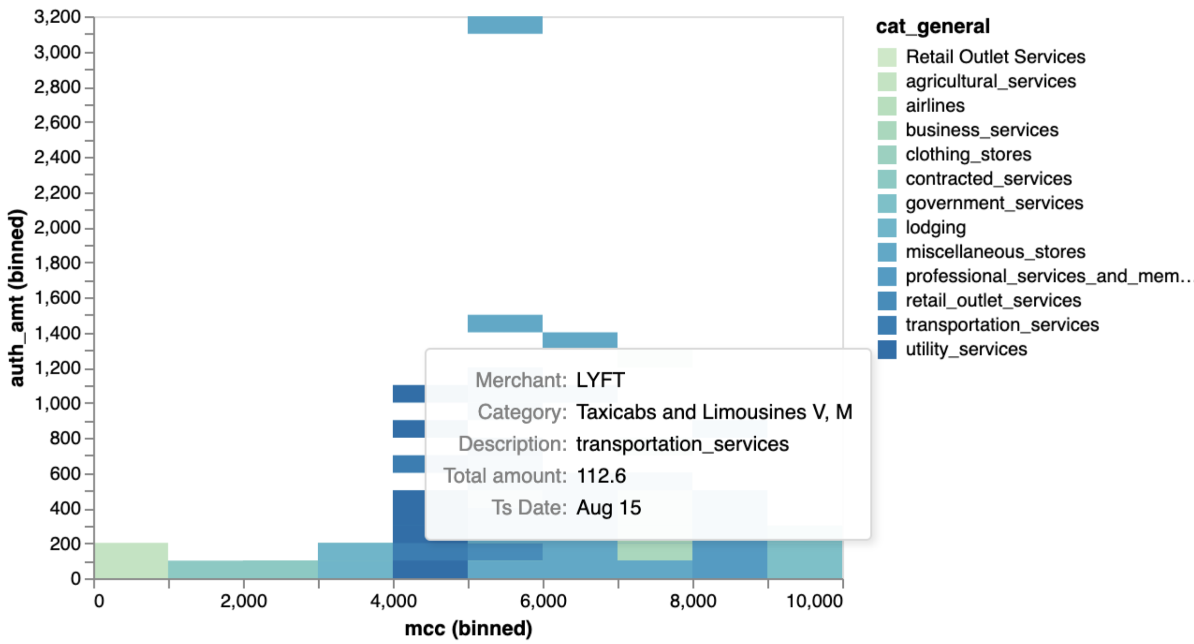
Train Accuracy: 83%
Test Accuracy: 82%

PREPROCESSING: Expanded MCC and Merchant String

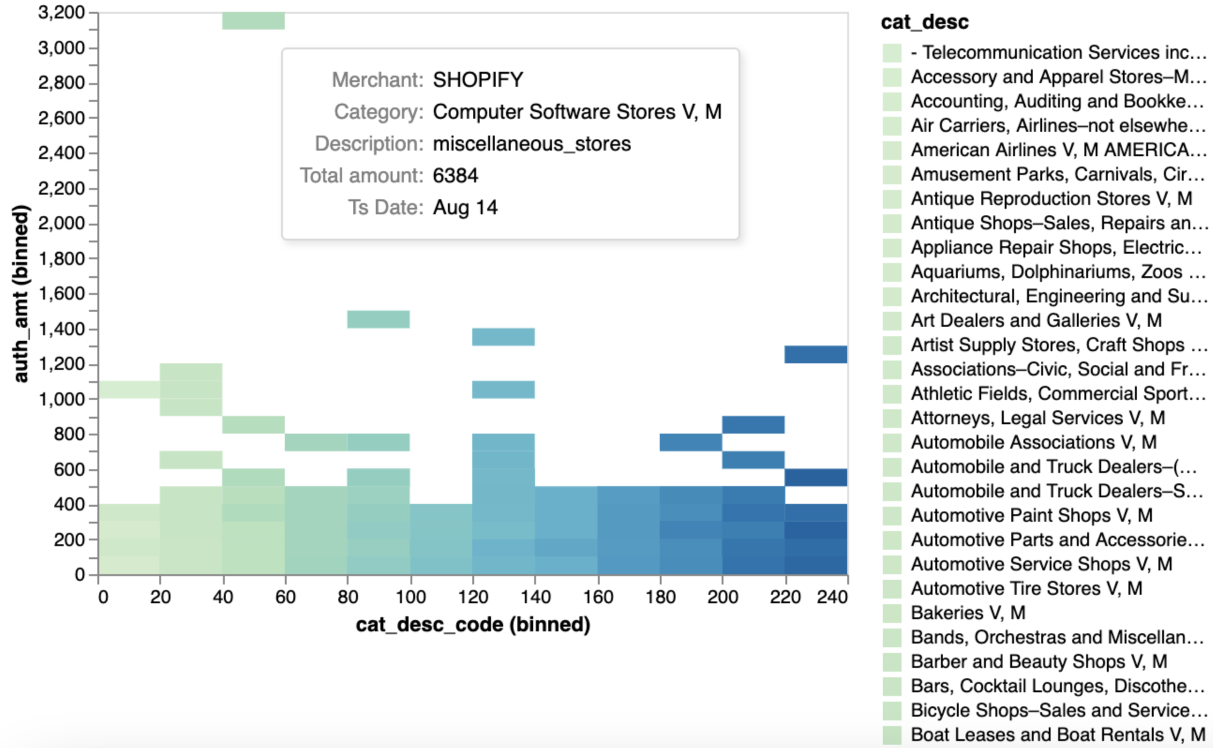
- MCC → 20 characters
- Merchant String → 40 characters

INPUT: MCC+Merchant String → 60X49 vector (49 unique characters)
ARCHITECTURE: 4 layers CNN
OUTPUT: 36X1 vector (confidence for each merchant category)

Output Analysis is as follows:



Transaction by General High level category



Transaction by specific categories

As you can see, merchant category model training is yielding better results.

```python
#load model from file

#Test your text
test_text = 'AMCON bill.com'
doc = prdnlp(test_text)
for ent in doc.ents:
    print(ent.text, ent.start_char, ent.end_char, ent.label_)

test_text = " .COM/BILL   AMZN      866-712-7753  CA"
doc = prdnlp(test_text)
for ent in doc.ents:
    print(ent.text, ent.start_char, ent.end_char, ent.label_)
```

```
AMCON 0 5 MN
AMZN 13 17 MN
```

```python
nlp1 = spacy.load("spacymodel")

#Test your text
test_text = "AMZON. bill"
doc = nlp1(test_text)
for ent in doc.ents:
    print(ent.text, ent.start_char, ent.end_char, ent.label_)
```

```
AMZON 0 5 MN
```

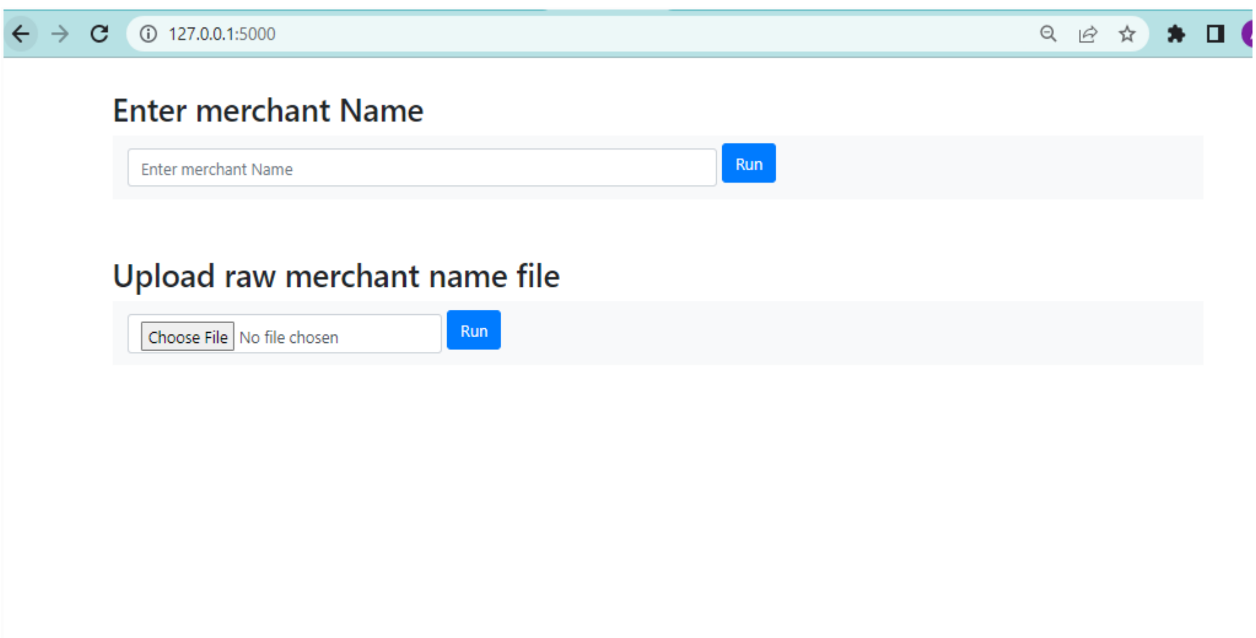
```python
TRAIN_DATA = [
('Amazon co ca', {'entities': [(0, 6, 'MN')]}),
('AMZN Mktp US', {'entities': [(0, 4, 'MN')]}),
('AMZNMKTPLACE AMAZON CO', {'entities': [(13, 19, 'MN')]}),
('APPLE COM BILL', {'entities': [(0, 5, 'MN')]}),
('BOOKING COM New York City', {'entities': [(0, 7, 'MN')]}),
('STARBUCKS Vancouver', {'entities': [(0, 9, 'MN')]}),
```

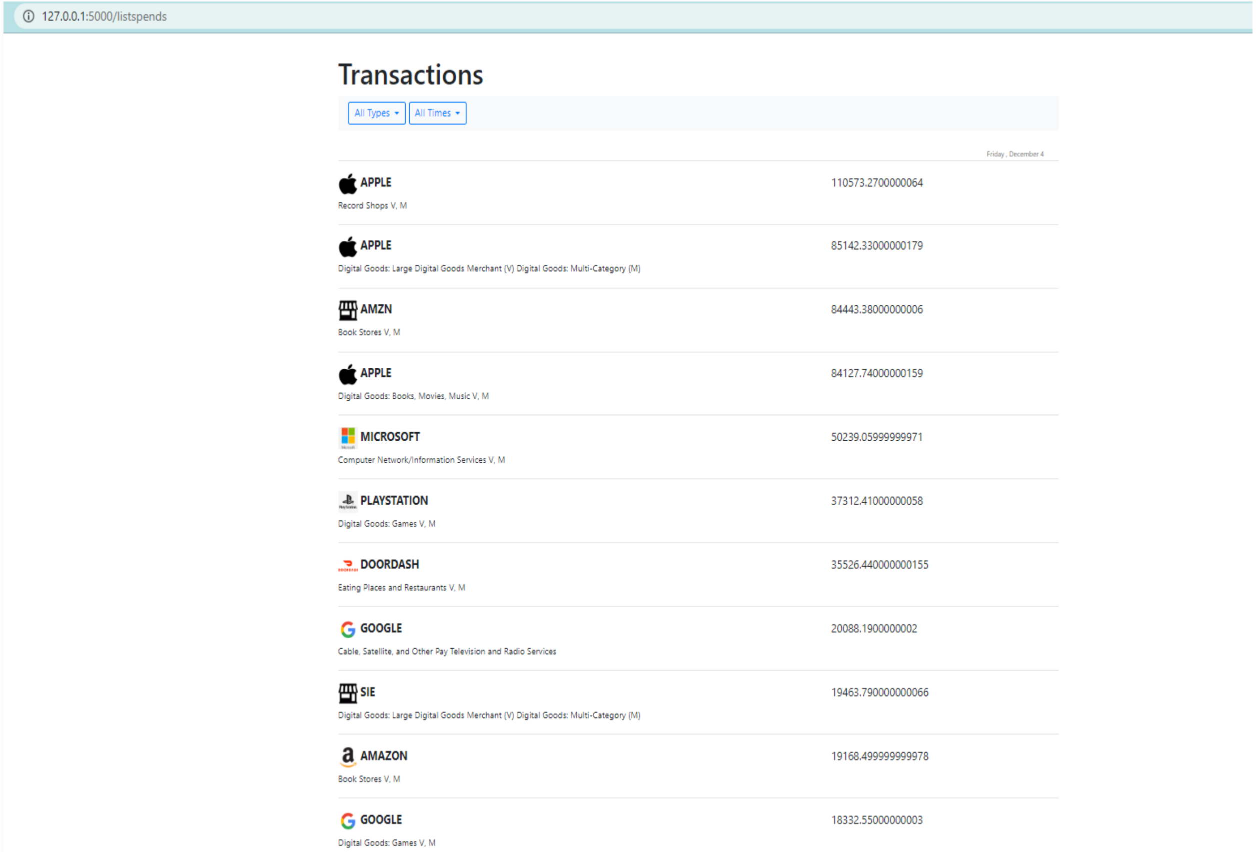
```python
prdnlp = train_spacy(TRAIN_DATA, 20)
```

Result:

	auth_ts	mcc	mid	auth_merch_name	auth_amt	local_amt	cleaned_mname_1	mname	state	city	ratio	similar_name
0	2021-08-29 08:20:50.000	7999	188418000053360	SQ *PG POOL Mount Rainier MD	1.00	1.00	sq pg pool mount rainier md	sq pg pool mount rainier	md	NaN	86	sq sidelines
1	2021-08-14 01:41:27.000	5815	112137000108778	APPLE.COM/BILL 866-712-7753 CA	12.83	12.83	apple ca	apple	ca	NaN	100	apple
2	2021-08-14 01:45:28.000	5818	342475000144509	SIE*PLAYSTATIONNETWORK 877-971-7669 CA	4.97	4.97	sie playstationnetwork ca	sie playstationnetwork	ca	NaN	90	playstation
3	2021-08-03 06:14:58.000	5815	112137000108778	APPLE.COM/BILL 866-712-7753 CA	0.00	0.00	apple ca	apple	ca	NaN	100	apple
4	2021-08-29 08:14:41.000	5735	311204598883	APPLE.COM/BILL www.apple.com CA	9.99	9.99	apple apple com ca	apple apple com	ca	NaN	90	apple
5	2021-09-09 08:48:58.000	5942	784959000762203	Amazon.com Amzn.com/bill WA	2.02	2.02	amazon com amzn wa	amazon com amzn	wa	NaN	90	amazon

UI





References:

1. Kaggle Dataset - Bank transaction data https://www.kaggle.com/code/kerneler/starter-bank-transaction-data-7e62c9d2-a/data
2. PClean by MIT https://news.mit.edu/2021/system-cleans-messy-data-tables-automatically-0511
3. Data Cleansing Tools in Azure Machine Learning https://techcommunity.microsoft.com/t5/azure-developer-community-blog/data-cleansing-tools-in-azure-machine-learning/ba-p/336536
4. Spend Category Guide **https://law.yale.edu/most-commonly-used-spend-categories**
5. Merchant Category Codes, Definitions, Purpose and Examples **https://www.investopedia.com/terms/m/merchant-category-codes-mcc.asp**
6. Visualizing spending behaviors thru open banking and GIS https://towardsdatascience.com/visualising-spending-behaviour-through-open-banking-and-gis-9e7045674538
7. Merchant Category Identification Using Credit Card Transactions https://doi.org/10.48550/arXiv.2011.02602
8. Elo Merchant Category Recommendation — An Machine Learning Case Study **https://towardsdatascience.com/elo-merchant-category-recommendation-a-case-study-33e84b8465c7**