Flight to Finance: Using Aircraft Design Techniques for Mixed Data Multi-class NLP

Abstract

A summary of the deliverables for the Challenge are as follows:

- **Deliverable 1:** Description of approach, satisfied by this document
- **Deliverable 2:** Categorized evaluation dataset, satisfied by "*categorized_eval_data.xlsx*"
- **Deliverable 3:** Commented Code, satisfied by **.py* and **.jmp* files in top level directory.

By categorizing transactions, Wells Fargo can help customers identify frequent purchases and subscriptions, sort their income and activity liability with higher accuracy, and reduce credit risks. While in the past transaction categorization was done manually, this process is now largely automated. Therein lies the **Challenge Objective**: *utilize Machine Learning to predict which category a transaction will fall into, given the description of the transaction.*

The **Challenge Objective** is met by creating a Solution that uses a novel combination of existing machine learning and/or natural language processing concepts. This process is summarized in the Solution process diagram on the following page.

The novelty of this approach is in the following aspects:

1. Feature selection based on underlying data (Cleaning the Data) and aerospace techniques
2. Class weighting based on unbalanced data
3. Training sub-models on each feature and combining into a metamodel
4. Optimizing the metamodel using the Auto_ViML python library

Specifically, the strongest strength is in the creation of a metamodel. This is a technique commonly used in aircraft conceptual design. In this stage of the design process, engineers rely on various analysis tools with greatly varied outputs and extremely long run times. In order to perform a large design space exploration, a model is trained on each tool and then integrated into a higher level metamodel controlled by the environment wrapper. Doing so greatly decreased runtime at a minimal cost to accuracy. Furthermore, the approach is flexible in that a new sub-model can be trained if a new tool (or in the case of the **Challenge,** a new feature) is added.

Interestingly the **Challenge Judging Criteria** does not make any mention of accuracy, but this report discusses the various strength and limitations of the Solution. Furthermore, these strength and limitations are extended to a discussion of real-world applications.

This report also mentions an idealized solution, making use of existing government registries. While simpler, this approach was not pursued since it did not meet the **Challenge Requirements**.

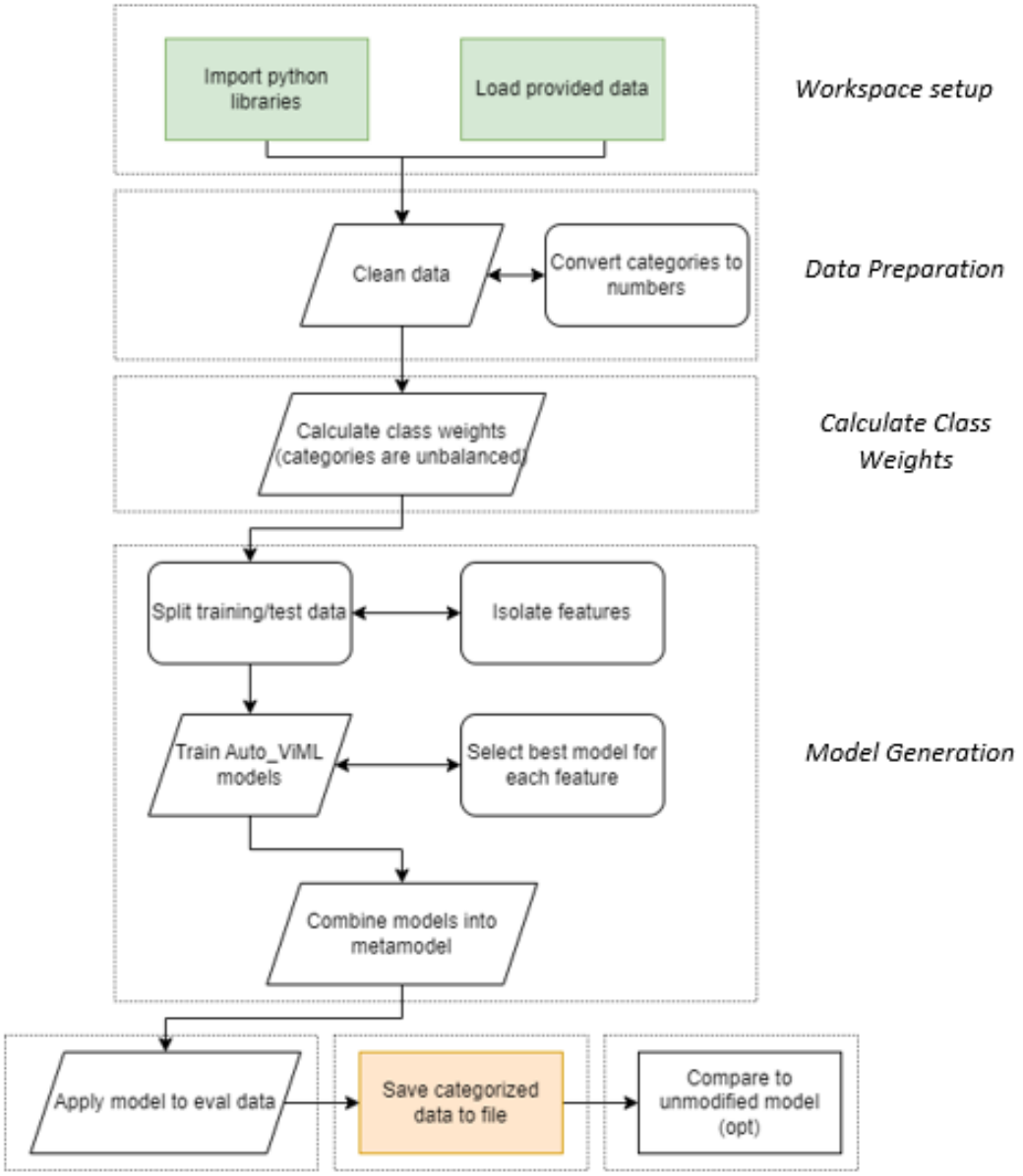

Figure 1: Solution process diagram

Table of Contents

Introduction

If you have used a debit or credit card in the past few years, it is likely that it came with an expense tracking service, categorizing your transactions. This "spend analyzer" [1] feature helps customers know what they spend their money on in order to keep their balances in check.

Simply put, the value proposition is that by categorizing transactions and building better customer engagement tools, Wells Fargo can help customers identify frequent purchases and subscriptions, sort their income and activity liability with higher accuracy, and reduce credit risks.

In the past, transaction categorization was done by manually using the transaction description and associated information. With the advent of more powerful computing technology like Natural Language Processing, transaction categorization is largely automated. Therein lies the **Challenge Objective**: *utilize Machine Learning to predict which category a transaction will fall into, given the description of the transaction.*

While automatic transaction categorization algorithms are already in use by companies such as Discover and Wells Fargo, the processes are proprietary due to their applications in anti-money laundering programs **Error! Reference source not found.**. As such, the **Challenge Objective** will be met by creating a Solution that uses a novel combination of existing machine learning and/or natural language processing concepts, or a newly developed method.

Approach Overview

Having a structured approach to data analysis is crucial to not only producing results, but also in generating an understanding of underlying processes. "Every story has already been told" is a phrase often repeated by writers [2] and a similar concept applies to data analysts- every problem might be slightly different, but still contains the same core elements.

The first step in a structured approach to data analysis is a set of initial data explorations. The goal of these explorations is to identify core elements of the data set and to create/validate assumptions of the problem.

From the initial data explorations, we can then derive key features (i.e., what aspects of the data set drive a given result). These features form the basis for our search for researching past approaches. Of course, this search need not be limited to a specific field! As an example, collaborative optimization algorithms were used to design a single-stage-to-orbit launch vehicle [4].

With past knowledge in hand (God bless Stack Overflow), we can then generate potential model concepts for the Solution. No approach is perfect; these last few steps are an iterative process.

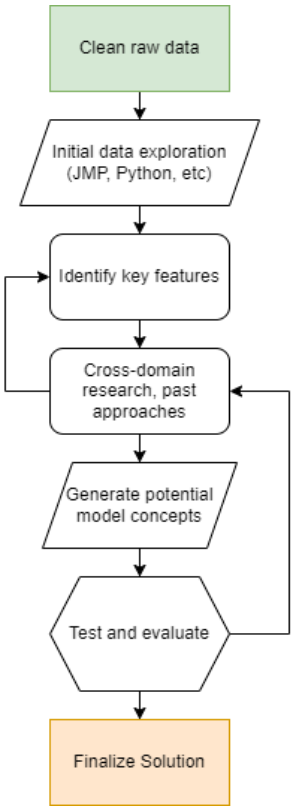

Figure 2: Problem solving approach

Initial Data Exploration

JMP Data Visualizations

The program JMP offers intuitive methods for quick data visualizations. Users can look at distributions, correlations, and create various plots.

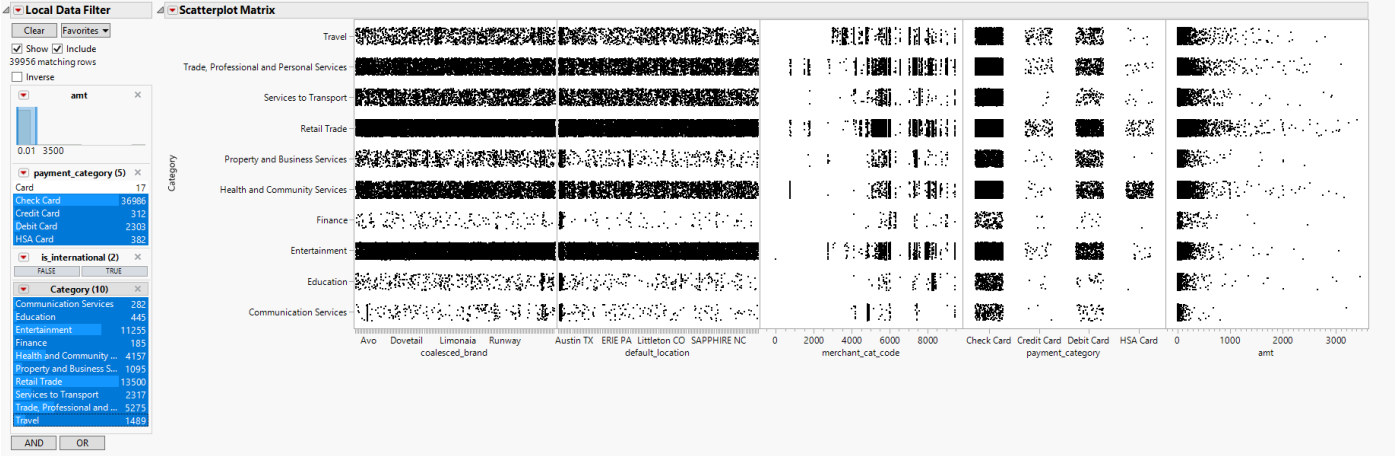

Figure 3: Scatterplot Matrix of raw training data

The initial data exploration in JMP revealed several key findings:

- Entertainment and Retail trade have the largest number of transactions
- International transactions are a small portion of the dataset (2.755%)
    - Even though international transactions are a small portion, Entertainment and Retail Trade still dominate for the international transactions subset
- From prior knowledge, I would expect HSA Card to be used for “Health and Community Services” (253 items) although retail trade also shows up quite a bit (70 items)
    - Interestingly, there are two international transactions labeled as “Travel” paid with an HSA Card
- Most transactions (96%) are less than $2500

In addition to providing insights about the data, this first step helped to identify several data cleaning requirements. For example, the field `payment_category` has 17 transactions labeled “Card” has 17 representing 0.0425% of data and is likely misclassified. This, and several other aspects are addressed in the subsequent section.

Cleaning the Data

Looking through the data, there are instances with missing information. These instances will be addressed in the order of occurrence, by the headers.

Firstly, the field `sor` (source of record) only contains two values: HH (39688 items) and BK (312 items). While HH is described as "ACH" in the provided metadata file, there is no info on BK. Given that the majority of the data is labeled HH (99.22%), this column was dropped due to lack of relevance/info.

Looking at the second field, `cdf_seq_no` text is a unique number to identify each transaction. From inspection, it appears to be a timestamp [YYYYMMDD] plus a transaction number. Given that this identifier is unique to each transaction, it does not provide any information about the classification (*implicitly assuming date has no impact either* as that is replaced by 1's in the `trans_desc`).

The third field, `trans_desc` contains the transaction description. It is parsed and cleaned up in later fields, so it can be considered redundant information and is discarded.

Turning next to `payment_category`, there are 17 instances of "Card". Instead, this value should be "Debit Card" or "Credit Card." Luckily, the field `db_cr_cd` indicates the type of card, allowing us to replace the "Card" instances with the appropriate `payment_category`.

Similarly, `payment_reporting_category` has a value of "Card" for all items. Since it is the same of all items in the dataset, it is removed (provides no unique information for categorization).

Looking at `default_brand`, `qrated_brand`, `coalesced_brand`, these fields contain functionally similar information. In fact, `qrated_brand` and `coalesced_brand` are identical in ~92% of the test data. Therefore, only `coalesced_brand` was kept since it is the most sanitized version of the brand name and has no missing data.

Lastly, `default_location` often contains a string of 1's with dashes, in addition to the actual merchant location. This is likely as a result of parsing the `trans_desc` field, so the non-location data was removed from the items.

Table 1: Summary of Data Cleaning by field

Field	Action	Rationale
sor	Deleted	Lack of info on BK, 99%+ transactions are HH
cdf_seq_no	Deleted	Unique ID, implicit assumption that date has no impact
trans_desc	Deleted	Redundant info, contained in later fields
payment_category	*Modified*	Replaced "Card" with either "Debit Card" or "Credit Card" based on db_cr_cd field
payment_reporting_category	Deleted	Same value for all items
default_brand	Deleted	Same information as `coalesced_brand`
qrated_brand	Deleted	See note above
default_location	*Modified*	Removed non-location information (e.g., dashes, numbers)

Cross-Discipline Research

Before creating a novel solution, it is prudent to first conduct a literature review of existing solutions. Furthermore, it is imperative that this review include disciplines beyond Machine Learning- in the past, novel AI/ML solutions (e.g., genetic algorithms) have been derived from fields such as biology [5]. Note that the Challenge Objective requires that a transaction be categorized into one of ten distinct categories using the provided data (ranges from price [float] to merchant name [string]). Consequently, this is a ***multi-class NLP classification problem with mixed data***. Knowing this is critical when looking for "similar" problems in the initial research.

The first area of inspiration is that of *cybersecurity*. Often network defenders have a mountain of event logs from various user devices, firewalls, and routers. These event logs contain a large variety of mixed data (timestamp, description, network location, user, etc.). While there are numerous papers on this topic [9][10][11], the main takeaway is that of sectioned learning. Each feature of the data is isolated and trained on separately. Then those models are combined into a larger metamodel that performs analysis. However, multi-class data is often converted into a numeric representation for ease of use [12].

Aircraft design is the second area of inspiration. In preliminary design, engineers must consider the powerplant, airframe, wing structures, aerodynamics, flight planning, mission requirements, and economics, all while producing a physically possible aircraft. Furthermore, after an aircraft is in service, the sheer magnitude of flight, tracking, airport operations, and maintenance data is both overwhelming and highly diversified. Although relatively new, the field of aircraft design has utilized machine learning techniques to aid in conceptual design and sustainment operations.

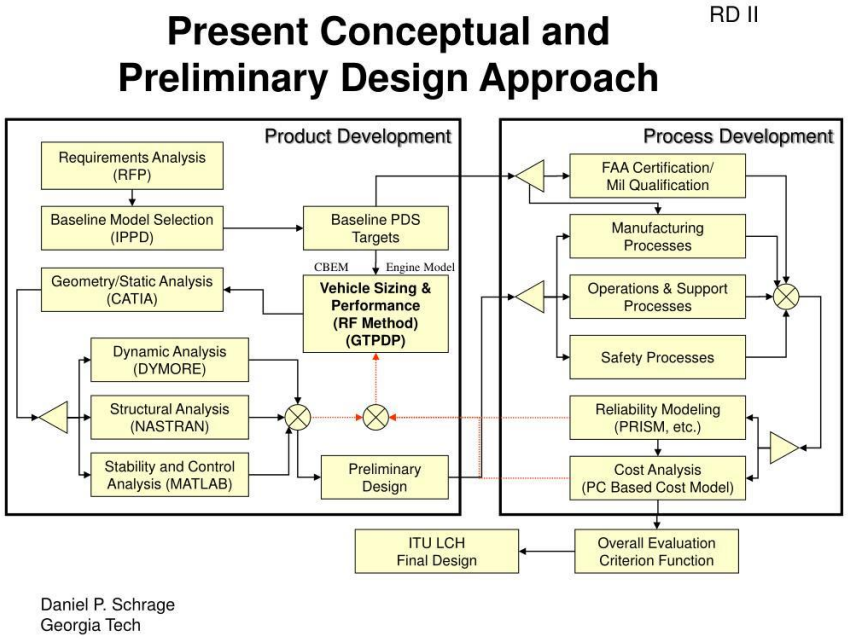

Figure 4:Aircraft design involves a multitude of varied data

Specifically, the main takeaways from the area of aircraft design were process related to autoencoding a large number of diversified features [13] and parameter selection [14]. Given similar situation in the Wells Fargo provided dataset, the former[13] assists in cutting down the relevant features to model, while the latter[14] helps to identify the parameters to adjust in the model.

Solution Approach

Idealized Approach

Although this is not the solution selected, it is still worth mentioning due to its simplicity. This idealized approach is derived based on a simple axiom: all transactions occur at a business.

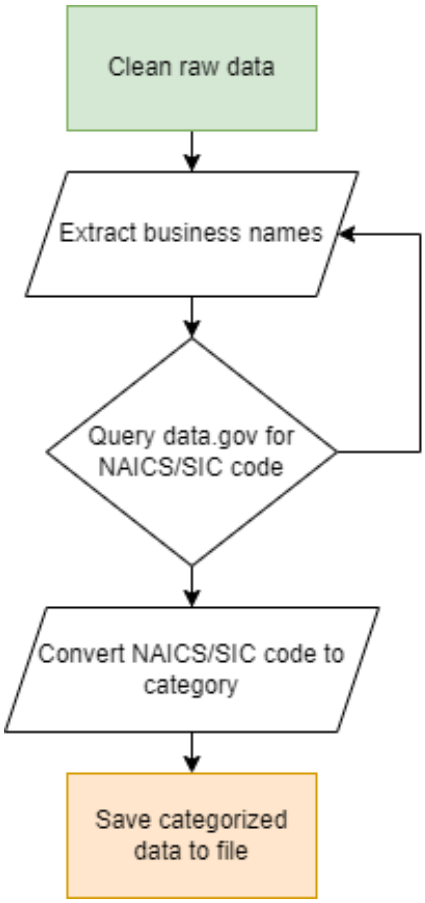

Figure 5: Idealized Approach process flow

In order to conduct business (at least in the United States), the merchant must register their business with the relevant authorities. The exact governing body varies locally, but it is almost always the local city and the federal level IRS [6]. This data is public and often searchable by business name [7]. Furthermore, business are given a NAICS/SIC code describing the industry it operates in [8]. Therefore, determining the category of a transaction is as simple as looking up the business, extracting the NAICS/SIC code and converting the designation to a category.

Implemented Approach

The implemented approach (also referred to as the **Solution**) draws on elements from the Cross-Discipline Research. The solution is summarized in the process flowchart below. *Note that the larger dashed boxes represent the code cells in main.py, denoted by "#%% === DESCRIPTION ==%%#."*

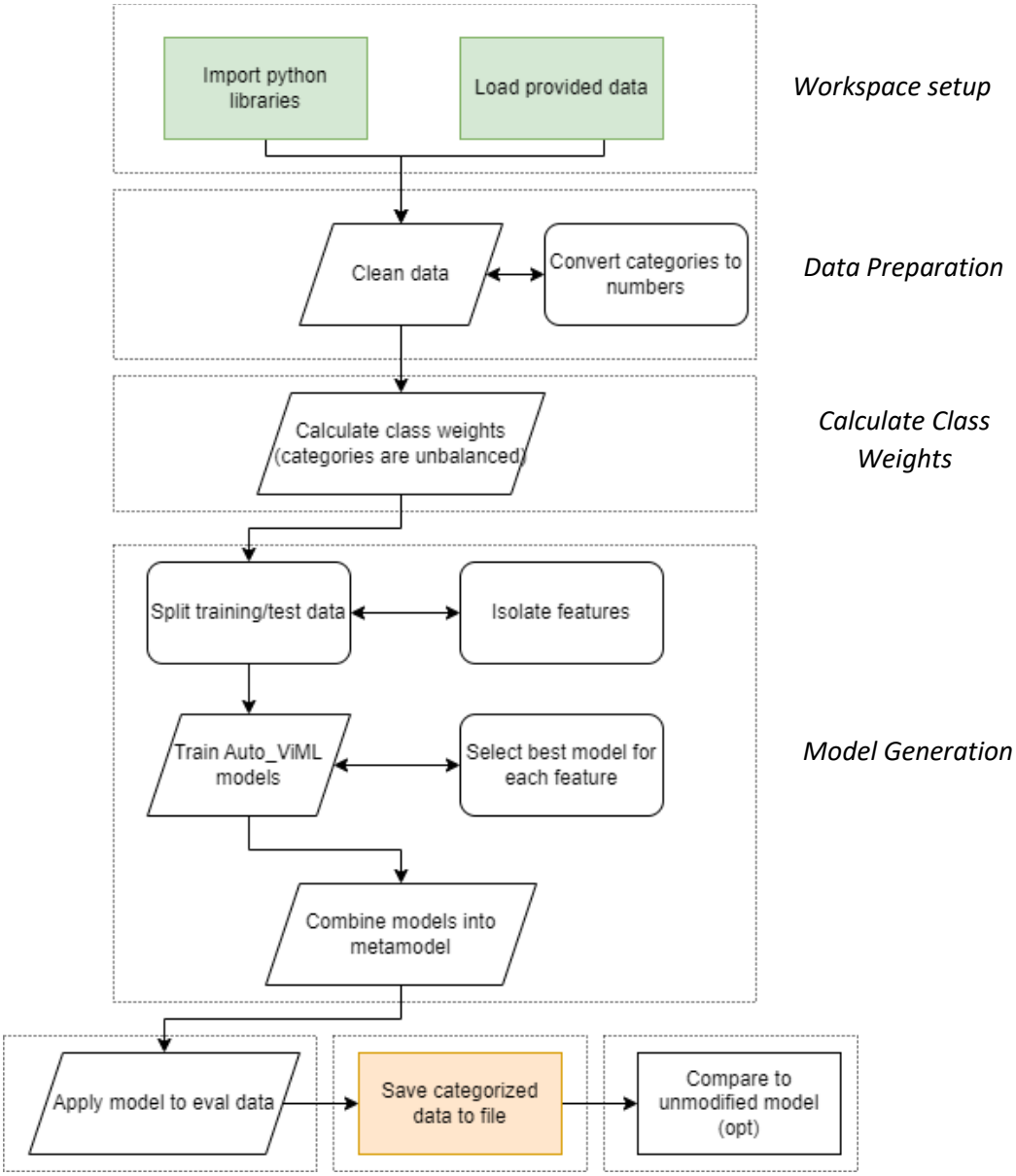

Figure 6: Implemented Approach (Solution) process flowchart with code cells identified

The novelty of this approach is in the following aspects:

1. Feature selection based on underlying data (Cleaning the Data) and aerospace techniques
2. Class weighting based on unbalanced data
3. Training sub-models on each feature and combining into a metamodel
4. Optimizing the metamodel using the Auto_ViML python library [15]

Workspace Setup

In the first dashed box (indicating a code cell), the python libraries are imported. The Wells Fargo provided challenge data is loaded from Excel into a pandas dataframe.

Data Preparation

First, the training data is cleaned to replace any missing data and properly format any datapoints (see Cleaning the Data). After this, the category (target label or "class") is converted to a numeric identifier to make it easier to work with.

Calculate Class Weights

From our initial data explorations, we know that the target labels are unevenly distributed. When generating our model, each of the classes will have different weights to account for this.

```
In [13]: df['Category'].value_counts()
Out[13]:
Retail Trade                                13500
Entertainment                               11255
Trade, Professional and Personal Services    5275
Health and Community Services                4157
Services to Transport                        2317
Travel                                       1489
Property and Business Services               1095
Education                                     445
Communication Services                        282
Finance                                       185
Name: Category, dtype: int64
```

Figure 7: Uneven target label distribution requires weighting of each "class"

Model Generation

In accordance with classical machine learning model preparation procedures, the training data is split into training (80%) and testing (20%) sub-sets. At the same time, the key features used for the first set of models is identified in accordance with Wang, et al [13].

From there, multiple models are trained using the identified features. An advantage of using the Auto_ViML library is that it automatically selects the best model for each feature. Once the training is complete, the models are combined into a second, larger metamodel (another feature of Auto_ViML).

Evaluation Predictions

Using the metamodel, predictions are made on the provided evaluation dataset ("CAC 2022 Test Data Set New.xlsx"). These predictions are added to the evaluation dataframe for use in the next block.

Saving the Evaluation Data

The categorized evaluation data is then saved to an Excel file called "categorized_eval_data.xlsx" located in the top level directory.

Compare to Unmodified Model

Comparison to an unmodified model (i.e., training on every feature together instead of building a metamodel is triggered by setting the variable `compare` to `True`.

Strengths and Limitations

As discussed previously, the novelty of this approach is in the following aspects:

1. Feature selection based on underlying data (Cleaning the Data) and aerospace techniques
2. Class weighting based on unbalanced data
3. Training sub-models on each feature and combining into a metamodel
4. Optimizing the metamodel using the Auto_ViML python library [15]

Specifically, the strongest strength is in the creation of a metamodel. This was a technique I was exposed to while working on my master's thesis in Aerospace Engineering, which focused on developing the framework for a conceptual design environment that tied together various analysis tools from propulsion to aerodynamics. The output of each of these tools was greatly varied and running each tool was quite slow. In order to perform a large design space exploration, a model was trained on each tool and then integrated into a higher level metamodel controlled by the environment wrapper. Doing so greatly decreased runtime at a minimal cost to accuracy.

The metamodel approach essentially enabled conceptual designers to examine a vast design space of various airframe-propulsion-mission configurations in a reasonable time. Similarly in the **Challenge**, the metamodel approach allows each sub-model to be tailored to a specific feature which is then rolled up into a high order model. This in turn carries with it the added benefits of flexibility (simply train a new sub model if new features are added) and reduce runtime.

While the previously described Solution does meet the **Challenge Requirements**, it is not without its limitations. The initial sub-models were very inaccurate (as seen in the figure below).

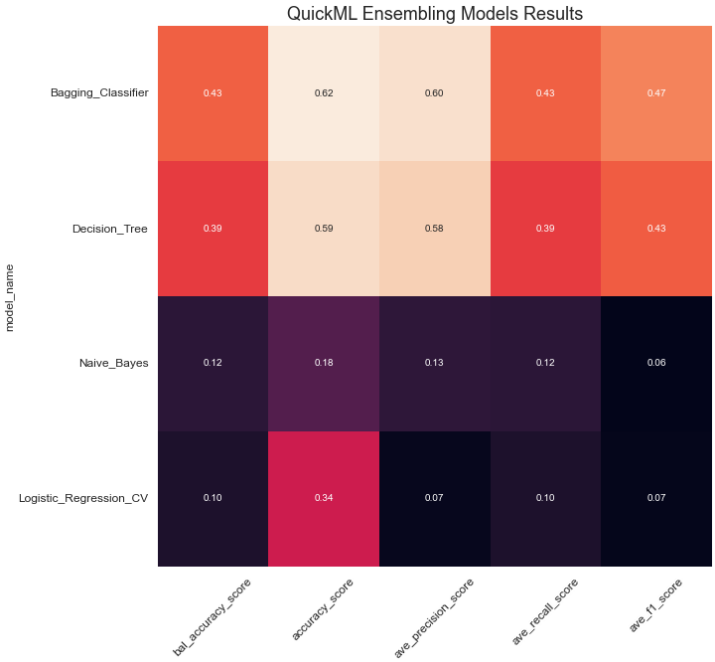

Figure 8: Initial training accuracy of sub-models

This was eventually improved as the model continued to train, but it is worrying that the initial models performed so poorly- perhaps indicating that the choice of features should be expanded or more transaction information be provided. Although the problem appeared to resolve itself through the

combination of the sub-models into a singular metamodel, if given more time, it would be worthwhile to consider a different optimizer setup in Auto_ViML to better select the ideal model for each feature.

Another limitation is that of the `default_location` feature. When training, Auto_ViML discarded this feature since the NLP aspects of the library were not designed to include geolocation data. However, from experience, the geolocation data should provide information about the transaction category. For example, consider Lititz, PA (Amish town) and Los Angeles, CA. The former is likely to have predominately "Property and Business Services," whereas LA would be more diversified and almost certainly have more "Retail Trade" transactions.

Figure 9: Feature weight in sub-model 1, note that default_location was discarded

Real-world Potential and Applications

As mentioned previously in the Introduction, transaction classification offers numerous benefits to both the customer and financial institution. More specifically, it can help customers identify frequent purchases and subscriptions, sort their income and activity liability with higher accuracy, and reduce credit risks. Speaking from the perspective of a consumer of products containing transaction categorization, it greatly helps to identify spending categories and makes budgeting more accurate.

Of course, data is ever evolving, to the metamodel aspect of the model presents the greatest real-world potential. Instead of retraining a new model on the entire dataset every time a new feature is added (e.g., a rewards credit on a transaction), only a sub model needs to be trained on the new feature.

Beyond the scope of basic consumer financial transaction classification, this Solution as additional applications in a more detailed analysis. Currently, the data only contains singular transactions, so if a customer bought multiple items in a transaction, they would all be categorized as one type. In the future, the categorization could apply to different items of the transaction.

Speaking more broadly, this Solution has applications in tailored financial planning and targeted advertising. It has been routinely shown that giving the consumer more information generally results in a more informed, and specific decision. With regards to targeted advertising: as an example, if a customer has predominately "Finance" transaction, they would be better suited for financial product ads instead of travel ads.

References

[1] Discover Card Today Introduced Spend Analyzer Tool. (2009). Retrieved 12 July 2022, from https://investorrelations.discover.com/newsroom/press-releases/press-release-details/2009/Discover-Card-Today-Introduced-Spend-Analyzer-Tool/default.aspx

[2] Mohan, B., Bharanidharan, K., Chennakesavan, R., & Guhan, P. (2020). CREDIT CARD TRANSACTION CLASSIFICATION USING MACHINE LEARNING. Retrieved 13 July 2022, from https://www.ijirt.org/master/publishedpaper/IJIRT149915_PAPER.pdf

[3] Quindlen, Anna. *Commencement Speech*; Mount Holyoke College, May 23, 1999

[4] Braun R.D., Kroo, I.M., and Moore, A.A., 1996, Use of the collaborative optimization architecture for launch vehicle design, AIAA-96-4018, Proceedings of the Sixth AIAA/NASA/USAF/ISSMO Symposium on Multidisciplinary Analysis and Optimization, Bellevue, Washington.

[5] Parmar, M. (2020). Genetic Algorithms: Biologically-Inspired Deep Learning Optimization. Retrieved 13 July 2022, from https://medium.com/ml-brew/genetic-algorithms-biologically-inspired-deep-learning-optimization-e4125e04053

[6] State and Federal Online Business Registration | Internal Revenue Service. (2022). Retrieved 13 July 2022, from https://www.irs.gov/businesses/small-businesses-self-employed/state-and-federal-online-business-registration

[7] Listing of Active Businesses - Data.gov. (2022). Retrieved 13 July 2022, from https://catalog.data.gov/dataset/listing-of-active-businesses

[8] NAICS & SIC Identification Tools | NAICS Association. (2022). Retrieved 13 July 2022, from https://www.naics.com/search/

[9] Cassetto, O. (2020). Machine Learning for Cybersecurity: Next-Gen Cyber Defense. Retrieved 13 July 2022, from https://www.exabeam.com/information-security/machine-learning-for-cybersecurity/

[10] Hung, E. (2020). Machine Learning in Cyber Security— Windows User Anomaly Detection. Retrieved 13 July 2022, from https://medium.com/analytics-vidhya/cyber-security-in-machine-learning-windows-user-anomaly-detection-e0d3457dea32

[11] Lancaster, L. (2022). Log Anomaly Detection Using Machine Learning | Zebrium. Retrieved 13 July 2022, from https://www.zebrium.com/blog/using-machine-learning-to-detect-anomalies-in-logs

[12] Deshmukh, A., & Barlow, E. (2020). Debunking the Myths. How Machine Learning (ML) Benefits Cyber Security. Retrieved 13 July 2022, from https://www.securityhq.com/blog/debunking-the-myths-how-machine-learning-ml-benefits-cyber-security/

[13] Wang, L., Lucic, P., Campbell, K., & Wanke, C. (2021). Autoencoding Features for Aviation Machine Learning Problems. *AIAA AVIATION 2021 FORUM*. doi: 10.2514/6.2021-2388

[14] Mangortey, E., Monteiro, D., Ackley, J., Gao, Z., Puranik, T., & Kirby, M. et al. (2020). Application of Machine Learning Techniques to Parameter Selection for Flight Risk Identification. *AIAA Scitech 2020 Forum*. doi: 10.2514/6.2020-1850

[15] Auto-ViML Documentation. (2022). Retrieved 13 July 2022, from https://pypi.org/project/autoviml/