** **

**BT4103 Business Analytics Capstone Project**

**Text Classification and Association Analysis for Procurement**

**Final Project Report**

**Project Group 10**

|  |  |
| --- | --- |
| **Name** | **Matriculation Number** |
| Chan Shi Jun | A0185752W |
| Keith Tan De Shun | A0190067E |
| Victor Cheong Jun Wei | A0180319H |
| Yong Zi Ping | A0183529W |

**Contents**

[**Executive Summary**](#_heading=h.2h2wf1g4bd0r) **4**

[**Background**](#_heading=h.agzz7jxj54os) **5**

[Problem Statement](#_heading=h.ocyaxcrjspe2) 5

[**Business Objectives**](#_heading=h.u7n27fj56xyh) **6**

[Reducing Manual Labor](#_heading=h.leoi3yv919ov) 6

[Making Better Decisions](#_heading=h.yjeknf8g7ta6) 6

[**Dataset**](#_heading=h.u8zul17v4nsm) **7**

[Constraints of Dataset](#_heading=h.ovkukojrpq2o) **8**

[**Proposed Workflow**](#_heading=h.ybt13v8xykjk) **9**

[**Data Exploration**](#_heading=h.lyo36mn3rhno) **10**

[Descriptive Analysis](#_heading=h.5qo1nlvr7sfq) **10**

[Number of Records](#_heading=h.plrxkyj77u3y) **10**

[Number of Classes](#_heading=h.q5y2bu4atgk) 11

[Accuracy of Labels](#_heading=h.hyfv01niycs1) **12**

[Target Variable - UNSPSC](#_heading=h.nn2cyg9e4xp) 13

[**Data Cleaning**](#_heading=h.h1uwlfmbj8tl) **14**

[Tokenization](#_heading=h.o1tkl4kqzbsk) 14

[Lemmatization/ Stemming](#_heading=h.ucalbepvwker) 14

[**Data Cycle**](#_heading=h.d1tggv4vwqo0) **15**

[**Human Cleaning**](#_heading=h.bazqpvxx970j) **16**

[Similarity Scores](#_heading=h.ma9jcja53vdl) 16

[Sparsity Reduction](#_heading=h.x0wog3xeqjou) 16

[**Back-Testing**](#_heading=h.bedqro9igsv5) **18**

[Threshold Selection](#_heading=h.z71qw31s6b90) 19

[**Forward Prediction**](#_heading=h.m6cwrqqhqiuk) **20**

[**Model Selection**](#_heading=h.qnj9hryxdj1r) **21**

[Model Justifications](#_heading=h.sjpkn5ecdkdi) 21

[Naive Bayes](#_heading=h.l0cvne5b20zb) **21**

[Linear Support Vector Machine (SVM)](#_heading=h.fb74vs1bhcev) **21**

[Logistic Regression](#_heading=h.t6qzd2bzn4f4) 22

[Long Short Term Memory (LSTM)](#_heading=h.7j81pzi53lxt) 22

[**Model Evaluation**](#_heading=h.ny7c4ewhv8hz) **23**

[**Market Basket Analysis**](#_heading=h.xrpuhx6iov8b) **24**

[Association Mining](#_heading=h.xknk1ipsm85) **24**

[Lower Inventory Cost](#_heading=h.uvtv4y42g7db) 24

[Better Projection of EPV](#_heading=h.ilqmfei9pkbe) **24**

[Proposal](#_heading=h.kp6q62hil0l5) 25

[Apriori Algorithm](#_heading=h.1c5aze4i1fa) 26

[**Dashboards**](#_heading=h.m29kshdx6t9f) **27**

[Assumptions](#_heading=h.c4rrgn81uu4h) 27

[Home Page](#_heading=h.b76vccu1n620) 28

[Spending Analysis](#_heading=h.vdfd7ioenz3b) 29

[Supplier Diversity & Price Benchmarks](#_heading=h.4vvx5j7np4so) 32

[Market Basket Analysis](#_heading=h.fpxnz1gigxbq) 34

[**Future Directions**](#_heading=h.ym0amlti87hm) **36**

[**Conclusion**](#_heading=h.w2kg5ynw48bv) **36**

[**Key Takeaways**](#_heading=h.2t35cc9ncs) **37**

[**References**](#_heading=h.hgnntj10ajl8) **38**

# 

# Executive Summary

Being a significant player in the procurement industry, DSTA faces an uphill task in manually labeling every transaction. With approximately 10,000 transactions monthly, this inefficient and laborious process would approximate 33 full man hours. Coupled with the projection of a total of 15.09 billion on National Defence and 889 million on the National Defence Program, automation and optimization of the procurement process must be prioritized.

An automated text classification system, which can achieve accurate and reliable labellings efficiently, is proposed. Arguably, this will eradicate the need for manual labelling, allowing for better allocation of resources to other departments. To generate clean and reliable dataset to work on, a Data Cycle framework is implemented to leverage on machines consistency and human accuracy. The proposed text classifier boasts a forward prediction accuracy of 74%, utilising Long Short Term Memory (LSTM) Neural Networks.

Further, interactive dashboards that allow DSTA to utilise the machine labelled data to gather purchasing insights and make better value-for-money contracts, will be introduced. To ensure useful insights are displayed, Association Mining, with the use of Apriori Algorithm, is first conducted to draw insightful relationships between transactions. The final dashboard will display 15 months of procurement data labelled from the text classifier and seeks to provide 3 key analysis: Spending Analysis, Supplier Diversity & Price Benchmarks, and Market Basket Analysis. This report documents the approach undertaken to achieve the above. Codes supplementary to the approach will be documented in the notebooks provided.

Special acknowledgement to Academic Mentor Professor Teo and Teaching Assistant Jia Min for the valuable guidance and DSTA Client Mr Seah for the constant supervision.

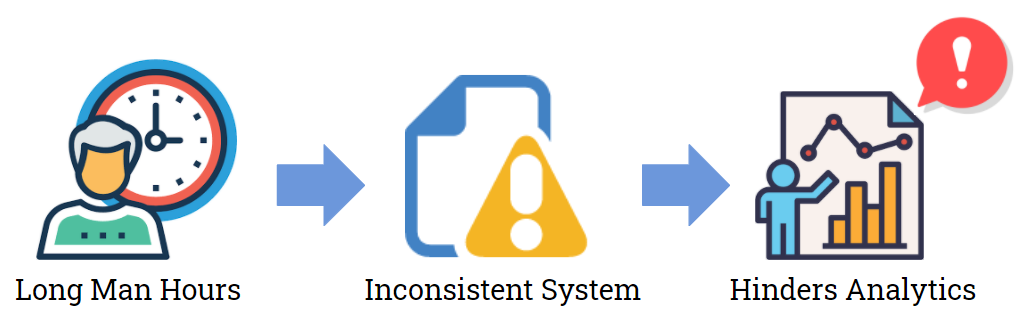
# Background

DSTA boasts itself as the centre of excellence in the procurement industry and the largest buyer of non-profit government organization. With a vast amount of procurement transactions happening daily, a significantly large sum of money is directed to the procurement system. This is further manifested in Singapore's budget 2020 where solelyNational Defence is projected to spend a total of 15.09 billion and 889.2 million for the National Defence Program for the Armed Forces.

## Problem Statement

Being a significant player in the procurement industry, DSTA faces an uphill task in manually labeling every transaction. To put this into perspective, with approximately 10,000 transactions monthly, this inefficient and laborious process would approximate 33 full man hours. This is worsened with the projection of a total of 15.09 billion on National Defence in 2020.

To make matters worse, the use of manual labour inexplicably leads to an increase in inconsistency due to human errors and differences in perspective. Considering the number of manpower working on this, the seemingly insignificant human deviation can snowball into a huge question mark. Thus, the accuracy of any further analysis becomes questionable.

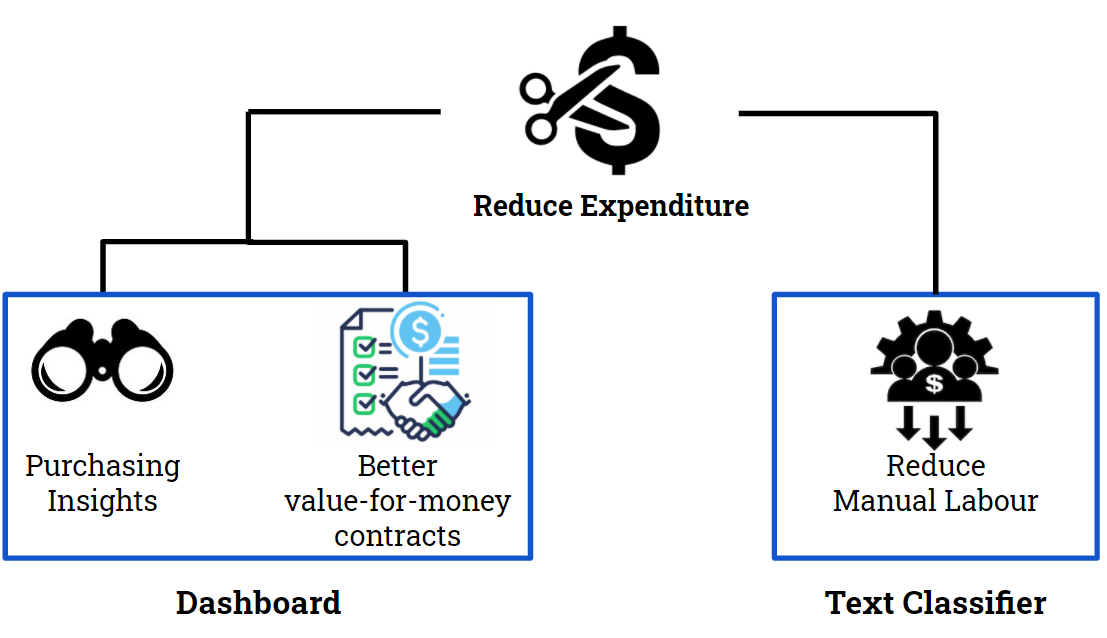


*Figure 0: Issue DSTA is facing*

To keep up with the increasing transaction frequency and injections of funds into the procurement industry, the presence of an optimized and efficient procurement system is glaringly imperative more than ever before. Automation and optimization of the procurement process must be prioritized. Hence, a reliable classifier model to replace manual labellings and the consolidation of procurement information to generate actionable insights to reduce the total expenditure must be the first step.

# Business Objectives

DSTA identified their business objective as reducing the expenditure from their current procurement system. To achieve the objective, the problem can be tackled in 2 dimensions, reducing manual labor and generating insights from purchasing patterns and thereafter making calculated decisions.



*Figure 1. Summary of Business Objectives*

## Reducing Manual Labor

Instead of manually labelling dataset for every quarter of transactions, a text classification system is proposed. The automation of this process, if proved reliable, will significantly reduce the need for human labelling moving forward. While slight alterations might be admittedly needed, a huge improvement is expected. The text classification will also lay as a foundation for association mining where it will help DSTA to transform procurement data into analytics and insights.

## Making Better Decisions

To make better and informed decisions, past actions must be analysed. Association mining will be performed sequentially after the development of the text classification system. With the insights gathered from the procurement information such as frequency and quantity purchased, as well as supplier information, this allows the establishment of a pool of suppliers for different products and price benchmarks for different purchases. With such insights, better value-for-money contracts can be derived for items frequently purchased together.

# 

# Dataset

Due to confidentiality reasons, an open-source procurement dataset by the California Department of General Services will be used as a proxy to DSTA transactions. The dataset is carefully chosen to model after DSTA’s procurement dataset in terms of size and number of classes. Both datasets also follow the United Nations Products and Services Code, in short UNSPSC, when used to categorize line item information for consistency. UNSPSC is the class label where both datasets aim to label. This similarity must be emphasised to allow for smooth integration into Singapore’s context in the future.

Each row of the dataset corresponds to each item purchased, as well as their corresponding purchasing details. Terminologies widely used in this report includes both tangible and intangible items (i.e. services and contracts). The dataset consists of 3 fiscal years (2012 - 2015). However, due to time constraints, only the first year of data will be used primarily, together with the first few months of data in the second year. Table 1 illustrates the description of the dataset.

|  |  |
| --- | --- |
| **Column Name** | **Description and Usage** |
| Creation Date | Indicates when the purchase was created.  This is an important feature to view and identify purchasing time trends. |
| Purchase Requisition Number | An unique number generated when a buyer creates a request to purchase item(s).  This is used to identify a transaction. |
| Purchase Order Number | An unique number generated when the purchase is confirmed.  This will be used to identify a transaction when the purchase requisition number is not provided. |
| Department Name | Name of Department that made this purchase request  Provides insights on purchasing trends of a department across time. It can potentially spark collaborative procurement across departments. |
| Supplier Name | Name of Supplier that supplies that item  This allows the user to identify the pool of suppliers supplying that item  . |
| Item Name | Name of the item  This column will be concatenated with the ‘Item Description’ column as text inputs for text classification. |
| Item Description | Description of the item  This column will be concatenated with the ‘Item Name’ column as text inputs for text classification. |
| Quantity | Quantity of item purchased.  This is used to track the procurement of items. |
| Total Price | Total cost = cost of good \* quantity  This is used to track the procurement of items. |
| Normalized UNSPSC | UNSPSC code of the product  Identify the product in the UNSPSC structure. This is the target variable for classification. |
| Commodity Title | Label of the product  This is used for text classification to classify the products to the corresponding labels. |

*Table 1. Description of features*

## 

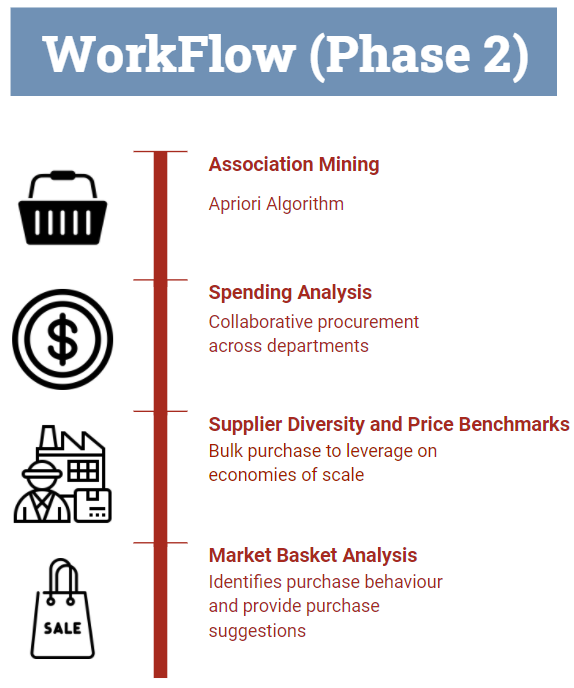
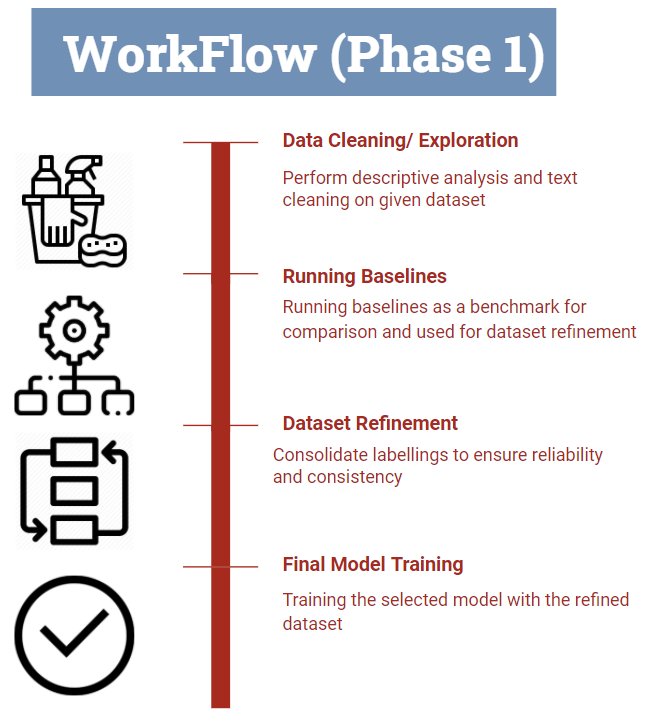
## Constraints of Dataset

The true quantity of the item might be misleading. Due to the differences in units of measurement, quantity purchased can be interpreted differently. For instance, a single quantity of bananas purchased can be interpreted as a single banana, a bunch of bananas or a crate of bananas. This will subsequently impact the unit and total price, since prices depend on the unit of measurement. In this example, a crate of bananas would definitely cost much more compared to a single banana.

Each transaction might also comprise more than 1 item. For instance, the item name could be “Office Supplies”, but item description contains multiple items such as ‘Staples’, ‘Paper’, and ‘Scissors’. For simplicity, these transactions would be only reflected as ‘Office Supplies’ instead of their individual components.

# Proposed Workflow

To ensure both business objectives are met, a project workflow with a series of steps are planned out. It is to note that both processes will be running sequentially as the association mining and dashboards built on the predictions by the text classifier built in phase 1. The individual steps taken will be further elaborated in each section of the report.

****

*Figure 2: Workflow of Project*

Throughout the project, SCRUM framework was also used to ensure a systematic and consistent workflow in the timeframe provided. Further details can be found in the SCRUM logs attached.

# 

# 

# 

# 

# 

# 

# 

# 

# Data Exploration

Data Exploration is performed for 2 main reasons:

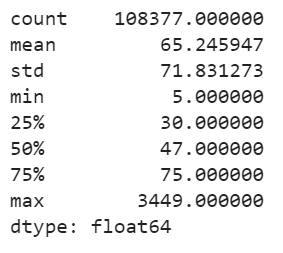
1. To identify key characteristics of dataset to support any selection of approach
2. Understand and interpret the target variable given to perform specific preprocessing steps

## Descriptive Analysis

Here, only a general description of the dataset will be discussed. Further analysis of the dataset on technical specifications used later on will be discussed in the Notebooks provided.

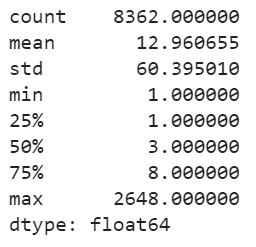
### Number of Records

The number of records is first explored. A total of 108377 records are present in the first year of training dataset *(refer to figure 3)*. The conclusion of a large dataset will be imperative in model selection, with particular emphasis in the computational complexity. The RAM of the platform used, Google Collab will also be a source for considerations. Hence, both hardware and time constraints will be considered at every step taken.



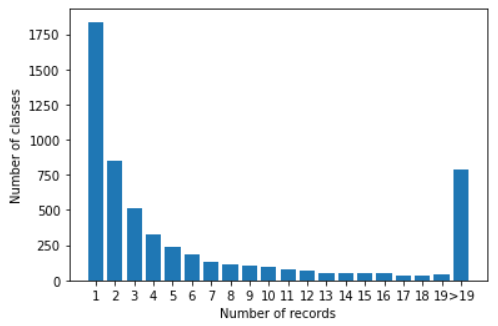
*Figure 3. Distribution statistics of training dataset*

### Number of Classes

**

*Figure 4A. Distribution statistics of class frequency in the training dataset*

Due to the complexity of multi-class classification tasks, a number of unique classes must be carefully handled. In this dataset, 8362 classes are present which motivates sparsity reduction (*further elaborated later*) and also implies the possibility of an imbalanced dataset *(refer to figure 4A)*. Hence, distribution of classes must be analysed due to their effects on machine learning models, where most are biased towards the majority classes.



*Figure 4B. Distribution of class frequency in training dataset*

As illustrated in Figure 4B which shows the plot of number of classes across different numbers of records, the distribution is highly skewed implying the presence of an imbalanced dataset. Thus, different methods of handling the dataset must be proposed, including the use of F1 score as a metric for the selected models to weigh both the precision and recall.

## 

## 

## 

### Accuracy of Labels

During manual inspection of selected features; Item Name, Item Description and their corresponding labels are analysed. Random stratified sampling is conducted on those classes with at least 20 counts to monitor the accuracy and validity of the dataset. The plot below shows a sample of the classes that checked, as well as the accuracies of the original labels (Figure 5).

# Points scored

*Figure 5 . Accuracies of original labels’ classes*

The absence of classes having accuracy of above 50% is a worrying concern. The conclusion of more than 50% of incorrectly classified labels emphasizes the need for manual labelling.

## Target Variable - UNSPSC

To effectively formulate the problem statement, the target variable must first be explained. For the text classification problem, the UNSPSC code is the target variable to be predicted. United Nations Standard Products and Services Code, also referred to as UNSPSC, is a taxonomy of products and services for use in eCommerce. It is a four-level hierarchy coded as an eight-digit number.

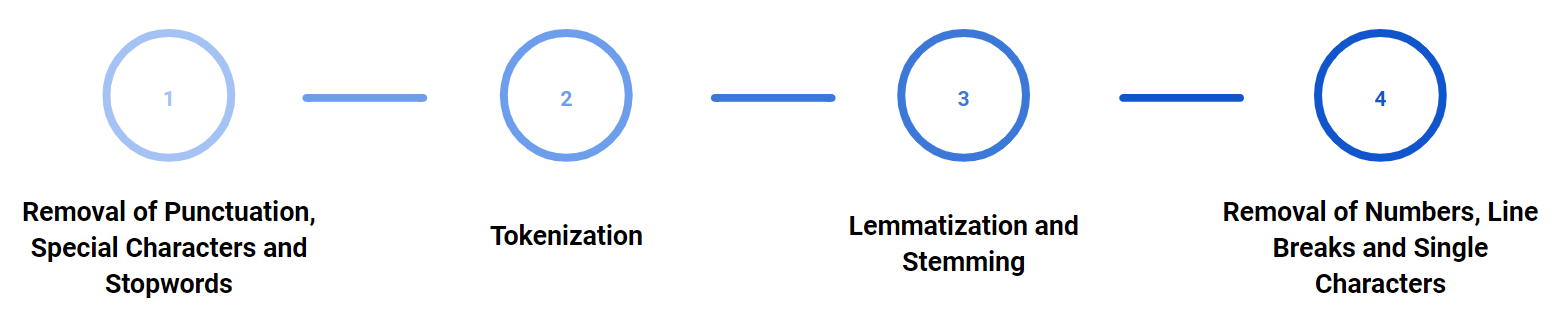


*Figure 6. Example of UNSPSC hierarchy*

Figure 6 illustrates an example of the hierarchical tree. The items at the bottom of the trees, also referred to as the leaves, represents the most specific categories for the items. For instance, Fresh milk or butter products are more specific and can be seen as a subset of the parent node, Milk and butter products. The difference will be reflected in the conversion of the digits to zeros up the tree structure from the back. Since it is possible to identify items using the given UNSPSC, categories are represented as their respective UNSPSC, removing the need for one-hot encoding. As of February 2020, there are 156,478 unique codes.

# Data Cleaning

Upon the initial exploratory data analysis, the selected features, Item Name and Item Description are identified to be unstructured, which implies the need for data cleaning.



*Figure 7. Data cleaning process*

For data cleaning, a series of steps were planned out which started with the removal of punctuation, special characters and stop words. Tokenization is then done before lemmatization and stemming on the text are performed. Lastly, numbers, line breaks and single characters are removed. It is imperative to note that the order of pre-processing performed is significant. Suppose the string is ‘\na’. If single characters were removed first before line breaks, which is \n in this case, the algorithm will fail to remove the letter ‘a’ after removing the line breaks in the second step.

## Tokenization

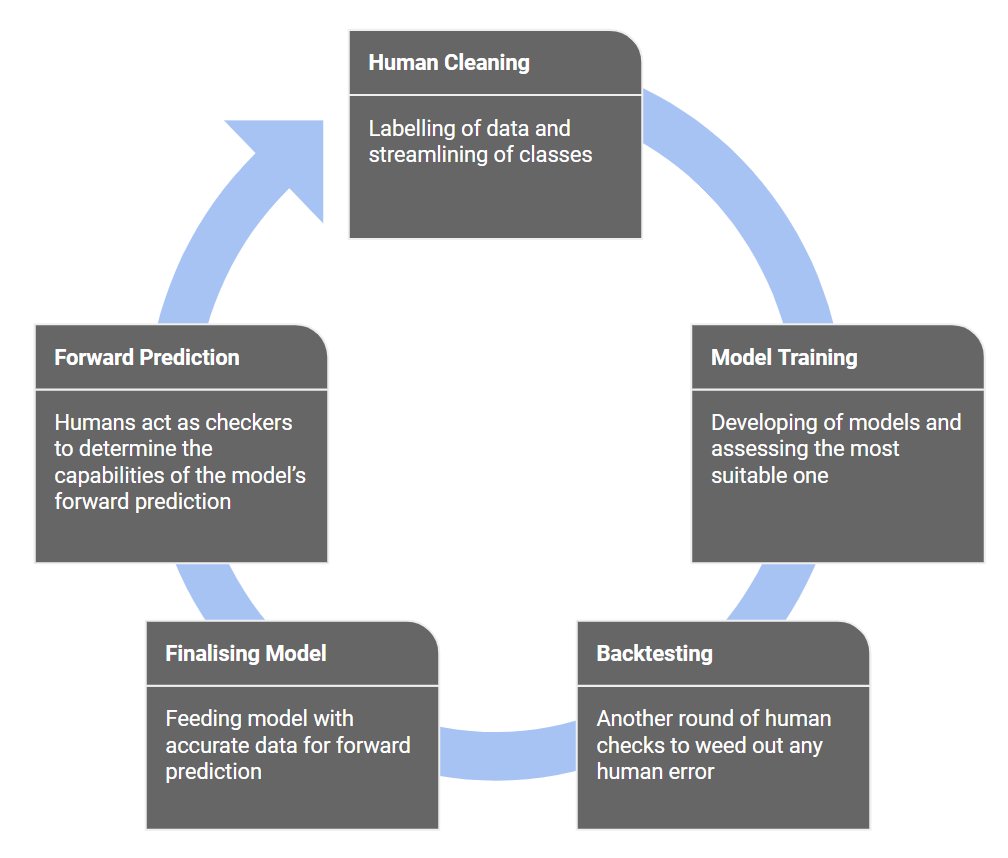
Word tokenization is the process of splitting text into words. This is required to conduct further cleaning such as removal of numbers from text in their individual word form. Text classification model which is the target deliverable for the project also requires each word to be inputted separately instead of a large sample of text. Here, RegexpTokenizer is used from the NLTK package.

## Lemmatization/ Stemming

The use of algorithms to reduce inflectional forms of a word to a common base form is crucial in the dataset. For grammatical reasons, Item Name and Item Description which are used as features to predict on the target variables will present words in different forms, such as Printers and Printer. It is therefore required to treat these items as the same labels as failure to do so will treat both as different features. This not only reduces the predictive power of the classifier but also exponentially increases the dimensions resulting in the explosion of space and time complexity. This effect is further amplified and relevant for the dataset used with a high number of records and unique words.

# Data Cycle

The aforementioned conclusion of an inaccurate and dirty dataset prompted the use of manual labelling to ensure the validity of the training set as the model will only be as good as the data fed in. While ensuring the validity of the training set, time constraints must be considered. Coupled with the fact that multiple years of dataset are inaccurately labelled, amounting to more than 300,000 records to be labelled, a framework named Data Cycle, is proposed to clean them efficiently.



*Figure 8. Data Cycle process*

For the process of data refinement, 5 processes, Human Cleaning, Model Training, Back testing, Finalising Model and Forward Prediction were implemented *(refer to figure 8)*.

# Human Cleaning

## Similarity Scores

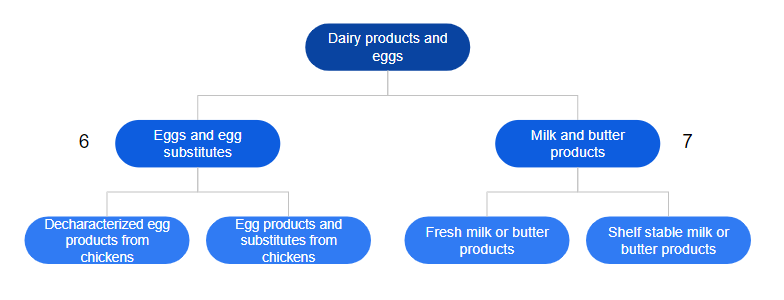
Similarity scores are generated from constructing a similarity matrix with the use of the gensim package. Here, the labels are first pre-processed and passed through a vectorizer to get the probability of importance for each word. The similarity scores are generated across all combinations of labels present and the top 3 labels will be presented. This serves as a recommendation to users by surfacing similar labels in different dimensions. The usage of these similarity scores will not only reduce the number of classes, spelling errors and inconsistency in human labelling will be surfaced.

## Sparsity Reduction

Recalling the presence of more than 8000 classes from data exploration, coupled with the highly imbalanced dataset which will possibly lead to overfitting of classes with low counts, the concept of Sparsity Reduction is proposed. To visualize the problem, Figures 9 and 10 illustrate how we can achieve a smaller context and how the number of classes can be reduced.

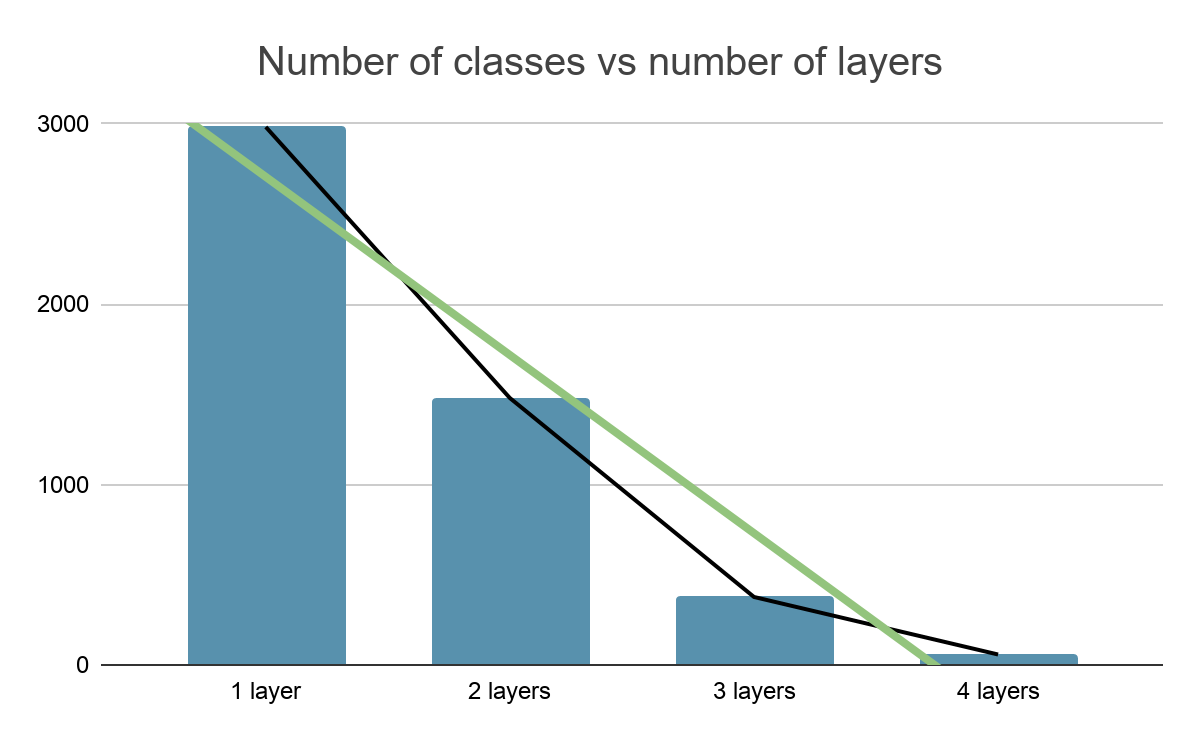


*Figure 9. Classification without sparsity reduction*



*Figure 10. Classification with sparsity reduction*

From Figure 9, 4 classes with low frequency counts are present in the leaves of the tree structure respectively as 2, 4, 4 and 3. By grouping them in Figure 10, classes are elevated to one layer above. As manifested in the diagram, from 4 classes of labels, grouping is done to reduce to 2 new classes, thus reducing the risk of overfitting. In this case, versatility in the model is achieved. While the acknowledgement of reducing the specificity of labels must be made, a trade-off between the 2 factors is weighted which results in the conclusion of elevating the classes only by at most a layer. Figure 11 shows the tradeoff of specificity and degree of sparsity reduction.



*Figure 11. Number of classes across different layers (specificity)*

The black line shows the observed trend in number of classes while the green line represents a linear trend fitted. The plot thus shows that an increase of 1 more layer will result in the decrease of disproportionately large number of classes as the gradient is larger than a linear decrease. Hence, the optimum tradeoff is determined by elevating only 1 layer. An assumption is also made that should the labels be too generalised for the user, it is very much possible and convenient to go down one layer by searching with the other predicted 6 digits.

To perform sparsity reduction, a series of steps are executed:

1. The corresponding UNSPSC for each label is extracted
2. UNSPSC is carefully sorted by the front digits which can be seen as grouping the subtrees in the UNSPSC hierarchical tree structure
3. Similarity matrix is also constructed to recommend similar labels via similarity scores
4. Mapping of the previous labels to the updated labels and corresponding UNSPSC

# Back-Testing

After each record in the training set is manually labelled, several issues ensued:

1. Presence of spelling errors in the labels
2. Inconsistency due to differences in individuals’ perspectives

To mitigate these issues, back-testing is conducted. Back-testing is defined as the process of using a trained model to predict the training set. This seeks to leverage on the model’s prediction to suggest predicted labels that might be similar to the actual labels. To ensure efficiency, not all labels will be relooked but only those that have a confidence below a proposed threshold.

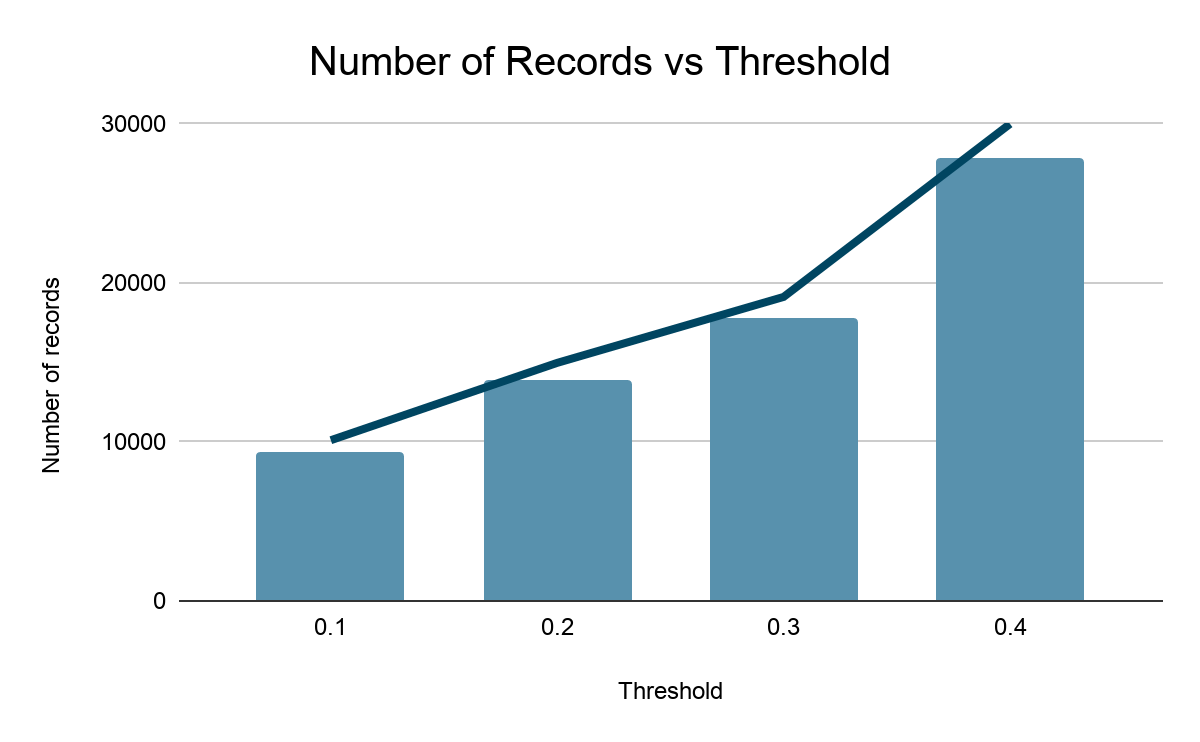
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Item Name + Item Description** | **1st Predicted Class** | **1st Predicted Class Probability** | **2nd Predicted Class** | **2nd Predicted Class Probability** | **Difference in Probability** |
| Vegetables Zucchini Peas Mix Veges | Vegetables | 0.8469 | Fresh Vegetables | 0.7310 | 0.1159 |
| Vegetables Zucchini Peas Mix Veges | Vegetables | 0.8469 | Fruits | 0.3581 | 0.4888 |

*Table 2. Illustration of backtesting threshold*

Table 2 illustrates 2 hypothetical examples where 2 predicted labels are generated from the model for each record. For the first row, vegetables are predicted with the highest probability and fresh vegetables as the second best. Predicted probabilities for both labels extracted from the model will be subtracted to find the differences in probabilities. If the difference is lower than the threshold, manual inspection is performed to ensure the consistency of the actual labels. The second example shows the case where differences in probabilities exceed the threshold, indicating that the model is confident in separating it from other labels. In this case, an assumption is made that since the next best predictions are not close to the first predicted label, the labels are unique and hence the conclusion that there is no inconsistency from the absence of similar labels that the model might have predicted. This seeks to provide 2 key merits:

1. Flags out any form of spelling errors
2. Eliminate inconsistency in human labelling by ensuring no similar labels

## Threshold Selection

This process requires a parameter to be decided upon. Firstly, random sampling is conducted to monitor the percentage of changes made at each threshold. With the limitation of time, the threshold is selected with labour hours and effectiveness within considerations. The plot below shows the tradeoff between the number of labour hours and the threshold selected (Figure 12).

*Figure 12. Barplot showing the number of records for setting each threshold*

Ideally, all the labels should be relooked, however, when the threshold is 0.4, the number of labour hours needed to label the records increases very sharply from when the threshold 0.3 testament by the steep gradient. From 0.1 to 0.3, the increase in the number of labour hours needed is rather consistent. The threshold, 0.3 is thus selected as it offers the highest marginal benefits.

# Forward Prediction

Forward Prediction is the process of using the modified labels to retrain the model and predict on the unlabelled training data. The framework stems from 2 motivations:

1. Inability to manually tag all training data given time constraints
2. Reduce reliance on manual tagging which may result in human errors and inconsistency

Given the huge number of categories possible in the UNSPSC structure, unseen labels in the unlabelled training data is expected. In such cases, a reasonable assumption is made, that the items that are not previously seen by the model will be predicted poorly. As subsequent manual inspection is conducted, new labels will be proposed.

To ensure that new labels are predicted correctly, a threshold of 0.3, selected similarly to that in back-testing, is proposed. Simply put, forward prediction checks the records that the model has low confidence in implied from their class probabilities. The assumption that the model correctly predicts records above the threshold is made in this process.

# Model Selection

## Model Justifications

Upon secondary research on the models available, further analysis and comparison will be focused on the selected models, namely, Naïve Bayes, Support Vector Machines, Logistic Regression and Long Short-Term Memory (LSTM).

### 

### *Naive Bayes*

In text classification, each unique word is a feature. Considering the high dimensionality of the dataset due to the vectorization of the unique words, a simple method can be proposed which is low in computational and space complexity for efficient training. Aiyappan, J. (2019) explained that Naive Bayes consumes less memory. Naive Bayes achieves this by assuming independence between words. In particular, Laplace smoothing can be leveraged to adjust the posterior probabilities of unseen words instead of letting the probability be 0. This ensures that words or nomenclature that did not appear in the training set are not simply ignored. The disadvantage, however, might boil down to treating text as independent words which more often have semantic meanings in them.

### 

### *Linear Support Vector Machine (SVM)*

Kowalczyk, A. (2020) pointed out that text is often linearly separable and this is the basis of model selection. In addition, using a more complex kernel like Radial Basis Function (RBF) to fit the high-dimensional data is not necessary as the hyperplanes for both the linear and RBF versions would be largely similar. Chen, L. (2019) also explained that SVM will find the hyperplane which maximises the distance between the hyperplane and the closest training point. Hyperparameters can also be tuned to ensure maximum margin hyperplane to reduce overfitting Considering the imbalance dataset, coupled with the low counts in every class labels, overfitting is especially crucial. One important point to note is the high complexity of the model which will be weighed during the evaluation process in time and accuracy trade-off.

## 

### *Logistic Regression*

As mentioned earlier that text is often linearly separable, a linear classifier is used. Contrary to generative models like Naive Bayes, Logistic Regression is a discriminative model that attempts to directly compute the posterior probability of classes (Jurafsky, Martin, 2019). This also implies the assignment of a high weight to features that directly improve its ability to discriminate between possible classes, even at the expense of generating an example. This serves as an alternative perspective from Naive Bayes and is arguably important for our dataset as many classes are not well separated. However, it is also to note that logistic regression uses an iterative process to optimise the weights via minimising the loss function. This could be indeterministic and converge to local optimum.

### *Long Short Term Memory (LSTM)*

For Long Short Term Memory, semantic context of the text is preserved. This is further elaborated by Djaja, F. (2020), who described the hidden state that is in LSTM as acting like a short term memory to learn long term patterns. Since sequencing of words in phrases in item names and description often contains certain semantic meanings, taking into account such relationships will be beneficial compared to traditional algorithms. In the case of the given dataset, given 2 item descriptions here, ‘printer supporting scanner’ and ‘printer, does not support scanner’, will be weighted independently by other models. LSTM, however, remembers the order of the words and hence is able to seperate the second product from the first product.

# Model Evaluation

Throughout the workflow, various models were trained in parallel to test the effectiveness the hypothesis made. The accuracies should improve progressively, which is manifested in the summary table below *(refer to table 3)*.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  | **Naive Bayes** | **Linear SVM** | **Logistic Regression** | **LSTM** |
| **Before labelling** | Accuracy | 40 | 65 | 61 | 61 |
| F1 Score | 33 | 64 | 59 | 58 |
| **After labelling** | Accuracy | 50 | 77 | 74 | 68 |
| F1 Score | 42 | 76 | 72 | 65 |
| **Sparsity Reduction** | Accuracy | 52 | 78 | 76 | 70 |
| F1 Score | 45 | 77 | 74 | 68 |
| **Back testing** | Accuracy | - | - | - | 71 |
| F1 Score | - | - | - | 69 |
| **Forward prediction (Cycle 1)** | Accuracy | - | - | - | 69 |
| F1 Score | - | - | - | 68 |
| **Forward prediction (Cycle 2)** | Accuracy | - | - | - | 76 |
| F1 Score | - | - | - | 75 |
| **Forward prediction (Final evaluation)** | Accuracy | - | - | - | 74 |
| F1 Score | - | - | - | 73 |

*Table 3. Overview of Models’ Results*

As aforementioned, due to the presence of an imbalance and skewed dataset, both accuracy and F1 score are considered. It is worth noting that the F1 score is consistently slightly lower than the accuracy score implying that the models are not performing as well in classes with low frequency counts. Due to the nature of the target variable, UNSPSC, which has more than 100,000 possible categories, this is the limitation of the multiclass classification task. Median performance hovers around 60% from the distribution of F1 scores across classes.

From backtesting onwards, only LSTM is used due to the computational complexity in other models chosen. This is due to time and hardware constraints in the project, where the complexity and accuracy trade-off must be carefully weighted. As across the cycles, data refinement will expand the size of the dataset, considering the exponential increase in records and dimensions due to the increase in unique words LSTM is used. Further, taking into account workplace considerations, due to the influx of transactional data, model training is frequently conducted. It is advisable for this process to be completed within the off hours, to ensure that insights are accurate and updated by working hours. Given the current training size of approximately 100,000, the time taken to train Support Vector Machines and Logistic Regression is well beyond 24 hours. Hence, compromise must be made on a slight decrease in accuracy for a shorter training time. The slight decrease in model performance is due to having an unseen dataset during forward prediction.

# 

# 

# Market Basket Analysis

## Association Mining

To optimize the procurement process, it is important to understand purchase behaviour and provide purchase suggestions through the identification of relationships between items. This process is called Association Mining. Association mining aims to find out whether people tend to buy item B when item A is bought. This optimize the procurement process in 2 dimensions:

## Lower Inventory Cost

The identification of key relationships between items allows for proper planning of purchases. This arguably gives negotiation power to the buyer as associated items can be purchased in bulk rather than in separate orders, resulting in lower cost via demand aggregation. For instance, staples and staplers are identified as associated items. This relationship is drawn from the user’s previous purchasing behaviours and allows for the user to focus all the purchases to a certain supplier. The concentration of purchases on a supplier and hence increased volume of purchases will leverage on the economy of scale for a lower cost.

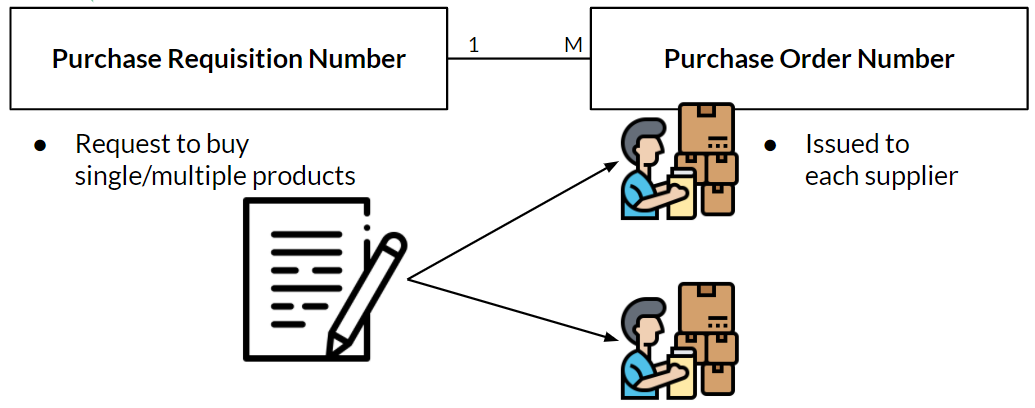
## Better Projection of EPV

Associative rules can be used to determine the items typically bought together for a certain event. A more reliable estimate of the cost incurred can thus be derived. This implies the capability of optimizing the procurement process in the macroscopic perspective rather than the microscopic view of purchases mentioned earlier. Suppose to hold a certain event, a list of items can be generated based on the associative rules. From the macroscopic view, users can choose to delegate the entire purchases to a certain supplier or delve deeper into individual transactions. Based on the decision, the costs for that event can be efficiently calculated. This is useful for calculating the Expected Projected Value (EPV) of the order before submitting at the purchase request stage.

## 

## Proposal

Before performing any kinds of association mining, transactions must first be defined. In the dataset, 2 features were promising, purchase order number and purchase requisition number. A purchase requisition number is issued when a buyer submits a request to the procuring department to purchase several products. This disregards the assumption that there exists a supplier who sells all of the products and hence can only be used to explain the relationship of items that are requested. A purchase order number, however, is to split the request into different suppliers with each given an order number. In such cases, it is likely that the items can be bought from the same supplier.

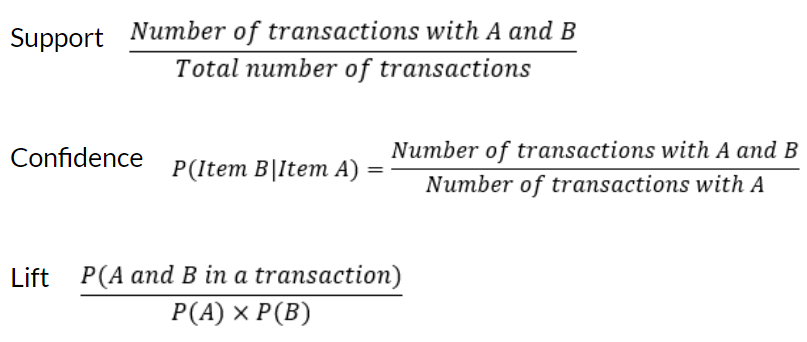


*Figure 13. Relationship between purchase requisition and purchase order number*

Since the focus lies on the purchasing patterns of the user and not the supplier, transactions are defined as buying products at the same time by a user. Thus, it is more logical to use the purchase requisition number as it controls for the assumption that there exists a supplier who sells all of the products. It is however important to note that most records do not have a purchase requisition number, hence, for such records, purchase order numbers will be used.

### Apriori Algorithm

Apriori algorithm calculates 3 measures, support, confidence and lift which are different metrics to determine the usefulness of an associative rule. Poorly performing associative rules in terms of confidence level will be pruned as the algorithm scores every possible association. This allows for the discovery of interesting rules which may otherwise be neglected.



*Figure 14. Formulas for the metrics of apriori algorithm*

The formulas of the metrics are illustrated in Figure 14. The support is a naive calculation of the probability of both items A and B appearing in a transaction. Confidence represents the conditional probability of a customer buying item B given that he is buying item A. The higher the confidence, the stronger the association between items A and B. In practice, a confidence of 70% is deemed strong. Lift, on the other hand, represents the likelihood of purchasing both items together as opposed to individual items. A lift greater than 1 implies it is more likely that a customer will buy both items together rather than separately. Hence, to deem 2 products as associated, typically both the confidence and lift will be considered. In the project, the decision is left to the user when using the dashboard for insights.

# 

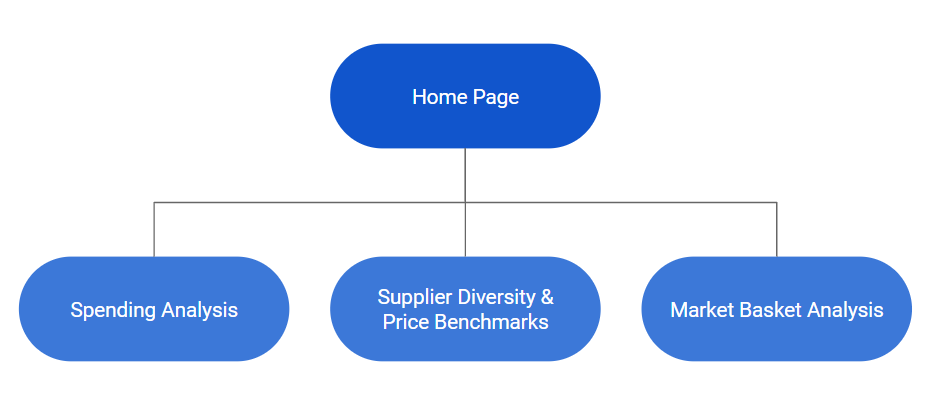
# 

# Dashboards

Revisiting the business objectives, the other deliverable will be the production of a dashboard. This dashboard is built upon the foundation laid by the text classifier as it utilizes the labels that were predicted as the basis for the design. 2 keys functionality were considered during the design:

1. Allow users to gather purchasing insights from past procurements
2. Allow for better value-for-money contracts

Figure 15 illustrates the layout of the dashboards. Spending Analysis, Supplier Diversity & Price Benchmarks, and Market Basket Analysis are proposed to answer the objectives.



*Figure 15. Layout of the dashboards*

## Assumptions

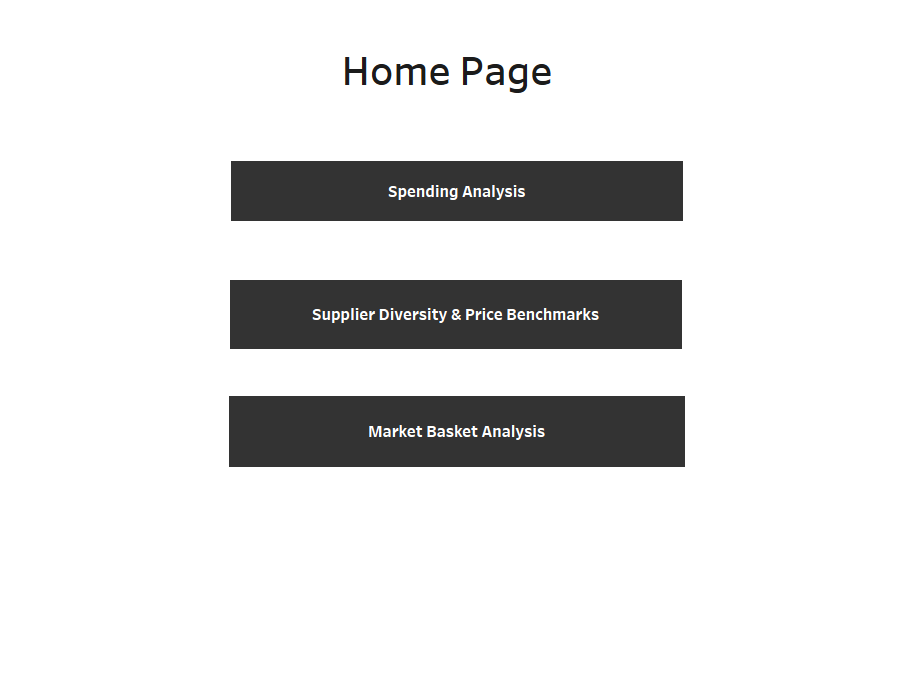
Due to the aforementioned constraints in the dataset, 2 main assumptions were made when designing the dashboards with the selected dataset:

1. While the dataset provided both the quantity and unit price column, the unit of measurement of each quantity assigned to each record in the dataset is not specified. For instance, an order of bananas having a quantity of 1 could imply 1 banana, 1 bunch of bananas, or even one crate of bananas. This issue surfaced when comparison of unit item cost is made across various suppliers. This will result in huge variations in unit cost since denominations differ across rows of the dataset. Discussion with the client concludes the minimal severity of this assumption due to presence of consistent procurement process.

Taking into consideration the stakeholders, the unit cost in the final product will still be displayed in the current design, noting the presence of some discrepancies with the current dataset. Since the dashboard is designed for DSTA, consistency of unit costs can be assured once the dataset is replaced.

1. The dataset that we are using does not have sourcing information of suppliers, meaning that it only contains successful bids and transactions by suppliers. This results in us having lessinformation about the available pool of suppliers, as suppliers might actually supply more variety of goods, but did not manage to sell these goods to the company. This would mean that with our client’s dataset, we would have a greater potential when comparing suppliers’ goods and price benchmarks.

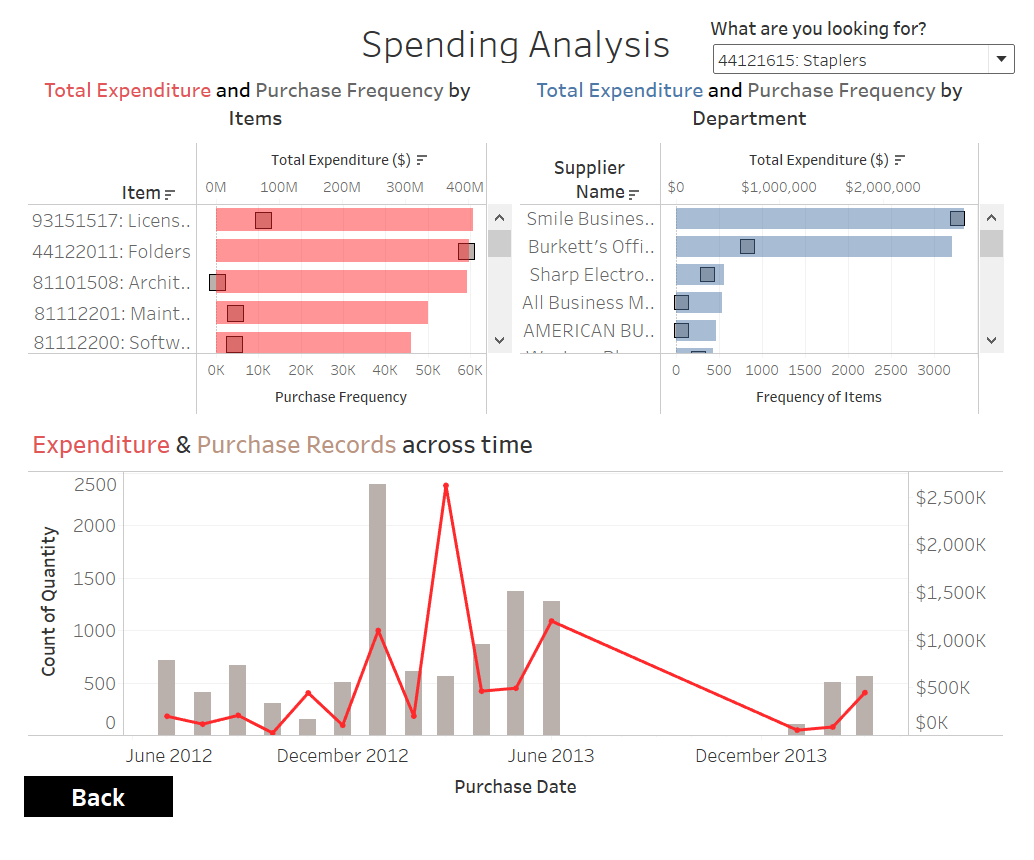
## Home Page



*Figure 16. Layout of Home Page in the dashboard*

The Home Page is the main navigation page for the series of dashboards. There are 3 buttons, each button corresponding to the different dashboard, allowing the user to navigate through each dashboard.

## Spending Analysis

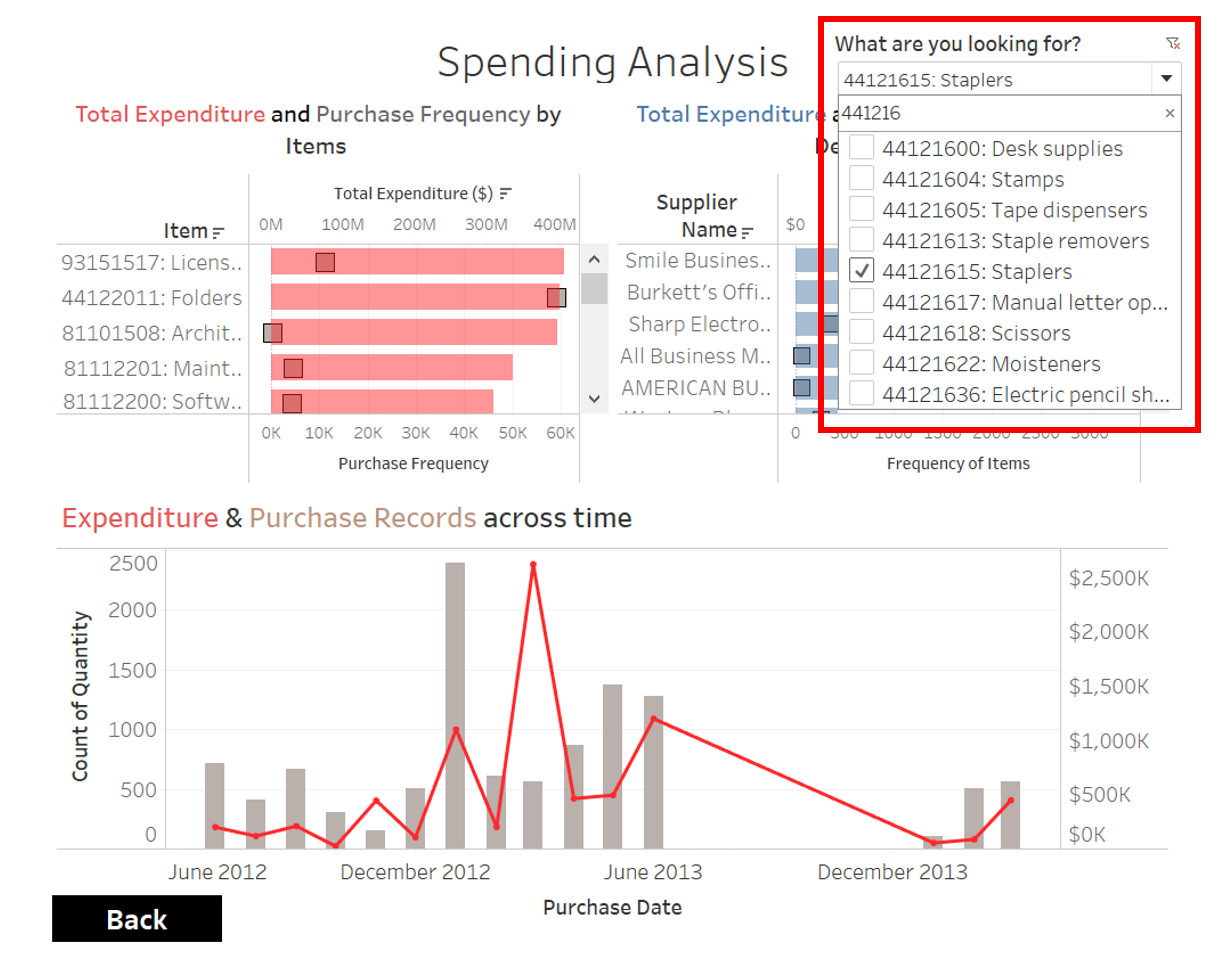
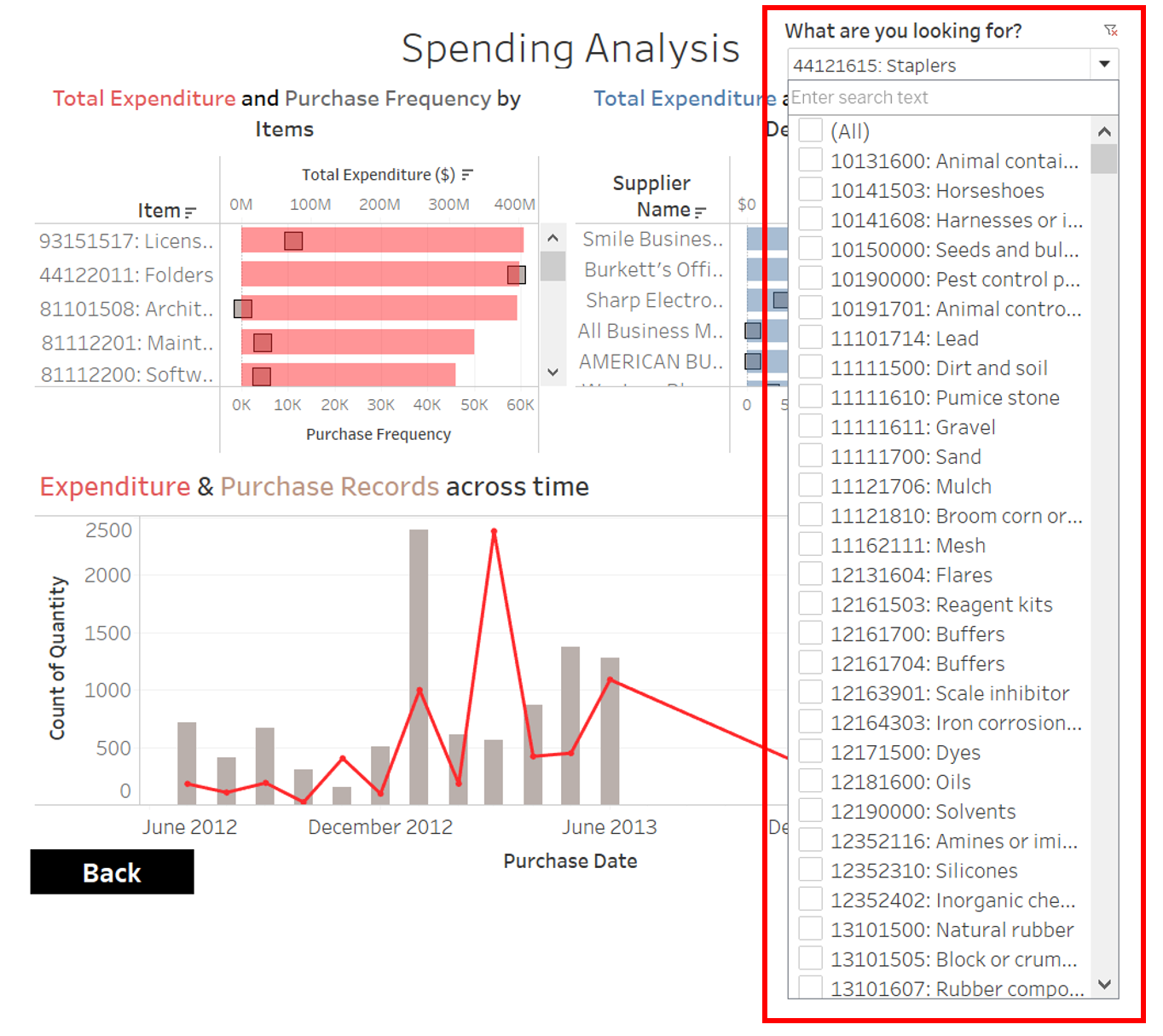


*Figure 17. Overview of Spending Analysis dashboard*

The first dashboard, Spending Analysis aims to:

1. Visualise order and price trends from past records, by item or by department
2. Plan for future procurement and budget as a department or as a company
3. Allow for potential inter-department collaborative procurement

On the top right, there is a dropdown filter that can filter the entire dashboard by item or its UNSPSC code *(refer to figure 18)*. Keying in part of the UNSPSC code will display all possible items with that code, allowing the user to find other items in the same family, which can be helpful if the user wants to look at close relatives of that item.

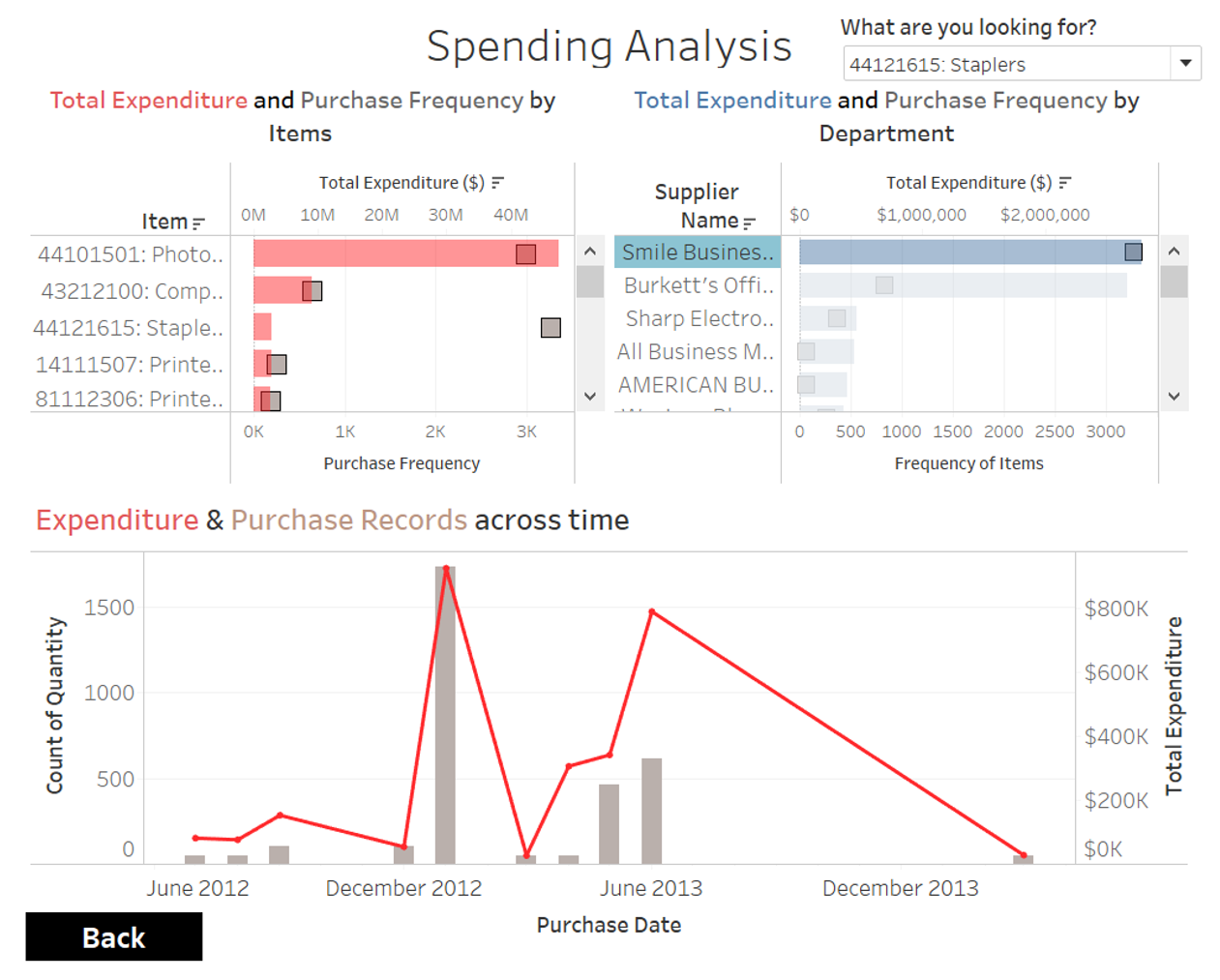


*Figure 18. Illustration of drop down filter*

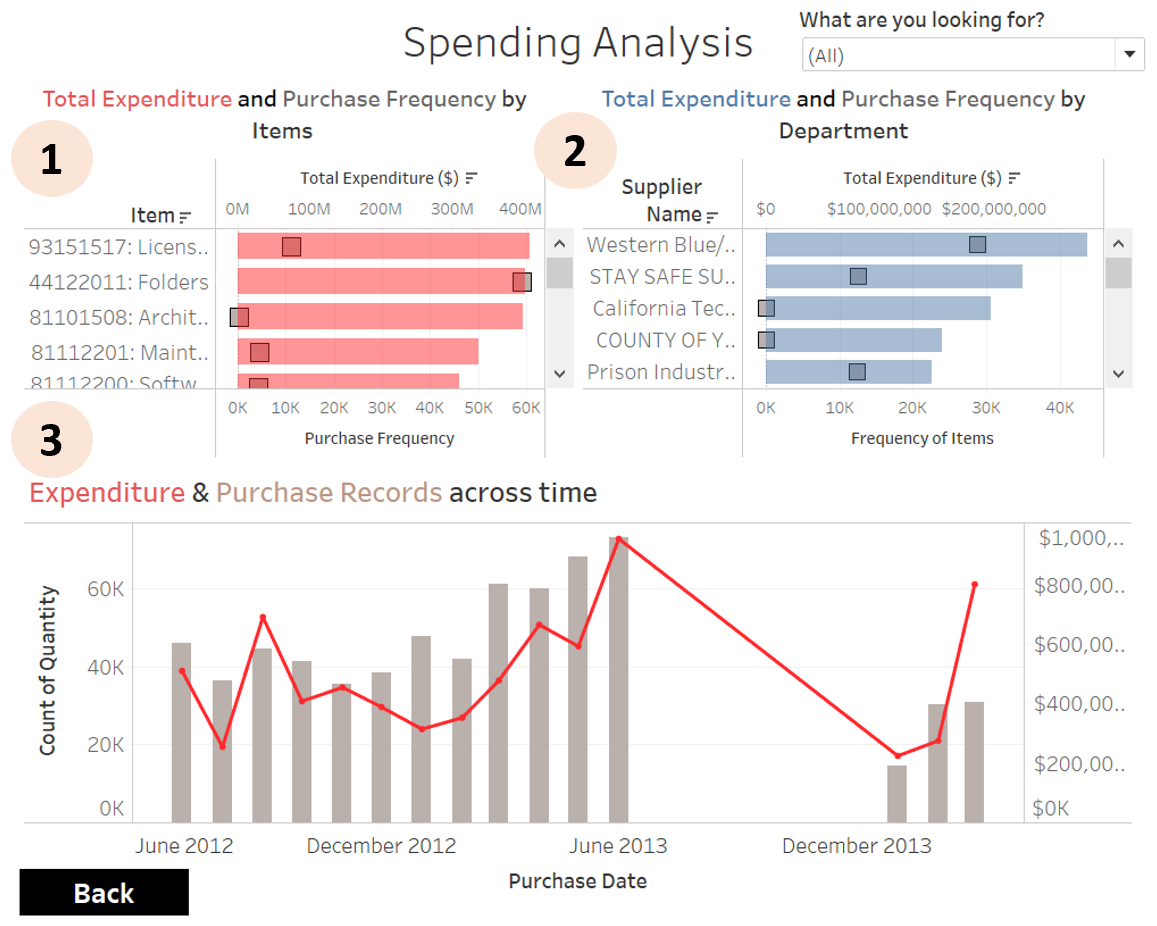
This dashboard is made up of 3 main visualizations. The top 2 dual-axis bar charts show the total expenditure and number of orders by items and department. The graph’s time element can be adjusted by various denominations along the ‘Purchase Date’ axis, providing the user with the flexibility to see trends across time periods such as quarters or months.

The dual-axis combination chart at the bottom shows the expenditure and purchase records across time for the entire company, showing the total expenditure on that item, and how frequent that item has been purchased at a particular time period, or across different time denominations.





*Figure 19. Expenditure and purchase records for an item by a department*



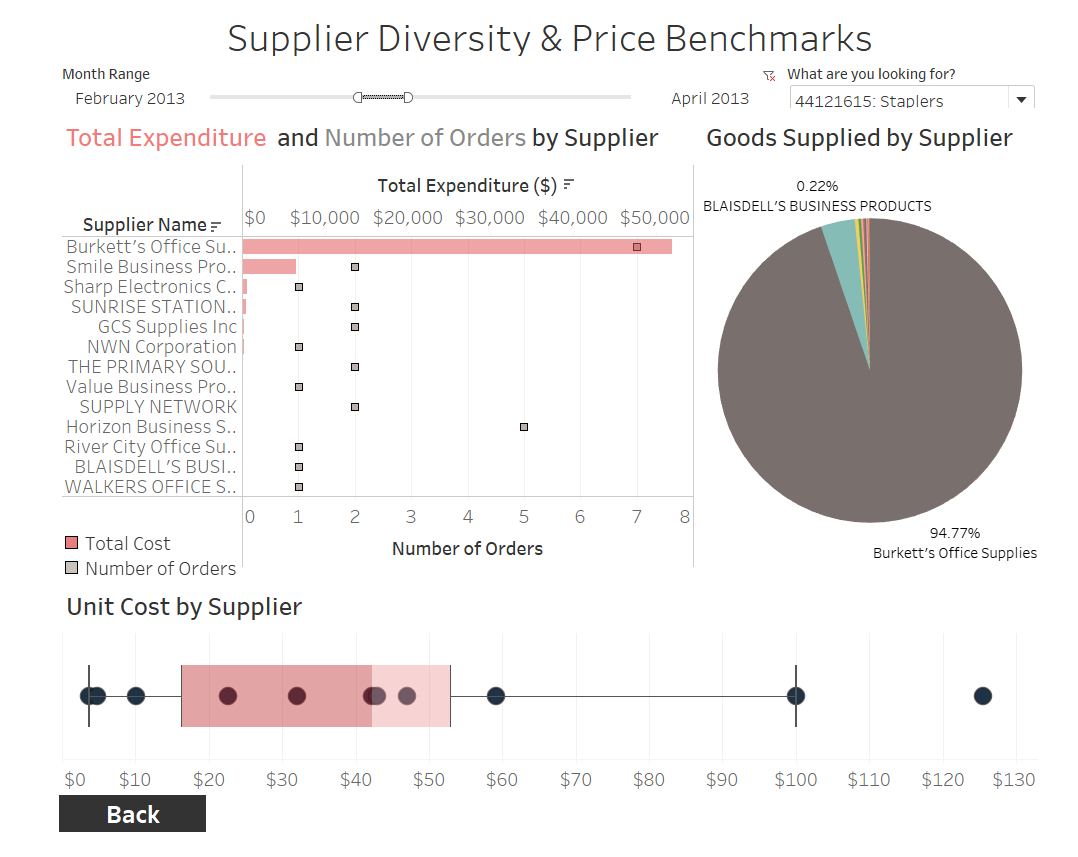
*Figure 20. Breakdown of Spending Analysis dashboard*

Clicking on the bars of any of these charts would result in the following filter process of the other charts *(refer to figure 20)*:

1. Clicking on the bar or square of the item graph would filter the combination chart for department and across time by that particular item. Since the items are sorted by total expenditure in descending order, items with more spending would appear first, telling the user which items should be focused on first in this procurement optimisation process.
2. Clicking on the bar or square of the department graph would cause the combination chart at the bottom to zoom in on that particular item and/or department. The item chart would also be filtered to show the items that that department purchased, in descending order of total expenditure. This would be useful in gathering insights on a specific department or high demand item by that department, looking at its order and price trends.
3. Clicking on the bar or line of the bottom chart would cause the department and item charts to display how much each department contributed to procuring items and how frequent items are purchased in that specific time period. This would be useful in focusing on a specific time period.

These actions will allow for potential inter-department collaborative procurement, as different departments can keep track of their fellow counterpart’s purchases.

## Supplier Diversity & Price Benchmarks



*Figure 21. Overview of Supplier Diversity & Price Benchmark dashboard*

The second dashboard, Supplier Diversity and Price Benchmarks aims to:

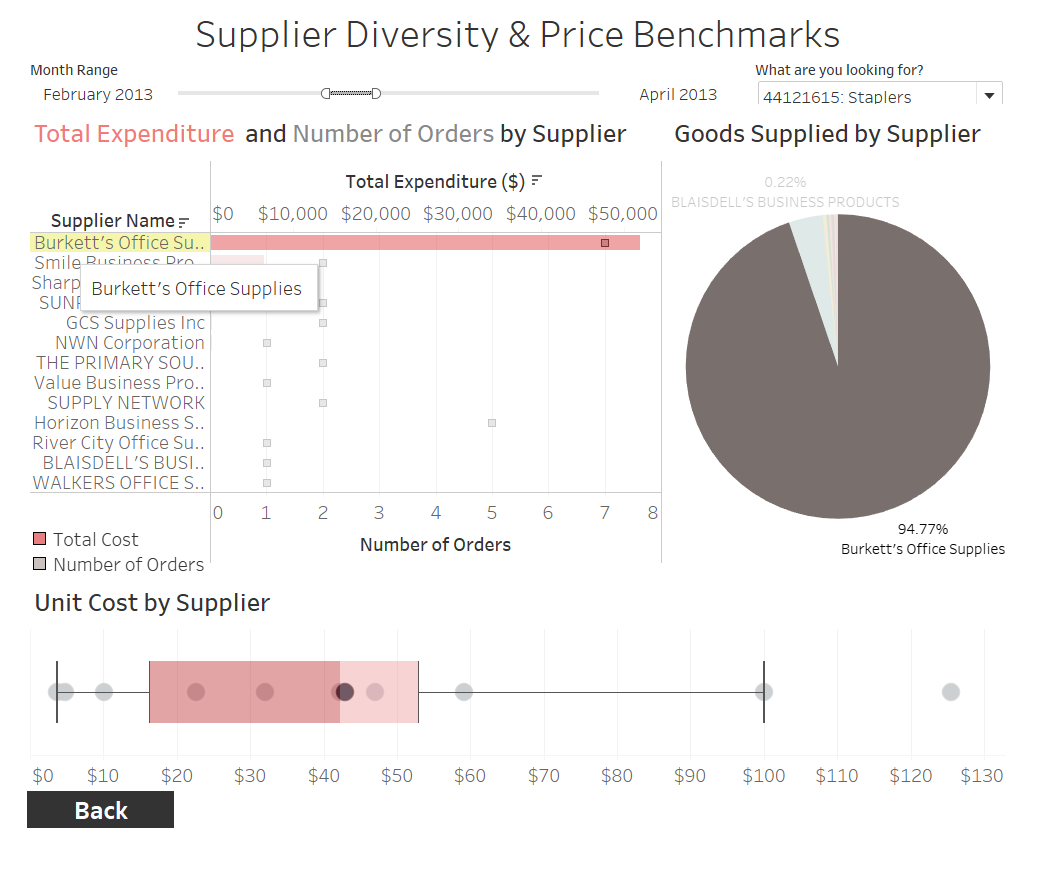
1. Compare unit prices between different suppliers, finding the best price for that item
2. Combine purchases into one bulk purchase, leading to economies of scale and lower prices

On the top right, there is the same dropdown filter to filter this dashboard by item or UNSPSC code while on the top left, a time filter is present to allow the user to decide which time period to look at. Adjusting this would change all the graphs respectively.

This dashboard is made up of 3 main graphs. The first graph is a dual-axis combination chart displaying the total expenditure and number of purchases made, for each supplier. If many purchase instances are observed for a certain supplier, this indicates the possibility of consolidating all these orders in the future, and purchasing a one time bulk order for that item, resulting in cheaper prices due to economies of scale.

The pie chart on the right shows the pool of suppliers supplying the item by proportion. This allows users to identify the suppliers which supplies the largest proportion of past purchases.

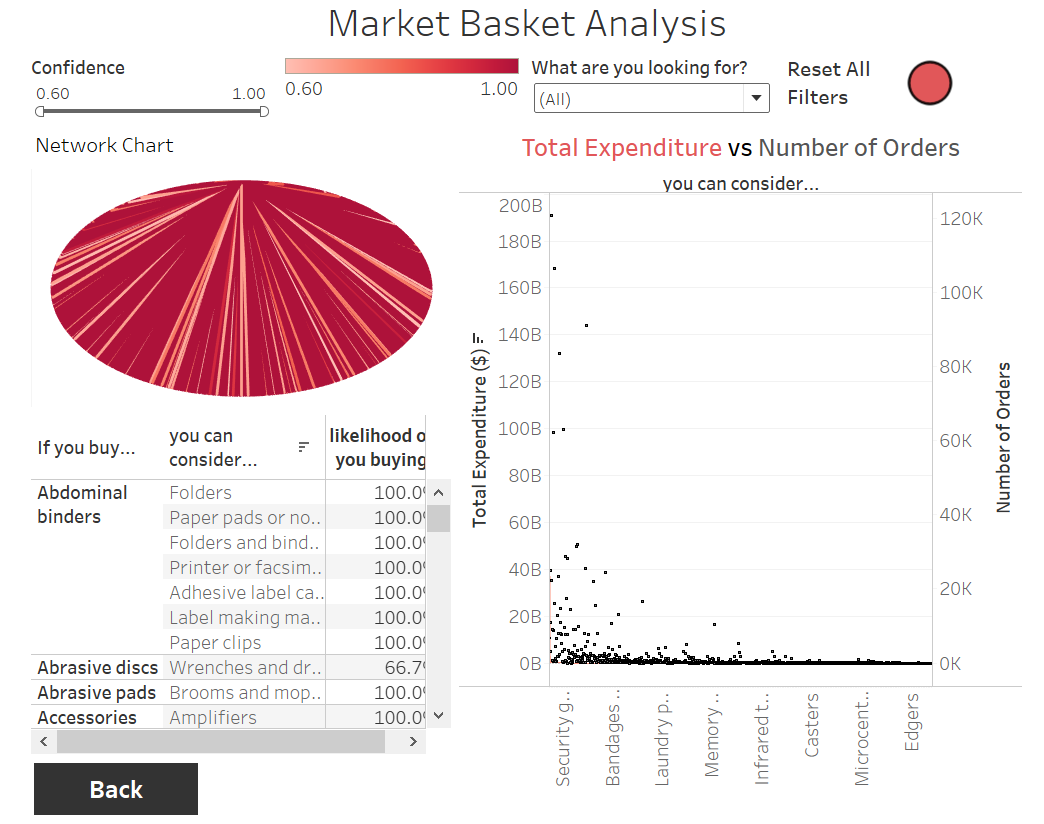
The box plot at the bottom shows the distribution of unit prices for that particular item. Each dot represents the price of that item supplied by its respective supplier. This boxplot can help the user gauge and compare prices of that item among the different suppliers. By looking at the quantiles or potential outliers, the user can tell if the supplier is offering a good price relative to the other price benchmarks.



*Figure 22. Illustration of filtering process in Supplier Diversity & Price Benchmarks dashboard*

Hovering over any bar, square, segment or point in the combination chart, pie chart or box plot respectively would result in highlighting the particular supplier, providing the user with the ability to focus on an individual supplier *(refer to figure 22)*.

## Market Basket Analysis



*Figure 23. Overview of Market Basket Analysis Dashboard*

The third dashboard, Market Basket Analysis aims to:

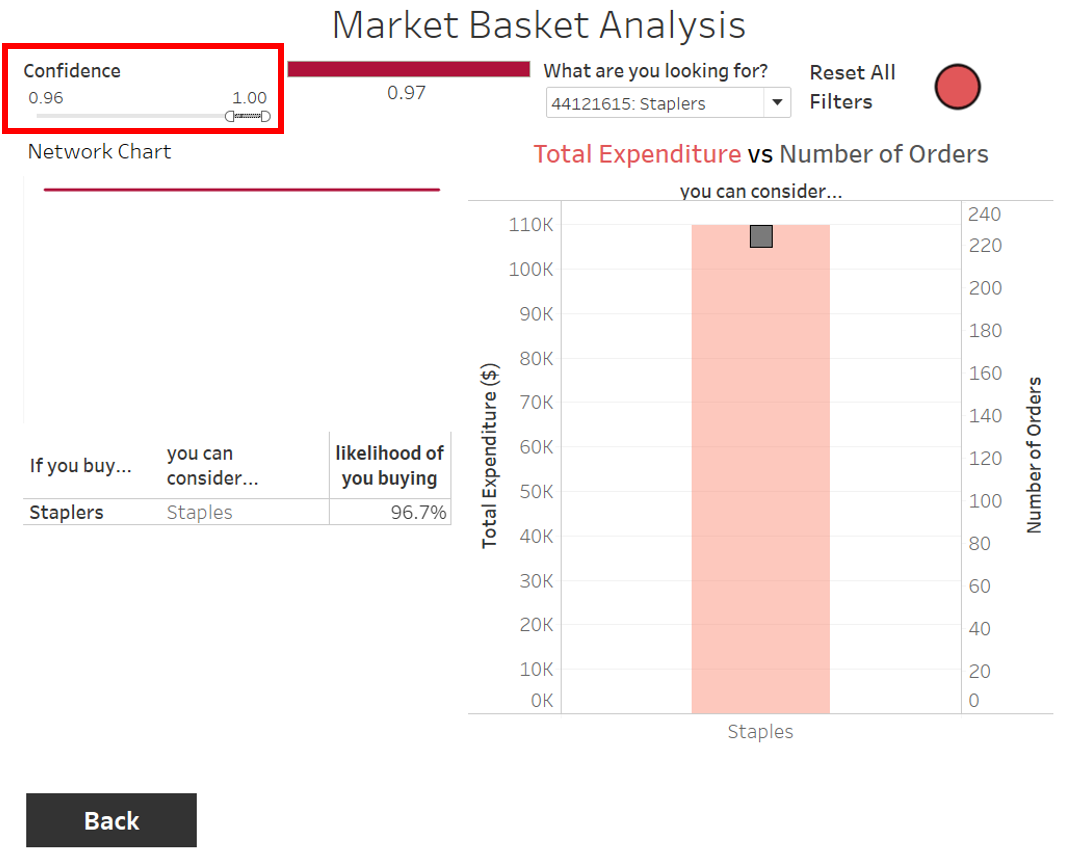
1. Provide purchase suggestions after association mining
2. Combine purchases of different items, leading to package deals and cheaper prices

On the top right, there is the same dropdown filter to filter the dashboard by item or UNSPSC code.

This dashboard is made up of 3 main graphs. The top left graph is the network diagram, providing an illustration of a network chart that is filtered by that item. The colour intensity indicates the strength of the association between the 2 items. In this instance, darker streaks of red indicate a stronger association between the 2 items. Users can hover over each line to see the associated items.

The bottom left shows the association table, depicting the information of the network diagram in tabular format and inclusive of the percent of likelihoods. This table serves to provide purchase suggestions by showing the associated items.

On the right, we have a dual-axis combination chart showing the total expenditure of every associated item, and the number of purchases made for both items. With this plot, users can prioritise the more important items which may be purchased very frequently or contribute the most to the total expenditure. From here, users can look to see if they include these items in package deals to reduce the expenditure on these items.



*Figure 24. Illustration on Confidence slider*

The likelihood slider at the top left allows the user to filter for a range of desired likelihoods. All the visualizations in this dashboard will change according to this slider.

Lastly, pressing the reset button on the top right will reset all filters to its original setting. This provides convenience for the user to look for different items.

# Future Directions

*Threshold Analysis/ Tuning*

In the project, thresholds were chosen by random sampling and weighed between time and accuracy improvements due to time constraints to maximise the marginal benefits. However, the tuning of these threshold values should warrant further investigation to see the best performance as they directly impact the reliability of the cleaning process.

*Consistency in Results*

As from the results, it is evident of a slight fluctuation of accuracy across the different stages of the project. Coupled with the F1 score being consistent slightly below the accuracy implying the failure of prediction in classes with lower frequency counts, it is worth investigation on the improvement that Snowball effect will bring after many Data Cycles. While it is reasonable to assume that as more datasets are clean and trained upon, the accuracy will improve, it remains inconclusive.

*Usage of more Complex Models*

Due to computational complexity on a huge dataset and limited hardware resources, the use of state of the art methods are not thoroughly explored. It remains to see if such complex methods will work, for the better or worse, in terms of accuracy of predictions. A survey of more variety of methods can be performed for model selection.

# Conclusion

This project seeks to address the urgent need for DSTA to increase the efficiency and effectiveness of their procurement process. 2 deliverables were presented; a text classifier and a dashboard. The former seeks to automate the manual labelling of procurement dataset to reduce the strain on human resources while the latter uncovers insights to make more informed decisions. Together, both can be seen as different dimensions to tackle the same objective; to reduce cost. Throughout the procedure, 2 key ideas were also proposed. A Data Cycle framework is pushed forward in an attempt to efficiently clean the dataset and potentially expand the clean training data size for more accurate predictions in the future. Apriori Algorithm is also used in association mining to suggest relationships that are otherwise less obvious. While there are many assumptions made and faced constraints in time and resources, all are carefully weighted and made with reasonable justifications. Future directions could involve a test on their validity.

# Key Takeaways

Throughout the project, there were several key takeaways:

1. Behind every dataset, much work has to be done to ensure that the model is trained with clean and quality data. The first half of the timeframe was delegated to focus on manually tagging the first year of procurement data, which consist of approximately 100,000 rows. This gruelling process is testament to the importance of data cleaning, offering a new found perspective on this field of work.
2. Accuracy, while important, is not the sole consideration in deciding the final model. In the real working environment, there exists a need to weigh the pros and cons. In this instance, time complexity and computational power that the models require as well as model performance are considered. For instance, the SVM model though produced better forward prediction accuracies, it would not be practical due to the long training hours. As such, with real-world constraints and limitations, a balance between the fine line of model efficiency and performance must be striked.
3. A potentially better classifier could be proposed should the project timeline be lengthened. Acknowledgement must be made that after every forward prediction, the right protocol is to manually check every single row to ensure that the items are correctly classified, rather than setting a threshold to reduce the number of checks. The impact on the model accuracy should be thoroughly assessed with this proposed change in protocol.
4. Much thought and discussions are put into designing dashboards to ensure that the client’s objectives are met. Putting ourselves in the shoes of users, thinking of what insights user’s will want to gather from the dashboards, and most importantly whether the dashboards are easily interpretable, user-friendly and interactive enough for a layman to use.

# References

Aiyappan, J. (2019, September 8). Naive Bayes Classifier for Text Classification - Analytics Vidhya. Medium. <https://medium.com/analytics-vidhya/naive-bayes-classifier-for-text-classification-556fabaf252b#:%7E:text=The%20Naive%20Bayes%20classifier%20is,time%20and%20less%20training%20data>.

Kowalczyk, A., & Kowalczyk, A. (2020, July 26). Linear Kernel: Why is it recommended for text classification ? SVM Tutorial. <https://www.svm-tutorial.com/2014/10/svm-linear-kernel-good-text-classification/#:%7E:text=Text%20has%20a%20lot%20of,features%20(words)%20are%20large>.

Chen, L. (2019, January 7). Support Vector Machine — Simply Explained - Towards Data Science. Medium. <https://towardsdatascience.com/support-vector-machine-simply-explained-fee28eba5496>

Jurafsky, Martins (2019, October 2) Chapter 5 Logistic Regression

<https://web.stanford.edu/~jurafsky/slp3/5.pdf>

Djaja, F. (2020, June 9). Multi Class Text Classification with Keras and LSTM. Medium. <https://djajafer.medium.com/multi-class-text-classification-with-keras-and-lstm-4c5525bef592>