Masters of Technology (Intelligent Systems)
Intelligent Reasoning Systems

Project Report

Order Forecasting System

Team Members

Yap Pow Look (A0163450M)
Chow Kok Peng (A0195403H)
Nguyen Minh Tien (A0229981N)
Tan Nicole Ongoco (A0229980R)

2nd May 2021

CONTENTS

I. EXECUTIVE SUMMARY ... 3
II. SPONSORING COMPANY INTRODUCTION ... 4
III. BUSINESS PROBLEM BACKGROUND ... 6
IV. MARKET RESEARCH ... 7
V. PROJECT OBJECTIVES AND SUCCESS MEASUREMENTS ... 8
VI. PROJECT SOLUTION ... 9
VII. PROJECT IMPLEMENTATION ... 10
1. Data Acquisition ... 10
2. Data Exploration ... 10
3. Data Preprocessing ... 12
4. Data Modelling - Forecasting ... 14
5. Data Analysis - Association Mining for Recommendations ... 19
6. Database ... 24
7. Web Application ... 27
8. Dockerization and deployment ... 28
VIII. PROJECT PERFORMANCE AND VALIDATION ... 30
IX. PROJECT CONCLUSIONS ... 35
Findings ... 35
Recommendations ... 35
ANNEXURES ... 36
Appendix 1 – Project Proposal ... 36
Appendix 2 – Team Formation & Registration ... 39
Appendix 3 – Mapped System Functionalities ... 40
Appendix 4 – Installation and User Guide ... 42
Appendix 5 – Individual member's report ... 51
Report by : Yap Pow Look ... 51
Report by : Chow Kok Peng ... 52
Report by : Minh Tien ... 54
Report by: Tan Nicole Ongoco ... 55
Appendix 6 – List of Abbreviations ... 56
Appendix 7 – Questionnaire on Forecasting PoC ... 57
Appendix 8 – Questionnaire on Feedback for Forecasting PoC ... 59

I. EXECUTIVE SUMMARY

BandLab Technologies is a collective of global music brands that delivers authentic content, products and experiences for all music lovers. Besides their flagship digital product, BandLab - a social music platform that enables creation, collaboration and sharing of music, the company also focuses on retail through its several eCommerce websites.

BandLab's eCommerce team currently engage in manual forecasting processes which relies mainly on Excel spreadsheets. These processes are not only onerous and time-consuming, but also inaccurate. To stay competitive in a modern and saturated eCommerce environment, the company is looking for a solution which would improve the forecasting process. Not only would this save time, but it would also lead to several opportunities, such as minimizing unsold inventory, improving availability of products, and minimizing cost of procurement.

Team Telescope's objective will thus be to develop a web application which will ingest the company's existing data, and generate not only forecast orders, but also market basket analysis results for data discovery.

The models will be run monthly using a cron job to retrieve, process, predict and store the data.

The models used in the project for forecasting consist of a basic rolling moving average, exponential smoothing, statistical time-series models using AutoRegressive and Moving Average (ARIMA) parameters and a deep learning model using a hybrid CNN-LSTM implementation. The performance of these models are measured and the results compared to show which models work better for most items.

We also did association mining (Market Basket Analysis) to assist the company to:

a. Bundle commonly bought together products for marketing campaign & promotions
b. Send recommendations to customers post sales to entice repeat order

Feedback from the company management has been positive and they are interested to continue to pursue using the models we have identified.

In this PoC, we have therefore completed the following:

1. Develop a web application which can be accessible from anywhere with access control
2. Automate a pipeline from data ingestion to final results for forecasting
3. Provide several models for forecasting and select the best models for the forecasting
4. Develop a Marketing Basket Analysis for product recommendations

II. SPONSORING COMPANY INTRODUCTION

Established in 2016 and headquartered in Singapore, BandLab Technologies is a collective of global music brands with a vision to connect the world of music. This includes BandLab, the group's flagship digital product – a social music platform (https://www.bandlab.com/) that enables creators to make music – and share the creative process with musicians and fans. BandLab has been adopted by over 25 million people worldwide since its inception in 2015. The music platform has a collection of sounds (called BandLab Sounds) which can help musicians to create their music.

Fig. 1 - BandLab's brands[1]

Besides the music platform, BandLab also has several eCommerce platforms. Collectively, the company handles over 150+ brands and more than 16,000 products across several countries, through it's 5 eCommerce websites:

Swee Lee	Founded and headquartered in Singapore, it carries over 150 brands and 16,000 products across 10 stores in Singapore, Malaysia, Indonesia and Vietnam, with an e-commerce platform delivering the Swee Lee experience to online shoppers throughout Asia.[2]
MONO	Founded in San Francisco, California and now headquartered in Singapore, MONO is a globally recognised, IDSA award-winning brand that's synonymous with product innovation and creativity for musicians, creators and the creative community.[3]

1 https://bandlabtechnologies.com/
2 https://bandlabtechnologies.com/brands/swee-lee/
3 https://bandlabtechnologies.com/brands/mono/

Harmony	Harmony carries a refreshed lineup of electric guitars, amplifiers and accessories that are inspired by the past, designed for the present and crafted to last into the future.[4]
Heritage	Since 1985, Heritage Guitars has been crafting the finest American-made guitars from the famed 225 Parsons Street factory, in the heart of Kalamazoo, Michigan.[5]
Teisco	Teisco started out in 1948 as a Japanese manufacturer of affordable musical instruments, establishing an early reputation as a leader in electronic innovation and unconventional aesthetics.[6]

[4] https://bandlabtechnologies.com/brands/harmony/
[5] https://bandlabtechnologies.com/brands/heritage/
[6] https://bandlabtechnologies.com/brands/teisco/

III. BUSINESS PROBLEM BACKGROUND

With over 150 brands and 16,000 products, forecasting of product sales can be an onerous task. Currently forecasting is done on a monthly basis and is done using Excel spreadsheets[7]. It is time consuming and often not as accurate as expected.

The company would therefore like to develop a forecasting model which will be better than the current model, which can be adopted by the company for planning item procurement quantity. The solution can potentially be used across 5 different e-commerce platforms. Our project proposal was therefore to search and develop a model which BandLab can adopt for use, minimizing both man-power and time spent on procurement planning.

Our web application will be a Proof of Concept which we will demonstrate the accuracy of the models we have tried and tested. As such, we will not be forecasting the entire range but rather the top 20 items in one of the product channels. This model, if adopted, will be able to be scaled to the rest of the product channels.

[7] Appendix 7 - Questionnaire on Forecasting PoC

IV. MARKET RESEARCH

Companies operate in a hypercompetitive environment where there are threats from competitors, uncertain market conditions and uncertain customers' demand. More and more companies are making use of predictive analytics models for exploiting the patterns in transactional and historical data to predict the future demand with higher accuracy.

Building predictive analytics models helps the company to better match its supply and demand. The predictive model helps the company to make intervention decisions at the optimum junction balancing between minimum stock out and minimize depreciation cost of carry stock.

Specifically for the retail industry, it includes the process of stocking the right products, promoting the right products to the right customers, providing the most optimal discounts to persuade sales, having the right strategy for marketing and advertising.

Techniques used in the market are the evergreen linear regression, polynomial regression and other machine learning regression methods such as decision tree and so on.

Benchmark against the pertinent technique of using moving average to predict the future demand, we are exploring methods in time series analysis, deep learning (LSTM) with intentions to achieve higher prediction accuracy.

V. PROJECT OBJECTIVES AND SUCCESS MEASUREMENTS

Our project proposal will be to search and develop a model which the company can adopt for use, minimizing both manpower and time spent on procurement planning. The following are the objectives:

Objective 1:

The system will be a complete pipeline from ingesting raw data from the database and writing back processed data back to the DB. The web application will be able to draw from the processed database to display and to decide.

Objective 2:

To develop several forecasting models and select the best for the company to use for their application. The recommended model will be selected based on the lowest Maximum Absolute Error (MaxE) and the lowest Root Mean Squared Error (RMSE).

Objective 3:

The objective is to extract insights from the historical transaction data to provide recommendations for products bundling during marketing promotions and making post sales recommendations to previous customers. We will make use of associative mining techniques (market basket analysis) to accomplish this objective.

VI. PROJECT SOLUTION

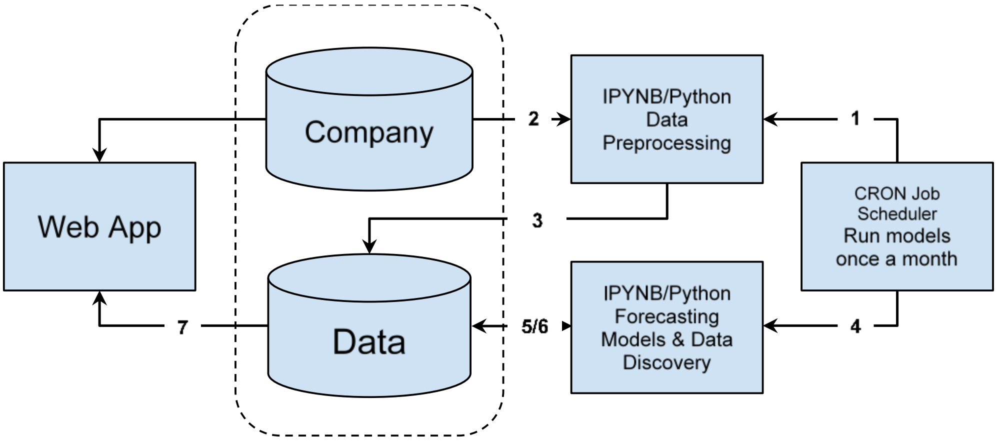


Fig 2 - System Architecture

The above diagram shows the overall system architecture. The components and the interactions between them are detailed below.

1. **Database**
   The database contains company data, processed data, and model results. Database tables, fields, and mappings are further detailed under *Section VII - Project Implementation, 6. Database*

2. **CRON Job Scheduler**
   The Job scheduler is scheduled to run once a month, given that company order forecasting is not needed on-demand. Depending on the product brand, the company usually performs order forecasting activities once per month or per quarter. It is tasked to run the Python Scripts (Data Preprocessing, Data Forecasting, Data Discovery) at the said schedule.

3. **Python Scripts (Data Preprocessing)**
   The data preprocessing script reads the company data, and cleans the data into a format which can easily be used as input to our forecasting and data discovery models. A Ipynb file will be transform into a python script by using nbconvert

4. **Python Scripts (Data Forecasting and Data Discovery)**
   The solution consists of five (5) different data forecasting techniques - ARIMA, SARIMA, Rolling Moving Average, Exponential Smoothing and CNN-LSTM. For data discovery, Market Basket Analysis is done via item-item filtering.

5. **Web Application**
   The Web Application provides the interface to be used by Project Managers / Procurement Staff to view and export forecasts, as well perform data discovery.

VII. PROJECT IMPLEMENTATION

1. Data Acquisition

The dataset was made available to us from the company. As the company manages a number of eCommerce sites, the dataset is from one of the sites, MONO, which provide a comprehensive lineup of cutting-edge musical instrument cases, pedalboards and lifestyle products (https://monocreators.com). The data covers orders and products created from January 2018 to March 2021.

Interviews with the Subject Matter Expert were also conducted in order to gain insight on how the current forecasts were done, it's pain points, the requirements for the PoC, and validation of data [8].

2. Data Exploration

Exploration of the products table showed that there are 126 items in the product list. Further investigation showed that there are actually 6 items which are duplicated. These items were dropped which makes the number of unique product items to be 120.

Exploration of the dataset (from orders_products table)showed that there are 7545 rows which indicates 7545 individual items sold. The sales cover a period from Jan 2018 to Mar 2021.This is actually good as we can use the data from 2018-2020 for training and Jan-Mar 2021 for forecasting.

The dataset was then groupby product_sku and the number of sales items summed up. The groupby showed that the top items sold in 3 years was 813 while the top 20th items sold was 309 items. From here we can see that as the number of items are large, it does not make sense to forecast all the items. We thus focus on forecasting the top 20 items sold for the period

product_sku	EFX-FLY-BLK	M80-2B-BLK	M80-2G-BLK	M80-AC-BLK	M80-AD-BLK	M80-BTY-BLK-L	M80-BTY-BLK-S	M80-EB-BLK	M80-EG-BLK	M80-SEB-BLK	M80-SEG-ASH	M80-SEG-BLK	M80-TICK-V2-BLK	M80-TOUR-V2-BLK	M80-VAD-BLK	M80-VEB-BLK	M80-VEB-GRY	M80-VEG-BLK	M80-VEG-GRY	M80-VHB-BLK
row_0																				
2018-01-31	16	12	22	12	15	11	8	11	17	27	12	15	25	16	14	8	15	33	5	18
2018-02-28	24	9	13	10	5	12	3	22	24	14	15	13	24	22	6	6	12	7	3	7
2018-03-31	21	6	7	0	16	14	6	13	14	2	4	11	16	14	6	14	5	6	9	18
2018-04-30	39	25	37	22	22	26	20	26	35	28	31	31	35	19	25	35	17	100	29	38
2018-05-31	27	10	23	11	17	14	6	13	21	10	14	13	18	18	9	17	12	19	4	14
2018-06-30	37	13	16	6	8	11	15	19	16	7	8	10	19	8	22	20	8	18	13	20
2018-07-31	34	8	21	17	14	12	14	12	19	9	10	9	17	9	11	16	10	14	13	14
2018-08-31	12	6	8	6	7	8	8	3	10	6	7	4	10	6	8	15	6	21	3	12
2018-09-30	22	4	3	4	3	7	1	4	3	3	2	0	13	3	4	7	2	5	2	8
2018-10-31	14	2	7	2	4	3	3	1	6	5	0	0	11	2	1	17	5	5	3	6

Fig 3 - Partial table of top 20 items sold (Jan - Oct 2018)

[8] Appendix 7 - Questionnaire on Forecasting PoC

product_sku	EFX-FLY-BLK	M80-2B-BLK	M80-2G-BLK	M80-AC-BLK	M80-AD-BLK	M80-BTY-BLK-L	M80-BTY-BLK-S	M80-EB-BLK	M80-EG-BLK	M80-SEB-BLK	M80-SEG-ASH	M80-SEG-BLK	M80-TICK-V2-BLK	M80-TOUR-V2-BLK	M80-VAD-BLK	M80-VEB-BLK	M80-VEB-GRY	M80-VEG-BLK	M80-VEG-GRY	M80-VHB-BLK
row_0																				
2020-11-30	4	3	5	0	1	2	1	0	1	0	1	1	0	0	3	5	1	2	3	0
2020-12-31	2	3	5	7	1	0	4	2	0	2	2	1	3	1	6	8	0	6	2	1
2021-01-31	2	1	5	0	3	3	0	3	2	0	1	2	4	1	3	2	0	2	3	0
2021-02-28	2	1	4	0	1	1	2	1	0	3	1	0	3	1	0	3	1	3	2	4
2021-03-31	6	1	5	3	0	1	2	1	3	0	1	1	5	2	0	2	0	2	0	3

Fig 4 - Partial table of top 20 items sold (end 2020 - Jan 2021)

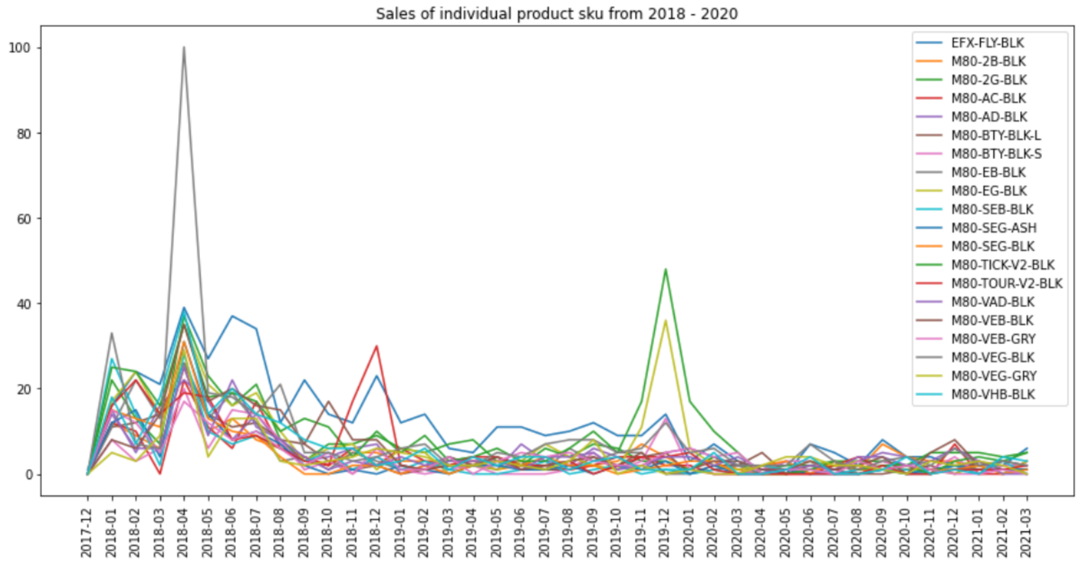


Fig 5 - Trend of product sales from 2018 - 2021

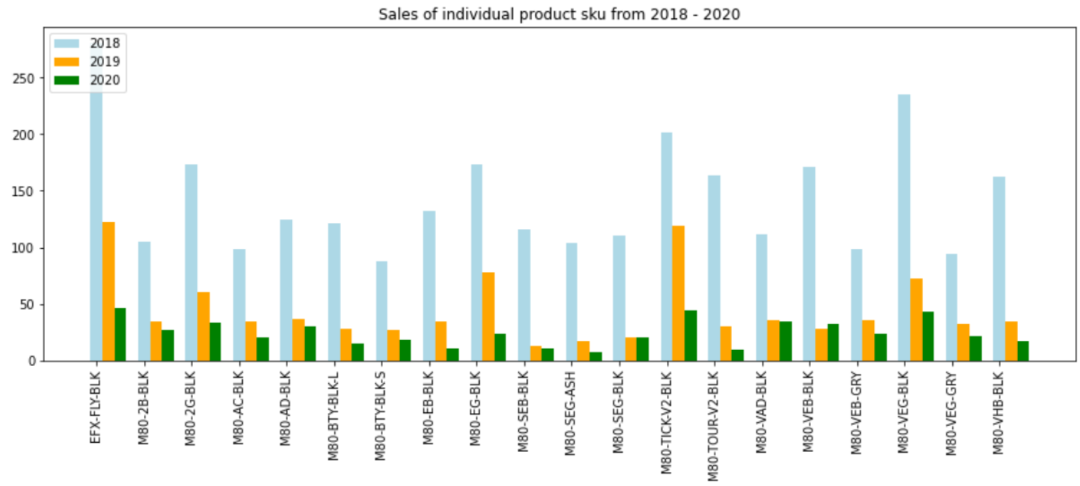


Fig 6 - Sales of top 20 items from 2018 - 2020

From the data, we note some seasonal peaks. However, we also noted that there is a downward trend in most of the products. There was a significant drop in sales from 2018 to 2019. It dropped further in 2020. This was expected since 2020 is a pandemic year and most retailed items have also dropped quite drastically. This is especially for bags and pedalboards which are used by travelling musicians. If they are not travelling, they will most probably not need to buy any.

3. Data Preprocessing

When reading from the raw data, there are some columns that need to be formatted and some entries need to be fixed if they are null value.

a. Data analyzing

First, we will take a look into the raw data and analyze it. Data are be stored as different format types and may not match with the AI model which is used to process it in the next phase. Another factor impacted to the final result is null or empty row data. Preprocess phase will ensure that the data will fit to the AI model and provide good results.

```
RangeIndex: 7618 entries, 0 to 7617
Data columns (total 25 columns):
 #   Column                             Non-Null Count  Dtype
---  ------                             --------------  -----
 0   order_id                           7618 non-null   int64
 1   order_created_at                   7618 non-null   datetime64[ns]
 2   closed_at                          7326 non-null   datetime64[ns]
 3   processed_at                       7618 non-null   datetime64[ns]
 4   order_total_price                  3840 non-null   float64
 5   financial_status                   7618 non-null   object
 6   order_status                       7618 non-null   object
 7   refunds                            7618 non-null   object
 8   order_discounts                    7618 non-null   object
 9   customer_id                        7580 non-null   float64
 10  customer_created_at                7580 non-null   object
 11  customer_country                   7580 non-null   object
 12  line_items.title                   7618 non-null   object
 13  product_quantity                   7618 non-null   int64
 14  product_sku                        7610 non-null   object
 15  line_items.grams                   7618 non-null   int64
 16  line_items.price                   7618 non-null   object
 17  product_discount                   7618 non-null   object
 18  line_items.fulfillment_status      7230 non-null   object
 19  line_items.discount_allocations    7618 non-null   object
 20  order_date                         7618 non-null   datetime64[ns]
 21  product_price                      7618 non-null   float64
 22  product_title                      7594 non-null   object
 23  product_category                   7594 non-null   object
 24  product_tags                       7594 non-null   object
dtypes: datetime64[ns](4), float64(3), int64(3), object(15)
```

Fig 7 - Raw data columns

There are 25 columns in varying format types; some of them have a lot of null data (order_total_price only has 3840/7618 non-null data). The rest of the fields have a high non-null data rate. As the below table, most fields have full non-null data. There are 7618 entries in total in the data table and some null values there. There are some columns that have json format, which we need to run some further steps to flatten them into the final data tables.

b. Fix missing data

The above table is showing that there are some null values in the data tables. Almost all the columns have no non-value (all entries have data), the other columns which have null data will be dropped later in case they are not invoked in the forecast models.

c. Flatten data

Some columns in the raw data table are in JSON format. They are present as objects containing nested data. They are `customer`, `line_items`, `fulfillments`, `refunds`, and `discount_applications`. Depend on the meaning of them, the preprocessing phase will treat them differently.

Parsing JSON object data and add more columns into the data tables. In example an customer object which is `'{"id":149426569237,"created_at":"2018-01-01T17:23:26-08:00","country":"United States"}'` will be parsing into 3 new labels which are: `customer.id`, `customer.created_at`, and `customer.country`. New data will be added into data tables within new columns having their name same as the labels.

d. Transform data format

Some columns either do not contain too much value or are not in usable data format, they will be transformed to different formats which let AI models easier to use later.

For example, column `line_items.discount_allocations` is currently using JSON format, which is the same with the input of flatten data phase. However, as it is a less important column and may not be used in the AI models. The preprocessing process only transforms it from JSON format to a basic string.

e. Remove duplicated data

In the process to collect data, some information in the raw data tables is duplicated and will not need to process twice in AI models to avoid over complicated computing and even provide wrong data when some value is cloned, But, we need to verify the data first to make sure that we didn't remove useful information from the dataset.

In this dataset, we can see there are 2 columns which share a lot of the same information, they are `line_items` and `fulfillments`. The preprocessing needs to provide a function which can compare how many entries in these 2 columns share the same values and how many of them are different. After comparing them, the result has proved that they share more than 99% of the same values, which is that only 2 entries have different values. To minimum the effort to process the dataset, we will only keep one column to process later.

f. Drop useless data

Some data which is not used in AI, forecasting models. They will be dropped in the preprocessing phase to keep the data size small and save the performance of the system when processing them.

g. Enrichment data

The original data is not contained in one table, which also come from different sources and different tables. To process data with full features, we need to add a function which can help to join all the data together and extract them as the table format as we want.

In this project, the raw data is collected from different tables which are the order table and the product table. The preprocess will query data from both tables and join them together as the correct format which is needed from AI models.

4. Data Modelling - Forecasting

For the forecasting, it was considered if we are to forecast on a weekly basis or monthly. However looking at the data, weekly numbers are too small to forecast meaningfully. Forecasting yearly is too long and so monthly forecasts are considered appropriate. It is also noted that the company forecasts their sales on a monthly basis when ordering inventory.

The model for forecasting is done using 3 types of forecasting models:

- Moving Averages
- Statistical Models
- Deep Learning Model

a. Rolling Moving Average

The rolling moving average simply takes the average of the last 3 months to forecast for the following month.

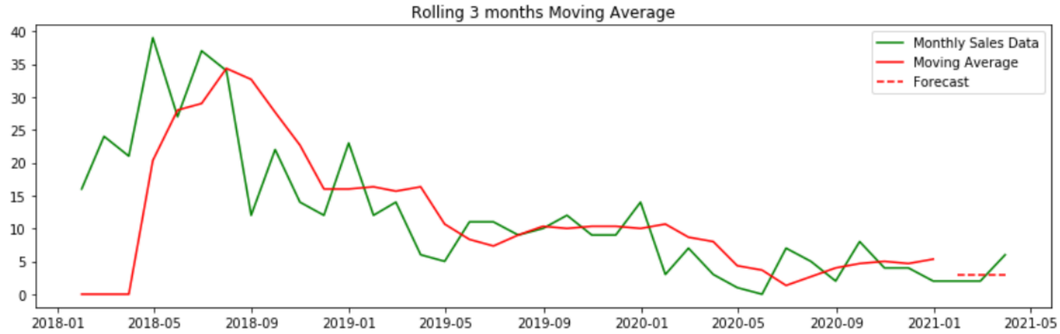


Fig 8: A 3 months rolling moving average for 1 item

From the moving average chart above, we can see the forecast(dotted red line) in comparison to the tested points. While moving average may not be able to forecast accurately for each month, it is possible that they can still average a small difference across 3 months.

b. Exponential Smoothing[9]

Rolling Moving Averages give equal weights to all the datapoints being used. In exponential smoothing, most recent observations are given more weights than distant data. It is intuitive as you will think the event the day before will have more impact than the event 2 days before. Double or triple exponential smoothing are employed that can take care of trend and seasonal movements.

The figure below shows the impact of exponential smoothings on the output of the predicted values.

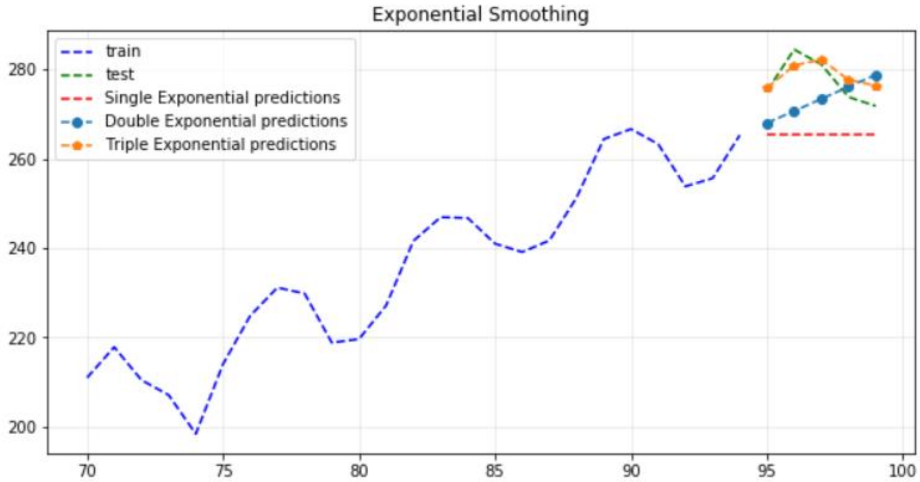


Fig 9: Exponential Smoothing

[9] statsmodels.tsa.holtwinters.ExponentialSmoothing — statsmodels

c. Statistical Models - Univariate Times Series

The statistical time-series are predominant forecasting models used in many use cases from sales to stock prices forecasting.

The TimeSeries model stated that a time-series can be decomposed to seasonal, trend and noise components. If a dataset can be decomposed properly with its seasonal and trend components removed leaving a noise component centred on the mean and variance 0, it is possible to forecast a time-series with some accuracy. While most models only work with one set of data, more complicated models using exogenous data in addition to the main dataset, are also being used in forecasting. For our model, we will only work with univariate data.

Decomposing a time-series

As shown in the diagram below, a time-series can be decomposed to its various components. The diagram shows that the series has a downward trend and has seasonal or cyclical changes across the months. This can be shown by the peaks at every Jan and highs around the month of June. A time-series can only be more accurately forecast if the residues have mean and variances around 0. More work needs to be done since the residues shown have a mean component. Often this may mean that there is a need to difference one time steps from the next to remove the mean component.

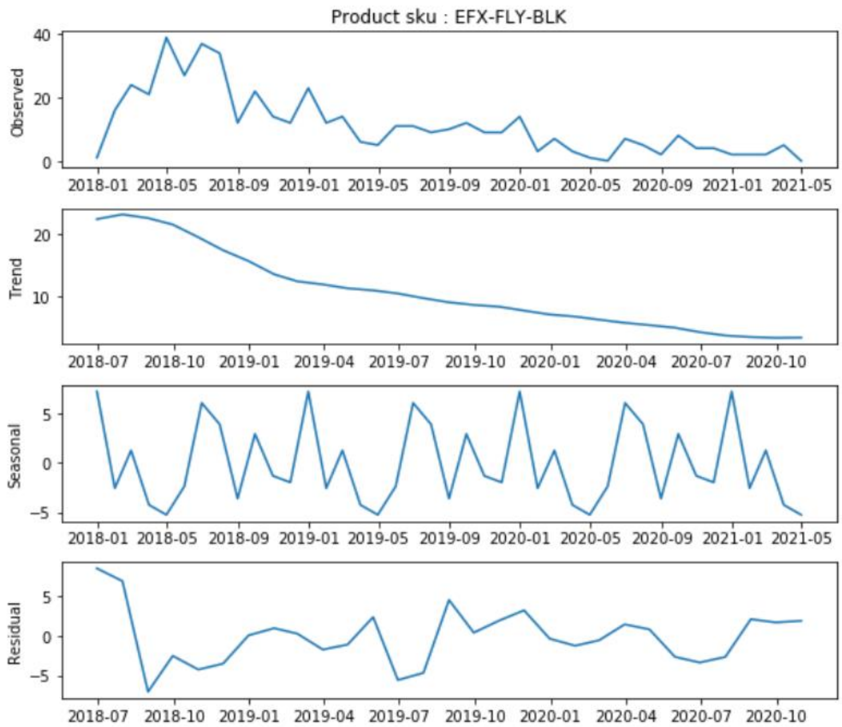


Fig 10 - Components of a time-series

Deriving Autocorrelation and Moving Average components

Besides decomposition, there is also a need to figure out if there is autocorrelation within the series. Autocorrelation is correlation of a time-series with itself, with a lag of one or more time-steps. Autocorrelation which is positive may imply that there is a tendency for an indicator to move up in the next time steps while a negative correlation implies the reverse. There is also a seasonal moving average component, MA which needs to be figured out. The MA tells how much influence the previous time steps have an effect on the noise component. Knowing this value will help in the forecasting of the time-series.

Using the ARIMA and SARIMA library for time-series forecasting

While understanding the reasons of how time-series is forecasted, many of these operations can be done using the ARIMA and SARIMA models from statsmodel library. The ARIMA model can take inputs for AR, I and MA. Deriving the AR, I and MA need not be manual as there are also libraries the the auto_arima package which can help to find the optimum, AR, I and MA for the time-series. Using these packages will help us to forecast the time-series.

d. Deep Learning Models - CNN-RNN

While the statistical time-series methods have traditionally been used for forecasting, increasingly, deep learning models are starting to be used. The advantage of using deep learning is the reduced need to re-compose or de-compose the data in order to predict. There is no need to derive the trend or the seasonal components. The complete dataset can just be plugged into the model to derive. The drawbacks is that deep learning works well with large datasets and is computationally intensive which of course is less of an issue these days.

In exploring using deep learning models, Recurrent Neural Networks (RNN) is the approach to take since it is designed to handle serial data. RNNs have been successfully used in natural language translation where each word coming in from one end is translated to another language at the output (Nowadays, another deep learning model called Transformers is the preferred model). Using RNNs, different configurations were tested out and it is found the following configuration gives the best results from the tests. That configuration will be discussed below.

CNN-LSTM[10]

In exploring RNNs, a version of RNN called LSTM is considered. RNNs have a problem of vanishing or exploding gradients. When gradients (or weights) are

[10] https://machinelearningmastery.com/how-to-develop-lstm-models-for-time-series-forecasting/

backpropagated, they have a tendency to continue to increase as it goes back the chain (exploding) or decrease (vanishing). As a result, a modified RNN model called LSTM is normally used.

Convolution Neural Networks (CNN), very popular for image classification have also been found to be very effective at extracting and learning features from one-dimensional sequence data like a univariate time series. 1D CNN can be used to discover features and patterns in fix length sequences like a time-series.

For our model, we use a hybrid CNN-LSTM model. For our configuration, a 1D CNN will be used to take sequence inputs from the time series. The 1D Dense Layer will consist of a 1D filter which will slide through the data to discover for patterns. This is passed to a hidden layer which is then pooled and flattened to be sent to the LSTM layer for additional processing.

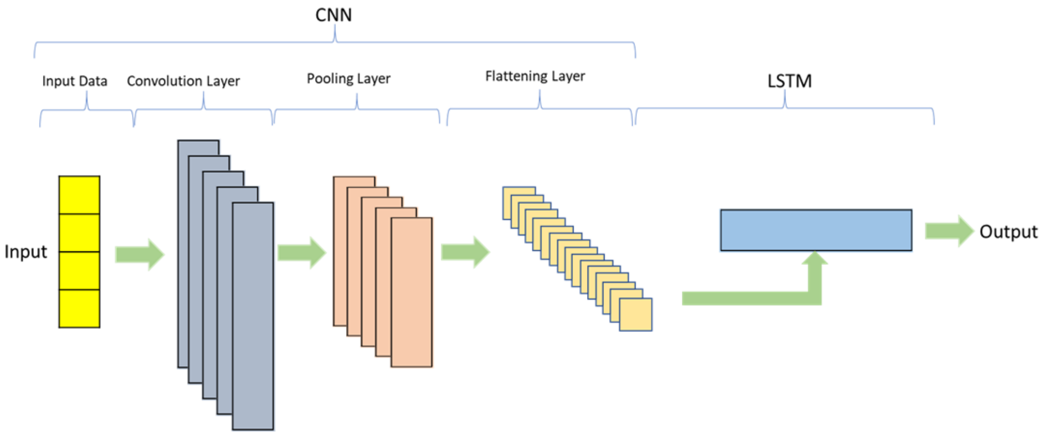


Fig 11: CNN-LSTM Configuration

In our selection, the model will use the last 4 months' data to forecast the following month's result.

The monthly sales of one of the top items eg, :
[16, 24, 21, 39, 27, 37, 34, 12, 22, 14, 12, 23, 12, 14, 6, 5, 11, 11, 9, 10, 12, 9, 9, 14, 3, 7, 3, 1, 0, 7, 5, 2, 8, 4, 4, 2, 2, 2, 6]

The sequence will be reshaped into the following shape for input into the model :

X	y
[16 24 21 39]	27
[24 21 39 27]	37
[21 39 27 37]	34
[39 27 37 34]	12
[27 37 34 12]	22
[37 34 12 22]	14

As shown above, the first 4 months of the sales data will be fed in to forecast the 4th, and the 2nd 3 months will be used to forecast the 5th etc.. For the prediction, the last

4 months of the data will be used to predict the first unseen month. We will use the walk forward method to predict the 2nd and 3rd month. Eg

X1, X2,X3,X4 predict y1 (mth+1)
X2, X3,X4,Y1 predict y2 (mth+2)
X3, X4,Y1,Y2 predict y3 (mth+3)

5. Data Analysis - Association Mining for Recommendations

Based on the current analysis, proceeded to do an item based recommendation.

The objective is to provide a recommendation via Market Basket analysis based on previously bought items. The company can make use of this recommendation to:

1. Send a suggestion via email a few days after initial sales or,
2. Package the frequently sold together items as a combo as part of a marketing campaign.

Proceeded to perform item based recommendation analysis on three configurations of the data set:

a. Only multiple products in single order data set.
b. Both multiple and single product purchase, aggregated over the entire dataset.
c. Addition of virtual items to experiment if it improves the quality of the recommendations.

a. First Analysis: Frequent items for multiple products in single order data set.

In the first analysis of multiple products bought in a single order, the top products that appeared in the data set are plotted out for better understanding the product composition in the data set.

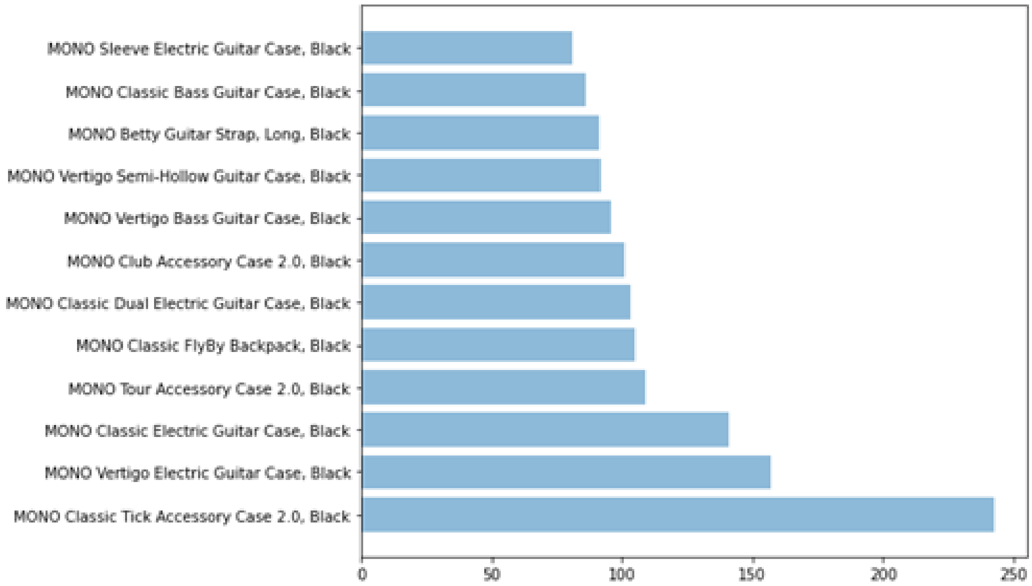


Fig 12 - Frequent items from "multiple items in single order" data set

From the chart, it is observed that the most popular item is the Classic Tick Accessory, 200+ transactions out of 3000+ rows of data, around 7%.
The rules generated for the top 10 items are:

Confidence	LHS	RHS
0.92	{'MONO Tour Accessory Case 2.0, Black'} <=	{'MONO Pedalboard Medium, Black'}
0.82	{'MONO Club Accessory Case 2.0, Black'} <=	{'MONO Pedalboard Small, Black'}
0.79	{'MONO Classic Tick Accessory Case 2.0, Black'} <=	{'MONO Pedalboard Lite, Black'}
0.65	{'MONO Classic Tick Accessory Case 2.0, Black'} <=	{'MONO Classic Electric Guitar Case, Black'}
0.50	{'MONO Pedalboard Medium, Black'} <=	{'MONO Tour Accessory Case 2.0, Black'}
0.46	{'MONO Classic Tick Accessory Case 2.0, Black'} <=	{'MONO Vertigo Electric Guitar Case, Black'}
0.41	{'MONO Pedalboard Small, Black'} <=	{'MONO Club Accessory Case 2.0, Black'}
0.36	{'MONO Classic Electric Guitar Case, Black'} <=	{'MONO Classic Tick Accessory Case 2.0, Black'}
0.22	{'MONO Vertigo Electric Guitar Case, Black'} <=	{'MONO Classic Tick Accessory Case 2.0, Black'}
0.16	{'MONO Pedalboard Lite, Black'} <=	{'MONO Classic Tick Accessory Case 2.0, Black'}

Top 10 rules generated with only multiple products in single order dataset

Discussions

This recommendation might be less accurate due to a smaller data set, however due to the items used in the analysis are only selected from those items that are bought together, the confidence of the recommended items is actually higher.

Application of this: This recommendation is suitable for packaging items together for combo packaging sales. This is applicable even before first sales, so this is appropriate for new customers as well.

b. 2nd analysis, Frequent items for both multiple and single item purchase, aggregated over the entire dataset

In the second analysis of both single item & multiple products aggregated for the entire data set, the top products are plotted out for understanding the data. The top products plotted out :

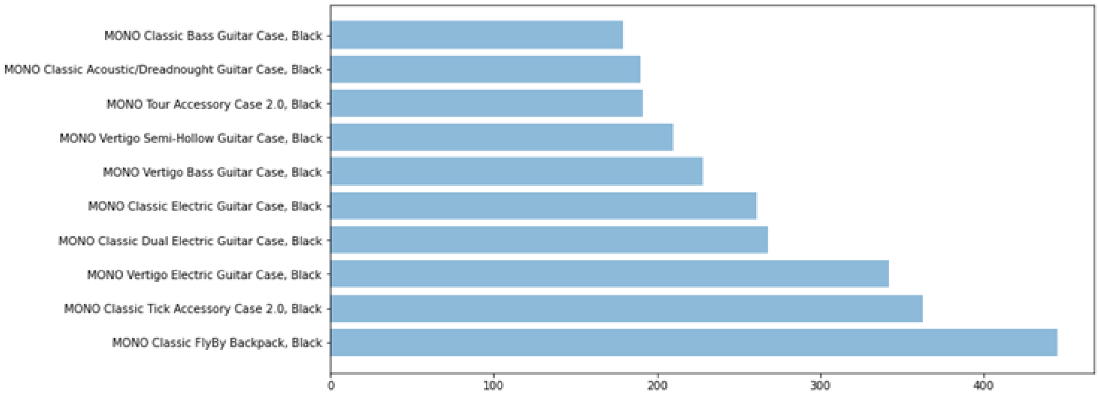


Fig 13 - Frequent items from all transactions data set

The most popular products from the analysis of the entire data set are slightly different from the previous analysis.

Confidence	LHS	RHS
0.83	{'MONO Tour Accessory Case 2.0, Black'} <=	{'MONO Pedalboard Medium, Black'}
0.36	{'MONO Classic Tick Accessory Case 2.0, Black'} <=	{'MONO Classic Electric Guitar Case, Black'}
0.29	{'MONO Vertigo Electric Guitar Case, Black'} <=	{'MONO Vertigo Semi-Hollow Guitar Case, Black'}

0.27	{'MONO Pedalboard Medium, Black'} <=	{'MONO Tour Accessory Case 2.0, Black'}
0.24	{'MONO Classic Electric Guitar Case, Black'} <=	{'MONO Classic Tick Accessory Case 2.0, Black'}
0.21	{'MONO Vertigo Electric Guitar Case, Black'} <=	{'MONO Vertigo Bass Guitar Case, Black'}
0.19	{'MONO Classic Tick Accessory Case 2.0, Black'} <=	{'MONO Vertigo Electric Guitar Case, Black'}
0.18	{'MONO Vertigo Semi-Hollow Guitar Case, Black'} <=	{'MONO Vertigo Electric Guitar Case, Black'}
0.15	{'MONO Vertigo Electric Guitar Case, Black'} <=	{'MONO Classic Tick Accessory Case 2.0, Black'}
0.15	{'MONO Vertigo Bass Guitar Case, Black'} <=	{'MONO Vertigo Electric Guitar Case, Black'}

Top 10 rules generated with both single item & multiple products over entire data set

Discussions

This recommendation analysis was made with a larger data set and should provide a better accuracy, however the confidence is actually lower. This is because of multiple sales items, the data was also aggregated from single items bought at different time.

Application of this: if the recommendation is to be made after sales of the first item has taken place, this recommendation will be suitable. For example, after the customer added item A to the sales cart, a recommendation can be made to the customer to suggest buying item B.

c. 3rd analysis: Frequent items over the entire dataset with the addition of virtual items analysis

In the third analysis, we added “virtual” items into the associative mining with intent to get recommendation with higher lift.

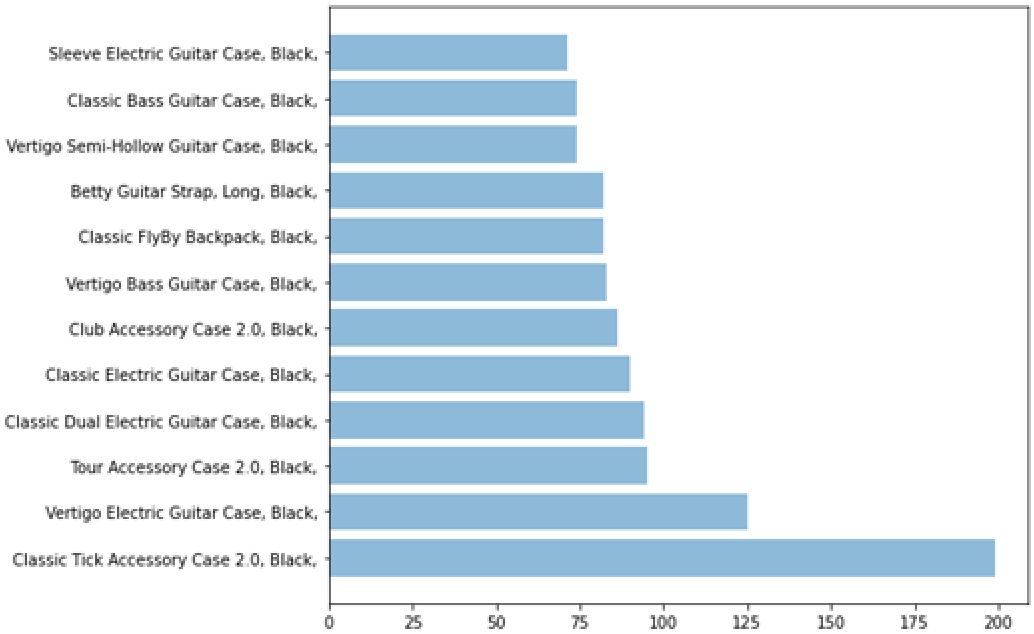


Fig 14 - Frequent items from all transactions data set with addition of virtual items

The top products from the analysis with the addition of virtual items using the larger data set is slightly different from the second analysis.

Confidence	**LHS**	**RHS**
0.63	{'Vertigo Electric Guitar Case, Black,'} <=	{'Vertigo Semi-Hollow Guitar Case, Black,'}
0.48	{'Vertigo Electric Guitar Case, Black,'} <=	{'Vertigo Bass Guitar Case, Black,'}
0.42	{'Vertigo Electric Guitar Case, Black,'} <=	{'Classic Electric Guitar Case, Black,'}
0.42	{'Vertigo Semi-Hollow Guitar Case, Black,'} <=	{'Vertigo Electric Guitar Case, Black,'}
0.36	{'Vertigo Electric Guitar Case, Black,'} <=	{'Classic Dual Electric Guitar Case, Black,'}
0.34	{'Classic Electric Guitar Case, Black,'} <=	{'Vertigo Electric Guitar Case, Black,'}

0.32	{'Classic Tick Accessory Case 2.0, Black,'} <=	{'Classic Electric Guitar Case, Black'}
0.32	{'Vertigo Bass Guitar Case, Black,'} <=	{'Vertigo Electric Guitar Case, Black,'}
0.29	{'Classic Dual Electric Guitar Case, Black,'} <=	{'Vertigo Electric Guitar Case, Black,'}
0.26	{'Classic Electric Guitar Case, Black'} <=	{'Classic Tick Accessory Case 2.0, Black,'}

Top 10 rules generated with “Discount” as virtual item

Discussions

Using the provision of discount as a virtual item for MBA did seem to improve the performance, in terms of getting recommendations with higher confidence.

Summary

In Summary, the objectives for performing the analysis for making a recommendation engine were met. We experimented with different segmentation of the data set and features to suit different avenues for the delivery of the recommendations. We only demonstrated three types for this practice module. For example, we can experiment with different “virtual” items to get further improvement over the current analysis.

In our current analysis, we were constrained by lack of customer profile type of data, otherwise we could have performed more comprehensive collaborative filtering for the analysis.

6. Database

Database is a MySQL 8.0 database hosted on AWS Cloud (https://aws.amazon.com/rds/mysql/). For installation, refer to *Appendix 4 - Installation and User Guide*

Company Database Model

ORDERS	PRODUCTS
id (pk) (int)	id (pk) (int)
created_at (timestamp)	title (text)
closed_at (timestamp)	sku (text)
processed_at (timestamp)	price_usd (double)
total_price (double)	category (text)
fulfillment_status (text)	tags (text)
line_items (text)	created_at (timestamp)
fulfillments (text)	
refunds (text)	
discount_applications (text)	
customer (text)	

Fig 15 - Solution Database Model

Solution Database Model

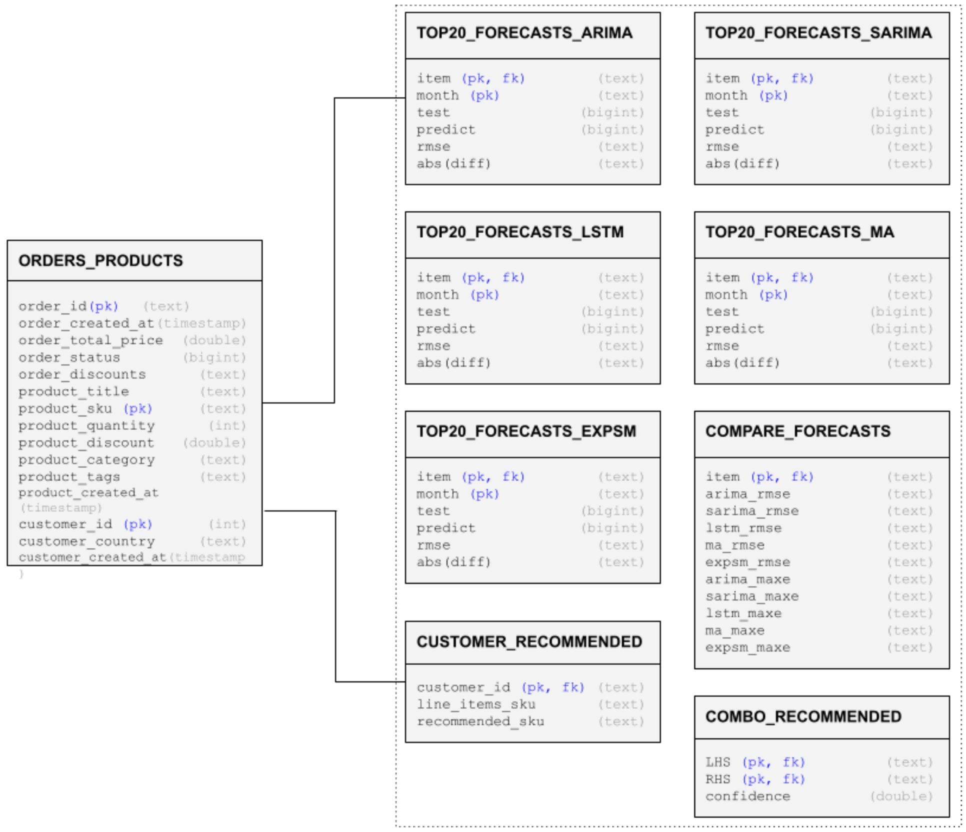


Fig 16 - Solution Database Model

Tables	Description
ORDERS	Orders data provided by company
PRODUCTS	Products data provided by company
ORDERS_PRODUCTS	Processed data, where each row represents only one (1) product in an order. Unique Key is <order_id>_<product_sku>.
TOP20FORECASTS_ARIMA	Forecast results for ARIMA, where: **item**: Product SKU **month**: Month of prediction **test**: Actual order count **predict**: Predicted order count **rmse**: Root Mean Square Error per SKU **abs(diff)**: Absolute Difference per SKU
TOP20FORECASTS_SARIMA	Forecast results for SARIMA. Refer above to field descriptions
TOP20FORECASTS_LSTM	Forecast results for LSTM. Refer above to field descriptions

TOP20FORECASTS_MA	Forecast results for Rolling MA. Refer above to field descriptions
TOP20FORECASTS_EXPSMOOTHING	Forecast results for Exponential Smoothing. This table reflects the results of the current forecasting model used by the company. Refer above to field descriptions
COMPARE_FORECASTS	Contains the MaxE and RMSE of each forecast
COMBO_RECOMMENDED	Contains top products and their respective recommended products with confidence
CUSTOMER_RECOMMENDED	Contains Customer IDs, SKUs which brought by customer and top 3 recommended SKUs with confidence

7. Web Application

The Web Application was bootstrapped using Flask Bootstrap 5 by App Volt[11], which is based on Flask, a micro web framework written in Python. The framework was chosen due to the following characteristics:

a. **Lightweight:** Reduces bloat from unused features
b. **Modular**: Allows us to add what we need (e.g. SQLAlchemy for Database Connection)
c. **Open source**: Flask has a vibrant community with lots of tutorials and sample code online, which eases development
d. **Language**: As python was used for the back-end, the front-end code could be easily understood by all team members

[11] https://appseed.us/admin-dashboards/flask-dashboard-volt

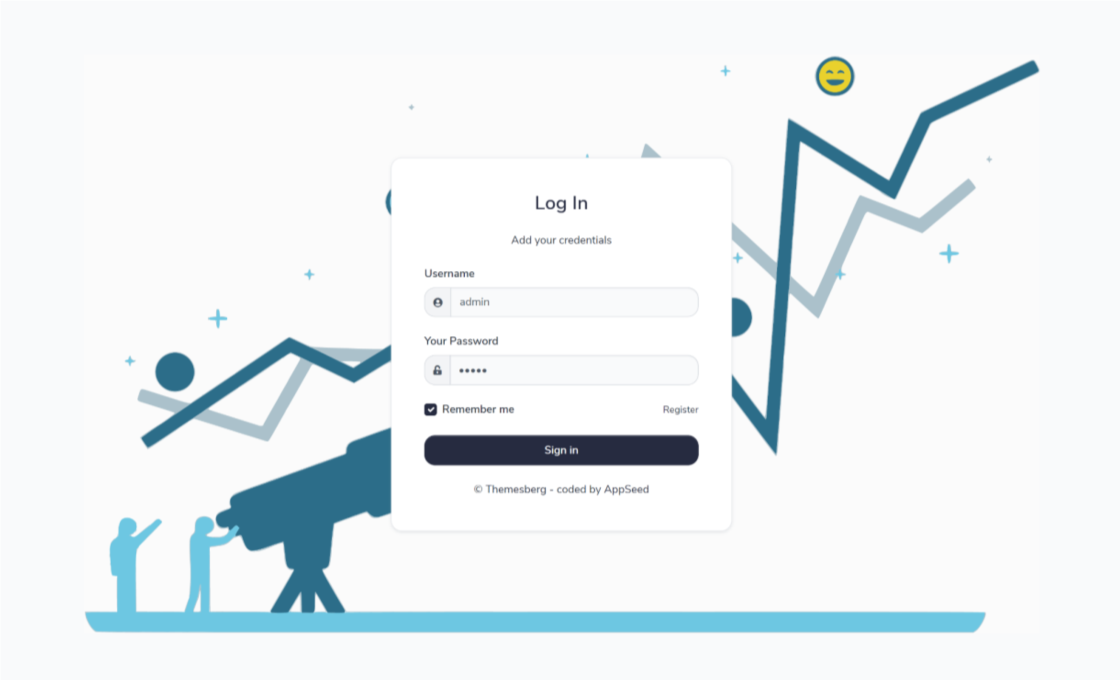


Fig 17 - Login Page for the web application

For installation, refer to *Appendix 4 - Installation and User Guide*

8. Dockerization and deployment

Docker is a tool designed to make it easier to create, deploy, and run applications by using containers. Containers allow a developer to package up an application with all of the parts it needs, such as libraries and other dependencies, and ship it all out as one package.” [12]

Docker Compose is a tool for defining and running multi-container Docker applications. With Docker Compose, you use a YAML file to configure your application’s services. Then, with a single command, you create and start all the services from your configuration.[13]

Deployment dividing components

As the architecture design in section VI, we divide the system into 3 different components:

- Web application: contain web api application and web frontend application.
- Backend component: contain source code of data preprocessing block, forecasting models block and cron job scheduler block.
- Database component: contain all database of this system (raw database and data database)

[12] https://blog.usejournal.com/what-is-docker-in-simple-english-a24e8136b90b
[13] https://docs.docker.com/compose/

Deployment solution

As the dividing to 3 different components, we will picking the most suitable deployment way for each of components:

With Database, it is the heart of the system. After considering all factor related to database deployment, we decide to deploy it to cloud solution which bring us some advantage:

- Other components can connect databases from anywhere.
- The database will be consistent when demonstrating and reporting.
- Increase working performance, easier for teamwork.

The Database deployment detail is described in Appendix 4

With web application component and backend component, it is not required to deploy it online. Moreover, this project is a sponsor project so it is not encouraged to run it on air. After discussing, we picked up docker technical for deployment, which contain advantages as below:

- Easier and simpler for deploying, there are only 2 to 3 steps to deploy a component. it guarantees that the deployment will be done as designed.
- Be consistent in different environments, which let us feel confident when deploying it in any OS or any machines.
- Reduce issue number when deploying. The project required a complicated environment which was hard to control in a normal deployment.

The Webapp and Backend component deployment detail are described in Appendix 4

VIII. PROJECT PERFORMANCE AND VALIDATION

1. Performance Evaluation for forecasting

For the sales forecasting, 5 models were tested:
- Rolling Moving Average
- Exponential Smoothing
- AutoRegressiveMoving Average (ARIMA)
- Seasonal AutoRegressiveMovingAverage (SARIMA)
- CNN-LSTM

There are several metrics to express model prediction error:
1. RMSE (Root Mean Squared Error)

$$\mathrm{RMSE} = 1/n\sqrt{\textstyle\sum_{j=1}^{n} (y_i - y_{pred})^2}$$

2. MAE (Mean Absolute Error)

$$\mathrm{MAE} = 1/n\sum_{j=1}^{n} |y_i - y_{pred}|$$

3. MaxE (Maximum Absolute Error)

$$\mathrm{MaxE} = \left|\sum_{j=1}^{n} (y_i - y_{\ pred})\right|$$

The company has chosen to stick to 2 metrics (RMSE and MaxE).

MaxE was selected for its simplicity. On the business side, it can easily be explained that the model can have a maximum error equivalent to MaxE.

Given that RMSE and MAE are similar methods of calculating prediction error, it was decided to purely use RMSE, as it gives a higher weight on large errors, which are highly undesirable in our use case.[14]

Computing the errors:

	month	item	test	predict	ARIMA_rmse	ARIMA_diff
0	m+1	EFX-FLY-BLK	2	0	3.83	10
1	m+2	EFX-FLY-BLK	2	0		
2	m+3	EFX-FLY-BLK	6	0		

[14] https://medium.com/human-in-a-machine-world/mae-and-rmse-which-metric-is-better-e60ac3bde13d

For an item with the 3 months forecast, the error is between the predicted value to the test value,
Using Root Mean Square Error = SQRT((2-0)^2 + (2-0)^2 + (6-0)^2) = 3.83

Using Max Absolute Error = ABS((6-0) +(2-0) + (2-0)) = 10

Using the 2 methods of error measure, we compute these errors over the 20 top items to see which models gives the best performance.

a. Using RMSE as a measurement

In using RMSE, the root mean squared errors over the 3 months forecast were computed. The comparison is then done with all the 5 models used to see which model gives the best performance. The results of the RMSE is as below:

			Benchmark	
ARIMA_rmse	SARIMA_rmse	MA_rmse	ExpSm_rmse	LSTM_rmse
3.83	3.83	1.91	3.7	1.29
1.73	4.24	1.73	1.41	0.58
5.48	3.11	2.94	6.19	2.38
2.08	2.65	2.45	1.83	0.0
0.82	3.0	1.41	1.73	1.29
2.16	2.16	1.29	7.39	0.58
2.08	2.08	1.29	2.08	0.58
1.83	2.16	1.91	11.09	0.82
3.11	3.11	3.11	2.94	2.08
2.94	2.94	1.41	1.73	1.41
1.29	1.15	0.58	1.41	0.0
2.52	3.16	1.41	7.62	1.29
4.51	4.51	3.56	4.51	0.82
1.91	3.16	1.91	2.16	0.58
3.11	5.07	1.73	0.58	0.82
2.38	5.42	1.29	0.58	0.82
2.08	2.94	0.82	2.16	0.58
1.91	7.77	1.41	1.83	2.38
2.38	0.58	1.41	2.38	0.82
1.73	2.45	2.16	2.08	2.16

From the results, we can observed the following on the best results for the 20 items forecasted:

Models	Counts
LSTM	14
ARIMA	2
SARIMA	1
Rolling MA	2
Exponential Smoothing	2

From the results above we see that LSTM perform best in 14[15] of the 20 items measured. LSTM also performed much better than the Exponential Smoothing, which is a model employed by the company for one of their eCommerce sites. From these results we can safely conclude that LSTM gives the best performance of all the 4 models tested.

b. Using MaxE over 3 months as a measurement

In using difference over 3 months, the comparison then thus on the difference in the sales forecast over the 3 months compared to the actual. Using difference may be easier to see since it is the total difference.

[15] It noted that the total count is more than 20. If two models perform best for an item, they are each given one count.

			Benchmark	
ARIMA_maxe	SARIMA_maxe	MA_maxe	ExpSm_maxe	LSTM_maxe
10	10	1	7	1
1	12	5	2	1
14	3	6	17	3
1	5	6	2	0
2	9	2	1	3
6	6	3	18	1
5	5	1	5	1
4	0	5	29	0
7	7	7	4	1
8	8	0	3	0
3	2	1	4	0
1	6	2	22	1
13	13	10	13	0
3	6	5	2	1
9	15	3	1	0
7	16	3	1	0
5	8	2	6	1
5	23	4	4	7
7	1	4	7	2
3	6	4	5	0

From the results, we can observed the following on the best results for the 20 items forecasted:

Models	Counts
LSTM	17
ARIMA	2
SARIMA	3
Rolling MA	4
Exponential Smoothing	2

From these results using difference, we can also see that LSTM gives the best performance of all the 5 models tested. Similarly, LSTM performed much better than the benchmark model. We also noted that, except for one item, the Max Absolute Error for the LSTM for the 3 months forecast is 3.

Measurement of errors over 3 months may be a better method since if you over-forecast for one month and under-forecast for the next, you eventually even out

2. Performance Evaluation for Recommendation Analysis

Using LIFT as a performance criterion for recommendation analysis

For the recommendation analysis, we made use of "lift" to compare the quality of the analysis.[16] Lift is a metric that compares the improvement in prediction with reference against random guessing.

Results for the first analysis: Only transactions with multiple items ordered in a sales transaction are used in this analysis.

```
#holdbacks= 57 recitems= 107 hits= 31 (28.97%) randrecitems=
107 randhits= 1 (0.93%) rulelift=31.00
```

The lift is up to 31 over random. It shows that compared with random choice, the recommendation made after the associative mining provided a much higher hit rate.

Results for the second analysis: all transactions are used. Dataset is inclusive of both single and multiple items ordered.

```
#holdbacks= 106 recitems= 166 hits= 24 (14.46%) randrecitems=
166 randhits= 1 (0.60%) rulelift=24.00
```

The lift is up to 24 over random. This is lower compared with the previous analysis which yielded a lift of 31. This was however expected since the second analysis was made over a larger data set. The confidence from this recommendation is also lower than the previous.

[16] https://en.wikipedia.org/wiki/Association_rule_learning#Lif

IX. PROJECT CONCLUSIONS

Findings

1. The current dataset provided for the particular site is somewhat small in the volume of items sold. Many of these items (maybe because they are specialised music accessories) only sell one or 2 items a month. It will be difficult to forecast their sales with reasonable accuracy in this respect. Because of the sales volume, only the top 20 items (with sales varying from 300 to 800 in 3 years) were considered.
2. Notwithstanding the small dataset, the LSTM model has been shown to give the best performance of the 5 models tested. The maximum error over 3 months of 3 is also within acceptable limits for the administration. This model also performed better than the benchmark model (Exponential Smoothing) which is currently being used by the company. This model can be further fine-tuned to give even better results.
3. The current analysis for recommendation provides a much higher probability wrt random guessing, however, this is only available to the top 10, top 15 most popular products. For the less frequently sold products, there is no recommendation because there is too little data to support this.

Recommendations

1. As the present project is a PoC for the company and only covers one eCommerce site, this application which if approved can be extended to the other site. In fact, with the other retailers, product recommendations can be even more effective as there can be cross-selling and up-selling. [17]
2. Extend forecasting to product types and product brands. [17]
3. Test model with a larger set of data for better results. [17]
4. Current models for forecasting are univariate, we can use multivariate data and also exogenous data to help in forecasting.
5. There are many more deep learning models that can be explored to see if they can give better performance. We should continue to work on this area.
6. The current application only forecasts the sales and thus re-ordering. Taking inventory into consideration will further help the administration in planning for new orders.

[17] Appendix 8 - Questionnaire on Feedback for Forecasting PoC

ANNEXURES

Appendix 1 – Project Proposal

GRADUATE CERTIFICATE: Intelligent Reasoning Systems (IRS)
PRACTICE MODULE: Project Proposal

Date of proposal: March 22, 2021
Project Title: ISS Project - Sales forecasting system
Sponsor/Client: *(Name, Address, Telephone No. and Contact Name)* COMPANY NAME: BandLab Technologies COMPANY ADDRESS: 12 Jln Kilang Barat, Singapore 159354 CONTACT NAME: Tan Nicole Ongoco / Web Developer Lead, E-commerce & Media CONTACT TELEPHONE: +65 8117 1063 CONTACT EMAIL: nicole.tan@bandlab.com
Background/Aims/Objectives: The company currently handles over 150+ brands and 16,000+ products. Procurement of stocks is done manually, where additional stock is determined by: 1. Current stock available 2. Forecasted orders per month Calculations and order forecasting are currently done on Excel, and due to the number of products, this can get very time-consuming. To create this PoC, the company will provide us data for 1 brand, with over 100+ products. The output will be a web application which will: 1. Forecast sales per product SKU using different models 2. Provide additional reports for Data Discovery

Requirements Overview:

Research ability:

- Research suitable methods for order forecasting

Programming ability

- Python (Used for modeling and web application)
- MySQL (Queries to Read and Insert data)
- JavaScript (Used for web application)
- HTML (Used for web application)

System integration ability

- Integrate with Cloud Database, which is the current setup of the company

Resource Requirements (please list Hardware, Software and any other resources)

Hardware proposed for consideration:

- Windows / Linux / MacOS instance

Software proposed for consideration:

- Cloud-hosted Database (Amazon RDS)
- Application container (Docker)

Number of Learner Interns required: (Please specify their tasks if possible)
A team of four (4) project members

Methods and Standards:

Procedures	**Objective**	**Key Activities**
Requirement Gathering and Analysis	The team should meet with ISS to scope the details of the project and ensure the achievement of business objectives.	1. Gather & Analyze requirements 2. Define internal and external design 3. Prioritize & consolidate requirements 4. Establish Functional Baseline
Technical Construction	To develop the source code in accordance to the design.	1. Setup Development Environment 2. Understand the System Context, Design 3. Perform Coding 4. Conduct Testing
Acceptance Testing	To obtain ISS user acceptance that the system meets the requirements	1. Obtain Customer Acceptance Sign-off
Delivery	To deploy the system locally	1. Software must be packed by following ISS's standard 2. Deployment guideline must be provide

Appendix 2 – Team Formation & Registration

Team Name: Team Telescope
Project Title (repeated): ISS Project - Order forecasting system
System Name (if decided): N/A
Team Member 1 Name: Yap Pow Look
Team Member 1 Matriculation Number: A0163450M
Team Member 1 Contact (Mobile/Email): +65 96390998 / e0147014@u.nus.edu
Team Member 2 Name: Chow Kok Peng
Team Member 2 Matriculation Number: A0195403H
Team Member 2 Contact (Mobile/Email): +65 93860367 / e0385034@u.nus.edu
Team Member 3 Name: Nguyen Minh Tien
Team Member 3 Matriculation Number: A0229981N
Team Member 3 Contact (Mobile/Email): +65 90456474 / e0687389@u.nus.edu
Team Member 4 Name: Tan Nicole Ongoco
Team Member 4 Matriculation Number: A0229980R
Team Member 4 Contact (Mobile/Email): +65 81171063 / e0687388@u.nus.edu

Appendix 3 – Mapped System Functionalities

Requirement	Satisfaction of requirement
1. Knowledge Discovery using suitable data mining techniques	We have conducted interviews with the Subject Matter Expert (SME) on how current forecasting is done, how it can be improved upon, and the validation of results. We have also conducted exploratory data analysis to **<u>extract the patterns and insights from historical data.</u>** The data were extracted, transformed, and loaded back to the database. This extracted data will be used to predict the future behaviour, in terms of supply and demand. We also used associative mining techniques to extract information between items bought (market basket analysis). This extracted knowledge allows the company to do combo packaging of products for marketing promotion. Recommendations can also be sent to customer post sales to entice repeat sales.
2. Business rule	From the analytics (time series, deep learning, Market basket analysis for recommendation), we will formulate **<u>simple business rules</u>** for assistance to decision making. One example is the optimum time for restocking of goods for the manager's decision making. We have automated the demand forecasting process from extracting past data from SQL database, data mining and eventually to prediction & visualization.
3. Business resource optimization techniques: Search	Business resource optimization is performed in terms of **<u>searching</u>** for the best quantity of stock to carry, with consideration for the predicted customer demand in the near future. It is striking a balance between minimizing carry stock depreciation while meeting most of the sales demand with minimum occurrence of stockout. Stockout is undesirable because it will incur ill will among potential customers.

4. Cognitive frameworks	The intuitive and interactive user interface of the application created provides a pseudo cognitive impression for the user. The user is able to interactively select the desired product for visualization. The system, after performing machine learning analysis, responds by providing charts displaying the historical performance and predicted future demand for the selected stock item. We can further automate the GUI to provide a more “Cognitive” user interface in the future.

Appendix 4 – Installation and User Guide

Database
Uses the cloud database (AWS RDS)
Note: Please ensure your network doesn't block connections to amazonaws.com

Credentials:

```
HOST: idm5peipdsus5o.crcvo0yw3sz7.ap-southeast-1.rds.amazonaws.com
PORT: 3306
DATABASE: iss_project

# Read Only Credentials (For Web App)
USERNAME: iss_readonly
PASSWORD: *************

# Full Permissions
USERNAME: iss
PASSWORD: *************
```

Docker environment
Using docker to get more stability and easier for debugging in deployment. In this project implementation, Our docker environment will be setted up in the AI ISS-VM OS [18], which is shared by Sam. For other machines and other OS, please follow the installation guide in this link https://docs.docker.com/engine/install/.

Docker compose[19] also needs to be installed which helps to provide a more simple and stable way to run and deploy our product. Please follow this installation guide in this link https://docs.docker.com/compose/install/ to install docker compose.

In Windows and MacOS, you also can install Docker Desktop[20], which provides a completed docker environment for the deployment. Please follow this installation guide in this link https://docs.docker.com/docker-for-windows/install/ to install docker desktop.

Please take note that you should upgrade Docker Desktop in Windows to the latest version. You may meet some issues when running an old Docker Desktop version. You should also switch to using Linux containers when using Docker on Windows. For linux, using docker-compose is enough.

Back-End & Front-End

```
# Using git client to clone the source code into the local machine

# Navigate to the project source code folder
$ (Unix/Mac) cd SystemFiles/
```

[18] https://github.com/telescopeuser/iss-vm
[19] https://docs.docker.com/compose/
[20] https://docs.docker.com/docker-for-windows/

```
$ (Windows) cd "SystemFiles\"
$ (Powershell) cd "SystemFiles\"

# Install Docker environment* (Read above section for install
docker/docker-compose environment)[21]

# Run service
$ docker-compose up &

# Check docker instance & services
$ docker ps
root@iss-vm:/media/sf_NUS-IS-IS/IRS-MR Machine Reasoning/Projects/ISS_MR_2021_Anxiety_Diagnosis/SystemFiles# docker ps
CONTAINER ID   IMAGE                    COMMAND                  CREATED         STATUS         PORTS                    NAMES
f01f684d7ffd   systemfiles_backend-app  "cron -f"                4 seconds ago   Up 2 seconds                            systemfiles_backend-app_1
d938ce658b44   systemfiles_appseed-app  "gunicorn --config g…"   4 seconds ago   Up 1 second    0.0.0.0:5005->5005/tcp   systemfiles_appseed-app_1

# Access the dashboard in browser: http://127.0.0.1:5005/
```

Front-End

In case deploying a frontend app only, we can navigate to the webapp folder and run docker compose from that one. In case we want to run webapp in a different port (the default port is 5005), please change the config in `SystemFiles/webapp/docker-compose.yml` file. In that file, we can change the port to port 80 or other port which we want.

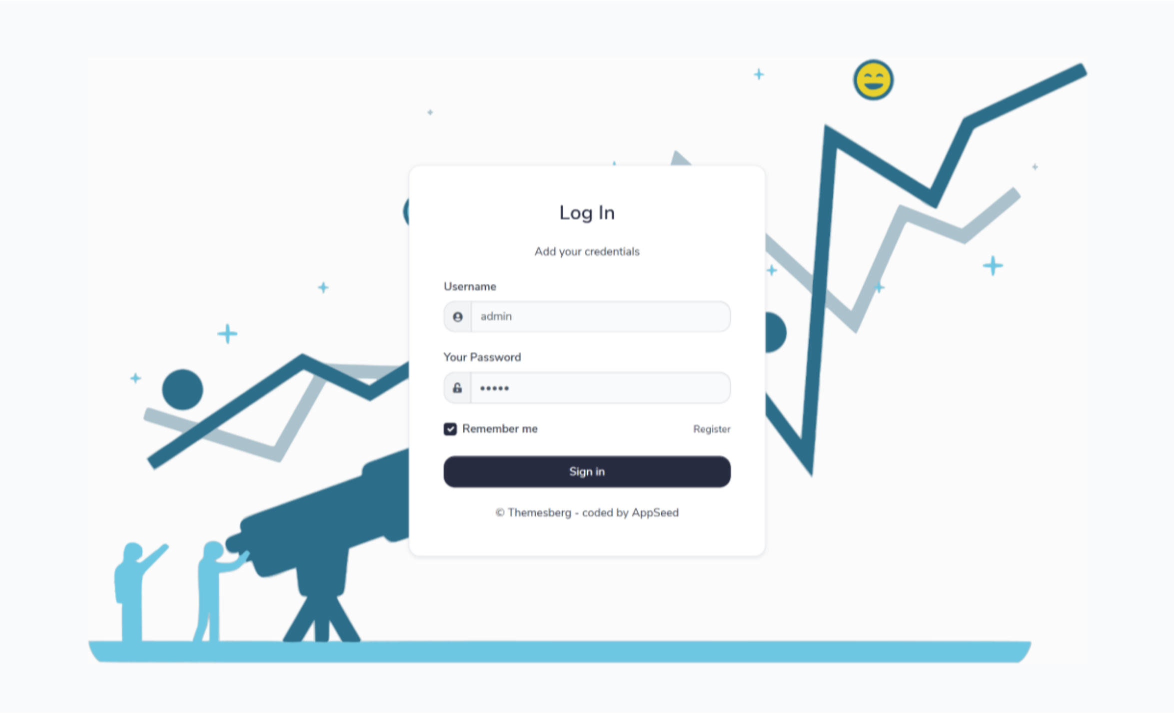
```
# Using git client to clone the source code into the local machine

# Navigate to the webapp folder
$ (Unix/Mac) cd SystemFiles/webapp
$ (Windows) cd "SystemFiles\webapp"
$ (Powershell) cd "SystemFiles\webapp"

# Install Docker environment* (Read above section for install
docker/docker-compose environment)[22]

# Run service
$ docker-compose up &

# checking docker instance & services
$ docker ps
root@iss-vm:/media/sf_NUS-IS-IS/IRS-MR Machine Reasoning/Projects/ISS_MR_2021_Anxiety_Diagnosis/SystemFiles# docker ps
CONTAINER ID   IMAGE                COMMAND                  CREATED          STATUS         PORTS                    NAMES
160c13d80eaf   webapp_appseed-app   "gunicorn --config g…"   11 seconds ago   Up 2 seconds   0.0.0.0:5005->5005/tcp   webapp_appseed-app_1
root@iss-vm:/media/sf_NUS-IS-IS/IRS-MR Machine Reasoning/Projects/ISS_MR_2021_Anxiety_Diagnosis/SystemFiles#

# Access the dashboard in browser: http://127.0.0.1:5005/
```

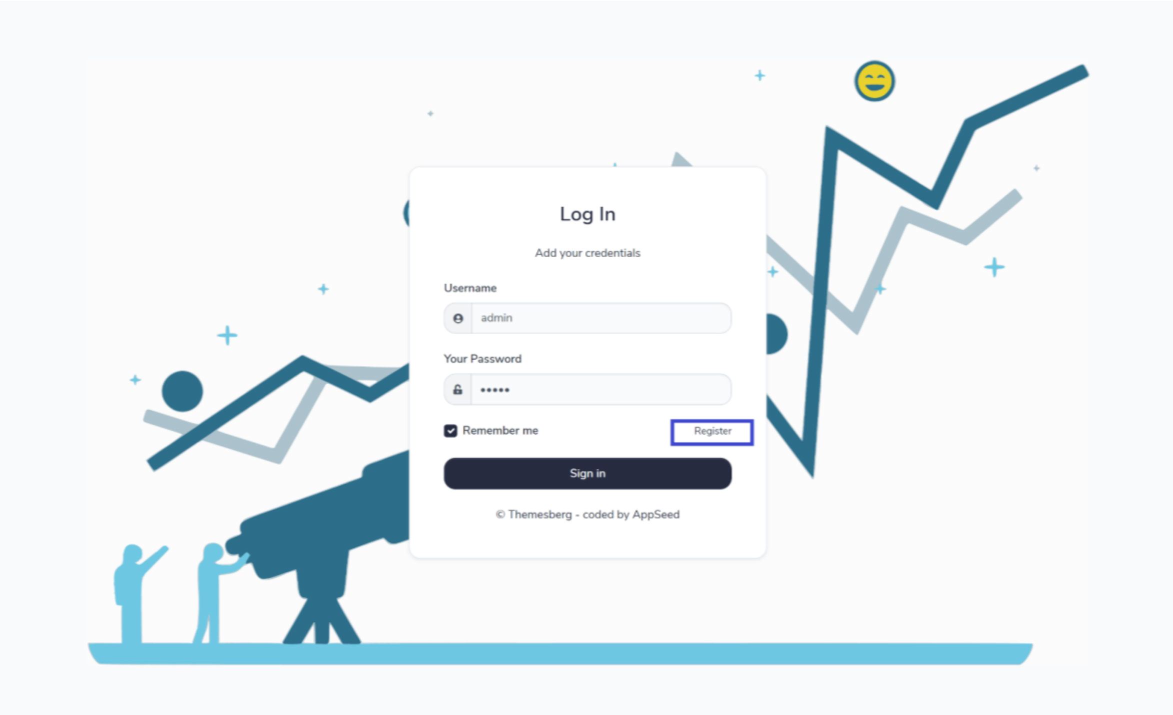
[21] https://docs.docker.com/compose/
[22] https://docs.docker.com/compose/

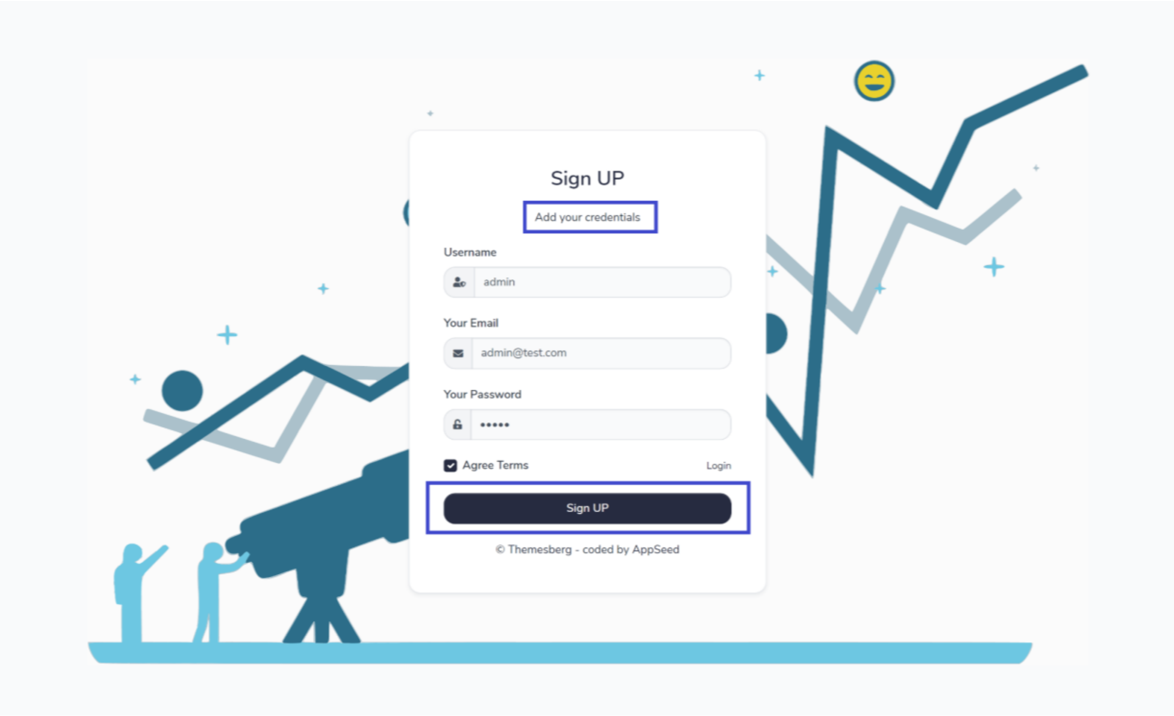
User Guide:

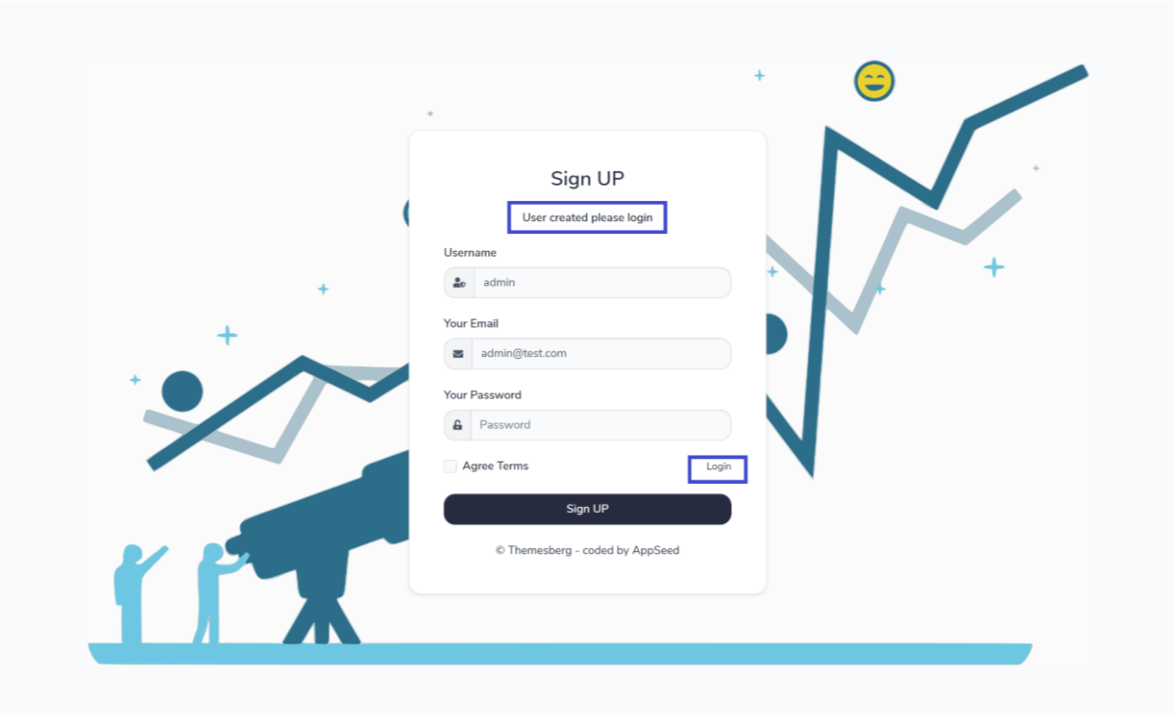
Log In

- For data protection, user is required to Log In before the app can be accessed.
- If Cloud DB is used, you may use the following test credentials:
`Username: admin`
`Password: admin`
- If Local DB is used, registration is required *(See "Register" in the following section)*



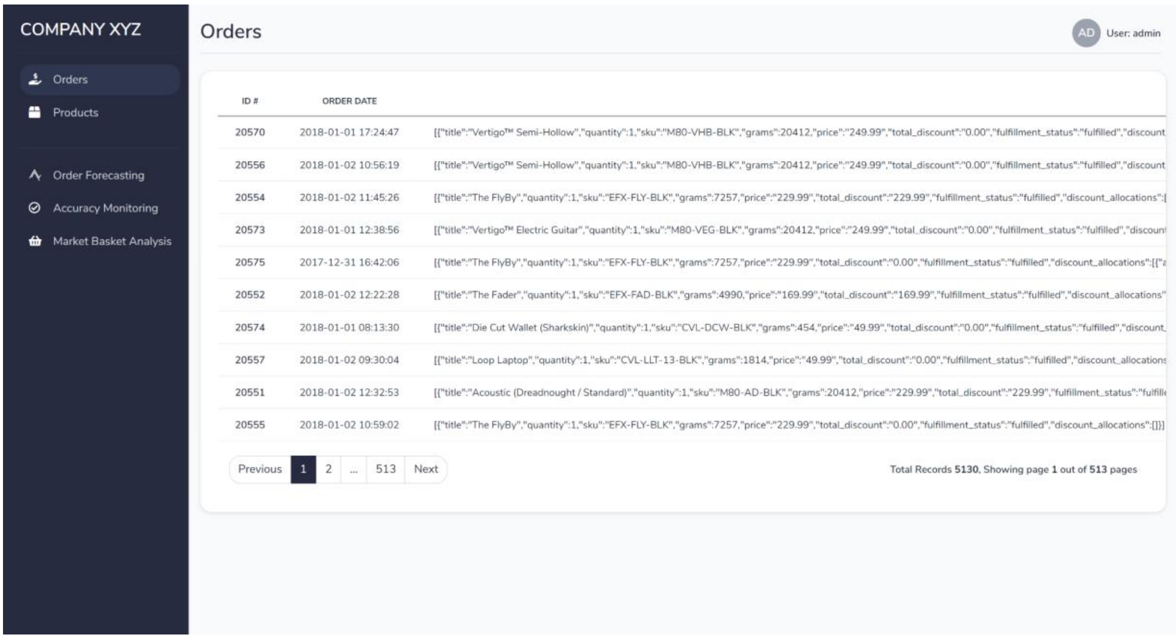
Register
- If Local DB is used, registration is required - Steps: 1. On the Log In Page, click Register 2. You should see the label "Add your credentials". 3. Input username, email, and password. Note that username and email must be unique 4. You should see the label change to "User created please log in" 5. Log in as normal (See previous step)
   *(Continued on next page)*

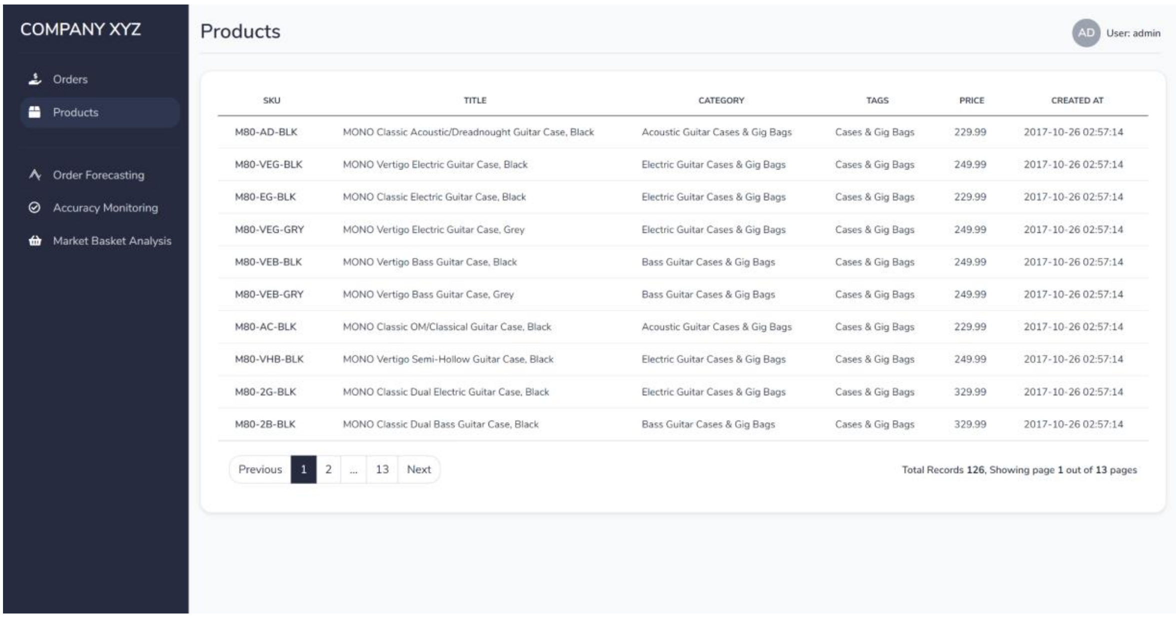

Sign UP
Add your credentials
Username
admin
Your Email
admin@test.com
Your Password
Agree Terms
Login
Sign UP
© Themesberg - coded by AppSeed


Sign UP
User created please login
Username
admin
Your Email
admin@test.com
Your Password
Password
Agree Terms
Login
Sign UP
© Themesberg - coded by AppSeed

Viewing Company Data (Raw)

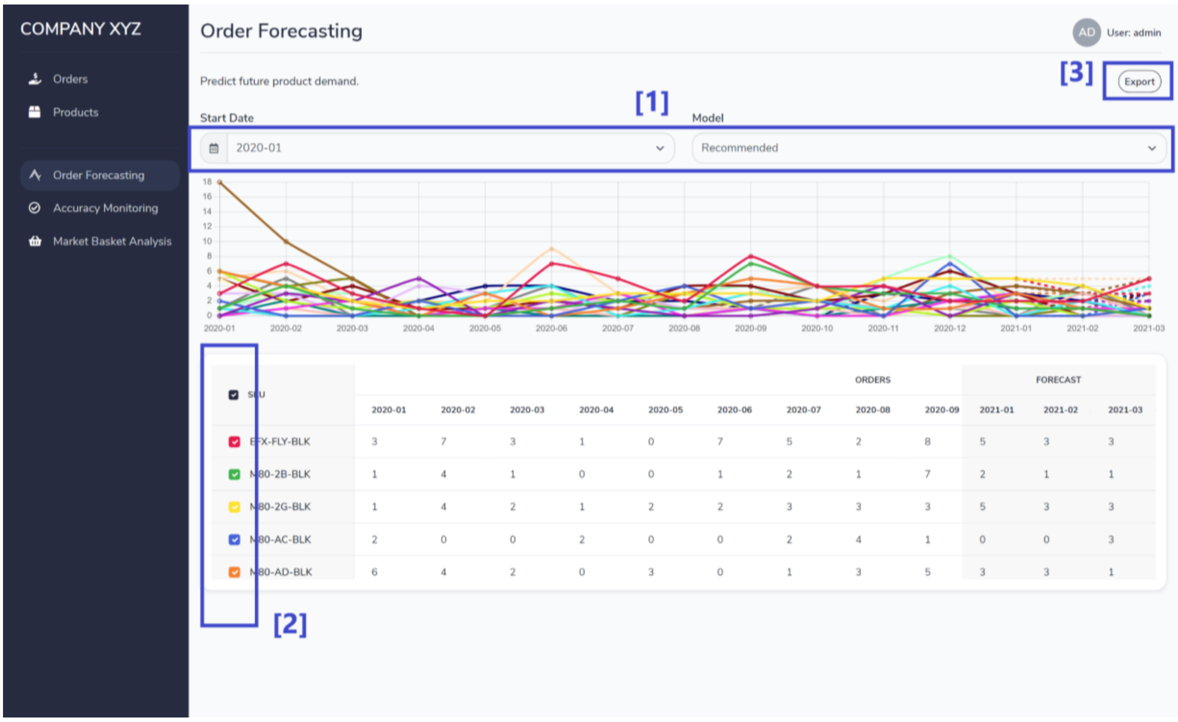
- Raw data can be accessed through the “Orders” and “Products” tab on the left-hand navigation
- Orders and Products display the raw, unprocessed data of the company
- This can be used for quick visualization of the raw data, such as the number of rows, number of columns, and the latest updated date of orders and products included in the forecast





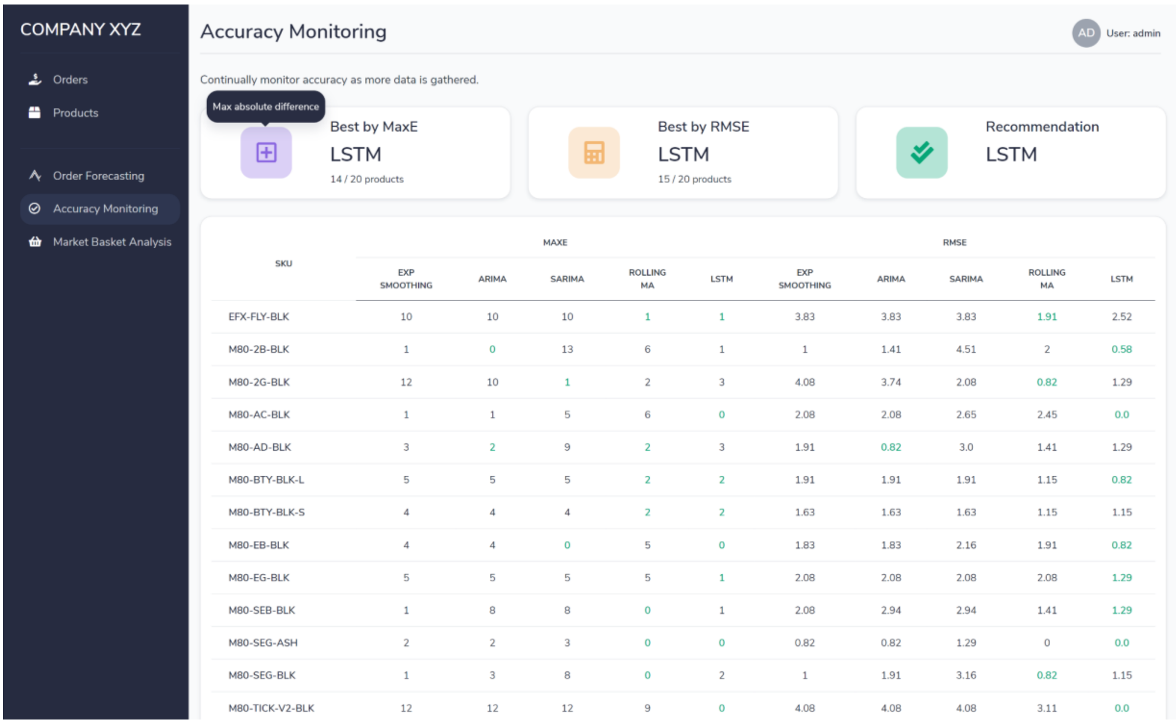
Viewing / Visualization of Orders and Forecasts

- Viewing and visualization of Orders and Forecasts may be accessed from the *Order Forecasting* tab on the navigation
- Start Date can be adjusted to visualize order trends. Default is *1 year*. [1]
- Model can be adjusted to display model performance. Default is *Recommended*. [1]
- SKUs can be filtered by the table checkboxes [2]
- Forecasts can be exported via the Export button on the top right [3]



Monitoring Accuracy

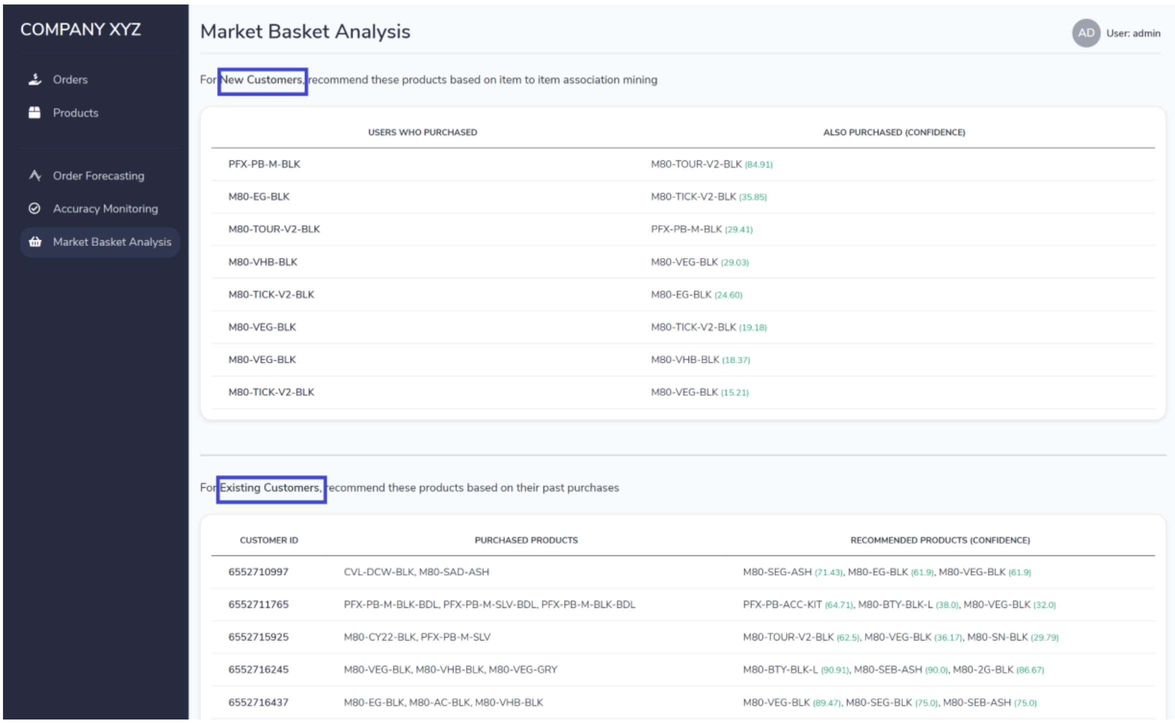
- Accuracy Monitoring may be accessed from the *Accuracy Monitoring* tab on the navigation
- The two (2) criteria for accuracy monitoring is displayed, each showing which model performed best. These are Maximum Absolute Difference Error (MaxE) and Root Mean Squared (RMSE)
- The recommended model is displayed last, which is retrieved by getting the lowest MaxE and RMSE
- The table below displays the MaxE and RMSE per Product SKU.



SKU	MAXE					RMSE				
	EXP SMOOTHING	ARIMA	SARIMA	ROLLING MA	LSTM	EXP SMOOTHING	ARIMA	SARIMA	ROLLING MA	LSTM
EFX-FLY-BLK	10	10	10	1	1	3.83	3.83	3.83	1.91	2.52
M80-2B-BLK	1	0	13	6	1	1	1.41	4.51	2	0.58
M80-2G-BLK	12	10	1	2	3	4.08	3.74	2.08	0.82	1.29
M80-AC-BLK	1	1	5	6	0	2.08	2.08	2.65	2.45	0.0
M80-AD-BLK	3	2	9	2	3	1.91	0.82	3.0	1.41	1.29
M80-BTY-BLK-L	5	5	5	2	2	1.91	1.91	1.91	1.15	0.82
M80-BTY-BLK-S	4	4	4	2	2	1.63	1.63	1.63	1.15	1.15
M80-EB-BLK	4	4	0	5	0	1.83	1.83	2.16	1.91	0.82
M80-EG-BLK	5	5	5	5	1	2.08	2.08	2.08	2.08	1.29
M80-SEB-BLK	1	8	8	0	1	2.08	2.94	2.94	1.41	1.29
M80-SEG-ASH	2	2	3	0	0	0.82	0.82	1.29	0	0.0
M80-SEG-BLK	1	3	8	0	2	1	1.91	3.16	0.82	1.15
M80-TICK-V2-BLK	12	12	12	9	0	4.08	4.08	4.08	3.11	0.0

Market Basket Analysis

- Market Basket Analysis may be accessed from the *Market Basket Analysis* tab on the navigation
- Recommendations are provided for both new users and existing users
- New users are given recommendations on product view of items in the LHS
- Existing users are given recommendations based on the past purchases



Appendix 5 – Individual member's report

Report by : Yap Pow Look

a. Personal contribution

I initially proposed the idea of chat-bot that help to diagnose anxiety disorder behaviour. This was because of the COVID situation last year which has created much stress on people with lockdowns and loss of jobs. I also see this project having a long run-way as it can be extended to deep learning using voice pattern recognition in emotion detection. However a lack of available datasets is a cause of concern for the project members. While rules can be developed from the current anxiety disorder diagnosis handbooks, it will take too much time and also the availability of a domain expert which we can refer to. Subsequently, we decide on a project from another member (Nicole) where the company she worked for is interested to find a better forecasting model for the sales orders that they are currently using. Having to solve a real-world problem is a good project to do.

My part in this project is the development of forecasting models. For this I explored the variety of models that are currently being used, from the simple moving average to the statistical models using ARIMA and SARIMA to the deep learning models using CNN and LSTM. I concentrated mainly on univariate models mainly due to time constraints. In all I considered more than 10 different models before short-listing it to about 5 for comparison. I would have considered using HMM as one of the models if I have come across it earlier in the course.

b. What I have learned

Besides learning what I have done for my part, it is important for the project like this to have complementary skill sets. We have members which contributed to building the system design to developing the web-front to one with expertise in deployment and continuous integration. We also have members with understanding of business rules and also understand the business environment the company operates in. It's working as a team that makes it successful.

c. How can I apply the knowledge and skills learned

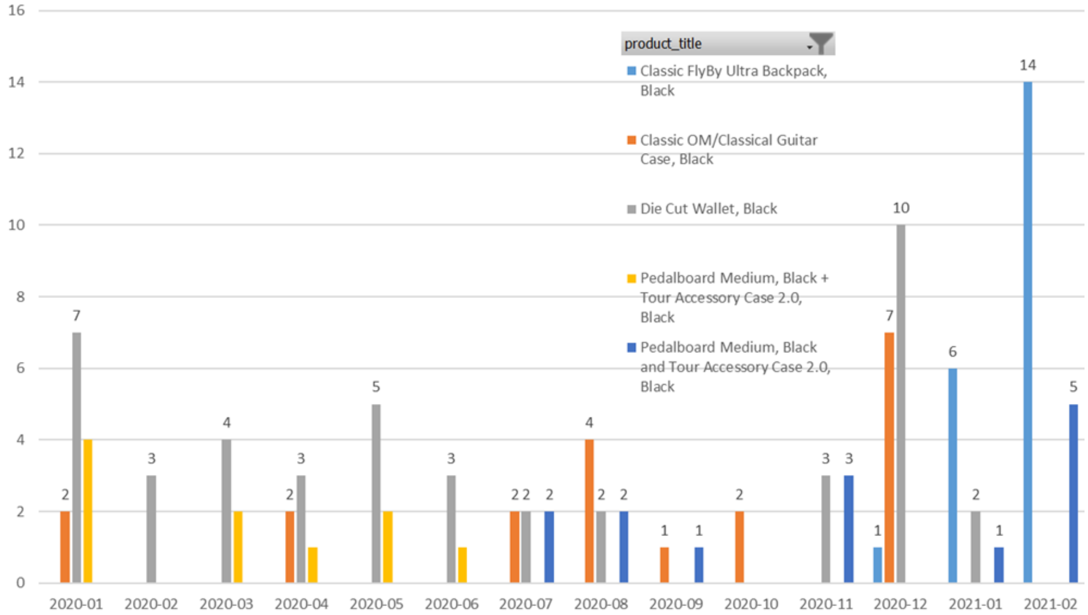
In doing the task of forecasting, I have barely covered 10% of the scope. There is much more that can be done. Times Series have many parameters to consider like exogenous data or having multivariate outputs. Deep Learning models itself can have inexhaustible configurations. Having accurate forecasts whether for sales, inventory, stocks or weather is central to the business objectives. However I find that sequence modelling like HMM will be an invaluable skill to add to the forecasting arena.

Report by : Chow Kok Peng

a. Personal contribution

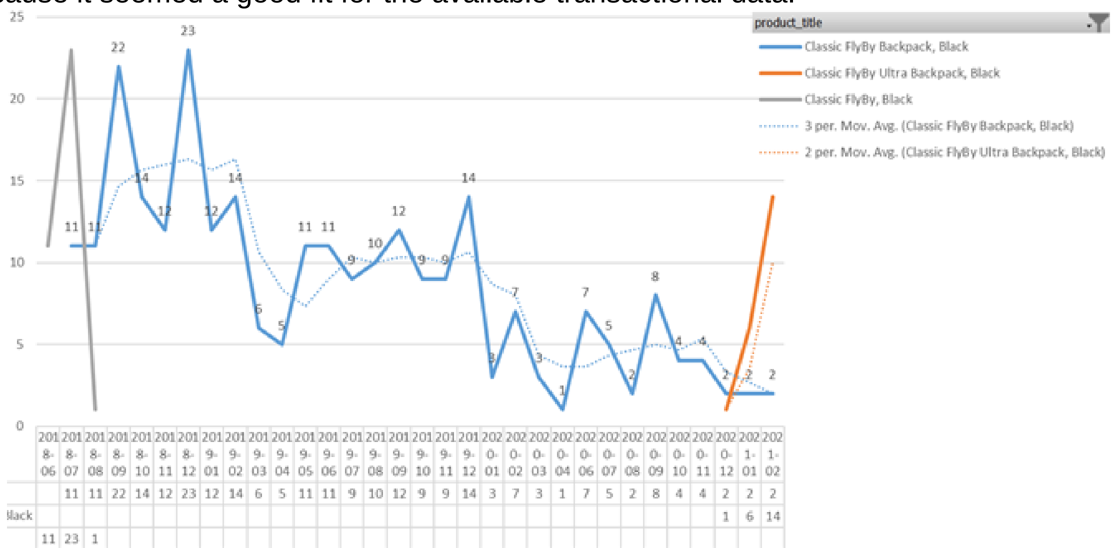
<u>Diary of Events</u>

The initial intention was to plot out the sales of items against time to support demand forecasting. It can be observed from the chart below, sales of some items disappeared, while new items for sales appeared. Generally, the sales of the item against time showed much discontinuity and too few items. Therefore, move on to attempt to do a product life cycle analysis instead.



<u>Typical product sales against time plot</u>

From the chart below, I discovered that the products in the data set do not really exhibit any obvious pattern that support life cycle analysis either. Eventually moved on to do MBA because it seemed a good fit for the available transactional data.



<u>Best chart for lifecycle analysis</u>

Scope
1. Perform data exploration for data understanding.
2. Generates charts/visualizations to help understanding.
3. Performed data mining to support forecasting demand, though not successful.
4. Performed data mining (Market Basket Analysis, MBA) to extract the association relationships for building a recommendation engine.

Approach for Market Basket Analysis:
Performed Market Basket Analysis on three types of data sets:
1. Subset the data set to only using multiple items purchase,
2. Expand the data set to include single item purchase,
3. Addition of virtual item "discount" into the data set.

b. What I have learned

Interpretation of Results
Results of the first is for combo bundling sales, while the latter is for sales recommendations at the checkout stage or post sales recommendation.

Reflections: Thanks to teammates with expertise in SQL, docker, App development and deployment techniques, otherwise would not be able to do this end-to-end project from data extraction to visualization to docker deployment.

c. How can I apply the knowledge and skills learned

Takeaways: Applied the MBA python codes on actual data and it yielded good results. This will provide more confidence in future projects involving transactional data.

Report by : Minh Tien

a. Personal contribution

I was lucky when teaming up with PowLook, Nicole, and KokPeng. Together, we are a good team, we have spent time brainstorming, finding solutions, and working together. Starting with a chatbot system idea that can apply in medical for helping people, an idea for finding security events from server activities, we finished with a sponsored project in e-commerce by Nicole's company. With my teammate, I spent my time understanding the problem and the data, doing research on the internet for similar problems, and finding solutions together with them. Fastly, after some discussions, we have found some potential solutions for forecasting and recommending systems.

I have taken my part in applying knowledge in the lecture into recommendations and came up with the feature to recommend items for every customer of the e-commerce ecosystem. Besides that, I have built the Cron job scheduler and backend components. I have used nbconvert to transform IPYNB files into runnable python scripts, which helps to make the complete automatic working flow for the system. Another contribution from me is doing preprocessing data after receiving raw data from the sponsor. I also played the dev-ops role in the team and response for deployment, which is to study and find a packaging solution. With dockerization, our product will be deployed in very good status. It will be bugless, stable, scalable, and ready to deploy on the cloud.

b. What I have learned

When implementing recommendations for every customer feature, I have realized the difference between the theory and the real system. There are a lot of new issues which we haven't been taught when solving the solution. The good thing is we can keep following the direction learned in class to achieve the target. I also learn how hard it is to start a new project, how important data is. You can have a very clear idea, a clear solution, but if you don't have enough data, you still can not start to solve the problem. Our team is lucky to have a sponsored project after an effort from Nicole to work with her company. Another lesson that I have learned is how big the AI area is. There are a lot of ideas for improvement and different AI models, approaches to apply to improve the solution. Our team also talked about using a different model, HMM, in forecasting. I also have learned about a simple working flow for providing an intelligent system.

c. How can I apply the knowledge and skills learned

Doing the class workshop helps to give me some first sight of how to use them in a real project. The architecture design and data processing knowledge being taught in class also help a lot to build an intelligent system. Spending time practicing knowledge acquisition, knowledge representation, training, and testing models makes me more familiar with AI projects. I hope that in the future, I will have more chances to apply all the knowledge to the real project and into real life.

Report by: Tan Nicole Ongoco

a. Personal contribution

During brainstorming, I suggested the idea of doing an order forecasting project, which was a problem raised by one of our company project managers. The team decided to push through with this suggestion, due to the availability of data and the real-life solution it would solve. With this, I coordinated with the project manager to obtain the datasets needed by the team, conduct interviews to determine the current pain points of the company's forecasting methods, and gather feedback on the product result.

For the solution itself, I worked on developing the web application with flask, setting up the database in the cloud, and proposing the initial system architecture. As expected from each member, we each did our part for the documentation and the presentation.

b. What I have learned

On the technical side, I have learned that data truly is the fuel for AI. We were lucky to have "clean" data, as the company's raw data was stored in the database. Knowing the data structure and the fields available, the team quickly came up with several ideas for models in the early stages of the project. However, as initial data-discovery and quick modeling were done, we realized that the dataset provided by the company had too little order transactions to create accurate predictions and product recommendations. Due to the limitations in the dataset, we have decided to only focus on popular products.

I also started this course with limited knowledge in Python. By developing a Web App based on python and reviewing the notebooks done by my teammates, I'm glad that I was able to practice and hone my skills in this programming language.

On the non-technical side, I feel fortunate to work with a team with complementary skills. We were thus able to segregate the workload based on each of our expertise. I've learned knowledge discovery from Chow Kok Peng, deployment from Minh Tien, and forecasting models from Pow Look. I've also realized the importance of a project lead and I'm thankful that it was one of the requirements set by ISS. Our project lead, Pow Look, did a great job in getting the team coordinated and delivering the project on time.

c. How can I apply the knowledge and skills learned

With regards to the course, the project helped us understand the whole process of creating an intelligent system - from knowledge acquisition, knowledge representation, training and testing models. Being familiar with this process, hopefully we can give greater focus on models and algorithms in the upcoming projects.

In terms of work, by knowing the importance of data gathering, I hope to spend more time in gathering data (e.g. Inventory History, Anonymous User Behaviour such as clicks). I also hope to continually monitor and improve the forecasting models to the company's larger datasets.

Appendix 6 – List of Abbreviations

Abbreviation	Definition
MaxE	Maximum Absolute Difference Error
SKU	Product's Stock Keeping Unit. It is unique per product
RMSE	Root Mean Squared Error
ARIMA	AutoRegressive, Integration, Moving Average
SARIMA	Seasonal AutoRegressive, Integration, Moving Average
Rolling MA	Rolling Moving Average
LSTM	Long-Short Term Memory RNN

Appendix 7 – Questionnaire on Forecasting PoC

POC will be done using MONO order data. However, we wish to scale this to Swee Lee / other brands in the future:

1. **How is order forecasting currently done?**
    a. Exponential time smoothing formula using Shopify Orders (Unfortunately document can't be shared due to confidentiality)
    b. We also take an average of the number of stocks sold in a month over the past X months and identify how many to order for a stipulated set of months. E.g. order 3 months worth of sales for a specific brand or SKU.

2. **How often is forecasting needed? Does this change depending on the store?**
    a. For most brands - stock ordering is done monthly
    b. For less in-demand brands, quarterly

3. **What would be the main motivation/s for order forecasting (Pick all that apply)**
    a. [X] Minimize unsold inventory (e.g. due to storage costs / value depreciation)
    b. [X] Improve availability of products (e.g. Reduce OOS)
    c. [X] Minimize cost of procurement from dealers (e.g. Unpopular products can be procured quarterly instead of monthly) - some products have minimum order quantities
    d. [ ] Others:

4. **Aside from past sales, are there other data used to supplement order forecasting data?**
    a. [X] Views on product page (E-commerce only) - this helps identify stocks that are getting more popularity, good to order more if we're stocking low amounts
    b. [X] Promotions / Events which would affect sales
    c. [X] Others: Cost of goods sold - but usually confidential

5. **Desired output for Order Forecasting?**
    a. [X] Excel
    b. [X] Custom Dashboard
    c. [ ] Others:

6. **Other features that would be useful to CMs / Sales?**
    a. [X] Cluster/Associative Analysis - Determine which products are associated / bought frequently together) ?
    b. [X] Filtering/Notification on Low Stock items
    c. [ ] Others:

7. **Will we be able to know the inventory level of the products, say at start of month etc..?**
    a. Past data won't be available, but we can pass on current inventory

8. Are the product_sku in the products.csv file complete. We noticed some product_sku in orderproducts are not listed when matched with products.csv file. The list is as shown :

{'50-K61-FOAM', 'B-M80-AC-BLK', 'B-M80-EB-BLK', 'B-M80-SAD-BLK', 'B-M80-VEG-GRY', 'CPN-BAG-STRAP-DPS-BLK', 'M80-TT-BLK', 'PFX-PB-L-BLK', 'PFX-PB-L-SLV', 'PFX-PB-M-BLK', 'PFX-PB-M-SLV', 'PFX-PB-S-BLK', 'PFX-PB-S-SLV', 'SRB-L-BLK', 'SRB-S-BLK', 'STICKER', 'TEAR-AID', 'TEX-BLK', 'WCS-L-BLK', 'WCS-M-BLK', 'ZP1-L-BLK'}

a. Prefixed with `B-` are B-Stock / Refurbished items (Usually only 1 stock, and won’t be available regularly)
b. TEAR-AID / STICKER / 50-* / TEX-* / SRB-* / WCS-* / ZP1 (Spare items, discontinued
c. PFX - Data to be given Manually

Appendix 8 – Questionnaire on Feedback for Forecasting PoC

QUESTIONS:
Multiple choice. Mark with [X]
Feel free to expound if needed!

Question 1: Do you find the order-forecasting visualisation helpful?

[ ] Yes - It would be useful to see the trends
[X] Yes - But it would also be good to see ________ (Would be good to filter by product type)
[ ] No - Not so useful and can be ommitted. Shopify should be enough for visualization. Only the numbers are needed

Question 2: Do you have a current method to measure accuracy?

[ ] Yes - We use _____
[X] No (We currently measure the difference between actual and forecasted. Good to have it automated now!)

Question 3: How do you find the accuracy of the models used?

[X] Good - It would be interesting to test LSTM on our larger dataset
[ ] Good - But would be good to achieve an accuracy of ______
[ ] N/A - No opinion. More testing needed

Question 4: Any comments (Optional)

This PoC is a good first step. Interested to see the accuracy in the next few months