Retail Time Series Analysis & Product Recommendation

1. Introduction

When successful retail stores like Ace Hardware grow, executives are most interested in finding out expected sales figures, recommending products to users, analyzing various trends and understanding the factors for maximizing sales. This project seeks to bring the power of data science to enable analysis involving high dimensional data that is hard and even impossible to perform using conventional analysis. Using big data and data science techniques, we seek to create a system for product recommendation, time-series predictions on sales and margin figures and find out the best way to maximize profits and sales while minimizing costs. We want to provide recommended actions based on the insights drawn from the data, with prioritization placed on the largest business impact. The key methods used in this study are Skip Gram with Negative Sampling for finding product embeddings, time series decomposition, ARIMA and regression models for forecasting. We also propose different methods for validating our results.

2. Goals and Objectives

As described above, in this project we want to draw inferences and insights and provide recommendations that are useful for the business. Along this line, the high-level goals for this project are as follows -

2.1 Drawing insights from the data

We want to provide recommended actions based on the insights drawn from the data, with prioritization placed on largest business impact. As such we analyzed the past trends of the net sales across the top stores and the top selling product.

2.2 Product recommendation based on sales data

A product recommendation is basically a filtering system that seeks to predict and show the items that a user would like to purchase. Collaborative filtering is a technique that can filter out items that a user might like on the basis of reactions by similar users. Recommendation systems generally require some user history to predict what a user may want to buy next. However, we did not have enough data points regarding user's history in our dataset. Hence, we approached this problem in a novel way by doing Item-Item based collaborative filtering instead of the traditional user based or content based methods. We define the notion of similarity as *products that are bought together should have a similar representation irrespective of the features that the product might have*. Our proposed model can be used by Ace Hardware to predict similar items for a given item such that when a user views a certain product online, they'll be suggested the most similar items. Also, based on the current cart of the user we can suggest items that are bought frequently with the items in the cart.

2.3 Time series prediction for sales, revenue and product cost

We seek to build time series model using SARIMA to forecast sales, cost and profit which would be a very useful for Ace Hardware in planning and strategizing. In addition, valuable insights about seasonality and trend of sales will be used to determine the best time to promote and stock certain products to encourage sales.

3. Dataset and Data Preprocessing

3.1 Dataset

The dataset provided by Costello contains customer transaction data from Jan 1, 2015 to Dec 31, 2018 where we can derive Net Sales, Gross margin, time purchased, promotion usage and other fields. The external dataset was utilized for analysis using Dow Jones Industrial Average (DJIA) and weather data from NOAA from the same time period as the Costello's. The DJIA dataset contains daily closing prices and timestamps and weather dataset contains daily precipitation, average temperature, the maximum and low temperatures.

3.2 Preprocessing

When it comes to net sales, gross margin and cost, these columns had less than 40 missing values total, out of 36 million rows of data. These rows were simply dropped. Columns like ‘Promo/Discount’ column, missing values were filled with ‘NA’ values. A new feature was added to indicate whether a promotion was used or not. In case of missing weather data, temperatures were filled using linear interpolation. To compare different fields with different scales, data was normalized to make comparison easier. Additionally, for time series analysis, we aggregated the data as described in the later sections.

3.3 Insights

1. Trends in Sales over months and the effect of holidays

We observed strong trends in the monthly sales over the two years. We also observed some patterns in the sales over the different days of the week and also over different weeks of the year. We can see that there is a very clear pattern in the number of sales over the years. We can see that the sales go down in the month of January, February, March, and pick up in the summer and go down again before they shoot up around November.

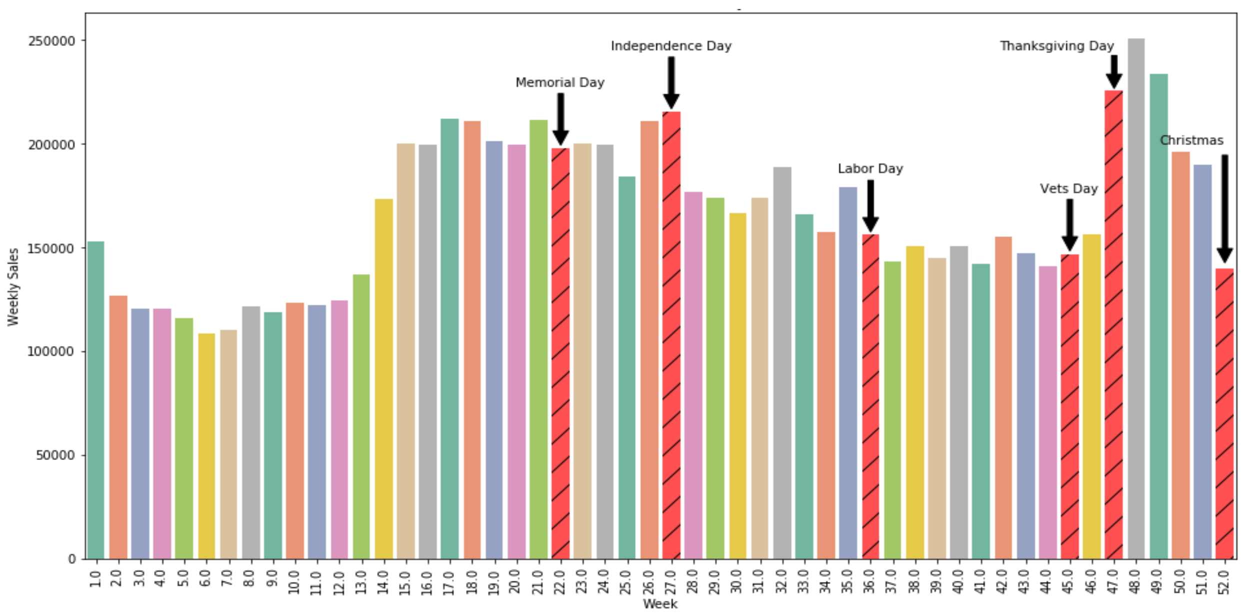
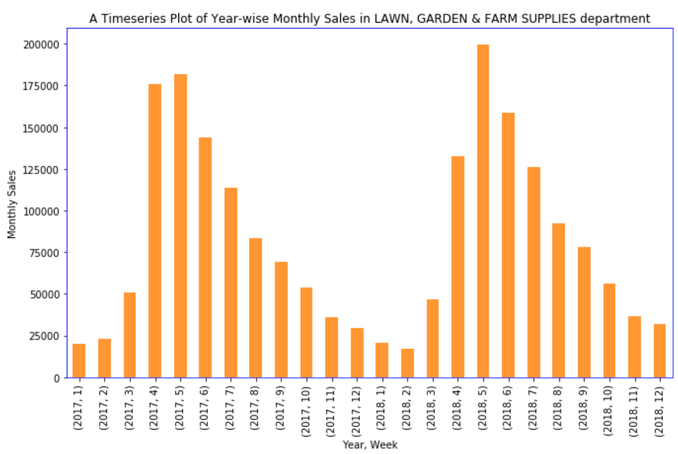
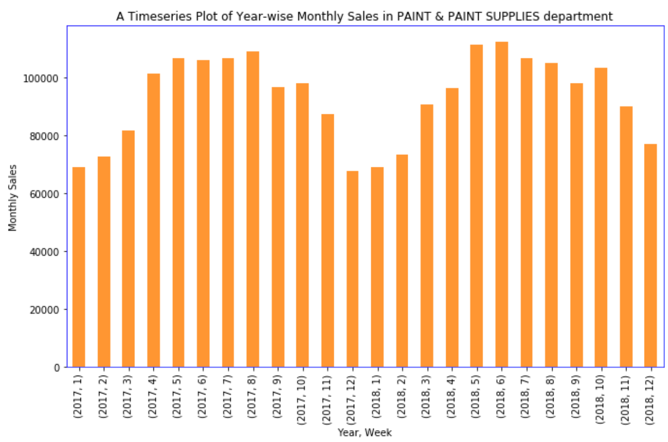
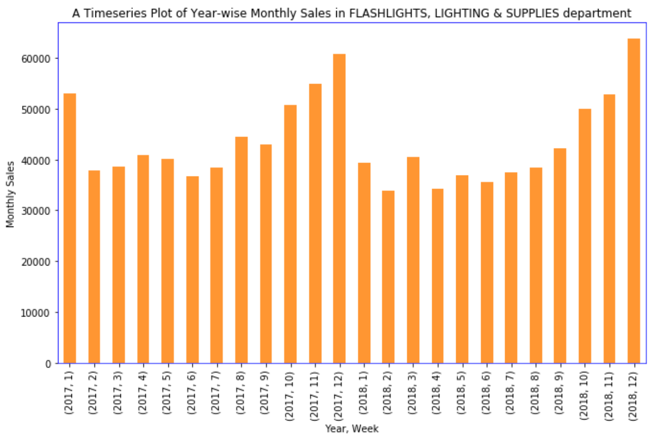


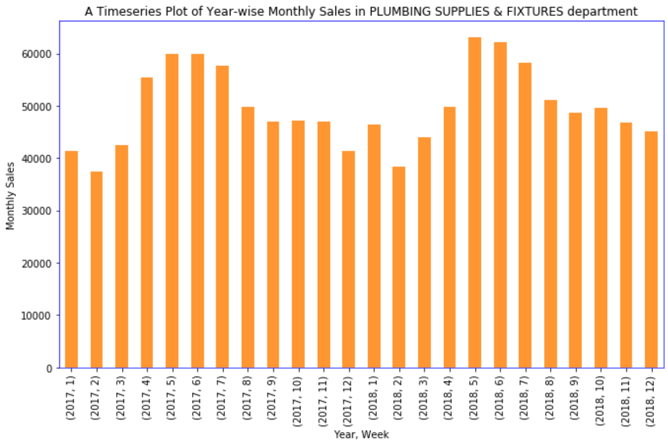
Figure 1 A time series plot of weekly sales

We annotated the major US holidays on the graph that shows the average monthly sales over two years. We can see that there is a remarkable increase in the sales before Thanksgiving Day and the sale goes down during Christmas. Therefore, we suggest Costello to stock on items in the month of April and October so as to not lose out on customers due to items being out of stock.









The above 4 graphs show the monthly sales in top departments (by the number of sales) across 2 years. We can see that sales in "Lawn, Garden & Farm Supplies" department has a very strong seasonality. The number of sales is really high in the summer season as compared to the fall and winter season. Similarly, we see a trend in the sales of other departments as well. The sales in the "Lighting" department shoots up during Christmas which is reasonable as people buy lightings for Christmas.

2. Relationship between Net Sales of the best-selling product and weather

The top performing store by the Net sale is Island Park location and their top selling department is building materials. Both building material sales and precipitation show seasonal patterns. Overall, both seasonality patterns are similar. It rises in August and December and falls sharply in July. But in February, 2015, building materials sales spiked very high, while precipitation remained very low. Given that building material sales are high in January and February each year, other factors like time and temperature may explain the sales trend better.

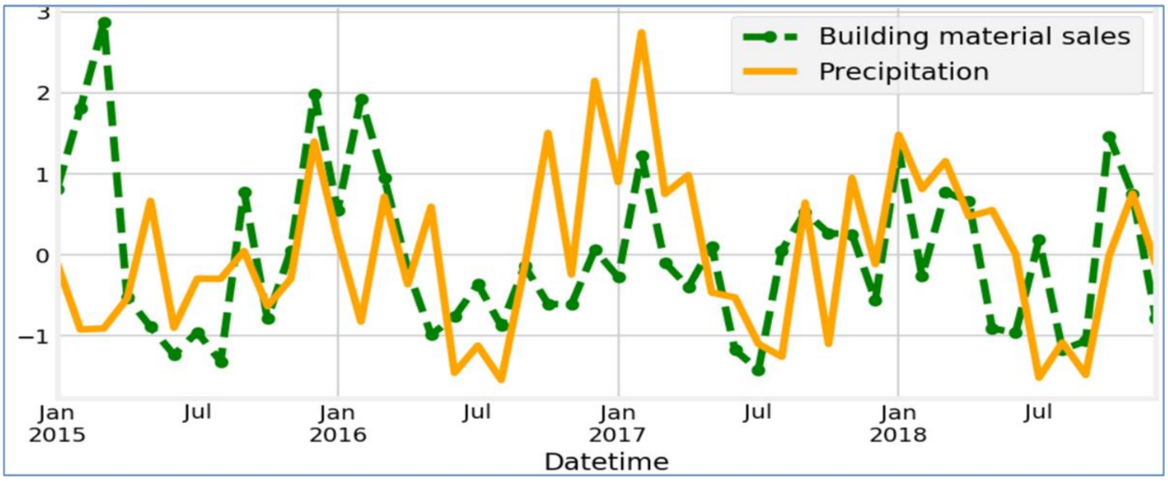


Figure 2 Building Material Sales and precipitation in Island Part 2015-2018

3. Finding correlation between the Net Sales and DJIA stock market performance

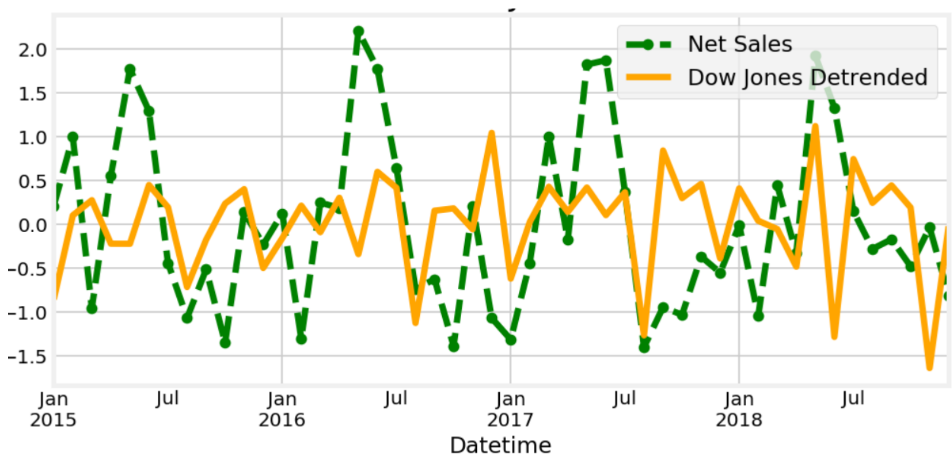


Figure 3 Net Sales and Detrended DJIA Stock Market 2015-18

The stock market performed very well, compared with the total net sales of Costello (an insight Ace Hardware really wanted). In order to compare two factors more clearly, the stock market data were detrended. Both Net sales and the stock market show seasonality but their overall trend does not share much in common. The stock market is much more unpredictable to be useful for correlation analysis.

4. The relationship between Promotion and Net Sales

In order to increase Net Sales, it is necessary to find out whether promotions help encourage customers to buy products. However, the Net Sales and promotion usage are negatively correlated with a correlation coefficient of -0.11. However, with time series analysis, we can identify May and January as months in which both Net Sales and promotion usage spikes very high. But in December 2016, promotion usage was very high while Net Sales remained very low. For this reason, Costello managers should encourage promotion during the month of May (period of summer vacations) when the store historically performed very well and avoid doing promotions during months when Net Sales is expected to be low (July - September, 2017) since sales and promotions are negatively correlated.

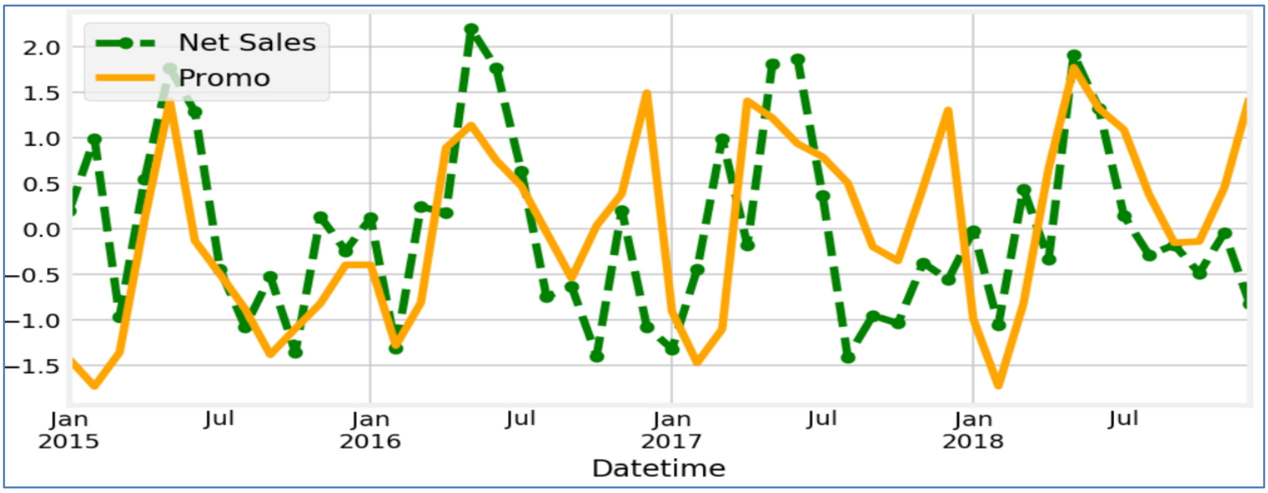


Figure 4 Net Sales vs Promo Trend in 2015-2016

4. Time Series Analysis

The aim of time series analysis is to forecast a future value based on the past observations where the observations are taken at specified times usually at equal intervals. Time series analysis helps us to understand the past trend so we can forecast and plan for the future. Predicting future sales can set a benchmark for the business; we can use to set a level that the business will achieve if nothing changes in the strategy. Ace Hardware would be interested in understanding the observed trends in the past sale so as to answer questions such as when do the stores tend to do well and if they can predict the future sales so that they could formulate better strategies to increase their sales such as planning promotions and stocking up items in inventory. We look at two methods for time series analysis - Regression based approaches and SARIMA based approach.

Approach 1: Forecasting monthly sales with Regression Models

First, we explore a few simple and sophisticated Regression methods for forecasting sales. We will also evaluate the performance of LSTMs on this task. Long Short Term Memory networks – usually just called "LSTMs" – are a special kind of RNN, capable of learning long-term dependencies. They have internal mechanisms called gates that can regulate the flow of information. These gates can learn which data in a sequence is important to keep or throw away. By doing that, it can pass relevant information down the long chain of sequences to make predictions.

Data Transformation: We need to process the data before we can use it with our regression models. Our task is to forecast monthly total sales. We need to aggregate our data at the monthly level and sum up the net sales in a given month. Therefore, we group all the transactions by month and find the monthly aggregated sales. Then, we convert the data to stationary i.e. detrend it by finding the difference in sales over two consecutive months. We need to use previous monthly sales data to forecast the next ones. Therefore, we will use a lookback period of 12 months to forecast the number of sales for a specific month. For this, we create 12 new features where the ith feature corresponds to the monthly sales difference in the month that comes i months before the current month. Adding these features will create a supervised dataset such that each monthly sale will have corresponding 12 features. We divide our dataset into two parts - training set and testing set. The training set consists of sales data for the first 3 years and the test set consists of the sales in 2018.

Checking the usefulness of our futures: We fit a simple ordinary least squares regressor on our data and check the adjusted R-squared metrics. We observe that the r2 score for our model is 0.70, thus it explains 70% variation in the dataset. Thus, we can move forward with training more sophisticated models. Before we build the model we will scale the data such that each feature is between -1 and 1. For this, we will use a MinMaxScaler and transform our dataset before fitting the model.

Modelling: We evaluate different regression models on this task. Specifically, we built and tuned a Linear Regression model, SVM model using linear kernel, Decision Tree Regressor and LSTM. The following figure shows the actual net sales and the predicted net sales for 2018 using our best model. A comparison between the different models can be found in the later section.

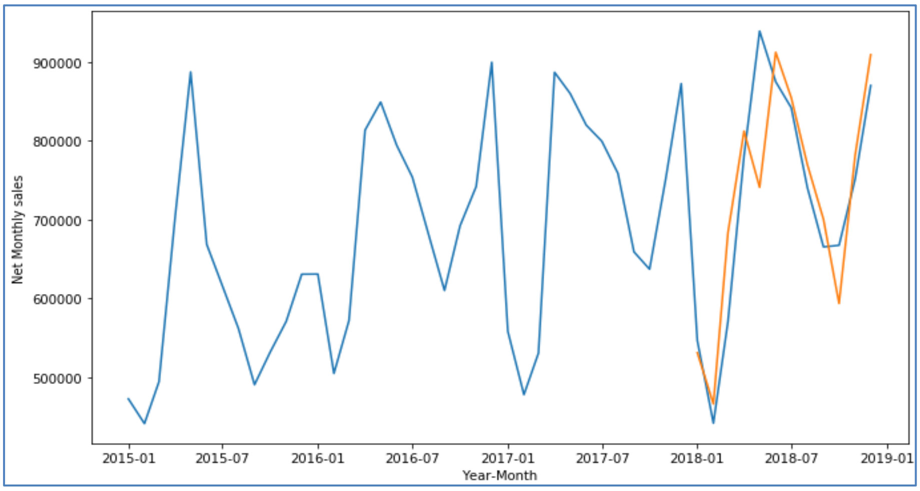


Figure 5 Net sales prediction using Decision Tree Regressor

Approach 2: Forecasting with SARIMA

1. Decomposing Net Sales Trend, seasonality and error

Decomposing time series into trend, seasonality and error is necessary to help make it clear to interpret the historical chart. From the trend graph in figure 6, the net sales declined between 2016 and 2017 and started to rise back to the level of early 2016 in 2018. The seasonality graph captures seasonality in the data very well. It shows spikes in March, May and October and dips in January and July very well. These graphs confirm seasonal patterns we observed from weather dataset as well. Understanding seasonality can be useful for lowering the number of returns being made by the customers. Returns are also negatively correlated with the Net Sales and Gross margin.

2. Net sales and Gross Margin outlook

The net sales trend is clearly sidelining in the past four years data as shown in figure 7 and in the prediction of the next 100 months of time. Meanwhile, the Gross margin is showing rising trend during the same time period as shown in figure 8. While the net sales need to improve, it is a very good sign for Costello that their Gross margin is improving. While they focus on improving gross margin, they should also look at ways to improve their net sales.

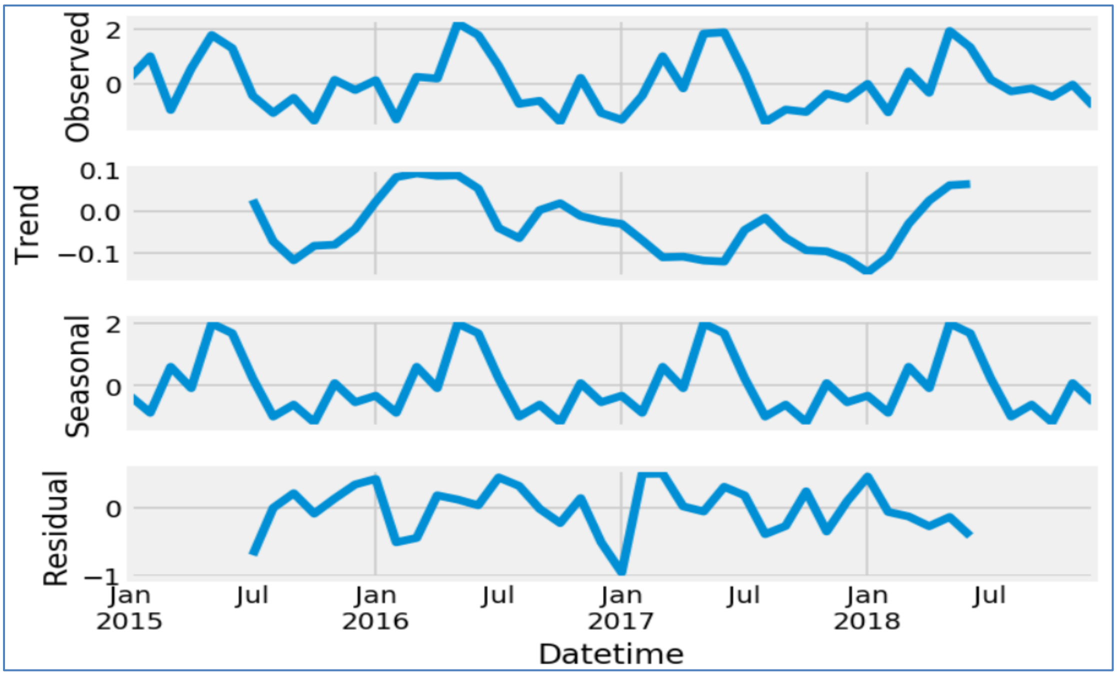


Figure 6 Analyzing Seasonality and trend in sales

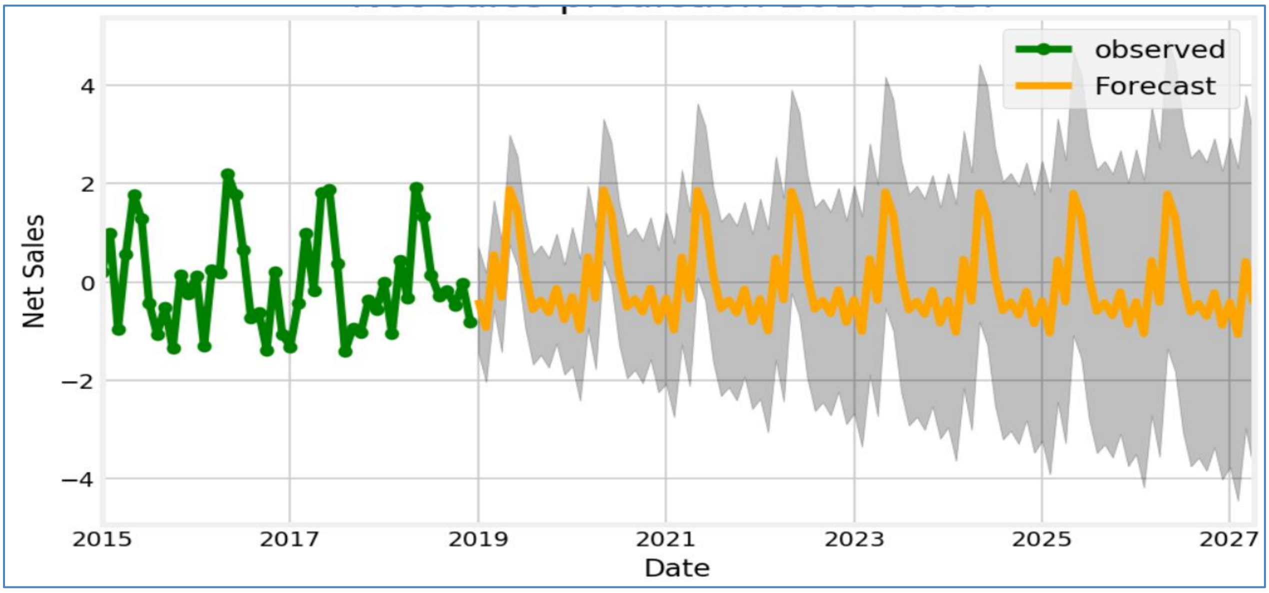


Figure 7 Net Sales prediction 2019-27

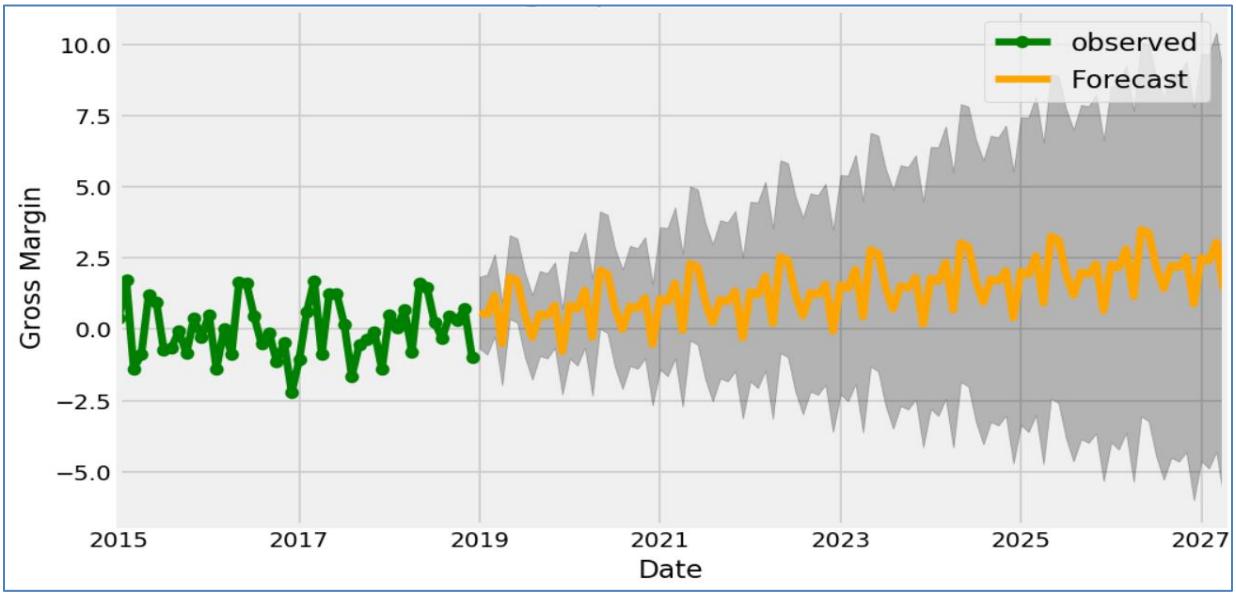


Figure 8 Gross Margin Prediction 2019-27

3. Validating SARIMA Forecasting model

Forecasting was done using SARIMA for 2018 and compared with ground truth data. As shown in the plot, forecasted data fits closely with the actual dataset. The net sales does not exhibit trend in its data but shows seasonality. The hyper parameters for the SARIMA model was fine tuned to reflect this pattern in the dataset.

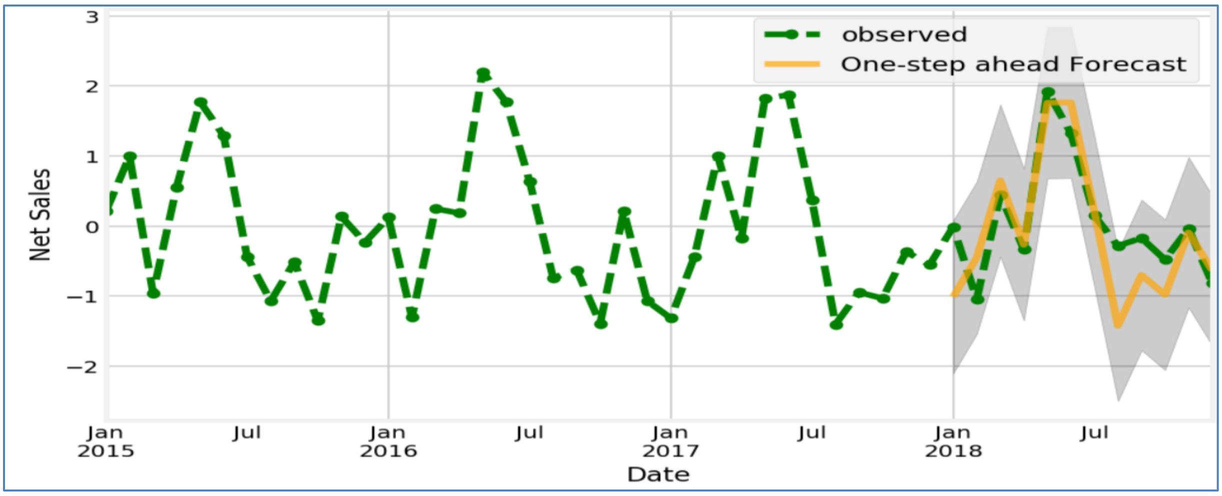


Figure 9 Forecasting Net Sales for 2018 using SARIMA

4. Predicting the Net Sales trend of the top two best selling products (by fineline)

From previous plots, we have examined seasonality and trend of Sales to find out the best time to promote and stock certain products to increase sales and maximize gross margin. It is crucial to know which particular fineline products will perform in the next 5 years. Both indoor paint and bird seed are top two best performing fineline products and show clear rising trend in the next 5 year period. However, bird seed is projected to more than double its net sales, while indoor paint will rise only about 50% during the same time period. In order to continue increasing sidelining Net Sales, it is critical to identify which fineline products like bird seed will continue to grow in terms of net sales in the near future.

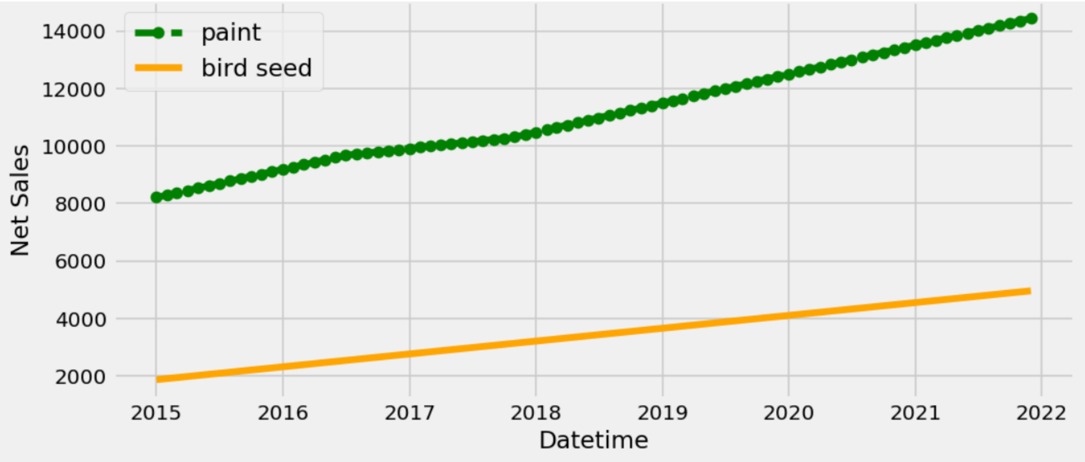


Figure 10 Paint v/s Bird Sales Trend

5. Results

We used Mean Squared Error and R-squared for comparing different models. After hyper parameter tuning and based on forecasting on validation dataset, i.e., forecasting for 2018, we found the decision tree regressor better than other models. Table 2 shows the MSE and R-squared values for all the models.

Table 2: Evaluation Metric for Regression Models

Model Used	Mean Squared Error (MSE)	R^2 Score
Linear Regression	0.06	0.66
Decision Tree Regressor	0.02	0.85
SVR (linear kernel)	0.05	0.72
LSTM	0.06	0.68
SARIMA	0.23	0.65

6. Product Similarity and Recommendation

6.1 Background Research

Recommender Systems are categorized into collaborative filtering (CF) methods [1], [2], [3] and content-based methods [4]. Collaborative filtering is based on user-item interactions and predicts which products a user will most likely be interested in by exploiting purchase behaviour of users with similar interests or by using user's interaction with other products. Item similarities are also at the heart of item-based CF algorithms that aim at learning the representation directly from the item-item relations. Recent progress in neural embedding methods attempt to map words and phrases to a low dimensional vector space that captures semantic relations between words. Specifically, Skip-gram with Negative Sampling (SGNS), known also as word2vec [7], capture the relations between different items in collaborative filtering datasets. We have used SGNS for training our model and capturing the similarity score between two items. We show that our model can induce a similarity measure that is competitive with an item based CF using SVD.

6.2 Approach

In NLP, distributional hypothesis is a popular hypothesis for representing words. It says that a word is defined by the company it keeps i.e. its context. Therefore, the representation of the word should depend on its context such that two words that occur in similar contexts have similar representations. We extend this analogy to our problem statement and say that two products that are bought together frequently should have a similar representation. The context for a word is defined as the words that appear around it in a window of size say "k". Here, instead of sentences (as in the case of traditional NLP), we have unordered sets of items that are brought together. We have considered *unordered* sets because the ordering of items in a bill has no meaning unlike the order of words in a sentence. We can now apply popular NLP methods for learning the embeddings of our products. Please note that in our approach, we only care about which items are bought together and as such, we are not concerned about the specific features of the item or the transaction. Thus the notion of similarity is only dependent on which items are bought together and which items are not.

We propose to apply Skip Grams with Negative Sampling (SGNS) for item-based collaborative filtering. Item-item SVD is used as a baseline model.

In NLP applications, each word is represented as a feature vector. The SGNS method aims at finding words representation that captures the relation between a word to its surrounding words in a sentence. Given a sequence of words $(w_i)_{i=1 \text{ to } K}$ from a finite vocabulary $(w_i)_{i=1 \text{ to } W}$, the Skip-gram objective aims at maximizing the following term:

$$\frac{1}{K}\sum_{i=1}^{K}\sum_{-c\leq j\leq c, j\neq 0}\log p(w_{i+j} | w_i)$$

where c is the context window size (that may depend on w_i) and $p(w_j \mid w_i)$ is the softmax function:

$$p(w_j \mid w_i) = \frac{\exp(u_i^T v_j)}{\sum_{k\in I_W}\exp(u_i^T v_k)}$$

where u_i and v_i are latent vectors that correspond to the target and context representations for the word w_i and the parameter m is chosen empirically and according to the size of the dataset. Negative sampling is used to reduced computational complexity of the above equation.
The application of SGNS to CF data is straightforward once we realize that a sequence of words is equivalent to a set or basket of items. Since items have no spatial information between them, we treat each pair of items that belong to the same receipt number as a positive example.

6.3 Implementation

We implemented the approach by considering CLASS NAME and FINELINE NAME for every product. However, we did not find the results convincing. A number of dissimilar items (which were never bought together) were grouped together since they belonged to the same class name. Hence, we trained our model with ITEM DESCRIPTION field which gave promising results. Following is the detailed implementation of how we are recommending items to users for a given item:

1. Loaded the entire 4-year data and applied the data cleaning and preprocessing steps as mentioned above.
2. Grouped the entire data by their receipt numbers & date and for each receipt common to a number of transactions, we aggregated the results of Item Description fields.
3. So now for each receipt, we have a list of all the items that were bought together.
4. We removed all the transactions that were bought independently, i.e., receipts having just one item on their list. This is because, we found out that by including these items, our model was getting distorted. For example, if the item WATER in the dataset appeared 80 times independently and 20 times with another item OREO COOKIES, we found out that the similarity score between them was better if we removed the independent occurrences of WATER.
5. Next, we used the SGNS model with the above list of items as the input parameter and tuned the following hyper-parameters:
    a. Window Size = 5, since this was the average number of items in every bill.
    b. Size = 100, i.e., we want to learn word vectors of 100 dimensions.
    c. Min Count = 1, since we do not wish to eliminate out any item even though they may be rare.

6.4 Results

Given an item, we can now recommend a customer which product he/she would likely buy next based on our SGNS model. From our model, we can now find all the items having a higher cosine similarity score with a specific given item. We grouped items having a higher similarity score for a given seed item. We observed that most of the recommended items belonged to the same class thus supporting the hypothesis that customers generally buy multiple items from the same class in a single transaction. Table 1 below presents a list varied items from different departments (topmost selling item from each department) with their top 4 item recommendations when compared to our baseline SVD model. So for each item a customer buys, Costello can now recommend a similar item that the customer would likely want to buy next.

Table 1: Top 4 recommendations for Item2vec model and our baseline model - SVD with items of the same class

Seed Item	**Item2vec - Top 4 recommendations**	**SVD - Top 4 recommendations**
Key Kwikset Kw1	Key Schlage Sc1250pk, Key Arrow Ar1, Photo Frame Keychain, Key Yale Y1	Photo Frame Keychain, Pntbrsh Xl Glide 2.5, Roller Frame 9" 5-Wire, Key Schlage Sc1-Ace250pk
Drain Opnr Hd Liq 32oz	Drain Opnr Hd Liq 64oz, Drain Liq Plumr Pro 80oz, Cleanr Liqud Pro 32oz, Plunger 18" Handle Red	Styrene/Pvc Swr/Drain Ftg, Pvc Pressure Fittings Dom, Tire Gauge/Valves, Drain Liq Plumr Pro 80oz
Switch Qt Wh 15a120/277v	Swtch Qt3w Wh15a120/277v, Decora Wallplate 2g Wht, Wallplat1g Gfci Decor Wh, Decora Wallplate 1g Wht	Swtch Qt3w Wh15a120/277v, Bulk Toggle Switches,Joint Compounds, Slide Dimmer Switches

Packing Tape Clr 54.6yd	Carton Seal Tape Clear, Ez-Start Tape1.88"X60yd, Bubble Wrap 12"X60, Corrugated Box 18x18x24	Tape Rules, Masking Tapes & Machines, Bubble Wrap 12"X60, Teflon Tapes
Liquid Ant Bait 6pk	Ant Killer Spray 16oz, Killr Ant Terro Liq 1 Oz, Killr Termite/Ant 32oz, Insect Killr Indoor 24oz	Ant Killer Spray 16oz, Insect Baits/Traps, Killr Termite/Ant 32oz, Insect Killr Indoor 24oz

Now we may also want to recommend users, products belonging to different classes and not just of the same class. So for this, we took around 100 items one by one and looked for their class name in the data and checked if it had the same class name as that of the seed item. Table 2 lists down the Top item recommendations with items belonging to a different class for a seed item.

Table 2: Top recommendations for Item2vec model and our baseline model - SVD with items of different class

Seed Item	**Item2vec - Top 4 recommendations**	**SVD - Top 4 recommendations**
Key Kwikset Kw1	Luggage Lock 1-3/8 Tsa, Toggle 5/8plastic Lg Cd2	Knife Util Hvydty Pwrfst, Led Book Light
Drain Opnr Hd Liq 32oz	Fantastik Cleaner 32oz, Gorilla Glue Wht 2oz	Cleanr Drain Acid Roto64, Toilet Seat Rnd Plst Wh
Switch Qt Wh 15a120/277v	Decor Sngl Swtch120/277v, Jnt Cmpnd All Purp 4.5g	Dimmer Slide Led/Cfl La, Gloves Latex Sm 2pack
Packing Tape Clr 54.6yd	Gorilla Tape Black 35yd, Rule Tape 3/8x50' Lufkin	Rule Tape 3/8x50' Lufkin
Liquid Ant Bait 6pk	Bait Station Disposable, Fly Catchers 5 Pack	Bait Station Disposable,

6.5 Validation

Now, that we have a method for finding the top similar items to a given item, we need a method of identifying whether our recommendations make sense or not. Although looking at the recommendations, one could argue that they belong to a similar category and could be bought together more frequently. However, we still wanted to quantify this and understand the true meaning behind our cosine similarity scores. As such, we define another metrics to help us understand if the similarity scores make sense or not. We will consider the Jaccard Index or Intersection Over Union for this purpose. Specifically, we want to find out "how frequently" two products are bought together. We can find the number of times the two products are bought together and divide that by the total number of times the two products are bought.

$$Jaccard\ Index\ = \frac{A\ \cap B}{A\ \cup\ B} = \frac{Number\ of\ times\ A\ and\ B\ are\ bought\ together}{Number\ of\ times\ A\ is\ bought\ +\ Number\ of\ times\ B\ is\ bought}$$

We can now compare the Jaccard Index of two similar items with two dissimilar items and validate if the cosine similarity actually makes sense or not. For this task, the Jaccard Index will serve as the ground truth.

Let us take an example of two similar items and two dissimilar items found using the cosine similarity defined in the previous section.

Table 3: Cosine Similarity and Jaccard Index of different item pairs

Seed Item	Recommended Item	Cosine Similarity	Jaccard Index
Plastic Deep Well Tray	Economy Paint Tray	0.9402	0.3475
	Diet Coke 20oz	0.5346	0.2112
Moist Tol Quick Set-Catalyst Qt	Moist Tol Quick Set-Clear Gal	0.9875	0.4864
	Birdseed Wildbird 20#Ace	0.2352	0.1464

We can see that two similar products also have a higher Jaccard Index than two dissimilar products. This means that products that are bought together frequently will have higher similarity scores. An important thing to note here is that we should not compare cosine similarity with Jaccard Index directly, but instead we should compare the relative difference between two cosine similarities and the relative difference between two Jaccard index.

Future Work

In future we plan to investigate more complex CF models such as Matrix factorization techniques and compare between them and our item2vec model. We will further explore Bayesian variants of skip gram for the application of item similarity. With respect to Time Series Analysis, we would like to use more regression methods such as XGBoost Regressor and ensemble methods which usually give very good results.

References

[1] G. Linden, B. Smith and J. York, "Amazon.com recommendations: Item-to-item collaborative filtering," IEEE Internet Computing, no. 1, pp. 76- 80, 2003.

[2] A. Karatzoglou, X. Amatriain, L. Baltrunas and N. Oliver, "Multiverse recommendation: N-dimensional tensor factorization for context-aware collaborative filtering," Proceedings of the Fourth ACM Conference on Recommender Systems, RecSys '10, pp. 79-86, 2010.

[3] X. Ning and G. Karypis, "SLIM: sparse linear methods for top-n recommender systems," in 11th IEEE International Conference on Data Mining, ICDM, Vancouver, Canada, December 11-14, 2011.

[4] M. J. Pazzani and D. Billsus, Content-Based Recommendation Systems, Brusilovsky P., Kobsa A., Nejdl W. (eds) The Adaptive Web. Lecture Notes in Computer Science, vol 4321, 2007.

[5] Deep recommender engine based on efficient product embeddings neural pipeline, Laurentiu Piciu et. al.

[6] Item2Vec: Neural Item Embedding for collaborative filtering.

[7] Mikolov T, Sutskever I, Chen K, Corrado GS, Dean J. Distributed representations of words and phrases and their compositionality. In Proceedings of NIPS 2013 (pp. 3111- 3119).