Cognitive Demand Forecasting with Novel Features Using Word2Vec and Session of the Day

Rishit Dholakia, Richa Randeria, Riya Dholakia, Hunsii Ashar and Dipti Rana

Abstract Demand Forecasting is one of the most crucial aspects in the supply chain business to help the retailers in purchasing supplies at an economical cost with the right quantity of product and placing orders at the right time. The present investigation utilizes a years' worth of point-of-sale (POS) information to build a sales prediction model, which predicts the changes in the sales for the following fortnight from the sales of previous days. This research describes the existing and newly proposed features for demand forecasting. The motivation behind this research to provide novel features is to obtain an improved and intuitive demand forecasting model. Two features proposed are: Item categorization using word2vec with clustering and session of the day based on the time. The demand forecasting models with traditional features like seasonality of goods, price points, etc. together with our proposed novel features achieve better accuracy, in terms of lower RMSE, compared to demand forecasting models with only traditional features.

Keywords Retail industry · Product categorization · Demand forecasting · Novel features · Word2vec · Word embeddings · Session of the day

1 Introduction

The retail business comprehensively is expanding at a rapid pace. With increasing competition, every retailer needs to viably adapt to the impending demand. This additionally implies there is a growing shift towards efficiency and a conscious step away from excess and waste of the product. In recent times a company's most valuable asset is the data generated by its customers. Consequently, it has become popular to try and win business benefits from analysing this data. Using this approach in big and small scale industries is our aim, hence the focus would be to provide a much more intuitive approach to utilize the data generated, by using latest advancements

R. Dholakia (✉) · R. Randeria · R. Dholakia · H. Ashar · D. Rana
Sardar Vallabhbhai National Institute of Technology, Surat, India

D. Rana
e-mail: dpr@coed.svnit.ac.in

V. K. Gunjan et al. (eds.), *Modern Approaches in Machine Learning and Cognitive Science: A Walkthrough*, Studies in Computational Intelligence 885,
https://doi.org/10.1007/978-3-030-38445-6_5

like word2vec based word embedding and session of the day. This would help the retailers in business strategy improvement, by using accurate forecasting of the sales of every item.

1.1 Motivation

In the past research papers, some of the traditional and obvious features such as price, holiday, stock code were used to perform demand forecasting. On detailed analysis and observation, it was identified that categorizing each product using word embedding based on word2vec and forming their clusters would be more helpful to generalize each product, which provides more intuition in predicting the quantity of the product. Also, in order to prevent the fluctuating variation of the time series data, it was analysed that categorizing the day into sessions would prove to provide more accurate results. So utilizing the traditional features along with the proposed novel features will provide a better accuracy.

1.2 Problem Statement

Improvised demand forecasting using a more intuitive approach using novel features such as word2vec based item category and session of the day based on time.

The further sections of the report are organised as follows. Section 2 talks about the survey done on the prior forecasting techniques and the theoretical background of those techniques. Section 3 describes the proposed framework and methodologies used in this research. Section 4 consists of the pre-processing techniques used on the raw data to prepare it for prediction using machine learning models. Section 5 consists of feature engineering techniques which include clustering of items on word2vec based data and relevant attributes creation to improve accuracy in prediction. Section 6 describes the experimental analysis of forecasting models for analysing the trade-off between different models and trade-off between inclusion and exclusion of novel features. Section 7 consists of the research conclusion and future work.

2 Theoretical Background and Literature Survey

Traditional features such as past weeks' sales data, price of each item, presence of holidays etc. have been used to generate a predictive model. Also, various statistical techniques such as exponential smoothing, ARIMA regression models, SVR, etc. have been analysed for this application. Word2Vec algorithm has earlier been used with respect to different applications requiring word embedding.

2.1 Research and Analysis

Past work in this field includes work related to features and various predictive models used for demand forecasting. Analysis on numerous applications of word2vec was also done.

Word2vec Based Word Embeddings Word2Vec is used in various applications. Some applications are dependency parsers, name entity recognition, sentiment analysis, information retrieval etc. One such application is mentioned by Sachin et al. that is to evaluate exam papers automatically, using word2vec word embedding [1].

Input Features Retail forecasting research done by Fildes et al. [2] mentioned the need to view the POS (Point of Sale) data with respect to different perspectives such as seasonality, calendar events, weather, and past week sales data of an item mentioned in Martin's blog [3]. This was done in order to capture hidden yet significant trends from the data.

Retail product sales data have strong seasonality and usually contain multiple seasonal cycles of different lengths, i.e. sales exhibit weekly or annual trends. Sales are high during the weekends and low during the weekdays, high in summer and low in winter. Data may also possess biweekly or monthly (pay check effects) or even quarterly seasonality, depending on the nature of the business and business location [2]. For this reason, models used in forecasting must be able to handle multiple seasonal patterns, hence gain maximum knowledge from the data. Retail sales data are strongly affected by some calendar events. These events may include holidays, festivals and special activities (e.g., important sport matches or local activities) [2]. Most research includes dummy variables for the main holidays in their regression models. Certain variables are not related to the chosen dataset and hence are ignored for example weather.

To capture the trends in change of demand comparing to past few weeks, be it upward or downward, another set of variables must be included which are the sales of the item in past one week, past two weeks and past three weeks [3].

Predictive Models Kris et al. suggested a few models such as regression trees, principal components regression etc., to be used in this retail forecasting scenario [4]. Regression Trees are decision trees which are used for continuous dependent variable dataset containing too many features interacting in nonlinear ways. To handle nonlinearity, the space is partitioned into smaller regions recursively, to form sub-regions. The resulting chunks are progressively managed to fit simple model [5]. XGBOOST and LGBM are based on regression trees [5]. Principal Component Regression is a regression tree based on principal component analysis. The principal values are calculated and they are used as predictors in a linear regression model [6].

Another researcher Xia et al. suggested that in order to deal with seasonality and limited data problems of retail products, a seasonal discrete grey forecasting model such as Fuzzy Grey regression model and ANN model should be used [7].

Current trends of machine learning models are bagging and boosting which includes XGBOOST and LGBM and other models include SVR and ARIMA for formulating predictive solutions. Bagging and boosting algorithms drastically increase accuracy by learning from weak learners. LGBM is one such boosting algorithm which is fast, distributed, high-performing, and produces generalized results that grows decision trees leaf wise. XGBOOST is another such powerful model which grows trees level wise which provides inbuilt regularization to exponentially increase speed and efficiently work around the problem of overfitting. SVR's are used to perform non-linear regression by interpolating data to multidimensional information space using kernels. Time-series data is handled by using ARIMA models.

XGBOOST, ANN, LGBM were considered for implementation. PCR was not considered for implementation, as PCA was not required on our data. Fuzzy grey regression model was not implemented as it works on limited time series data (50 observational time stamps) for predicting the sales output. SVR and ARIMA were considered for implementation, but as ARIMA could only be used individually on each product it is not mentioned in the paper. SVR uses multidimensional kernels to perform regression, requiring large computational power and memory for large dataset; as a result it was not mentioned in paper.

3 Proposed Framework

From the literature review and after analysing the latest trends, the proposed framework is shown in Fig. 1. for the cognitive demand forecasting with the following objectives:

- Collect and pre-process the data.
- Group the items based on item similarities to make effective models such that the disparity of items is less and accurate predictions can be made on the data using word2vec based word embeddings.
- Derive the Session of the day feature from timestamp information.
- Aggregate features for prediction.
- Predict stock requirement using eclectic machine learning algorithms.

The workflow of the proposed framework along with a brief introduction of each step is as follows and the detailed workflow is mentioned in the later sections.

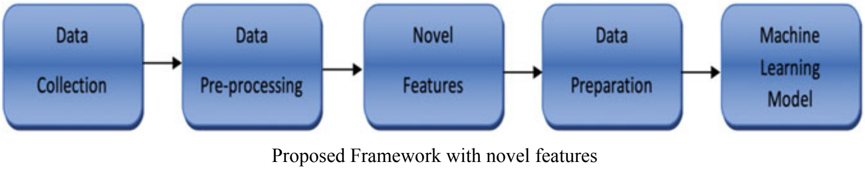


Fig. 1 Proposed framework with novel features

3.1 Dataset Description

UCI repository dataset was used for this research work [8].

- Invoice No: Invoice number uniquely assigned to each transaction. If this code starts with letter ‘c’, it indicates a cancellation.
- Stock Code: Product (item) code uniquely assigned to each distinct product.
- Description: Product (item) name.
- Quantity: The quantities of each product (item) per transaction.
- Invoice Date: Invoice Date and time, the day and time when each transaction was generated.
- Unit Price: Unit price. Product price per unit in sterling.
- Customer ID: Customer number uniquely assigned to each customer.
- Country: Country name. The name of the country where each customer resides.

The data set is a time-series data, consisting of exact time and date of items purchased. This data consists of many transactions throughout the year. The data set consists of around 5,41,910 tuples. It contains one of the most important attribute like Invoice Date and other features such as seasons, weekends, weekdays, etc. can be generated from this attribute.

3.2 Data Pre-processing

Real-world data is often incomplete, inconsistent, and/or lacking in certain behaviours or trends, and is likely to contain many errors. Data pre-processing [9] is a proven method of resolving such issues. Data pre-processing prepares raw data for further processing by transforming raw data into an understandable format.

Anomalous rows and noise were identified and rectified by identifying a pattern. Anomalous rows include negative values for quantity of an item, quantities of products above 70,000 and invalid stock codes. These were removed by using regular expressions and identifying a pattern throughout the dataset. Duplicate rows were removed and un-required columns were dropped. This data set was highly left skewed in terms of unit price as a result logarithm transformation [10] was applied onto the price column. Heteroscedasticity was also identified and removed using box-cox transformation [11].

3.3 Feature Engineering Approach

One of the characteristics of any data set is garbage in - garbage out. It means that if a dirty data is passed in a model we would get garbage values out of the model resulting in low accuracies. So in-order to get the best accuracies the concept of feature engineering is used. Feature engineering consists of feature extraction i.e. only including the required features and eliminating the irrelevant features, feature segmentation, feature creation and feature augmentation.

Segmenting the Day into Sessions Splitting the Invoice Date into hour categories known as sessions of the day (e.g.: morning, night etc.). This is done in order to analyse the trends of the product market during different time periods of the day. This would even help in finding the anomalies in the data set that might not be figured out through normal database scan.

Analysis of Past few Weeks Another strategy includes finding out the variations in quantity sold for items over a range of past few days. This gives an important weight age in weekly or fortnightly analysis for each product, by deriving a feature like past few day sales. Since the data is of non-perishable products, weekly analysis of past 3 weeks would provide a much better trend intuition. Using this feature to accurately predict the safety stock required for the product as suggested by Martin [3].

Day of the Week InvoiceDate is an important feature for stock prediction. The date attribute can also be broken down into the days of the week like Monday, Tuesday etc., after which it is converted into numerical attribute, signifying weekdays and weekends, indicating the past daily or weekly sales of the product [3]. This is even used to gain knowledge and insights on the seasons for a better forecast.

Word2vec Based Categorical Data This is an approach where a categorical feature is created by clustering of similar products, as mentioned in the later section of this paper, which is generated using word2vec based word embeddings to provide the required information about their similarity with other products.

3.4 Machine Learning Model

This research also focuses on training the data using recently popular machine learning models like XGBOOST, LGBM and classic model like Artificial Neural Networks. It also focuses on the accuracy trade-off between different models and accuracy trade-off between inclusion and exclusion of the word2vec based categorical data and sessions of the day feature. The output of this model is the forecasted stock value of the product.

4 Feature Engineering

Feature Engineering involves using domain knowledge to create, extract and reduce features. This is necessary to generate comprehensive knowledge from the data for more accurate results. Features are augmented and processed. Incubating features allows the model to derive higher accuracy and knowledge from the data.

4.1 Feature Segmentation

Feature segmentation is performed for better analysis of the data. The InvoiceDate attribute is broken down into year, month, week, day and hour. These categories would generate one of our proposed novel feature, session of the day and other traditional features like numerical attribute for particular day of the week and past 3 weeks' sales of each product.

Session of the day provides a more intuitive knowledge into the data that can be used in developing particular market schemes for a given product to be sold, depending on its highest demand time of the day in hours i.e. morning (8:00–12:00), afternoon (12:00–16:00), evening (16:00–20:00) or night (20:00–24:00). For example, the greatest percentage of newspaper sold is during the morning session. This provides an intuitive knowledge regarding the outflux session during the day of the particular product.

Identification of day of the week i.e. weekday or weekend caters into analysing trend of which period of the week an item is bought. Usually decorative party items or drinks are bought during the weekend as compared to the weekdays.

As suggested by Martin in his blog [3], previous sales for past 3 weeks for each product are derived. It is an important feature to predict the requirement of a particular item for the current week with respect to its sales in the past few weeks. This would provide an additional incite to the model, to identify the sudden rise or drop in sale of an item which can be attributed to an indirect un-identified attribute, only identified by using past week sales. 3 weeks were chosen as retail data contains non-perishable goods such as furniture.

4.2 Feature Augmentation

Feature augmentation is used to add additional features from different domains other than the one the data has come from. This helps in increasing the accuracy of the model by considering possible relationships outside the scope of the data provided. Another important feature for the construction of the model is the Boolean feature of holiday. The feature is generated by considering InvoiceDate and mapping this date to a holiday dataset, obtained from a holiday api. This feature helps to identify the surge in sales of an item prior to the holiday and during the holiday period [3].

4.3 Word2vec Based Categorical Data

Each of the 500 unique products is mapped to a category manually, in order to create a feature which would provide more intuition into every product. For example category handbags consist of the unique handbags such as floral handbags, checkered handbags and tote bags. This results in the total creation of 351 different categories.

For easier organization of these categories in the dataset and modularity of each category, it is better to term them as subcategories and group them into their respective categories. For example, let us consider the fact that we have different types of bags like handbags, soft bags, and water bags that belong to the category "bag". These categories are the clusters which is discussed later in this section. Aggregation of these different sub categories into categories can be done, as shown in Fig. 2.

To perform categorization of these sub-categories, human labour is needed to individually identify similarities between the subcategories. This is a more cumbersome task, as compared to the former task of sub categorization. This creation of categories from subcategories can be completed in less time by providing the machine with the intelligence of knowing the meaning of the subcategories and finding similarities between them to form categories. So in order to resolve this problem word2vec based word embedding is used [12]. Meaning of each word is found using the concept of word2vec based word embedding and categorized into clusters. Figure 3 shows the workflow for the creation of the categorical data.

Working with Word2vec Based Word Embeddings The below common words as shown in Fig. 4 are represented in one hot encoded form, which is in a m * n matrix. Presence of a word in a particular row is marked with 1.

For applications like language translation and chatbots, it becomes difficult for learning algorithms, like Recurrent Neural Networks to predict the next word as there is no relation among them. For example: the words pencil and handbag are nowhere close to each other in the one hot encoded representation. So, if the sentence "I want to buy a pencil for school" and "I want to buy a bag for school", it should predict "for school" for the later as well. The solution to this is to create a matrix

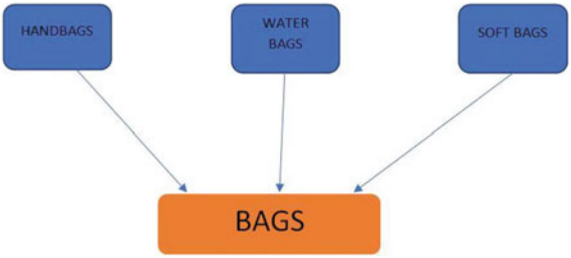


Fig. 2 Categorization of sub-categories

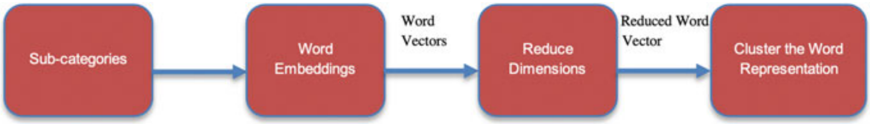


Fig. 3 Workflow of creating the categorical data using word2vec based word embedding

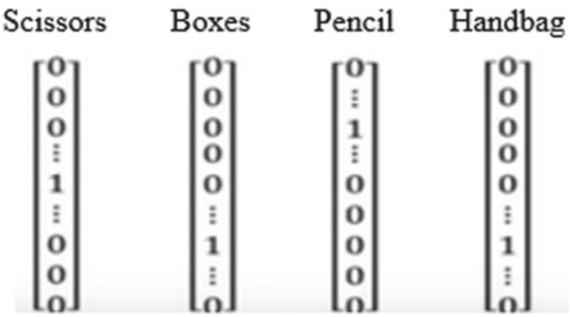
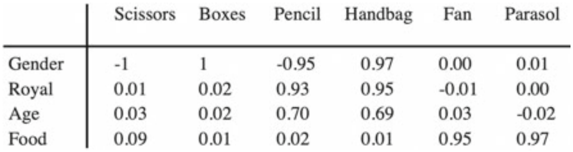


Fig. 4 One-hot encoded values of the products

that describes the word. The description of these words is considered depending on different relations they have with the factors that represent them. Some examples of these factors are the words age, gender, royal, food etc. which are then related to each of these words present. This concept of relating words to factors based on the values is known as word embedding. In order to generate these word embeddings, word2vec model is used [13]. These embeddings are then used as input into the machine learning models. The meaning of each subcategory like pencil, box etc. is provided by the word embeddings. Using these word embeddings the similarity of each sub-category is known, so that they could be clustered together. Representation of these embedding values is in numeric form. Figure 5 shows a part of the embedding matrix that is used for each of the vocabulary words.

The dimensions of these embedding matrices can vary. There might be about 300–1000 dimensions' space based on the amount of corpus available. Usually there are ready embedding matrices available to be used. These words can be trained using a word2vec model, but this requires a lot of computation, longer time period and requires a huge corpus of data to work with. For this paper a pre trained model of the word vector has been used. Google News data set is one such corpus whose vocabulary word embedding is used [14]. Since this is a 300 dimensions vector it is computationally expensive to create clusters of the words and would be difficult to visualize those words. For this reason, the dimensionality of the vector has to be reduced. PCA, principal component analysis is used in order to reduce these dimensions for easier computation and for better visualization of the object space [15]. For this paper a two-dimensional space is used to view the points.

Working with Clustering of Data Now that the representations have been made, these representations have to be clustered into groups to represent the metadata to be worked with. For this research k-means clustering is used because it provides



	Scissors	Boxes	Pencil	Handbag	Fan	Parasol
Gender	-1	1	-0.95	0.97	0.00	0.01
Royal	0.01	0.02	0.93	0.95	-0.01	0.00
Age	0.03	0.02	0.70	0.69	0.03	-0.02
Food	0.09	0.01	0.02	0.01	0.95	0.97

Fig. 5 Embedding matrix of the products

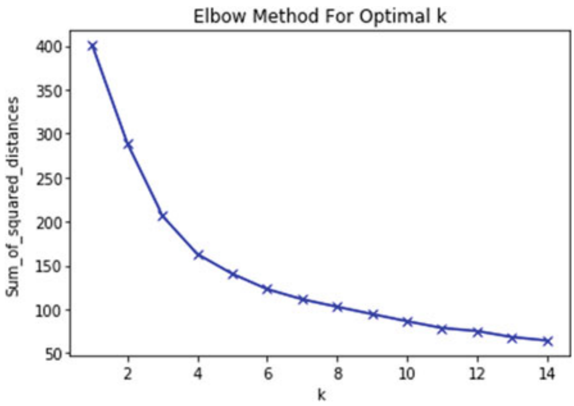
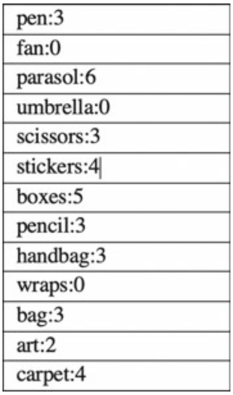


Fig. 6 Optimum cluster representation

tighter clusters. The clusters formed represent the group to which the subcategories of products belong to. As shown in Fig. 6, elbow method is used to find out the optimum number of clusters [16].

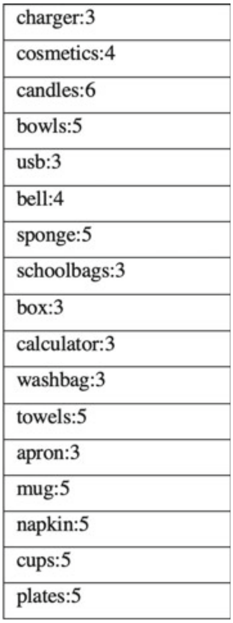
A cluster value range from 5 to 8 and a dimension range of 2–5 were analysed using a cross-grid to determine which values form the correct clusters for the sub-categories. The ranges of dimensions are taken to know how many dimensions provide sufficient amount of information for the similarity of the products. A combination of 4 dimensions and 7 clusters are used to create the categorical feature. Due to the creation of this novel feature, it is required that the results of clustering be compared with a benchmark to see if the categorization done is right. Due to absence of previous benchmarks to be compared, this data was compared with online stores like Jarir Bookstore [17] and Home Centre [18]. The results proved to be accurate but with only a few miscellaneous categorizations. The screenshots below provide the cluster to which each product belongs to after applying clustering.

From Fig. 7., it can be evaluated that art is a different category and subcategories like pencil, pen, and bags belong to the same category in both the online website of



pen:3
fan:0
parasol:6
umbrella:0
scissors:3
stickers:4
boxes:5
pencil:3
handbag:3
wraps:0
bag:3
art:2
carpet:4

Fig. 7 Novel cluster comparison with Jarir bookstore



charger:3
cosmetics:4
candles:6
bowls:5
usb:3
bell:4
sponge:5
schoolbags:3
box:3
calculator:3
washbag:3
towels:5
apron:3
mug:5
napkin:5
cups:5
plates:5

Fig. 8 Novel cluster comparison with home centre

Jarir bookstore as well as the clusters that have been made using word2vec based word embedding. Similarly, Fig. 8. depicts that mugs, cups, plates belong to a category dining, in home-centre's website as well as, in the above assigned cluster.

5 Experiment Analysis

This section describes implementation of the proposed features and application of various models on the aforesaid data set. The experiment was executed using the Anaconda Navigator on the python 3.6 platform. The models ANN, LGBM and XGBOOST were trained and evaluated on the basis of calculated root mean squared errors values.

5.1 Model Trade-off

In this section the data is trained on different models like artificial neural networks, regression trees like extreme gradient boosting and light gradient boosting.

The ANN model is used for both continuous and discrete dependent feature, it is one of the most flexible models as it requires less amount of parameter tuning and provides good output result. The XGBOOST model is used for its speed in

Table 1 Comparison of various models accuracy

Model	Train-set RMSE	Test-set RMSE
ANN (rectifier)	30.73	31.86
ANN (tanh)	24.84	28.29
LGBM	34.34	34.342
XGBOOST	8.22	11.86

training the data and even provides effective output values for both, the train and test set compared to other gradient boosting algorithms. The LGBM model is used as it provides faster training and a much more generalized model i.e. less variance between the training and test accuracies.

The pre-processed data contains an overlap of a month that is used for analysis and showcasing the output of the working prediction model. The data is first randomised; this is done to provide different variations from features throughout the year. The data is split as 90% train-set and 10% test-set data.

In this paper, the ANN model is trained on two different activation functions namely rectifier and tanh with two hidden layers as there are 475,455 tuples to be trained. Each node consists of 100 nodes as this an optimum parameter tuning that provides the least RMSE value by ANN. In order to get the least RMSE value for XGBOOST number of iterations have been set to 100, maximum depth of the tree is set to 25 and the number of CPU cores used for this training is 3. The LGBM model works as XGBOOST does but uses extra parameters like number of estimators, which is tuned to 6000. It is used in comparison to XGBOOST because it has a faster execution rate.

It is observed from Table 1 that ANN and LGBM have high RMSE values but a good generalization in-comparison to XGBOOST, where the training RMSE value is the least and has a bit higher test RMSE. Since the variance of test set and train set of XGBOOST are not very distinct as compared to the other two algorithms and since it provides the least RMSE value, it would be safe to consider this model for further analysis.

5.2 *Feature Trade-off*

Once the model analysis is done and the best model is selected, it is required to check if the additional features added to the dataset helps in improving the predictions or not. The two main features, the categorical feature based on word2vec and sessions of the day are analysed. In this feature trade-off, the RMSE values are compared by removing one of these two features at a time and training it. This includes consideration of over-fitting also for checking the importance of the two models. Since the best model was XGBOOST as mentioned above in the model trade-off, this same model with its same parameter tuning is used for training and testing. It can be depicted from Table 2 that the exclusion of novel features over-fits the model by a large margin.

Table 2 Comparison of XGBOOST accuracy with different features

Model	Train-set RMSE	Train-set RMSE
XGBOOST (without proposed novel item category feature)	14.33	189.24
XGBOOST (without proposed novel sessions of the day feature)	11.03	25.78
XGBOOST (with proposed item category and sessions of the day features)	8.22	11.87

But the inclusion of these features reduces the variance margin by a large amount, hence generalizing the model.

6 Conclusion

Demand forecasting was evaluated with novel features using word2vec based word embedding which was used to generate clusters where each product belonged. Another novel feature, session of the day was also generated. The improved data set was trained using three models namely ANN, XGBOOST and LGBM. Upon evaluation, it was found that XGBOOST on a dataset using clusters and sessions of the day provided an improved accuracy in terms of lower RMSE as compared to previous research papers on demand forecasting.

7 Future Works

The following is the proposed future work:

- Recency, Frequency, Monetary (RFM) segmentation allows retailers to target specific set of clusters of customers with offers that are much more relevant for their particular behavioural patterns.
- Recommendation system for retailers suggesting the new items to be kept in the store.
- Sentiment analysis of product to generate better sales brand wise.
- Using stacking of training models LGBM and XGBOOST.
- Applying the proposed novel features into other applications such as tweets categorization, weather forecasting etc.

References

1. Sachin, B.S., Shivprasad, K., Somesh, T., Sumanth, H., Radhika A. D.: Answer script evaluator: a literature survey. International Journal of Advance Research, Ideas and Innovations in Technology **5**(2) Ijariit (2019)
2. Fildes, R., Ma, S., Kolassa, S.: Retail forecasting: research and practice. MRPA Paper, University Library of Munich, Germany (2019)
3. Retail store sales forecasting. https://www.neuraldesigner.com/blog/retail-store-sales-forecasting
4. Johnson Ferreira, K., Hong Alex Lee, B., Simchi-Levi, D.: Analytics for an online retailer: demand forecasting and price optimization. Manufacturing and Service Operations Management **18**(1), 69–88 (Winter 2016)
5. Regression trees. http://www.stat.cmu.edu/~cshalizi/350–2006/lecture-10.pdf
6. Principal Components Regression. https://ncss-wpengine.netdna-ssl.com/wp-content/themes/ncss/pdf/Procedures/NCSS/Principal_Components_Regression.pdf
7. Xia, M., Wong, W.K.: A seasonal discrete grey forecasting model for fashion retailing. Knowledge-Based Systems **57**, 119–126. Elsevier (2014)
8. Online retail dataset. https://archive.ics.uci.edu/ml/datasets/online+retail
9. Data pre-processing. https://en.wikipedia.org/wiki/Data_pre-processing
10. Dealing with Skewed data. https://becominghuman.ai/how-to-deal-with-skewed-dataset-in-machine-learning-afd2928011cc
11. Dealing with Heteroscedasticity. https://www.r-bloggers.com/how-to-detect-heteroscedasticity-and-rectify-it/
12. Word embeddings: exploration, explanation, and exploitation (with code in Python). https://towardsdatascience.com/word-embeddings-exploration-explanation-and-exploitation-with-code-in-python-5dac99d5d795
13. Word embedding and Word2Vec. https://towardsdatascience.com/introduction-to-word-embedding-and-word2vec-652d0c2060fa
14. Google news dataset. https://code.google.com/archive/p/word2vec/
15. Principal component analysis. https://machinelearningmastery.com/calculate-principal-component-analysis-scratch-python/
16. Elbow method (clustering). https://en.wikipedia.org/wiki/Elbow_method_(clustering)#cite_note-3
17. Jarir bookstore website. https://www.jarir.com/sa-en/
18. Home center website. https://www.homecentre.in/