Exploratory and Predictive Analytics of User Preferences from Kaggle LEGO-Toys Datasets Using Spark ML

Pritika Bahad¹, Preeti Saxena¹ and Raj Kamal²

¹School of Computer Science and IT, Devi Ahilya University, Indore, MP, India
²Department of Computer Science and Engineering, Prestige Institute of Engineering Management and Research, Indore, MP, India

E-mail: bahad.pritika@gmail.com

Abstract. Apache Spark is an open-source distributed data processing framework. The paper presents a processing architecture for exploring and predicting user preferences using Apache Spark. The architecture is evaluated on LEGO-toys datasets of period 1949-2019 using the Spark Machine Learning (ML) algorithms. The large datasets analyzed consist of LEGO-toys parts, categories, themes and colour features. Spark ML algorithms are applied as (i) k-means analyses of clusters to identify commonalities in LEGO-toys themes and colours, (ii) classifications using the Support Vector Machines (SVMs), Naïve Bayes (NB) and Random Forest (RF) algorithms for theme-preference identification, and (iii) linear regression, decision tree regression, RF, and Gradient Boost for regression analyses to identify the colour-shift in user preferences. The paper elucidates the steps for analytics based on Spark. The results for exploratory and predictive analytics are presented. The evaluation metrics shows that the ensemble regression prediction is better when compared to other algorithms. The analytics give many interesting results. For example, LEGO company’s products have become more colourful (children preferences exhibiting colours spectral-shift and width), diversified and multifaceted over-the-time. The architecture helps in discovering future directions for the new designs in future LEGO products. The proposed architecture can be successfully employed in the related domain to predict product and user’s preferences.

1. Introduction

Data generated by industries is expanding day by day. Many industries are exploiting data resources for competitive advantage and to formulate future strategies [1]. The analytics improve the sales performance, and increase the understanding about the customer behaviour and user’s interest. The NB, NB multinominal (NBM) and SVM have been applied for classification of user preferences in the SMS [2]. SVM has been applied for user-preference recommendations in Stanford Large Network dataset [3]. Present paper focuses on analytics on large scale datasets for user preferences and exploratory and predictive analytics using Apache Spark, a Big Data Analytics tool.

The LEGO Group [4] is one of the largest toy companies in the world. Exploratory and predictive analytics of user preferences using their 1949-2019 large datasets is interesting aspect of present study. Apache Spark has been widely used as an efficient data processing system due to in-memory computations [5]. It helps to perform complex analytics faster [6]. PySpark [7][8] is the Python API for Spark.

Section 2 provides the review of earlier research in the field. The research includes the ones related to analytics on LEGO-toys and other large datasets using Apache Spark. Section 3 describes the
methodology adopted for various classifications and clustering algorithms in distributed environment. Section 4 presents the results as evaluation metrics. The metrics used for the evaluation of classification algorithms are Accuracy, Precision, Recall and F1-Score. The accuracies of prediction model are figured out by Root Mean Square Error (RMSE). Section 5 gives the conclusion drawn and possible future applications of proposed processing architecture are suggested.

2. Review Literature

Statistical and machine learning techniques can be applied to explore popularity and predict demand of specific LEGO part based on user preferences. Table 1 shows the previous work.

| Previous Study and Year of Publication | Dataset | Analysis | Tools Used | Techniques Used |
|---------------------------------------|---------|----------|------------|-----------------|
| Xiaodan Zhang [9] 2015                | LEGO    | LEGO parts Construction Patterns Discovery | Gensim, scikit-learn, D3.js | Latent Dirichlet Allocation Model, K-Means Clustering algorithm |
| Bartneck C et al. [10] 2018           | LEGO    | Temporal trends of new sets and bricks, Colour distribution pattern | R using data table package | Generalised linear models, Shannon's entropy for the colour distribution |
| Joel Carron [11] 2019                | LEGO    | Correlation between Sets, Most dominant colour in set | Plotly, D3.js | Statistical Tools |
| Archenaa J. et al. [12] 2016          | Health Data | Risk for getting cardiac ailment by analyzing heartbeat rate | Apache Spark ML | Naïve Bayes, K-Nearest Neighbour |
| Mohamed et al. [13] 2018             | Daily News for Stock Market Prediction, StockTwits, Market News data | Real time forecasting of stock market, Sentiment analysis on Stock Twits and Market News data | Apache Spark ML | Naïve Bayes, K-Nearest Neighbour, Logistic Regression, Decision Tree, Random Forest, Support Vector Machine, Canonical Correlation Analysis, Linear Discriminant Analysis, Naïve Bayes, Random Forest, Decision Tree |
| Dahiyaa P et al. [14] 2018           | UNSW-NB15network | Network Intrusion Detection | Apache Spark ML | Naïve Bayes, K-Nearest Neighbour, Logistic Regression, Decision Tree, Random Forest, Support Vector Machine, Canonical Correlation Analysis, Linear Discriminant Analysis, Naïve Bayes, Random Forest, Decision Tree |
| Alsaedi A. et al. [15] 2019          | Alzheimer’s Disease | Detecting cognitive impairment | Apache Spark ML | Naïve Bayes, K-Nearest Neighbour, Logistic Regression, Decision Tree, Random Forest, Support Vector Machine, Canonical Correlation Analysis, Linear Discriminant Analysis, Naïve Bayes, Random Forest, Decision Tree |
| Bahad P et al. [16] 2020             | Pima Indian Diabetes Database | Classify patients with diabetes | scikit-learn | AdaBoost Classifier, Gradient Boost Classifier |

The previous research focused on forecasting, predictions and decision making using statistical and ML techniques [8-16]. Examples applications of analytics are predicting the market trends, accurate planning of the operations, rendering better services to consumers, and faster logistics, fake-news detection, market-analysis, disease-prediction, and intrusion detection. Table 1 summarizes the related research carried on large datasets and includes the studies on the LEGO dataset. The presented work uses Spark ML and Python for analytics of LEGO datasets.
3. Methodology
The present study used Spark ML library for analytics. It deploys scalable machine learning library which implements processing techniques of Classification, Regression, and Clustering. Figure 1 shows the proposed processing architecture. The architecture shows the steps for exploring and predicting user’s preferences using datasets.

First phase does pre-processing and feature selection. Clustering identifies the commonalities within dataset on the basis of feature selected. Classification is performed to identify any specific feature preference using supervised ML algorithms and ensemble techniques. The paper access Model performance by accuracy performance of various classifications and clustering algorithms in distributed environment based on the Apache Spark framework.

Figure 1. A Processing Architecture for Exploring and Predicting User’s Preferences Using Spark

The steps shown in processing architecture are:

3.1. Conversion of Data into DataFrame
A DataFrame is an immutable distributed collection of data in the form of named column. DataFrames can process large collection of structured as well as semi-structured data. DataFrame supports a broad range of data formats and sources. DataFrames in PySpark created from the multiple file formats of data sets, such as CSV, JSON, XML, or Parquet file. DataFrame created using an existing RDD or the databases, such as Hive, Cassandra, HDFS, or a local file system. A Spark Context represents the connection to a Spark cluster. It is used to create DataFrames on that cluster [17].

3.2. Pre-processing and Cleaning DataFrame
The real-world data is far from being clean or complete [18]. Data cleaning and preparation are the critical steps in the exploratory and prediction tasks. The primary goal of cleaning is to detect and remove errors and anomalies. That increases the value of data for analytics and decision making. One
must either remove the noisy data or fill in the missing data in data pre-processing. Summary statistics helps in identifying the missing or corrupt data. Further many ML techniques, such as logistic regression and SVM are algebraic, hence need the numerical inputs. The datasets include theme names and colour names as categorical feature. Therefore, OneHotEncoder encoding is used to convert a category features into the binary vectors.

3.3. Identify theme correlation and colour variety
Scatter matrix is used for approximately determining, whether a linear correlation exists between the multiple independent variables [19]. The most important feature of correlation is used to predict the colour-shift preferences over the time.

3.4. Clustering similar themes
K-means algorithm along with the Principal Component Analysis (PCA) is used to discover the clusters of similar themes [20]. PCA is a powerful statistical tool for analysing datasets [21]. It gives simpler way to find the most significant features of dataset. It attempts to combine the highly correlated features and represent the data with fewer linearly uncorrelated features. It gets features set as an input and projects them to a lower-dimensional feature set. Further it reduces them to a few important principal features, and finally visualizes them.

PCA with k-means is a potential method to visualize high-dimensional voluminous data. Further elbow method is used to find out the optimal number of clusters. The algorithm takes the inputs, and partitions the data into $k$ clusters. The clustering is such that the similarity within the cluster is very high and across the clusters is very low. The cluster example used in the work prepares 3 clusters of features theme_id and colour_id.

3.5. Classification to identify theme-preferences
Apache Spark ML’s SVM, NB and RF classifiers are used to identify the theme-preferences.

3.5.1. Support Vector Machine
SVM is employed for both regression and classification tasks [22]. It carries out classification by deciding the hyperplane. Hyperplane is a vector to divide a plane into the two classes for two-dimensional space. The vectors that define the hyperplane are support vectors. A high dimensional dataset can be classified efficiently using support vectors machine.

3.5.2. Naïve Bayes
NB is a supervised learning algorithm. It is based on Bayes’ theorem. It computes conditional independence between every pair of features given the value of class variable [23]. The posterior probability of target class is computed as

$$P(y|X) = \frac{P(X|y)P(y)}{P(X)}$$

(1)

where $X = \{x_1, x_2, x_3, \ldots, x_n\}$ represents the feature set within features.

By substituting $X$, the posterior probability of target class can be represented as

$$P(y|x_1, x_2, x_3, \ldots, x_n) = \frac{P(x_1|y)P(x_2|y)\ldots P(x_n|y)P(y)}{P(x_1)P(x_2)\ldots P(x_n)}$$

(2)

where $P(x_i|y)$ is probability of predictor $x_i$ given class $y$; $P(y)$ is prior probability of class $y$; $P(x_i)$ is prior probability of predictor $x_i$ and $1 \leq i \leq n$. NB is a fast and highly scalable algorithm. The algorithm can be utilized for binary and multiclass classification.
3.5.3. Random Forest
RF algorithm [24] is employed for classification and regression methods. The RF uses ensembles of regression tree, where each tree grows randomly in selected subspace of data. The final prediction is obtained by aggregating over the ensemble.
RF is fast and easy to execute. It provides highly accurate predictions and competent to handle a large number of input variables without over-fitting. RF [24] is a predictor consists of a group of randomized base trees \( r_n (x, \Theta_m, D_n), m \geq 1 \) where \( \Theta_1, \Theta_2, ..., \Theta_m \) independent and identically distributed outputs of a random variable \( \Theta \). These random regression trees are pooled to shape the aggregated regression estimate.

\[
\tilde{r}_n (X,D_n) = E_{\Theta} \left( r_n (X,\Theta_m, D_n) \right)
\]

(3)

where \( E_{\Theta} \) represents expectation of the random parameter on \( X \)s and the data set \( D_n \).

3.6. Regression Analysis to predict colour-shift in Children Preferences
Different regression models having different regularization parameters are used to determine how well either of these models predicts the shifts based on the feature sets. This paper evaluates the use of ensemble machine learning Linear Regression, Decision Tree, RF and Gradient Boosting techniques [22-24] to predict colour-shifts preferences. Bagging minimizes the variance while Boosting reduces variance and bias of the classification [25]. The paper [16] provides further details regarding these methods.

3.6.1. Linear Regression
Linear regression models the relationship between two variables by fitting a linear equation to input data. Logistic regression is a well-liked method to predict a categorical response. In spark.ml, binomial and multinomial logistic regression can be used to predict a binary and multiclass outcome respectively [25].

3.6.2. Gradient Boost
Gradient boosting is an ensemble learning technique. It is utilized for regression and classification problems [26]. It aims to minimize a loss function by moving in the reverse direction of the gradient. A loss function describes “how well” model is at building predictions for a given set of parameters [16]. The choice of loss function depends on the nature of problem. RMSE is calculated to evaluate the processing architecture prediction performance.

4. Results
The experiments have been carried out on a grid of Core™ processor Intel® CPU i7-4790U 3.60 GHz with 16 GB RAM each. The codes were written in Python 2.7 using Spark 2.3.1-Scala 2.11 and NumPy. A benchmark LEGO-toys dataset which is used to evaluate the ML algorithms performance contains the Parts/Sets/Colours and Inventories from 1949 to July, 2019. The specification of LEGO datasets is depicted in Table 2.

| Table 2. Specification of Kaggle LEGO Datasets |
|-----------------------------------------------|
| Table and Table size (in rows*columns) | Attributes | Data Type | Unique values |
|------------------------------------------|------------|------------|---------------|
| Sets 14863*5                             | set_num    | object     | 14863         |
|                                           | Name       | object     | 12902         |
|                                           | Year       | int64      | 69            |
|                                           | theme_id   | int64      | 629           |
|                                           | num_parts  | int64      | 1188          |
| Themes                                   | Id         | int64      | 671           |
Exploratory Analysis is performed on initial investigations on LEGO datasets for (i) discovering patterns, (ii) checking assumptions with the help of summary statistics, and (iii) graphical representations. Visual and quantitative methods are used to understand the datasets, identify the missing values and outliers of datasets.

The set_id is key attribute to relate sets, colours, part_categories, part_relationships and themes. The visualization through graphs presented here helps in interpreting the results by putting them into visible form.

Figure 2 shows the count of new sets evolved during 1949 to 2019. This shows significant growth in release of new sets. Also, the average number of parts in sets manufactured is increasing in current decade. The statistical analysis shows that sets have become larger and more diverse in terms of participating parts.
Figure 2. Set Released Over Years

Figure 3 shows that many colours and shades have emerged and used over time. The colours introduced later on in the decade have a shorter lifespan than those that were introduced in early years.

Figure 3. Distinct Colour Used Over Years

The box plot in Figure 4 represents the colour variety over the current decade. Figure 4 shows box plot representing colour varieties. The interquartile range (IQR) is utilized to calculate the year wise limits of LEGO colour varieties and to discover outliers. Further outliers are removed by considering observations that are falling outside of the rectangle or hyper-rectangle.
Figure 4. Box Plot Representing Colour Variety

The top ten colour share of inventory part is shown in Figure 5.

Figure 5. Top 10 Colour Share

Themes of LEGO are designed over the years on the basis of previously designed themes. The Figure 6 shows number of themes released over the years. The graph in Figure 6 shows the ratio of parent themes (represented by blue colour) is considerably lesser than the current themes (represented by red colour) during the years 2010- July, 2019. This implies that the participation-shares of the parent themes in current themes evolving in current decade is reduced.

Figure 6 shows the number of themes released from Jan, 1949 to July, 2019.
The LEGO dataset contains vast information and insights about how each LEGO part is important for a particular LEGO set. Different sets and parts belong to a theme. Thus, each LEGO set is different from the other in the terms of theme and part components. The analysis on the LEGO database can lead to get information that helps the manufacturer of LEGO toys for improving the design. It also helps the buyers more familiar with the toys in order to make efficient purchasing decision.

The specialization of LEGO themes is identified using k-means clustering. LEGO dataset contains about 614 themes, 135 colours and 11673 sets till 2019. Most of the sets having more than 175 parts in them. All these are used to generate the number of clusters among all LEGO sets. The K-means clustering takes the input data as set, colour, themes dataset. A correlation matrix is figured out that
represents the correlation of theme on different attributes. The attributes highly influencing the theme are generated by Principle Component Analysis are shown in Figure 7. It shows plots of theme classes based on two principal components, (a) set and year (b) parts and set of LEGO dataset. The elbow method is utilized to determine optimal number of cluster to be formed. Figure 8 shows the results of Elbow method.

![The Elbow Method](image)

**Figure 8.** Result of Elbow Method to Find Out Optimal Cluster Count

Figure 9 shows the results of identified similarities between all sets by computing similarity measures of attributes of these sets. The results give the cluster of similar themes. The relationship amongst the theme is identified and clustered view of similar themes is prepared as shown.

![Clustered View of different Themes during Jan, 1949-Jul, 2019](image)

**Figure 9.** Clustered View of Different Themes During Jan, 1976 To July, 2019
The results of identified similarities between all sets by computing similarity measures of attributes of these sets clustered view of different themes during Jan,1976 to July,2019. The resultant clusters are represented using scatter plot. The clusters are represented in various colours with centroid drawn with X marker within each cluster. Eventually, creates data visualizations of theme clusters for recommendations of sets from different themes that use similar categories of parts. Commonalities of different themes are identified by clustering. SVM, NB and RF classifiers are used to perform multi-class classification for identifying theme-preferences. The performance accuracy of classifiers used in model can be evaluated by various evaluation metrics. The evaluation matrices namely accuracy, precision, sensitivity, specificity and F1-score are chosen to measure the classification accuracy of the test data shown in Table 3. The observation leads to set sizes is increased overtime. Theme-preferences show that parts have become more specialized as parts are having fewer similarities.

Table 3. Results of SVM, Naïve Bayes and Random Forest Classifier to Identify Theme Preferences

| Classifier      | Accuracy | Precision | Recall | F1-Score |
|-----------------|----------|-----------|--------|----------|
| SVM             | 0.57     | 0.44      | 0.83   | 0.67     |
| Naïve Bayes     | 0.78     | 0.74      | 0.47   | 0.65     |
| Random Forest   | 0.79     | 0.77      | 0.48   | 0.57     |

The colour-shift of theme is predicted using Linear Regression, Decision Tree regression, RF and Gradient Boost. The performance accuracy of candidate regression used in predictive analytics model is measured by using RMSE evaluation metric. RMSE measure is computed as follows:

\[
RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2}
\]  

(4)

where \(y_i\) is predicted value and \(y_i\) is actual value of N samples.

The RMSE is an absolute measure. It is square root of variance of the residuals. It specifies the absolute fit of the model to data points. It represents the closeness of the observed data points to the model’s predicted values. The lower values of RMSE indicate superior fit. The results in Table 4 shows that number of colours in a set has been increasing and will tend to increase gradually by 3-4% per year in future.

Table 4. Results of Linear Regression, Decision Tree Regression, Random Forest and Gradient Boost to Predict Colour-Shift

| Regression            | RMSE |
|-----------------------|------|
| Linear Regression     | 190.805 |
| Decision Tree Regression | 177.117 |
| Random Forest Regression | 183.104 |
| Gradient Boost Regression | 176.543 |
5. Conclusions

The present work gives an architecture for effective predictive analytics for toys datasets, particularly the LEGO datasets. The analytics result leads to insights for user preferred design in future for parts, sets and themes.

Apache Spark ML’s Support Vector Machine, Naïve Bayes and Random Forest classifiers are used to perform multi-class classification for identifying theme preferences. The present work gives the results of evaluation of the the accuracy, precision, recall and F1-score of classifiers, and the comparison. The results suggest Random Forest is very high-quality, robust and versatile method for multiclass datasets. It is not a good choice for high-dimensional sparse data. The colour-shift of theme is predicted using Linear Regression, Decision Tree regression, Random Forest and Gradient Boost. The least value of RMSE evaluation metric for Gradient Boost shows that the ensemble predictor is better than the others.

The suggested processing architecture for exploring and predicting analytics and algorithms are applicable to many products, for example, car models, garments, and children dresses.

6. References

[1] Provost F and Fawcett T 2013 Data Science and its relationship to Big Data and Data-Driven Decision Making Big Data1(1) pp 51-59.
[2] BosaeedS, Katib I and MehmoordR2020 A Fog-Augmented Machine Learning based SMS Spam Detection and Classification System Fifth Int. Conf. on Fog and Mobile Edge Computing (Paris:IEEE) pp. 325-330.
[3] Chidambarathanu K and Shunmuganathan K.L. 2019 Predicting user preferences on changing trends and innovations using SVM based sentiment analysis.Cluster Computing 22 pp. 11877-11881.
[4] Lego Database, https://www.kaggle.com/rtatman/lego-database Accessed 20 July, 2020.
[5] Chellappan S and Ganesan D 2018 Introduction to Apache Spark and Spark Core Practical Apache Spark. (CA :Apress Berkeley) pp 79-113.
[6] Chellappan S and Ganesan D 2018 Practical Apache Spark: Using the Scala API, (CA :Apress Berkeley)
[7] Spark Python API http://spark.apache.org/docs/latest/api/python/index.html, Accessed 20 July 2020.
[8] Sung Y J, Jeong-Hee H and Taek-Soo K, 2005 A statistical model for user preference IEEE Transactions on Knowledge and Data Engineering17(6) pp 834-843.
[9] Xiaodan Z 2015 Mining Lego Datasets to Support Lego Design Dissertation University of Illinois Urbana-Champaign.
[10] Bartneck C and Moltchanova E 2018 LEGO products have become more complex. PloS one, 13(1).
[11] Joel Carron 67 Years of Lego Sets https://mode.com/blog/lego-data-analysis Accessed 20 July, 2020.
[12] Archenaa J and Anita E M 2016 Interactive Big Data Management in Healthcare using Spark. Proc. 3rd Int. Symp. on Big Data and Cloud Computing Challenges (ISBCC–16) (Switzerland :SpringerCham) pp 265-272.
[13] Seif M M Hamed EM and Hegazy A E 2018 Stock Market Real Time Recommender Model using Apache Spark Framework. Int. Conf. on Advanced Machine Learning Technologies and Applications (Switzerland:SpringerCham) pp 671-683.
[14] Dahiya P and Srivastava DK 2018 Network Intrusion Detection in Big Dataset using Spark Procedia Computer Science 132 pp 253-262.
[15] Alsaeedi A, Fong A, Abdelqader I, Delano M. N and Altaie K 2019 Modelling Big Medical Survival Data Using Decision Tree Analysis with Apache Spark.
[16] Bahad P and Saxena P 2020 Study of AdaBoost and Gradient Boosting Algorithms for Predictive Analytics Int. Conf. on Intelligent Computing and Smart Communication 2019. Algorithms for Intelligent Systems. (Singapore: Springer). pp. 235-244, https://doi.org/10.1007/978-981-15-0633-8_22.
[17] Kamal R and Saxena P 2019 *Big Data Analytics: Introduction to Hadoop, Spark, and Machine-Learning* (India: McGraw-Hill Higher Edu.)

[18] Guller M 2015 Machine Learning with Spark. *Big Data Analytics with Spark* (CA: Apress)

[19] Rensink RA Baldridge G 2010 The perception of Correlation in Scatterplots *Computer Graphics Forum* (UK: Oxford Blackwell Publishing Ltd) 29(3) pp 1203-1210.

[20] Ding C and He X 2004 K-means clustering via principal component analysis. *Proc. twenty-first Int. Conf. on Machine learning* pp. 1-9.

[21] Jonathon Shlens 2014 A tutorial on Principal Component Analysis. arXiv preprint arXiv:1404 p 1100.

[22] Wang L 2005 Support Vector Machines: Theory and Applications *Springer Science & Business Media* 177.

[23] Murphy K P 2006 Naive Bayes Classifiers *University of British Columbia* pp 18-60.

[24] Breiman L 1996 Bagging predictors *Machine Learning*. 24(2) pp 123-140.

[25] Schapire RE and Freund Y 2013 Boosting: Foundations and Algorithms *Kybernetes*.

[26] Friedman J H 2001 Greedy Function Approximation: A Gradient Boosting Machine. *Annals of Statistics* pp 1189-1232.