YAŞAR UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

MASTER THESIS

FASHION TREND PREDICTION

USING MACHINE LEARNING TECHNIQUES

İHSAN HAKAN KÖKSAL

THESIS ADVISOR: ASST. PROF. DR. KORHAN KARABULUT
CO-ADVISOR: PROF. DR. OĞUZ DİKENELLİ

MASTERS IN COMPUTER ENGINEERING

PRESENTATION DATE: 28.08.2018

BORNOVA / İZMİR
AUGUST 2018

We certify that, as the jury, we have read this thesis and that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Jury Members:	**Signature:**
Asst. Prof. Dr. Korhan KARABULUT Yaşar University	....
Assoc. Prof. Dr. Mehmet Süleyman ÜNLÜTÜRK Yaşar University	....
Asst. Prof. Dr. Mete EMİNAĞAOĞLU Dokuz Eylül University	....

Prof. Dr. Cüneyt GÜZELİŞ
Director of the Graduate School

ABSTRACT

FASHION TREND PREDICTION
USING MACHINE LEARNING TECHNIQUES

Köksal, İhsan Hakan
MSc Thesis, Computer Engineering
Advisor: Asst. Prof. Dr. Korhan Karabulut
Co-Advisor: Prof. Dr. Oğuz Dikenelli
August 2018

This study addresses the problem of forecasting fashion trend concepts using machine learning techniques. In the fashion world, fashion concepts are continuously and rapidly emerging. These concepts can be either be new fashion concepts or previous fashion concepts that are making a return after not being seen for some time. Recently, with the increasing competition in the fashion industry, forecasting of emerging fashion concepts that will be a trend has provided a great opportunity for the textile companies that want to get one step ahead of their competitors. The main objective of this study is to use the textile database acquired from Followl.io web application to recognize emerging trends, extract technical indicators and predict the likelihood of them becoming a trend with sufficient performance. The forecasting problem for this study is identified as supervised binary classification. Well-known classifiers of various machine learning approaches are evaluated and compared with each other in order to find the classifiers with the most efficient performance. As a result, the ensemble classifiers have provided the most efficient performances, especially the Random Forest classifier with 67.9% accuracy. The ensemble methods using majority voting are also employed and the accuracy increased up to 70.3%.

Key Words: machine learning, fashion trend forecasting, supervised binary classification

ÖZ

MAKİNE ÖĞRENME TEKNİKLERİ KULLANARAK MODA TREND TAHMİNLEME

Köksal, İhsan Hakan
Yüksek Lisans Tezi, Bilgisayar Mühendisliği
Danışman: Dr. Öğr. Üyesi Korhan Karabulut
İkinci Danışman: Prof. Dr. Oğuz Dikenelli
Ağustos 2018

Bu çalışma, makine öğrenme tekniklerini kullanarak moda trend kavramlarını tahmin etme problemini ele almaktadır. Moda dünyasında moda kavramları sürekli ve hızla gelişmektedir. Bu kavramlar ya yeni moda kavramları ya da bir süredir görülmedikten sonra geri dönüş yapan önceki moda kavramları olabilmektedir. Son zamanlarda moda endüstrisindeki artan rekabet ile, bu yeni moda konseptlerinin trend olacağını tahmin edebilmek, rakiplerinden bir adım önde olmak isteyen tekstil şirketlerine büyük bir fırsat sunmaktadır. Bu çalışmanın temel amacı, gelişmekte olan trendleri yakalamak, teknik göstergelerini çıkarmak ve yeterli performans ile trend olma eğilimlerini tahmin etmek için Followl.io web uygulamasından elde edilen tekstil veri tabanını kullanmaktır. Bu çalışma için tahmin problemi denetimli ikili sınıflandırma olarak tanımlanmıştır. Farklı makine öğrenme yaklaşımlarının iyi bilinen sınıflandırıcıları en verimli performansı veren sınıflandırıcıları bulmak için değerlendirilmiş ve birbirleriyle karşılaştırılmıştır. Sonuç olarak, en etkili performansı topluluk sınıflandırıcıları, özellikle de %67,9 doğrulukla Rastgele Orman sınıflandırıcısı sağlamıştır. Daha sonra topluluk yöntemleri çoğunluk oyu ile birleştirilmiş ve doğruluk %70.3'e kadar yükseltilmiştir.

Anahtar Kelimeler: makine öğrenmesi, moda trend tahminleme, denetimli ikili sınıflandırma

ACKNOWLEDGEMENTS

First of all, I would like to thank my supervisors Asst. Prof. Dr. Korhan Karabulut and Prof. Dr. Oğuz Dikenelli for their guidance and patience during this study. I would also like to thank Galaksiya Information Technologies Ltd. for providing the data and support through the study.

I would like to express my enduring love to my parents, who are always supportive, loving and caring to me in every possible way in my life.

İhsan Hakan KÖKSAL
İzmir, 2018

TEXT OF OATH

I declare and honestly confirm that my study, titled “FASHION TREND PREDICTION USING MACHINE LEARNING TECHNIQUES” and presented as a Master’s Thesis, has been written without applying to any assistance inconsistent with scientific ethics and traditions. I declare, to the best of my knowledge and belief, that all content and ideas drawn directly or indirectly from external sources are indicated in the text and listed in the list of references.

İhsan Hakan KÖKSAL

Signature

………………………………..

September 18, 2018

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

SYMBOLS AND ABBREVIATIONS

ABBREVIATIONS:

ANN	Artificial Neural Network
AUC	The area under the ROC Curve
DBMS	Database Management System
FLC	Fashion Concept Life Cycle
FPR	False Positive Rate
EMA	Exponential Moving Average
k-NN	K-Nearest Neighbors
LR	Logistic Regression
ML	Machine Learning
MLP	Multilayer Perceptron
PLC	Product Life Cycle
RDF	Resource Description Framework
RF	Random Forest
RMSE	Root Mean Square Error
ROC	Receiver Operating Characteristic
RT	Random Tree
SGD	Stochastic Gradient Descent
SKU	Stock Keeping Unit
SMA	Simple Moving Average
SVM	Support Vector Machine
TFR	Tag Frequency Ratio
TNR	True Negative Rate
TPR	True Positive Rate
WMA	Weighted Moving Average

CHAPTER 1
INTRODUCTION

This chapter provides an overview of the motivation and the background behind the problem presented in this thesis. The context of the chapter also provides a brief introduction to the concepts that are mentioned in the study.

1.1. Fashion and Fashion Industry

Fashion is a term that defines the popular styles and it is in relation to appearance especially in clothing, footwear, accessories and cosmetics (Kawamura, 2018). As a word, it originally comes from the word modus, which means the manner in English (Barnard, 2002). Inherently, fashion and fashion seasons do not remain stable and change continuously, although some certain fashion trends can remain for a considerable time and some fashion trends can make a return after a period of time (Kertakova, et al., 2018).

In the recent century, changes in the fashion have become faster and this encouraged people to possess the clothes only for a short period of time and later consume more due to the need to keep up with the new trends. This action continued after the millennium and supported the rise of the fashion industry (Turker & Altuntas, 2014). The constantly changing fashion trends and endless demands of the fashion industry increased the competition between the textile companies. These competitions led the big textile companies to find a way to predict the fashion concepts that are going to be trend ahead of time in order to outpace other companies in the market (Easey, 2009). These thoughts and pursues of the textile companies made them turn their attention into the field of machine learning and made them ask the question of “Can machine learning techniques be used to predict the fashion and contribute to the fashion industry” (Dadoun, 2017).

Machine learning techniques have been used in the fashion industry for the motivation of forecasting the fashion trends using a different variety of approaches (Mello, Storari, & Valli, A Knowledge-Based System for Fashion Trend Forecasting,

2008) & (Mello, Storari, & Valli, Application of Machine Learning Techniques for the Forecasting of Fashion Trends, 2010). The application of the trend prediction by using machine learning techniques has been studied for the last couple of decades for the prediction of the stock market trends with considerably successful performances (Patel, Shah, Thakkar, & Kotecha, 2015). These performances encouraged the author of this thesis study to inspire from the trend prediction studies that were conducted for the stock market along with the studies that were conducted for the fashion market.

1.2. Galaksiya and Followl.io

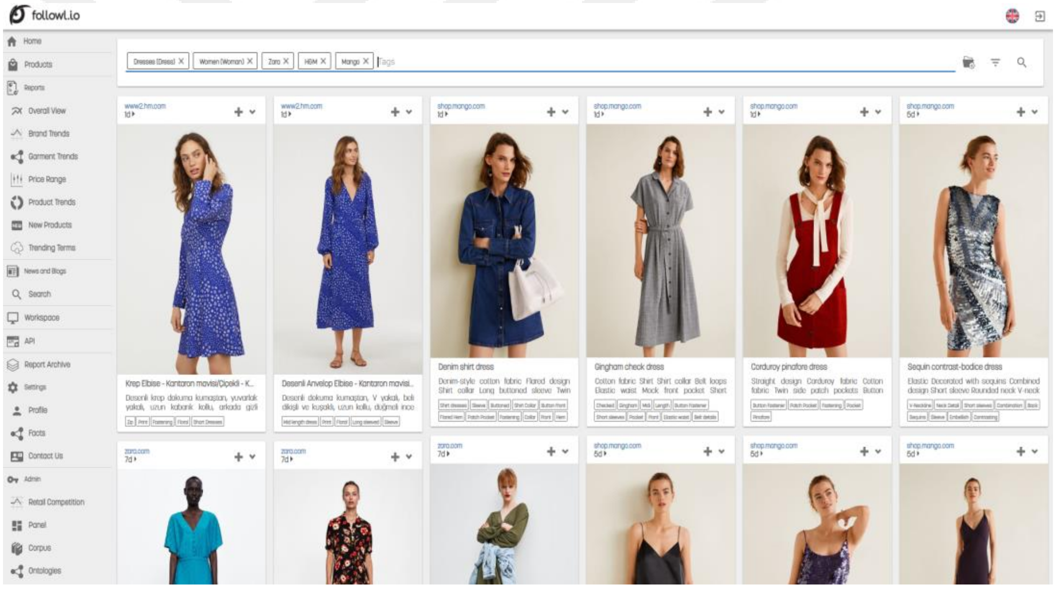
This study is conducted in collaboration with Galaksiya Information Technologies Ltd. and the textile database that is acquired from Followl.io web application that is developed by the company.

Galaksiya Information Technologies Ltd. was founded in 2010 in Ege University by Erdem Eser Ekinci and Prof. Dr. Oğuz Dikenelli with the purpose of doing research and development projects concerning big data, providing consultation and education services as well as developing software projects.

Figure 1.1. Galaksiya Information Technologies Ltd. Logo

Followl.io is a web application that tracks and monitors hundreds of thousands of textile brands on the e-commerce business. It detects real-world concepts in web content, tags the textile products semantically and creates an inquirable knowledge base for preparing a detailed analysis and comparison reports on textile brands, fashion trends, and product prices. With the motto "Fashion needs data", the application aims to become customers' eye on fashion events, emerging trends, and innovations.

Figure 1.2. Followl.io Logo



Krep Elbise - Kantaron mavisi/Çiçekli - K...
Desenli krep dokuma kumaştan, yuvarlak yakalı, uzun kabarık kollu, arkada gizli

Desenli Anvelop Elbise - Kantaron mavisi...
Desenli dokuma kumaştan, V yakalı, beli dikişli ve kuşaklı, uzun kollu, düğmeli ince

Denim shirt dress
Denim-style cotton fabric Flared design Shirt collar Long buttoned sleeve Twin

Gingham check dress
Cotton fabric Shirt Shirt collar Belt loops Elastic waist Mock front pocket Short

Corduroy pinafore dress
Straight design Corduroy fabric Cotton fabric Twin side patch pockets Button

Sequin contrast-bodice dress
Elastic Decorated with sequins Combined design Short sleeve Rounded neck V-neck

Figure 1.3. Followl.io Product Search

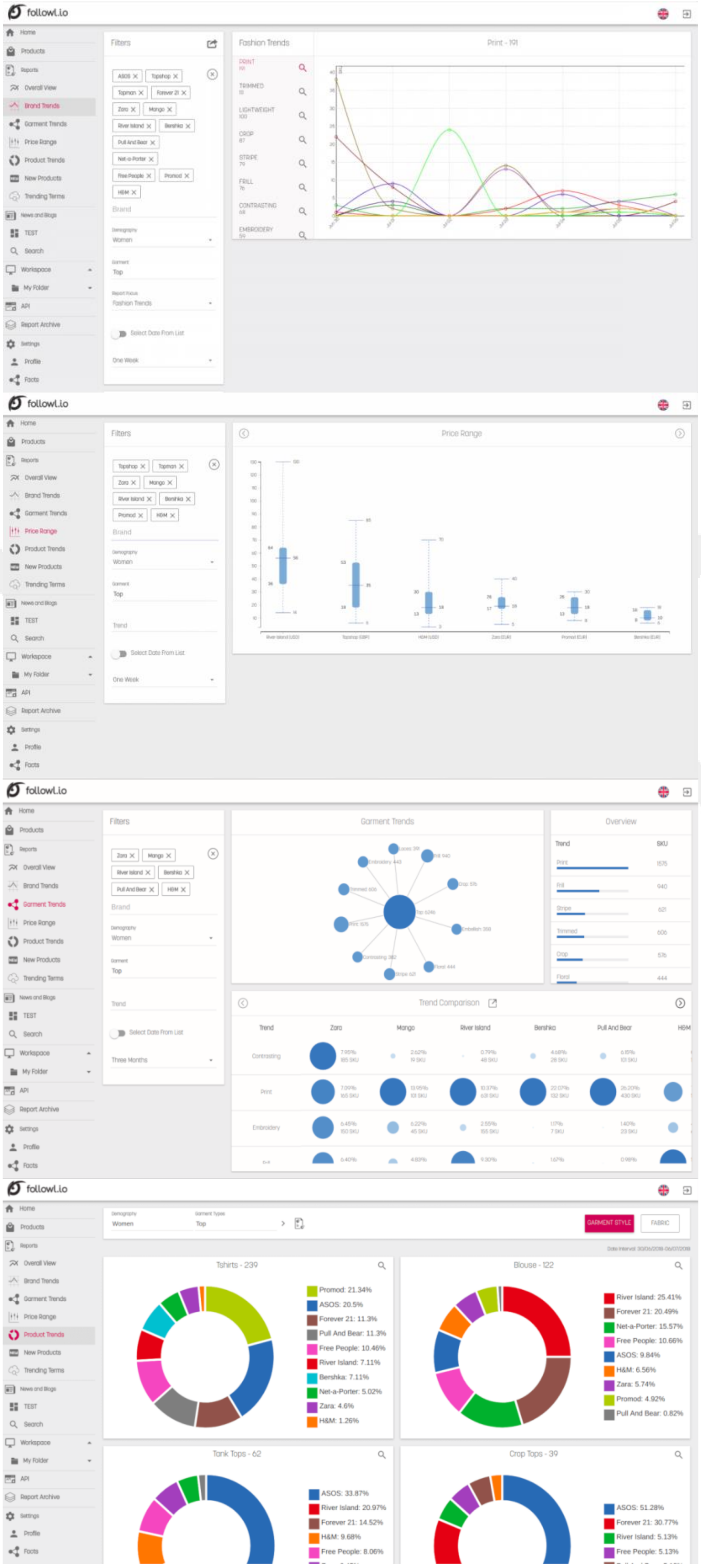


Figure 1.4. Followl.io Fashion Reports

1.3. Fashion Concept Life Cycle

In order to compose a life cycle strategy for marketing, there needs to be an understanding of the shape of cycling of fashion trends and how the marketing mix varies in each stage. Product life cycle (PLC) is a measurement of customer’s demanding amount with presenting how each change of needs is followed within stages (Soltani, 2012). A similar logic can be applied to compose a life cycle for fashion concepts with using the stock keeping unit (SKU) of the textile products with the specified fashion concept for measurement and following its changes for certain time frames.

Stock Keeping Unit (SKU) is a term that is widely used in the retail business. By definition, an SKU is a number assigned to a product by a retail store to identify the price, product options and manufacturer of the merchandise. An SKU is used in order to track the inventory information in the retail store. It is a very valuable statistical indicator to help maintain a profitable retail business.

Based on timeframes, a fashion life cycle (FLC) can simply be analyzed in two ways; long-run that can span up to decades and even centuries and short-run that can span up to several months and years (Sproles, 1981). In the scope of this thesis study, short-run FLC is composed with a strategy that is in line with the requirements of the Followl.io textile database and the research problem.

In Figure 1.5, FLC for this study is demonstrated. Like all life cycles, it contains the necessary parts of birth, life, and death. The life cycle begins for a specified fashion concept with the proposed stage. A fashion concept passes through this stage by increasing its SKU from zero. These fashion concepts can be either new concepts that are not seen in the system before or returning concepts that are seen in the past and re-entering the FLC again. In the fashion industry, the proposal stage is often used as a trial stage for fashion concepts. Textile brands propose the fashion concept and test the market before deciding on investing even more in it. This state usually lasts about a month and it is in either the right before or at the early stages of the fashion season. After the proposal state, depending on the SKU change, the state passes through to either emerging state with increasing even more or to demode state with an SKU that is decreasing or staying the same. This state indicates the early stages of the fashion season and similar to the proposal stage, it lasts about a month. Emerging state is the most crucial part of the FLC since it is the state that textile brands decide to invest in the concept or not. Therefore, in this stage the answer to the question "If the fashion concept is going to be trend or demode?" becomes very important. Because of that, in this study, this state is highly used for the fashion trend prediction applications. FLC ends its cycle with passing through either to trend state with showing an increase in SKU over the period of following 2-3 months or to demode state without showing any increase or decrease in SKU and completing its cycle.

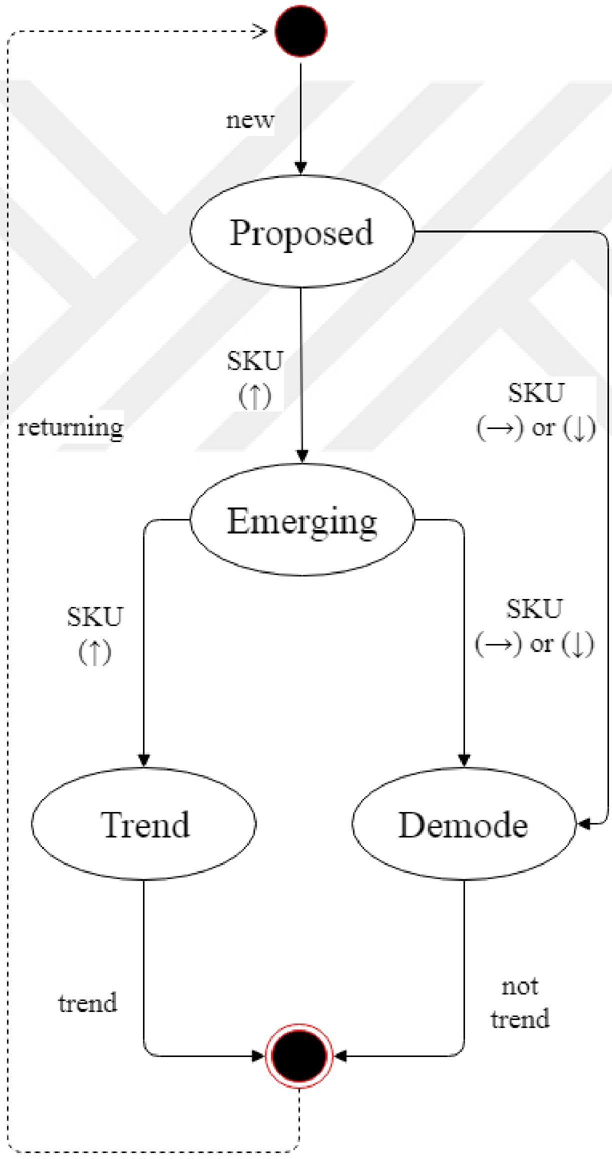


Figure 1.5. Fashion Concept Life Cycle (FCL)

1.4. Objectives and Problem Statement

Textile brands require the information of the likelihood of a newly proposed fashion concept becoming a trend and they require this as early as the fashion concept is still in the emerging state of the fashion life cycle in Figure 1.5. In order to find a solution to this problem, the objective of this thesis study is to find the machine learning techniques with the best performance for predicting the emerging fashion trend concepts by using the textile database provided by Followl.io. This study has a secondary aim to analyze and evaluate this provided textile database for the compatibility of machine learning applications and collecting information to improve the database in ways that make the database more compatible for machine learning applications thus, making more accurate predictions.

1.5. Data Limitations

Followl.io despite being a new application, has a textile database extracted over 3 million of textile product data over around 2 years' time period. The application also has a textile ontology dictionary used for tagging the textile products with proper textile concepts. The ontology dictionary contains thousands of textile concepts within a variety of sub-ontology dictionaries such as textile brands, garments, fabrics, colors, and demography.

Followl.io first started collecting data for the purpose of monitoring and reporting the textile information, its database was not built for the intention to be used by machine learning models. For that reason, the applicability and performance of these machine learning models were limited. Therefore, through the course of this thesis study, database and ontology dictionary of the application are revised and improved to make it more suitable to be used by machine learning models.

1.6. Outline

In the following chapters, this thesis provides previous and related works in regards to the topic of this study in the second chapter. In the third chapter, the used textile database and its properties are deeply analyzed for machine learning applications. After this analysis, features that are extracted to be used for machine learning and the used feature extraction strategies are discussed in the fourth chapter. After that, in the fifth chapter, machine learning strategies and their suitability to the nature of the problem discussed in this thesis study and the provided textile database are discussed and performance analysis results are presented. Finally, the conclusion and the future work are presented in the sixth chapter.

CHAPTER 2
ACADEMIC REVIEWS AND RELATED WORKS

Research studies that are related to the prediction with machine learning are summarized in detail in this chapter. Along with the studies that are conducted in the area of fashion trend prediction using machine learning techniques, the studies that are conducted in the field of stock market index movement predictions by using machine learning techniques are also explicitly focused on due to problem similarities. It should be noted that the number of studies about stock market index movement is a lot more than the researches about fashion trend prediction.

2.1. Fashion Trend Predictions

Most of the research about fashion prediction focus on the fashion sales prediction and the topic of fashion trend prediction focus on the fashion color prediction. Although fashion colors can also be predicted using the machine learning features prepared for this research, the main aim of this study is to focus on predicting the fashion trend concepts.

In 2010, Paola Mello from University of Bologna, Italy and Sergio Storari from University of Ferrara, Italy published an article named “Application of Machine Learning Techniques for The Forecasting of Fashion Trends” in the Artificial Intelligence journal. Their research focuses on showing that it is both possible and useful to apply machine learning techniques for forecasting of fashion trends. The research describes a prototype of a knowledge-based system that can automatically generate trend predictions in the fashion industry. In the scope of the research, two machine learning algorithms are used, the Bayesian Networks and Decision Trees. The article describes how these machine learning models are generated from using the predictions made in the past years and how to evaluate their performance.

Recently, another master thesis was published in 2017 by Mona Dadoun from KTH Royal Institute of Technology, Sweden with a topic very similar to this study named “Predicting fashion using machine learning techniques”. In the study, the question of “To what extent it is possible to predict fashion by applying machine learning?” is investigated by using the data of the Apprl web application. Similar machine learning features were extracted: category, brand, gender, vendor, publisher, and color. Using these features three types of predictions have been made: the number of sales, the number of clicks and the ratio of popularity. The performance evaluations are made by performing K-fold Cross Validation and by using confusion matrix metrics such as accuracy, precision, and recall as performance evaluation metrics. Among machine learning methods, Decision Tree, Extremely Randomized Tree, Random Forest, AdaBoost, K-Nearest Neighbors, Logistic Regression and Stochastic Gradient Descent are considered.

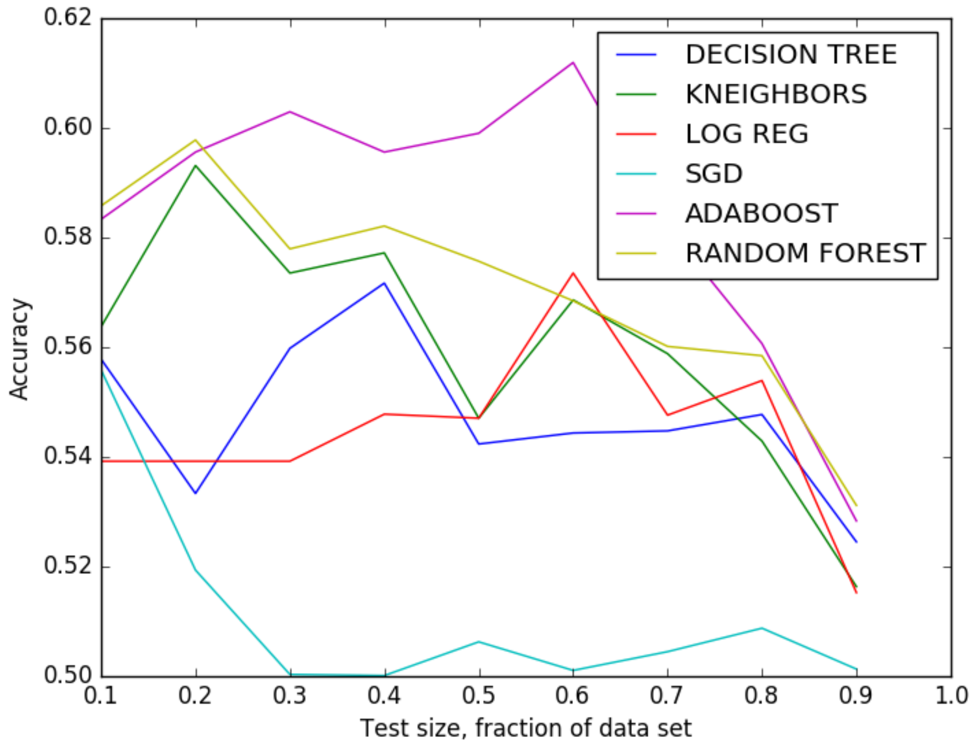


Figure 2.1. Average Accuracy by the Classifiers *(Dadoun, 2017)*

As a result of the performance evaluations shown in Figure 2.1, the research showed that for the amount of sales prediction, the datasets that were used were too noisy and not big enough for the prediction. On the other hand, for the number of clicks and ratio of popularity, the dataset was less noisy and large enough for the predictions. The algorithm performances for these predictions in the study showed that linear models were not suitable. Also, K-Nearest Neighbors and AdaBoost models were the best performing models with the best prediction results achieved in predicting the ratio of popularity with an average accuracy between 65% and 70%. For future work, the research suggests working with bigger datasets for predicting the number of sales and using Artificial Neural Networks and Deep Learning models might improve the prediction performance. Furthermore, the author also suggests that using the regression approach instead of the classification approach for the prediction for the number of clicks might improve the performance.

2.2. Stock Market Index Movement Prediction

In 2014, Jigar Patel and Sahil Shah from Nirma University, India published an article about market trend prediction named "Predicting stock and stock price index movement using Trend Deterministic Data Preparation and machine learning techniques" in Expert Systems with Applications journal. The research mainly addresses the problem of predicting the direction of movement of stock (increase and decrease) and stock price index using data from 10 years (2003-2012) of Indian stock market while focusing on two stocks: Reliance Industries and Infosys Ltd. and two stock price indices: CNX Nifty and S&P Bombay Stock Exchange (BSE) Sensex.

The research focuses on comparing the performance of four prediction models, Naïve Bayes, Support Vector Machines (SVM), Artificial Neural Networks (ANN) and Random Forest. The research uses these algorithms with two approaches. The first approach is using the computation of 10 technical stock trading parameters such as opening, high, low and closing prices and the second approach is representing these technical parameters as trend deterministic data. For both approaches prediction model accuracies were evaluated.

The prediction models in their research used moving average and momentum as the technical parameters or indicators which are also used in this study. Two types of moving averages are used for the period of 10 days: Simple Moving Average (SMA) and Weighted Moving Average (WMA) for extracting an indicator for the short-term future and suggested that the trend is up if the price is above the moving average and similarly, the trend is down if the price is below the moving average. The momentum indicator is used to measure the rate of the rising and the falling of the stock prices and the positive value of the momentum indicates the rising trend which is represented by ‘+1’ and the negative value of the momentum indicates the falling trend which is similarly represented by ‘-1’. Put in a nutshell, the trend deterministic data is prepared by taking advantage of the fact that each of the technical indicators has its own inherit opinion about the direction of the stock price movement.

The result of these experiments reveals that for the first approach Random Forest algorithm outperforms the Naïve Bayes, SVM and ANN algorithms on overall performance where technical parameters are represented as continuous values. The research results also suggest that the overall performance of all the algorithms that are studied on improves when these technical parameters are represented as trend deterministic data.

Another article about market trend prediction named “Predicting direction of stock price index movement using artificial neural networks and support vector machines: The sample of the Istanbul Stock Exchange” in Expert Systems with Applications journal was published in 2010 by Yakup Kara form Selçuk University, Turkey. The research focuses on the prediction of the stock price index movement using financial time series. The study attempts to develop two efficient models that are based on the classification techniques, Artificial Neural Networks (ANN) and Support Vector Machines (SVM). The performance of these two classification techniques is compared for predicting the direction of movement in the 10 years of (1997-2007) daily Istanbul Stock Exchange (ISE) National 100 Index. Similar to the previous research, this research also uses ten technical indicators for the proposed ANN and SVM models.

The Artificial Neural Network model that was used in their research uses a simple three-layered feedforward model that represented in Figure 2.2 in order to predict the stock price index movements. This ANN model consists of three layers, an input layer, a hidden layer and an output layer. The ten technical indicators are represented by ten nodes in the input layer and the one output that can have only two value (0 and 1) is represented by the single node on the output layer.

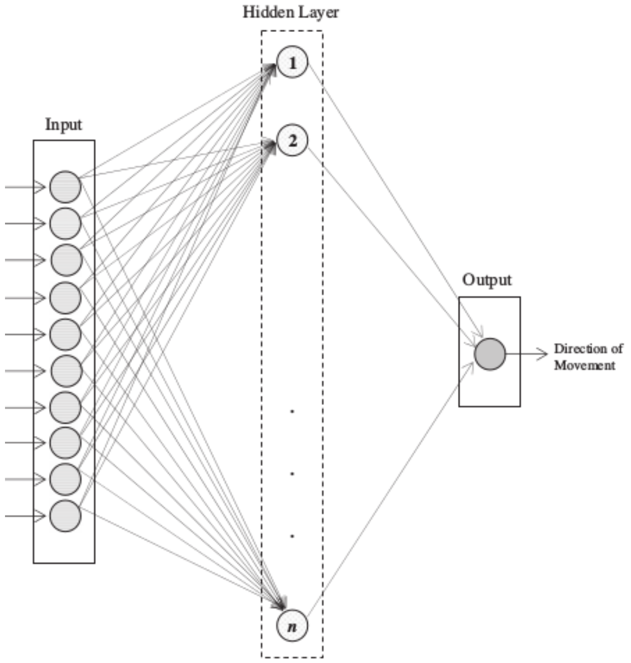


Figure 2.2. Three Layered Feedforward Artificial Neural Network Model *(Kara, Acar Boyacıoğlu, & Baykan, 2011)*

The Support Vector Machine model in their research was prepared for a binary classification model that shown in Figure 2.3 and as an SVM model, it constructs a hyperplane in order to be used as the decision surface such that the margin of separation between positive and negative examples is as wide as possible. This allows the model to have a small probability of misclassifying a future sample. SVM constructs an Optimal Separating Hyperplane (OSH) by mapping all the input vectors into a high dimensional feature space. This results in maximizing the margin, the distance between the hyperplane and the nearest data points of each class in the space H.

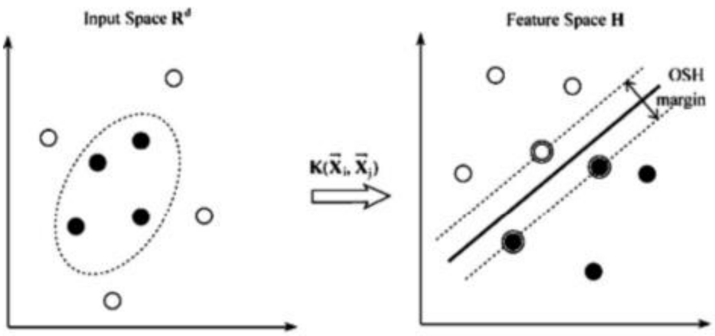

Input Space $\mathbf{R}^d$
Feature Space H
$\mathbf{K}(\overline{\mathbf{X}}_i, \overline{\mathbf{X}}_j)$
OSH
margin

Figure 2.3. Binary Classification Support Vector Machine Model *(Kara, Acar Boyacıoğlu, & Baykan, 2011)*

As the previous study by Patel and Shah, ANN, and SVM models that are used this research also use moving average and momentum technical indicators and again uses both Simple Moving Average (SMA) and Weighted Moving Average (WMA) with the time period of 10 days.

The experimental results of the research show that both ANN and SVM models showed significant performance in predicting the direction of stock price movement. Therefore, the study suggested that both of the models are useful for this prediction. Although, 75.74% accuracy of the ANN model was accepted to be significantly better than the accuracy of the SVM model which is 71.52%. Moreover, the research also suggested that the performance of the models can be improved in two ways. The first one is adjusting the parameters in the model in a more sensitive and comprehensive way, and the second one is using different parameters instead of the parameters that are used in the models or using additional parameters along with the parameters that are already used in the models.

Another article related to the topic was also published in the same journal named as “Evaluating Multiple Classifiers for Stock Price Direction Prediction” at 2015 by Michel Balling from The University of Tennessee, USA and Dirk Van den Poel from Ghent University, Belgium. This article especially focuses on benchmarking the ensemble methods such as Random Forest, AdaBoost, and Kernel Factory for predicting stock price direction against single classifier methods such as Logistic Regression, K-Nearest Neighbors, Neural Networks and Support Vector Machines. The data used in the research was constructed from 5767 publicly listed European companies and the area under the receiver operating characteristic curve (AUC) with the formula represented in Figure 2.4 is used for the algorithm performance evaluation comparisons.

$$AUC = \int_0^1 \frac{TP}{(TP+FN)} d\frac{FP}{(FP+TN)} = \int_0^1 \frac{TP}{P} d\frac{FP}{N}$$

Figure 2.4. The Area under the Receiver Operating Characteristic Curve Formula
(Kara, Acar Boyacıoğlu, & Baykan, 2011)

Ensemble methods can solve statistical, representational and computational problems in nature by averaging the models. Using specific combinations of the data, representations, objective functions and optimization methods, many different classifiers can be trained and the set of all these possible classifiers that might be trained are called the hypothesis space (H). In order the identify the best possible hypothesis, the algorithm searches H. In addition to that, identifying the best possible hypothesis method changes according to the problem types mentioned before.

The result of this research indicates that by comparing with AUC, ensemble methods as a group with the only exception being the Random Forest did not perform better than single classifier methods. After evaluating individually, Random Forest performs the best result among all classifiers and rest of the algorithms in the scope of this research followed in order by Support Vector Machines (SVM), Kernel Factory, AdaBoost, Neural Networks (NN), K-Nearest Neighbors (k-NN) and Logistic Regression.

In the light of these researches about stock market index movement prediction, some deductions can be made. First, in almost all of the researches mentioned before the ensemble method, Random Forest provides the best performance evaluation results compare to the other machine learning algorithms. Second, the technical indicators namely, the moving average (both simple moving average and weighted moving average) and momentum are commonly used in the stock market index movement prediction problems. This two deduction strongly suggest that since stock market index movement prediction and fashion trend prediction problems are considered similar to each, these same two cases might be valid for fashion trend prediction problem as well.

CHAPTER 3
DATABASE ANALYSIS

In order to predict the fashion trends by using machine learning techniques, a textile database was required. The textile database is obtained from Followl.io. In this chapter, the structure and properties of this obtained database are deeply analyzed and their applicability for machine learning models are discussed in detail.

3.1. Database Structures

3.1.1. Elasticsearch

The Textile database of the Followl.io web application is stored in the Elasticsearch search engine. Due to the rapidly increasing amount of data in the world, using a high-performance search engine becomes more important (Thacker, Pandey, & Rautaray, 2018). Elasticsearch is a real-time distributed Java-based analytics and search engine. It allows exploring the data with high speed and performance. It provides full-text search, structured search, analytics and their combinations with an HTTP (Hypertext Transfer Protocol) web interface and schema-free JSON (JavaScript Object Notation) documents (Gormley & Tong, 2015).

Figure 3.1. Elasticsearch Logo

Elasticsearch is one of the most popular database management systems (DBMS). It ranks 8th in overall DBMS as shown in Figure 3.2 and 1st in search engines according to the DB-Engines Ranking. The DB-Engines Ranking is a monthly updated DBMS popularity ranking (DB-Engines Ranking, 2018).

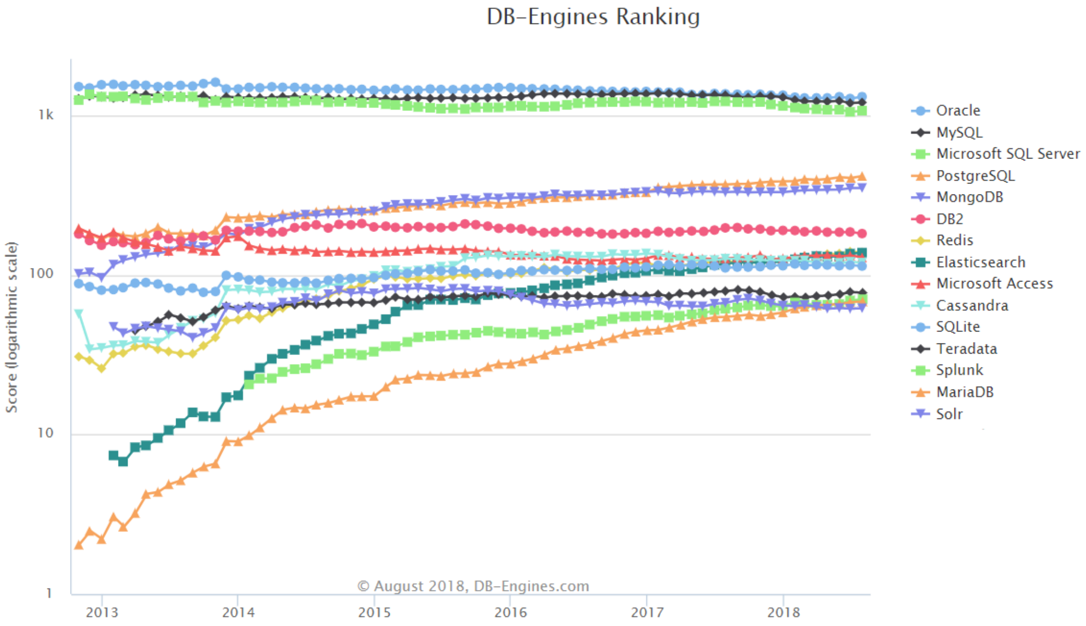


Figure 3.2. DB-Engines Ranking *(DB-Engines Ranking, 2018)*

3.1.2. Ontology

Followl.io web application uses a variety of ontology dictionaries such as textile brands, garments, fabrics, colors, and demography to semantically tag real-world textile concepts. Ontology is the study of what things exist and while building a list of all things that exist, the ontologist is not interested in the existence of old things (Effingham, 2013). In computer and information sciences, an ontology defines a set of representational primitives such as classes, properties or relations with which to model a knowledge domain. These representational primitives also include information about their meaning and constraints on their logically consistent application within their definition (Gruber, 2008).

Ontology dictionaries are stored as Resource Description Framework models. The Resource Description Framework (RDF) is the World Wide Web Consortium (W3C) recommendation in the Semantic Web for semantic annotations. Similar to conceptual modeling approaches like class and entity-relationship diagrams, the idea is to make statements about resources in the form of triples that contain a subject, a predicate and an object (Pan, 2009). In Followl.io, subject information of the each semantically tagged triple of the textile product is stored in the Elasticsearch database.

Figure 3.3. Resource Description Framework (RDF) Logo

3.2. Database Properties

The textile data is stored in a custom mapping which is constructed to be compatible with the textile data within the modern fashion industry. It holds general textile product information such as name, description, brand, color and extraction date. Each property could affect the fashion trend and could be used to predict the fashion trend in some shape or form. In the following properties part, the textile product data properties and their individual applicability for extracting machine learning features and constructing training and testing sets are discussed.

3.2.3. Elasticsearch Tags

The resource information of the ontology triples is stored in the textile products index of the Elasticsearch database within the Elasticsearch resource tags (estags) property. These stored resource tags are stored for the purpose of querying the detailed ontology information using the RDF query language SPARQL (Simple Protocol and RDF Query Language).

There are multiple specialized ontology dictionaries present within the Follow.io web application for the purpose of describing the real world textile concepts better. After the textile product is extracted from a textile brand website, Follow.io tags all of the content with the resources that match to the resources of these ontology dictionaries. These ontologies use triple notions in a way to build a hierarchical and relational model for textile metadata to fully organize the textile concepts. After analyzing all ontology dictionaries, garment type, garment style, fashion trend, fashion style, color, demography, and textile brand ontology dictionaries are determined to contain helpful information regards to extracting machine learning features for predicting fashion trend concepts.

3.2.3.1. Garment Ontology

Garment type and garment style ontology dictionaries, despite being two different ontology dictionaries, share relational triples with each other. Garment type ontology contains collective garment metadata. It contains resource values such as "Top", "Bottom", "Accessory", "Sportswear" and "Swimwear". Likewise, their sub triples resource values like "T-Shirt", "Shirt", "Bikini", "Jacket", "Hat", "Bag", "Trousers", "Dress" and "Skirt" are also included in garment type ontology. On the other side, the garment style ontology dictionary contains more specialized garment triples. These triples are usually related to the garment type triples with a parent-child relationship that is created using broader triples. Garment style ontology resources contain values such as "Mini Skirt", "Cocktail Dress", "Messenger Bag", "Baseball Hat" and "Capri Trousers". A detailed example of top to bottom relational garment ontology model structure of "Bottom", "Skirt" and "Mini Skirt" is shown in Table 3.1, Table 3.2 and Table 3.3. The trending fashion concepts are likely to be seen in multiple different numbers of garments. Therefore, it can be assumed that the number of different garment concepts that the fashion concept has can affect the fashion trend condition.

Subject	Predicate	Object
<http://data.follow.io/resource/Bottom>	<http://www.w3.org/2004/02/skos/core#prefLabel>	“Bottom”@en
<http://data.follow.io/resource/Bottom>	<http://www.w3.org/2000/01/rdf-schema#label>	“Bottom”@en
<http://data.follow.io/resource/Bottom>	<http://www.w3.org/2000/01/rdf-schema#label>	“Alt Giyim”@tr
<http://data.follow.io/resource/Bottom>	<http://ontoloji.galaksiya.com/vocab/belongsTo>	<http://data.galaksiya.com/tagbase/garment_type>

Table 3.1. Bottom Triples

Subject	Predicate	Object
<http://data.follow.io/resource/Skirt>	<http://www.w3.org/2004/02/skos/core#prefLabel>	“Skirt”@en
<http://data.follow.io/resource/Skirt>	<http://www.w3.org/2000/01/rdf-schema#label>	“Skirt”@en
<http://data.follow.io/resource/Skirt>	<http://www.w3.org/2000/01/rdf-schema#label>	<http://data.galaksiya.com/tagbase/garment_type>
<http://data.follow.io/resource/Skirt>	<http://ontoloji.galaksiya.com/vocab/belongsTo>	“Etek”@tr
<http://data.follow.io/resource/Skirt>	<http://www.w3.org/2004/02/skos/core#broader>	<http://data.follow.io/resource/Bottom>

Table 3.2. Skirt Triples

Subject	Predicate	Object
<http://data.follow.io/resource/Mini_Skirt>	<http://www.w3.org/2004/02/skos/core#prefLabel>	“Mini Skirt”@en
<http://data.follow.io/resource/Mini_Skirt>	<http://www.w3.org/2000/01/rdf-schema#label>	“Mini Skirt”@en
<http://data.follow.io/resource/Mini_Skirt>	<http://www.w3.org/2000/01/rdf-schema#label>	<http://data.galaksiya.com/tagbase/garment_style>
<http://data.follow.io/resource/Mini_Skirt>	<http://ontoloji.galaksiya.com/vocab/belongsTo>	“Mini Etek”@tr
<http://data.follow.io/resource/Mini_Skirt>	<http://www.w3.org/2004/02/skos/core#broader>	<http://data.follow.io/resource/Skirt>

Table 3.3. Mini Skirt Triples

3.2.3.2. Fashion Trend Ontology

Fashion trend ontology contains a variety of trending fashion terms that are extracted from real-world fashion concepts. The trend is a term that is used when defining something popular in a certain time. Similarly, any popular fashion term in a certain time such as printed, dyed, patterned and decorated textile products in a specific way is a fashion trend. In line with this statement, it can be assumed that the density of a specific fashion term in a certain time can be an indicator to predict the fashion trend.

Fashion trend resources in fashion trend ontology tend to have a broader relationship with each other within the fashion trend ontology as well as with garment type and garment style ontologies. An example of a top to bottom relational fashion ontology triples of “Print”, “Animal Print” and “Monkey Print” are given in Table 3.4, Table 3.5 and Table 3.6.

Subject	Predicate	Object
<http://data.follow.io/resource/Print>	<http://www.w3.org/2004/02/skos/core#prefLabel>	“Print”@en
<http://data.follow.io/resource/Print>	<http://www.w3.org/2000/01/rdf-schema#label>	“Print”@en
<http://data.follow.io/resource/Print>	<http://www.w3.org/2000/01/rdf-schema#label>	“Baskı”@tr
<http://data.follow.io/resource/Print>	<http://ontoloji.galaksiya.com/vocab/belongsTo>	<http://data.galaksiya.com/tagbase/fashion_trend>

Table 3.4. Print Triples

Subject	Predicate	Object
<http://data.follow.io/resource/Animal_Print>	<http://www.w3.org/2004/02/skos/core#prefLabel>	“Animal Print”@en
<http://data.follow.io/resource/Animal_Print>	<http://www.w3.org/2000/01/rdf-schema#label>	“Animal Print”@en
<http://data.follow.io/resource/Animal_Print>	<http://www.w3.org/2000/01/rdf-schema#label>	“Hayvan Baskı”@tr
<http://data.follow.io/resource/Animal_Print>	<http://ontoloji.galaksiya.com/vocab/belongsTo>	<http://data.galaksiya.com/tagbase/fashion_trend>
<http://data.follow.io/resource/Animal_Print>	<http://www.w3.org/2004/02/skos/core#broader>	<http://data.follow.io/resource/Print>

Table 3.5. Animal Print Triples

Subject	Predicate	Object
<http://data.follow.io/resource/Monkey_Print>	<http://www.w3.org/2004/02/skos/core#prefLabel>	“Monkey Print”@en
<http://data.follow.io/resource/Monkey_Print>	<http://www.w3.org/2000/01/rdf-schema#label>	“Monkey Print”@en
<http://data.follow.io/resource/Monkey_Print>	<http://www.w3.org/2000/01/rdf-schema#label>	“Maymun Baskı”@tr
<http://data.follow.io/resource/Monkey_Print>	<http://ontoloji.galaksiya.com/vocab/belongsTo>	<http://data.galaksiya.com/tagbase/fashion_trend>
<http://data.follow.io/resource/Monkey_Print>	<http://www.w3.org/2004/02/skos/core#broader>	<http://data.follow.io/resource/Animal_Print>

Table 3.6. Monkey Print Triples

3.2.3.3. Demography Ontology

Demography is a term that is used when defining the study of a human population’s behavior (Preston, Heuveline, & Guillot, 2000). The demography ontology dictionary contains the collective information that combines gender specifications and age group classifications together. Its triples contain resources such as “Woman”, “Man”, “Unisex”, “Kid” and “Baby”. Much like garment type ontology dictionary, their sub triples such as “Boy” and “Girl” for “Kid” are also in the scope of this ontology dictionary. Example fashion ontology triples of “Woman” are given in Table 3.7.

Trend fashion concepts tend to be seen in a variety of different genders and age groups. Some of these groups such as “Woman” and “Unisex” can be observed in the provided textile database to have a much higher impact on affecting the likelihood of becoming a fashion trend than the others. Therefore, the assumption of the number of the different demographics of which the fashion concept is containing can affect the fashion trend condition can be made.

Subject	Predicate	Object
<http://data.follow.io/resource/Woman>	<http://www.w3.org/2004/02/skos/core#prefLabel>	“Woman”@en
<http://data.follow.io/resource/Woman>	<http://www.w3.org/2000/01/rdf-schema#label>	“Woman”@en
<http://data.follow.io/resource/Woman>	<http://www.w3.org/2000/01/rdf-schema#label>	“Kadın”@tr
<http://data.follow.io/resource/Woman>	<http://ontoloji.galaksiya.com/vocab/belongsTo>	<http://data.galaksiya.com/tagbase/demography>

Table 3.7. Woman Triples

3.2.3.4. Brand Ontology

Textile Brand ontology dictionary contains textile brand names that are tracked by Follow.io web application. The textile brand has an undeniable effect on what is the trend in the fashion market. For the scope of the thesis study, Zara, Mango, H&M, Pull&Bear and Stradivarius textile brands are selected due to their huge influence on the fashion market, their high frequencies in the Follow.io textile database and considering the requirements of the company’ customers.

3.2.3.5. Color Ontology

Color ontology dictionary contains textile product color names. A textile product can have multiple colors and each resource of these colors are stored in the “estags” property. However, a textile product can only have one descriptive color assigned by the manufacturer of this textile product and this color value is stored in a separate “color” property as a string value. This color information is also stored in the “estags” property like a color resource. Both of these color information is considered to be used in machine learning feature extraction phase but neither of them is decided to be used. The reason is that in spite of the effect of the color information on the prediction of the fashion trend is unreliable. After consulting with the fashion professionals, the effect of the color information is decided to be not significant enough to be taken into account. However, it is also decided that with a different approach to textile product color information, it could be possible to produce a machine learning feature. Furthermore, a different study could be pursued in the future by using color information to build a machine learning model to predict fashion color trends.

3.2.4. Product Extraction Date

Timestamp information for the first time the textile product is extracted from the website is stored in the “datePublished” property. Followl.io web application constantly extracts data from textile brand websites and stores them with their extracted date. One of the most important features of this particular property is that it enables the data in the database to be represented as time series data. Time series data is the time-ordered sequence of observations (Bontempi, 2013). Using time series provides an opportunity to work with time series specific machine learning features.

3.2.5. Stock Status

Stock status information of the textile product is represented by a Boolean value “inStock” property. Textile product is in stock when its value is true and out of stock when it is false. The “inStock” property is very suitable for extracting machine learning feature despite that, the property holds Boolean information rather than numeric stock availability information. This might be the case, however, after examining the property in detail throughout the whole instances, a considerable amount of the instances has shown uncertain behavior. In addition to these, some of the textile brands put their products on the market with out of stock status and this behavior contradicts with machine learning feature expected to be obtained from the “inStock“ property. For these reasons, the “inStock” property is decided to not be included in the feature extraction stage of this study.

3.2.6. Discount Status

Discount status information indicates if the product is currently on sale or not. It is represented by the Boolean "discount" property. Straightforwardly, the property value equals true when the specified textile product is currently on sale and false when it is not currently on sale. The "discount" property useful for obtaining a machine learning feature with the following idea that, a textile product should not be considered a trend if it is currently on sale. However, much like the "inStock" property, the "discount" property showed uncertain behavior. Some of the textile brands often put their products on the market with on sale status and this behavior as well contradicts the machine learning feature that is expected to be obtained from the "discount" property. Due to this reason, the "discount" property is also decided not to be included in the feature extraction stage of the study.

3.2.7. Servicing Country

Most of the leading textile brands operate in multiple countries. This case is represented in the provided textile database with the "areaServed" property. The property simply holds the detailed information of the same textile product for different countries as an array of "areaServed" values. For example, Euro (€) is used in Spain and Dollar ($) is used in the USA in data. Another example is the shoe size information, shoe size is 7 for the United Kingdom and 40 for Turkey.

Each object has multiple properties and "addressCountry", "inStock", "discount", "price", "priceCurrency" and "size" properties are the ones to be considered to have an effect on the fashion trend prediction and used for extracting machine learning features.

Textile product "addressCountry" property value of the "areaServed" property holds a unique country code value so that each "areaServed" object represents the detailed information of a specific country. The country code value represents the code of the country that textile products are currently available on the market. As an example, some of the textile "addressCountry" properties are shown below in Table 3.8 with country codes and their respective country pairs.

addressCountry	Country
UK	United Kingdom
US	United States
DE	Germany
ES	Spain
RU	Russia
TR	Turkey

Table 3.8. Country Code Table

Due to the requirements of the study, the “addressCountry” property is considered to be only taken as “TR” to predict fashion trends specifically for Turkey. However, the “addressCountry” property is taken into account when analyzing the effect of each textile brands’ fashion trend prediction. In addition, this property holds an utter importance along with the whole “areaServed” property for the first step of the following study when it comes to trying to predict fashion trends not specifically for Turkey.

Textile product size information for that specific country is represented by the “size” property. This property holds a string value that differentiates according to the type of the fashion product. For example, a t-shirt garment type commonly has size values such as extra small (XS), small (S), medium (M), large (L) and extra-large (XL). On the other hand, shoe garment type sizes take values like 36, 37, 40 and 43. Another different size value can be encountered in the kid and the baby textile products. The size value of these types of textile products can be a representation of a certain age group or body size groups like body weight or height. An example could be given for these type of “size” properties as “Boy 3-6 Age”, “50-70 cm“, and “3-5 kilos”. The string “size” property is not suitable for extracting machine learning feature alone. Moreover, the property value is hard to cluster into different size groups for the reasons that single property tries to represent a property of multiple textile types and also these size values can show differences when referencing from different textile brands. In consideration of these, the size value is decided not to be used for extracting machine learning feature.

The “size” property is currently being stored using a different strategy. Within this new strategy, “size” property is no longer stored as a string but an object with its own properties. This size object holds the specific information about that size like stock, discount, and price. At the moment, stored textile products within Elasticsearch with this type of “size” property does not have the required amount of database instances. With using this property and store strategy fully functional as a future work could make it possible to build a machine learning model that predicts if the textile product is a trend for each of its different sizes.

The price and price currency information is represented in the price and the priceCurrency properties, respectively. Price currency information provides an additional information to the price information and for that reason, they can be considered as one property. This combined price information does not have any effect on the prediction of the fashion trend. The reason is that both high and low priced textile products have an equal chance of being a fashion trend and when it comes to learning a fashion trend behavior from different textile products, it is not wise to compare different type of textile product within the same price property. Due to this reason, price information is not included in the machine learning feature extraction stage.

CHAPTER 4
FEATURE SELECTION

In machine learning, a feature or a technical indicator is defined as an individual measurable property or characteristic of a behavior or an event being observed (Witten, Frank, Hall, & Pal, 2016). Selecting the features that are informative, discriminating and relevant is a major point of an effective machine learning algorithm (Blum & Langley, 1997). Based on this information, the properties of each textile product data provided by Follow.io web application database that is in the emerging fashion concept state is used to extract the machine learning features. The training set for the machine learning algorithms is built from the fashion concept that is in the emerging state of the textile product lifecycle. This chapter investigates the definition and extraction process of each feature in detail.

4.1. Trend

As it will be discussed in more detail in the next chapter, it is stated that the suitable machine learning technique for the problem of this study is supervised binary classification. Supervised machine learning needs a class in order to make a prediction. The class should have a categorical value and can have two possible different values by the definition of the binary classification problems. The trending feature in this study is used as the class of the train set. It takes values according to the state that the fashion concept data in the emerging state will take on the next state. If the state passes through the trend state after the emerging state, the feature takes the nominal value “YES” and otherwise, if it the state passes through the demode state after the emerging state, the feature takes the nominal value “NO”. The following graphs show the example of trend and not trend fashion concepts respectively.

In Figure 4.1, SKU/Timeframe graph of bejeweled fashion concept is given. As seen in the graph, on March bejeweled fashion concept become apparent for the first time with SKU curve slightly increasing, resulting in the fashion concept to pass through the proposed state. At the end of April, the curve increases even more and causes fashion concept to pass through the emerging state. After the emerging state, fashion concept curve increases even more in the period of the following two to three months and pass through the trend state. Therefore, the class of the train instance is taken as “YES” and features of this train instance is extracted through April since it is the time period the bejeweled fashion concept was in the emerging state.

Similarly, in Figure 4.2, SKU/Timeframe graph of faded fashion concept is given. Fashion concept passes through the proposed state in April and the emerging state in May. However, after the emerging state, fashion concept does not show any positive progress for the following two to three months. Consequently, the class of the train instance for faded fashion concept instance is taken as “No” and associated features are extracted from the time period of September since diamonds fashion concept was in the emerging state.

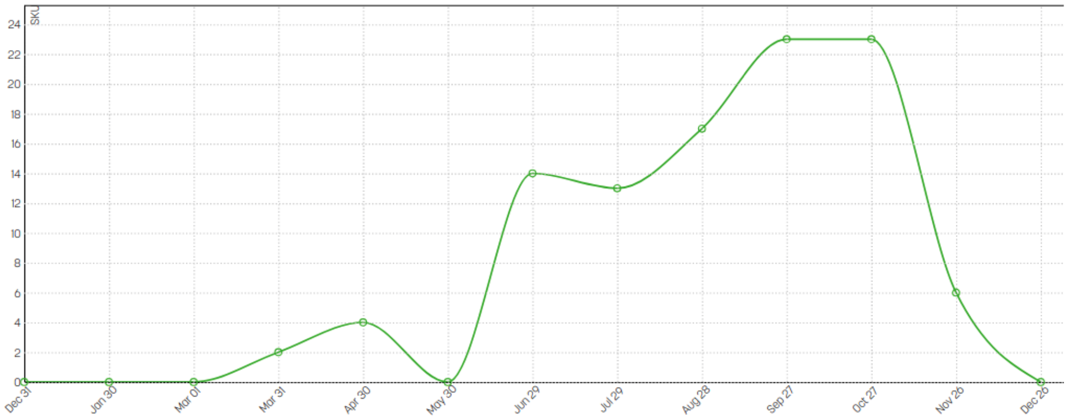


Figure 4.1. Zara Bejeweled Fashion Concept SKU/Timeframe Graph

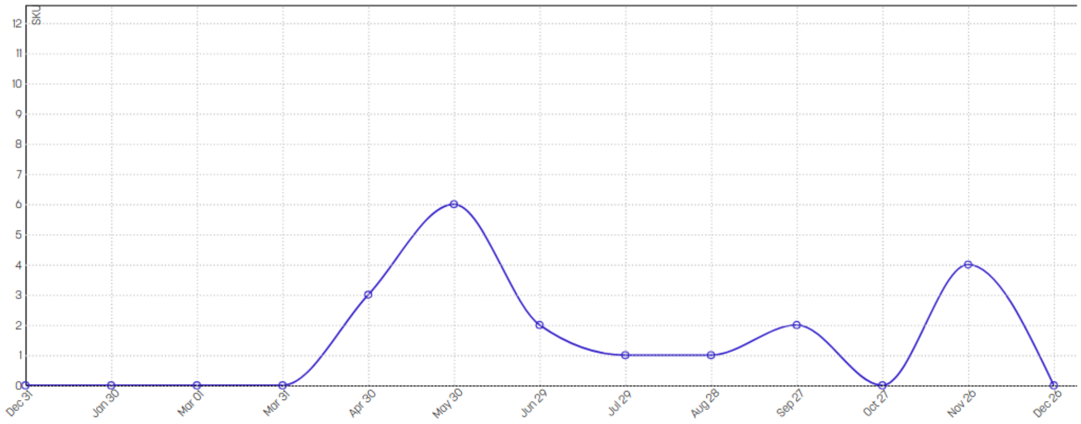


Figure 4.2. Mango Faded Fashion Concept SKU/Timeframe Graph

4.2. Returning

The term returning for the feature represents the fashion concepts that are rejoining to the fashion concept life cycle again after not being seen for a period of time. The feature holds a binary nominal value and if a fashion concept is passing through the proposed state from the returning state, the feature takes the nominal value “YES” and otherwise, the feature takes the nominal value “NO”. This feature helps to distinguish the fashion concepts that are seen again and the fashion concepts that are completely new.

In Figure 4.3, the diamonds fashion concept is seen to pass through the proposal state in August, but the fashion concept is not new to the system as it could be seen that the concept was seen in the system from the beginning of the year through the end of the April. Therefore, diamond fashion concept passes through the proposal state from the returning state and as result, the feature takes the nominal value “YES”.

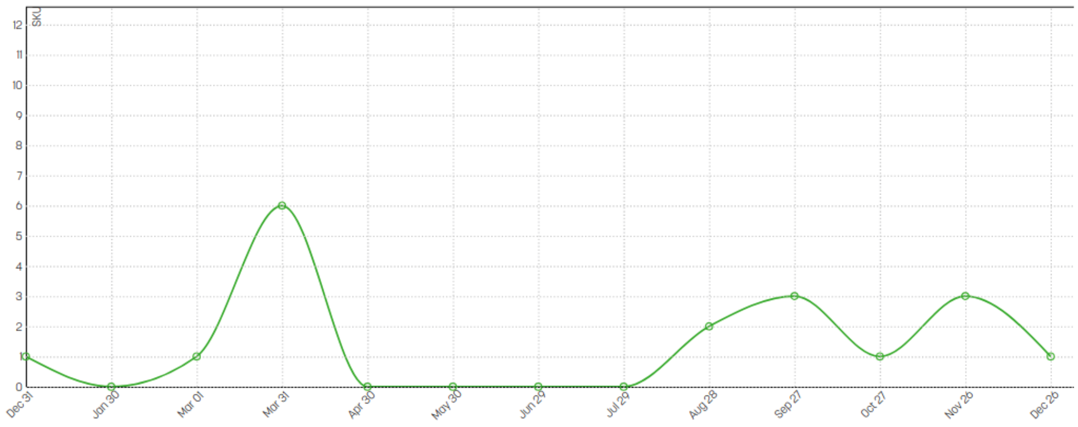


Figure 4.3. Zara Diamonds Fashion Concept SKU/Timeframe Graph

4.3. Tag Frequency Ratio

Tag Frequency Ratio (TFR) is the frequency percentage of the SKU on a specific date and it is considered a feature for the reason that the attributes that are used for calculating the feature are very crucial information regarding predicting the fashion trend. TFR is also used for extracting other features and it is calculated by dividing Stock Keeping Unit (SKU) of the product that contains the specific fashion trend tags over the SKU of all other products extracted in that specific date.

$$TFR_{(t)} = \frac{SKU_{(t)}}{SKU_{\left(total_{(t)}\right)}} \tag{1}$$

The specific date information mentioned above is extracted from the published date (datePublished) property of the textile product data. This property gives the date information of when the specified textile product is extracted and stored in the system. The “datePublished” property is widely used in the machine learning feature extraction of this study for the reason that it is the property that gives the database time series feature.

4.4. Moving Average

The moving average (MA) is a feature that is widely used in the studies of “trend prediction using machine learning” that use databases based on the time series data. It is a simple technical analysis tool that smooths out the specified value of the data by creating a constantly updated value (Zakamulin, 2017). There are different types of moving average approaches, the most widely used ones are the simple moving average (SMA) and weighted moving average (WMA).

The simple moving average (SMA), like any other moving average, is calculated from the average of the past daily values of a property for a given time period. The SMA takes every daily value in the given time period as it is. However, WMA also adds constants to each individual instance in the given time period. These constants are determined by the problem requirements. Because of the requirement of this study is making a prediction based on the past data by taking advantage of provided textile data being a time series using the exponential moving average (EMA) considered to be more suitable (Bontempi, 2013). EMA is a type of WMA in which its constant increases towards the recent instance to improve the importance of the more recent instance.

Tag frequency ratio (TFR) is decided to be the most suitable value that the moving average technical indicators can be extracted from. Since the problem of fashion trend prediction is being aimed to be predicted according to the emerging state of the trend life cycle, the time period of the moving averages is defined as a month and depending on the month, the time period can be either 28, 29, 30 or 31 days.

$$SMA = \frac{\mathrm{TFR}_{(n)} + \mathrm{TFR}_{(n-1)} + \mathrm{TFR}_{(n-2)} + \ldots + \mathrm{TFR}_{(1)}}{n} \qquad (2)$$

$$EMA = \frac{\mathrm{TFR}_{(n)} * n + \mathrm{TFR}_{(n-1)} * (n-1) + \cdots + \mathrm{TFR}_{(1)} * 2 + \mathrm{TFR}_{(1)}}{n} \qquad (3)$$

Most academic studies include both SMA and WMA together in their set of technical indicators for the different characteristics and approaches they use. In this study, it is also beneficial to include both features together.

4.5. Momentum

Momentum feature for this study represents the change of TFR between two following days. The value is not a direct distance value between two values. The value is calculated as below zero when momentum is decreasing and above zero when momentum is increasing. The average of the all these calculated momentum values in the time period of emerging state is taken as an ML feature similar to moving average features. The benefit of including this type of momentum as an ML feature is that it is able to show the day to day changes on the system and we can study its effects on the machine learning model.

$$Momentum = \Delta\left(TFR_{(t)}, TFR_{(t-1)}\right) = TFR_{(t)} - TFR_{(t-1)} \qquad (4)$$

4.6. Textile Brand, Demography, and Garment Type Variances

Variance features are extracted from the Elasticsearch tags (estags) property of the textile product data. It represents the number of different textile brand, demography and garment types on the emerging state of the specified fashion concept. The weighted options for brand and demography variance features are also extracted. The set of these variances, as well as their weights, are formed with the help of the fashion consultants of the Galaksiya considering both their importance for the problem and their database frequencies.

The garment type variety of the database is very high and noisy. For that reason, any sets and weights for the garment type variety are not formed but the feature is included for the performance evaluation experiments. Weighted variance features considered for the study can be seen in Table 4.1 for the textile brand and 4.2 for the demography.

Textile Brand	Weight
Zara	4
H&M	2
Mango	2
Pull&Bear	1
Stradivarius	1

Table 4.1. Textile Brand Weight Table

Demography	Weight
Unisex	4
Woman	4
Man	3
Girl	2
Boy	2
Baby	1

Table 4.2. Demography Weight Table

CHAPTER 5
MODEL SELECTION

The aim of the study is to predict whether fashion concepts are going to be a trend or not. Therefore, the output values are simply categorized as a trend and not a trend. Being able to categorize the output values suggests that this machine learning problem is a classification problem.

In the world of Data Science, there are two types of machine learning tasks. These are supervised learning and unsupervised learning. In the process of supervised learning, the model is expected to learn from the externally supplied instances in order to produce hypotheses and after that make predictions about the future instances (Kotsiantis, 2007). The training dataset instances contain the correct and incorrect answers. The algorithm iteratively makes predictions and later gets corrected by the supervisor until the algorithm achieves an acceptable performance. Whereas in the process of unsupervised learning, only the inputs are given and the outputs are unknown. The model has no supervisor and is expected to learn only from given inputs and make predictions accordingly. The provided database for this classification problem contains training instances that the model can learn from. So it means that the classification problem is a supervised machine learning.

There are multiple supervised classification algorithms which can be used to predict fashion trends. These algorithms can be grouped by similarities of handling the classification problem. These are Bayes', Lazy, Regression and Decision Tree learning types. Despite belonging to the same groups, every classification algorithm approaches the given problem from a slightly different perspective. When compared to each other, each algorithm has advantages, disadvantages, and trade-offs in different aspects. These aspects are mainly concerning the speed and the memory efficiency. Since the textile data can be categorized as a non-stationary time series data, non-linear machine learning techniques have also been used (Patel, Shah, Thakkar, & Kotecha, 2015). The approach of some of the classifiers might be convenient for the trend prediction problem, while others can be less or more convenient in comparison. Therefore, the beneficial thing to do would be to build a classifier model from each well-known classification algorithm of each classification learning types using the same training set with optimized settings and then compare their performance results in order to find the most suitable classification algorithm for predicting the fashion trends.

These algorithm performances are compared with each other in order to decide on the best classification algorithm to predict fashion trends. The algorithm performances are obtained by using a machine learning software called Weka. Weka (Waikato Environment for Knowledge Analysis) is an open source machine learning software developed at the University of Waikato in New Zealand by using Java. Being an open source software and being easy to use due to having a graphical user interface were the main reasons that the software is selected used for the classification algorithm performance analysis and comparisons (Weka 3: Data Mining Software in Java, 2018).

Figure 5.1. Weka Logo

All Machine Learning features that will be mentioned in the related chapter are used in performance evaluations. After evaluating different machine learning algorithms, some performance metrics showed improvement in the accuracy and decrease in the error rates after discarding certain features but including others. Different feature sets are applied on the performance evaluations and feature set that contains exponential moving average (EMA), momentum, weighted brands, weighted demographics and returning features are selected. Tag frequency ratio (TRF) and simple moving average (SMA), as well as brands and demographics features without the weighted calculations, compared to ones with the weighted calculations, showed a decrease in the general performance of the evaluations. Garment parameter contained values with on a big scale and similar to brands and demographics features, excluding it increased the general performance of all algorithm evaluations. In addition to the algorithm evaluations, attribute selection algorithms are applied to the features as well. Both applications showed that the exponential moving average (EMA), the momentum and the returning features have more significance compared to the other machine learning features and excluding them decreased the general performance.

The training set used in algorithm performance comparisons is constructed by hand by observing the trend tags that are in the emerging state from the database the dates between 01.01.2017-30.06.2018 (1.5 years). It contains 165 training instances, 92 of them are specified as trend class and 73 of them are specified as not trend class. 32 of the 92 trends are specified as returning trends. The training set has been built by using the textile product instances that from Man and Woman demographics of Zara, H&M, Mango, Pull&Bear, and Stradivarius textile brands.

A brief information and features of the well-known classification algorithms, the training set and the performance evaluation metrics that used in performance comparisons in the scope of this study will be discussed under the context of the following topics.

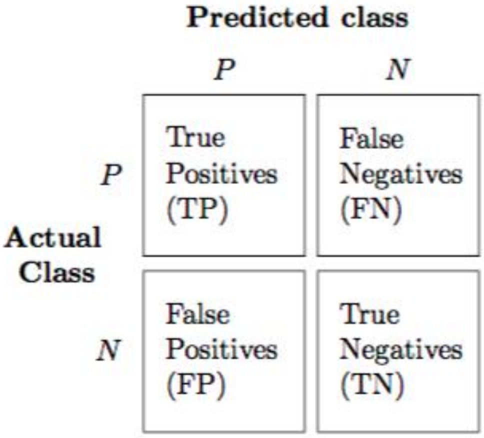
5.1. Evaluation Methods

The best performance evaluation result of each individual classification algorithm is performed by applying the 10 folds cross-validation method and using the same training dataset. The performance values are put through a comparison table to compare each performance attributes.

In the course of comparing the algorithm performances, number of correctly and incorrectly classified instances, accuracy, true positive rate (TPR), true negative rate (TNR) and area under the Receiver Operating Characteristic (ROC) curve (AUC) performance evaluation metrics are taken into consideration.

5.1.1. Confusion Matrix

The number of correctly and incorrectly classified instances, accuracy, true positive rate (sensitivity), false positive rate and true negative rate (specificity) is calculated using the confusion matrix values represented in Figure 5.2. The confusion matrix is a table used for the purpose of describing the performance of a classifier on a set of data in which true values are no longer unknown.



		Predicted class	
		P	*N*
Actual Class	*P*	True Positives (TP)	False Negatives (FN)
	N	False Positives (FP)	True Negatives (TN)

Figure 5.2. Confusion Matrix

$$Correctly\ Classified = TP + TN \tag{5}$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{6}$$

$$Sensitivity = TPR = \frac{TP}{TP + FN} \tag{7}$$

$$Specificity = TNR = \frac{TN}{TN + FP} \tag{8}$$

5.1.2. The Area under the ROC Curve

The area under the Receiver Operating Characteristic (ROC) Curve (AUC) represents the curve between sensitivity and specificity. An area of 1 represents a perfect performance while the area of 0.5 represents an unworthy performance. We can roughly grade AUC value intervals for classifying as following Table 5.1.

Grade	AUC
A (excellent)	[0.90, 1.00]
B (good)	[0.80, 0.90]
C (fair)	[0.70, 0.80]
D (poor)	[0.60, 0.70]
F (fail)	[0.50, 0.60]

Table 5.1. The area under the ROC Curve (AUC) Grade Table

$$AUC = \int_{0}^{1} \frac{TP}{(TP+FN)} d \frac{FP}{(FP+TN)} = \int_{0}^{1} \frac{TP}{P} d \frac{FP}{N} \qquad (9)$$

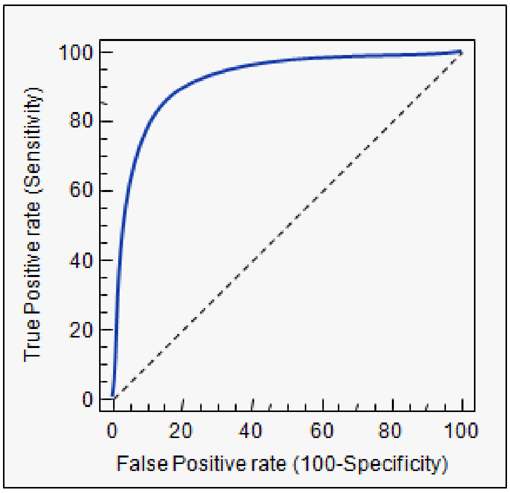


Figure 5.3. Receiver Operating Characteristic (ROC) Curve

5.2. Classification Algorithms

5.2.1. Naïve Bayes Algorithm

Naïve Bayes (NB) classifiers are built based on applying the Bayes' theorem. The classifier uses the method called maximum likelihood. The algorithm is extremely fast and it only requires a small amount of training data for the estimation. Since the training data size on this study can be considered as small, the Naïve Bayes algorithm is suitable for solving the fashion trend prediction problem. Therefore, the performance of the estimation for the Naïve Bayes algorithm is required for comparing with other algorithms. Nevertheless, most of the time the performance of the estimation is not acceptable. The examples of Naïve Bayes classifier performance evaluations using the training set with different major settings can be seen in Table 5.2. The UKE parameter represents the usage kernel estimator for numeric attributes rather than normal distribution and the USD parameter represents the usage of supervised discretization to convert numeric attributes to normal ones. As a result of the performance evaluation, it can be observed that the classifier performs better with the inclusion of USD parameter on the training set. The best accuracy obtained using the Naïve Bayes classifier is 60.0%.

UKE	USD	Correctly Classified	Accuracy (%)	The Area under the ROC Curve (AUC)
False	False	84/165	50.9%	0.537
True	False	81/165	49.1%	0.547
False	True	99/165	60.0%	0.547
True	True	81/165	49.1%	0.547

Table 5.2 Naïve Bayes Classifier Performance Evaluations Table

5.2.2. K-Nearest Neighbors Algorithm

K-Nearest Neighbors (k-NN) is a lazy learning or instance-based learning classifier. In k-NN classification, an object is classified by the majority votes of its neighbors. The object assigned to another neighbor object amounts the k nearest neighbors. The constant k in the algorithm is a positive integer that usually takes small values like 1,2,3,4. For example, when the k is equal to 1, then the object is simply assigned to that single nearest neighbor. The algorithm is very simple to implement, able to handle large training data and very robust to noisy training data. The efficient value of the constant k is needed to be determined beforehand. The computation cost of the algorithm is very high due to requiring to compute the distance of each instance to all the training instances. Performance of the K-Nearest Neighbors algorithm is evaluated for first 20 k values (1, 2, 3, …, 19, 20) to analyze the performance of the algorithm with different constants. The performance of the algorithm is used to represent the lazy learning method’s performance. The accuracy changes of the k-NN classifier performance evaluations with the different k values (1, 2, 3, …, 19, 20) are represented in Figure 5.4. The evaluations in this figure are made by making the window size parameter equal to the number of instances and not skipping the identical instances. The best accuracy obtained using the k-NN classifier is 62.4% with k equals to 9.

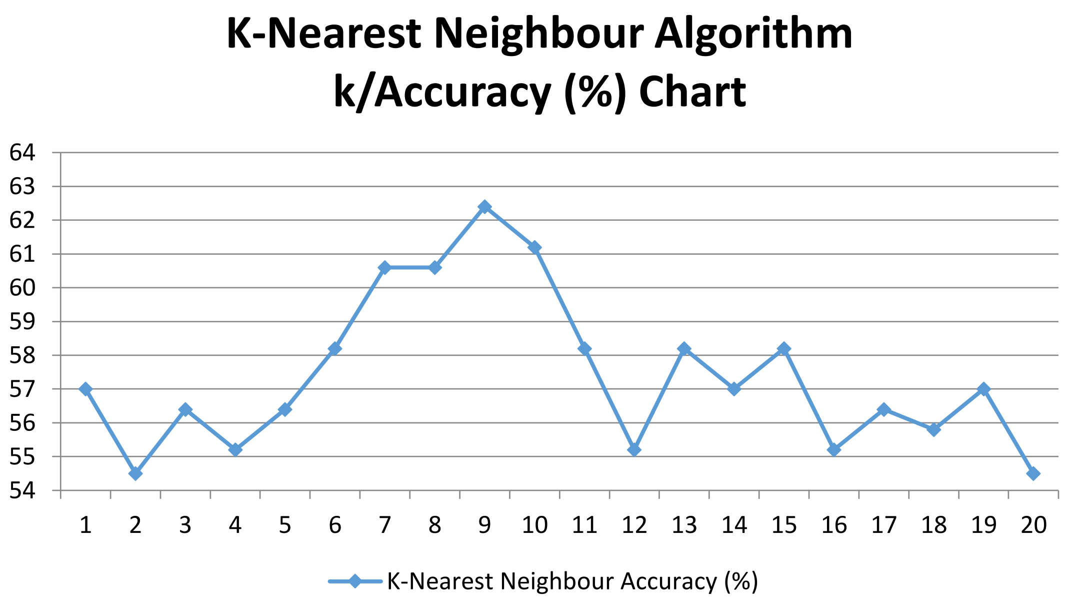


Figure 5.4. K-Nearest Neighbors Algorithm K/Accuracy (%) Chart

5.2.3. Regression Algorithms

Regression algorithms approach the prediction problems different than classification algorithms. In classification algorithms, the task is predicting a discrete class label wherein regression algorithms the task is predicting a continuous quantity. Despite being different, there are overlaps between classification and regression algorithm approaches. A classification algorithm may predict a continuous value in the form of a probability for a class label. In addition to that, a regression algorithm may also predict a discrete value in the form of an integer quantity. The performance evaluations of these two algorithms are also different. Classification algorithms evaluated using accuracy, whereas regression algorithms are evaluated using root mean square error (RMSE).

It is possible to accept the fashion prediction problem as a regression problem and attempt to solve the problem using regression algorithms. Therefore, the analysis and performance comparisons are also needed for regression algorithms. Linear regression and logistic regression algorithms are the two main regression algorithms this study is dwelled on.

In linear regression, the output is continuous. Its value can be any one of an infinite number of possible values. The Logistic regression, on the other hand, is considered as a binary classification algorithm represented in Figure 5.5, therefore, the output is not continuous and has only a limited number of possible values. Since in this study fashion trend prediction problem is considered as a binary classification problem, the logistic regression algorithm is the more suitable regression algorithm.

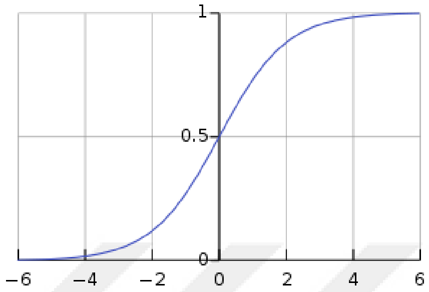


Figure 5.5. Binary Logistic Regression

The best accuracy (60.0%) performance evaluation metrics for the Logistic Regression classifier is obtained when Akaike Information Criterion (AIC) is used to determine when to stop boosting iterations instead of the cross-validation or training errors.

5.2.4. Support Vector Machines

Support vector machine (SVM) is a representation of the training data as points in space. That points are separated into categories by a clear gap that should be as wide as possible. New training instances are mapped into the same space and they are expected to belong to a category based on which side of the gap they fall (Madge, 2015). SVM uses a subset of training points in the decision function and this makes the algorithm memory efficient. On the other hand, support vector machines do not provide class probabilities. SVM can be used on binary classification problems as seen in Figure 5.6, a similar SVM can be applied to fashion trend prediction problem as well. The problem on binary classification is to find the only optimal margin of the separating hyperplane that provides a maximum wide boundary between the classes. This optimal margin guarantees the lowest rate of misclassification (Stanevski & Tsvetkov, 2005).

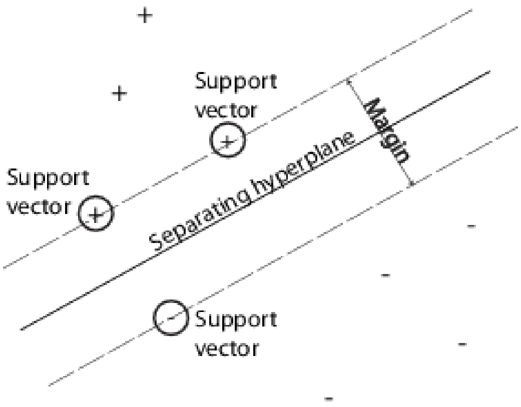


Figure 5.6. Binary Support Vector Machine

The best performance evaluation results for the SVM classifier are obtained when Logistic Regression is used as a calibrator and it produced the same results with 60.0% accuracy.

5.2.5. Stochastic Gradient Descent

Stochastic Gradient Descent (SGD) is a simple yet efficient approach to linear classifiers. The algorithm excels when the training set is very large. It also supports a variety of loss functions and classification penalties. SGD is very efficient and has a low-cost implementation but a high-cost feature scaling. SGD is considered to be a more efficient approach compared to SVM and Logistic Regression algorithms. For that reason, the performance of the SGD algorithm is also included in the classifier comparison.

The best performance evaluations result that is obtained from the SGD algorithm did not show any improvement. The evaluations have been conducted using both Hinge Loss and Log Loss functions, however, the performance evaluations still resulted in 60.0% accuracy at best.

5.2.6. Multilayer Perceptron

Multilayer Perceptron (MLP) is a class of feed-forward Artificial Neural Network (ANN). MLP algorithm is very robust against the noise in the data and the high length time series (Nanopoulos, Alcock, & Manolopoulos, 2001). MLP contains at least three layers of nodes. These are the input layer, hidden layer, and output layer. An example model can be observed in Figure 5.7. Apart from input layer nodes, each node is a neuron that uses a non-linear activation function. MLP uses back-propagation for training. Using non-linear activation and having multiple layers distinguish MLP from Linear Perceptron. MLP can have multiple hidden layers and MLP with the single hidden layer is referred to as a vanilla neural network. Nodes in the input and output layers of the MLP can be arranged to build a binary classification model that offers a solution to the fashion trend prediction problem.

The MLP classifier was one of the hardest classifiers in this study to tune its parameters since even a minor change in the learning rate or momentum parameter could affect the performance of the classifier drastically. After performance evaluations with multiple variations of the settings, the performance obtained from previous evaluations neither matched nor improved. The best accuracy (58.8%) for the classifier is obtained by using only one hidden layer, learning rate as 0.28 and momentum value as 0.3.

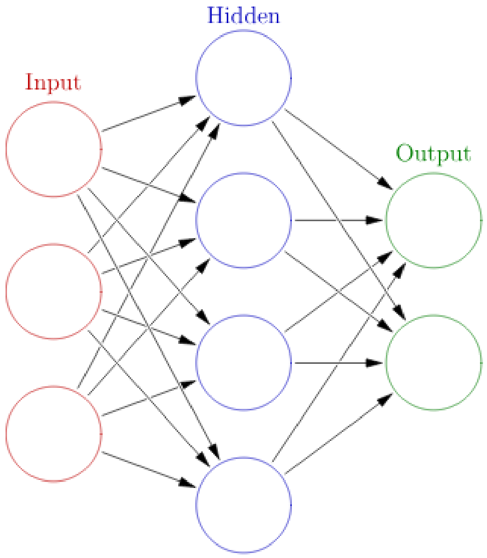


Figure 5.7. Multilayer Perceptron

5.2.7. Decision Tree Algorithms

Decision Tree algorithm classifies the data by producing a sequence of rules. It requires a small amount of data, has a low-cost implementation and simple to understand and visualize. However, it can create very complex trees that do not perform well. In addition, decision trees can be unstable because even a very small difference in the training data can cause generating a completely different tree.

The C4.5 algorithm also referred to as the J48 algorithm is used to generate a decision tree. The C4.5 algorithm generates decision trees from the training data using the information entropy concept. Those generated decision trees can be used for classification and therefore can be used for solving fashion trend prediction binary classification problem.

Random Forest (RF) Algorithm fits a number of decision trees generated by Random Tree (RT) algorithms on various sub-samples of the training data and uses the average for improving the prediction accuracy of the model as well as controlling over-fitting. The sub-sample size does not change but samples are drawn with replacement. Random Forest algorithm provides a reduction in over-fitting and in most cases, it is more accurate than decision tree algorithms. However, they are very complex and have a high-cost implementation. Random Forest algorithm is one of the top algorithms to solve classification problems (Ballings, Van den Poel, Hespeels, & Gryp, 2015). For that reason, the Random Forest algorithm is expected to solve the fashion trend prediction problem with a high performance. In addition to that, Random forest algorithm provides a direct comparison to the C4.5 algorithm due to both being decision tree based algorithms.

The tree classifiers have shown greater overall success compared to the previous classifiers. The C4.5 classifier produced the best accuracy (61.2%) when evaluated as unpruned and used a minimum of 3 instances per leaf. The Random Tree classifier produced its best accuracy (65.5%) when evaluated with choosing a single randomly chosen attribute at each node. The Random Forest classifier performed better than all the other previous classifiers. The evaluations are performed with constantly increasing the number of iterations until the peak point for accuracy (67.9%) is found after 20000 iterations. The accuracy changes of the Random Forest classifier with increasing iterations are shown in Figure 5.8.

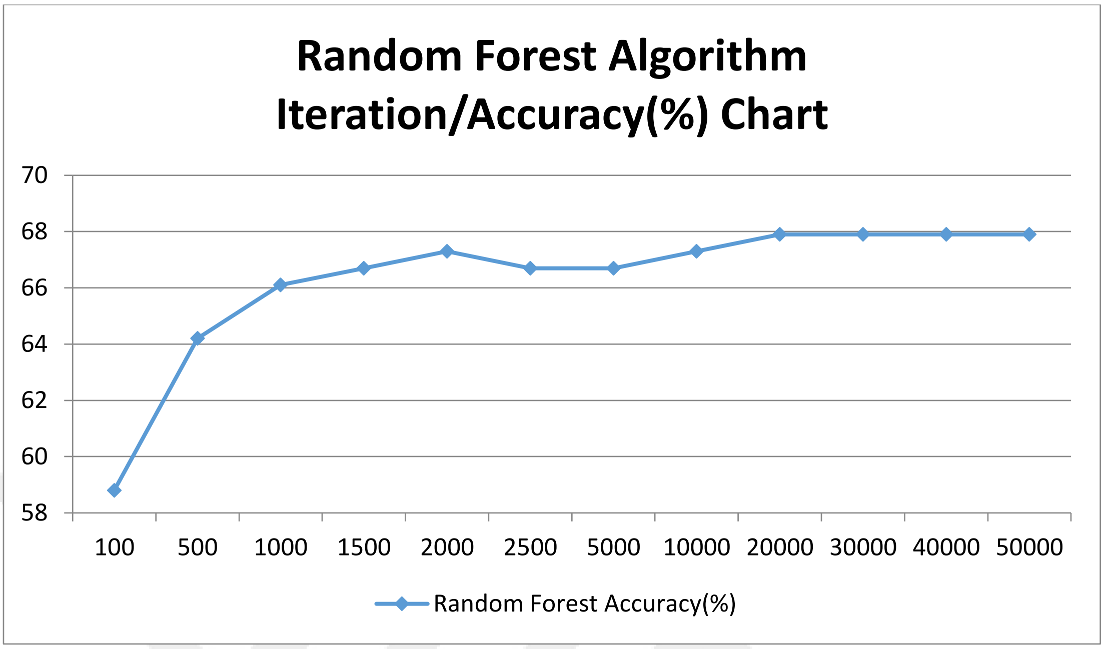


Figure 5.8 Random Forest Algorithm Iteration/Accuracy Chart

5.3. Machine Learning Performance Comparisons

n = 165	**True Positive (TP)**	**True Negative (TN)**	**False Positive (FP)**	**False Negative (FN)**	**Correctly Classified**
NB	60	39	34	32	99
k-NN	71	32	41	21	103
LR	60	39	34	32	99
SVM	60	39	34	32	99
SGD	60	39	34	32	99
MLP	58	39	34	34	97
C4.5	60	41	32	32	101
RT	68	40	33	24	108
RF	72	40	33	20	112
MV	74	42	31	18	116

Table 5.3. Machine Learning Performance Evaluation Comparisons Table 1

n = 165	**Accuracy (%)**	**Sensitivity (TPR)**	**Specificity (TNR)**	**The Area under the ROC Curve (AUC)**
NB	60.0 %	0.652	0.534	0.547
k-NN	62.4%	0.772	0.438	0.559
LR	60.0%	0.652	0.534	0.533
SVM	60.0%	0.652	0.534	0.546
SGD	60.0%	0.652	0.534	0.561
MLP	58.8%	0.630	0.534	0.581
C4.5	61.2%	0.652	0.562	0.550
RT	65.5%	0.739	0.548	0.644
RF	67.9%	0.783	0.548	0.674
MV	70.3%	0.804	0.575	0.690

Table 5.4 Machine Learning Performance Evaluation Comparisons Table 2

The abbreviations used for the classifiers in Table 5.3, Table 5.4 and Table 5.5:

- NB, Naïve Bayes
- k-NN, K-Nearest Neighbor
- LR, Logistic Regression
- SVM, Support Vector Machines
- SGD, Stochastic Gradient Descent
- MLP, Multilayer Perceptrons
- C4.5, C4.5 (J48)
- RT, Random Tree
- RF, Random Forest
- MV, Majority Voting

The performances of the well-known algorithms are compared in Table 5.3, Table 5.4 and Table 5.5. The algorithms are compared with respect to higher accuracy and higher AUC metric values. Based on the comparisons, the best result is provided by ensemble machine learning methods, mainly the Random Tree algorithm provided the best performance with 67.9% accuracy.

The sensitivity and specificity evaluation metric results show that the true class of the training set is overall more successful than the false class of the training set. Therefore, the training set needs healthier, informative and homogenous instances.

In order to improve the evaluation performance metrics, the ensemble classifiers (the C4.5 classifier, the Random Tree classifier and the Random Forest classifier) are combined using the Majority Voting technique and as a result, the accuracy increased to 70.3%.

CHAPTER 6
CONCLUSION AND FUTURE WORK

In this study, fashion trend prediction is conducted by using machine learning techniques. A fashion concept life cycle that consists of the proposed state, the emerging state, the trend state and the demode state is suggested in the study. Based on this life cycle, it is stated that, in order to make safer investments on a certain fashion concept, the textile brands in the fashion industry would like to know the likelihood of a fashion concept in the emerging stage becoming a trend on the following stage of the life cycle. Therefore, the problem of the study focused on predicting the fashion trend for the emerging fashion concepts.

The textile database for this study is acquired from Followl.io web application of the Galaksiya Information Technologies Ltd. The database contained textile product information from January 2017 to July 2018 (1,5 years). The behavior of each textile product containing a certain emerging fashion concept and their properties are analyzed in detail in order to extract the most informative and relevant machine learning features.

The fashion trend prediction problem in this study is approached as a supervised binary classification problem. Based on this knowledge, a training set is prepared using the extracted machine learning features. These features in the training set are later edited by the results of feature comparison methods and classifier performance evaluations.

A group of classifiers that can be applied to solve a binary supervised learning problem is examined. These classifiers are Naïve Bayes, K-Nearest Neighbor (k-NN), Linear Regression, Logistic Regression, Support Vector Machines (SVM), Stochastic Gradient Descent (SGD), Multilayer Perceptrons (MLP), C4.5 (J48), Random Tree and Random Forest.

The performance evaluation for these classifiers is conducted by applying 10 Folds Cross-Validation technique. Repetitive evaluations are tested for each classifier by tuning the algorithm-specific parameters in order to obtain the best evaluation performance for that classifier. The evaluation methods used by performance comparisons are the confusion matrix evaluation methods, (accuracy, TPR, and TNR) and AUC values. The best-obtained performances of each classifier are later compared with each other by using the same evaluation methods in order to obtain the classifier with the best performance. The result of the performance comparisons has shown that the performances of the ensemble machine learning methods, mainly the Random Forest algorithm provide the best performance with 67.9% accuracy and after combined with other ensemble classifiers with majority voting method, accuracy can be boosted up to 70.3%.

The algorithm performance metrics in this study especially the accuracy, true positive rate (TPR) and true negative rate (TNR) when compared to other machine learning prediction studies suggest that there is still a significant room for improvement. The author of this study suggests that overall classifier performance might be improved by selecting healthier, more informative and more homogeneous classed training instances. In addition to that, including the substantial properties of the textile database such as discount and stock in the machine learning feature extraction stage after eliminating the uncertainties on the textile product data will also likely make an improvement on the general classifier performances. Furthermore, the author also suggests that the addition of the features that are extracted from the numeric versions for these properties, for example, a percentage information for the discount and the numeric information for the available stock alongside with their binary categorical values will also likely make an improvement on the general classifier performances.

The weighted features brand, demography and garment variance features require deeper analysis on how to decide on each individual weight value. The more informative and relevant weights can be calculated using the fuzzification and the analytic hierarchy process (AHP) techniques.

Experimenting more on the classifier parameter tuning, using more informative and relevant training set and experimenting with different classifiers with investing a sufficient time period for these works as well as using different resampling techniques besides 10 Folds Cross-Validation such as Bootstrapping might also provide a performance improvement for the problem.

In conclusion, this study shows that the fashion trend prediction with using machine learning on the Followl.io textile database can be accomplished with an accuracy of over 60% with a variety of machine learning models and can be increased further to over 70% with combining ensemble machine learning models using majority voting.

The performance results of the study can be considered sufficient due to the fact that the provided database was not built for the purpose of being used by machine learning techniques. This performance result provides a start point for the Galaksiya company to put these studies into practice and start predicting the emerging fashion concepts. The study also encourages the company to support more machine learning studies in the future for fashion predictions and improve the Followl.io textile database in order to accommodate the machine learning models better with providing more informative and relevant features.

The study also reveals that the number of fashion trend prediction studies are very limited. More studies can be conducted based on fashion trend prediction problem, providing a wide variety of machine learning problems and feature suggestions and set of machine learning performance evaluation comparisons.

REFERENCES

Ballings, M., Van den Poel, D., Hespeels, N., & Gryp, R. (2015, November). Evaluating Multiple Classifiers for Stock Price Direction Prediction. *Expert Systems with Applications, 42*(20), 7046-7056. Retrieved from https://doi.org/10.1016/j.eswa.2015.05.013

Barnard, M. (2002). *Fashion as Communication* (2nd ed.). Routledge.

Blum, A. L., & Langley, P. (1997). Selection of Relevant Features and Examples in Machine Learning. *Artificial Intelligence, 97*(1-2), 245-271. Retrieved from https://doi.org/10.1016/S0004-3702(97)00063-5

Bontempi, G. (2013). Machine Learning Strategies for Time Series Prediction. *Machine Learning Summer School.* Hammamet. Retrieved from http://www.ulb.ac.be/di/map/gbonte/ftp/time_ser.pdf

Dadoun, M. (2017). Predicting Fashion Using Machine Learning Techniques. Retrieved from http://www.nada.kth.se/~ann/exjobb/mona_dadoun.pdf

DB-Engines Ranking. (2018, August). Retrieved from DB-Engines: https://db-engines.com/en/ranking

Easey, M. (2009). *Fashion Marketing* (3rd ed.). Blackwell Publishing.

Effingham, N. (2013). *An Introduction to Ontology.* Polity.

Gormley, C., & Tong, Z. (2015). *Elasticsearch: The Definitive Guide: A Distributed Real-Time Search and Analytics Engine* (1st ed.). O'Reilly Media.

Gruber, T. (2008). Ontology. In *Encyclopedia of Database Systems.* Springer-Verlag.

Kara, Y., Acar Boyacıoğlu, M., & Baykan, Ö. K. (2011, May). Predicting direction of stock price index movement using artificial neural networks and support vector machines: The sample of the Istanbul Stock Exchange. *Expert Systems with Applications, 38*(5), 5311-5319. Retrieved from https://doi.org/10.1016/j.eswa.2010.10.027

Kawamura, Y. (2018). *Fashion-ology: An Introduction to Fashion Studies* (2nd ed.).

Bloomsbury Publishing.

Kertakova, M., Mojsov, K. D., Andronikov, D., Janevski, A., Jordeva, S., Golomeova, S., . . . Ignjatov, I. (2018). Fashion in the Early Twentieth Century - Fashion and Fashion Trends Analysis During the First and Second Decades of the Twentieth Century. *Tekstilna Industrija, 66*(2), 35-43.

Kotsiantis, S. B. (2007). Supervised Machine Learning: A Review of Classification Techniques. *Informatica*, *31*, 249-268.

Madge, S. (2015). Predicting Stock Price Direction using Support Vector Machines. Retrieved from https://www.cs.princeton.edu/sites/default/files/uploads/saahil_madge.pdf

Mello, P., Storari, S., & Valli, B. (2008, June). A Knowledge-Based System for Fashion Trend Forecasting. *New Frontiers in Applied Artificial Intelligence: 21st International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems*, 425-434. doi:10.1007/978-3-540-69052-8_45

Mello, P., Storari, S., & Valli, B. (2010, January). Application of Machine Learning Techniques for the Forecasting of Fashion Trends. *Intelligenza Artificiale*, *4*(1), 18-26. Retrieved from https://www.researchgate.net/publication/220672760_Application_Of_Machine_Learning_Techniques_For_The_Forecasting_Of_Fashion_Trends

Nanopoulos, A., Alcock, R., & Manolopoulos, Y. (2001). Feature-based Classification of Time-Series Data. Retrieved from https://www.researchgate.net/publication/234800113

Pan, J. Z. (2009). Resource Description Framework. In S. Staab, & R. Studer, *Handbook on Ontologies* (pp. 71-90). Springer.

Patel, J., Shah, S., Thakkar, P., & Kotecha, K. (2015, January). Predicting stock and stock price index movement using Trend Deterministic Data Preparation and machine learning techniques. *Expert Systems with Applications*, *42*(1), 259-268. Retrieved from https://doi.org/10.1016/j.eswa.2014.07.040

Preston, S. H., Heuveline, P., & Guillot, M. (2000). *Demography: Measuring and Modeling Population Processes: Measuring and Modelling Population*

Processes. Blackwell Publishers.

Soltani, S. (2012). Strategic Marketing Plan in Product Life Cycle.

Sproles, G. B. (1981). Analyzing Fashion Life Cycles: Principles and Perspectives. *Journal of Marketing, 45*(4), 116-124. doi:10.2307/1251479

Stanevski, N., & Tsvetkov, D. (2005). Using Support Vector Machine as a Binary Classifier. *International Conference on Computer Systems and Technologies - CompSysTech.*

Thacker, U., Pandey, M., & Rautaray, S. S. (2018). Review of Elasticsearch Performance Variating the Indexing Methods. *Progress in Intelligent Computing Techniques: Theory, Practice, and Applications. Advances in Intelligent Systems and Computing, 719*. doi:https://doi.org/10.1007/978-981-10-3376-6_1

Turker, D., & Altuntas, C. (2014, October). Sustainable Supply Chain Management in the Fast Fashion Industry: An Analysis of Corporate Reports. *European Management Journal, 32*(5), 837-849. doi:https://doi.org/10.1016/j.emj.2014.02.001

Weka 3: Data Mining Software in Java. (2018, August). Retrieved from Machine Learning Group at the University of Waikato: https://www.cs.waikato.ac.nz/ml/weka/

Witten, I. H., Frank, E., Hall, M. A., & Pal, C. J. (2016). *Data Mining: Practical Machine Learning Tools and Techniques* (4th ed.). Morgan Kaufmann.

Zakamulin, V. (2017). *Market Timing with Moving Averages: The Anatomy and Performance of Trading Rules.* Springer Nature. doi:10.1007/978-3-319-60970-6