

# **TAG-BASED DYNAMIC RANKING SYSTEM FOR ORGANIZATION RELATED NEWS**

**A Thesis Submitted to  
the Graduate School of Engineering and Sciences of  
İzmir Institute of Technology  
in Partial Fulfillment of the Requirements for the Degree of**

**MASTER OF SCIENCE**

**in Computer Engineering**

**by  
Mustafa Tunahan ÖZKAN**

**May 2018  
İZMİR**

We approve the thesis of **Mustafa Tunahan ÖZKAN**

Examining Committee Members:

---

**Dr. Tuğkan TUĞLULAR**

Department of Computer Engineering, İzmir Institute of Technology

---

**Dr. Selma Tekir**

Department of Computer Engineering, İzmir Institute of Technology

---

**Assoc. Prof. Dr. Derya BİRANT**

Department of Computer Engineering, Dokuz Eylul University

**15 May 2018**

---

**Dr. Tuğkan TUĞLULAR**

Supervisor, Department of Computer Engineering  
İzmir Institute of Technology

---

**Assoc. Prof. Dr. Yusuf Murat ERTEN**

Head of the Department of  
Computer Engineering

---

**Prof. Dr. Aysun SOFUOĞLU**

Dean of the Graduate School of  
Engineering and Sciences

## **ACKNOWLEDGMENTS**

First of all, I would like to thank my Supervisor, Tuğkan TUĞLULAR, who advised and encouraged me all the way during this thesis study. I am thankful for his assistance, patience, continuous support and for sharing his knowledge with me. His guidance helped me in all the time of research and writing of this thesis.

I would like to thank my friends for all the support and their motivation during busy and tiring times.

I would like to thank company BİMAR for providing the data.

Finally, I would like to express my infinite gratitude to my parents for their unconditional love and endless support without any expectations. I dedicate this thesis work to them.

# **ABSTRACT**

## **TAG-BASED DYNAMIC RANKING SYSTEM FOR ORGANIZATION RELATED NEWS**

In information systems, tags are keywords or terms, which represent a piece of information. They provide to define an item and help it to be found again through searching or browsing. Tags have gained popularity due to the growth of social sharing, social bookmarking, organization network and social network websites. In addition, tags are also used to express prominent events and noticeable topics in the news. In this thesis, we propose a tag-based statistical learning approach to predict the shareability of news in an organization network. We represented features with tags by using different methods and adopted several classifiers to predict the shareability of news. We model this problem with a binary classification problem, where shareable news are considered as the positive and non-shareable news are considered as the negative class. The experimental results indicate that there is no general best classifier for the study of shareability prediction for organization related news but depending on the dataset and represented features we can adopt an optimal classifier.

# ÖZET

## ETİKET TABANLI KURUMSAL DİNAMİK HABER SIRALAMA SİSTEMİ

Bilgi sistemlerinde, etiketler bilgi kümelerini temsil eden anahtar kelimeler veya terimlerdir. Bir ögeyi tanımlamayı sağlarlar ve bu ögenin araştırılarak veya göz atılarak tekrar bulunmasına yardımcı olurlar. Etiketler sosyal paylaşım, sosyal imleme, kurumsal ağ ve sosyal ağ sitelerinin büyümesi nedeniyle popüler olmuşlardır. Bununla birlikte, etiketler ayrıca haberlerde öne çıkan olayları ve dikkat çekici konuları ifade etmek için kullanılır. Bu tez çalışmasında, kurumsal bir organizasyon ağındaki haberlerin paylaşılabilirliğini tahmin etmek için etiket tabanlı bir istatistiksel öğrenme yaklaşımı önermekteyiz. Farklı yöntemler kullanarak etiketlerden öznitelikler çıkardık ve haberlerin paylaşılabilirliğini tahmin etmek için birkaç sınıflama yöntemi kullandık. Bu problemi, paylaşılabilir olarak tahminlenen haberlerin olumlu ve paylaşılabilir olarak tahminlenen haberlerin olumsuz sınıf olarak kabul edildiği ikili bir sınıflandırma problemi olarak modelledik. Deneysel sonuçlar, organizasyonla ilgili haberlere yönelik paylaşılabilirlik tahmini çalışması için genel bir en iyi sınıflayıcının olmadığını ancak veri setine ve çıkarılan özniteliklere bağlı olarak bu çalışma için en uygun sınıflayıcıları kullanabileceğimizi göstermektedir.

# TABLE OF CONTENTS

LIST OF FIGURES .....	viii
LIST OF TABLES .....	ix
CHAPTER 1. INTRODUCTION .....	1
1.1. Motivation .....	2
1.2. Thesis Goals and Contributions .....	2
1.3. Outline of Thesis .....	3
CHAPTER 2. RELATED WORK.....	4
CHAPTER 3. RESEARCH BACKGROUND .....	7
3.1. Artificial Intelligence .....	7
3.2. Machine Learning .....	9
3.2.1. Machine Learning Methods.....	13
3.2.2. Machine Learning Applications .....	14
3.2.3. Feature Engineering and Feature Selection .....	15
3.2.3.1. Feature Extraction and Engineering.....	16
3.2.3.1.1. Feature Engineering on Text Data.....	17
3.3. Data Mining .....	19
3.3.1. Functions and Application Fields of Data Mining .....	21
3.3.2. The CRISP - DM Methodology .....	24
CHAPTER 4. PROPOSED METHOD.....	26
4.1. Prediction Challenges .....	26
4.2. Model Architecture .....	27
4.3. Classification Methods For Predictive Model .....	28
4.3.1. Support Vector Machine Classifier .....	28
4.3.2. Naive Bayes Classifier.....	29
4.3.3. Neural Network Classifier .....	29
4.3.4. Decision Forest Classifier.....	30

4.4. Feature Representation, Hashing and Selection .....	30
<b>CHAPTER 5. EXPERIMENTAL RESULTS .....</b>	<b>34</b>
5.1. Preparing Dataset .....	34
5.2. Data Preprocessing.....	37
5.3. Splitting Method of Dataset.....	38
5.4. Evaluation Metrics .....	39
5.5. Classifiers and Results.....	42
5.5.1. SVM .....	42
5.5.2. Naive Bayes.....	44
5.5.3. Neural Network .....	45
5.5.4. Decision Forest.....	46
<b>CHAPTER 6. CONCLUSION AND FUTURE WORK .....</b>	<b>48</b>
6.1. Conclusion.....	48
6.2. Future Work .....	49
<b>REFERENCES .....</b>	<b>50</b>

# LIST OF FIGURES

<b><u>Figure</u></b>	<b><u>Page</u></b>
Figure 3.1. Some disciplines related with Artificial Intelligence. ....	8
Figure 3.2. Machine Learning as a multidisciplinary domain. ....	11
Figure 3.3. A Standard Machine Learning Pipeline. ....	12
Figure 3.4. A standard pipeline for feature engineering, scaling, and selection.....	16
Figure 3.5. Sample Text Corpus (Collection of Text Documents).....	17
Figure 3.6. Pre-processed Corpus. ....	18
Figure 3.7. Bag of Words Model for Sample Corpus.....	18
Figure 3.8. Bag of Bi-Gram Model for Sample Corpus. ....	19
Figure 3.9. TF-IDF based Matrix for Sample Corpus. ....	19
Figure 3.10. Phases of knowledge discovery.....	21
Figure 3.11. Simple linear partitioning on loan dataset. ....	22
Figure 3.12. Simple linear regression where total debt is a linear function of income. .	22
Figure 3.13. An example clustering visualization over loan dataset. ....	23
Figure 3.14. Four level abstraction of CRISP-DM concept. ....	25
Figure 4.1. Representation of model architecture.....	27
Figure 4.2. Representation of SVM hyperplane. ....	29
Figure 4.3. N-Gram Features. ....	31
Figure 4.4. TF- IDF values for each bigram tag. ....	31
Figure 4.5. Hashed Features. ....	32
Figure 5.1. Sample News' Tags.....	35
Figure 5.2. Distribution of Tags.....	36
Figure 5.3. Distribution of #Türkiye in shared news.....	36
Figure 5.4. Percent of shared news that consist of both #Türkiye and #Ihracat.....	37
Figure 5.5. Sample text preprocessing.....	37
Figure 5.6. 10-fold Cross Validation. ....	39
Figure 5.7. Confusion Matrix. ....	40
Figure 5.8. Sample ROC curve for SVM classifier. ....	41
Figure 5.9. Evaluation results for first pipeline. ....	43
Figure 5.10. Evaluation results for second pipeline.....	44
Figure 5.11. Fully Connected Neural Network.....	45



# LIST OF TABLES

<b><u>Table</u></b>	<b><u>Page</u></b>
Table 5.1. Statistics and information about tags. ....	35
Table 5.2. Evaluation Results for Naive Bayes. ....	44
Table 5.3. Evaluation Results for Neural Network.....	45
Table 5.4. Evaluation Results for Decision Forest. ....	46
Table 5.5. Experimental Results for Machine Learning Pipelines. ....	47

# CHAPTER 1

## INTRODUCTION

Due to the increase and growth of the information dissemination through organization networks and social networks people started to need to represent information with tags to make information easier to access. They use tags to link similar or related contents and then they share these contents for other people to access information. Tagging was became popular by websites related with Web 2.0 and it is an important characteristic of many Web 2.0 applications (Breslin, Passant, & Decker, 2009; Smith, 2007). It is now also used in database systems, software applications, operating systems (Jones & Hafner, 2012). Tags are also used in machine learning systems. Some of these systems use tags to understand and predict what data users will be interested in and that will allow them to make decisions.

Thanks to evolution of machine learning, computer systems can automatically learn from past data. These systems can apply what has been learned in the past to unlabeled data using labeled examples to predict future labels. Beginning from the analysis of a labeled training dataset, the learning algorithm produces a function to make predictions about the labels of test dataset.

In organization networks people read news that have specific tags associated with their organizations' scope. They analyze and interpret these news and if they think that the news are important or they think news have potential of opportunity for organization's scope, they share them with other people in organization network. They can add new tags to news or they can edit existing tags.

In this thesis, we focus on developing a learning tag based approach to a specific organization network for the purpose of predicting the shareability of organization related news by using different machine learning methods.

## **1.1. Motivation**

As mentioned in previous section people read and analyze organization related news and then they share some of them with other people in organization networks. The organization which we work together in the scope of this thesis collects news with their tags from third party systems. During the day, huge amount of organization related news flows into the organization network from these third party systems. It is very exhausting and time consuming to read and analyze these huge amount of news for the people who works in organization. This can cause important news to be missed, or it can prevent quick action for potential opportunities in the news. Therefore, the ability to predict and rank important news for users by a dynamic system is quite important.

Tags are a kind of summary of the news and they tell what the news are about. They represent prominent events and remarkable topics in the news. When news are shared by a user, other users are primarily concerned with their tags. They tend to share specific news where specific tags are included together. Also, most of the third party news providers can only provide the tags of the news in textual format to organization network. Articles of the news are not all in textual format, they are mostly in image format. Therefore, tags can be used to predict shareability of the news.

## **1.2. Thesis Goals and Contributions**

This thesis aims to develop an approach to predict the shareability of organization related news for a specific organization. We used different statistical learning-based classifiers that predict whether a news will be share or not. We analyzed which machine learning approaches can best model the shareability prediction problem. We applied different feature representation techniques to represent features from tags to predict shareability of news.

### **1.3. Outline of Thesis**

This thesis is organized as follows. The next chapter provides a literature overview. Chapter 3 gives a background information about artificial intelligence, machine learning and data mining concepts and their applications. In Chapter 4, our predictive approach is explained. In Chapter 5, we explain the dataset and experiments that we have conducted. Finally, Chapter 6 provides final remarks and discusses future research.

## CHAPTER 2

### RELATED WORK

An increasing and growing line of research has been followed on information dissemination through organization networks and social networks. These studies assert that network shares can play an important role for the dissemination of diverse information. These researches based on the idea that the information is spread by diverse infection mechanisms (Granovetter, 1978; Kempe, Kleinberg, & Tardos, 2003). In this context, it is important to disseminate interesting and relevant information to reduce irrelevant information pollution. At this point, there is a need for automated systems that allow users to quickly access relevant information. These systems should be able to predict what content users will be interested in and that will enable them to make strategic decisions.

The study of predicting the tendency of sharing of news based on tags will be modeled as a binary classification study, in an other saying the approach underlying this study is a content-based prediction approach. Content based prediction approach is one of the essential research areas of the machine learning domain and it has been widely studied. In this chapter, this approach in different domains will be explained. In a study Szabo & Huberman (2010) used two content sharing portals Youtube and Digg to demonstrate how by monitoring responses to the stories, they can predict the popularity of such stories with considerable accuracy.

Another study was done by Hong et al. (2011) to predict the popularity of tweets. They proposed a feature-weighted model, which predicts the popularity of tweets in terms of the number of potential retweets. This study is a multiple classification study in which a tweet will be assigned to one of the four possible classes. The classes are as follows: 0: not retweeted, 1: retweeted less than 10 times, 2: retweeted less than 100 times and 3: retweeted more than 100 times. Their feature extraction model extracts a set of features from the tweet and from the user who published the tweet.

Chaturvedi et al. (2017) used various machine learning methods to predict the user ratings for an article. To solve this problem, they used supervised learning methods

such as linear regression, Naive Bayes and logistic regression. Each of these methods have different character which impacts the solution of the given problem. The aim of their study is to explore the impact of features extracted from feedback categories on recommendations and also on other features. To reach this aim, they used the extracted features in machine learning models with different feedbacks in the dataset. They have generated new datasets by choosing the articles belonging to the most frequent categories and removing the articles belonging to the infrequent categories and then run experiments on them to see the impact of number of classes on the efficiency of the system. Finally, they have performed analysis of the outcomes acquired from using feature combinations in different models to understand the right model and features required to make sensible recommendations.

Mooney & Roy (2000) developed a next-generation content-based recommender system. Their system uses information extraction and a machine learning algorithm for text categorization. They made a subject search on the Amazon website to get a list of book definition URLs. System downloads these pages and extracts information belong to the book such as author, title, publications etc. After that text pre-processing is performed on these information. The content related with the summaries, published reviews and comments were clustered into a single feature named description. They trained the system by querying on certain authors and titles to obtain relevant books. The books were presented to the users who were asked to rate their interest in the book. These ratings form the explicit input on which to create the user profile. The system then learns a profile of the user using a Bayesian learning algorithm and creates a ranked list of the most recommended additional titles from the system's catalog. After that, the classifier predicts user rankings for the other books. Finally, the top-scoring books were recommended to the user. System was executed on a few datasets. Experiments were performed on the book recommendations and two different users rated the books. The efficiency of the system was examined using a 10-fold validation method for different number of training documents. The outcomes showed that the top recommendations performed by system were found interesting to the users according to randomly selected documents.

Phelan et al. (2009) concentrate on building a recommender system which uses micro-blogging activity to recommend news stories. Their recommender system, Buzzer, uses RSS feeds to which the users have subscribed. Buzzer obtains the content terms from RSS and Twitter feeds and ranks articles by using them. Buzzer uses three

different methods to recommend news stories. First one is called PublicRank which obtains tweets from Twitter's timeline. Second one is called Friend's-Rank which obtains tweets from people the user follows. Last one is called Content Rank which ranks articles according to term frequency and scores the articles according to the frequency of occurrence of the top RSS terms. Their system calculates the score of each article by gathering the TF-IDF (term frequency- inverse document frequency) scores across all the terms related within each article.

## CHAPTER 3

### RESEARCH BACKGROUND

In this chapter Artificial Intelligence, Machine Learning and Data Mining concepts, their methods and application fields are introduced.

#### 3.1. Artificial Intelligence

Since the innovation of computers or machines, their ability to execute different tasks went on increasing exponentially. Humans have evolved the ability of computer systems. They expanded their various working realms, they increased machine's speed, and they reduced size with respect to time. A branch of Computer Science named Artificial Intelligence strives creating the computers or machines as intelligent as mankind.

Artificial intelligence is a capability of a computer or computer controlled system to accomplish tasks commonly related with intelligent beings. It describes machines programmed to think, work and react like humans. It tries to figure out and build intelligent architectures (Russell & Norvig, 2016). It also focuses on the logic of brain thinking, learning, deciding and working while trying to solve a problem. Then it takes advantage of the outcomes of this brain process as a principle of developing intelligent software and systems.

The disciplines related with the artificial intelligence are very diverse (Johnsen, 2017). Such as Social Sciences, Cognitive Sciences, Computer Science, Biology, Psychology, Mathematics, Linguistics and Engineering. The disciplines which are illustrated in Figure 3.1. can contribute to build an intelligent system.



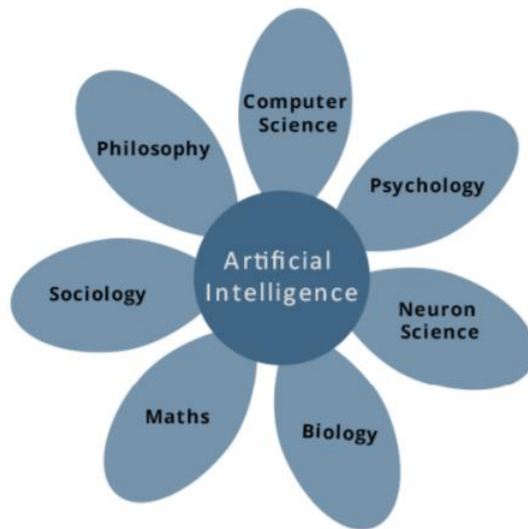


Figure 3.1. Some disciplines related with Artificial Intelligence.

Artificial Intelligence is used in the following areas:

- Expert Systems

There are some applications such as monitoring systems, process control systems, knowledge domain systems which integrate software, machine and information to impart reasoning and advising. They provide explanation and advice to the users.

- Natural Language Processing

It is feasible to communicate with the computer that understands natural language spoken by people.

- Vision Systems

These systems understand, evaluate, and figure out visual input on the computer. For example,

- A surveillance aeroplane takes photographs which are used to find spatial data or map of geographical region.
- Doctors use clinical expert system to diagnose the patient.
- Computer software is used to recognize the face of criminal with the stored portrait.

- Gaming

Artificial Intelligence plays important role in strategic games such as tic-tac-toe, poker, chess etc., where machine can compute various possible moves based on heuristic knowledge.

- Intelligent Robots

Robots can execute the operations given by people. The data such as temperature, sound, pressure, light can be detected by robots thanks to their sensors. Thanks to their various sensors, large memory, effective processors they can reveal intelligence. Additionally, they have ability to learn from their wrong decisions and they can accommodate to new environment.

- Speech Recognition

Some intelligent systems have ability to hear and understand the language in terms of sentences and their meanings while people talk to it. It can understand different accents, noise in the background, changes in human's voice due to cold or sadness etc.

- Handwriting Recognition

The handwriting recognition software has ability to read the text written on paper by a pen or on screen by a stylus. It can recognize the letters and convert it into text.

- Machine Learning

Identifying whether an email is spam or not, predicting the price of a house to sell, placing the most relevant web search results at the top of the list are some of intelligent machine learning applications.

### **3.2. Machine Learning**

Machine learning is a subdomain of computer science that provides computers to learn automatically. Thanks to machine learning, computers are capable of learning without manually programmed. The name of Machine Learning was invented by Arthur Samuel in 1959. It is developed from the study of computational learning theory and pattern recognition in artificial intelligence. Machine learning studies the work and structure of algorithms (Kohavi, 1998). These algorithms can learn from data and makes data-driven decisions or predictions through creating a model from sample inputs.

Machine learning is connected to computational statistics, which also concentrate on prediction-making through the use of computers. It has strong connections to mathematical optimization and delivers procedures, theories and application domains to the computer science field. Machine learning is also related with data mining (Mannila, 1996), where the data mining studies more on exploratory data analysis and is known as unsupervised learning. Machine learning can also be

unsupervised. It is used to understand and set up basic behavioral profiles for diverse entities and then used to detect significant anomalies.

From the perspective of data analytics, machine learning is a process used to formulate complex models and algorithms that lend themselves to prediction. In business use, it is called as predictive analytics. These analytical models provides engineers, researchers, analysts and data scientists to build dependable, reusable decisions and outcomes and expose hidden insights through learning from past relationships and trends in the data.

Some machine learning terms and concepts can be list as below:

- **Data Exploration:** Data exploration is a process of collecting information about a huge and often unstructured dataset for the purpose of exploring characteristics for focused analysis.
- **Data Mining:** Data Mining is the process of exploring actionable information from huge sets of data.
- **Descriptive Analytics:** Descriptive analytics is a process of examining a dataset for the purpose of understanding what occurred. It's useful to represent things like average money spent per customer, total stock in store, changes in sales for a specific time period.
- **Predictive Analytics:** Predictive analytics is a process of building models from historical or current data for the purpose of predicting later outcomes. Producing credit scores is a common application of predictive analysis. Financial services uses these scores to find the probability of consumers making future credit payments on time.

Machine learning has notions that have been obtained and borrowed from multiple domains and it is also a multidisciplinary domain. Figure 3.2. shows main domains that overlap with Machine learning based on methods, notions, techniques, and thoughts.

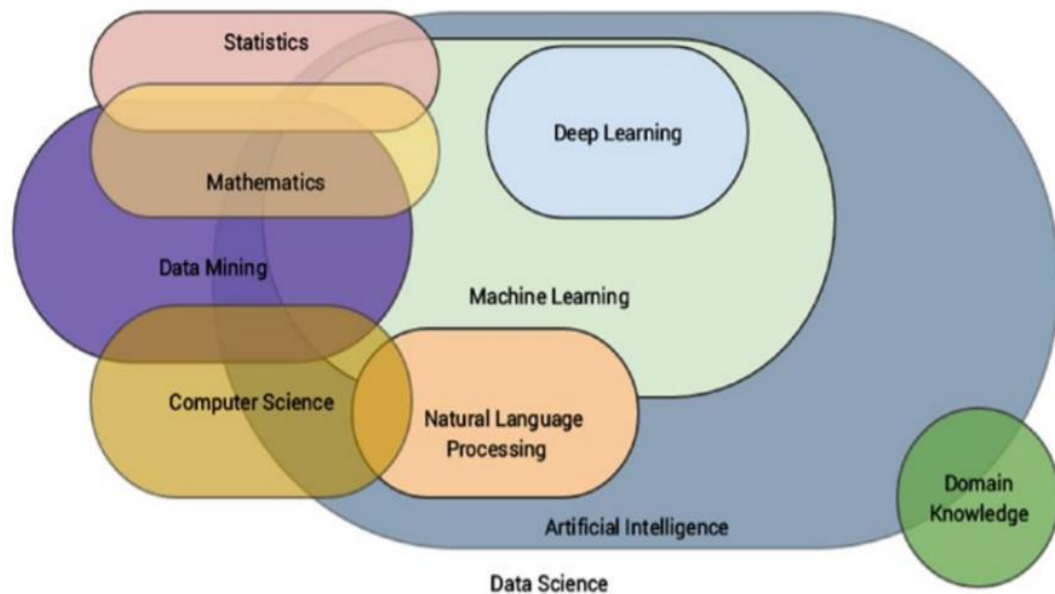


Figure 3.2. Machine Learning as a multidisciplinary domain.

The aim of machine learning, data mining, or artificial intelligence is to make people's lives more comfortable and easier, automate jobs, and take smarter resolutions. Constructing machine intelligence includes machine learning notions and implementation of models. Using a machine learning pipeline starting from collecting data to converting it into information by using machine learning algorithms is the best way to solve a machine learning problem.

A standard machine learning pipeline, as shown on Figure 3.3, substantially contains processes like data retrieval, data preparation, modeling through machine learning algorithm, model evaluation and tuning, if necessary, and deployment.

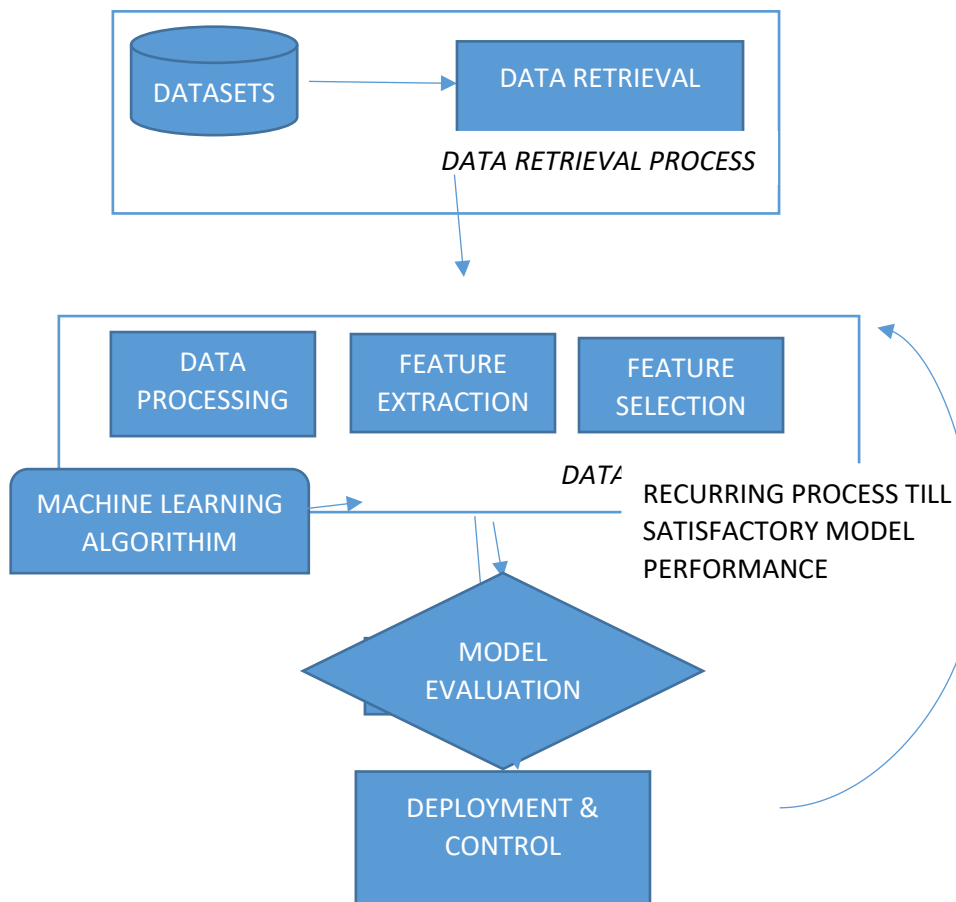


Figure 3.3. A Standard Machine Learning Pipeline.

- **Data Retrieval:**

This stage is merely about collecting and extracting data from diverse data sources.

- **Data Preparation:**

This stage covers processing, cleaning and manipulating the raw data in order to make the data ready for further processing, analysing, feature extraction and feature selection.

- **Data Processing:**

Data Processing process mainly deals with initial analysis, picking correct samples and eliminating the unnecessary noisy and dirty data in order to prepare the initial raw data for correct feature extraction and selection.

- **Feature Extraction:**  
In this section, important and relevant features are selected from the data source. Any possible potential features may be created by using existing features to make the data more meaningful.
- **Feature Selection:**  
For the purpose of preventing machine learning algorithms from getting biased, features need to be normalized. Furthermore, features set must be filtered based on their quality and importance.
- **Modeling:**  
Machine learning algorithms use the normalized data features to train the model.
- **Model Evaluation:**  
After the process of training models, they are evaluated by using test datasets. Then, they are scored based on metrics like recall, accuracy, F1 score.
- **Deployment & Control:**  
Selected models are deployed in production. They are monitored based on their predictions and outcomes.

### 3.2.1. Machine Learning Methods

Depending on amount of human supervision in the learning task, Machine Learning methods can be list as follows:

- **Supervised Learning:** Supervised learning techniques contain learning algorithms that take in data inputs and related outputs during the model training operation. The major goal is figure out and learn mapping or relation between input data and output data. This learned experience is used to to predict new output for any new input data. Regression and Classification processes are major samples for supervised learning method.
- **Semi-Supervised Learning:** Semi-supervised learning methods generally use lots of unlabeled input data and a small number of labeled data. The aim is building a supervised model based on limited labeled data, and then applying the same to large numbers of unlabeled data in order to obtain more labeled samples, train

the model on them and repeat the task. Image tagging is a common sample for semi-supervised learning process.

- **Reinforcement Learning:** Reinforcement Learning is a subdomain of machine learning, where an agent learns by communicating with its environment to reach a goal. Reward feedback is needed in order to agent to learn its behaviour; this is called reinforcement signal. It helps software agents to automatically specify the ideal behaviour within a particular condition, in order to increase its efficiency. Games such as chess, Go, Pacman are sample reinforcement learning applications.
- **Unsupervised Learning:** In unsupervised learning, learning algorithm does not need labels. It separates the data in a data set in which the data is unlabeled based on some undercover features in the data. This method is helpful for discovering the hidden structure of data and for processes like anomaly detection.

### **3.2.2. Machine Learning Applications**

Depending on the desired outputs of a machine learned system, some machine learning applications can be listed as below:

- **Classification:** Inputs are split into multiple classes, and the learner must create a model that assigns unseen inputs to one or more of these classes. Spam email message detecting is an example of classification, where the inputs are email messages and the classes are spam and not spam.
- **Regression:** Linear regression was developed in the field of statistics. It is studied as a model for understanding the relation between input and output numerical variables.
- **Clustering:** Clustering is a method of unsupervised learning and a common technique for statistical data analysis used in several fields. It helps grouping similar unlabeled entities together.
- **Density Estimation:** Density estimation is the setting up of an estimate, based on observed data, of an unobservable underlying probability density function.
- **Dimensionality Reduction:** Dimensionality reduction makes inputs simple by mapping them into a lower-dimensional space. Topic modeling is a sample

study for dimensionality reduction. A list of documents are given to program and is tasked to find out which documents cover similar topics.

- **Anomaly Detection:** Anomaly detection is a process that identifies extraordinary events or values and find problems. Fraud Detection is an example of anomaly detection. For example it is used to detect unusual credit card purchases.
- **Feature Engineering:** Feature Engineering process is a selecting or extracting features from a dataset for the purpose of improving dataset and outcomes. For example flight ticket data could be improved by days of the week and holidays.
- **Recommendation Systems:** Recommendation Systems obtain personalized information by learning the user's interests and make predictions about them.

### **3.2.3. Feature Engineering and Feature Selection**

While building machine learning systems, features and attributes are very important in building models on top of data. Machine learning algorithms can only process numeric values as inputs. On the basis of the algorithm usually mathematical equations, computations and optimizations are used. Because of this, it is nearly impossible to give raw data to algorithm as input and expect results. Data is consist of various fields, attributes, or variables. Each attribute is an natural feature of the data. More features can be derived from these natural features by using feature engineering techniques. Feature selection is another important process which provides data scientists to select best subset of features while building the right model. Because of the cyclical nature of machine learning systems (Chapman et al., 1999) extracting and engineering features from the dataset is an iterative process. It may be needed to extract new features and try out multiple selections each time build a model to get optimal model for machine learning.

Figure 3.4 represents a standard pipeline for feature engineering, scaling, and selection, as seen in (Sarkar, Bali, & Sharma, 2018).



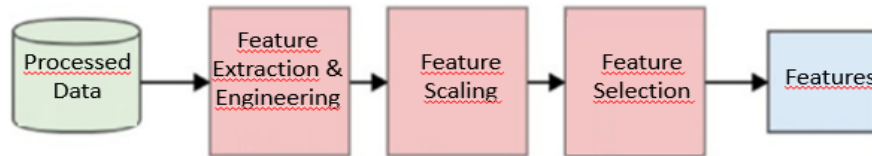


Figure 3.4. A standard pipeline for feature engineering, scaling, and selection.

### 3.2.3.1. Feature Extraction and Engineering

The task of feature extraction and engineering is probably the most important process in the Machine Learning pipeline. They are synonyms which specifies the task of using domain knowledge and mathematical transformations to transform data into features. Basically, feature engineering is the task of extracting or creating new features from data. Proper and great features provides building powerful machine learning models. So, identifying these features is the most important factor in achieving the success of the machine learning process (Domingos, 2012).

Features are also based on the underlying problem. The underlying problem here means a specific business problem or usecase to solve with the process of Machine Learning. For example classification of movies and books according to their categories is similar in terms of machine learning task but the features extracted in each case will be dissimilar. The following examples represent some feature engineering use cases:

- Obtaining peoples age from birth date and the current date
- Acquiring the average and median retweet count of specific tweets
- Obtaining word and phrase counts from news articles
- Extracting pixel information from images
- Tabulating frequencies of various grades entered by teachers

Different data types have different methods for feature extraction. For the following major data types various feature engineering methods are used:

- Text data
- Numeric data
- Categorical data
- Temporal data
- Image data

### 3.2.3.1.1. Feature Engineering on Text Data

Dealing with unstructured attributes like text and images is more challenging than dealing with structured data attributes like numeric or categorical variables. Coping with the unpredictable structure of the syntax, format, and content of documents is first difficulty in case of text documents. It is hard to extract useful information from these documents for building models. The other difficulty is converting these textual representations into numeric representations which are understandable for Machine Learning algorithms. There are diverse feature engineering methods to obtain numeric feature vectors from unstructured text. Without text pre-processing and normalization, the feature engineering methods will not process efficiently. Because of this, it is essential to preprocess textual documents before applying feature engineering methods. Following are some of the popular pre-processing techniques:

- Text tokenization and lower casing
- Removing special characters
- Contraction expansion
- Removing stopwords
- Correcting spellings
- Stemming
- Lemmatization

Figure 3.5 shows a sample text corpus. There are six different documents and two different categories.

	Document	Category
0	The sky is blue and beautiful.	weather
1	Love this blue and beautiful sky!	weather
2	The quick brown fox jumps over the lazy dog.	animals
3	The brown fox is quick and the blue dog is lazy!	animals
4	The sky is very blue and the sky is very beaut...	weather
5	The dog is lazy but the brown fox is quick!	animals

Figure 3.5. Sample Text Corpus (Collection of Text Documents).

Figure 3.6 shows normalized corpus here by lowercasing, removing special characters, tokenizing, and removing stopwords. Now, corpus is ready for feature engineering.

```
['sky blue beautiful', 'love blue beautiful sky',
 'quick brown fox jumps lazy dog', 'brown fox quick blue dog lazy',
 'sky blue sky beautiful today', 'dog lazy brown fox quick'],
```

Figure 3.6. Pre-processed Corpus.

The Bag of Words Model is a technique of representing text data. It describes the occurrence of words within a document here by vectorizing features from unstructured text data. The main basis of this model is to transform text documents into numeric vectors. The size of each vector is N where N indicates all possible distinct words in the collection of text documents. Each text document is a numeric vector of size N. Values in the vector represent the frequency of each word in that specific document. Figure 3.7 represents sample bag of words model.

	beautiful	blue	brown	dog	fox	jumps	lazy	love	quick	sky	today
0	1	1	0	0	0	0	0	0	0	1	0
1	1	1	0	0	0	0	0	1	0	1	0
2	0	0	1	1	1	1	1	0	1	0	0
3	0	1	1	1	1	0	1	0	1	0	0
4	1	1	0	0	0	0	0	0	0	2	1
5	0	0	1	1	1	0	1	0	1	0	0

Figure 3.7. Bag of Words Model for Sample Corpus.

N-gram is essentially a group of words in a text document such that these words are contiguous and occur in a sequence. Bi-grams represent n-grams of order 2 (two words), Tri-grams represent n-grams of order 3 (three words), and so on. The bag of words model can be widened to use a bag of n-grams model to obtain n-gram based feature vectors. Figure 3.8 represents sample bag of bi-gram model.

	beautiful sky	beautiful today	blue beautiful	blue dog	blue sky	brown fox	dog lazy	fox jumps	fox quick	jumps lazy	lazy brown	lazy dog	love blue	quick blue	quick brown	sky beautiful	sky blue
0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1
1	1	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0
2	0	0	0	0	0	1	0	1	0	1	0	1	0	0	1	0	0
3	0	0	0	1	0	1	1	0	1	0	0	0	0	1	0	0	0
4	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	1	1
5	0	0	0	0	0	1	1	0	1	0	1	0	0	0	0	0	0

Figure 3.8. Bag of Bi-Gram Model for Sample Corpus.

When Bag of Words model has a large corpus, there might be some terms which occur frequently across all documents and these may cause to shade other terms in the feature set. The TF-IDF(Term Frequency-Inverse Document Frequency) model tries to solve this problem by using a scaling or normalizing factor. It uses two metrics which called term frequency (tf) and inverse document frequency (idf). Figure 3.9 shows TF-IDF based Matrix for Sample Corpus.

	beautiful	blue	brown	dog	fox	jumps	lazy	love	quick	sky	today
0	0.60	0.52	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.60	0.00
1	0.46	0.39	0.00	0.00	0.00	0.00	0.00	0.66	0.00	0.46	0.00
2	0.00	0.00	0.38	0.38	0.38	0.54	0.38	0.00	0.38	0.00	0.00
3	0.00	0.36	0.42	0.42	0.42	0.00	0.42	0.00	0.42	0.00	0.00
4	0.36	0.31	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.72	0.52
5	0.00	0.00	0.45	0.45	0.45	0.00	0.45	0.00	0.45	0.00	0.00

Figure 3.9. TF-IDF based Matrix for Sample Corpus.

### 3.3. Data Mining

Data mining is a subfield of computer science that consists the operation of exploring fascinating and perceptive patterns in huge datasets using data analysis and uncovering algorithms. The main goal is exploring a special subset of patterns within given data. Since area of patterns is infinite, the process is highly hinges on declarations of restrictions. These restrictions could be incorporated either by working on a sub-domain of information or by stating some sort of threshold to reduce the work space at hand. The main goals of data mining can be briefly specified as description and prediction. Prediction concentrates on using features that exist within a data set in order

to predict future or unknown values of variables. Description concentrates on exploring new patterns that defines data in a way that has not been yet stated, which enables approaching the data from different perspectives. Since these two goals complement each other, many models use a combination of them, meaning some of the descriptive models also offer prediction to a degree, and some of the predictive models also offer an comprehensible description of the data. The difference and balance between two approaches, however, is significant when specifying a exploration goal. After specifying a exploration goal, various alternative data mining methods could be used to attain it. The selected exploration goal highly depends on the scope and the application domain.

Data mining is an interdisciplinary area that is closely associated with statistics, natural language processing, database systems, big data, machine learning and artificial intelligence and considered an important stage in knowledge discovery area, the comprehensive operation of extracting useful patterns and models from data at hand. Some of the main application areas of knowledge discovery includes customer relationship management, marketing, recommender systems, finance, manufacturing, telecommunications. Figure 3.10 shows an overview of the steps of knowledge discovery that follow selection of a discovery goal which can be briefly listed as below:

- Selection: Selecting data that is relevant to the scope of the analysis task (the discovery goal) at hand.
- Preprocessing: Combining separate data sources if there are any, and removing noisy and inconsistent data.
- Transformation: Transforming data into a suitable form to perform mining and evaluation tasks.
- Data mining: Choosing a suitable data mining algorithm and applying it on the working space.
- Interpretation/Evaluation: Interpreting the patterns into domain knowledge and translating the extracted patterns into human understandable, relevant terms.

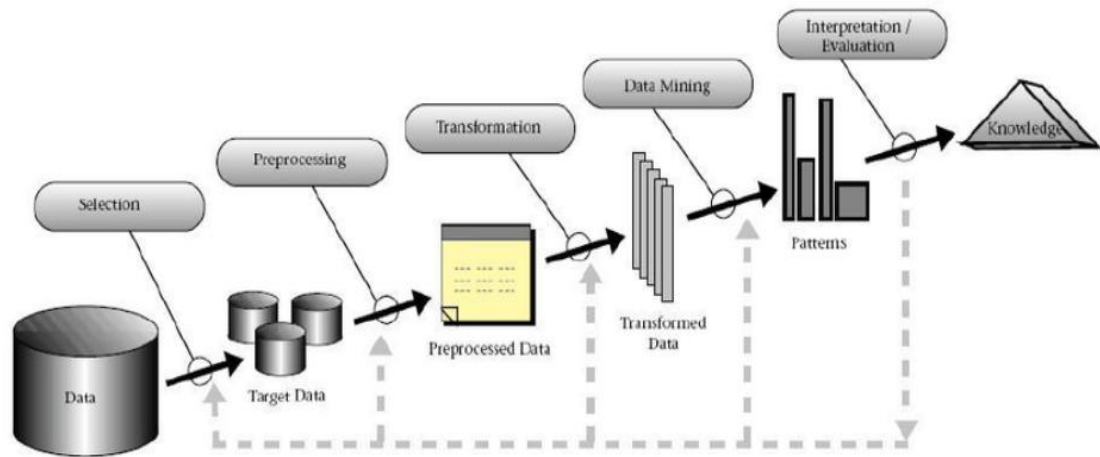


Figure 3.10. Phases of knowledge discovery.

### 3.3.1. Functions and Application Fields of Data Mining

(Han, Kamber, & Pei, 2000) defines data mining as the operation that makes it possible users to examine data from different perspectives and to categorize and summarize the relationships that are identified during the mining operation. (Piatetsky-Shapiro, 1996) analyzes data mining with having six basic functions: Classification, Regression, Clustering, Summarization, Dependency Modeling and Change (deviation) detection.

Classification is the phase that maps a specific item into one of the predefined classes (Hand, 1981; Weiss & Kulikowski, 1991). Financial market trend analysis and automatic detection of objects of interest within a large image database (Fayyad, Djorgovski, & Weir, 1996) can be listed as examples that use classification methods as part of a larger knowledge discovery application. Figure 3.11 represents a simple linear partitioning on a loan dataset, as seen in (Fayyad, Piatetsky-Shapiro, & Smyth, 1996).

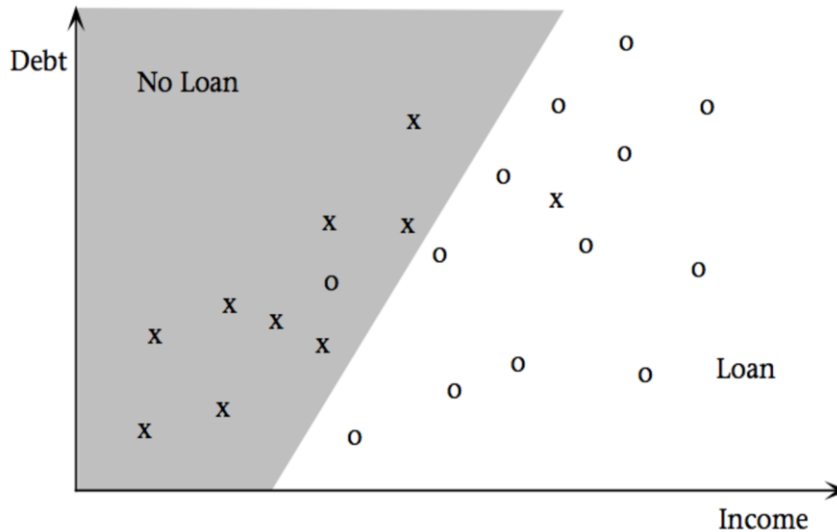


Figure 3.11. Simple linear partitioning on loan dataset.  
 (Source: Fayyad, Piatetsky-Shapiro, & Smyth, 1996)

Regression is the procedure of associating a data object with a real valued prediction item. Some samples of regression applications are predicting customer request for a new service, estimating the probability of survival according to the results of a patient's diagnosis, prediction of the quantity of biomass using remotely sensed microwave measurements. Figure 3.12 represents a simple linear regression where total debt is linear function of income as described in (Fayyad, Piatetsky-Shapiro, & Smyth, 1996). Due to the weak correlation between the two variables of choice, fit is poor.

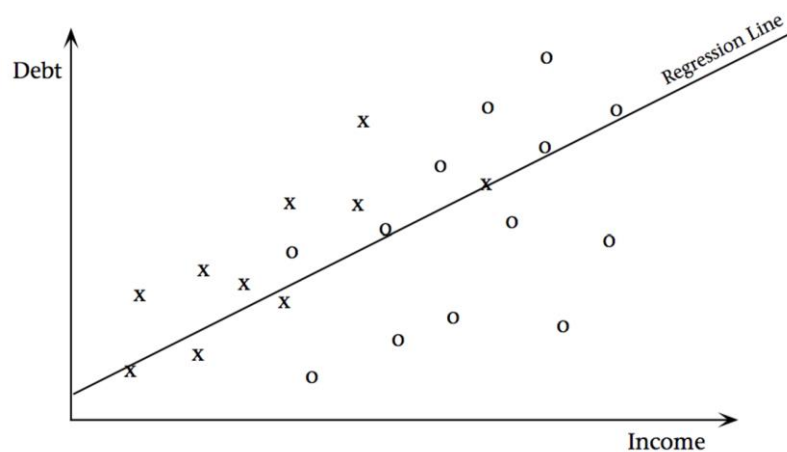


Figure 3.12. Simple linear regression where total debt is a linear function of income.  
 (Source: Fayyad, Piatetsky-Shapiro, & Smyth, 1996)

Clustering can be defined as detecting a finite set of categories within a given data. It is a common descriptive task in data mining (Jain & Dubes, 1988; Titterington, Smith, & Makov, 1985). Figure 3.13 shows a sample representation of data clustering over the loan dataset can be seen as described in (Fayyad, Piatetsky-Shapiro, & Smyth, 1996). Exploring subpopulations of customers within a marketing dataset and defining subcategories of spectra from infrared sky measurement (Cheeseman & Stutz, 1996) are some samples of clustering applications within knowledge discovery applications. Process of probability density estimation is the another task which is related to clustering and used for defining methods for estimation of the joint multivariate probability density function of all the variables or fields within a database (Silverman, 2018).

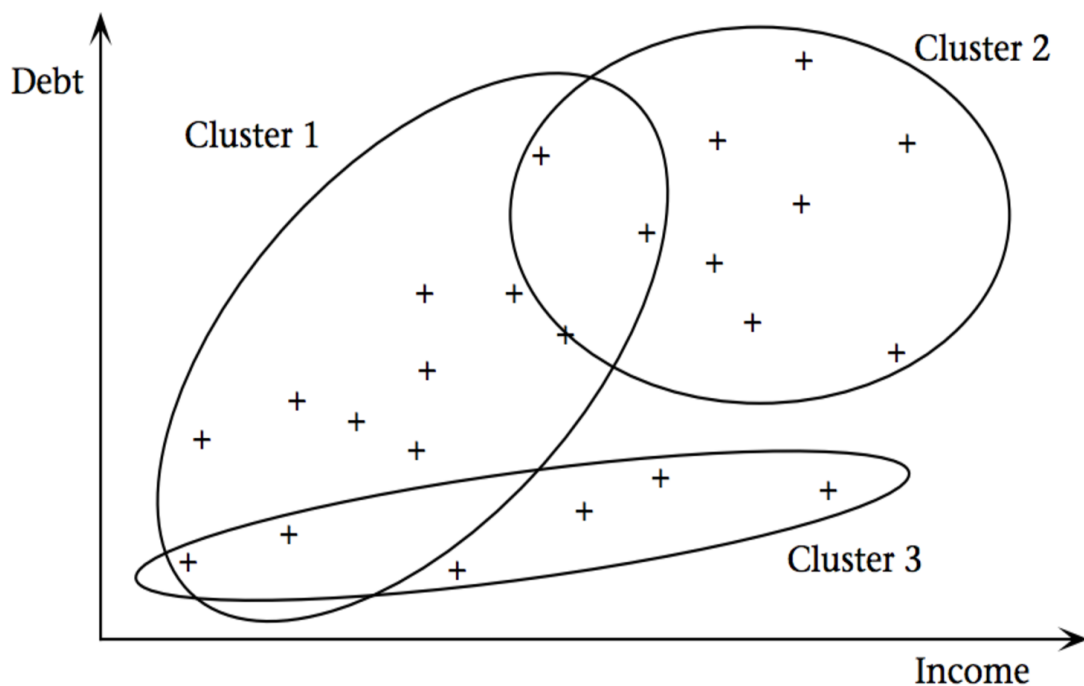


Figure 3.13. An example clustering visualization over loan dataset.

(Source: Fayyad, Piatetsky-Shapiro, & Smyth, 1996)

Summarization is the process of representing given data within a compact expression. Tabulating the standard and mean deviations for all fields within a given data set is an example of summarization operation. More advanced techniques include deriving some summary rules (Agrawal et al., 1996), multivariate visualization techniques and the discovery of functional relationships between variables (Zembowicz



& Żytkow, 1996). These methods are frequently used with the domains of exploratory data analysis and automated report generation.

Dependency Modeling is the process of discovering a model that identifies dependencies between variables. Dependency models can be listed under two different levels as below:

- The structure level: Variables are locally dependent on each other and are frequently represented in a graphical form.
- The quantitative level: It also identifies coupled dependencies using numerical values.

Probabilistic dependency networks use conditional independence to identify the model and the correlations in a structural way to specify how strong each dependency is (Glymour, Scheines, Spirtes, & Kelly 1987; Heckerman, 1996). Probabilistic dependency networks often discover applications such as medical expert systems development that borrow probabilistic analysis approaches, information retrieval and modeling of the human genome (Fayyad, Piatetsky-Shapiro, & Smyth, 1996).

Change/deviation detection is the process that concentrates on exploring the most significant changes from previously measured or normative values (Basseville & Nikiforov, 1993; Berndt, 1996; Guyon et al., 1996; Klösgen, 1996; Matheus et al., 1996) within a given dataset.

### **3.3.2. The CRISP - DM Methodology**

The CRISP-DM model means Cross Industry Standard Process for Data Mining. The CRISP-DM concept is defined in terms of a gradual process model, takes place from sets of tasks defined at four levels of abstraction: phase, generic task, specialized task, and process instance (Chapman et al., 1999). Figure 3.14 shows four levels of abstraction.

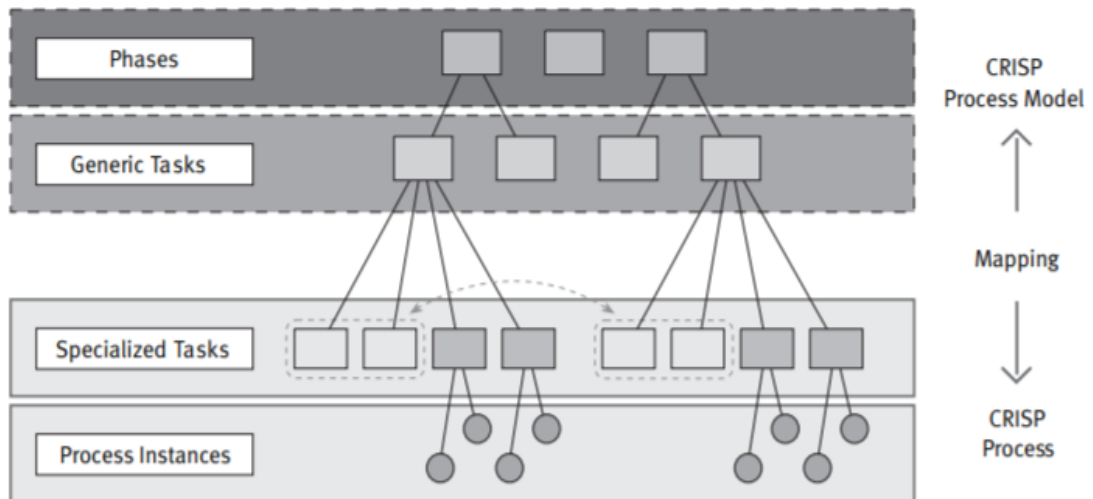


Figure 3.14. Four level abstraction of CRISP-DM concept.

(Source: Chapman et al., 1999)

At the first level, the data mining task contains phases; each phase consists of a few second-level generic tasks. This second level is named generic because it is aimed to be general to contain all possible data mining cases. This level includes entire process of data mining and all potential data mining applications. The model should be available for unpredicted developments like new modeling methods. The specialized task level is the phase to define how actions in the generic tasks should be implemented in particular cases. The process instance level is a record of the practices, resolutions, and conclusion of a main data mining task. A process instance is arranged according to the higher level tasks, but symbolizes what actually happened in a particular task, rather than what happens in general.

## CHAPTER 4

### PROPOSED METHOD

As mentioned in introduction chapter, the organization network is fed by news in a very large amount. People, who read and interpret organization related news put a lot of effort to find shareable news for their organization. For this reason, people may miss important news. Some potential opportunities in the news can be overlooked.

In this chapter content-based proposed supervised learning method for predicting the shareability of news in organization network is described. The problem of sharing prediction is modeled as a binary classification problem. To formulate the binary classification problem, it is need to define what exactly constitutes shareable and not shareable content. In this work different techniques are considered to define the shareable news and different experiments are performed in order to also obtain the best possible method for classification problem.

The rest of this chapter is organized as follows. The prediction challenges are reviewed and why it is important to predict shareable news in organization is explained. The overall architecture of proposed learning-based system is then described. The next section introduces the classifiers. Finally, feature representation, hashing and selection methods are introduced.

#### 4.1. Prediction Challenges

Successful prediction can provide the most relevant contents to users and improve user's experience. A key challenge of news websites is to help users find the articles that are interesting to read (Liu, Dolan, & Pedersen, 2010). The ability to predict sharable news for users in organizations is quite important. The meaning of the shareable news here is the news which makes it possible to take strategic decisions by users for their organizations as mentioned before. If a news is shared by a user it means that it is an important news for organization. For the purpose of predicting important news for organization, prediction model must be trained with correct features and

training data. In another words finding the features that are able to distinguish important news from those which are not, is quite important. On the other hand, the unshared news are much more than shared news in organization network. This situation is an another prediction challenge for this study and so it's hard to develop accurate model.

## 4.2. Model Architecture

Proposed approach to news prediction is based on a feature-based classification model in which a set of features is represented by news tags and classified as shareable/not shareable classes. It will be interesting to see to what extent the shareability of news in organization network can be predicted. In other words research interest reduces to a binary classification problem in which a news will be assigned to a shareable (positive) or not shareable (negative) class. In this section the overall architecture of study which includes different components is introduce.

The features that are extracted from the data and the selection of classifier is two important decision for supervised learning systems. Figure 4.1 illustrates the overall architecture of proposed model. The model then extracts several features from news' tags and different machine learning approaches are used to train classifiers. The classifier is then used to predict whether a news will be shareable or not.

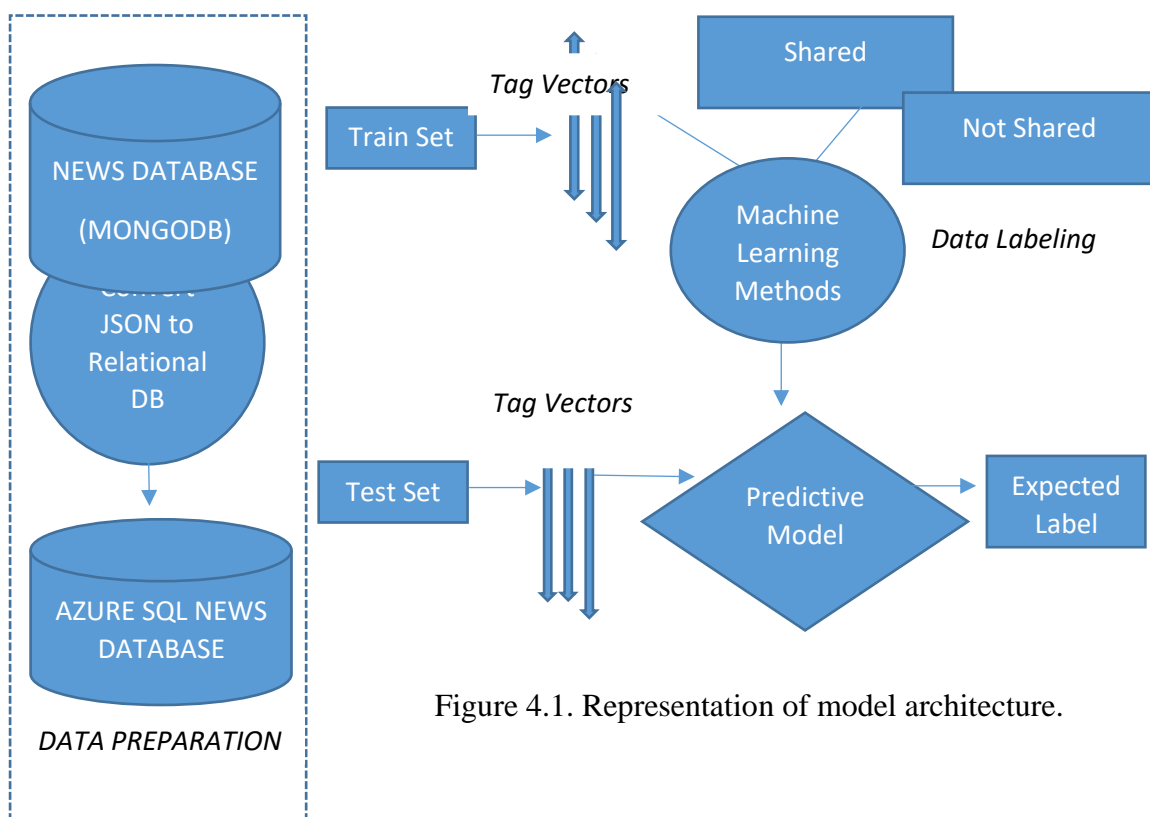


Figure 4.1. Representation of model architecture.

### **4.3. Classification Methods For Predictive Model**

The operation of classification of data can be achieved with the help of a few different classifiers. The machine learning community has introduced several feature-based classifiers which are suitable for different applications. Several classifiers can be used for a prediction task according as the features and structure of the data.

Depending on the nature of the data, features and desired performance and complexity different models can be trained. In this study Support Vector Machine, Naive Bayes, Neural Network and Decision Forest classifiers were used. In the next chapter the performance of each method will be experimentally show and the optimal model for problem is introduce.

#### **4.3.1. Support Vector Machine Classifier**

Support Vector Machine is a supervised machine learning algorithm that can be employed for both classification and regression purposes and commonly used in classification problems. This classification method is based on the idea of finding a hyperplane that divides a dataset into two classes as shown in Figure 4.2. Support vectors are the data points nearest to the hyperplane. The operation of the SVM algorithm is based on finding the hyperplane that gives the largest minimum distance to the training examples. Distance is called margin and the optimal separating hyperplane maximizes the margin of the training data. The aim is to choose a hyperplane with the greatest possible margin between the hyperplane and any point within the training set, giving a greater chance of new data being classified correctly.

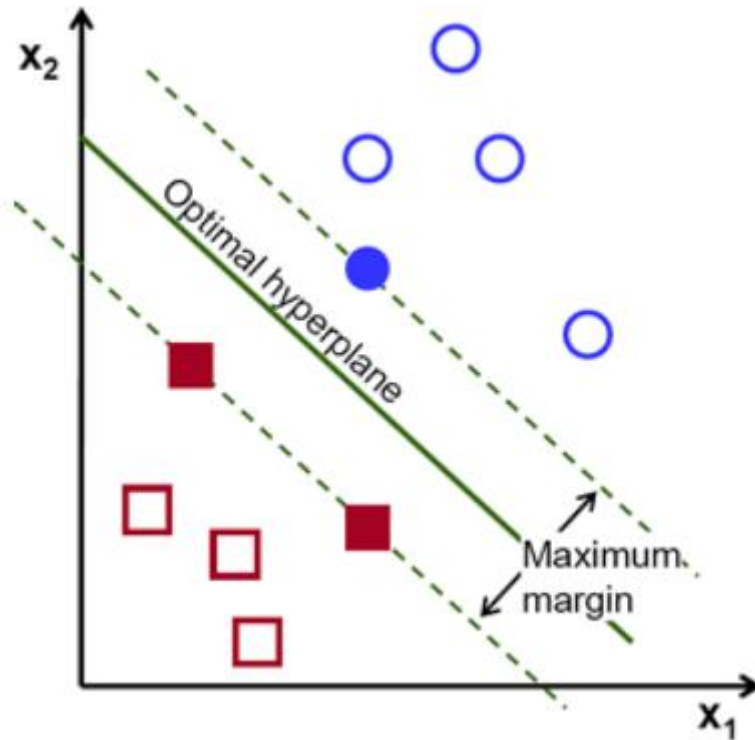


Figure 4.2. Representation of SVM hyperplane.

### 4.3.2. Naive Bayes Classifier

The Bayesian Classification represents a supervised learning method as well as a statistical method for classification. Assumes an underlying probabilistic model and it allows to capture uncertainty about the model in a principled way by determining probabilities of the outcomes. It uses Bayes' theorem to classify objects. Naive Bayes classifiers assume strong, or naive, independence between attributes of data points. Popular uses of naive Bayes classifiers include spam filters, text analysis and medical diagnosis.

### 4.3.3. Neural Network Classifier

Neural network classifier creates a binary classifier by using a neural network algorithm. Classification using neural networks is a supervised learning method, and requires a tagged dataset, which includes a label column. A neural network contains interconnected layers. The inputs are the first layer, and they are connected to an output

layer by an acyclic graph comprised of weighted edges and nodes. Between the input and output layers, there are multiple hidden layers. Most predictive tasks can be accomplished easily with only one or a few hidden layers.

The relationship between inputs and outputs is learned from training the neural network on the input data. The direction of the graph proceeds from the inputs through the hidden layer and to the output layer. All nodes in a layer are connected by the weighted edges to nodes in the next layer. To compute the output of the network for a particular input, a value is calculated at each node in the hidden layers and in the output layer. The value is set by calculating the weighted sum of the values of the nodes from the previous layer. An activation function is then applied to that weighted sum.

#### **4.3.4. Decision Forest Classifier**

Decision Forest classifier is an ensemble learning classifier which creates multiple related models and combines them. There are many ways to create individual models and combine them in an ensemble. Implementation of decision forest works by building multiple decision trees and then voting on the most popular output class. Voting is one of the methods for generating results in an ensemble model.

Decision Forest Classifier algorithm creates many individual classification trees by using the entire dataset. Each tree in the decision forest tree outputs a non-normalized frequency histogram of labels. The aggregation process sums these histograms and normalizes the result to get the probabilities for each label. The trees that have high prediction confidence will have a greater weight in the final decision of the ensemble.

#### **4.4. Feature Representation, Hashing and Selection**

An important factor when developing a prediction model is to represent samples with a good set of features. Good features should be illuminating and should have differential power. It means that the features should be able to discriminate between the news that will be shared and those which will not.

In this study the tags of news are used as raw input data. A set of features are represented by these tags for the purpose of converting these textual representations into

numeric representations which are understandable for Machine Learning algorithms. For this purpose, first of all the N-Gram feature extraction technique was used to obtain bigram tags as shown in figure 4.3.






	Id	Ngram	DF	IDF
view as 				
	1	rusya_ayakkabi	6	4.124939
	2	kriz_pazar	6	4.124939
	3	pazar_der	6	4.124939
	4	istanbul_ceket	6	4.124939
	5	ceket_vergi	6	4.124939
	6	germany_fresh	6	4.124939
	7	arabia_new	6	4.124939
	8	tailor	6	4.124939
	9	boss_rise	6	4.124939
	10	woman_trade	6	4.124939
	11	ceo_boss	6	4.124939
	12	sheer_think	6	4.124939
	13	day_likely	6	4.124939
	14	likely_roman	6	4.124939
	15	roman_unique	6	4.124939

Figure 4.3. N-Gram Features.

Then, the bag of words model was created. After that, Term Frequency – Inverse Document Frequency (TF-IDF) values were calculated for each bigram tag as shown in figure 4.4. TF-IDF method mostly used on text articles. We used this technique to score the importance of a tag which belongs to a news based on how often did it appears in that news’ tag corpus and a given collection of news tag corpuses.









STATUS	Preprocessed TAG. [azerbaijan_montenegro]	Preprocessed TAG. [montenegro_construction]	Preprocessed TAG. [military_denmark]	Preprocessed TAG. [denmark_russia]	Preprocessed TAG. [party_finland]	Preprocessed TAG. [energy_iceland]	Preprocessed TAG. [sweden_export]
							
0	7.772113	11.65817	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	7.772113	11.65817	7.772113	11.65817	11.65817
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Figure 4.4. TF- IDF values for each bigram tag.



The problem with bag of words model is that such dictionaries take up a large amount of storage space and grow in size as the training set grows. As an alternative, the machine learning framework called Vowpal Wabbit was also used as feature hashing method. This method hashes news tags into in-memory indexes, using a open source hash function called murmurhash3. This hash function is a non-cryptographic hashing algorithm that maps news' tags to integers. The purpose of hashing is to convert variable-length tags into equal-length numeric feature vectors, to support dimensionality reduction and make the lookup of feature weights faster.

Each hashing feature represents one or more n-gram tags, depending on the number of bits (represented as k) and depending on the number of n-grams. It projects tag names to the machine architecture unsigned word using the murmurhash3 algorithm which then is AND-ed with  $(2^k)-1$ . That is, the hashed value is projected down to the first k lower-order bits, and the remaining bits are zeroed out. If the specified number of bits is 14, the hash table can hold  $(2^{14})-1$  entries.

After the dictionary has been built, the feature hashing method converts the dictionary terms into hash values, and computes whether a feature was used in each case. For each row of tag data, the module outputs a set of columns, one column for each hashed bigram tag as shown in figure 4.5.




rows	columns			
100000	131074			
		<b>status_label</b>	tag_text	tag_text_HashingFeature_1
		view as 		
		0	azerbajjan montenegro construction eu energy markets turkey trend economy georgia gas pipeline management	0
		0	ipo ro ro turkey military denmark russia	0

Figure 4.5. Hashed Features.

If the value in the column is 0, the row did not contain the hashed feature. If the value is 1, the row contains the feature.

After representing and hashing features, Chi Square feature selection method was used for the purpose of training model classifiers. Chi Square test is used in statistics to test the independence of two events. Given dataset about two events, observed count and the expected count can be obtained. Chi Square Score measures how much the expected counts and observed count deviate from each other. In feature selection, the two events are occurrence of the feature and occurrence of the class. In other words, the goal is to test whether the occurrence of a specific feature and the occurrence of a specific class are independent.

If the two events are dependent, the occurrence of the feature can be used to predict the occurrence of the class. The aim is to select the features, of which the occurrence is highly dependent on the occurrence of the class. When the two events are independent, the observed count is close to the expected count, thus a small chi square score. So a high value of chi square indicates that the hypothesis of independence is incorrect. In other words, the higher value of the chi square score, the more likelihood the feature is correlated with the class, thus it should be selected for model training.

## CHAPTER 5

### EXPERIMENTAL RESULTS

In this chapter collection of data and experimental study is explained. Dataset was collected from an organization's database and different experiments were performed on it to see what extent the shareability of the news could be predicted in an organization network. It is further explained that collection of data and splitting it for training and testing. Then the evaluation metrics that is used in this study is introduced. Finally, performance of different classifiers is compared.

#### 5.1. Preparing Dataset

In this study, dataset was collected from an organization's news sharing network. News are provided from a third party system to this organization network. On average 1500 news are transmitted per day to this network with their tags. As mentioned before, in organization network, users read these news and they add new tags to them if they prefer. After that if they think current news is valuable for organization, they share this news with other users.

News and tags are stored in MongoDB document database in JSON format. As mentioned before, predictive experiment runs on Microsoft Azure Machine Learning Studio Workspace. To achieve high performance from experiment, data and experiment should be in the same environment. For this reason, it is necessary to transfer data from MongoDB document database to Microsoft Azure SQL database. To accomplish this, data was retrieved by NoSQL queries from MongoDB. Then, it was transformed into SQL insert scripts by using a MongoDB tool. Finally, these scripts were executed and SQL table which named News was created. After table was created data was inserted in to this table. Totally there are 100,000 news in dataset and 3,236 of them were shared by users and remaining 96,764 of them were not shared by them. Figure 5.1. shows sample News' Tags.

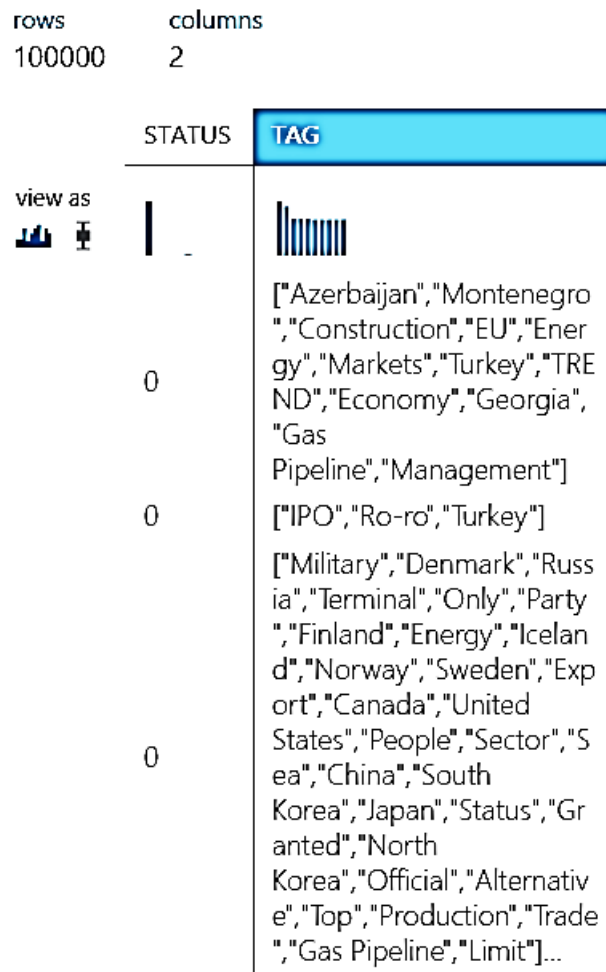


Figure 5.1. Sample News' Tags.

The statistics and information about tags are shown in table 5.1. There are 100,000 news and 1,212,786 tags in dataset. On average, 20 news are shared per day in this organization network.

Table 5.1. Statistics and information about tags.

Total Number of Tags	Average Number of Tags	Number of distinct tags	Least common tag	Most common tag
1,212,786	12	2154	KibarHolding	Türkiye

Türkiye, Iran, Ticaret and Ihracat are the most common tags in dataset. Figure 5.2 represents count of the occurrences.

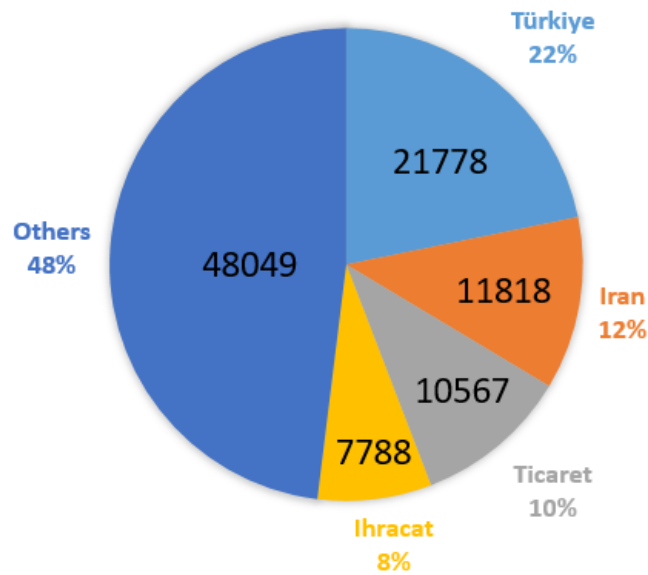


Figure 5.2. Distribution of Tags.

The tag Türkiye takes part in 81 percent of shared news as represented in figure 5.3. It means that this tag has an major influence on sharing the news.

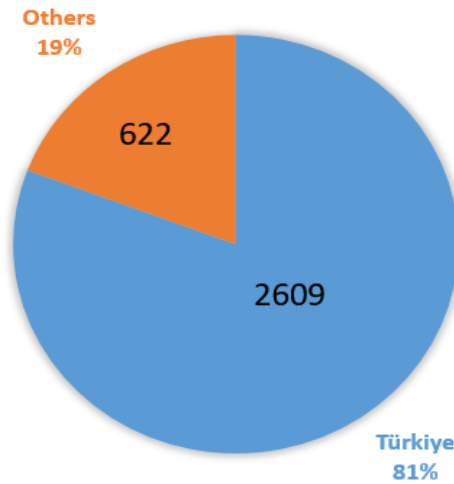


Figure 5.3. Distribution of #Türkiye in shared news.

The tags Türkiye and Ihracat takes part in together in 53 percent of shared news as shown in Figure 5.4. The coexistence of these two tags makes a news shareable.

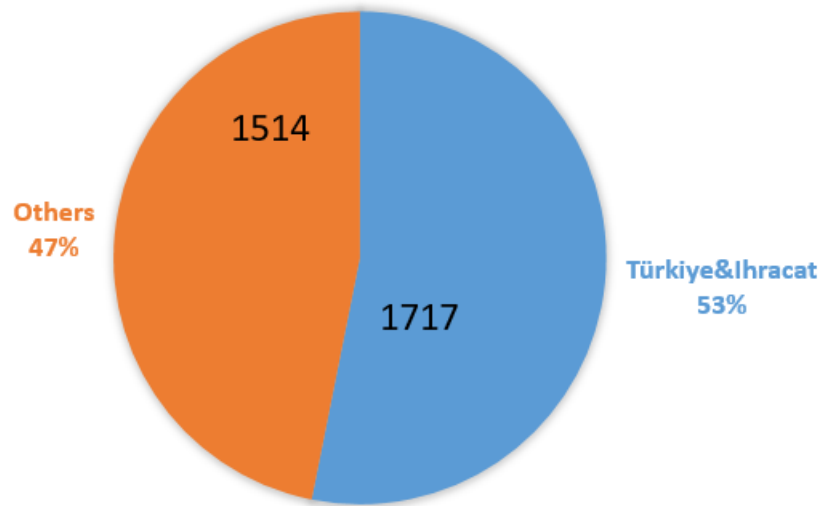


Figure 5.4. Percent of shared news that consist of both #Türkiye and #Ihracat.

## 5.2. Data Preprocessing

Tags were preprocessed by using below methods. Figure 5.5 represents a sample tag preprocess.

- Removing stop words
- Removing special characters
- Normalazing case to lowercase
- Removing numbers
- Removing duplicate characters
- Splitting tokens on special characters
- Lemmatization

STATUS	TAG	Preprocessed TAG
0	["Azerbaijan","Montenegro","Construction","EU","Energy","Markets","Turkey","TRENND","Economy","Georgia","Gas Pipeline","Management"]	" azerbaijan " montenegro " construction " eu " energy " market " turkey " trend " economy " georgia " gas pipeline " management "

Figure 5.5. Sample text preprocessing.

### 5.3. Splitting Method of Dataset

Splitting data into training and testing sets is an significant task of evaluating data mining models. The machine learning pipeline uses the training data to train models to understand patterns, and uses the testing data to score the predictive efficiency of the trained model. It scores predictive efficiency by comparing predictions on the test data set with true values using a variety of metrics.

In order to test the performance of classifiers, dataset is splitted for training and testing. In this study, K-folds Cross Validation strategy is used. This strategy is widely used in machine learning domain. In this strategy, data is splitted in to K different subsets (folds). k-1 folds are used for training and the remaining fold k is used for testing. This process is repeated until all folds are tested. This is performed as per the following steps:

1. Split the data set into k equal subsets. Each subset is named a fold. Let the folds be named as  $f_1, f_2, \dots, f_n$ .
2. For  $j = 1$  to  $j = n$ 
  - Take the fold  $f_j$  as test set and take all the remaining k-1 folds in the training set.
  - Train model using the training set and calculate the accuracy of model by validating the predicted outcomes against the test set.
3. Calculate the accuracy of model by averaging the accuracies obtained in all the k cases of cross validation.

In the k-fold cross validation technique, all the records in the original training data set are used for both training and testing. Also, each record is used for testing just once. Usually, the value of k is taken to be 10, but it is not a exact principle, and k can take any value. Figure 5.6 represents a 10-fold cross validation.

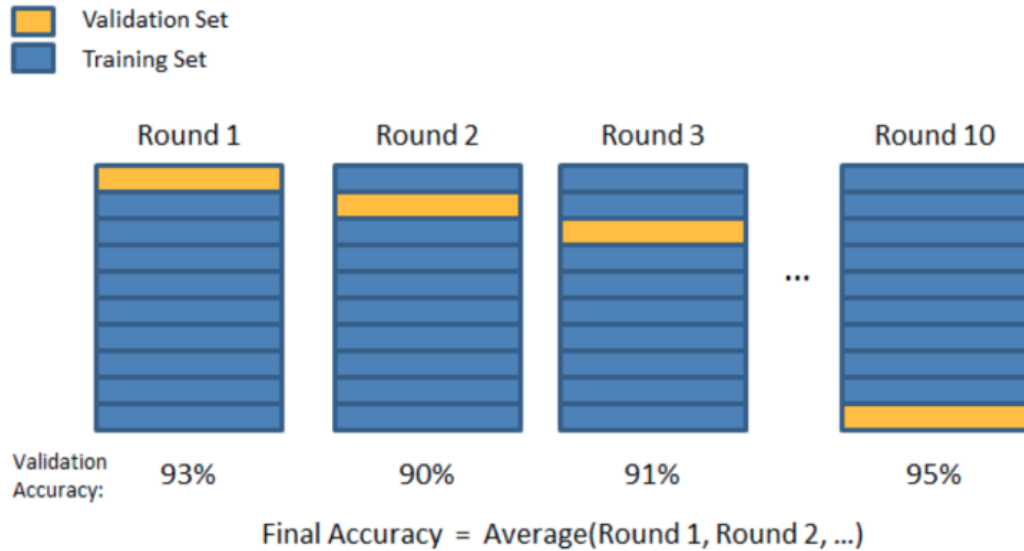


Figure 5.6. 10-fold Cross Validation.

## 5.4. Evaluation Metrics

As mentioned before, the aim of this study is to predict the shareability of news. It is important to explain what the good prediction means and how it can be measured. The measurement techniques adapted for classification efficiency is important for design and selecting classifiers, particularly when faced with an imbalanced dataset. To measure the efficiency of classifiers various metrics can be used. Depending on the aim of study the different selections for evaluation metric can be made. The general evaluation metrics are described below:

True Positive Rate (TPR) or Recall, evaluates to what extent all the examples that needed to be classified are to be considered as positive. If a positive sample is predicted as positive, it is supposed as a true positive.

$$\text{TPR} = \text{Sensitivity} = \frac{\text{true positives}}{\text{true positives} + \text{false negatives}} \quad (5-1)$$

True Negative Rate (TNR) (Recall -) is the percentage of negative samples correctly predicted within negative class. If a negative sample is predicted as negative, then it is supposed as a true negative.

$$\text{TNR} = \text{Specificity} = \frac{\text{true negative}}{\text{true negative} + \text{false positive}} \quad (5-2)$$



False Positive Rate (FPR) is the percentage of negative samples mispredicted as belonging to the positive class.

$$\text{FPR} = 1 - \text{Specificity} = \frac{\text{false positives}}{\text{false positives} + \text{true negatives}} \quad (5-3)$$

Figure 5.7 illustrates confusion matrix which represents TP, TN, FP and FN metrics.

		Actual	
		Positive	Negative
Predicted	Positive	True Positive	False Positive
	Negative	False Negative	True Negative

Figure 5.7. Confusion Matrix.

Accuracy is simply a proportion of correctly predicted observation to the aggregate observations. Accuracy is a significant measure when there are balanced datasets where values of false positive and false negatives are almost same.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{FN} + \text{TN}} \quad (5-4)$$

Precision is the proportion of correctly predicted positive samples to the total predicted positive samples.

$$\text{Precision} = \frac{\text{true positives}}{\text{true positives} + \text{false positives}} \quad (5-5)$$

F1 Score is the weighted average of Precision and Recall. For this reason, this score is related with false positives and false negatives. F1 is usually more helpful than accuracy, particularly if there is an unbalanced class distribution. Accuracy works better if false positives and false negatives have same cost. If the cost of false positives and false negatives are very dissimilar, it's better to analyze both Precision and Recall.

$$\text{F1 Score} = 2 * (\text{Recall} * \text{Precision}) / (\text{Recall} + \text{Precision}) \quad (5-6)$$

Receiver Operating Characteristic (ROC) Curve is a generally used way to visualize the efficiency of a binary classifier. It is a plot of the true positive ratio against the false positive ratio. A ROC plot shows the relation between specificity and sensitivity. It also represents the test accuracy. The closer the graph is to the top and left-hand borders, the more accurate the test. Likewise, the closer the graph to the diagonal, the less accurate the test. A perfect test would go straight from zero up the top-left corner and then straight across the horizontal. Test accuracy is also shown as the area under the curve. The greater the area under the curve, the more accurate the test. A perfect test has an area under the ROC curve (AUROCC) of 1. Figure 5.8 illustrates a sample ROC curve for SVM classifier that is used in this work.

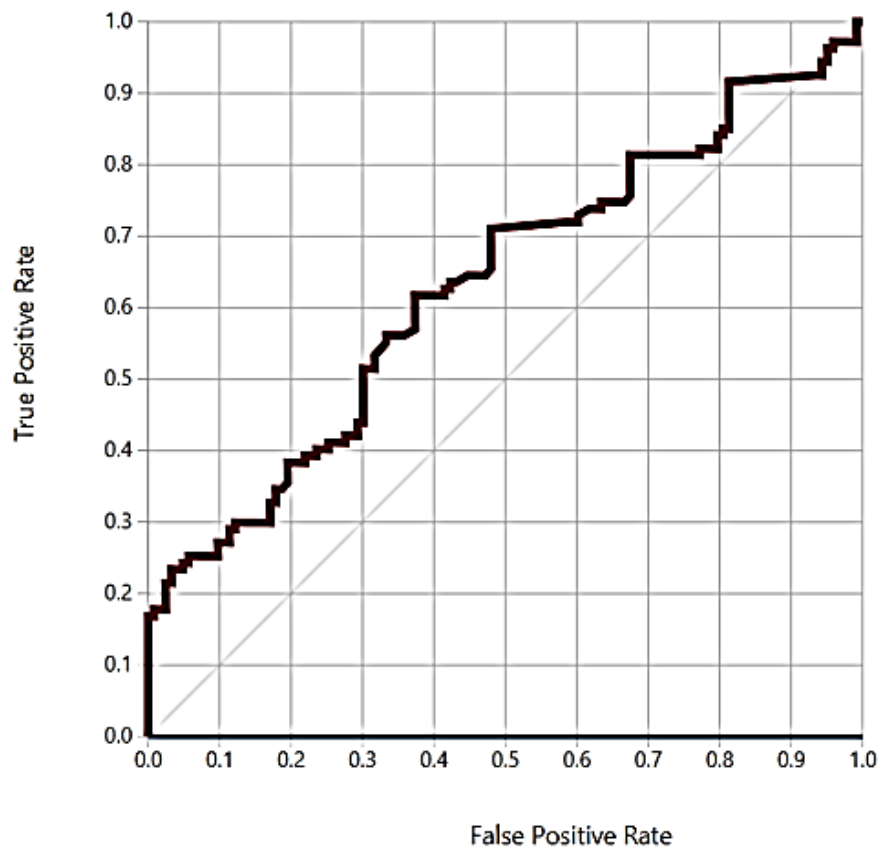


Figure 5.8. Sample ROC curve for SVM classifier.

As mentioned earlier the selection of evaluation metric is quite important to properly evaluate the efficiency of models. In this study however, the classes are highly

imbalanced, that is, the distribution of positive and negative samples are different, negative samples are much more. It is more important for us to predicting the news as shared which are actually shared by users. So, we are intended to find positive samples as much as possible, that means more true positive samples and less false negative samples.

A fine method to define a proper evaluation measure should usually depend upon particular application requirements. Selecting suitable evaluation measure according to different cases can help making correct judgment to the classification efficiency. In this thesis goal is to predict all sharable news (Positive labels). However by maximizing the TPR, FPR will also be increased, hereby it is needed to use a proper classifier which is considered a trade off and maximizes the accuracy of both positive and negative classes.

## **5.5. Classifiers and Results**

In this section the performance of classifiers is reported for each classifier separately, and then proposed classifiers are further compared using different evaluation metrics and under two feature extraction methods.

### **5.5.1. SVM**

Support vector machine classifier is a supervised learning classifier. This classifier is used in two different machine learning pipelines for the purpose of training machine learning models. The input of the classifier is the training dataset with extracted and selected features. For this classifier, Lambda value is 0,001 and the number of iterations is 1. For first pipeline, features are extracted by using Vowpal Wabbit feature hashing method and N-Gram feature extraction technique. After that features are selected by chi-squared method. The experiments are setup by doing a 10-fold cross-validation on dataset. Figure 5.9 shows evaluation metrics for this pipeline.

Fold Number	Number of examples in fold	Model	Accuracy	Precision	Recall	F-Score
0	10000	SVM (Pegasos-Linear)	0.9978	0.963855	0.969697	0.966767
1	10000	SVM (Pegasos-Linear)	0.9977	0.971154	0.955836	0.963434
2	10000	SVM (Pegasos-Linear)	0.9969	0.951456	0.948387	0.949919
3	10000	SVM (Pegasos-Linear)	0.9973	0.950495	0.96	0.955224
4	10000	SVM (Pegasos-Linear)	0.9969	0.977273	0.926154	0.951027
5	10000	SVM (Pegasos-Linear)	0.9969	0.970874	0.931677	0.950872
6	10000	SVM (Pegasos-Linear)	0.9965	0.96	0.934132	0.946889
7	10000	SVM (Pegasos-Linear)	0.9959	0.954268	0.923304	0.938531
8	10000	SVM (Pegasos-Linear)	0.9973	0.965625	0.950769	0.95814
9	10000	SVM (Pegasos-Linear)	0.9983	0.984709	0.964072	0.974281
Mean	100000	SVM (Pegasos-Linear)	0.99715	0.964971	0.946403	0.955508

Figure 5.9. Evaluation results for first pipeline.

For second pipeline features are extracted by using TF-IDF method and N-Gram feature extraction technique. After that features are selected by chi-squared method. The experiments are setup similarly by doing a 10-fold cross-validation on datasets. Figure 5.10 shows evaluation metrics for this pipeline.

Fold Number	Number of examples in fold	Model	Accuracy	Precision	Recall	F-Score
0	10000	SVM (Pegasos-Linear)	0.9692	0.57971	0.242424	0.34188
1	10000	SVM (Pegasos-Linear)	0.9713	0.607143	0.268139	0.371991
2	10000	SVM (Pegasos-Linear)	0.9715	0.582781	0.283871	0.381779
3	10000	SVM (Pegasos-Linear)	0.9711	0.536913	0.266667	0.356347
4	10000	SVM (Pegasos-Linear)	0.9711	0.640625	0.252308	0.362031
5	10000	SVM (Pegasos-Linear)	0.9717	0.630872	0.291925	0.399151
6	10000	SVM (Pegasos-Linear)	0.969	0.588235	0.239521	0.340426
7	10000	SVM (Pegasos-Linear)	0.9694	0.62406	0.244838	0.351695
8	10000	SVM (Pegasos-Linear)	0.9694	0.573643	0.227692	0.325991
9	10000	SVM (Pegasos-Linear)	0.9709	0.63354	0.305389	0.412121
Mean	100000	SVM (Pegasos-Linear)	0.97046	0.599752	0.262277	0.364341

Figure 5.10. Evaluation results for second pipeline.

### 5.5.2. Naive Bayes

Naive Bayes classifier uses a Bayesian approach to linear classification. This method efficiently approximates the optimal Bayesian average of linear classifiers by selecting one average classifier. This classifier is used in two different machine learning pipelines for the purpose of training machine learning models. For this classifier, number of training iterations is 30. Similar to the SVM, this classifier is used in two different machine learning pipelines for the purpose of training machine learning models. Table 5.2 shows evaluation metrics for machine learning pipelines.

Table 5.2. Evaluation Results for Naive Bayes.

Machine Learning Pipeline	Accuracy	Precision	Recall	F1 Score
1	0,999	0,994	0,964	0,978
2	0,972	0,642	0,271	0,382

### 5.5.3. Neural Network

As mentioned in Chapter 4, Neural Network Classifier creates a binary classifier using a neural network algorithm. The utilized network's architecture is fully connected. It has one hidden layer. The output layer is fully connected to the hidden layer as shown in figure 5.11, and the hidden layer is fully connected to the input layer. The number of nodes in the input layer equals the number of features in the training data and the hidden layer has 50 nodes. All inputs map to one of two nodes in the output layer.

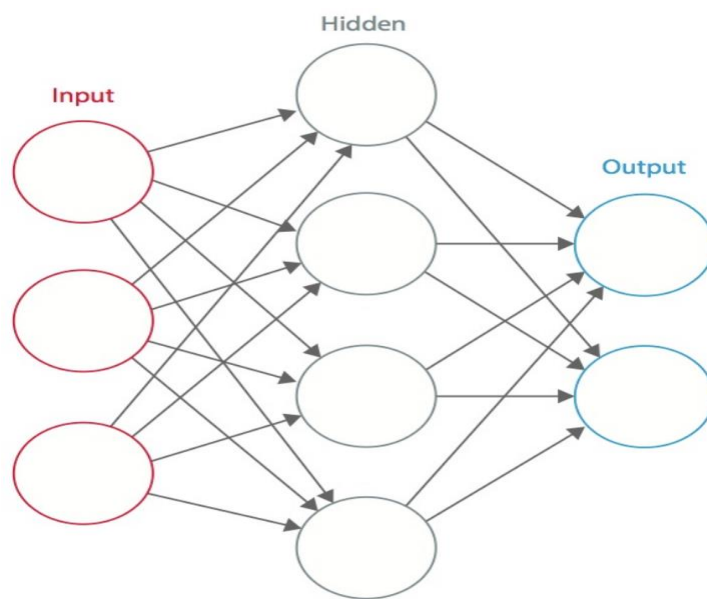


Figure 5.11. Fully Connected Neural Network.

Similar to the SVM and Naive Bayes, this classifier is used in two different machine learning pipelines for the purpose of training machine learning models. Table 5.3 shows evaluation metrics for machine learning pipelines.

Table 5.3. Evaluation Results for Neural Network.

Machine Learning Pipeline	Accuracy	Precision	Recall	F1 Score
1	0,999	0,995	0,986	0,990
2	0,984	0,850	0,350	0,495

#### 5.5.4. Decision Forest

As mentioned in previous chapter, Decision Forest Classifier creates a two-class classification model using the decision forest algorithm. The utilized Decision Forrest Classifier has 8 decision trees. Increasing the depth of the tree increases precision, at the risk of some overfitting and increased training time. The maximum depth of the decision trees is 32. Similar to the other classifiers, this classifier is used in two different machine learning pipelines for the purpose of training machine learning models. Table 5.4 shows evaluation metrics for machine learning pipelines.

Table 5.4. Evaluation Results for Decision Forest.

<b>Machine Learning Pipeline</b>	<b>Accuracy</b>	<b>Precision</b>	<b>Recall</b>	<b>F1 Score</b>
1	0,973	0,951	0,179	0,301
2	0,974	0,932	0,237	0,377

The aim of this thesis is to find all shared news. Classifiers should properly predict shareable news and sharable news should not be overlooked. It is related with true positive and false negative samples. Therefore, the recall metric plays an important role to evaluate and to compare performance of classifiers. The experiments are done based on 10-fold cross-validation. As shown in Table 5.5 the accuracy values are nearly same for all pipelines. This is most probably due to the fact that the distribution of classes are unbalanced. The number of shared news is far less than the number of unshared news. The difference is huge between recalls of pipeline-1 and pipeline-2. It means that the number of false negative samples are higher for pipeline-2. It means that the features which are extracted with Vowpal Wabbit method are more effective than the features extracted with TF-IDF method. As a result, this pipeline performs poorly on predicting shared news despite they were actually shared by users.

Table 5.5. Experimental Results for Machine Learning Pipelines.

<b>Machine Learning Pipeline</b>	<b>Feature Extraction Method</b>	<b>Classifier</b>	<b>Accuracy</b>	<b>Recall</b>	<b>Precision</b>	<b>Run Time(second)</b>
1	Vowpal Wabbit	SVM	0.997	0.946	0.964	244
2	TF - IDF	SVM	0.970	0.262	0,599	428
3	Vowpal Wabbit	Naive Bayes	0.999	0.964	0,994	504
4	TF - IDF	Naive Bayes	0.972	0.271	0,642	1483
5	Vowpal Wabbit	Neural Network	0,999	0,986	0,995	1526
6	TF - IDF	Neural Network	0,984	0,350	0,850	763
7	Vowpal Wabbit	Decision Forest	0,973	0,179	0,951	1986
8	TF - IDF	Decision Forest	0,974	0,237	0,932	876

As discussed earlier, these results can be explained by fact that the data set is highly imbalanced. Dataset has a huge number of samples labeled unshared. This means classification of samples to positive class is very difficult. Therefore, it is more challenging for classifier to learn the distribution of positive classes when the number of samples is very low. Generally it can be concluded that there is no universal best classifier for the study of sharing prediction in organization network. It is the task of the designer to select best classifiers and parameters based on the question of interest and available information.

To show that conclusions are reliable, all experiments were performed on all dataset based on a 10-fold cross-validation to make sense that the classifiers are not over-trained on a certain test set or on a certain dataset.



## CHAPTER 6

### CONCLUSION AND FUTURE WORK

#### 6.1. Conclusion

Predicting the shareability of organization related news is quite important for organization networks to make strategic decisions about events which are mentioned in news. Through the detection of important news, organizations can take action on events by early planning. In this work, we proposed a tag based statistical learning approach that extracts different features from news and tries to predict whether the news be a shareable news or not.

We experimentally tested our approach using dataset that we collected from an organization network. We converted the dataset into a relational database to make it easier to process data and extract features. We extracted different features from the tags of news. In order to see whether the shareability of news can be predicted or not, we used different statistical classifiers which are trained based on the features that we extracted. The goal of classification task is to predict whether a news can be shareable or not. We employed different classifiers and performed a set of experiments to find out the optimal classifiers for this prediction problem. We performed all experiments with cross-validation to make sure that the classification methods are not optimized for one particular test set.

Our experiment revealed that there is no general best classifier that can always perform good for the sharing prediction study. In fact, depending on the problem and available data, a certain classifier might be the best choice and the other might not. Depending on the problem and available information the designer can choose the best choice for classifier.

## 6.2. Future Work

Our work can be extended from different point of views for future studies. In this study we analyzed different classifiers independently. One interesting addition to our study would be to implement methods to combine all the classifiers and benefit from the advantage of all of them.

Another addition to our work would be to implement more feature extraction methods to see if more efficient and accurate classifiers can be trained. Similarly, more different classifiers would be implement to increase model performance. Different feature selection methods can be applied to extracted features.

Dataset can be extended. The distribution of positive and negative classes can be changed in dataset. Data transformation techniques like smooth filtering can be used. Different datasets can be used for generalizing study.

Machine learning models can be deployed as a webservice in Azure Machine Learning Studio and they can be run at specific intervals dynamically. In this way, system automatically trains itself for recent news and classify news dynamically.

As mentioned before, in this study only news' tags are used as inputs for machine learning models. Some of third party news providers also provides news articles to organization network. So, news articles can be used as inputs together with tags. In addition, news articles can be summarized using article summarization methods and after that these summarized articles may be used as features for machine learning models.

Topic modeling is a text mining approach for organizing the knowledge with different content under specific topics. It is usually being used in various areas such as library science, search engines and statistical language modeling. This approach can be used for modelling topics for news articles. Than, these topics may be used as features for machine learning models.

## REFERENCES

- Agrawal, R., Mannila, H., Srikant, R., Toivonen, H., & Verkamo, A. I. (1996). Fast discovery of association rules. *Advances in knowledge discovery and data mining*, 12(1), 307-328.
- Basseville, M., & Nikiforov, I. V. (1993). *Detection of abrupt changes: theory and application* (Vol. 104). Englewood Cliffs: Prentice Hall.
- Berndt, D. J. (1996). Finding patterns in time series: a dynamic programming approach. *Advances in knowledge discovery and data mining*, 229-248.
- Breslin, J. G., Passant, A., & Decker, S. (2009). *The social semantic web*. Springer Science & Business Media.
- Chapman, P. (1999). Julian Clinton (SPSS), Randy Kerber (NCR), Thomas Khabaza (SPSS), Thomas Reinartz (DaimlerChrysler), Colin Shearer (SPSS) and Rüdiger Wirth (DaimlerChrysler),". *CRISP-DM 1.0. Step-by-step data mining guide*.
- Chaturvedi, A. K., Peleja, F., & Freire, A. (2017). Recommender System for News Articles using Supervised Learning. *arXiv preprint arXiv:1707.00506*.
- Cheeseman, P., & Stutz, J. (1996). Bayesian classification (autoclass): Theory and results in advances in knowledge discovery and data mining eds. *Articles FALL.*, 51.
- Domingos, P. (2012). A few useful things to know about machine learning. *Communications of the ACM*, 55(10), 78-87.
- Fayyad, U. M., Djorgovski, S. G., & Weir, N. (1996). From digitized images to online catalogs data mining a sky survey. *AI magazine*, 17(2), 51.
- Fayyad, U., Piatetsky-Shapiro, G., & Smyth, P. (1996). From data mining to knowledge discovery in databases. *AI magazine*, 17(3), 37.
- Glymour, C., Scheines, R., Spirtes, P., & Kelly, K. (1987). Discovering causal structure: Artificial intelligence. *Philosophy of science, and Statistical Modeling*, 205-212.
- Granovetter, M. (1978). Threshold models of collective behavior. *American journal of sociology*, 83(6), 1420-1443.
- Guyon, I., Matic, N., & Vapnik, V. (1996). *Discovering Informative Patterns and Data Cleaning*.
- Han, J., Kamber, M., & Pei, J. (2000). *Data mining: concepts and techniques* (the Morgan Kaufmann Series in data management systems). *Morgan Kaufmann*.
- Hand, D. J. (1981). *Discrimination and classification*. *Wiley Series in Probability and Mathematical Statistics*, Chichester: Wiley, 1981.

- Heckerman, D. (1996). *1 1 Bayesian Networks for Knowledge Discovery*.
- Hong, L., Dan, O., & Davison, B. D. (2011, March). Predicting popular messages in twitter. In *Proceedings of the 20th international conference companion on World wide web* (pp. 57-58). ACM.
- Jain, A. K., & Dubes, R. C. (1988). *Algorithms for clustering data*.
- Johnsen, M. (2017). *The Future of Artificial Intelligence in Digital Marketing: The next big technological break*. Maria Johnsen.
- Jones, R. H., & Hafner, C. A. (2012). *Understanding digital literacies: A practical introduction*. Routledge.
- Kempe, D., Kleinberg, J., & Tardos, É. (2003, August). Maximizing the spread of influence through a social network. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 137-146). ACM.
- Kohavi, R. (1998). Glossary of terms. *Machine Learning*, 30, 271-274.
- Klösger, W. (1996, February). Explora: A multipattern and multistrategy discovery assistant. In *Advances in knowledge discovery and data mining* (pp. 249-271). American Association for Artificial Intelligence.
- Liu, J., Dolan, P., & Pedersen, E. R. (2010, February). Personalized news recommendation based on click behavior. In *Proceedings of the 15th international conference on Intelligent user interfaces* (pp. 31-40). ACM.
- Mannila, H. (1996, June). Data mining: machine learning, statistics, and databases. In *Scientific and Statistical Database Systems, 1996. Proceedings., Eighth International Conference on* (pp. 2-9). IEEE.
- Matheus, C. J., Piatetsky-Shapiro, G., & McNeill, D. (1996). 20 selecting and reporting what is interesting: The kefir application to healthcare data.
- Mooney, R. J., & Roy, L. (2000, June). Content-based book recommending using learning for text categorization. In *Proceedings of the fifth ACM conference on Digital libraries* (pp. 195-204). ACM.
- Phelan, O., McCarthy, K., & Smyth, B. (2009, October). Using twitter to recommend real-time topical news. In *Proceedings of the third ACM conference on Recommender systems* (pp. 385-388). ACM.
- Piatetsky-Shapiro, G. (1996). *Advances in knowledge discovery and data mining* (Vol. 21). U. M. Fayyad, P. Smyth, & R. Uthurusamy (Eds.). Menlo Park: AAAI press.
- Russell, S. J., & Norvig, P. (2016). *Artificial intelligence: a modern approach*. Malaysia; Pearson Education Limited.

- Sarkar, D., Bali, R., & Sharma, T. (2018). *Practical Machine Learning with Python*.
- Silverman, B. W. (2018). *Density estimation for statistics and data analysis*. Routledge.
- Smith, G. (2007). *Tagging: people-powered metadata for the social web*. New Riders.
- Szabo, G., & Huberman, B. A. (2010). Predicting the popularity of online content. *Communications of the ACM*, 53(8), 80-88.
- Titterington, D. M., Smith, A. F., & Makov, U. E. (1985). *Statistical analysis of finite mixture distributions*. Wiley.
- Weiss, S. M., & Kulikowski, C. A. (1991). *Computer systems that learn: classification and prediction methods from statistics, neural nets, machine learning, and expert systems*. Morgan Kaufmann Publishers Inc.
- Zembowicz, R., & Żytkow, J. M. (1996, February). From contingency tables to various forms of knowledge in databases. In *Advances in knowledge discovery and data mining* (pp. 328-349). American Association for Artificial Intelligence.