



EÖTVÖS LORÁND UNIVERSITY
FACULTY OF INFORMATICS

SENTIMENT ANALYSIS AND OPINION MINING TECHNIQUES FOR LEARNING ANALYTICS.

TOMÁŠ HORVÁTH

HEAD OF THE DATA SCIENCE AND ENGINEER-
ING DEPARTMENT AT ELTE

IMANE BOUACIDA

COMPUTER SCIENCE

BUDAPEST, 2018

Acknowledgement

First and foremost, i would like to praise God for bestowing me with never ending patience, grace and will to pursue this work.

I would like to express my gratitude to my supervisor Dr Tomáš Horváth, for all the support, help, guidance as well as for his valuable advice and encouragement in producing my dissertation. It was a great pleasure to work with him. I would like to thank him warmly and to express my sincere gratitude to him.

I would also like to acknowledge Phd student Tsegaye Misikir of the faculty of informatics at Elte university. His guidance and motivation helped me a lot the process of researching and writing this thesis.

My sincere thanks to the Eötvös Loránd University for giving me this chance to study at such a great university. Thank you for all members of Elte university, faculty of informatics and data science and engineering department.

Last and not least, I must express my very profound gratitude to my parents, my family and to my friends for providing me with unfailing support and continuous encouragement throughout my years of study.

Contents

1	Introduction	1
1.1	Motivation	1
1.2	Thesis outline	2
2	Sentiment Analysis and Opinion Mining	3
2.1	Sentiment Analysis Research and Application	3
2.2	Sentiment Analysis and Opinion Mining Definition	4
2.2.1	Opinion/sentiment definition	4
2.2.2	Opinion/Sentiment Components	5
2.2.3	Opinion/Sentiment Types	6
2.3	Opinion Mining Tasks	7
2.4	Opinion Mining Procedure	8
2.4.1	Data Collection	8
2.4.2	Opinion Identification	8
2.4.3	Aspect Extraction	9
2.4.4	Opinion Classification	9
2.4.5	Production Summary	9
2.4.6	Evaluation	9
2.5	Level of Sentiment Analysis	9
2.5.1	Document Level	10
2.5.2	Sentence Level	11
2.5.3	Entity and Aspect Level	11
2.5.4	Concept Level	12
2.6	Types of Sentiment Analysis	13
2.6.1	Fine-grained Sentiment Analysis	13
2.6.2	Emotion Detection	13

CONTENTS

2.6.3	Aspect-based Sentiment Analysis	13
2.6.4	Intent Analysis	13
2.6.5	Multilingual Sentiment Analysis	13
2.7	Sentiment Analysis/Opinion Mining Approaches	14
2.7.1	Machine Learning Approach	15
2.7.2	Lexicon based Approach	15
2.7.3	Hybrid Approach	16
2.7.4	Graph based Approach	16
2.7.5	Other approaches	16
3	Proposed Model	17
3.1	Overview of The Proposed Model	17
3.2	Preprocessing	18
3.2.1	Tokenization	19
3.2.2	Text Cleaning	19
3.2.3	Stemming and Lemmatization	20
3.2.4	Feature Extraction using Bag of Word	21
3.3	Topic Modeling	23
3.3.1	Latent Dirichlet Allocation (LDA)	24
3.3.2	Latent Semantic Indexing (LSI)	26
3.4	Sentiment Classification using SentiWordNet	28
4	Experiments	29
4.1	Dataset	29
4.1.1	Mark My Professor Website	29
4.1.2	Data Collection	31
4.1.3	Import.io Website	32
4.1.4	Collected Data	33
4.2	Performance Measures	35
4.2.1	Matrix Similarity	36
4.2.2	Evaluation Metrics	37
4.3	Implementation Details	40
4.3.1	Packages Used	40
4.4	Results and Discussion	42

CONTENTS

4.4.1	Topic Modeling (LDA) Evaluation	42
4.4.2	SentiWordNet sentiment classification Evaluation	45
5	Conclusion and Future Work	48
5.1	Final Considerations	48
5.2	Future Work	49

List of Figures

2.1	Opinion Mining process [20]	8
2.2	Different levels of sentiment analysis	10
2.3	Overview of sentiment analysis approaches	14
3.1	Overview of the proposed model	18
3.2	Text preprocessing techniques	19
3.3	Example for text preprocessing	21
3.4	Term document matrix	22
3.5	Example of feature extraction.	23
3.6	Topic modeling framework [3]	24
3.7	LDA generative process [5]	25
3.8	LDA model [5]	26
3.9	Schematic of the Singular Value Decomposition (SVD) of a rectangular term by document matrix [13]	28
4.1	Elte informatics Mark My Professor Website	30
4.2	Example of Mark My Professor user interface	31
4.3	Data extraction using Import.io	33
4.4	Collected data from Import.io	33
4.5	The first five rows of the ratings dataset	34
4.6	The first five rows of the reviews dataset	35
4.7	Example of Cosine similarity	37
4.8	Confusion matrix	38
4.9	Accuracy	39
4.10	Precision	39
4.11	Recall	40

LIST OF FIGURES

4.12 Results of similarity matrix of the first ten professors	43
4.13 Confusion matrix of each attribute based on the LDA results and students the ratings	44
4.14 Confusion matrix based on students average rating and the sentiment analysis results	46

List of Tables

4.1	Final model statistics at each attribute based on the LDA results and the students ratings	45
4.2	Final model statistics based on students average rating and the sentiment analysis results	46

Chapter 1

Introduction

1.1 Motivation

Huge amount of digital data is generating daily from websites, forums, news sites, and social media and it becomes more difficult to get insight-full information without using computers. One of the domains were such huge textual data coming is the educational sector specially, professor's performance evaluation in the form of rating and review by students. When teacher performance is evaluated by students, varied opinions are collected from the same established criteria. Therefore, using computer and automated tools to filter and get useful information for decision making will be mandatory.

When we have such opinion and sentiment data, the use sentiment analysis and opinion mining methods to the analysis these comments will be crucial and mandatory. Sentiment Analysis is an application of natural language processing, text mining and computational linguistics, to identify information from the text. Students represent their emotions in comments, so it is a way to learn about various aspects of the students. Using sentiment analysis will help to know their opinion and determine whether there is a connection between their opinion and rating of the professor. Student feedback on quality and standards of learning is considered as a strategy to improve the teaching process and can be collected through a variety of social network, blogs and surveys.

In this research, we will apply topic modeling techniques to check whether the comments of the students are aligned with the given five attributes by computing the

similarity between the predicted five topics with actual five attributes. We will also apply the lexical-based sentiment classification on the review data to classify them into negative and positive to check weather reviews of the students are matching with the average rating of the students.

1.2 Thesis outline

The thesis is organized as follows:

- Chapter 2 provides an overview about sentiment analysis and opinion mining and their techniques.
- Chapter 3 gives an detailed explanation of proposed model, preprocessing step, feature extraction and an overview of the algorithms are used.
- Chapter 4 gives an detailed explanation of the datasets that are used in this experiment, performance measure that are used to evaluate the results and an explanation of the experiments made and discussion.
- Chapter 5 present the conclusion of the work and provides the future direction of research.

Chapter 2

Sentiment Analysis and Opinion Mining

2.1 Sentiment Analysis Research and Application

Sentiment analysis and opinion mining gain a great deal of importance as active research areas in natural language processing which explain the existence of extensive research. In this area, those two processes introduce a huge problem and some different tasks, mainly focuses on opinions which express or imply positive or negative sentiments by deleting it, analyzing it and extracting it. The research about the opinion mining began from the early 2000, but the term sentiment analysis perhaps first appeared in [33], and the term opinion mining first appeared in [12, 10, 32, 37, 44, 45, 47] [28].

The research in the sentiment analysis has become a very active research area because of the huge volume of opinionated data in the web and special in social media. It become now right at the center of the social media research. In the past 15 years, various researches have been conducted to examine and analyze the opinions within news, articles, and product and service reviews [41]. Nowadays, researches are conducting their research by intends to extract the sentiment embedded in messages posted on social media websites or sentiment extraction from Internet websites such as blogs and forums [20]. Discovering the knowledge embedded in social multimedia is of a great importance since it is vital for many promising applications which clearly explains the reason behind the existence of numerous sentiment analysis ap-

plications in all the domains such as Social media monitoring, Brand monitoring, Voice of customer, Customer service, Workforce analytics and voice of employee Product analytics, Market research and analysis [1].

2.2 Sentiment Analysis and Opinion Mining Definition

Sentiment analysis (also called opinion mining, opinion extraction, sentiment mining, subjectivity analysis, effect analysis, emotion analysis, review mining) is type of natural language processing that builds systems that try to detect, analyze, study, extract the humane sentiments, opinions, moods, evaluations, appraisals, attitudes, and emotions towards entities and their aspects expressed in text. This topic is applied on reviews and survey responses, online and social media, blogs, forums. Due to [20] two major definitions of opinion mining can be seen in the literature. The first definition is proposed in [39], as *“The automatic processing of documents to detect opinion expressed therein, as a unitary body of research”*. The second major definition says: *“Opinion mining is extracting people’s opinion from the web. It analyzes people’s opinions, appraisals, attitudes, and emotions toward organizations, entities, person, issues, actions, topic and their attribute”* [22, 28, 30].

2.2.1 Opinion/sentiment definition

Before going into further details, let’s first give a definition of opinion, the opinion is point of view about the specific object or the features of this object done by the opinion holder at specific time. We use the following review segment on Galaxy to introduce the problem (an id number is associated with each sentence for easy reference):

“(1) last month, I bought a Samsung Galaxy. (2) It was a nice smartphone. (3) The picture quality is amazing. (4) The voice quality was clear too. (5) However, my friend thought the Galaxy was too expensive”

From this review, we notice that:

The first thing that we notice is that there are several opinions in these review positives and negatives, Positives like sentences (2), (3) and (4) while sentence (5) express negative opinion. Then we also notice that the opinions all have some features this features is called targets. The target of the opinion in sentence (2) is “*the Galaxy*” as a whole, and the targets of the opinions in sentences (3) and (4) are “*the picture quality*” and “*voice quality*” of the Galaxy respectively. The target of the opinion in sentence (5) is “*the price of the Galaxy*”, This review has opinions from two persons, The holder of the opinions in sentences (2), (3), and (4) is the author of the review “*me*”, but for sentence (5), it is “*my friend*”, the date of the review is “*last month*”.

Due to [30] An opinion is a quintuple, (e, a_e, s_{a_e}, h, t) , where e is an entity, a_e is an aspect of e , s_{a_e} is the orientation of the opinion about a_e , h is the opinion holder, and t is the time when the opinion on a_e is expressed by h . The opinion orientation s_{a_e} can be positive, negative or neutral, or be expressed with different strength/intensity levels. When an opinion is on the entity e itself as a whole, we use the special aspect $a_e = GENERAL$ to denote it.

2.2.2 Opinion/Sentiment Components

An opinion sentiment consist of of five key components:

- **Entity:** Is the object in which it was given the opinion, it can be product, person, service, topic, event or organization; It is associated with a pair, $e:(T, W)$, where T is a hierarchy of components (or parts), sub-components, and so on, and W is a set of attributes of e . Each component or subcomponent also has its own set of attributes [30]. In the previous example the entity it is the “*Galaxy*”.
- **Aspect of the entity:** Is a feature or an attribute of the entity, “*The picture quality*”, “*The voice quality*” and “*the price*”. In the example, it define an aspect of the entity “*Galaxy*”.
- **Opinion holder:** This is the person who gives a specific opinion about an object. The author and his friend are the opinions holders in the previous example.
- **Time:** The time of the expression of the opinion. “*Last month*” was the time of the expression the opinions in the given example.

- **Orientation:** Find the opinion either positive or negative or neutral. The opinions orientations of the given example is splitted between positive and negative.

2.2.3 Opinion/Sentiment Types

Regular and Comparative Opinions

1. **Regular type:** A regular opinion is often referred simply as an opinion in the literature and it has two subtypes [2]:
 - **Direct Opinion:** A direct opinion denotes an opinion referring directly to an entity or aspects of entity, e.g., *"It was a nice smartphone"*.
 - **Indirect Opinion:** An indirect opinion denotes an opinion referring indirectly to an entity or aspects of entity based on its effects on some other entities, e.g., *"After buying this smartphone, i can take amazing pictures"*, describes an desirable effect of the smartphone on the picture quality, which indirectly gives a positive opinion to the smartphone.
2. **Comparative type:** A comparative opinion expresses a relation of similarities or differences between two or more entities and/or a preference of the opinion holder based on some shared aspects of the entities [23, 24].
 - **Gradable comparison:** Such a comparison expresses an ordering relationship of entities being compared [28], e.g., *"Samsung Galaxy better than Iphone"*.
 - **Non-equal gradable comparison:** Such a comparison expresses a relation of two or more entities but does not grade them [28], e.g., *"Samsung camera differently from Iphone"*.

Explicit and Implicit Opinions

1. **Explicit opinion:** Is an regular or comparative opinion explicitly expressed in a subjective sentence, e.g:
"It was a nice smartphone".
"Samsung Galaxy better than Iphone".

2. **Implicit opinion:** Is an regular or comparative opinion implied in an objective sentence, e.g:

“After buying this smartphone, I can take amazing pictures”.

“The battery life of Samsung Galaxy is longer than Iphone”.

2.3 Opinion Mining Tasks

Due to [20] Opinion mining contains several tasks with different names which all of them are covered by it [28]:

- **Sentiment analysis:** Sentiment analysis is considered as a research area in the field of text mining. The purpose of sentiment analysis is the sentiment recognition and public opinion examination.
- **Opinion extraction:** The process of extraction of users’ opinions and find out the users’ ways of thinking from the web documents is called opinion extraction.
- **Sentiment mining:** Sentiment mining it is the process of determines whether the given text contains objective or subjective sentences and the extraction of the opinions and classifies them into three categories of positive, negative and neutral. A sentence is called objective (or factual), when it contains the factual information about the product. The subjective sentences represent the individual emotions about the desired product [15].
- **Subjection analysis:** The purpose of Subjection analysis is to identify, classify, and collect subjective sentences.
- **Sentiment mining:** Affect analysis specifies the aspects that are expressing emotions positive or negative in the text using the natural language processing techniques [19].
- **Review mining:** Review mining is a sub-topic of text sentiment analysis and its main purpose is to extract aspects from the authors’ sentiments and is to produce a summary of the sentiments [49].

2.4 Opinion Mining Procedure

[20] have modeled the opinion mining process in Figure 2.1 in which, each part has some obligations which are as follows:

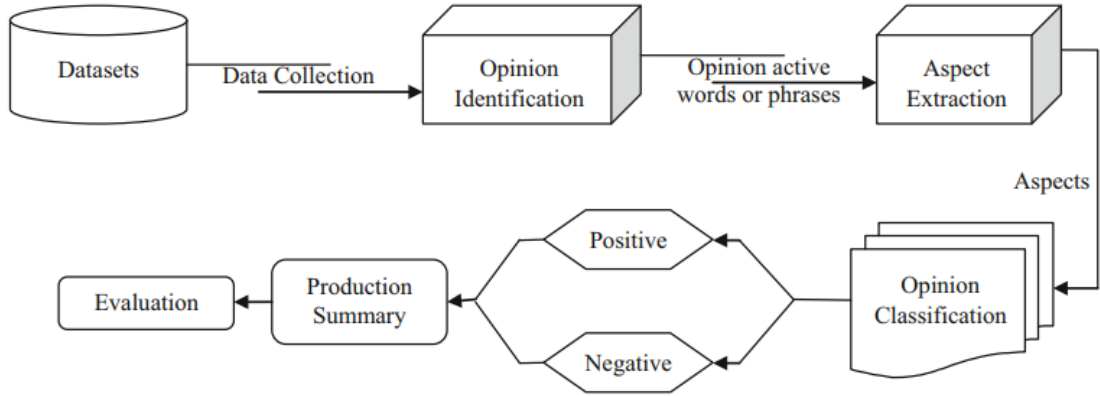


Figure 2.1: Opinion Mining process [20]

2.4.1 Data Collection

Having a comprehensive and reliable dataset is the first step to perform opinion mining process. The necessary information could be collected from various web resources, such as weblogs, micro blogs (such as Twitter), social networks (such as Facebook) and review websites. Using tools that are developed for extracting data through web, and using various techniques such as web scraping [35], can be useful to collect appropriate data. Some datasets are provided in English which can be used as references [37, 36, 6]. Researchers can apply their methods on these datasets for their simplicity.

2.4.2 Opinion Identification

In this phases, all the comments should be separated and identified from the presented texts. Then the extracted comments should be processed to separate the inappropriate and fake ones.

2.4.3 Aspect Extraction

In this phase, all the existing aspects are identified and extracted according to the procedures. Selecting the potential aspects could be very effective in improving the classification.

2.4.4 Opinion Classification

After the preprocessing phase that is opinion identification and aspect extraction, the opinion classification step can be applied, in this step the opinions are classified using different techniques which this paper summarizes, classifies and compares them.

2.4.5 Production Summary

In the production summary level, a summary of the opinion results is produced which can be in different forms such as text, charts etc, based on the results of the previous steps.

2.4.6 Evaluation

The performance of opinion classification can be evaluated using four evaluation parameters, namely accuracy, precision, recall and f-score.

2.5 Level of Sentiment Analysis

As shown in Figure 2.2, the sentiment analysis has been investigated mainly at four levels namely document level, sentence level, aspect level, and concept level. Document level is the abstract level of the sentiment analysis, that focuses on the opinion of the whole document which cannot be very accurate. Instead, the sentence level can be more accurate because it focuses on the polarity of each sentence in the document. Both the document level and the entity level don't discover exactly all the necessary details. Rather than the aspect level that extract all the opinion in the document, concept level is the fourth level of sentiment analysis that was introduced by [8] which focuses on the semantic analysis of the text.

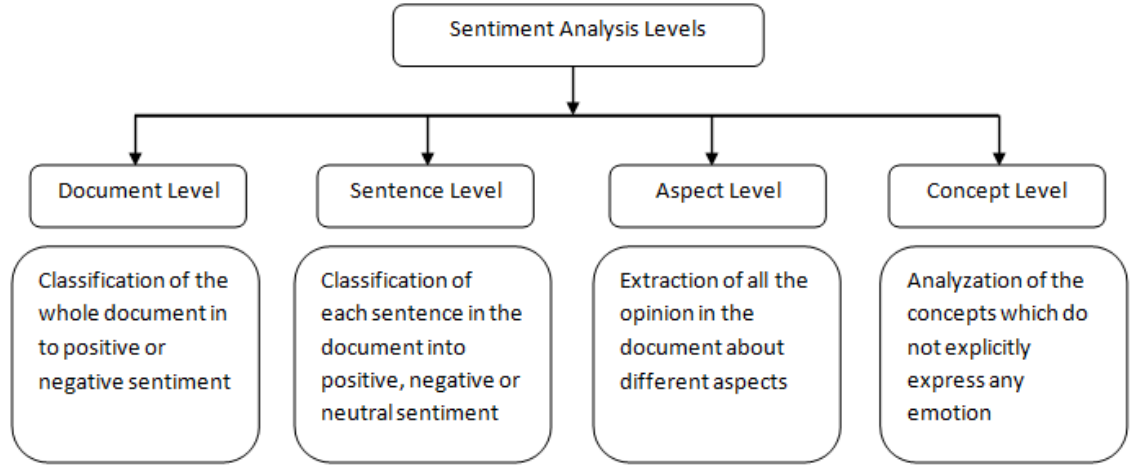


Figure 2.2: Different levels of sentiment analysis

2.5.1 Document Level

Sentiment analysis in this level focuses in the opinion of the whole document, it is a classification problem considering the problem of classifying an opinion documents not by topic but by overall sentiment [37, 45] to determine whether this document is positive or negative. This task of classification it known as the document level sentiment classification. It aims to automate the task of classifying a textual review which is given on a single topic as expressing a positive or negative sentiment or opinion [31]. It is defined by [28] as:

Given an opinion document d evaluating an entity, determine the overall sentiment s of the opinion holder about the entity, e.g., determine s expressed on aspect *GENERAL* in the quintuple $(_, GENERAL, s, _, _)$, where the entity e , opinion holder h , and time of opinion t are assumed known or irrelevant (do not care). If s is with categorical values then it is a classification problem. If it takes numeric values or ordinal scores within a given range, the problem becomes regression [28]. In this level of sentiment analysis the whole document it considered as a single entity Thus, it is not applicable for precise evaluation and comparison.

2.5.2 Sentence Level

Sentiment analysis in this level focuses in the opinion of each sentence in the document. The goal of this level of sentiment analysis is to classify opinions in each sentence into positive, negative or neutral opinion, This task of classification is known as the sentence level sentiment classification. Sentence sentiment classification can be solved either as two separate classification problems. The first problem (also called the first step) is to classify whether a sentence expresses an opinion or not. This classification problem is usually called subjectivity classification which determines whether a sentence expresses a piece of subjective information or factual (objective) information. The second problem (also called the second step) then classifies those opinion sentences into positive and negative classes [28].

It is defined by [28] as: Given a sentence x , determine whether x expresses a positive, negative, or neutral (or no) opinion.

In this level of sentiment analysis the whole document is broken into several sentences. It provides more entities that means more accuracy on the polarity of the document and naturally entails more challenges than the level of the document.

2.5.3 Entity and Aspect Level

This level of sentiment analysis does not care about the language structures (document, sentence). It is based on the idea that every opinion has a sentiment and a target. The result of this level can be a summary of the sentiments about different aspect of the entity. Aspect-based sentiment analysis (or opinion mining), or feature-based sentiment analysis (or opinion mining) as it was called in [21, 29] is summarized by [28] in this following six main tasks:

Task 1 (entity extraction and categorization)

Extract all entity expressions in the document and categorize synonymous entity expressions into a unique entity clusters.

Task 2 (aspect extraction and categorization)

Extract all aspect expressions of the entities and categorize these aspect expressions into a unique aspect clusters.

Task 3 (opinion holder extraction and categorization)

Extract opinion holders for opinions and categorize them into a unique opinion holder clusters.

Task 4 (time extraction and standardization)

Extract the times when opinions are given and standardize different time formats. Similar to the above tasks.

Task 5 (aspect sentiment classification)

Determine whether an opinion on an aspect is positive, negative or neutral, or assign a numeric sentiment rating to the aspect.

Task 6 (opinion quintuple generation)

Produce all opinion quintuples (e, a_e, s_{a_e}, h, t) expressed in document based on the results of the above tasks (summary of the sentiments).

Aspect level sentiment analysis extracts all the aspect from the document and then specifies their polarity. It provides all the necessary details in contrast of the document level and the sentence level and discover what exactly people liked and did not liked.

2.5.4 Concept Level

[8] discovered novel approaches to sentiment analysis and opinion mining which turns unstructured textual information to structured machine processable data. Conceptual approaches focus on the semantic analysis of the text through the use of web ontology or semantic networks which allow the aggregation of conceptual and affective information associated with natural language opinions, and also analyze the concepts which do not explicitly express any emotion [38]. The analysis at this level is intended to infer the semantic and affective information associated with natural language opinions, and hence to enable a comparative fine-grained feature-based sentiment analysis [8].

2.6 Types of Sentiment Analysis

Sentiment may include polarity or valence (e.g., *positive*, *negative*, *neutral*), emotion or feelings (e.g., *angry*, *happy*, *sad*, *proud*, *disappointed*, *etc.*), identify intentions (e.g., *interested*, *not interested*) and other affective states [1].

2.6.1 Fine-grained Sentiment Analysis

Fine grained sentiment analysis presents the level or the flavors of the polarity: very positive, positive, neutral, negative, very negative, mapped onto 5-star rating: Very Positive = 5 stars and Very Negative = 1 star, and it can be presented also with a particular feeling such as, anger, sadness, or worries for negative feelings or happiness, love, or enthusiasm for positive feelings.

2.6.2 Emotion Detection

Emotion detection systems are systems that can detect emotions like happiness, sadness, fear, etc., by resort to the lexicon or machine learning.

2.6.3 Aspect-based Sentiment Analysis

It is sentiment analysis in the aspect level used usually when analyzing the sentiment in subjects, for example to extract opinion which particular aspects or features of the product.

2.6.4 Intent Analysis

Intent analysis basically detects what people want to do with a text rather than what people say with that text.

2.6.5 Multilingual Sentiment Analysis

Analyze data in different languages. Sentiment analysis in multiple languages is often addressed by transferring knowledge from resource-rich to resource-poor languages, or by using a machine translation system to translate texts in other languages into English [11].

2.7 Sentiment Analysis/Opinion Mining Approaches

Sentiment analysis techniques can be categorized into five main approaches such as machine learning approach, lexicon based approach, hybrid approach, graph based approach and other approaches as is illustrated Figure 2.3:

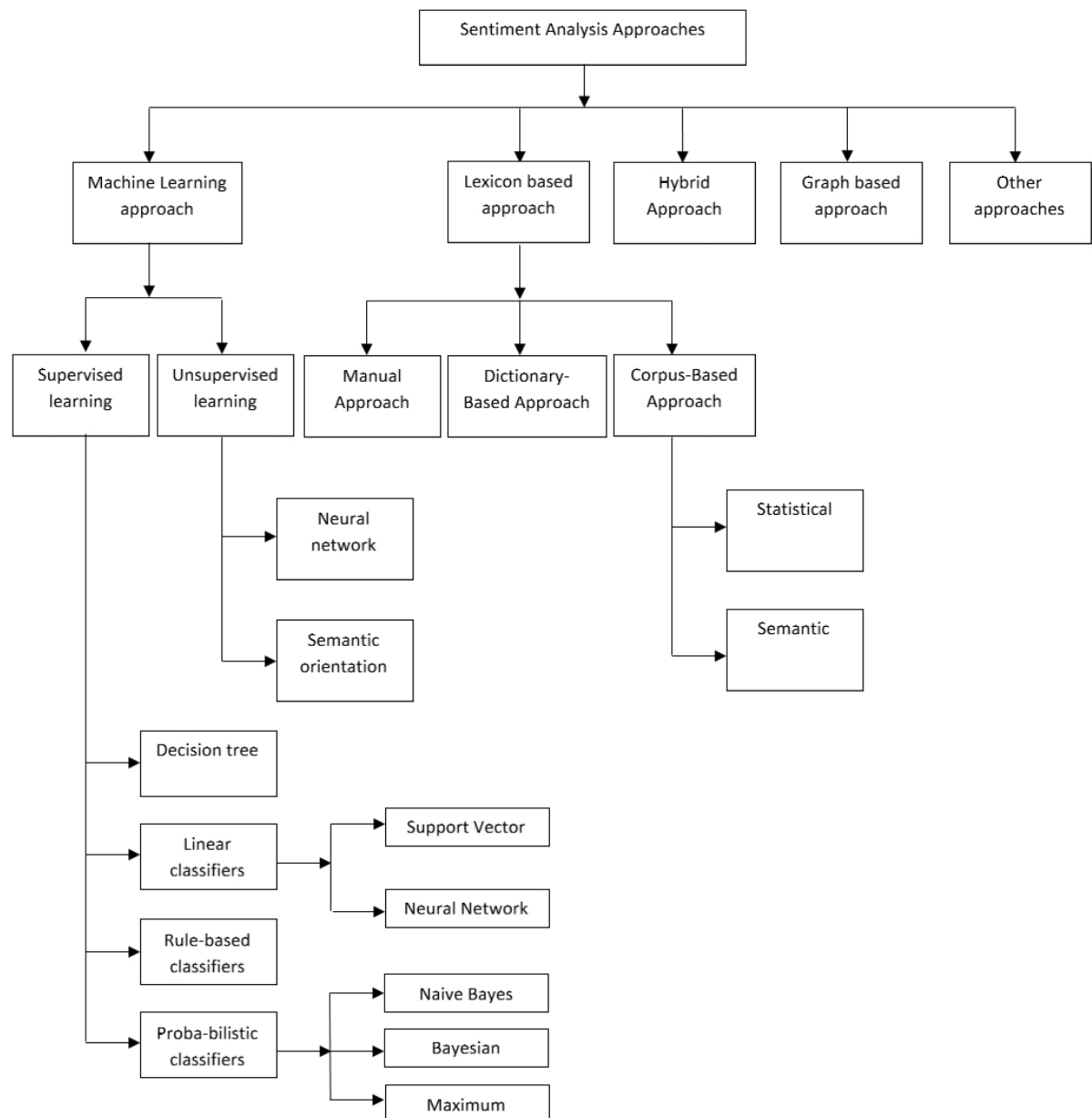


Figure 2.3: Overview of sentiment analysis approaches

2.7.1 Machine Learning Approach

Machine learning is one of the most interesting techniques and widely used due to its adaptability and accuracy, in the field of sentiment analysis. It is used to produce sentiment classification models. It uses semantic features by using supervised and unsupervised learning mechanisms. These methods first build a training set and label the training data by sentiment. A set of features are then extracted from the training data by the selection of appropriate features. Generally, unigrams (single word phrases), bi-grams (two consecutive phrases), tri-grams (three consecutive phrases) are selected as feature vectors, and this features are forwarded to a classifier model such as Naive Bayes (NB), Support Vector Machine (SVM), Logistic Regression (LR), Random Forest (RF), and so on. After training with the sentiment labels, the classifier can be utilized to predict the sentiment orientation of a sample which is not annotated [27].

1. **Supervised learning:** Builds a classification model to predict the class of labeled training documents based on predefined category [46].
2. **Unsupervised Learning:** Doesn't need to collect and create labeled training data and don't care about the domain and topic of training data [46].

2.7.2 Lexicon based Approach

Lexicon based approaches predict the overall sentiment based on an opinion lexicon and unlabeled data. It is a collection of positive and negative words along with opinion phrases. Lexicon based approach uses statistical or semantic methods that evaluate the words in the text based on opinion lexicon to find sentiment polarity of the text . This approach is categorized into three approaches based on the generation of the opinion lexicon manual approach, dictionary based approach, corpus based approach [46]. It is widely adopted in sentiment analysis because of the advantage that they do not need training data [2].

1. **Manual Approach** The collection of the sentiment word list is manually based on individuals domain knowledge and language understanding [46].
2. **Dictionary-Based Approach:** The collection of the sentiment word list is with know orientation from lexicographical resources like online dictionary

[42].

3. **Corpus-Based Approach:** Corpus-based approach exploits the syntactic pattern of co-occurrence words along with opinion words to identify and compile opinion words in large corpus, Corpus-based approach eliminates limitation of context-specific classification of opinion words in dictionary based approach [48].

2.7.3 Hybrid Approach

It is the combination of the two approaches such that machine learning approach and lexicon based approach, which could collectively exhibit the accuracy of a machine learning approach and the speed of lexical approach [27]. The advantage of hybrid approach is it makes the detection and measurement of sentiment at the concept level, and high accuracy from a powerful supervised learning algorithm [9].

2.7.4 Graph based Approach

Graph based methods are proposed to utilize the social graph and its attributes. considering the use of graphs to extract the most relevant words associated to the documents. These methods do not need large amounts of annotated data. However, they are domain dependent because the sentiment lexicons and the connection graphs are domain specific [27].

2.7.5 Other approaches

These approaches are sentiment analysis approaches which cannot be classified into the above categories.

Chapter 3

Proposed Model

3.1 Overview of The Proposed Model

In our proposed model, we apply topic modeling techniques to check whether the comments of the students are aligned with the given five attributes by computing the similarity between the predicted five topics with actual five attributes. We also apply the lexical-based sentiment classification on the review data to classify them into negative and positive to check whether reviews of the students are matching with the average rating of the students.

Our proposed model which describes a sentiment analysis and opinion mining for learning analytics is shown in Figure 3.1. As shown in Figure 3.1, the proposed system is running in three main phases such that 1) data preparation: this phase is the preprocessing of the collected data to make it suitable for the next step to extract the features correctly and, also, the feature extraction to represent the input text document into a numerical representation, 2) topic modeling: to check whether the comments of the students are aligned with the given five attributes by computing the similarity between the predicted five topics with actual five attributes, and, 3) sentiment analysis: to check whether reviews of the students are matching with the average rating of the students.

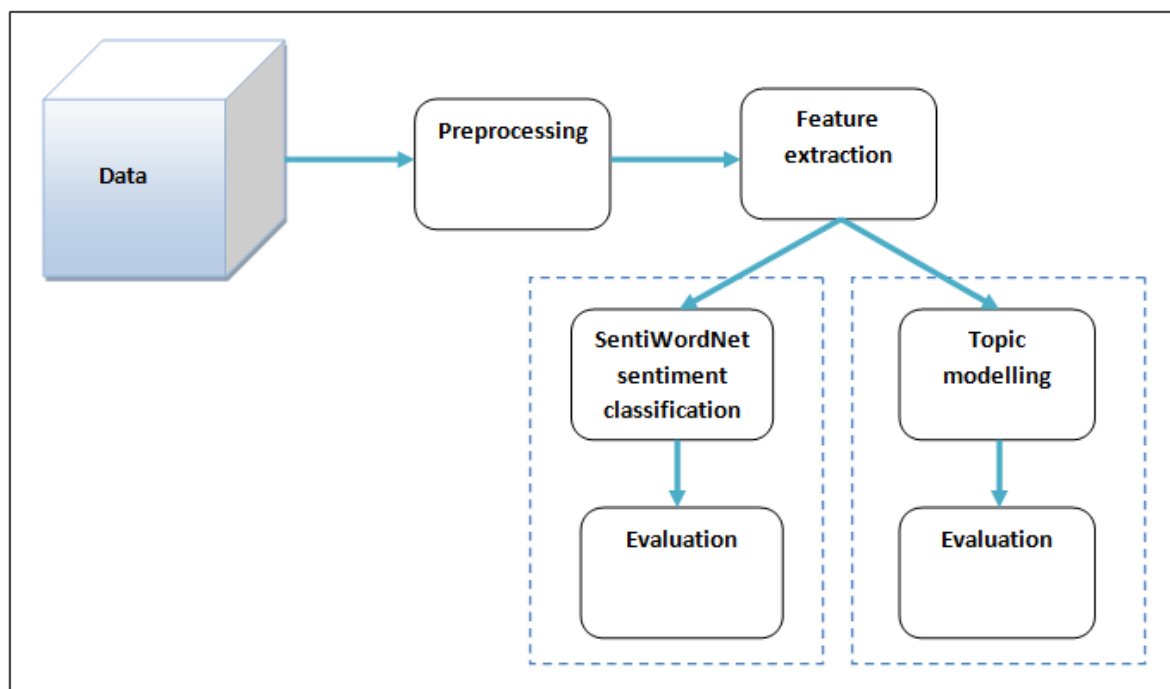


Figure 3.1: Overview of the proposed model

3.2 Preprocessing

Data preprocessing is an important task in machine learning and is the first step to be carried out. It is one of the steps in the natural language processing task that transforms data likely to contain many errors and not understandable into proper and understandable format. Preprocessing consists of several techniques that prepare raw data for further processing activity. As shown in Figure 3.2, the steps to be taken in text data preprocessing to assure the success of sentiment analysis and opinion mining techniques.

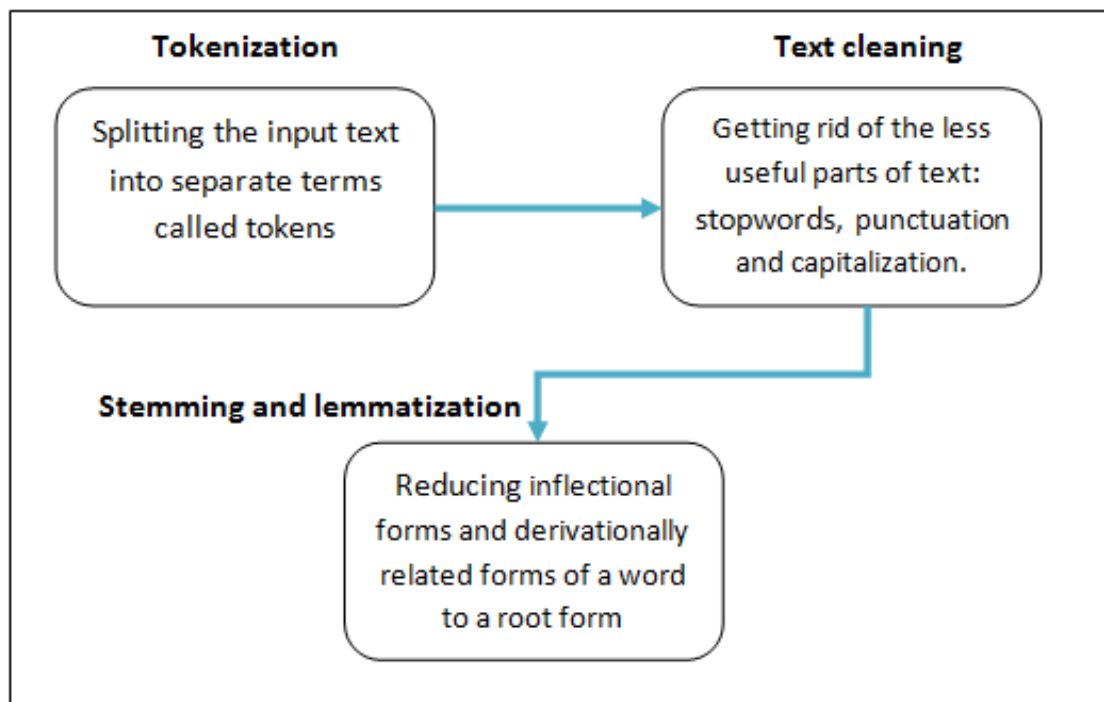


Figure 3.2: Text preprocessing techniques

3.2.1 Tokenization

The preprocessing task starts with tokenization. It is a mandatory step before any kind of processing and is considered as crucial step in natural language processing. Therefore, it is a crucial step in sentiment analysis and opinion mining because most sentiment analysis techniques comprehend words instead of text data. The most important thing to do with raw text is to split it into sentences and the sentences into separate terms called tokens. These tokens can be words, characters, punctuation symbols, each data row presented by list of tokens.

3.2.2 Text Cleaning

Not every token is apparent in the list of tokens. The output of tokenization step is valuable and some of them do not have a useful meaning and its analysis doesn't give any help to the sentiment analysis and opinion mining tasks. Therefore, the text cleaning task is needed before any sentiment analysis and opinion mining task.

Text cleaning task in our proposed system consist of several steps. Punctuation removing: it deletes all graphic signs which symbolizes a punctuation mark. The second step is stopword removal: stopwords are non significant words in a text that are the most common in a language such as *"the"*. finally Lowercase the words converting all text to the same case lower, removing small words such as *"my"*, words that have fewer than 3 characters.

3.2.3 Stemming and Lemmatization

For grammatical reasons, documents are using different forms of a word, and it contain families of derivationally related words with similar meanings. for the text mining process are different words without any relationship because it is not written in the same way. This explains the existence of methods in natural language processing to eliminate this issue. These methods are Stemming and Lemmatization.

Stemming is the process of producing the word stem by eliminating all kind of affixes from the word. It reduces words into their root form, however, lemmatization do not disperse too much about stemming differing in that lemmatization is able to find the base or dictionary form of a word based on a vocabulary and morphological analysis of words. The process of changing words into their root word is illustrated using Figure 3.3:

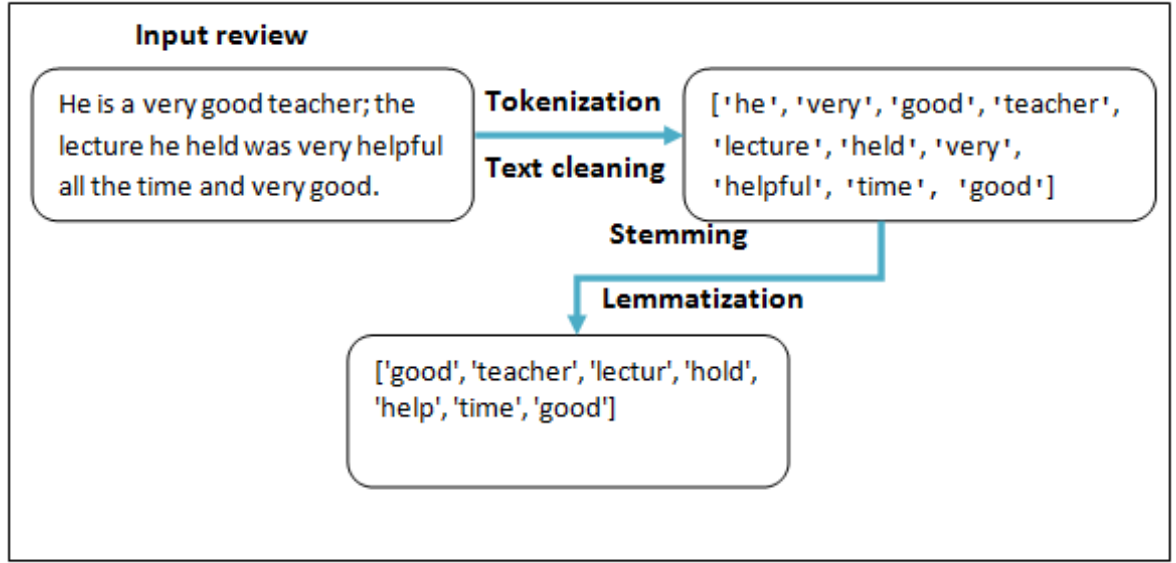


Figure 3.3: Example for text preprocessing

3.2.4 Feature Extraction using Bag of Word

To represent the input text document into a numerical representation, features that have to be extracted. In this section, we will present the type of features that will be used in the proposed system. In our proposed system, we used Bag-of-words (BoW). It is a commonly adopted and effective feature extractor for document representation, a representation of text that describes the occurrences (frequency) of words within a document. The bag involves two component: a vocabulary of known words and a measure of the presence of known words. Assume that we have a set of documents $D = \{d_1, d_2, \dots, d_n\}$ and the dictionary $T = \{t_1, t_2, \dots, t_m\}$. The output of the representation will be an $\mathbb{R}^{n \times m}$ matrix called the term document matrix. As shown in Figure 3.4:

	term₁	term₂	...	term_m
	↓	↓		↓
document₁ →	(1, C ₁₁)	(1, C ₂₁)	...	(1, C _{m1})
document₂ →	(2, C ₁₂)	(1, C ₂₂)	...	(1, C _{m2})
.
.
.
document_n →	(1, C _{1n})	(1, C _{2n})	...	(1, C _{mn})

Figure 3.4: Term document matrix

In our approach the feature extraction step consist of two branch steps:

- **Dictionary:** Create a dictionary containing all words appearing in the document by deleting the repetition.
- **Bow corpus:** For each document, we create a vector reporting the words presented by number. This number is it's arrange in the dictionary and how many times those words appear.

Let's apply the feature extraction on the previous example as shown in Figure 3.5:

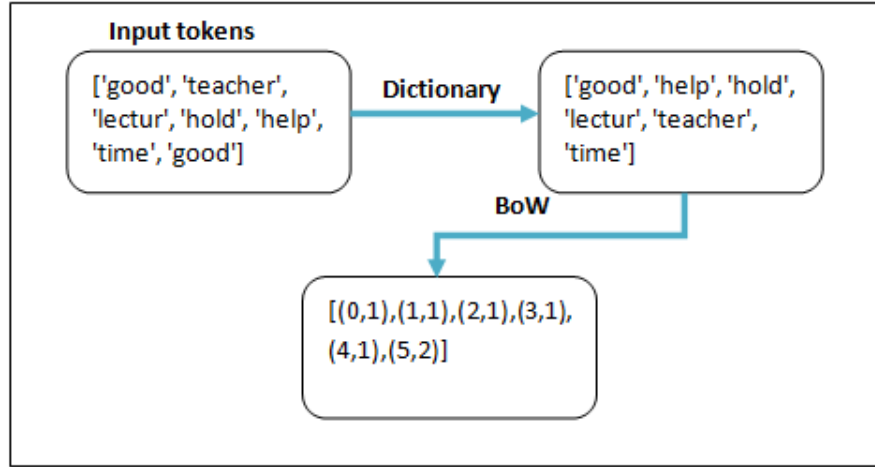


Figure 3.5: Example of feature extraction.

3.3 Topic Modeling

It becomes difficult to get meaningful information from large amount of text (unstructured data). The more data, the more information becomes available; It became necessary to exist a powerful techniques for analysis of a huge collection of a document. Topic modeling provides methods to organize, understand and summarize large collections of textual information

Topic modeling refers to a suite of algorithms or methods that identifies hidden thematic structure of a document in a large collection. The inputs of the algorithm are a document in collection of texts and its output is a set of topics and the degree to which each document exhibits those topics. A topic is a set of words that often occur together and have same content. This means that the main significance of topic modeling is to find the structure of word use and how to link documents that share the same structure. A topic model is a generative model for a collection of documents which describes a simple probabilistic procedure. By using probabilistic procedure, documents can be produced and a new document generated by choosing a distribution over topics. Then, each and every word in that document chooses a topic randomly depending on the distribution. After that, take a word from those topics. The results of topic modeling algorithms can be used for various text mining task such as summarization or document classification [3, 16].

There are a lot of Topic Modeling methods which include Vector Space Model (VSM), Latent Semantic Indexing (LSI), Probabilistic Latent Semantic Analysis (PLSA), Latent Dirichlet Allocation (LDA). In our experiment, we used LDA and LSI and we will address them later.

To get better understanding the framework of the topic modeling, we describe the basic concept behind it using Figure 3.6. It shows the steps in topic modeling which include bag of word, training of model and output of topic model.

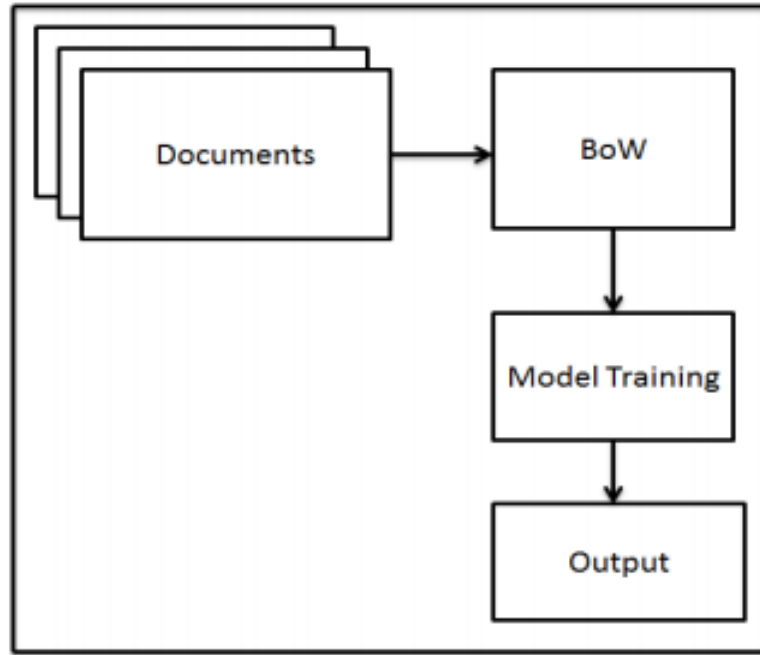


Figure 3.6: Topic modeling framework [3]

3.3.1 Latent Dirichlet Allocation (LDA)

[5] proposed the LDA model. Figure 3.8 shows a more general framework to overcome PLSA limitations [17]. LDA is a generative probabilistic model of a corpus. The basic idea is that documents are represented as random mixtures over latent topics, generated by a Dirichlet prior which is able to predict new documents, where each topic is characterized by a distribution over words [5].

In LDA, we assume that there are k underlying latent topics according to which documents are generated, and that each topic is represented as a multinomial distribution over the $|V|$ words in the vocabulary. A document is generated by sampling

a mixture of these topics and then sampling words from that mixture [4].

We assume that we have [5]:

- A word is an unit-basis vector from a vocabulary set indexed by $\{1, \dots, |V|\}$.
- A document is a sequence of N words denoted by $W = \{w_1, w_2, \dots, w_N\}$.
- A corpus is a collection of M documents denoted by $D = \{w_1, w_2, \dots, w_M\}$.

The LDA generative process for each document W in a corpus D can be defined as follows, as shown in Figure 3.7 and Figure 3.8:

1. Choose the length of the document.
2. Choose a multinomial distribution θ over topics (a k -vector lies in the $(k-1)$ -simplex if $\theta_i \geq 0$, $\sum_{i=0}^k \theta_i = 1$) where k is the dimensionality of the topic variable z and the parameter vector α is a k -vector (k is the number topics) with components $\alpha_k > 0$, $p(\theta|\alpha)$ is the probability density function of the Dirchlet distribution:

$$p(\theta|\alpha) = \frac{(\Gamma \sum_{i=1}^k \alpha_i)}{\prod_{i=1}^k \Gamma(\alpha_i)} \theta_1^{\alpha_1-1} \dots \theta_k^{\alpha_k-1}$$

3. For each of the N words w_n in the document:
 - Choose a topic z_n with $p(topic) = \theta$.
 - Choose a word w_n from a multinomial conditioned on z_n with $p(w = w_j | topic = z_n, \beta)$ word probabilities β is $k \times V$ matrix $\beta_{ij} = p(w_j = 1 | z_i = 1)$.

1. Choose $N \sim \text{Poisson}(\xi)$.
2. Choose $\theta \sim \text{Dir}(\alpha)$.
3. For each of the N words w_n :
 - (a) Choose a topic $z_n \sim \text{Multinomial}(\theta)$.
 - (b) Choose a word w_n from $p(w_n | z_n, \beta)$, a multinomial probability conditioned on the topic z_n .

Figure 3.7: LDA generative process [5]

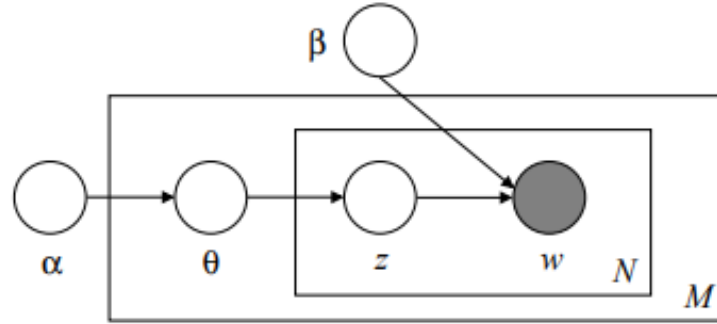


Figure 3.8: LDA model [5]

In general, we can summarize that LDA topic modeling considers the corpus like a mixture of topics that are presented in all the documents of the corpus, and each word in the corpus belongs in one topics of the corpus topics. The inputs of the LDA model are the corpus and the output is a different topic presented in the corpus and word distributions in each topic and the LDA process is to assign each word in every document of the corpus randomly to one of topics lists. Therefore, it gives topic representations of all documents and word distributions of all the topics.

3.3.2 Latent Semantic Indexing (LSI)

LSI also called as Latent Semantic Analysis (LSA), is a technique which analyzes relationships between a collection of documents and the terms they contain by producing a set of concepts related to the documents and terms [25]. The main idea behind LSI is to utilize term co-occurrence to derive a set of latent concepts, words which frequently occur together are assumed to be more semantically associated [40]. It uses SVD to identify patterns in the relationships between the terms and concepts contained in an unstructured collection of text [3]. LSI extracts latent topics from the corpus by decomposition or dimension reduction of the term-document matrix presentation of the corpus using sing singular value decomposition (SVD).

LSI Processing proposed by [7] is illustrated in the following steps:

1. A matrix A is formed, where in each row corresponds to a term that appears in the documents, and each column corresponds to a document. Each element

$a_{m,n}$ in the matrix corresponds to the number of times that the term m occurs in document n .

2. Local and global term weighting is applied to the entries in the term-document matrix. This weighting may be applied in order to achieve multiple objectives, including compensating for differing lengths of documents and improving the ability to distinguish among documents. Some very common words such as and, the, etc. typically are deleted entirely (e.g., treated as stopwords).
3. Singular value decomposition (SVD) is used to reduce this matrix to a product of three matrices:

$$A = U\Sigma V^T$$

Where $A \in \mathbb{R}^{t \times d}$ term document matrix corresponding to documents, $U \in \mathbb{R}^{t \times t}$ orthogonal matrix having the left singular vectors of A as columns, $\Sigma \in \mathbb{R}^{d \times d}$ orthogonal matrix having the right singular vectors of A as columns, and, $V \in \mathbb{R}^{t \times d}$ diagonal matrix whose elements are the singular values of A (the non-negative square roots of the eigenvalues of AA^T).

4. Dimensionality is reduced by deleting all but the k largest values of Σ , together with the corresponding columns in U and V , yielding an approximation of A shown in Figure 3.9.

$$A_k = U_k \Sigma_k V_k^T$$

Which is the best rank- k approximation to A in a least-squares sense.

5. This truncation process provides the basis for generating a k -dimensional vector space. Both terms and documents are represented by k -dimensional vectors in this vector space.
6. New documents (e.g., queries) and new terms are represented in the space by a process known as folding-in [18].
7. The similarity of any two objects represented in the space is reflected by the proximity of their representation vectors, generally using a cosine measure.

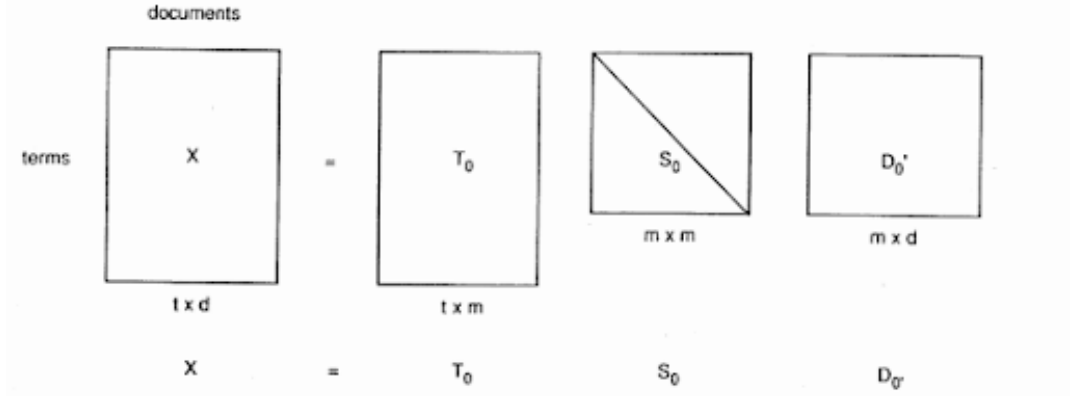


Figure 3.9: Schematic of the Singular Value Decomposition (SVD) of a rectangular term by document matrix [13]

3.4 Sentiment Classification using SentiWordNet

The aim of sentiment analysis is to extract the opinion of a given text at the document, sentence, or aspect level and classifying them whether the expressed opinion in a document, a sentence or an entity aspect is positive, negative, or neutral. In our research we apply the SentiWordNet lexical resource to the problem of sentiment classification of students reviews to classify them into positive or negative. SentiWordNet classification is lexicon based sentiment analysis and classification. It uses the opinion lexicon SentiWordNet which is derived from the WordNet database where each term is associated with numerical scores indicating positive and negative sentiment information [34].

SentiWordNet is an automatically generated lexical resource in which each WordNet synset is tagged with a triplet of numerical scores representing how Positive, Negative, and Objective a synset is. It is produced by asking an automated classifier Φ to associate to each synset s of WordNet, a triplet of numerical scores $\Phi(s, p)$ (for $p \in P = \{Positive, Negative, Objective\}$) describing how strongly the terms contained in s enjoy each of the three properties [14].

A WordNet is a lexicon. The idea behind WordNet is to create a “*dictionary of meaning*” integrating the functions of dictionaries and thesauruses. Lexical information is not organized in word forms, but in word meanings which is consistent with the human representations of meaning and their processing in the brain [26].

Chapter 4

Experiments

4.1 Dataset

All data science processes require the necessary amount of data to be able to work with. To evaluate the proposed model we need data. Our first task of the research is to collect it. In our case we need opinion text dataset that can be used for learning analytics in order to detect the mood and opinion of student on various topics of interest as well given lectures and we preferred use something related to our university, the Eötvös Loránd University in Budapest, Hungary. We used the Mark My Professor to collect data on ELTE Faculty of Informatics website¹.

4.1.1 Mark My Professor Website

Mark My Professor is a Hungarian website dedicated to evaluate higher education teachers and trainers by their students, the evaluation of teachers on the website is completely anonymous. The users of the website who have or are currently taking a particular professor's course may post a rating and review of any professor that is already listed on the site. Furthermore, users may create a listing for any individual not already listed, a student must rate the professor on a scale from one to five; five are the best, one is the worst in the following attributes: "*Performance of requirements*", "*Usefulness of Subject*", "*Helpfulness*", "*Preparedness*" and "*Diction*". Students also post review about a specific subject of this professor, every professor listed in the web site have computed the average on these five attributes. It is cal-

¹<http://www.markmyprofessor.com/>

culated from the students scales, and average of the five attributes results. There is existing version for all faculties on the Hungarian universities. In our experiment, we use the one that is custom of ELTE Faculty of Informatics that students of this faculty can evaluate teachers from the same faculty.

MARK my PROFESSOR [Bejelentés](#) [Regisztráció](#)

[Lépj be Facebookkal](#)

ELTE-IK

ISKOLA ADATLAP

Név
Eötvös Loránd Tudományegyetem
Informatikai Kar

Átlag
3.87

Rövidített név
ELTE-IK

Város
Budapest

honlap
www.elte.hu

[Ajánlom](#) [Forum](#) [Megosztás](#) [RSS csatorna](#)

ISKOLA TANÁRAI 412 TANÁR

	Név	Átlag	Követelmények teljesíthetősége	Szexi
1.	Abonyi-Tóth Andor	4.70	4.68	
2.	Ács Zoltán	4.38	4.50	
3.	Ambrus Tamás	0.00	0.00	
4.	Dr. Arató Miklós	3.10	2.76	
5.	Ásványi Tibor	2.91	3.69	
6.	ifj. Aszódi József	4.66	5.00	

MARK MY PROFESSOR

- ÁLLÁSKERESŐKNEK
- MUNKAADÓKNAK
- SZUPERDIÁK VERSENY
- NYEREMÉNYJÁTÉK

SAMSUNG Galaxy Watch

Figure 4.1: Elte informatics Mark My Professor Website



Figure 4.2: Example of Mark My Professor user interface

4.1.2 Data Collection

The collection of the data can be done in many different ways. There are accessible open data sources which are ready to use and to enforce for processes, or data which can be purchased from different companies and enterprises or it can be taken for free. Another way is get the data from the Internet. The process of getting the information from the web is called the web indexing, the tools of web indexing is the web crawling or also called spider or spiderbot which semantically browses the World Wide Web. In our case we collected it using web crawling, we used the online crawler Import.io², this crawler extract our data from Mark My Professor website. Of course, the extracted data was written in Hungarian language. The next step before to be able to work with the data we need to translate it into English language. The translation step performed by same translation tools available online in the internet such as Google translation³ and another tools. Let us give an overview about the Import.io crawler and the way of its uses.

²<https://import.io/>

³<https://translate.google.com/>

4.1.3 Import.io Website

Import.io is an automated web platform for crawling or extracting data from websites it transfer unstructured data to structured data usable in different processes which are semantically browsing the World Wide Web. It is machine learning based with no coding required. The Import.io has uses in various domains:

- Retail & Manufacturing.
- Equity & Financial Research.
- Machine Learning Model Training.
- Risk Management.
- Product, Marketing & Sales.

Import.io has a number of features, mentioned in the Import.io website:

- **Extract:** First, the user enters a URL or multiple URLs of website(s) that provide the data. If the data is behind a login, behind an image, or it need to interact with a website, Import.io will skip this pages and it will get the user to pages of data need. Once there, the app attempts to automatically extract the data that it thinks the user needs, or simply the user can point and click on the data and direct the crawler what to extract. After the extraction, the crawler allows the download of the structured data as CSV, Excel, JSON or accessed via API.
- **Transform:** Import.io allows the user to clean and transform the extracted data in various formats. Also, it lets the integration of new data and addition of new columns in the extracted data.
- **Insights:** The crawler lets the user to visualize and report on the data. The user can see the data in graphs and charts.
- **Integrate:** The Import.io APIs automate the integration of the extracted data into the user internal and external process.

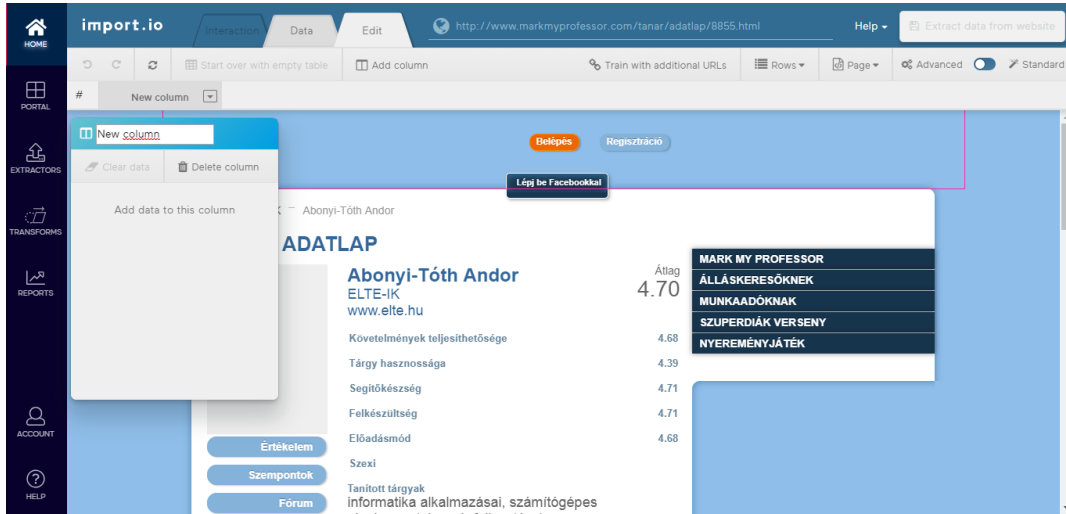
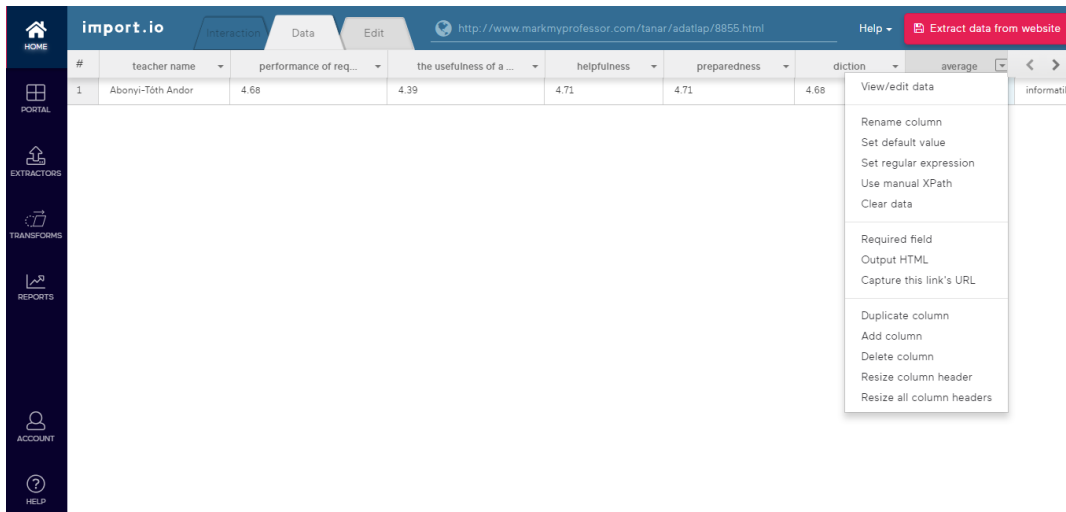


Figure 4.3: Data extraction using Import.io



sors. This rating is based on specific subject taught by the professor by evaluated him in five attributes: "*Performance of requirements*", "*Usefulness of Subject*", "*Helpfulness*", "*Preparedness*" and "*Diction*". The dataset is containing 355 columns, each column is described with the name of the teacher (teacher name), the average of the students ratings for each attribute from the above mentioned five attributes of ratings, and, also the professor's courses (courses).

	teacher name	performance_of_requirements	the_usefulness_of_subject	helpfulness	preparedness	diction	average	courses
0	Abonyi- Tóth Andor	4.68	4.39	4.71	4.71	4.68	4.634	information technology applications, computer ...
1	Ács Zoltán	4.50	3.50	4.83	4.66	3.66	4.230	Computer networks, Computer networks and Inter...
2	Arató Miklós	2.76	3.38	3.23	3.66	2.42	3.090	Multivariate data analysis, probability calcul...
3	Ásványi Tibor	3.67	3.56	3.56	3.09	2.01	3.178	Algorithms and Data Structures 1 GY, Algorithm...
4	jr.Aszódi József	5.00	3.00	5.00	4.00	5.00	4.400	Functional Programming

Figure 4.5: The first five rows of the ratings dataset

The second dataset is the reviews dataset representing a set of students reviews. The reviews are given for the professor and the subject taught by him based on the above mentioned five attributes. The dataset contains 5346 columns described with the name of the professor who has been evaluated (teacher name), the subject name on which the professor was evaluated (subject) and the comment in which describe the opinion (comment). All the comments labeled with the same professor name were merged in the same comment.

	teacher name	subject	comment
0	Abonyi-Tóth Andor	web development 1	The lecture he held, and he was my intern, was...
1	Abonyi-Tóth Andor	computer basics	One of the best students, the ZHK is well-craf...
2	Abonyi-Tóth Andor	web development 1	very good teacher, helpful, direct, understand...
3	Abonyi-Tóth Andor	computer basics	He is a very good practitioner, he is absolute...
4	Abonyi-Tóth Andor	web development 1	Useful subject and interesting presentation.

Figure 4.6: The first five rows of the reviews dataset

4.2 Performance Measures

In order to see the performance of the topic modeling algorithm and to check whether the comments of the students are aligned with the given five attributes, we computed the similarity between the predicted five topics with actual five attributes. After doing the similarity and getting some output in forms of binary classes (positive and negative), the next step is to find out how effective is the model based on some metrics. Different performance metrics are used to evaluate based on the predicted similarity and the ratings which are given by the students for every attributes given by the system.

The other experiment that we carried out is the sentiment analysis. For each professor, students give ratings and textual comments. To see the agreement between the ratings and the students textual feedbacks, we performed sentiment analysis. First we classify the comments into positive and negative using the SentiWordNet sentiment analysis toolkit. Then, we compare the predicted class with the average rating given by students for each professor. After doing the sentiment analysis and getting some outputs in forms of positive and negative classes, the next step is to find out how effective is the model. Depending on the use of some metrics. Different performance metrics are used to evaluate different machine learning algorithms. We can use classification performance metrics such as confusion matrix, accuracy, precision and recall.

4.2.1 Matrix Similarity

A document is represented using term document metric with the use of the bag of word feature extraction where the value of each dimension corresponds to the number of times that term appears in the document. In order to find which of them are relevant, a metric similarity is needed to calculate distance between the vectors. For this reason the cosine similarity is the more appropriate to us because property of the cosine similarity is its independence of document length, it then gives a useful measure of how similar two documents are likely to be in terms of their subjects matter.

To compute the similarity between the predicted topics and the actual attributes, we used cosine similarity. Cosine similarity of two documents corresponds to the correlation between the vectors, this is quantified as the cosine of the angle between vectors [43].

Given two topic vector representation $A, B \in \mathbb{R}_m$ over the word set $W = \{w_1, w_2, \dots, w_N\}$, their cosine similarity is:

$$Cos(A, B) = \frac{a^T b}{\|a\|_2 \|b\|_2}$$

For example, consider the following four documents: $D_1 = \text{"good teacher"}$, $D_2 = \text{"good lecture"}$, $D_3 = \text{"teacher lecture"}$, $D_4 = \text{"teacher teacher lecture lecture"}$. As shown in Figure 4.7, the Documents D_3 and D_4 representing the same topic, the cosine of the angle between the two vectors D_3 and D_4 is 1; D_1 and D_2 are very similar topics. While with cosine similarity the order of relevance to: D_1 will be D_4 , D_2 and D_3 the same order with D_4 , D_2 will be D_1 and D_3 the same order with D_4 , D_3 will be D_4 and D_1 the same order with D_2 , D_4 will be D_3 and D_1 the same order with D_2 .

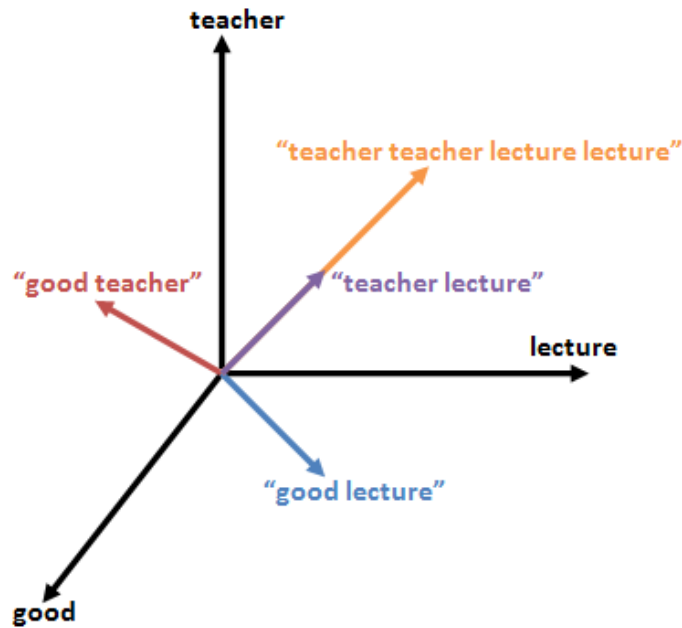


Figure 4.7: Example of Cosine similarity

4.2.2 Evaluation Metrics

Different performance metrics are used to evaluate different ML Algorithms. For now, we will focus on the ones used for classification problems. We can use classification performance metrics such that confusion matrix, accuracy, precision and recall. Before we express the used evaluation metrics, lets define terms associated with them, the terms true positive, true negative, false positive, and false negative compare the results of the classifier to be tested with external trusted evaluations. The terms positive and negative refer to the classifier's output, while the terms true and false refer to whether that classification is compliant with the one performed from the trusted external evaluation.

- **True Positives (TP):** True positives are the cases when the actual class was True and the predicted is also True.
- **True Negatives (TN):** True negatives are the cases when the actual class was False and the predicted is also False.
- **False Positives (FP):** False positives are the cases when the actual class was False and the predicted is True.

- **False Negatives (FN):** False negatives are the cases when the actual class of the data point was True and the predicted is False.

1. **Confusion matrix:**

The confusion matrix in itself is not a performance measure as such, but almost all of the performance metrics are based on it. The confusion matrix is a matrix presentation of the accuracy of a model with classes. In our proposed system this shows a more detailed breakdown of correct and incorrect classifications for each class. The confusion matrix is a matrix that our actual classifications are columns and the predicted ones are rows and sets of “**classes**” in both dimensions. Figure 4.8 shows the example of confusion matrix.

		Actual	
		Positive	Negative
Predicted	Positive	TP	FP
	Negative	FN	TN

Figure 4.8: Confusion matrix

2. **Accuracy:**

Accuracy measures how often the classifier makes the correct prediction. Is calculated as the ratio between the number of correct predictions and all the predictions made.

$$Accuracy = \frac{TP + TN}{all\ predictions}$$

		Actual	
		Positive	Negative
Predicted	Positive	TP	FP
	Negative	FN	TN

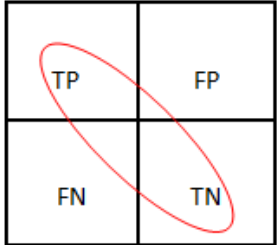
A 2x2 confusion matrix with 'Actual' as columns and 'Predicted' as rows. The cells are labeled TP (True Positive), FP (False Positive), FN (False Negative), and TN (True Negative). A red oval is drawn around the TP and TN cells, indicating the correct classifications used to calculate accuracy.

Figure 4.9: Accuracy

3. Precision:

Precision, or True Positive Accuracy, is a measure of exactness or fidelity and is calculated as the ratio of items correctly identified as positive and the total items identified as positive.

$$Precision = \frac{TP}{TP + FP}$$

		Actual	
		Positive	Negative
Predicted	Positive	TP	FP
	Negative	FN	TN

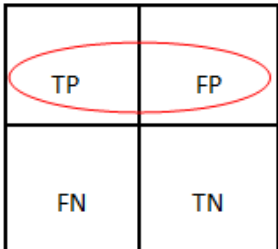
A 2x2 confusion matrix with 'Actual' as columns and 'Predicted' as rows. The cells are labeled TP (True Positive), FP (False Positive), FN (False Negative), and TN (True Negative). A red oval is drawn around the TP and FP cells, indicating the total items identified as positive used to calculate precision.

Figure 4.10: Precision

4. Recall:

Recall, or Sensitivity or True Positive Rate, is a measure of completeness and is calculated as the number of items correctly identified as positive out of total true positives.

$$Recall = \frac{TP}{TP + FN}$$

		Actual	
		Positive	Negative
Predicted	Positive	TP	FP
	Negative	FN	TN

Figure 4.11: Recall

4.3 Implementation Details

4.3.1 Packages Used

This section provides an overview of our proposed system implementation. It describes details about the programming language and its packages used in the implementation of the model and the data preprocessing. The models and the data processing were implemented in Python 2.7.14. The Python programming language is a multi-paradigm, general-purpose, interpreted, high-level programming language. It is one of the most popular languages for scientific computing and machine learning that can be used in many contexts and adapted to any type of use through specialized packages for each treatment. Python libraries offer open source implementations of many tasks and algorithms. In our implementation we used Pandas, Numpy, Gensim, Nltk, sklearn, Scikit-plot. We will give a brief description about this libraries and about our uses of it.

Gensim is a Python library that implements tools for work with topic modelling, document indexing and similarity retrieval. It is toolkit implementation of the natural language processing (NLP) and the information retrieval (IR) community. Gensim has very efficient implementations of popular algorithms, such as on-line Latent Semantic Analysis (LSA/LSI/SVD), Latent Dirichlet Allocation (LDA), Random Projections (RP), Hierarchical Dirichlet Process (HDP) or word2vec deep learning. Gensim is specifically designed to handle huge text collections. Gensim was chosen to be the implementation of topic modeling algorithms LDA and LSI, data

preprocessing, feature extraction, bag of words and performance evaluation matrix similarity.

NumPy (stands for Numerical Python) is the fundamental package for scientific computing with Python. It provides an abundance of useful features for operations on n -arrays and matrices in Python. The library provides vectorization of mathematical operations on the NumPy array type, sophisticated functions, tools for integrating C/C++ and Fortran code, useful linear algebra, Fourier transform, and random number capabilities. In our system, the term-document matrices was generated, for that the use of the NumPy libraries it is self evident.

The Natural Language Toolkit (NLTK) is a Python package for natural language processing. NLTK is a leading platform for building Python programs to work with human language data. It provides over 50 corpora and lexical resources such as WordNet, along with a suite of text processing libraries for classification, natural language processing, computational linguistics, parsing, tagging, tokenizing, syntax, linguistics, language, text analytics. NLTK was also used in the preprocessing such as stemming and lemmatization and also it was used to implement the SentiWordNet classification.

Pandas is a Python package designed to do work with “labeled” and “relational” data simple and intuitive. It is a perfect tool for data wrangling. It designed for quick and easy aggregation, and visualization. It allows manipulating data tables with labels of variables and individuals. These arrays are called DataFrames. One can easily read and write these DataFrames from or to a tabular file. Graphs can be easily drawn from these DataFrames using matplotlib. And also allows to handle missing data and powerful grouping by functionality.

Sklearn is a Python module for machine learning and image processing built on the top of SciPy, and makes heavy use of its math operations. It features various classifications, regression and clustering algorithms. It includes functions for support vector machines, random forests, gradient boosting, k -means and DBSCAN. In our case it was chosen for the implementation of the performance evaluation in the level of sentiment analysis to calculate the performance metrics accuracy, precision and recall.

Scikit-plot is a package that provides tools to generate quick and beautiful graphs and plots with as little boilerplate as possible. The library includes plots for machine learning evaluation metrics e.g., confusion matrix, plots built specifically for

classifiers and regressors, clusters of instances and dimensionality reduction.

4.4 Results and Discussion

This section details the results of the conducted experiments. The experiment was divided into two steps as follows:

- Applying different topic modeling algorithms on our datasets which are mentioned in chapter 3 and evaluating the results.
- Applying the SentiWordNet classification algorithm which is mentioned in chapter 3 and evaluating the results.

From the results of the topic modeling algorithms LDA and LSI, we found that the best model is LDA because it has a higher performance than the given baseline, LSA which gives us only one topic from every professor's comments; therefore it was not useful for our system. The calculated similarity matrix between the topics resulting from the LDA algorithm and the five attributes, and the rating given by the students for the five previous attributes are the measured values of different performance metrics which are mentioned in section 4.2. We will discuss the experimental results of the LDA performance in section 4.4.1. The results of the sentiment classification and the average rating given by the students are the measured values of different performance metrics; the experimental results will be discussed in section 4.4.2.

4.4.1 Topic Modeling (LDA) Evaluation

The objective of this experiment was to answer the first research question. According to the experimental results, the topics predicted using topic modeling are aligning with the actual five attributes given by the system. Figure 4.12 represents a sample consisting of ten professor's matrix similarity between the predicted topic for each attribute of the five attributes in the scale from 1 to 5.

teachers	performance_of_requirements	the_usefulness_of_subject	helpfulness	preparedness	diction
0	3.65619	3.65116	2.95952	1	1
1	3.64983	1	2.96406	1	1
2	3.64456	3.64336	2.9612	1	1
3	3.62947	3.62781	2.93904	1	1
4	1	1	1	1	1
5	1	3.65397	1	1	1
6	1	1	1	1	1
7	3.64499	3.64579	2.9619	1	1
8	1	1	2.99627	1	1
9	1	2.98165	1	1	1
10	3.63894	2.96551	2.97079	1	1

Figure 4.12: Results of similarity matrix of the first ten professors

Figure 4.13 and Table 4.1 show the results of LDA algorithm based on the measures of different performance metrics on dataset. Table 4.1 reports the accuracy, recall and precision for the first objective of the proposed system using topic modeling and Figure 4.13 present the confusion metrics. Each attribute from the five attributes is presented with the predicted similarity and the rating given by the students. As shown in Table 4.1, the reviews of the students are distributed to the five topics. Because of this, the recall and the precision vary from topic to topic:

As it is illustrated in the Table 4.1, concerning the performance of requirements (P of R) there are high precision 72% and low recall 28%. Which means that in one hand 72% from the reviews predicted to be positive; the ratings given by the students were positive as all. In the other hand percentage of 28% from the positive ratings resulted in positive reviews gathered from the students. Regarding the usefulness of a subject (U of S) the distinction between the precision and the recall is highly noticed, there are high precision 73% and low recall 38%. This indicates that 38% of the students' positive ratings have also positive reviews. As 73% from the students' positive reviews have a students' positive ratings as well. When it comes to the preparedness (P) and helpfulness (H) they have a very low precision 0,2% and very low recall 16%. which is reflected in 16% from the students' positive ratings are correspond to positive reviews and just 0,2% from the students' positive reviews are correspond to positive ratings. Finally the diction (D) has a low recall and low precision, the distinction between them was very high, i.e. 0,9% of students' positive

reviews are matching with the ratings and 30% of the positive ratings are matching with the reviews.

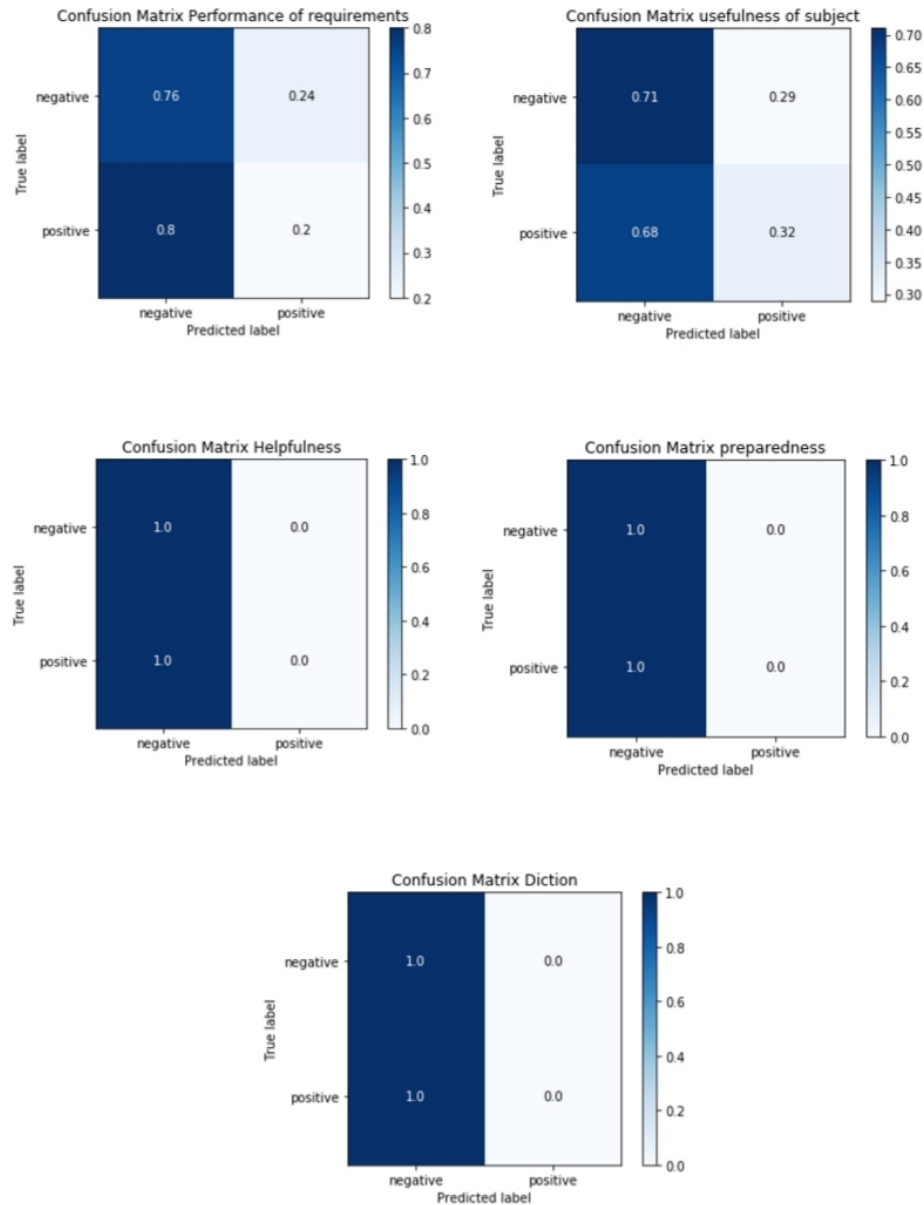


Figure 4.13: Confusion matrix of each attribute based on the LDA results and students the ratings

Performance Matrix	P of R	U of S	H	P	D
Accuracy	0.28	0.38	0.16	0.16	0.30
Precision	0.72	0.73	0.02	0.02	0.09
Recall	0.28	0.38	0.16	0.16	0.30

Table 4.1: Final model statistics at each attribute based on the LDA results and the students ratings

We performed the topic modeling experiment for two main objectives. The first one is to check whether the comments given by the students are based on the five attributes of the system or not. The second one is to see whether the comments of each topic and the ratings given by students for this topic were matching or not. According to the experimental results, 28% of the predicted comments and the average rating given by students concerning the performance of requirements attribute were matching and 72% were not matching. 38% of the predicted comments and the average rating given by students according the usefulness of a subject attribute were matching and 62% were not matching. The preparedness and the helpfulness have the same results, 16% were matching and 84% were not matching. Diction results states that 30% were matching and 60% were not matching. From this we can conclude that the textual comments given by the student and the ratings were not matching.

4.4.2 SentiWordNet sentiment classification Evaluation

The aim of this experiment is to answer the second research question. According to the experimental results, as shown in Figure 4.14 and Table 4.2, we measured the different performance metrics of the models on our dataset relative to the sentiment classification. Table 4.2 reports the accuracy, recall and precision of the second objective of the proposed system using sentiment analysis and Figure 4.14 present the confusion matrix. The matrix was calculated based on the predicted polarity from the comments and the ratings given by the students. As shown in table 4.2, the results are brilliant. We have a very high accuracy, recall and precision. The distinction between the recall and the precision is very low. The recall 79% and precision 81%. This indicates that 79% of the students' positive ratings have also

positive reviews. As 81% from the students' positive reviews have a students' positive ratings as well.

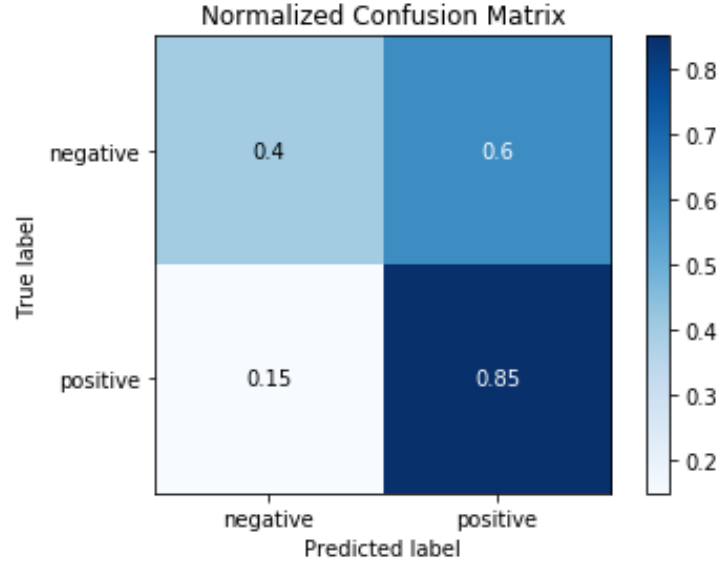


Figure 4.14: Confusion matrix based on students average rating and the sentiment analysis results

Performance Matrix	SentiWordNet Evaluation
Accuracy	0.79
Precision	0.81
Recall	0.79

Table 4.2: Final model statistics based on students average rating and the sentiment analysis results

The main objective of the sentiment analysis experiment is to see whether the comments given by the students and the average rating were matching or not. According to the experimental results, 79% of the textual comments given by the students and the average rating were perfectly matching. It was 21% from the textual comments and the average rating were not matching. From this we can conclude that the majority of students are given appropriate feedback.

From the conducted experiments and their results the majority of the students are not given feedback according the all the five attributes *performance of the require-*

ments, usefulness of a subject, preparedness, helpfulness, and diction, the majority of them concentrate to give feedback based on some of the attributes.

Chapter 5

Conclusion and Future Work

5.1 Final Considerations

The objective of this study was to develop and implement text mining techniques, with particular interest in opinion mining and sentiment analysis, to be used in the area of learning analytics in order to detect the mood and opinion of students on various topics of interest as well as on given lectures. We conducted experiments using state of the art opinion mining and sentiment analysis models. First of all, we collected two datasets from Elte Mark My Professor IK (Faculty of Informatics) website by using the Import.io web crawler and we translated it from Hungarian language to English language. The first data set was rating dataset and the second was reviews dataset which represent a set of ratings and a set of reviews about professors based on specific subject taught by the professor evaluated in five attributes: *performance of requirements*, *the usefulness of a subject*, *helpfulness*, *preparedness*, and *diction*. The datasets were preprocessed in a way that all the ratings of the same professor was presented by their average and all the reviews were merged in one review.

In this experiment, we extracted the features and, after that, we performed two experiments: We made topic modeling on the comments for each professor to check whether the comments was given based on the five attributes given by the system, and, to see whether the comments of each topic and the ratings given by the students for this topic were matching or not. For the topic modeling we used LDA and LSI algorithms. We have found that LDA have a higher performance than LSA. Also

we made sentiment analysis on each comment and classified them into positive and negative. We compared the results with average rating to see the alignment between the comments and the ratings. Our experiments were implemented in python and several of its packages were used. We found that most of the students give ratings based on some of the attributes given by the system.

5.2 Future Work

The experiment was carried out on a limited dataset. More useful and insightful results could be gained if the experiment was carried out on large scale dataset. Not only on large scale dataset but different methods of sentiment analysis could be used as well. When we collected the dataset, we translated the data from Hungarian language to English language using Google translation. Better result might be also obtained if the experiment was performed on the original data.

Developing an application in which students can provide their feedback and texts on a certain topic of interest. Will also be an interesting topic to obtain more data. The application could also contain a dashboard providing real-time feedback for teachers and learning managers helping them in further decision making in order to improve the learning experience of students.

Bibliography

- [1] Sentiment analysis nearly everything you need to know.
- [2] Penubaka Balaji, D.Haritha, and O.Nagaraju. An overview on opinion mining techniques and sentiment analysis. *International Journal of Pure and Applied Mathematics*, 118(19):61–69, Feb 2018.
- [3] B. V. Barde and A. M. Bainwad. An overview of topic modeling methods and tools. In *2017 International Conference on Intelligent Computing and Control Systems (ICICCS)*, pages 745–750, June 2017.
- [4] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. In T. G. Dietterich, S. Becker, and Z. Ghahramani, editors, *Advances in Neural Information Processing Systems 14*, pages 601–608. MIT Press, 2002.
- [5] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, March 2003.
- [6] John Blitzer, Mark Dredze, and Fernando Pereira. Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 440–447. Association for Computational Linguistics, 2007.
- [7] Roger Bradford. Techniques for processing lsi queries incorporating phrases. In Ana Fred, Jan L. G. Dietz, David Aveiro, Kecheng Liu, and Joaquim Filipe, editors, *Knowledge Discovery, Knowledge Engineering and Knowledge Management*, pages 99–117, Cham, 2015. Springer International Publishing.
- [8] Erik Cambria. An introduction to concept-level sentiment analysis. In Félix Castro, Alexander Gelbukh, and Miguel González, editors, *Advances in Soft*

- Computing and Its Applications*, pages 478–483, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.
- [9] Alessia D’Andrea, Fernando Ferri, Patrizia Grifoni, and Tiziana Guzzo. Approaches, tools and applications for sentiment analysis implementation. *International Journal of Computer Applications*, 125:26–33, 09 2015.
- [10] Sanjiv R. Das, Mike Y. Chen, To Vikas Agarwal, Chris Brooks, Yuk shee Chan, David Gibson, David Leinweber, Asis Martinez-jerez, Priya Raghubir, Sridhar Rajagopalan, Ajit Ranade, Mark Rubinstein, and Peter Tufano. Yahoo! for amazon: Sentiment extraction from small talk on the web. In *8th Asia Pacific Finance Association Annual Conference*, 2001.
- [11] Kia Dashtipour, Soujanya Poria, Amir Hussain, Erik Cambria, Ahmad Y. A. Hawalah, Alexander Gelbukh, and Qiang Zhou. Multilingual sentiment analysis: State of the art and independent comparison of techniques. *Cognitive Computation*, 8(4):757–771, Aug 2016.
- [12] Kushal Dave, Steve Lawrence, and David M. Pennock. Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. In *Proceedings of the 12th International Conference on World Wide Web, WWW ’03*, pages 519–528, New York, NY, USA, 2003. ACM.
- [13] Scott Deerweste, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. Indexing by latent semantic analysis. *Journal of the american society for information science*, 41(6):391–407, 1990.
- [14] Andrea Esuli and Fabrizio Sebastiani. Sentiwordnet: a high-coverage lexical resource for opinion mining. Technical Report ISTI-PP-002/2007, Institute of Information Science and Technologies (ISTI) of the Italian National Research Council (CNR), October 2006.
- [15] Noura Farra, Elie Challita, Rawad Abou Assi, and Hazem M. Hajj. Sentence-level and document-level sentiment mining for arabic texts. *2010 IEEE International Conference on Data Mining Workshops*, pages 1114–1119, 2010.

- [16] Lei Feng, Jose López, Li Feng, Sheng Zhang, Bormin Huang, and Fang Fang. Topic modeling of environmental data on social networks based on ed-lda. *international Journal of Environmental Monitoring and Analysis*, 6(3):77–83, 2018.
- [17] R. Fernandez-Beltran, A. Plaza, J. Plaza, and F. Pla. Hyperspectral unmixing based on dual-depth sparse probabilistic latent semantic analysis. *IEEE Transactions on Geoscience and Remote Sensing*, 56(11):6344–6360, Nov 2018.
- [18] G. W. Furnas, S. Deerwester, S. T. Dumais, T. K. Landauer, R. A. Harshman, L. A. Streeter, and K. E. Lochbaum. Information retrieval using a singular value decomposition model of latent semantic structure. In *Proceedings of the 11th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '88, pages 465–480, New York, NY, USA, 1988. ACM.
- [19] Gregory Grefenstette, Yan Qu, James G. Shanahan, and David A. Evans. Coupling niche browsers and affect analysis for an opinion mining application. In *Coupling Approaches, Coupling Media and Coupling Languages for Information Retrieval*, RIAO '04, pages 186–194, Paris, France, France, 2004. Le centre de hautes études internationales d’informatique documentaire.
- [20] Fatemeh Hemmatian and Mohammad Karim Sohrabi. A survey on classification techniques for opinion mining and sentiment analysis. *Artificial Intelligence Review*, Dec 2017.
- [21] Minqing Hu and Bing Liu. Mining and summarizing customer reviews. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '04, pages 168–177, New York, NY, USA, 2004. ACM.
- [22] A. Jeyapriya and C. S. Kanimozhi Selvi. Extracting aspects and mining opinions in product reviews using supervised learning algorithm. *2015 2nd International Conference on Electronics and Communication Systems (ICECS)*, pages 548–552, 2015.
- [23] Nitin Jindal and Bing Liu. Identifying comparative sentences in text documents. In *Proceedings of the 29th Annual International ACM SIGIR Conference on*

- Research and Development in Information Retrieval*, SIGIR '06, pages 244–251, New York, NY, USA, 2006. ACM.
- [24] Nitin Jindal and Bing Liu. Mining comparative sentences and relations. In *Proceedings of the 21st National Conference on Artificial Intelligence - Volume 2*, AAAI'06, pages 1331–1336. AAAI Press, 2006.
- [25] April Kontostathis and William M. Pottenger. A framework for understanding latent semantic indexing (lsi) performance. *Inf. Process. Manage.*, 42(1):56–73, January 2006.
- [26] Julia Kreutzer and Neele Witte. Opinion mining using sentiwordnet. Technical report, Uppsala University, Uppsala, Sweden, 2013.
- [27] Zuhe Li, Yangyu Fan, Bin Jiang, Tao Lei, and Weihua Liu. A survey on sentiment analysis and opinion mining for social multimedia. *Multimedia Tools and Applications*, Aug 2018.
- [28] Bing Liu. *Sentiment Analysis and Opinion Mining*. Morgan & Claypool Publishers, 2012.
- [29] Bing Liu, Mingqing Hu, and Junsheng Cheng. Opinion observer: Analyzing and comparing opinions on the web. In *Proceedings of the 14th International Conference on World Wide Web*, WWW '05, pages 342–351, New York, NY, USA, 2005. ACM.
- [30] Bing Liu and Lei Zhang. *A survey of opinion mining and sentiment analysis*, pages 415–463. Springer US, 8 2013.
- [31] Rodrigo Moraes, Joao Valiati, and Wilson Gavião Neto. Document-level sentiment classification: An empirical comparison between svm and ann. *Expert Systems with Applications*, 40:621–633, 02 2013.
- [32] Satoshi Morinaga, Kenji Yamanishi, Kenji Tateishi, and Toshikazu Fukushima. Mining product reputations on the web. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '02, pages 341–349, New York, NY, USA, 2002. ACM.

- [33] Tetsuya Nasukawa and Jeonghee Yi. Sentiment analysis: Capturing favorability using natural language processing. In *Proceedings of the 2Nd International Conference on Knowledge Capture, K-CAP '03*, pages 70–77, New York, NY, USA, 2003. ACM.
- [34] Bruno Ohana and Brendan Tierney. Sentiment classification of reviews using sentiwordnet. In *9th. IT&T Conference, EMNLP '02*, pages "22–23", Dublin, Ireland, 01 2009. Dublin Institute of Technology.
- [35] Rafeeqe Pandarachalil, Selvaraju Sendhilkumar, and G. S. Mahalakshmi. Twitter sentiment analysis for large-scale data: An unsupervised approach. *Cognitive Computation*, 7(2):254–262, Apr 2015.
- [36] Bo Pang and Lillian Lee. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42Nd Annual Meeting on Association for Computational Linguistics, ACL '04*, Stroudsburg, PA, USA, 2004. Association for Computational Linguistics.
- [37] Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. Thumbs up?: Sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing - Volume 10, EMNLP '02*, pages 79–86, Stroudsburg, PA, USA, 2002. Association for Computational Linguistics.
- [38] Soujanya Poria, Erik Cambria, Grégoire Winterstein, and Guang-Bin Huang. Sentic patterns: Dependency-based rules for concept-level sentiment analysis. *Knowl.-Based Syst.*, 69:45–63, 2014.
- [39] Mohammed Rushdi-Saleh, María Teresa Martín-Valdivia, Arturo Montejo Ráez, and Luis Alfonso Ureña López. Experiments with svm to classify opinions in different domains. *Expert Syst. Appl.*, 38:14799–14804, 2011.
- [40] Dharmendra Sharma and uresh Jain. Evaluation of stemming and stop word techniques on text classification problem. *International Journal of Scientific Research in Computer Science and Engineering*, 3:1–4, 2015.

- [41] V. S. Subrahmanian and Diego Reforgiato. Ava: Adjective-verb-adverb combinations for sentiment analysis. *IEEE Intelligent Systems*, 23(4):43–50, July 2008.
- [42] Dilipkumar A. Borikar Tanvi Hardeniya. Dictionary based approach to sentiment analysis - a review. *International Journal of Advanced Engineering, Management and Science*, 2, 2016.
- [43] Tsegaye Misikir Tashu and Tomas Horvath. Pair-wise: Automatic essay evaluation using word mover’s distance. In *Proceedings of the 10th International Conference on Computer Supported Education - Volume 2: CSEDU*, pages 59–66, 2018.
- [44] Richard Tong. An operational system for detecting and tracking opinions in on-line discussions. In *Working Notes of the SIGIR Workshop on Operational Text Classification*, pages 1–6, New Orleans, Louisiana, 2001.
- [45] Peter D. Turney. Thumbs up or thumbs down?: Semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL ’02, pages 417–424, Stroudsburg, PA, USA, 2002. Association for Computational Linguistics.
- [46] Binita Verma and Ramjeevan Singh Thakur. Sentiment analysis using lexicon and machine learning-based approaches: A survey. In Basant Tiwari, Vivek Tiwari, Kinkar Chandra Das, Durgesh Kumar Mishra, and Jagdish C. Bansal, editors, *Proceedings of International Conference on Recent Advancement on Computer and Communication*, pages 441–447, Singapore, 2018. Springer Singapore.
- [47] Janyce Wiebe. Learning subjective adjectives from corpora. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence*, pages 735–740. AAAI Press, 2000.
- [48] Qiang Ye, Ziqiong Zhang, and Rob Law. Sentiment classification of online reviews to travel destinations by supervised machine learning approaches. *Expert Syst. Appl.*, 36(3):6527–6535, April 2009.

BIBLIOGRAPHY

- [49] Li Zhuang, Feng Jing, and Xiaoyan Zhu. Movie review mining and summarization. In *CIKM*, 2006.