EMOTION MINING FROM TEXT AND ACTIONABLE PATTERN DISCOVERY

by

Jaishree Ranganathan

A dissertation submitted to the faculty of
The University of North Carolina at Charlotte
in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in
Computing and Information Systems

Charlotte

2020

Approved by:

Dr. Angelina A. Tzacheva

Dr. Zbigniew W. Ras

Dr. Jing Yang

Dr. Moutaz Khouja

ABSTRACT

JAISHREE RANGANATHAN. Emotion Mining From Text and Actionable Pattern Discovery. (Under the direction of DR. ANGELINA A. TZACHEVA AND DR. ZBIGNIEW W. RAS)

In the era of Web 2.0, people express their opinion, feelings and thoughts about topics including political and cultural events, natural disasters, products and services, through mediums such as blogs, forums, and micro-blogs, like Twitter. Also, large amount of text is generated through e-mail which contains the writer's feeling or opinion; for instance, customer care service e-mail. The texts generated through such platforms are a rich source of data, which can be mined in order to gain useful information about user opinion or feelings. Sentiment Analysis identifies and extracts information about the attitude of a speaker or writer on a subject, topic, polarity, or emotion in a document. Sentiments can be extracted from sources such as speech, music, and facial expression. Due to rich source of information available in the form of text data, we focus on sentiment analysis and emotion mining from text. We further discover Actionable Patterns from these sentiments, which suggest ways to alter the user's emotion to a more positive or desirable state. Little work has been done on extracting Actionable Recommendations from sentiments. We contribute to the solution of this challenging problem by applying machine learning methods such as decision forest, and support vector machines for emotion classification, and Action Rules Mining for Emotion altering recommendations. We experiment with live streaming Twitter Data, Student Evaluations data, Business Net Promotor Score data. Results show high accuracy for Emotion Detection, and successful discovery of Actionable Recommendations for more positive user sentiment. Applications of this work include: marketing, sale predictions, political surveys, health care, student-faculty culture, e-learning platforms, and social networks.

ACKNOWLEDGEMENTS

I would like to express my deep gratitude to my advisor Dr. Angelina A. Tzacheva for her valuable and constructive suggestions through out my research development process, patient guidance and enthusiastic encouragement. I would also like to thank my co-advisor Dr. Zbigniew W. Ras for his support, suggestions and guidance. I would like to extend my thanks to my committe members Dr. Jing Yang for her valuable feedback and fruitful comments and Dr. Moutaz Khouja for his support and guidance.

I thank my collaborator Arunkumar Bagavathi for his contribution and sharing resources for my research. I would also extend my thanks to Allen S. Irudayaraj, Nikhil Hedge, Sai Yesawy Mylavarapu, Rajendara Jadi, Sagar Sharma, Poonam Rajurkar, Majunath Shrishail Birajdar for helping with technical development and experimental results.

I wish to thank the technology solutions team Jonathan Halter, Chuck Price and Joseph Matesich for their support and administration of the Apache Spark on Hadoop cluster at UNCCharlotte.

Special thanks to Dr. Saif Mohammed from National Research Council for providing the National Research Council - NRC Emotion Lexicon.

I would like to extend my deep gratitude to the Office of Assessment at UNC Charlotte for their funding of student evaluations project.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1: INTRODUCTION

1.1 Human Emotions

Theorist classified emotion into two major categories discrete model and dimensional model. Discrete emotion model Fig. 1.1 states that specific core emotions are sub served by independent neural system, on the other hand dimensional model Fig. 1.2 states that all affective states or emotion arise from cognitive interpretations of core neural sensations where valence and arousal are the fundamental components of emotion. [4] [5].

Ekman [7] characterizes emotion into 6 basic forms as sadness, disgust, enjoyment, anger, fear, surprise. Plutchik et al. [8] agreed with Ekman's biologically driven perspective but developed the wheel of emotions on bipolar axes: joy versus sadness, anger versus fear, trust versus disgust and surprise versus anticipation. Shaver et al. [9] model a hierarchical tree structure for the basic emotions love, joy, surprise, anger, sadness, and fear and the leaves of the tree contain further categorization for each of these six basic emotions. Author Lovheim [10] presents a three-dimensional model for monoamine neurotransmitters and emotions. In this model, the monoamine systems are represented as orthogonal axes and the eight basic emotions, labeled according to [11] [12], are placed at each of the eight possible extreme values, represented as corners of cube.

There are many dimensional models for emotion, following are the widely accepted models as suggested by [13]: circumplex model, vector model, and Positive Activation - Negative Activation model. According to [14], circumplex model suggests that emotions are distributed in a two-dimensional space, containing arousal and valence dimensions. Authors Bradley et al. [15], propose a two-dimensional model in which

Figure 1.1: Discrete Model - Emotions. [6]

the base dimension is arousal and that the valence determines the direction in which the emotion lies. Authors Watson and Tellegen [16], develop the Positive Activation - Negative Activation model in which they suggest that positive affect and negative affect are two separate systems.

1.2 Importance and Applications of Human Emotion

Origin and etymology of emotion dates back to the year *1579*, it is known to have originated from the French word 'emouvoir' which means 'to stir up'. According to [17], 'emotion' is introduced into academic discussion to replace 'passion'. Emotion is one of the aspects of our lives that influences day-to-day activities including social behavior, friendship, family, work, and many others. Emotion mining has its root in many disciplines apart from computer science as follows: human science, psychiatry, nursing, psychology, neuro-science, linguistics, social science, anthropology, communication science, economics, criminology, political-science, philosophy etc.

Emotion mining gained attraction in the field of computer science due to the vast variety of systems that can be developed and promising applications. Some examples

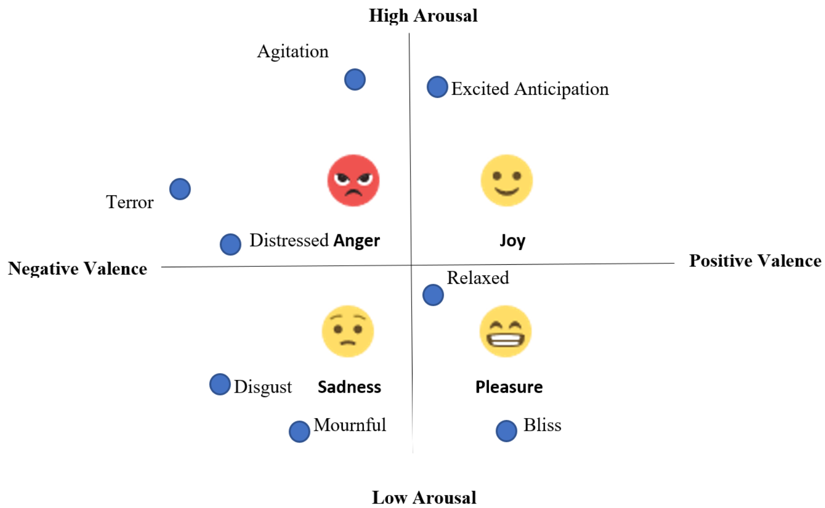

Figure 1.2: Dimensional Model Emotions.

from the literature includes: *customer care services* [18] where idetifying features such as frustration, dis-satisfaction etc., provide useful information to businesses on how to improve their existing services and to have better relationship with customers, which in turn, increases customer loyalty and retention there by improving revenue. In *Human Computer Interaction* [19], remote health care system where hands on care is not required and assess the mental and emotional state, suggest music based on human emotions. *Psychologists* can infer patients' emotions and predict their state of mind accordingly. On a longer period of time, they are able to detect if a patient is facing depression or stress [20] or even thinks about committing suicide, which is extremely useful, since they can be referred to counseling services [21]. *Social Network Analysis*, Social Network Data, such as Twitter, to assess the sentiment and the overall emotion of Tweets, as well as to analyze events [22]. *Education*, Identifying student emotions helps improve teaching methodologies and or quality, improve resources available and classroom environment [23], [24], [25], [26], [27].

1.3 Social Media Text - Sentiment Mining and Actionable Pattern Discovery

Social interaction websites like Facebook, Flickr, and Twitter have added a new dimension to the social life of internet-aware people. This trend provides a huge amount of raw data that can be processed to generate structured and useful information. Data mining promises to discover valid and potentially useful patterns in data. Often, discovered patterns are not useful to the user. Actionability addresses this problem in that a pattern is deemed actionable if the user can act upon it favourably. Actionable patterns in most cases can be created through rule reduction, model refinement, or parameter tuning by optimizing generic patterns [28]. Actionable patterns are revised optimal versions of generic patterns that capture deeper characteristics and understanding of the business and are also called in-depth or optimized patterns. Action Rules are specific patterns extracted from large datasets. To generate Action Rules, the attributes in the dataset are split into two groups called Flexible Attributes and Stable Attributes. Flexible attributes are those for which the state can change, and the Stable Attributes are those for which the state is always fixed. An association Action Rule is a rule extracted from an information system that describes a cascading effect of changes of attribute values listed in the left-hand side of a rule [29] on changes of attribute values listed in its right-hand side. Generating Action Rules based on a classification rules are expensive. This paper makes use of the ARAS algorithm proposed by Ras et al. [30], [31]. ARAS Action Rules Discovery Based on Grabbing Strategy which uses LERS - combines each Action Rule generated from single classification rule with the remaining stable attributes to offer more Action Rules. This work discovers more Action Rules as compared to the previous algorithms. The use of LERS in the pre-processing module for defining classification rules serves to decrease the complexity of ARAS algorithm. Sentiment Analysis is the process of identifying the polarity, opinion or emotion expressed by human. In this work we use the Stanford core NLP suite [32] for extracting sentiment from Twitter data.

Data mining techniques are used to analyze huge data sets, to identify the underlying data patterns and to reveal the hidden knowledge. Data digitization in social networking and the extensibility of the platform for social networking, from micro devices like watches and smart phones to macro devices like desktops and laptops, have greatly contributed to the huge amount of structured and unstructured data that can be processed to generate sensible and meaningful information. Action-ability extends the concept of data analysis to a level further, by which the user can attain his/her intended action through deducing the Action Rules from the dataset.

The attributes in a dataset are divided into flexible attributes, whose value is mutable, and stable attributes, whose values is immutable [33]. The Action Rules are specific data patterns extracted from huge dataset which intends to change the current value of the flexible attribute, under consideration, to a desired value. An association rule is a rule extracted from an information system that describes a cascading effect of changes of attribute values listed in the left-hand side of a rule [29] on changes of attribute values listed in its right-hand side.

New algorithms have been proposed in the past decade to find some special actions based on the discovered patterns in the form of Action Rules. Action Rules propose an actionable knowledge that the user can undertake to his/her advantage. An Action Rule extracted from a decision system describes a possible transition of an object from one state to another state with respect to the distinguished attribute called decision attribute [34]. Action Rules have established its applications in variety of industries like healthcare, automotive, advertising etc. Some businesses require Action Rules generation on batch data and some require the same on streaming data. Hence cost of a time is the critical parameter to be considered for the algorithms which are proposed for generating these Action Rules. Authors in [35], [36], [34], [37], [38], [30] proposed variety of algorithms to extract Action Rules from the given dataset. The eccentric exponential increase in the data in recent years, causes delay in computations on tasks that are dependent on Action Rules and thus causing applications relied on Action Rules to be slow. Hence, this mandate need to develop viable, scalable, time efficient and distributed methods to work on such huge volume of data for generating action.

Distributed database systems are the most appropriate system to handle huge data sets. They have substantiated the reliability and efficiency for storage and processing bulk data sets. Apache provides various open source like Hadoop [39], Spark [40], [41], Hive, and Pig to process and handle huge data in the distributed system. Hadoop is a distributed computing framework, to work with large datasets, across multiple computers, using a single programming model in a parallel fashion. This parallel processing aspect of the distributed computing plays a vital role in the cost of the processing time. Hadoop aims to provide scalable and fault tolerant computations on the given data. The main components of Hadoop are HDFS [42], YARN [43] and MapReduce [39].

Hadoop Distributed File System - HDFS is the data storage unit of the MapReduce operations. HDFS also keeps track of machines holding the data for a job [42]. Yet another Resource Negotiator - YARN [43] is an extra feature to the upgraded version Apache Hadoop framework. YARN supports multiple applications like MapReduce, Spark [41], [40], Storm, etc.

MapReduce is an open source cluster computing framework which uses HDFS to save and process huge data sets. The MapReduce framework works in such a way that it divides the input data into size mutable input splits and cascades them to the clusters [Hadoop performance prediction]. By default, the input splits are 64MB individually. The MapReduce works in 2 phases, map and reduce. In the map phase, the input splits are processed in parallel fashion in the cluster and the intermediate results are stored in the cluster. In the reduce phase the intermediate results are combined and saved in the HDFS. The frequent access to the HDFS system makes it less suitable for iterative algorithms, which might require more map and reduce cycles.

Apache Spark addresses the issue with the concept of Resilient Distributed Dataset. Its in-memory data operations makes it well-suited for applications involving iterative machine learning and graph algorithms. Thus, we move our algorithm to Spark framework on top of Hadoop Distributed File System (HDFS) cluster. In this paper, we present a system SARGS (Specific Action Rule discovery based on Grabbing Strategy) which is an alternative to ARoGS [38] and implement the system in Spark like our old system MRRandom Forest algorithm for Distributed Action rule Discovery [44] using Hadoop MapReduce, either of them to extract Action Rules from the twitter data in the HDFS. The primary intent of the Action Rules generated is to provide viable suggestions to make a twitter user positive. Finally, we compare our current proposed system against our old Hadoop system of extracting Action Rules.

1.4 Social Media Text - Emotion Mining and Actionable Pattern Discovery

Twitter is one of the popular social networking site with more than 320 million monthly active users and 500 million tweets per day. Tweets are short text messages with 140 characters, but are powerful source of expressing emotional state and feelings with the society of friends. According to author Fox [45] emotion is discrete and consistent response to internal or external events that have a significance for the organism. Emotion is one of the aspects of our lives that influences day-to-day activities including social behavior, friendship, family, work, and many others. There are two theories related to human emotions: discrete emotion theory and dimensional model. Discrete emotion theory states that different emotions arise from separate neural systems, dimensional model states that a common and interconnected neurophysiological system is responsible for all affective states [46].

Textual emotion mining has quite lot of applications in today's world. The applications include modern devices which sense person's emotion and suggest music, restaurants, or movies accordingly, product marketing can be improved based on user comments on products which in turn helps boost product sales.

Other applications of textual emotion mining are summarized by Yadollahi et.al [46] and include: in customer care services, emotion mining can help marketers gain information about how much satisfied their customers are and what aspects of their service should be improved or revised to consequently make a strong relationship with their end users [18]. User's emotions can be used for sale predictions of a particular product. In e-learning applications, the intelligent tutoring system can decide on teaching materials, based on user's feelings and mental state. In Human Computer Interaction, the computer can monitor user's emotions to suggest suitable music or movies [19]. Having the technology of identifying emotions enables new textual access approaches such as allowing users to filter results of a search by emotion. In addition, output of an emotion-mining system can serve as input to other systems. For instance, Rangel and Rosso [47] use the emotions detected in the text for author profiling, specifically identifying the writer's age and gender. Last but not least, psychologists can infer patients' emotions and predict their state of mind accordingly. On a longer period of time, they are able to detect if a patient is facing depression or stress [20] or even thinks about committing suicide, which is extremely useful, since he/she can be referred to counseling services [21]. Though this automatic method might help in detecting psychology related issues, it has some ethical implications as it is concerned with human emotion and their social dignity. In such cases it is always ethical to consult human psychiatrist along with the automatic systems developed.

Emotion classification is automated using supervised machine learning algorithms. Supervised learning involves training the model with labeled instances and the model classifies the new test instances based on the training data set. Most of the previous works in this area of emotion mining [48] and [49] have used manual labeling of training data set. Authors Hasan et. al. [50] use hash-tags as labels for training data

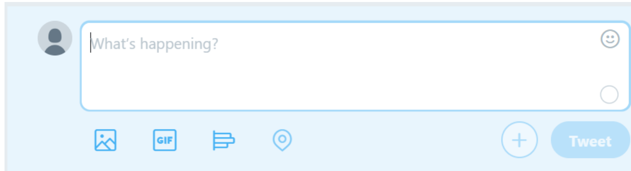

Figure 1.3: Twitter - User tweet template for post.

set. This work focuses on automatically labeling the data set and then use the data for supervised learning algorithms.

The previous works [48] [50] [49] have developed text classification algorithms like k-nearest neighbor and support vector machines. In this work, we use Decision Tree, Decision Forest , Rule-based Decision Table Majority , and also a Recurrent Neural Network classifiers for automatic emotion classification.

According to Merriam Webster dictionary [51] Micro-blogging is blogging done with severe space or size constraints typically by posting frequent brief messages about personal activities. There are wide variety of such micro-blog services available on the web including Twitter [52], Tumblr [53], Pownce (http://pownce.com) and many others. Among these Twitter is the most popular. According to ComScore [54], within eight months of its launch, Twitter had about 94,000 users as of April 2007 [55]. Also, micro-blogging user's may post several updates on a single day states authors Java et al. [55]. Approximately 500 million tweets are posted on Twitter per day. Thus the amount of textual data generated is huge when we consider the rate of growth of Twitter user's since 2007 and the periodicity of the posts on a single day by a user. Fig 1.3. shows the user template for posting tweets on the Twitter home page. As we can observe that it allows adding emoticons which are one of the powerful tools to express human emotions. Hashtag is a tagging convention that helps people associate tweets with certain events or contexts [56]. It is a keyword prefixed with '#' symbol. These hashtags sometimes indicate the writer's emotion. For example the tweet "Home made chicken soup is the best #happy indicates happiness [50].

Data Mining from such rich sources of text helps gain useful insights in a range of applications. For instance, authors Gupta et al. [18] study the customer care email in-order to identify customer dissatisfaction and help improve business. Analyzing the social media posts of a particular community might help government officials in public policy making to improve the quality of life of people in that area. In educational domain, identifying student's thoughts and emotion about the university, faculty helps improve the quality of education, In the field of psychology where online social therapy is used for assisting mental health as face-to-face early intervention services for psychosis is for limited time period and benefits may not persist after its termination [57] and in scenarios where machines are used as psychotherapist [58]. After information is gathered from such data, it is necessary to validate the mined information. For this purpose there are many supervised learning models that help automatically classify new set of test data, given a considerable amount of data for training.

With the proliferation of information through various sources there is access to enormous of data, at the same time leads to poor information in the raw form and inefficient decision making [59]. The volume of discovered patterns is huge despite the use of data mining strategies which leads to unreliable and uninteresting knowledge [59]. Actionable patterns are those that help users benefit by using it to their own advantage. Action Rules are special type of rules that help identify actionable patterns from the data [60].

Emotion is a combinatorial result of a persons evaluation of a situation along with the physiological arousal [61]. Classification of human emotions has been long under research since the early 60's. One of the earliest studies [7] suggests 'anger', 'fear', 'enjoyment', 'sadness', 'disgust', 'surprise' as the basic emotions Fig. 1.2 universal for all human beings. Any kind of information available holds a '*meaning*' to it. *Meaning* has associated *Connotation* and *Denotation*. *Denotation* is the dictionary meaning, while *Connotation* is the emotional association to the information. For instance the words 'childish' and 'childlike' both have identical denotations, whereas the former has a more insulting connotation when refered to a person and the latter has more positive meaning. According to researchers *Connotation* has gained more attention with psychology. Thus emotion plays an important role in our understanding of the information entities in day to day life. Internet naturally poses a wider platform for expressing this kind of emotional information. One such prominent platform is 'Social Media' which includes online social networks like 'Facebook' and microblogs like 'Twitter' [62].

We use the dataset formed of messages collected from the popular microblogging platform Twitter. Users of this platform tend to share their opinion or feelings about political events, natural disaster, products and companies, environment, community and much more on a day-to-day basis. This kind of user information is of interest to different communities. For instance, tweets about environment, community from a particular county and state will help the government officials understand the standard of living of people. This helps them make better amendments to the policies for the well being of the people.

In this work, we utilize Weka Affective Tweets package [63] to explore the tweet text characteristics such as the lexicon features, semantic features and word embeddings which are considered to improve the classification model performance [64], [65], [66].

Knowledge Discovery is the process of extracting interesting patterns and applying such patterns to specific areas of interest. To find such interesting patterns from data there are wide range of techniques likes classification, rule mining, clustering etc. Source of data in the era of internet has become huge with five V's *(Volume, Variety and Velocity, Veracity and Value)*of defining big data. With such huge amount of data being generated from variety of sources it is obvious that the discovered knowledge might not be of much use to applications which are in need of specific suggestions including: smart phones, remote health care systems, customer care services, pedagogical approaches for teaching (student teacher evaluation), and e-learning. One of the major problem in the knowledge discovery process is reducing the volume of discovered patterns and selecting the appropriate interestingness measure [67]. One of the techniques to extract appropriate interseting patterns is action rules proposed by authors Ras and Wieczorkowska [34]. Action rules are special types of rules which forms a hint to the users, show a way to reclassify objects with respect to some distinguished attribute called the decision attribute. Emotion mining gained attraction in the field of computer science due to the vast variety of systems that can be developed and promising applications like remote health care system, customer care services, smart phones that react based on user's emotion, vehicles that sense emotion of the driver. Emotions mined from text data is vast, in order to make sensible suggestions we use action rules that provide valuable knowledge on how to alter the users'emotion to a better or more positive state.

According to Merriam-Webster dictionary [51], emotions is defined as a *conscious mental reaction (such as joy, anger or fear) subjectively experienced as strong feeling usually directed toward a specific object and typically accompanied by physiological and behavioral changes in the body.* This definition identifies emotions as constructs involving something innate that is often invoked in social interactions and that aids in communicating with others. Emotions are an integral part of human life, and it affects our everyday decisions, and well-being (mentally and physically). Emotion detection models are important and have wide range of applications in this fast paced internet world. Such applications include analyzing customer service data to help improve business and customer loyalty, analyzing social media post of community, which help government officials in public policy making to improve quality of life, in educational domain - identifying students emotion helps improve quality of education.

Emotion detection has remained a challenging task, partly due to the limited availability of labeled data and partly due to the controversial nature of what emotions themselves are [68]. Recent advances in machine learning for Natural Language Processing (NLP), suggest that given enough labeled data, there should be an opportunity to build better emotion detection models [69]. Manual labeling of data, is costly and it is important and desirable to develop automatic emotion labeling systems. In this work, we exploit the data using: Decision Tree, Decision Forest, Rule-Based Decision Table, and also a Recurrent Neural Network to build automatic emotion classification model.

1.5 Educational Data Mining - Emotion Mining

Quality of education is one of the primary factors which requires constant attention and improvement. Student evaluations of teaching serve as both formative and summative measure in the process of quality education. Literature dates back to 1920's [70] with the works of Remmers to assess the student evaluation agreements with alumni and peers [71], [72].

Student evaluation of teaching is an important element in the process of evaluating and improving instruction in higher education as described by Zabaleta [73]. These evaluations help not only in teaching improvements but also in some of the decisions like future employment, retention, and promotion of faculty. It is now-a-days common in almost any educational institution to collect end of course evaluation, which allows students to express their feelings or opinion about the instructor. These evaluations are collected at the end of course typically end of semester. There are basically two types of question format in the evaluation system: Quantitative and Qualitative. Quantitative questions are Likert-type items which the students can respond in the scale of 1 to 5, starting with Strongly Agree - 1, Agree - 2, Neutral - 3, Disagree - 4, and Strongly Disagree - 5. Qualitative questions are open ended questions where students can write their opinion, and/or thoughts in a free style manner. According to author Clayson [74] since the 1970's the application of student evaluation in teaching has become nearly universal.

Data Mining is one of promising fields which involves the practice of searching through large amounts of computerized data to find useful patterns [51]. These patterns are then utilized by analysts to find interesting measures and apply strategies to improve the current methodology or practices. According to author's Spooren et al. [23] there are three main purpose for which student evaluations are used as follows: a) improve teaching methodology and/or quality, b) serve as input for tenure/ promotion decisions, and c) Demonstrate the evidence of institutional accountability in terms of resources and environment provided. Mining this kind of educational data is one of the important areas of research which is gaining importance in recent years due to increase in the demand of quality education and the demography of students attending higher education. Most of the students in recent years are Millennials and their mindset towards education is different which requires better understanding from University and the Instructors in order to provide a better experience in education.

In recent years there is an increase in the need for understanding what is said about a element. For instance, in an online store, customer reviews about a product - where customers convey their opinion about the quality and usefulness of the product and how well it suits their expectation. These kind of reviews helps business analyst improve their marketing strategies and apply to the quality of the products. Understanding people's feeling or emotion is a separate area of research which is called Sentiment Analysis.

The word Emotion dates to 1570's, derived from old French 'emouvoir' meaning 'stir up' according to online Etymology dictionary. Scientific research in understanding Human Emotion's dates to 1960's. For instance, Ekman [7] studied human emotions and their relation to facial expressions. According to Ekman there are six basic emotions 'anger', 'disgust', 'fear', 'joy', 'sadness', and 'surprise'. Similarly, there are other scientists who proposed emotion theories, Author James [75] and Plutchik [76]. In [77], the authors discuss different basic emotion models proposed by theorists since 1960. In this work we use the National Research Council - NRC Emotion lexicon [78], [79].

Emotion is a primary concern in younger generation students that have major impact on the productivity in school. The emotional influence does not stop at high school or university but may have lifelong consequences in future career outcomes. In psychology, emotion is often defined as a complex state of feeling that results in physical and psychological changes that influence thought and behavior. It is associated with a range of psychological phenomena, including temperament, personality, mood, and motivation. How students feel or their emotion towards a classroom, teaching style, and learning approach helps motivate them to achieve better outcomes. There is an increasing effort by Universities all over the world to collect student feedback. Besides various limitations, the student survey of teaching and learning provides valuable insights [80], [81], [82], [83].

Education is considrered to be an indispensible need in today's world. It is continuously evolving to meet the challenges of the fast-changing and unpredictable globalized world. There is a lot of importance and attention paid to improve students educational outcomes throughtout the world [84]. Therefore the educational institutions and the Instructors are expected to innovate the theory and practice of teaching and learning, and other aspects of the organization to ensure quality preparation of all students to life and work [85]. In 1964 the book "Innovation in Education" [86], states that changes and revolution are in progress in Education. It is almost 55 years since then, even now it is of high demand that Education at all level needs renewal [85]. According to Merriam Webster Dictionary Innovation is the introduction of a new idea, or change made to existing idea. When we think of innovation in terms of education, it can be applied as a teaching technique, pedagogical approach, learning style or process, and institutional structure.

Active Learning is one such pedagogy or approach that is gaining attention and popular in Higher education. Lightweight teams is an Active Learning approach where students work together in a group, but they have very little or no direct impact on their final grades [87]. There are lot of works on this area, but none of them use the psychological perspespective of identifying student emotions and identifying patterns to suggest how to enhance student emotions.

This innovative pedagogical approach has been studied in Computer Science undergraduate courses and has been reported to have high levels of student engagement [87], [88].

In this work we focus on mining student feedback collected from the end-of-semester course evaluations, in particular the qualitative results and identify student's emotion to understand whether incorporation of Light Weight teams [87], [88], and Flipped Classroom techniques [89] helped students during the course for the time period 2013 to 2017.

We also propose a novel approach of using student feedback for courses labeled with emotion and provide suggestions on how to enhance emotions. This in turn leads to better teaching style, learning outcomes and a comprehensive environment. For this purpose we use actionable pattern discovery method.

Educational Data Mining is a new field which involves identifying patterns of student behaviours and learning by use of Machine Learning and Data Mining technologies. Neural Networks in Data Mining is a mathematical model which has its roots in biological neural network. Neural networks have achieved impressive results in several classification tasks with Twitter dataset [66], [90], [91], and 20Newsgroups, Fudan Set, ACL Anthology Network, and Sentiment Treebank datasets [92]. It is widely perceived that Neural Networks performs well with huge volume of data. Since student evaluations of teaching has limited data availability considering the number of students registering for a course, very limited works have used Neural networks in the education data mining field. Researchers use the classical machine learning models like Naive Bayes, Support Vector Machine for sentiment classification of student evaluations data. In this work we use sequential learning model on the student feedbacks for emotion classification and compare with the traditional models.

1.6 Big Data

Massive volume of data is created every day by variety of sources including mobile phones, sensors, interaction of people over computer applications such as: social media platforms, medical records, emails, online transactions. Data is growing at unprecendented rate, with not just streams of data but also entirely new data every time. Big Data Analysis requires a leap forward from traditional data analysis. It concerns three important issues: variety, volume, and velocity Fig. 1.4 [93]. The variety and velocity of incoming data, leads to huge volume of data that can no longer be processed by traditional methods in Data Mining. Big Data has generated a whole new industry supporting suppoting architectures for distributed and cluster computing frameworks like Hadoop [42], [43] and Apache Spark [94]. In this work we adapt our algorithms to process Big Data and experiements are performed using our University Research Cluster.

1.7 Attribute Selection

Data is distinct pieces of information that contains several features or attributes. Most of the real world data comes with number of attributes attached that are formatted in a specific way and gives meaning to the data. For example - medical data can be high-dimensional with different parameters like symptoms, treatment, diagnosis, disease, blood pressure, blood type, medication, admission date, patient information, and many more. The more attributes the data contains, it becomes difficult when dealing with machine learning, data mining, statistical, and or pattern

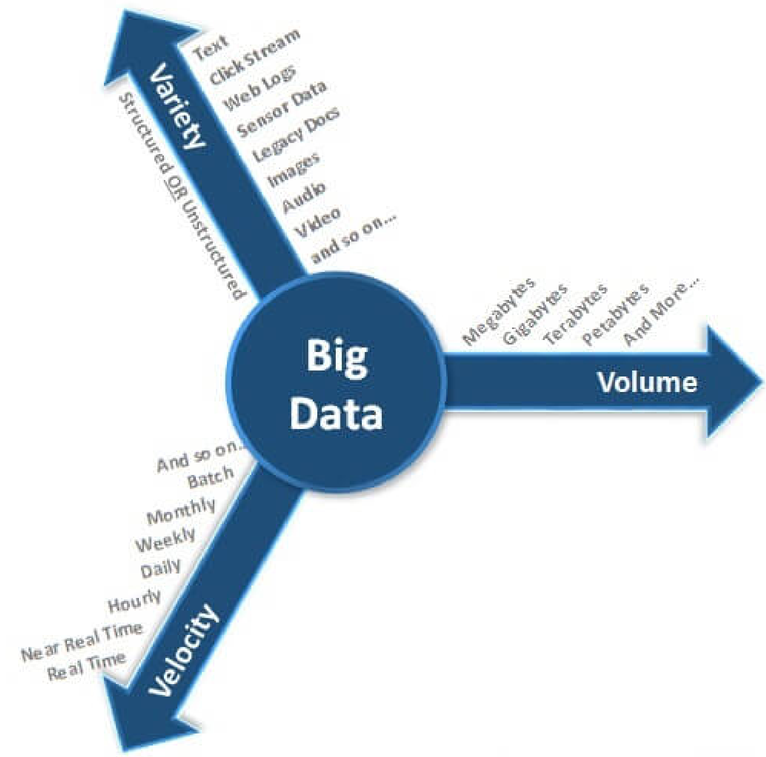

Figure 1.4: Big Data.

mining algorithms [95], [96].

Thus feature selection has become one of the most important technique in data processing. Some of the attributes in the data may be superfluous or redundant which degenerates the learning algorithm performance. Feature selection is the process of finding a subset of features (that are not redundant or superfluous) from set of all features in the given dataset, while preserving the inherent meaning of the data. Such attributes which fully characterize the knowledge in the database, are called Reducts [97]. This method evaluates the features based on measures such as dependency, distance, information gain, and similarity which are independent of the learning algorithm [98] is called Filter methods, which are considered to be efficient and faster.

Rough set theory [99] is a mathematical approach to deal with lack of certainity or distinctness. Rough set theory provides filter based mathematical frameworks for

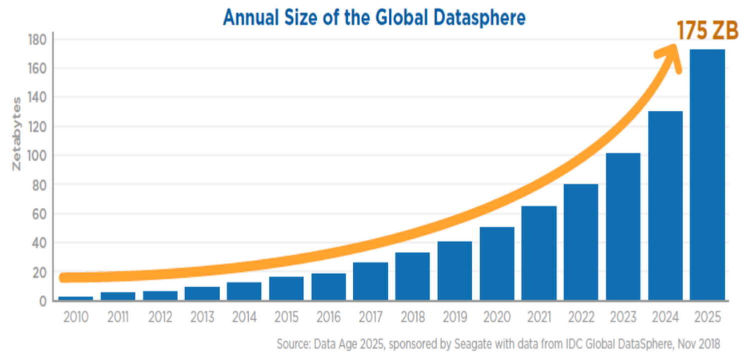

Figure 1.5: Annual Size of Global Datasphere [2].

dimensionality reduction in datasets that uses standard operations in conventional set theory. It uses the existing features in the data and does not require additional parameters to operate. Many methods are for feature selection have been proposed using Rough Sets. Discernibility matrix [100], [101] has been used to generate all possible reducts for the dataset. We are in the era of big data, with astonishing rate of growth of the data. According to International Data Corporation (IDC) [2], worlds data growth rate is at 66% per year which is expected to reach approximately equivalent to 175 zettabytes by the year 2025 (Figure. 1.5).

To address the BigData challenge, in this work we use Greedy algorithm and Discernibility matrix based algorithm for attribute reduction in a distributed cloud frameworks like Spark [40] for scalability and efficiency.

CHAPTER 2: RELATED WORK

2.1 Sentiment Analysis

Authors A. Balahur et.al [102] employs hybrid approach, using supervised learning with Support Vector Machines Sequential Minimal optimization (Platt 1998) linear kernel, on unigram and bigram features, but exploiting as features sentiment dictionaries, emotion lists, slang lists and other social media emotion features for a lexicon based sentimental analysis on the twitter data. The analysis involves two phases, pre-processing and then sentiment classifications. The processed tweets are then passed through the sentiment classification module. Training models were developed on the cluster of computers using Weka data mining software.

Authors A. Agarwal et.al [103] performed sentimental analysis on the twitter data. As part of the paper, they primarily experimented three types of models, unigram model, a feature based model and a tree kernel based model for two classification tasks, binary task which classifies the sentiment to positive and negative and 3-way task which classifies the sentiment to neutral along with the positive and negative category. The twitter data is first preprocessed using emotion dictionary, acronym dictionary and stop word dictionary. The comparative analysis on the models by experiment proved that tree kernel and feature based models outperform the unigram baseline.

Authors A. Chellal et.al [104] proposed multi-criterion real time tweet summarization based upon adaptive method. This method provides new relevant and non-redundant information about an event as soon as it occurs. The tweets selection is based on the following three criterions: informativeness, novelty and relevance with regards of the user's interest which are combined as conjunctive condition. Experiments were carried out on TREC MB RTF-2015 [104] data set.

Authors Yu. Xu et.al [105] proposed methods to infer a user's expertise based on their posts on the popular micro-blogging site twitter. They proposed a sentiment-weighted and topic relation-regularized learning model. Sentiment intensity of a tweet is used to evaluate user's expertise and the relatedness between expertise topics is exploited to model inference problem. The following four common metrics were used for evaluation: accuracy, precision, recall and F1- score.

Authors F. Marquez et.al [106] proposed a simple model for transferring sentiment labels from words to tweets and vice versa by representing both tweets and words using feature vectors residing in the same feature space. Tweet centroid model developed in this paper outperformed the classification performance of the popular emoticon-based method for data labelling and better results than a classifier trained from tweets labelled based on the polarity of their words.

Authors M. Al-Ayyoub and I. Alsmadi [107] proposed a lexicon based sentiment analysis of Arabic tweets. This method is based on Sentiment Analysis and Opinion mining of social network data twitter feedbacks and comments. Unsupervised approach of sentiment analysis was applied which built a sentiment lexicon SA tool. This sentiment lexicon was built with about 120,000 Arabic terms and a SA tool based on predicate calculus.

2.2 Emotions in General Text

A Rule-based approach for recognizing affective communication in text messages in [108] use the following emotional states: ‘anger’, ‘disgust’, ‘fear’, ‘guilt’, ‘interest’, ‘joy’, ‘sadness’, ‘distress’, ‘shame’, and ‘surprise’; and communicative functions including: ‘greeting’, ‘thanks’, ‘posing a question’, ‘congratulation’, and ‘farewell’. In this work Neviarouskaya et al. build a special affect database including emoticons, acronym's, abbreviations, adjectives, nouns, verbs, adverbs, words representing communicative functions and interjections with MySQL 5.0. Human annotators manually label the affect database with emotion categories and intensity values. Their Affect Analysis Model consists of five stages each with manually created rules: symbolic cue analysis, syntactical structure analysis, word-level analysis, phrase-level analysis, and sentence-level analysis. Their system has certain limitations like dependency on the database, failure to disambiguate word meanings and process expression modifiers.

Authors Ho and Cao [109] use High-Order Hidden Markov model (HMM) for emotion detection from ISEAR dataset. The idea is to transform the input text into a sequence of events that cause mental states. Then automatically construct HMM based on the training dataset and generate the model to process the sequence of states that cause emotion. By cross-validation their model shows promising results.

Authors Mishne et al. [110] classify writer's mood in blog text collected from LiveJournal a free weblog service using Yahoo API. To ensure the proper balance of training data across all moods, they select blog posts containing one of 40 top occurring moods in the entire corpus. They contribute a significant part of the work towards feature selection. Some classic features like frequency counts (words, Part-Of-Speech), and length of blog post; subjective nature of blogs like semantic orientation, Point-wise Mutual Information (PMI) which is a measure of the degree of association between two terms; features unique to online text like emphasized words, special symbols including punctuation's, and emoticons were used for training the SVMlight model from Support Vector Machine package. They attribute subjective nature of the corpus 'annotation' and nature of blog posts as major factors for low accuracy.

Authors Strapparava et al. [111] implement five systems for emotion analysis for news headlines using knowledge-based and corpus-based approaches. They evaluate the systems on the dataset of 1000 newspaper headlines from SemEval 2007 by conducting fine-grained and coarse-grained evaluations. Results show that each of the systems have specific strength and they compare the results with three baseline systems in SEMEVAL emotion annotation task: SWAT [112], UPAR7 [113] and UA [114]. UPAR7 obtains best results in terms of fine-grained evaluations whereas the developed system using WordNet-Affect gives best performance in terms of coarse-grained evaluation with highest recall and F-measure.

Authors Gupta et al. [18] present a method for identifying emotional customer care emails using 'Boostexter' [115] classifier which is based on boosting family of algorithms. Many 'weak' moderately accurate base classifiers combined to build a highly accurate classifier in boosting. They also extract salient features from emotional emails which reflect customer frustration, dissatisfaction with the business, threats to leave or take legal action and/or report to authorities. Their results show that the 'Boostexter' system with salient features resulted in a 20% absolute F-measure improvement compared to the baseline system using word ngrams.

Authors Danisman et al. [116] develop text classifier using Vector Space Model (VSM) where each document is a vector and terms correspond to dimensions. The basic hypothesis in using VSM for classification is the contiguity hypothesis where documents in the same class form a contiguous space whereas regions of different class do not overlap. The weighting scheme tf-idf (Term Frequency - Inverse Document Frequency) is used to calculate each term weight values. They have analyzed the effect of emotional intensity and stemming to the classification performance. Results show that Vector Space Model performs equally well compared to other well-known classifiers Naive Bayes, Support Vector Machine and ConceptNet. In addition to the classification model they also developed emotion enabled video player which shows emotional state of the video based on the subtitles.

Authors Hancock et al. [117] contribute research towards identifying emotions during text-based communication. They conducted experiments with eighty undergraduate students in 40 same sex dyads. The results suggest that irrespective of gender, agreement with the conversation partner was more with positive affect expressers.

Also, the following strategies are used to differentiate between positive and negative emotional states: frequency of disagreement, punctuation, negative affect terms, amount of words used. Negation and exclamation points are the two major linguistic cues that helped predict the textual emotion. These findings support the *"Social Information Processing Theory"* [118] and provide reasonable insights on how to automatically extract emotions in a text-based communication.

Authors Kim et al. [119] evaluate categorical model and dimensional model for four affective states anger, fear, joy, and sadness. They use the following three emotional datasets with sentence-level emotion annotations: SemEval 2007 [120], International Survey on Emotion Antecedents and Reactions (ISEAR) [121], fairy tales [122] and apply Vector Space Model with dimensionality reduction variants (Latent Semantic Analysis, Probabilistic Latent Semantic Analysis, Non-Negative Matrix Factorization) and dimensional model in MATLAB. Though their experiments show categorical Non-Negative Matrix Factorization and Dimensional model have better performances, it is also inferred that either of the techniques perform well on generalized dataset.

Authors Kao et al. [123] provide a comprehensive survey on the existing research methods (earlier 2009) for emotion detection from text, identify their limitations and propose an integrated system to improve the emotion detection capabilities of the existing systems. They classify textual emotion detection into keyword-based, learning-based and hybrid (combination of keyword, learning methods and other components). According to them, keyword-based method lack the use of linguistic information to detect emotions, learning-based methods still need keywords in the form of features, hybrid methods though they outperform the previous two approaches still limited with the category of emotions. Based on the above studies they propose an integrated architecture that includes semantic analysis, ontology design of emotion models and adopting case-based learning approach.

Authors Chaumartin et al. Author [113] proposes a rule-based linguistic system

UPAR7 to detect the emotion and valence of news headlines (SemEval 2007 dataset). This system uses statistical analyzer (Stanford Parser) to tag word dependencies. Further they use their own enriched version of lexical resources like WordNet, WordNet-Affect, SentiWordNet. The system tries to find the main subject of the news title by using the dependency graph, contrasts, and accentuation's. The rule-based system identifies emotions with 89.43% accuracy and valence with 55% accuracy. However, the recall is low. The difference between the ac-curacies of emotion and valence is due to the fact that it is easier to detect emotions of individual words rather than valence which needs a global understanding of the sentence.

Authors [124] provide a short survey of existing methods in textual emotion detection like Kao et al. but their method lacks to show the actual works. They develop a statistical model i.e. vector space model for automatic emotion detection from text using a comprehensive dataset created from ISEAR, WordNet-Affect and WPARD datasets. Their model uses bag-of-words approach because of which it lacks the ability to consider the semantic and syntactic information from the text. Also, they do not provide statistical results to the developed model. But they have handled the negation using 'not' by first finding the emotion of the sentence without considering negation and after that generate the reverse of the resulting emotion.

Autors [125] present a lexicon-based approach towards social emotion detection. They designed a new algorithm for document selection which has positive effect on the performance of social emotion detection systems. After document selection they exploit words and Part-of-Speech(POS) features. POS helps alleviate the problems of emotional ambiguity of words and the context dependence of the sentiment orientations. Finally generate the emotion lexicon based on the features. They gathered 40,897 news articles assigned with ratings over 8 social emotions including touching, empathy, boredom, anger, amusement, sadness, surprise, and warmness. Their method outperforms the baseline methods of SWAT [112], Emotion-Topic Model

(ETM), and Emotion-Term Model (ET) [126] and [127].

Authors [128] determine the aggregate mood levels across large number of blog postings, i.e. classify the blog posts into one of forty most frequent moods. They estimate the moods of complete blogs by identifying textual features (discriminating terms) and use these features in the learning models to predict the mood intensity in each time slot. Discriminating terms are collected by applying log likelihood measure to quantify the divergence between term frequencies across different corpora. Because of the nature of the dataset they did not achieve good results in the case studies performed.

2.3 Emotions in Social Network

Authors [48] automatically create large emotion-labeled dataset by collecting tweets using Twitter streaming API which contain emotion hash-tags. According to Merriam-Webster dictionary hash-tag is a word or phrase preceded by the symbol # that classifies or categorizes the accompanying text (such as a tweet). Their source of the emotion words is psychology paper by [9]. The list of basic emotional words in [3] were expanded by including lexical variants, e.g., 'surprising' and 'surprised' for 'surprise'. By using this approach, they collected 5 million tweets and further applied certain filtering heuristics as follows: retain only tweets with emotion hash-tags at the end, discard tweets having less than five words, remove tweets with URLs or quotations. After which they had a collection of 2,488,982 tweets. Features like N-gram, Adjectives, N-gram position, Part-of-Speech, Sentiment/Emotion lexicons were explored, and their results show that combination of n-gram, sentiment/emotion lexicons, part-of-speech yields higher accuracy with both the machine learning classifiers close to 60%. Also, they validate the effectiveness of larger training dataset by creating sequence of training dataset with increasing size and observe dataset size is directly proportional to accuracy.

Author [129] developed corpus from Twitter posts using emotion hash-tags like [48], [50] and [130] called as Twitter Emotion Corpus (TEC) consisting of 21,000 tweets. Support Vector Machines (SVM) with Sequential Minimal Optimization (SMO) classifier was used with uni-gram and bi-gram features. The automatic classifiers obtained an F-score much higher than the random baseline (SemEval - 2007, 1000 headlines dataset). Similar to [48], in this paper best results are achieved with higher number of training instances. For example, Joy-NotJoy classifier get the best results compared to Sadness-NotSadness. He also performed experiments to show the effectiveness of cross-domain classification by using the TEC corpus for classifying the newspaper headlines domain.

Authors Hasan et al. [50] evaluated the use of hashtags like [48] to automatically label Twitter messages with corresponding emotion tags. They used Circumplex model of human affect for defining the emotional states. In this work Hasan et al. validate and confirm that hash-tags are reliable features for automatic emotion labeling. To prove that hash-tags are reliable sources for automatic emotion labeling they conducted two sets of experiment one with novices and other with psychology experts and validated using Fleiss-Kappa results. In this work they also identified that human labeling of emotion using crowd sourcing is inconsistent and unreliable whereas expert labeling gives 87% accuracy with hash-tag labels. The system 'Emotex' was developed to classify Twitter messages achieved 90% accuracy.

Authors Roberts et al. [130] create emotion corpus from micro-blogging service Twitter. The corpus contains seven emotions annotated across 14 topics including Valentine's Day, World Cup 2010, Stock Market, Christmas etc. The emotions are based on [7] six basic emotions and 'Love'. The topics of each tweet are obtained by considering the tweet to be associated with a probabilistic mixture of topics using Latent Dirichlet Allocation (LDA) topic modeling technique. The system uses a series of binary SVM classifiers to detect each of the seven emotions annotated in the corpus. Each classifier performs independently on a single emotion, resulting in 7 separate binary classifiers implemented using the software available from WEKA. Each classifier uses specific set of features like punctuation, hypernyms, n-grams, and topics. According to the results 'Fear' is the best performing emotion and also suggests that this emotion is highly lexicalized with less variation than other emotions, as it has comparable recall but significantly higher precision.

Authors Bollen et al. [49] analyze the relationship between public mood patterns and social, economic, and other major events in media and popular culture over a time by using sentiment analysis on tweets extracted from micro-blogging platform Twitter. They use Profile of Mood States (POMS), a psychometric questionnaire composed of 793 adjective terms including synonyms and related word constructs. It is proved that POMS serve as a valid alternative to machine learning. They calculate aggregate of the mood vector for all tweets of a day. Their results show that social, political, cultural, and economical events have significant effect on public mood. The effect of economic events on public mood is equivalent to the degree of public response to rapid changes of economic indicators magnified by media.

Authors Purvet et al. [131] used Twitter data labeled with emoticons and hash-tags to train supervised classifiers. They used Support Vector Machines with linear kernel and uni-gram features for classification. Their method had better performance for emotions like happiness, sadness, and anger but not good in case of other emotions like fear, surprise, and disgust. They achieved accuracy in the range of 60%.

Authors Bruyne et al. [65] use ensemble of classifiers for the multi-class multi-label problem. They utilize different aspects of the tweet by extracting tweet features including lexicon features, n-gram features, syntactic, semantic features, and word embeddings. They suggest that classifier chain performs better than the individual binary classifiers.

Authors Himeno et al. [66] use convolutional neural networks to classify the tweet emotion intensity. Their idea is based on the observation that n-grams have vital effects to represent the tweet emotion. They achieve an average correlation co-efficient of 0.620.

Authors Turcu et al. [132] use supervised machine learning models like Naive Bayes, K-Nearest Neighbor and Support Vector Macines, and deep learning neural net tensor flow model, decision tree for affective tweets task and achieve best accuracy of approximately 0.73 for emotion Joy.

Authors Yassine and Hajj [133] use undupervised techniques on facebook data to study the friendship relations and emotions expression in online social networks. Similarly authors Sun et al. [134] use sentiment feedback on social media data to improve item recommendations. They validate different approaches including Support Vector Machines (SVM), SVM-Boosting, Naive Bayes and others along with the proposed ensemble learning-based sentiment classification method. It is observed that SVM method achieves best performance among all the machine learning classifiers for sentiment detection, but the proposed ensemble method further helps to overcome the challenges associated with special features of affective text and achieves an accuracy of 86.7% .

Recently, deep learning-based methods become more popular to classify the social media data. Authors Jianqiang et al. [135] classify tweets into positive or negative class using SVM, linear regression and deep convolution neural network (DCNN). Unigram and bigram features have been used as baseline feature models, and polarity score was calculated for classifying a tweet. Results were generated on Bag of Words(BoW) and GloVe [136] features by running mentioned three techniques. They use accuracy and average of precision, recall, F1-measure as evaluation measures. Using DCNN, highest accuracy was recorded with avg 85.63%

Authors Krebs et al. [137] combine emotion mining techniques and two neural networks (Convolution Neural Network and Recurrent Neural Network) build emotion miner for predicting Facebook posts reaction distribution ratios in order to enhance customer experience analytics. They apply mining techniques on around 26K Facebook reaction posts. Later preprocessing methods such as lowercase conversion, URL removal were applied on the data. This data was fed as input data to emotion miner. Finally, linear regression technique was used on the results (i.e. estimations and emotions were combined into single vector) from the above two parts to predict the post reaction distribution ratios.

Authors Lakomkin et al. [138] predict the tweet emotion intensity detection based on ensemble of two neural network-based models, by processing input at character level and word level with a lexicon-driven system. Character-level model use single multiplicative Long Short Term Memory (LSTM) with 4,096 hidden units while word-level model used Gated Recurrent Unit (GRU) technique for classification. Character-level model was trained with around 80 million Amazon product reviews and word-level model used pre-trained versions of GloVe embeddings [136] trained on Wikipedia and Twitter. Results show that character-level modeling of noisy short texts are effective compared to the Affective Tweet baseline model.

2.4 Emotions in Education

In this section, we review studies that have been done in the area of analyzing student evaluations, including text and quantitative data. Sentiment Analysis in education data has been widely applied to Massive Open Online Courses [6], and e-learning [7]. On the other hand we see minimal contribution towards actual classroom student feedbacks and impact of active learning methodologies on students emotion.

2.4.0.1 Sentiment Analysis on Regular Courses

Authors Kim et al. [25] perform Sentiment Analysis on the ratings and textual responses of student evaluation of teaching. They automatically rate the textual response as one of the three categories 'positive', 'negative', and 'neutral'. In which they have compared the performance of categorical model and dimensional model where 'joy' and 'surprise' are positive class, 'anger', 'fear' and 'sadness' are negative class respectively. In their work they have utilized two emotion lexicons WordNet-Affect and ANEW for the sentiment classification tasks. The following five approaches are modeled for automatic classification of three sentiments 'positive', 'negative', and 'neutral': a) Majority Class Baseline (MCB), b) Keyword Spotting (KWS), c) CLSA - LSA based categorical classification, d) CNMF - NMF based categorical classification, and e) DIM - Dimension based estimation. It is shown in terms of precision, recall and f-measure that NMF based categorical and dimensional models have a better performance than other models.

Typically, in an end-of-course evaluation the students do not benefit to see the actions taken as they move on from the section after that semester. In order to overcome it is required to obtain prompt feedback from students to instructors and necessary actions can be taken during the course. Authors Leong et al. [26] propose the use of short message service (SMS) for student evaluation and explore the application of text mining in particular Sentiment Analysis ('positive' and 'negative') on SMS texts. They show the positive and negative aspects of lecture in terms of the conceptual words extracted and text link analysis visualization.

Similar to [26] authors Altrabsheh et al. [27] explore approaches for real time feedbacks. This work discusses how feedback is collected via social media such as Twitter and apply Sentiment Analysis to improve teaching called as Sentiment Analysis for Education (SA-E). This system collects data from Twitter where the students provide their feedback. The text data after pre-processing and extracting features including: term presence and frequency, N-gram position, part-of-speech, syntax, and negation. Later the text is analysed via Naive Bayes and/or Support Vector Machine which categorizes the whole post as either 'positive' or 'negative'.

Authors Jagtap et al. [139] perform Sentiment Analysis on student feedback data classifying into 'positive' and 'negative' categories. They combine Hidden Markov Model (HMM) and Support Vector Machine (SVM) and use a hybrid approach for sentiment classification. Though they have concluded that applying advance feature selection method combined with hybrid approach work well for complex data, their works did not show the results of classification model for validation.

Authors Rajput et al. [140] apply text analytics methods on student's feedback data and obtain insights about teacher's performance with the help of tag clouds, and sentiment score. In this work the authors use sentiment dictionary Multi-Perspective Question Answering (MPQA) [141] to find words with positive and negative polarity. By combing the word frequency and word attitude the overall sentiment score for each feedback is calculated. Finally, they have compared the sentiment score with Likert scale-based teacher evaluation and conclude that Sentiment score with word cloud provide better insights than Likert-scale results.

2.4.0.2 Sentiment Analysis on MOOC and E-Learning Courses

Massive Open Online Course (MOOC) is wide-spread since it is first introduced in 2006 as part of distant education. It is basically free participation to users from any location and does not bind to any individual university or organization. It is observed that MOOC suffer a high drop-out rate close to 90% [142], [143]. There are lot of factors that influence drop-out, users perspective or emotions is one of the major concerns. Using survival modelling technique and lexicon based approach, [144] it is shown that student sentiment towards MOOC has impact on the attrition rate over time. On the other hand student performance and learning outcomes are important given the high attrition rates, improving factors that affect performance can help improve drop-out rates. Student generated text data is mined to quantify their impact of performance and learning outcomes [145]. Authors [145] use a lexicon based approach and study the correlation of student sentiment with quiz and assignment grades. E-

Learning is considered as internet based learning, use of technological and digital tools to deliver educational content [146], [147]. To build an effective e-learning system it is necessary that the instructor gains some insight and knowledge about users opinion and or sentiment towards the technology used and materials covered. Authors Kechaou et al. [148], use feature selection and hybrid classifier for sentiment classification of e-learning blogs. They suggest that Information Gain (a criteria for measuring goodness) outperforms the other two features Mutual Information and CHI statistics. Similarly [149], [150], [151] proposed lexicon-based approaches and machine learning classifiers including support vector machine, and probabilistic approach based on Latent Dirichlet Allocation (LDA) to identify the student sentiment as either positive, negative or neutral. Authors Binali et al. [152], identify emotional reaction of students towards the e-learning courses. They classify the student response into one of the following emotional state 'confusion', 'happy', 'interest', and 'sad'.

2.4.0.3 Emotion Mining on Student Feedback

Authors Kim et al. [25] compare the performance of categorical model and dimensional model by grouping the fine-grained emotion into more generic classes. They use lexicon-based approach and classification models including Majority Class Baseline (MCB), Keyword Spotting (KWS), and Dimension based estimation, CNMF - NMF based categorical classification. It is observed that NMF based categorical model and dimensional model have better performance.

2.4.0.4 Clasification - Traditional Machine Learning Models

Authors Altrabsheh et al. [153], collect real time student feedback and label the data into three sentiment class 'positive', 'neutral', and 'negative' with help of three experts. The learning performance was investigated with the following machine learning techniques: Naive Bayes, Complement Naive Bayes, Maximum Entropy, and Support Vector Machine. They achieve good results with Support Vector Machine and

Complement Naive Bayes. In a similar way authors Leong et al. [154] use prompt feedback and propose the use of short message service (SMS) for student evaluation and explore the application of text mining in particular Sentiment Analysis ('positive' and 'negative')on SMS texts. They show the positive and negative aspects of lecture in terms of the conceptual words extracted and text link analysis visualization.

Authors Dhanalakshmi et al. [155], classify student's feedback into 'positive' or 'negative' and suggest that Naive Bayes performs better with good recall. Authors Jagtap & Dhotre [139] classify student feedback data into 'positive' and 'negative' categories by using of hybrid approach combining Hidden Markov Model (HMM) and Support Vector Machine (SVM). Though they have concluded that applying advance feature selection method combined with hybrid approach work well for complex data, their works did not show the results of classification model for validation.

Authors Rajput et al. [140] apply text analytics methods on students feedback data and obtain insights about teachers performance with the help of tag clouds, and sentiment score. In this work the authors use sentiment dictionary Multi-Perspective Question Answering (MPQA) Stoyanov et al. [156] to find words with positive and negative polarity. By combing the word frequency and word attitude the overall sentiment score for each feedback is calculated. Finally they have compared the sentiment score with Likert scale based teacher evaluation and conclude that Sentiment score with word cloud provide better insights than Likert scale results.

2.4.0.5 Clasification - Neural Networks

Neural Networks is widely used in several classification tasks and proven to achieve best results. But it is still in the infancy stage with Educational Data. Most of the works in literature focus on predicting student performance using Artificial Neural Networks. For instance, Guo et al. [157] use multiple level representations with unsupervised learning and fine tune neural network layers through back propagation. They use High school data with different kinds of information including background and demographic data, past study data, school assessment data, study data, and personal data. Compared to the traditional methods like Support Vector Machines and Naive Bayes, their model achieve better performance. Authors Musso et al. [158], also use student background information along with cognitive and non-cognitive measures to predict student academic performance using Artificial Neural Networks achieve greater accuracy compared to discriminant analyses method.

While the above methods use non-text data for classification, the following researchers use text data. Online discussion forum is a popular tool for student communication and collaboration in web-based courses. Authors Wei et al. [159] use Stanford MOOC posts dataset [160] to identify ‘confusion’, or ‘urgency’ and sentiment of the posts. They propose a transfer learning framework based on convolutional neural network and long short-term memory model. Student Evaluation of Teaching Effectiveness (SETE) serves as an important aspect in validating the teaching models, resourses and effectiveness of teaching and learning outcomes. Authors Galbraith et al. [161] use Neural Networks to measure student learning outcomes from SETE 's.

There is not much work in applying Neural networks for sentiment classification from student evaluation of teaching. In this work we use sequential Neural network model with 1D convolution and word embedding for automatic classification of emotions from student evaluations.

2.4.0.6 Actionable Pattern Discovery in Student Evaluation Data

There is a wide range of research in the field of Education and Data Mining with different methods and applications. The applications are categorized as (1) applications that focus on the objective of the task and (2) applications that focus on the end user. In this section we have a brief discussion about such literature in the field of Education.

Authors Bakhshinategh et al. [162] classify the Education Data Mining tasks into different subcategories based on their applications. One of them is representing the cognitive aspects of students. Some of the works in this area include predicting student performance [163], identifying their motivational level [164], use of clustering and classification methods to predict undesirable student performance [165].

Sentiment Analysis has gained popularity in the recent years in the field of Education. Several researchers focus on the task of identifying sentiments (positive, negative, or neutral) from students comments. The main objective of their work is to evaluate the effect of teaching by using student ratings and feedback.

All of the above applications focus only on identifying if certains tasks work well or not in the Education setting. In this work we propose a novel approach of using student feedback data and labeling it with fine grained emotion to identify patterns and trends, and make actionable recommendations for improvement. This helps Teachers and the Management to assess important factors that need attention or change for enhancement of teaching and learning.

2.5 Actionable pattern Mining

Actionability is a property of the discovered knowledge. Patterns are considered Actionable if the user can act upon them, and if this action can benefit the user, or help them to accomplish their goals. Author [166] explore the paradigm shift of knowledge discovery from data to actionable knowledge discovery and delivery. He observes macro-level (methodological and fundamental issues) and micro-level (technical and engineering issues) perspectives to narrow down the gaps between delivered knowledge and desired knowledge and states that Actionable Knowledge Discovery and Delivery (AKD) framework help narrow the gaps in Knowledge discovery process. The author suggests that use of domain knowledge in the data mining process and engaging organizational and social intelligence in the KDD modeling process help furthering the paradigm shift.

Authors [167] extract actionable knowledge from data collected in schools that could be valuable to students, teachers, principals, district, state and national administrators. According to authors [168] patterns discovered from data are represented in the form of *'if ..., then...'* rules called decision rules. These patterns provide information about past events and utilized for prospective decisions. For instance, in medical diagnosis these rules can help identify the relationship between symptoms and sickness and also diagnose new patients based on these past records. Another prospective usefulness of decision rules is getting the desired effect on dependent variables by building strategy of intervention on the independent variables. In the medical example, this can be explained as modifying symptoms to get out from the sickness.

2.5.1 Action Rules

Action Rules Mining is a method to discover Actionable Patterns from large datasets. Action Rules are rules that describe a possible transition of data from one state to another, or in other words, Action Rules reclassify data from one category to another [169]. In Data Mining literature, we see two pre-dominant frameworks for Action Rule generation: Rule based (loosely coupled) and Object based (tightly coupled) methods.

Author dardzinska [67], summarize the frameworks for generating Action Rules from [59] as follows: loosely coupled and tightly coupled. The loosely coupled framework is often called rule-based. It is based on pairing certain classification rules which have to be discovered first by using for instance algorithms such as LERS [170] or ERID [35], [171]. The tightly coupled framework is often called object-based and it assumes that Action Rules are discovered directly from a database [172], [60], and [36]. Classical methods for discovering them follow algorithms either based on frequent sets (called action sets) and association rules mining [173] or they use algorithms such as LERS or ERID with atomic action sets used as their starting step. Action Rules are one way to mine Actionable knowledge from large dataset.

2.5.1.1 Information System and Decision System

Information system Table. 2.1 is percieved as a system $\mathbb{Z} = (\mathbb{X}, \mathbb{M}, \mathbb{V})$, where $\mathbb{X}$ is set of objects $\{x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8\}$ in the system; $\mathbb{M}$ is non-empty finite set of attributes $\{A, B, C, E, F, G, D\}$; $\mathbb{V}$ is the domain of attributes in $\mathbb{M}$, for instance the domain of attribute B in the system $\mathbb{Z}$ is $\{B_1, B_2, B_3\}$.

Table 2.1: Information System Z.

X	A	B	C	E	F	G	D
x_1	A_1	B_1	C_1	E_1	F_2	G_1	D_1
x_2	A_2	B_1	C_2	E_2	F_2	G_2	D_3
x_3	A_3	B_1	C_1	E_2	F_2	G_3	D_2
x_4	A_1	B_1	C_2	E_2	F_2	G_1	D_2
x_5	A_1	B_2	C_1	E_3	F_2	G_1	D_2
x_6	A_2	B_1	C_1	E_2	F_3	G_1	D_2
x_7	A_2	B_3	C_2	E_2	F_2	G_2	D_2
x_8	A_2	B_1	C_1	E_3	F_2	G_3	D_2

The information system in Table. 2.1 is denoted as Decision system if the attributes $\mathbb{M}$ are classified into flexible M_{fl} , stable M_{st} and decision d, $\mathbb{M} = (M_{st}, M_{fl}, \{d\})$. From Table. 2.1 $M_{st} = \{A, B, C\}$, $M_{fl} = \{E, F, G\}$, and $d = D$.

2.5.1.2 Action Term

The expression (y, $y_1 \rightarrow y_2$) is an atomic action term, where y is an attribute and y_1, $y_2 \in V_y$. If $y_1 = y_2$), then y is stable on y_1. In this case action term is denoted as (y, y_1) for simplicity.

- If t is an atomic action term, then t is an action term.

- If t_1 , t_2 are action terms, then $t_1 * t_2$ is an action term.
- If t is an action term containing $(y, y_1 \rightarrow y_2)$, $(z, z_1 \rightarrow z_2)$ as its sub-terms, then $y \neq z$.
- Domain of action term is denoted by Dom(t), which includes all attributes listed in t.

2.5.1.3 Action Rule

The expression $r = [t_1 \rightarrow t_2]$ is an Action Rule where, t_1 is an action term and t_2 is an atomic action term. The following is an example Action Rule from Table.2.1. $[B_1 \wedge C_1 \wedge (F, F_3 \rightarrow F_1) \wedge (G, \rightarrow G_1) \rightarrow (D, D_2 \rightarrow D_1)$.

2.5.1.4 Support and Confidence

Support and confidence of rule r is given as below:

- $sup(r) = min\{card(Y_1 \cap Z_1), card(Y_2 \cap Z_2)\}$.
- $conf(r) = \frac{card(Y_1 \cap Z_1)}{card(Y_1)} \cdot \frac{card(Y_2 \cap Z_2)}{card(Y_2)}$.
- $card(Y_1) \neq 0$, $card(Y_2) \neq 0$, $card(Y_1 \cap Z_1) \neq 0$, $card(Y_2 \cap Z_2) \neq 0$.
- $conf(r) = 0$ otherwise.

2.5.2 Learning from Rough Sets (LERS)

LERS [170] is classic bottom-up stragey that constructs rules with a conditional part of the length $k + 1$ after all rules with a conditional part of length k have been constructed. This method finds the certain and possible rules describing the decision attribute in terms of other attributes in the system. Let us assume that Table.2.1 as Decision system with the foolowing attributes $\mathbb{M} = (M_{st}, M_{fl}, \{d\})$, where $M_{st} = \{A, B, C\}$, $M_{fl} = \{E, F, G\}$, and $d = D$.

This is the list of certain and possible rules that LERS strategy finds from Table.2.1.

- Certain Rules

- $E_1 \rightarrow D_1$
- $G_3 \rightarrow D_2$
- $F_3 \rightarrow D_2$
- $E_3 \rightarrow D_2$
- $B_2 \rightarrow D_2$
- $B_3 \rightarrow D_2$
- $A_3 \rightarrow D_2$
- $A_2 \wedge G_1 \rightarrow D_2$
- $A_1 \wedge E_2 \rightarrow D_2$
- $A_2 \wedge C_1 \rightarrow D_2$
- $A_1 \wedge C_2 \rightarrow D_2$
- $E_2 \wedge C_1 \rightarrow D_2$
- $G_2 \wedge B_1 \rightarrow D_3$
- $G_1 \wedge E_2 \rightarrow D_2$
- $G_1 \wedge C_2 \rightarrow D_2$
- $A_1 \wedge C_1 \wedge B_1 \rightarrow D_1$
- $A_2 \wedge B_1 \wedge C_2 \rightarrow D_3$
- $A_2 \wedge E_2 \wedge B_1 \wedge C_2 \rightarrow D_3$
- $A_2 \wedge F_2 \wedge E_2 \wedge B_1 \rightarrow D_3$
- $A_1 \wedge G_1 \wedge C_1 \wedge B_1 \rightarrow D_1$
- $A_2 \wedge F_2 \wedge B_1 \wedge C_2 \rightarrow D_3$
- $A_1 \wedge F_2 \wedge C_1 \wedge B_1 \rightarrow D_1$
- $G_1 \wedge F_2 \wedge C_1 \wedge B_1 \rightarrow D_1$

- Possible Rules
  - $A_2 \wedge G_2 \wedge F_2 \wedge E_2 \wedge C_2 \rightarrow D_1$
  - $A_2 \wedge G_2 \wedge F_2 \wedge E_2 \wedge C_2 \rightarrow D_2$
  - $A_2 \wedge G_2 \wedge F_2 \wedge E_2 \wedge C_2 \rightarrow D_1$

2.5.3 Action Rules Based on Agglomerative Strategy (ARoGs)

ARoGs [30] uses LERS [170] to extract Action Rules without the need to verify the validity of the certain rules. By using LERS as pre-processing step the overall complexity of ARoGs method is reduced when compared to DEAR [37], [174] method.

Using the Table. 2.1, below is the sample Action Rule that ARoGs algorithm generates considering the change of decision value from $D_2 \rightarrow D_1$.

ARoGs method uses the certain rules extracted by LERS strategy and generates Action Rule schema. Then for each Action Rule schema the Action Rules are constructed. For instance let us take the classification rule "(2.1)".

$$G_1 \wedge F_2 \wedge C_1 \wedge B_1 \rightarrow D_1 \tag{2.1}$$

The Action Rule schema associated with the above Equation. 2.1 for the reclassification task $D_2 \rightarrow D_1$ is given as "(2.2)" and the coressponding Action Rule "(2.3)".

$$[\mathrm{B_1} \wedge \mathrm{C_1} \wedge (\mathrm{F}, \rightarrow \mathrm{F_1}) \wedge (\mathrm{G}, \rightarrow \mathrm{G_1})] \rightarrow (\mathrm{D}, \mathrm{D_2} \rightarrow \mathrm{D_1}). \tag{2.2}$$

$$[\mathrm{B_1} \wedge \mathrm{C_1} \wedge (\mathrm{F}, \rightarrow \mathrm{F_1}) \wedge (\mathrm{G}, \mathrm{G_3} \rightarrow \mathrm{G_1})] \rightarrow (\mathrm{D}, \mathrm{D_2} \rightarrow \mathrm{D_1}). \tag{2.3}$$

2.5.4 Apriori Based Association Action Rule Mining(AAR)

The Association Action Rules described by Ras et al. [29] generates association type Action Rules using frequent action sets in Apriori like fashion. The frequent action set generation is divided in two steps: merging step and pruning step.

- **Merging step**: The algorithm merges the previous two frequent action sets into a new action set.

- **Pruning step**: The algorithm discards the newly formed action set if it does not contain the decision action (e.g. the user desired value of decision attribute).

For our example, using the data from Table. 2.1, the primary action sets generated by AAR are shown in Table. 2.2. The frequent action sets generated by AAR are shown in Table.2.3.

Table 2.2: Primary Action Sets.

Attribute	Primary Action Set
B	(B, B_1), (B, B_2), (B, B_3)
C	(C, C_1), (C, C_2)
E	(E, E_1), (E, E_2), (E, E_3), (E, $E_1 \rightarrow E_2$), (E, $E_1 \rightarrow E_3$), (E, $E_2 \rightarrow E_1$), (E, $E_2 \rightarrow E_3$), (E, $E_3 \rightarrow E_1$), (E, $E_3 \rightarrow E_2$)
F	(F, F_2), (F, F_3), (F, $F_2 \rightarrow F_1$), (F, $F_2 \rightarrow F_3$), (F, $F_3 \rightarrow F_1$), (F, $F_3 \rightarrow F_2$)
G	(G, G_1), (G, G_2), (G, G_3), (G, $G_1 \rightarrow G_2$), (G, $G_1 \rightarrow G_3$), (G, $G_2 \rightarrow G_1$), (G, $G_2 \rightarrow G_3$), (G, $G_3 \rightarrow G_1$), (G, $G_3 \rightarrow G_2$)
D	(D, D_1), (D, D_2), (D, D_3), (D, $D_1 \rightarrow D_2$), (D, $D_1 \rightarrow D_3$), (D, $D_2 \rightarrow D_1$), (D, $D_2 \rightarrow D_3$), (D, $D_3 \rightarrow D_1$), (D, $D_3 \rightarrow D_2$)

Table 2.3: Frequent Action Sets.

Iteration	Frequent Action Set
Iteration 1	(A, A_1) $\wedge$ (D, $D_2 \rightarrow D_1$) (A, A_2) $\wedge$ (D, $D_2 \rightarrow D_1$) (A, A_3) $\wedge$ (D, $D_2 \rightarrow D_1$) (B, B_1) $\wedge$ (D, $D_2 \rightarrow D_1$) (B, B_2) $\wedge$ (D, $D_2 \rightarrow D_1$) (B, B_3) $\wedge$ (D, $D_2 \rightarrow D_1$) ....
Iteration 2	(A, A_1) $\wedge$ (B, B_1) $\wedge$ (D, $D_2 \rightarrow D_1$) (A, A_1) $\wedge$ (B, B_2) $\wedge$ (D, $D_2 \rightarrow D_1$) (A, A_1) $\wedge$ (B, B_3) $\wedge$ (D, $D_2 \rightarrow D_1$) ....
Iteration n	....

In our example, the action set is discarded if $(D, 2 \rightarrow 1)$ is not present in it. From each frequent action set, the association Action Rules are formed. Therefore, the algorithm generates frequent action sets and forms the association Action Rules from these action sets. For our example, using the data from the Information system in Table. 2.1, the algorithm generates Association Action Rules, an example is shown below:

$$(B, B_1 \rightarrow B_1) \wedge (C, C_1 \rightarrow C_1) \wedge (E, E_3 \rightarrow E_1) \rightarrow (D, D_2 \rightarrow D_1)$$

2.5.5 Rule Based Action Rule Mining

In Rule based method, extraction of Action Rules or actionable knowledge is dependent on the pre-processing step of classification rule discovery. These methods use pre-existing classification rules or generate rules using algorithms like Learning

Based on Rough Sets (LERS) [170] and Extracting Rules from Incomplete Decision (ERID) [175] Systems. Rule based methods are further sub-divided into methods generating Action Rules from certain pairs of classification rules like Discovering Extended Action Rules (DEAR) [37], [174], and methods that generate Action Rules from single classification rule Action Rules Based on Agglomerative Strategy (ARoGs) [30].

2.5.6 Object Based Action Rule Mining

Action Rule Extraction from Decision Table (ARED) [36], Association Action Rule [29] method extracts Action Rule directly from the database without the use of classification rules.

2.5.7 Actionable Pattern Mining Applications

Actionability is a property of the discovered knowledge. If user's can act upon a pattern and benefit from the action, these patterns are considered as actionable patterns. This section reviews literature for actionable pattern mining.

Authors Ras et al. [176] model a Net Promoter Score (NPS) Recommender system for driving business revenue mainly based on Action Rules and Meta Actions. Net Promoter Score (NPS) is a standard metric for measuring customer satisfaction. This system utilized around 400,000 records of the customer satisfaction telephone surveys containing details related to customer details, survey details and benchmark questions. Action Rules, knowledge in actionable format is collected from customers using a business and also from customers using semantically similar business. The concept of decision reducts (minimal set of attributes that keep the characteristics of the full dataset [176] is used to choose critical benchmarks. The triggers (Meta Actions) for Action Rules are extracted based on aspect-based sentiment analysis [177] and text summarization of the customer text comments in the survey. Feature-opinion pairs are identified with Stanford Parser. They also performed feature clustering based on pre-defined list of seed words. Meta Actions are generated by dividing feature class into several subclasses.

Authors [178] also explain application of decision reducts theory to solve business problem. Similar to authors [176], this paper focus on business recommendations to improve Net Promoter Score of companies. They detail the application area - Customer Loyalty Improvement, machine learning techniques used to develop the knowledge based system and visualization techniques for the interactive recommender system.

Action Rules are used to discover patterns in the form of rules called decision rules *'if ..., then...'* [168]. These patterns provide details about events in the past and suggestions for making prospective decisions. In medical field, for disease diagnosis, decision rules help identify the correlation between symptoms and sickness with the past data and help diagnose new patients. Another prospective usefulness of decision rules is getting the desired effect on dependent variables by building strategy of intervention on the independent variables. In the medical example, this can be explained as modifying symptoms or treating symptoms to cure sickness.

Authors Tzacheva et al. [179], discover low cost actionable patterns and recommendations in distributed environment. They use the algorithm described by [169] and develop the system using Apache Spark framework and Hadoop Distributed File System (HDFS) for scalability and efficiency. They evaluate the approach using car evaluation dataset, mammographic dataset and achieve best results.

In relation to Emotion mining, Actionable patterns may suggest a way to alter the user's emotion from a negative, or neutral to a more positive Emotion, or a desirable state / attitude. For example, for customer care services, recommendation systems for online shopping, or smart phones that are able to recognize human emotions, Emotion altering Actionable Patterns include: suggesting calming music, playing mood enhancing movie, changing the background colors to suiting ones, or calling caring friends (for smart phones). In [22] the primary intent of the Action Rules generated is to provide viable suggestions on how to make a twitter user feel more positive. For Twitter social network data, Actionable Recommendations may include - how to increase user's friends count, how to increase the user's follower's count, and how to change the overall sentiment from negative to positive, or from neutral to positive.

2.5.8 Algorithms Computational Efficiency

The above mentioned algorithms work faster for datasets of considerable size, but the Association Action Rule method is computationally extensive and time consuming because of the iterative nature. Scalability and processing time is one of the main attributes needed for Association rule mining with data increasing in terms of both dimensions and size.

Agrawal and Shafer [180] proposed three parallel distribution algorithms for association rule mining namely: Count distribution, Data set distribution and Candidate distribution algorithms. But each of these algorithms have its own disadvantages as follows, Count distribution algorithm does not efficiently utilize the aggregate system memory; Data distribution algorithm suffers communication overhead; Candidate distribution redistributes the database while scanning the local partition repeatedly and is worse compared to Count distribution algorithm [181]. Shintani and Kitsuregawa [182] used a hybrid approach of Hash Partitioned Apriori - Extremely Large itemset duplication (HPA-ELD) combined with the non-partitioned apriori method to ensure least amount of communication overhead. Another hybrid distribution method that combines Count distribution and intelligent data distribution is proposed by Han et al. [183]. This method reduces the database communication cost.

In recent years there are studies that use Mapreduce Hadoop framework [39] and Spark [40] for distributed Action Rules Mining. These work use random data partition [184] and information granules for partitioning [185] to extract Action Rules and

Association Action Rules.

In this work we propose a novel approach of hydrid Association Action Rule generation, combining the rule based and object based approach of Action Rule mining to reduce the overhead of the iterative procedure.

2.6 Rough Sets

Attribute selection is one of the key problems in Data Mining and Knowledge Discovery. This section outlines the existing approaches presented in the literature for Attribute selection. Authors have porposed several algorithms for reduct computation. Most common forms of attribute reduction is by use of Boolean Matrix [97], Decision Table and reduction rules [186], Discernibility Matrix [187] , Attribute weighting [188] and Multicriterion based approaches [189] [190].

All of the above methods are suitable for standard datasets. However, these traditional methods would take long processing time for BigData.

Author Yanhong [97], derive boolean matrix directly from the information system and generate set of reducts. The core idea of this algorithm is to use appearance regularity of the elements of reductions and supersets of reductions to construct the set of all reducts. Similarly in [186], initial information system or table is transformed into a special decision table in the form of matrix. Then by using set of simplification rules on the derived table they construct the set of reducts using dynamic programming. It is applied to medium small sized data tables from UCI Machine Learning repository **??**.

Another traditional approach [191] is using attribute consistency check for finding the reduct and core of consistent datasets. This algorithm removes one condition attribute at a time to check for consistency of the remaining table. A table is said to be consistent if for two or more rows or cases for the same values of condition attributes we have the same decision otherwise it is inconsistent. Thus the attributes that are required for the consistent table are marked as reducts of the dataset. This approach is extensive and it is only applicable for smaller datasets.

Authors Wang et. al., [187] and Skowron [101] use discernibility matrix based algorithm for finding attribute reducts. Similarly [100] propose a discernibility based funtion to find reducts. Authors Al-Radaideh et al., [188], also use the discernibility matrix modulo as input for the heuristic reduct computation approach by attributes weighting. They use different attribute weights such as global weight, local weight, and attribute value cardinalities for the purpose of reduct generating algorithm.

Authors Korzen and Jaroszewicz [192] find reducts without explicitly building the discernibility matrix, by use of conditional Gini index. The discernibility based algorithms are usually expensive escpecially for large datasets. There have been many heuristic and greeedy algorithms proposed for attribute reduction.

Rough set theory was first introduced by Zdzislaw Pawlak [193]. This approach of Rough Sets constitutes a base for Knowledge Discovery in Databases, Machine Learning, Decision Support Systems, Pattern Recognition. Some of the problems approached using rough set theory are feature selection, eliminate redundant data, identifiy data dependencies, pattern extraction like association rules, discover data similarities or differences, and approximate data classification. Rough set based methods have been applied in the area of business, web and text mining [194], image processing [195], medicine, bioinformatics [196], economics [197].

2.6.1 Basic concepts

In this section we describe basic terms including information system, decision table, and discernibility matrix associated with Rough sets.

2.6.1.1 Discernibility Matrix

Let us consider the Information System Z in section. 2.5.1.1. $\mathbb{DM}(\mathbb{IS})$ - discernibility matrix of the information system $\mathbb{I}$ is represented by a $n \times n$ matrix c_{ij}, given by Eq. 2.4.

$$c_{ij} = \{a \in A : a(x_i) \neq a(x_j)\}, for i, j = 1, 2, ..., n. \quad (2.4)$$

Similarly the discernibility matrix $\mathbb{DM}(\mathbb{DT})$, for a decision table (section 2.5.1.1) is denoted by a $n \times n$ matrix c_{ij}, given by Eq. 2.5.

$$c_{ij} = \begin{cases} \phi & f_d\ (x_i) = f_d\ (x_j); \\ \{a \in A : a(x_i) \neq a(x_j)\} & f_d\ (x_i) \neq f_d\ (x_j) \end{cases} \quad (2.5)$$

The discernibility matrix $\mathbb{DM}(\mathbb{DT})$, for a decision table (section **??**) is given in Table. 2.4.

Table 2.4: Discernibility Matrix Table.

X	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8
x_1								
x_2	ACEG							
x_3	AEG	ACG						
x_4	CE	ACG	ϕ					
x_5	BE	ABCEG	ϕ	ϕ				
x_6	AEF	CFG	ϕ	ϕ	ϕ			
x_7	ABCEG	B	ϕ	ϕ	ϕ	ϕ		
x_8	AEG	CEG	ϕ	ϕ	ϕ	ϕ	ϕ	

2.6.1.2 Indiscernibility Relation - Equivalence Relation

Let us consider the Information System $\mathbb{I} = \langle \mathbb{U}, \mathbb{A}, \mathbb{V} \rangle$. For every set of attributes $B \subseteq A$, an equivalence relation is denoted by $IND_A(B)$ and called the B-indiscernibility relation, which is given in Eq. 2.6.

$$IND_A(B) = \{(u, u') \in U^2 : a \in B, a(u) = a(u')\} \quad (2.6)$$

CHAPTER 3: DATASET

According to the International Data Corporation (IDC) [2], 48% of enterprise datasphere comprises of data from the Manufacturing, Healthcare, Financial services, Media and Entertainment industries. Out of the four industries, it is notable from Fig. 3.1 that Manufacturing is responsible for largest share of data. Fig. 3.2 shows the activity taking place on various platforms such as Twitter, Facebook, Google in each 60 second span in the year 2018.

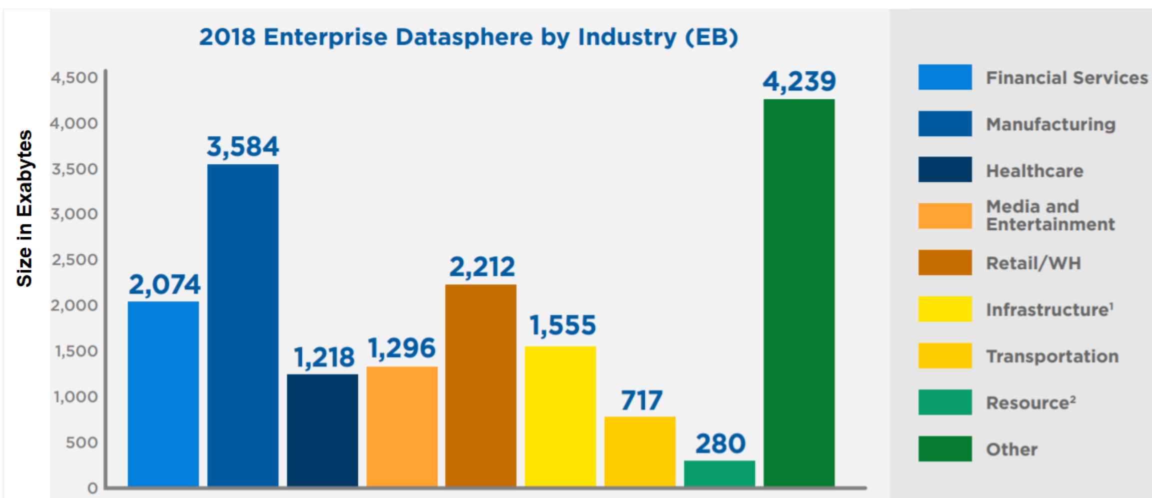

Figure 3.1: 2018 Enterprise Datasphere by Industry showing data in Exabytes [2].

Given the amount of data in the manufacturing industry and social media, in this study we use datasets from Business and Social Media - Twitter and Education - Student Evaluations data.

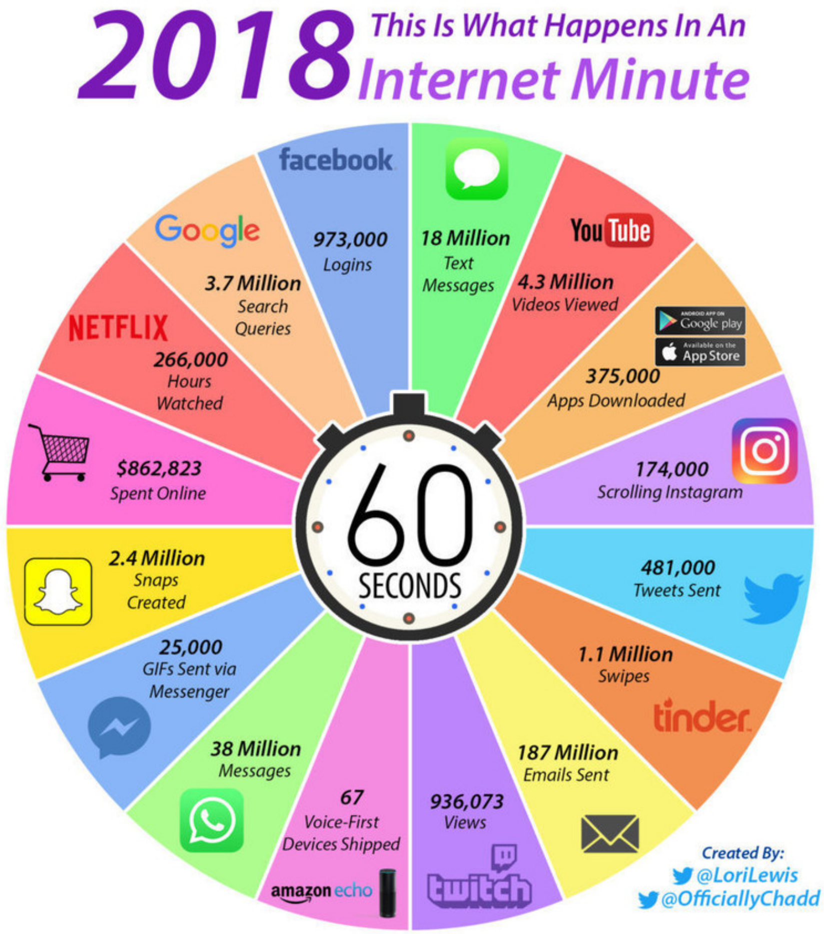

Figure 3.2: 2018 Internet by the Minute.

3.1 Social Network Data - Twitter

In data collection step we used Twitter streaming API [198] to collect the data with the following attributes TweetID, ReTweetCount, TweetFavouriteCount, TweetText (sample tweet is shown in Table .3.1), TweetLanguage, Latitude, Longitude, Tweet-Source, UserID, UserFollowersCount, UserFavoritesCount, UserFriendsCount, User-Language, UserLocation, UserTimeZone, IsFavorited, IsPossiblysensitive, IsRetweeted, RetweetedStatus, UserStatus, MediaEntities. We collected around 520,000 tweets as raw data.

Table 3.1: Sample Tweet Data.

S.No	Tweet
1	Also feeling the love from South America , Need to come see you guys soon
2	#HappyJiminDay Happy birthday Jimin, I love You
3	If it looks like slavery, sounds like slavery, and the logic of slavery is used verbatim to justify it, then it
4	A dream you dream alone is only a dream. A dream you dream together is reality.

3.2 Education Data - Student Evaluation

3.2.1 Qualitative Student Evaluation Data - Web Based Course Evaluation System

The data is collected from the Web-Based course evaluation system by UNC Charlotte. This system is administered by a third-party Campus Labs. In assistance with UNC Charlotte Center for Teaching and Learning, Campus Labs collect the student feedback for course evaluations. The student feedbacks for an instructor is collected for the terms of 2013 to 2018 including Fall, Spring and Summer sections of various courses handled by the instructor. We collect the html files from Campus Labs website for each of the semester. Next, we process the data as described in the Data Extraction subsection below. This data includes both quantitative and qualitative results. For this study we used qualitative feedback mainly focusing on Sentiment Analysis. Sample qualitative data shown in Table 3.2. The Table 3.3. shows the list of semesters for which the data is collected.

Table 3.2: Sample - Student Feedback Qualitative Data.

S.No	Student Comments
1	Easily available to communicate with if needed
2	The course has a lot of valuable information
3	Get rid of the group project
4	There was no enthusiasm in the class. The instructor should make the class more lively and interactive
5	Best professor

Table 3.3: Student Feedback - List of Semesters.

Year	Semester
2013	Spring, Summer, Fall
2014	Spring, First Summer, Second Summer, Fall
2015	Spring, First Summer, Second Summer, Fall
2016	Spring, Spring Midterm, First Summer, Second Summer, Fall
2017	Spring, First Summer, Second Summer, Fall
2018	Spring, First Summer, Second Summer

After the data collection from Campus Labs, jsoup [199] a Java library is used to process the html files and extract the comments. The following fields are extracted from the html file: Year, Term, Course, Questions, Comments. The data extracted consists 959 records with the five attributes as mentioned in Table 3.4.

Table 3.4: Sample - Student Feedback Dataset.

Year	Term	Course	Question	Comments
2014	Fall 2014	Operating Systems and Networking	Please list outstanding strengths of the course and/or instructor	Easily available to communicate with if needed
2014	Fall 2014	Operating Systems and Networking	Please list outstanding strengths of the course and/or instructor	The course has a lot of valuable information
2017	Fall 2017	Cloud Comp for Data Analysis	Please suggest areas for improvement of the course and/or instruction method	There was no enthusiasm in the class. The instructor should make the class more lively and interactive

3.2.2 Student Evaluation Data - Web Based Survey

Web-based student survey data is collected from a public research university in the United States. The survey was designed to provide insight on how students feel about the courses that include Active Learning pedagogies and other factors that help in their learning process. The data collected is part of the courses that implemented and followed the same teaching methodology and style. This dataset has close to 50 attributes. The details of data collection process is described in “Fig. 3.3”.

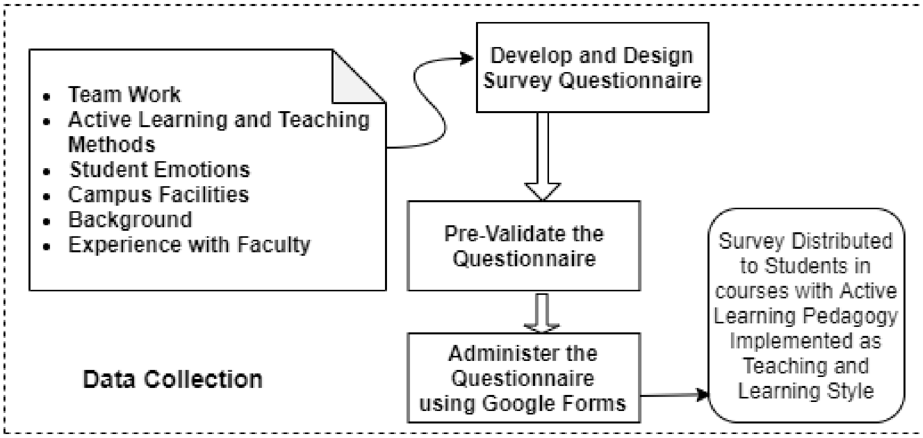

Figure 3.3: Data Collection Process.

Table 3.5: Sample Survey Questions.

S.No	Sample Survey Questions
1	Did you gain any Benefits from Group Assignments?
2	Course group helped me get acquainted with students from different background
3	The class discussions are with the subject matter

Table 3.6: Dataset Properties: Student Survey Data

Property	**Student Survey Data**
Attributes	59 attributes including - Team-Sense of Belonging - Team Member Responsibility - Team Work Helped Diversity - Group Assignment Benefits - Video Case Assignments - Helpfulness - Active Learning Method - Rating - Flipped Class Helped Better Learning - Rating - Peer Teaching Helped Better Learning - Rating - Student Emotion

The survey collected basic demographic information including gender, ethinicity, school year. “Fig. 3.4”, shows the gender distribution in the collected data including

'Male', 'Female', 'Other', and 'Prefer Not to Answer'.

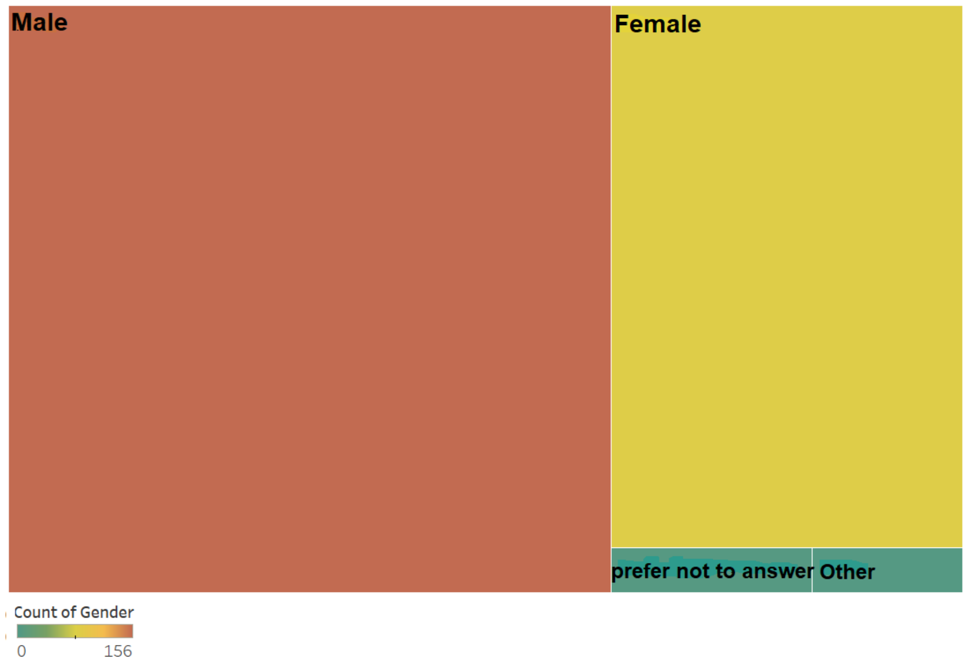

Figure 3.4: Gender Distribution in the Dataset.

The student population "Fig. 3.5", include students from different ethinicity, with majority White, and Asian.

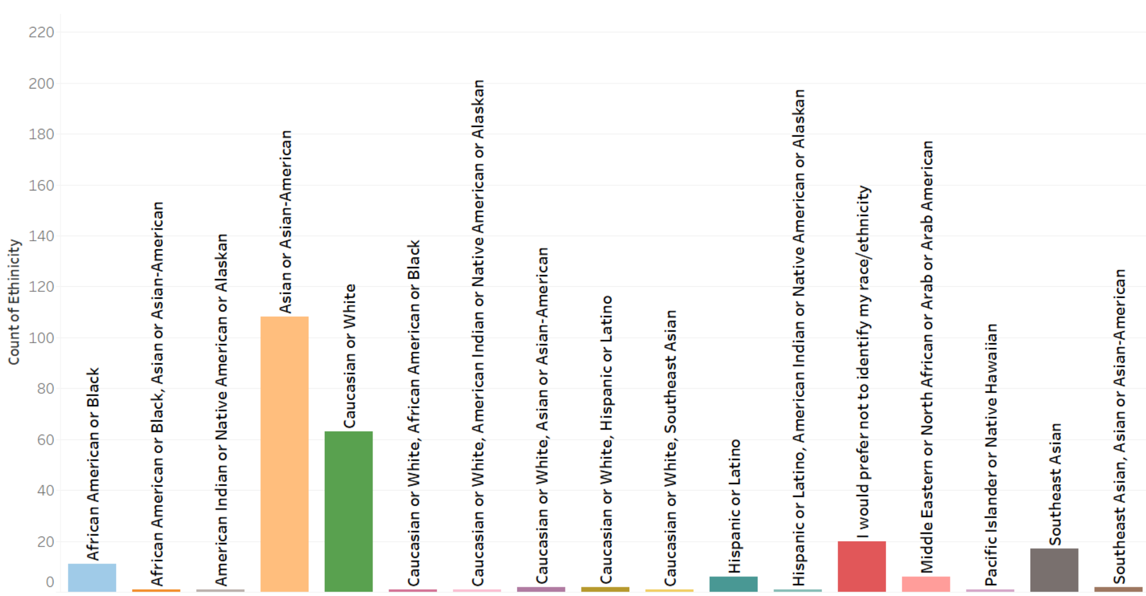

Figure 3.5: Ethinicity Distribution in the Dataset.

The student population include students from both graduate and undergraduate courses. The survey included options of freshman, sophomore, junior, senior or more and Masters. We could see the distribution of data based on School year in "Fig. 3.6"
.

Figure 3.6: School Year Distribution in the Dataset.

3.3 Business Data - Net Promotor Score

The Net Promotor Score (NPS) Data [176] was collected by telephone surveys on customer satisfaction. Net Promotor Score is a standard metric used for measuring customer satisfaction by labeling customers as 'Promotor', 'Passive', or 'Detractor'.The actual dataset was collected in a span of years. In this study we use the Text feedback from customers collected during the years 2015 - 2016 along with other features containing customer details, survey details, and benchmark questions on which the service is being evaluated. The surveys are for 38 companies located across United States and parts of Canada. We process the text feedback and derive the decision attribute as Customer Emotions. The decision problem is to improve customer satisfaction/loyalty by identifying factors to improve customer emotions. If a customer feels joy, or trust then it is highly likely that the customer remains loyal to the business and also improves Net Promotor Score.

Table 3.7: Dataset Properties: Business Data - Net Promotor Score (NPS)

Property	**NPS Business Data: Client Comments Parts 2015**	**NPS Business Data: Client Comments Parts 2016**	**NPS Business Data: Client Comments Service 2015**	**NPS Business Data: Client Comments Service 2016**
Attributes	23 attributes including - Client Name - Division - SurveyType - ChannelType - BenchmarkAll: DealerCommunication - BenchmarkAll: Likelihoodto-beRepeatCustomer - Emotion	37 attributes including - Client Name - Division - SurveyType - ChannelType - BenchmarkAll: ContactStatusofFutureNeeds - BenchmarkParts: AvailabilityYN - BenchmarkParts: EaseofCompleting-PartsOrder - Emotion	24 attributes including - Client Name - Division - SurveyType - ChannelType - Benchmark: All - Ease of Contact - Benchmark: Service - Repair Completed Correctly - Benchmark: Service - Repair Completed Timely - Emotion	38 attributes including - Client Name - Division - SurveyType - ChannelType - Benchmark: All - Contact Status of Future Needs - Benchmark: All - Contact Status of Issue - Benchmark: Dealer Offers solutions to support the success of cus - Benchmark: Referral Behavior - Emotion
# of instances	6656	10102	11121	17706

CHAPTER 4: METHODOLOGY

4.1 Pre-processing

Pre-processing is one of the important steps in handling text data. This involves removal of noisy and unwanted parts from the text fow which we use Python Natural Language Toolkit (NLTK) [200]. The following steps are involved in pre-processing of student course evaluation comments: Tokenization, lower case, stop words removal.

4.1.1 Tokenization

Tokenization is the process of splitting the text or sentence into words. In specific it is the task of chopping character sequences into pieces called tokens (words) and removing certain characters like punctuation. An example is shown in Fig. 4.1.

4.1.2 Stop Words Removal

Some of the words in English language are frequently used in order to make the sentence more complete in terms of grammar. These words are generally not much useful in terms of the context of the sentence in most of the cases. For instances words like ‘am’, ‘is’, ‘was’, ‘are’ etc. There is list of stop words available in the Python Natural Language Toolkit (NLTK) [200] corpus which is used as part of this stop words removal step.

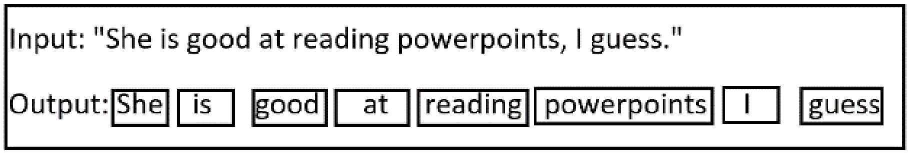

Figure 4.1: Tokenization.

4.1.3 Case Folding

Natural language text written by human beings contains both lower case and upper case. In terms of processing this kind of text using a machine requires all the text to be in same case for better performance. This step changes the text to lower case

4.1.4 Replace Slang Words

Microblog text contains abudance of slang words. This leads to incorrect tagging so these words are replaced with formal text, for example b4 $\rightarrow$ before, chk $\rightarrow$ check etc

4.1.5 Special Character and Tagging

The data (described in section 3.2.2) is pre-processed in order for it to be suitable for pattern discovery process. Pre-processing involves removal of special characters like '-', spaces, '/'. The whole survey data consisted upto 60 attributes in the following categories: Team Work, ActiveLearning and TeachingMethod, Emotions, Experience with Faculty, Background, Campus Facilities.

These categories are used to split the data into two parts to provide meaningful insights from the experiments Table. 4.1

Table 4.1: Student Survey Data Categories.

SubSet No	Category
Data1	TeamWork, Student Emotion
Data2	ActiveLearning, TeachingMethod, Student Emotion

<term><tab><Affect Category><tab><Association Flag>

<term> is a word for which emotion associations are provided;
<Affect Category> is one of eight emotions (anger, fear, anticipation, trust, surprise, sadness, joy, or disgust) or one of two polarities (negative or positive);
<Association Flag> has one of two possible values: 0 or 1. 0 indicates that the target word has no association with affect category, whereas 1 indicates an association.

Figure 4.2: National Research Council - Word Level Annotation.

<table>
<tr><th>Emotion</th><th>Emoticons</th></tr>
<tr><td>sadness</td><td>>:[:-(:(:-c :c :-< :< :-[:[:{</td></tr>
<tr><td>anger</td><td>:-|| :@> :(</td></tr>
<tr><td>joy</td><td>:) ;) =) :] :P :-P ;P :D ;D :> :3 :-) ;-) :^) :o) ;^) :-D :-></td></tr>
<tr><td>surprise</td><td>:-o :-O o_O O_o :$</td></tr>
<tr><td>disgust</td><td>D:< D: D8 D; D= DX v.v</td></tr>
</table>

Figure 4.3: Emoticons.

4.2 Emotion Labeling

To identify the emotion class, we use the National Research Council - NRC lexicon [78], [79]. The Annotations in the lexicon are at word-sense level. Each line has the format: <Term> <AffectCategory> <AssociationFlag> as shown in Fig. 4.2.

Apart from word level annotation, to increase the weightage of each emotion class assigned to tweet we also use the hashtags and emoticons inside the tweet text. For hashtags, we utilize the National Research Council - NRC Hashtag Emotion Lexicon [201] [129] which is a list of words and their associations with eight emotions (anger, fear, anticipation, trust, surprise, sadness, joy, and disgust). The associations are computed from tweets with emotion-word hashtags such as #happy and #anger. All emoticons were retained in the data collection process and validated while assigning weights to each emotion class for a tweet. Fig. 4.3. shows the list of emoticons used in this process. Fig. 4.4. Explains the steps involved in assigning final emotion class.

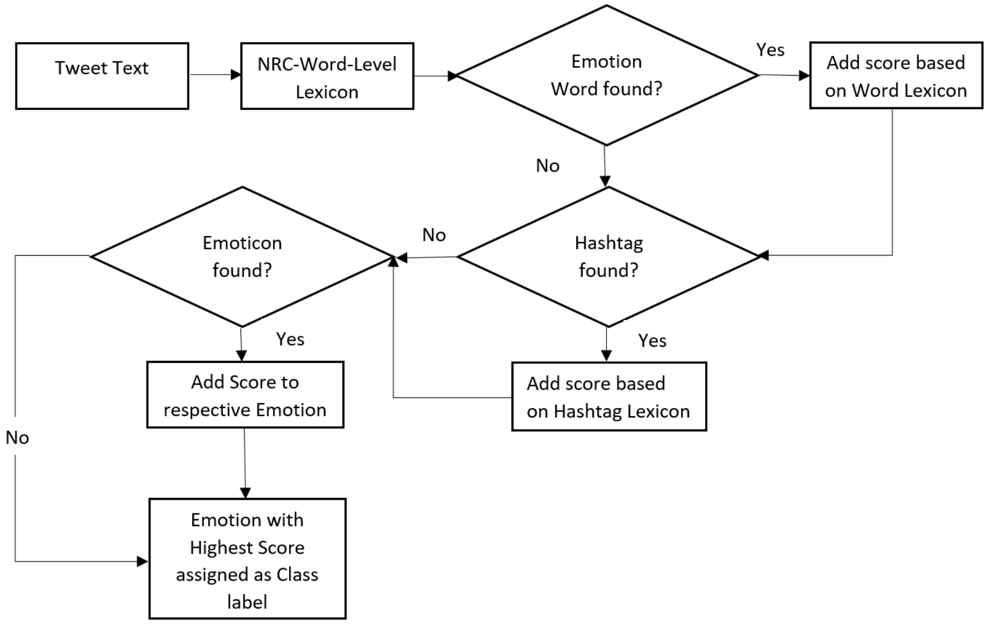

Figure 4.4: Emotion Labeling.

4.3 Classification

A systematic technique to build classification model from input data set is called Classification. It is the task of assigning objects to one of several predefined categories called class labels. Some of the classifiers include naive bayes, support vector machines, neural networks, decision tree classifiers, and rule-based classifiers. Authors [202] provide a review of common machine learning algorithms for text classification. Document classification can be divided into three categories based on the available methods as follows: supervised, un-supervised, semi-supervised methods. The automatic classification of documents into predefined categories has observed as an active attention, as the internet usage rate has quickly enlarged [202]. Some of the common machine learning approaches are Support Vector Machine (SVM), Bayesian classifier, Decision Tree, K- Nearest Neighbor(KNN), Neural Networks, Latent Semantic Analysis etc. Since this paper focus on text classification this section details specific classifiers widely used in the literature for text classification. In general, supervised learning techniques are used for automatic text classification. Here, pre-

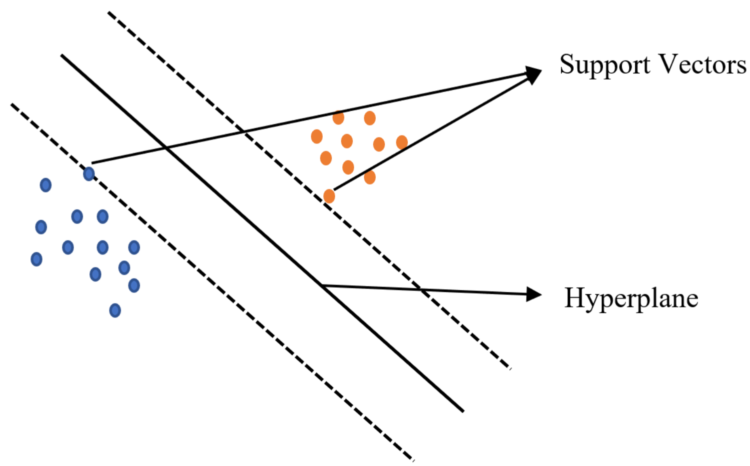

Figure 4.5: Support Vector Example.

defined category labels are assigned to documents based on the likelihood suggested by a training set of labeled documents.

4.3.1 Support Vector Machines - SVM

Support Vector Machine(SVM) is a statistical learning method introduced by [203]. Authors [204], details Support Vector Machine as follows: The main idea behind SVM is to find a decision surface that best separates the two class of documents in the n-dimensional space. The samples (documents) that are close to decision surface are called *support vectors* shown in Fig.4.5 as shown by author [205]. Major advantage of SVM is its superior runtime-behavior during the categorization of new documents because only one dot product per new document must be computed. A disadvantage is the fact that a document could be assigned to several categories because the similarity is typically calculated individually for each category. Nevertheless, SVM is a very powerful method and has outperformed other methods in several studies by [206], [207], [208], [209]. The following works use Support Vector Machine for emotion classification from text [110], [116], [130], [131].

4.3.2 Naive Bayes Classifier

Naive Bayes classifier is a simple probabilistic classifier based on applying Bayes Theorem with strong independence assumptions [202]. Authors [204], say that Independence assumption means the order of features is irrelevant and presence of one feature does not affect the presence of other features. The conditional assumption is given in equation (4.1).

$$p(x|y = c) = \prod_{i=1}^{D} p(x_i|y = c) \tag{4.1}$$

The computation of Bayes classifier is efficient because of this independence assumption and also limited applicability. Due to its apparently over-simplified assumptions, the naive bayes classifiers often work much better in many complex real-world situations [202]. The full Bayesian posterior predictive density on the class label Y given an input X and the training data D is given by equation (4.2) as explained by author [210].

$$p(y = c|x, D) \approx p(y = c|x, \hat{\theta}, \hat{\pi}) \propto p(x|y = c, \hat{\theta}_c)p(y = c|\hat{\pi}) \tag{4.2}$$

But the performance is relatively low compared to Support Vector Machines. One of the advantage of Naive Bayes classifier is that it requires only small set of training data to determine the classification instances. Naive Bayes is easy to implement compared to other algorithms, however because of conditional independence assumption it's performance is very poor when features are highly correlated and does not consider frequency of word occurrences. Authors [48] use Naive Bayes for Twitter emotion classification.

4.3.3 Vector Space Model

K-Nearest Neighbor is vector Space classification method. Vector space classification method represents each document as a vector with one real-valued component, for instance term frequency - inverse document frequency (tf-idf) [116] weight for each term. In general vector space model is based on contiguity hypothesis. This hypothesis states that documents in the same class form a contiguous region and regions of different classes do not overlap [211]. Given a test document, majority class (nearest neighbor) close to the test document is assigned as the class for test document. One of the advantage of K-NN is that, it does not require explicit training data. Because the training phase involved determining the value of 'k' and document pre-processing. KNN is also called as memory-based learning or instance-based learning because it simply memorizes examples in the training set and then compares the test document to them. For example, if there are documents of type science and sports. Given a test document K-NN classifies it based on the majority number of classes that are closest to the test document. Consider the example in Fig. 4.6. the test document is close to science document and hence classifies as *'science'*. Authors [119], [124] use Vector Space Model for text emotion classification.

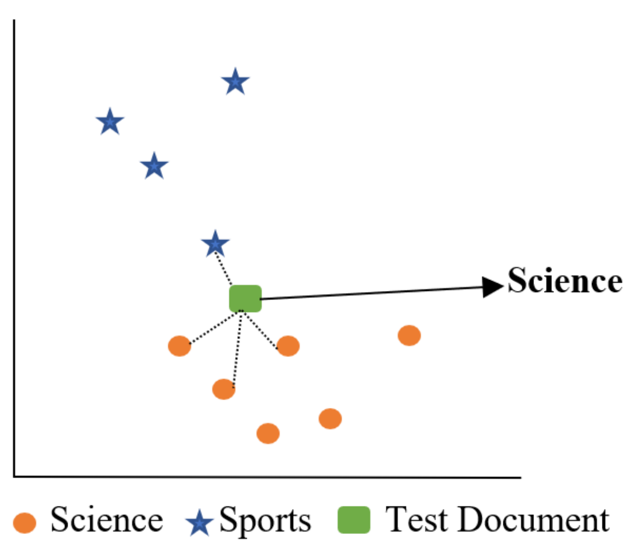

Figure 4.6: K-Nearest Neighbor Example.

4.3.4 Decision Tree Classifier

The classification problem is solved with the help of tree. The tree has root node, internal node and leaf node as shown in Fig.4.7. Here leaves represent the document category and branches represent features that lead to the specific category. The root node is the document for classification. According to authors [202], main advantage of decision tree is its simplicity in understanding and interpreting, even for non-expert users.

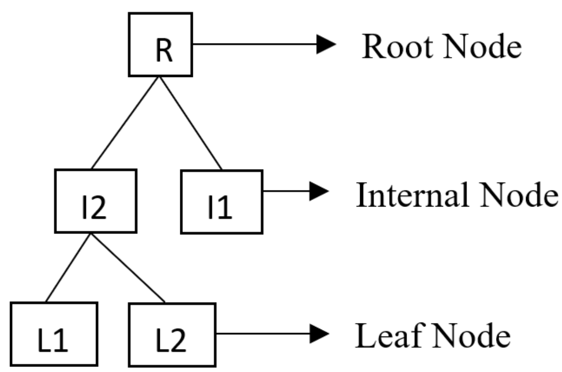

Figure 4.7: Decision Tree Nodes.

Text classification generally involves more number of features or attributes. Decision tree performs poorly with larger feature set. However, if the feature set is organized and limited according to the requirement then the performance of decision tree is an added advantage to the simplicity and understand-ability.

4.3.5 Recurrent Neural Networks

Recurrent Neural Network (RNN) concept was introduced by J. J. Hopfield [212]. The structure of basic RNN is given in Fig. 4.8.

Recurrent Neural Network (RNN) is a class of artificial neural network where connections between units form a directed graph along a sequence. This allows it to exhibit dynamic temporal behavior for a time sequence. Unlike feed- forward neural networks, RNNs can use their internal state (memory) to process sequences of inputs.

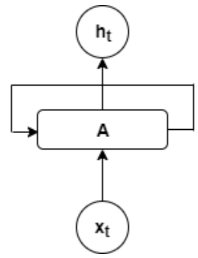

Figure 4.8: Structure of basic Reccurrent Neural Network (RNN).

RNN model uses context to provide most appropriate output based on input values. In RNN, all the input data is related to each other in order to predict the better result. This step is part of model training phase. Standard recurrent neural network (RNN) can map vectors of sentences of variable length to a fixed-length vector by recursively transforming current sentence vector with the output vector of the previous step. This can result into growing or decaying gradient exponentially over a long input sequence. This is known as gradient vanishing or exploding problem.

4.3.5.1 Gated Recurrent Unit Network

Gated Recurrent Neural Networks (GRNNs) [213] Fig. 4.9, is a modern variation of Recurrent Neural Networks (RNNs).The aim of Gated Recurrent Unit Network is to solve the vanishing gradient problem of standard recurrent neural network by using update gate and reset gate. These update gate and reset gate are the two vectors which decide what information should be passed to the output. The special thing about these vectors is these vectors can be trained to keep information from long ago, without washing it through time or remove information which is irrelevant to the prediction.

In this work we built a RNN - GRU using Spark Big DL [214] a distributed deep learning framework for Big Data platforms and workflows as shown in Fig. 4.10. It is implemented on top of Apache Spark, and allows users to write their deep learning applications as standard Spark programs (running directly on large-scale big data clusters in a distributed fashion).

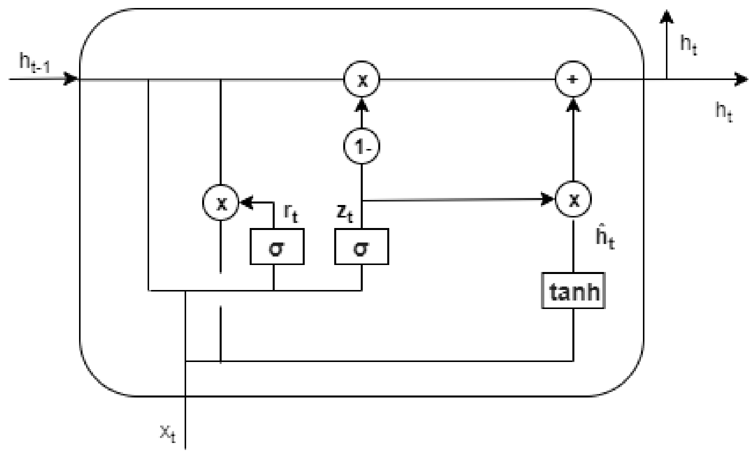

Figure 4.9: Gated Recurrent Unit (GRU).

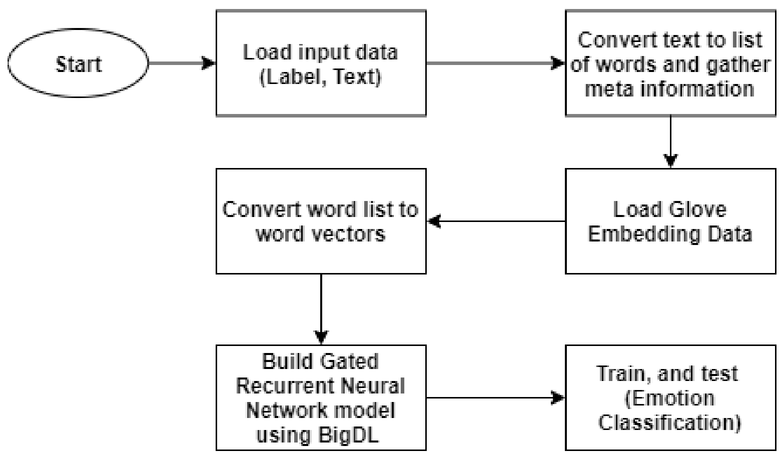

Figure 4.10: Methodology - Gated Recurrent Unit (GRU).

4.4 Evaluation

Classification involves the process of building an optimized classifier and use the optimized classifier to predict unknown data [215]. Generally data classification is divided onto one of the following categories, binary, multi-class, multi-labelled, hierarchical [216]. In order to generate an optimized classifier and assess the performance, evaluation metrics are used. There are three major categories of evaluation metrics [217] as follows: probablity metrics (Root Mean Squareed Error - RMS, Cross

Entrophy), threshold metrics (Accuracy, F-Score) and ranking metrics (Area Under Curve - AUC, Average Precision - APR). This section explains some of the commonly used metrics with formal definitions.

4.4.1 Confusion Matrix

Confusion matrix is a table that summarizes the results of the classifier for further inspection with appropriate metrics. The table rows denote the predicted class and columns denote the actual class (Table 4.2). True Positive (TP) and True Negative (TN) denotes the number of correctly classified instances, on the other hand False Positive (FP) and False Negative (FN) denotes the number of misclassified instances.

Table 4.2: Confusion Matrix - Example for Binary Class [1].

Class	Positive Class - Actual	Negative Class - Actual
Positive Class - Predicted	True Positive (TP)	False Negative (FN)
Negative Class - Predicted	False Positive (FP)	True Negative (TN)

4.4.2 Metrics - Accuracy, F-Measure, Precision, Recall

Table 4.3. lists some of the commonly used evaluation metrics.

1. *Accuracy* is the overall effectiveness of the classifier, which is the ratio of the correct predictions over the total number of instances evaluated.

2. *Precision* is the measure of correctly predicted positive instances from the total predicted instances in the positive class.

3. *Recall* is the measure of fraction of possitive instances that are correctly classified.

4. *F-Measure* is used to maximize the true positive (TP) rate and true negative (TN) rate, simultaneously keeping both the rates relatively balanced.

5. *Specificity* is the measure of fraction of negative instances that are correctly classified.

6. *Area Under Curve* measure the classifier's ability to avoid false classification.

7. *Average Accuracy* is the measure of overall effectives of the classifier.

8. *Error Rate* Average per class classification error.

Table 4.3: Evaluation Measures.

Measure	Formula
Accuracy	$\frac{TP+TN}{TP+FP+TN+FN}$
Precision (P)	$\frac{TP}{TP+FP}$
Recall (R)	$\frac{TP}{TP+FN}$
F-Measure	$\frac{2*P*R}{P+R}$
Specificity (S)	$\frac{TN}{FP+TN}$
Area Under Curve	$\frac{1}{2}(R+S)$
Average Accuracy	$\frac{\sum_{i=1}^{l}\frac{TP_i+TN_i}{TP_i+FN_i+FP_i+TN_i}}{l}$
Error Rate	$\frac{\sum_{i=1}^{l}\frac{FP_i+FN_i}{TP_i+FN_i+FP_i+TN_i}}{l}$

4.5 Evaluation Procedures

There are two general methods for evaluating the classifier models: Hold-Out and Cross Validation. Hold-Out method divides the data into train and test splits and is known for its speed, simplicity and flexibility. Cross Validation is used when only limited amount of data is available and to achieve unbiased estimate of the model performance. In K-fold cross validation, the dataset is divided into k subsets/splits and the model runs k times. This method provides better estimate of performance but slower than the hold-out method.

4.6 Actionable Pattern Mining

Actionability is a property of the discovered knowledge. Patterns are considered Actionable if the user can act upon them, and if this action can benefit the user, or help them to accomplish their goals. Action Rules mining is a method to extract Actionable patterns from the data. Action Rules are rules that describe a possible transition of data from one state to another more desirable state.

4.6.1 Data Split Method

For the initial experiments we use the method proposed by authors bagavathi et al. [218] to extract action rules for the emotion labeled Twitter dataset. The method 2 in [218] is to extract action rules by vertical split of the data. This method utilizes association action rules [29] which follows iterative method to extract all possible action rules. In order to overcome the computational complexity and expense, authors in [218] propose vertical data split method for faster computation and parallel processing. This method does not scale well for the dense Twitter dataset. Hence we propose a new novel method explained in section 4.6.2.

4.6.2 Hybrid Actionable Knowledge Discovery Method

We propose a novel approach of hydrid Action Rule generation, combining the rule based and object based approach of Action Rule mining to reduce the overhead of Action Rule iterative procedure. The algorithm pseudocode is given in Fig.4.11.

We now give an example of how the algorithm works with the help of Information System. The information system in Table. 2.1 is denoted as Decision system if

```
Algorithm(certainRules, decisionFrom, decisionTo, support, confidence)
    (where certainRules are provided by algorithm LERS)
    for each rule r in certainRules
        if consequent(r) equals decisionTo
            Form ActionRuleSchema(r)
            ARS ← ActionRuleSchema(r)
        end if
    end for
    for each schema in ARS
        Identify objects satisfying schema
        Form subtable
        Generate frequent action sets using Apriori
        Combine frequent action set to form Action Rules
        (Such that the frequent action sets satisfy the decisionFrom → decisionTo)
        Output ← Action Rules
    end for
```

Figure 4.11: Hybrid Action Rule Algorithm.

the attributes $\mathbb{M}$ are classified into flexible M_{fl} , stable M_{st} and decision d, $\mathbb{M} = (M_{st}, M_{fl}, \{d\})$. From Table. 2.1 $M_{st} = \{A, B, C\}$, $M_{fl} = \{E, F, G\}$, and $d = D$.

In this example we intend to re-classify the decision attribute D from $D_2 \rightarrow D_1$. First the algorithm Fig. 4.11. uses LERS method explained in section " 2.5.2" to extract the certain classification rules and generate Action Rule schemas as given in " 4.3" , " 4.4".

$$[\mathrm{B_1} \wedge \mathrm{C_1} \wedge (\mathrm{F}, \rightarrow \mathrm{F_1}) \wedge (\mathrm{G}, \rightarrow \mathrm{G_1})] \rightarrow (\mathrm{D}, \mathrm{D_2} \rightarrow \mathrm{D_1}). \tag{4.3}$$

$$[(\mathrm{E}, \rightarrow \mathrm{E_1})] \rightarrow (\mathrm{D}, \mathrm{D_2} \rightarrow \mathrm{D_1}). \tag{4.4}$$

Then the algorithm proceeds by creating subtable based for each of the Action Schema. For instance " 4.3", generates the following subtable Table. 4.4.

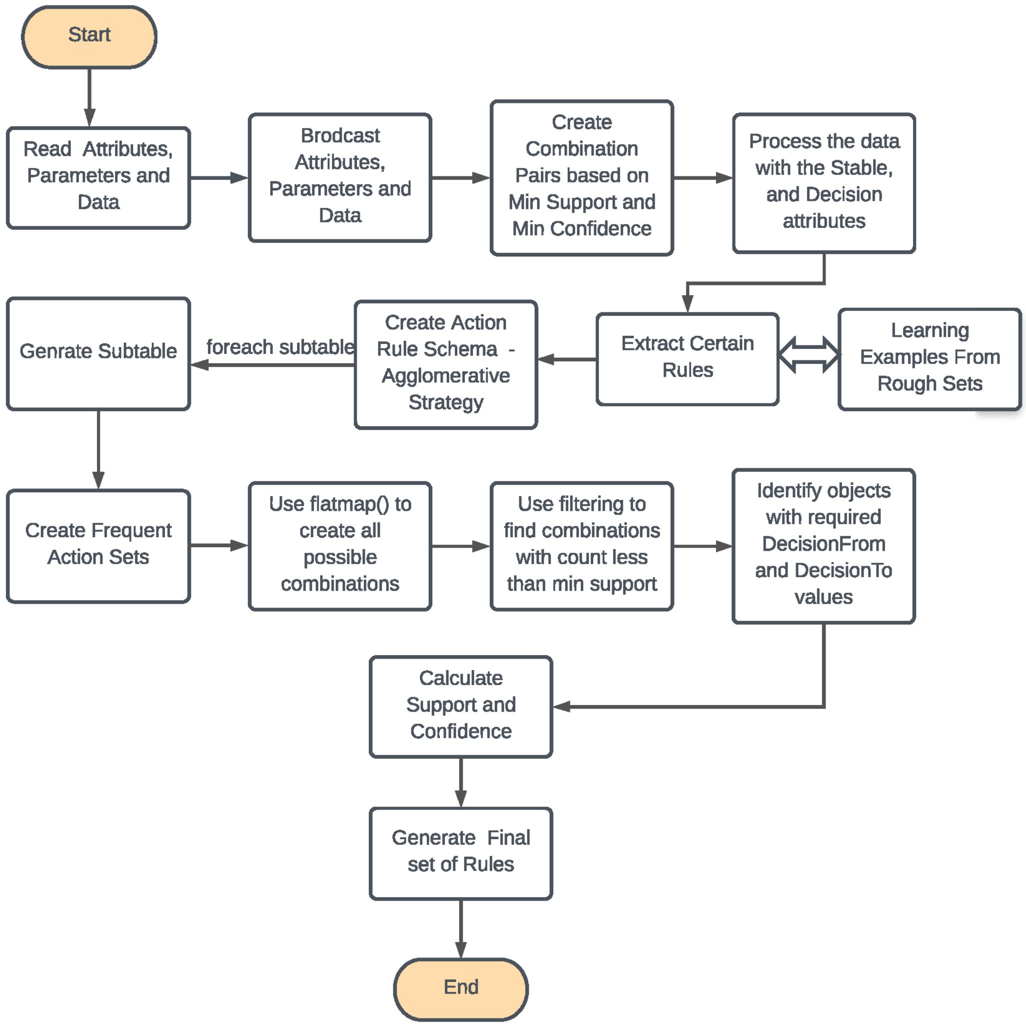

Figure 4.12: Hybrid Action Rule Algorithm - Flowchart.

Table 4.4: Subtable for Action Rule Schema

X	**B**	**C**	**F**	**G**	**D**
x_1	B_1	C_1	F_2	G_1	D_1
x_3	B_1	C_1	F_2	G_3	D_2
x_6	B_1	C_1	F_3	G_1	D_2
x_8	B_1	C_1	F_2	G_3	D_2

Association Action Rule extraction is an exhaustive Apriori based method which extracts complete set of Action rules by taking all possible combinations of the action terms. It is an iterative procedure and does not scale very well in case of dense and high dimensional dataset. In this work we create subtables by using the Action Rule Schemas in a highly dense data as explained above. We perform Association Action Rule extraction algorithm on each of the subtables in parallel which allows the algorithm to complete and generate rules in a much faster time compared to the existing algorithms and systems. In our sample dataset example, the algorithm generates following Action Rules " 4.5" based on the subtable Table. 4.4.

$$[\mathrm{B}_1 \wedge \mathrm{C}_1 \wedge (\mathrm{F}, \rightarrow \mathrm{F}_1) \wedge (\mathrm{G}, \mathrm{G}_3 \rightarrow \mathrm{G}_1)] \rightarrow (\mathrm{D}, \mathrm{D}_2 \rightarrow \mathrm{D}_1). \tag{4.5}$$

The hybrid algorithm is implemented in Spark [40], runs separately on each subtable and does transformations like map(), flatmap(), join() and other distributed operations. The overall operating methodology is shown in Fig. 4.12.

4.6.3 Intelligent Attribute Selection Method - Using Rough Sets

4.6.3.1 Computation of Reducts using Discernibility Matrix

This method computes the discernibility matrix [219] for the given dataset, which is then used for computing reducts [220].

4.6.3.2 Computation of Reducts using Quick Reduct Algorithm

We use the QuickReduct [221] algorithm, here the computation time depends on the number of attributes instead of the number of objects. Thus it runs faster than the one computation method, which uses the Discernibility Matrix section 4.6.3.1.

4.6.3.3 SparkR and Rough Sets

SparkR [3] is an R package for the use of Apache Spark with R. Spark 2.4.4, SparkR provides a distributed data frame implementation that supports operations like selection, filtering, aggregation etc. (similar to R data frames, dplyr) but on

Algorithm 1: Reduct Computation

Input: Information System A
Output: $Reduct_A$(A)

```
compute the indiscernibility matrix M(A) = (C_ij)
Reduce M using absorption laws
d - number of non-empty fields of reduced M
Build set of Reducts R_0, R_1, ... , R_d
begin
R_0 = φ
for i = 1 to d
begin
R_i = S_i ∪ T_i, where S_i = {R ∈ R_{i-1} : R ∩ C_i ≠ φ}
T_i = (R ∪ {a})_{a∈C_i, R∈R_{i-1} : R∩C_i=φ}
end
end
Remove the redundant elements from R_d
RED_A(A) = R_d
```

Algorithm 2: Quick Reduct Algorithm

Input: Decision Table A
Output: Superreduct R

```
R ← {} ;
repeat the following
T ← R ;
foreach x ∈ (A - R) do
if γ_{R∪{x}} > γ_T then T ← R ∪ {x};
R ← T;
end
until γ_R == γ_A
```

large datasets. SparkR also supports distributed machine learning using MLlib. The SparkR architecture Fig. 4.13, consists of R to JVM brige on the driver for submitting jobs to cluster and spark executor on the worker for running R. For the purpose of our experiements we use the SparkR Roughsets package [222].

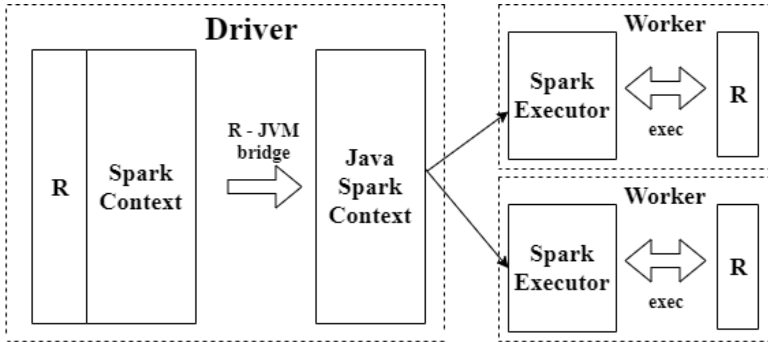

Figure 4.13: SparkR Architecture [3].

4.7 Visualization

Visual images has been an effective way to communicate since ancient history including but not limited to cave paintings, and Leonardo da Vinci's revolutionary methods of technical drawing for engineering and scientific purposes. In the present data era, one of the biggest challenges is representating the data in an understandble format. Visualization is a powerful tool for exploring large data, both by itself and coupled with data mining algorithms [223]. In computer science, data visualization focusses on use of computer supported tools to explore, represent, and understand large amounts of data. In this work, for the purpose of data visualization, we use Tableau software [224]. The following are some of the visualization techniques used:

4.7.1 Bar Chart

Bar Chart Fig. 4.14 [225] uses horizontal or vertical bars to represent discrete numerical comparisons between different categories. One of the axis in the bar chart represents the categories and other represents the discrete value on a scale. This type of visualization technique is mainly used to answer questions like “how many in a category?". In our work we use bar chart to represent the demography and gender distribution of student data, and the run time of algorithm executions.

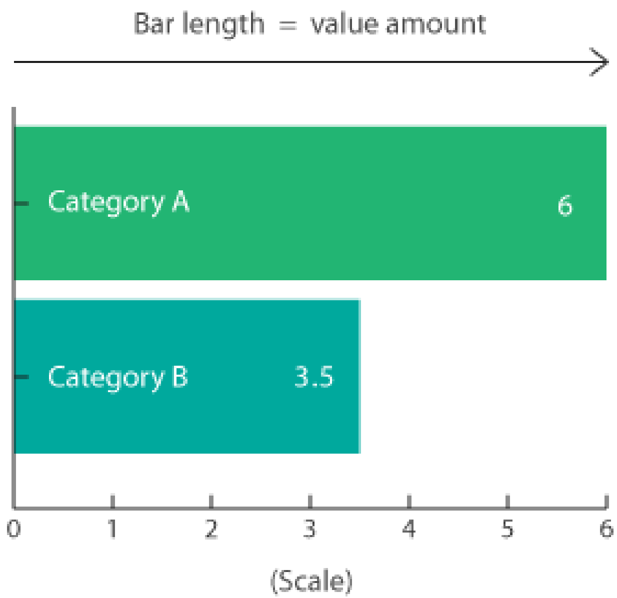

Figure 4.14: Sample - Bar Chart Visualization.

4.7.2 Tree Maps

Tree map Fig. 4.15 visualization technique is generally used to visualise tree structure and also display quantities for each category. In this work we use tree map visualization to display quantity in each category in terms of emotion labels in the data. Each category is assigned a rectangle area. When a quantity is assigned to a category, its area size is displayed in proportion to that quantity.

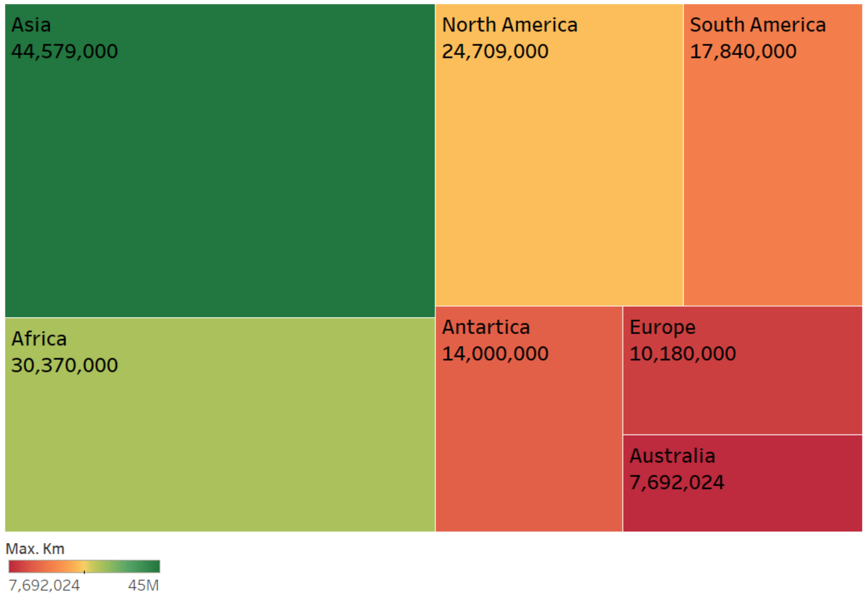

Figure 4.15: Sample - Tree Map Visualization - Area (Km^2).

4.7.3 Line Graph

Line Graphs Fig. 4.16 [225] display quantitative values over a continous interval or time period. It features how data changes over time. In a typical line graph, x-axis shows timescale or sequence of intervals and y-axis displays the quantitative values. Line with updard slope indicates there is an increase and downward slope indicates there is a decrease in the value. In our work we use the line graph to display how student emotions change over time - based on the active learning pedagogies implemeted in the course.

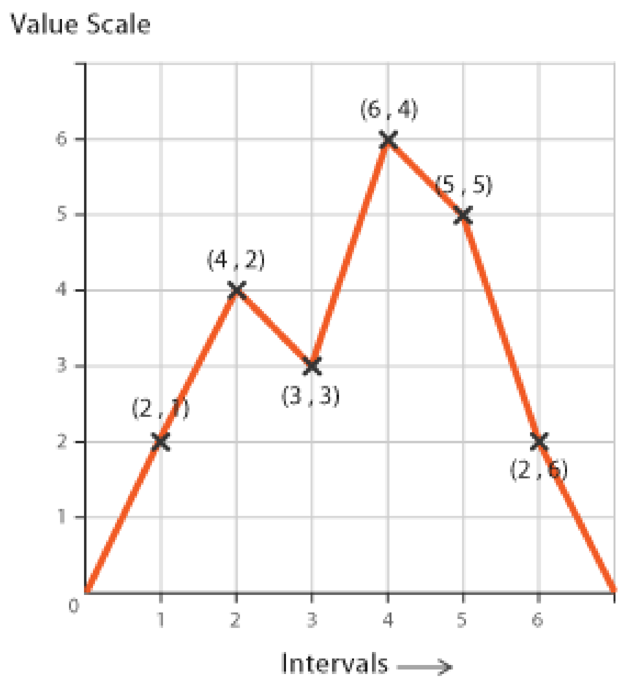

Figure 4.16: Sample - Line Graph Visualization.

4.7.4 Pie Chart

Pie Charts Fig. 4.17 [226] show proportions and percentages between categories (circle is divided into proportional segments). It gives a quick idea of the propotional data distribution.

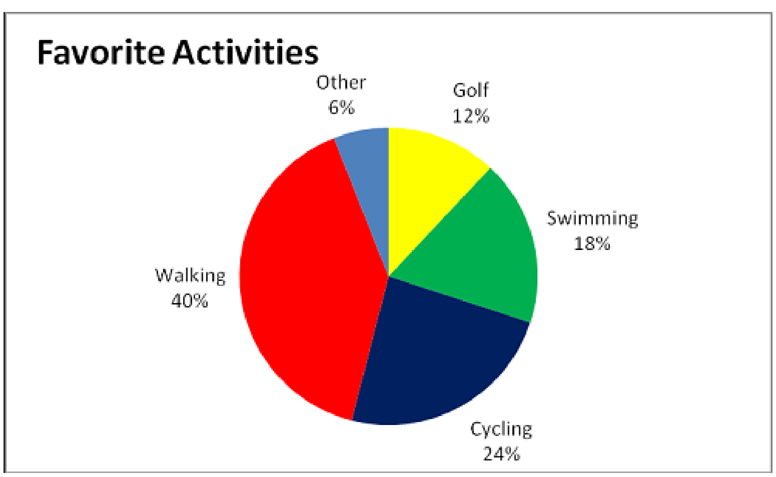

Figure 4.17: Sample - Pie Chart Visualization.

CHAPTER 5: EXPERIMENTS AND RESULTS

5.1 Social Network Data

5.1.1 Sentiment Analysis and Actionable Pattern Mining

Our research is focussed on recommending methods to improve emotions from negative to positive and neutral to positive and increasing friends count of a user. For this experiment, we used live tweets extracted using Twitter Search API on the latest tweets. The Twitter Search API searches against a sampling of recent tweets published in the past 7 days. Data Collection includes user-generated updates collected directly from social media API as they allow subscription to a continuous live stream of data. Our data contains the following attributes: RetweetCount, IsFavorited, UserID, UserFriendsCount, UserFavoritesCount, UserFollowersCount, TweetText, UserLanguage, TweetSentiment, and TweetVerb. We collected about 28,000 instances. Table 5.1. gives the description about the dataset. The Hadoop research cluster at University of North Carolina Charlotte was used to perform the experiments. This cluster has 6 nodes connected via 10 gigabits per second Ethernet network.

Table 5.1: Sentiment Analysis.

ReTweet	IsFavorited	Friends	Followers	Language	Text	Sentiment	Verb
0	FALSE	247	30795	en	PLS HELP. SAVEA GRADE.	Negative	HELP
0	FALSE	42	527	13	TheFlash: Thanks for watch-ing!New episodes return January 24.	Positive	watching

We used action rules to change the emotion from negative to positive and neutral to positive, also to change from lower number of friends count to higher number of friends. Our data contains the following attributes: RetweetCount, IsFavorited, UserID, UserFriendsCount, UserFavoritesCount, UserFollowersCount, TweetText, UserLanguage, TweetSentiment, and TweetVerb. In Fig. 5.1 we can see that most of the comments are negative. We focus on the issue of improving the comments emotion from negative to positive and neutral to positive by providing actionable patterns to improve the emotions. Also, we suggest actionable patterns to improve friends count. In our data, we noticed that positive comments have more favorites compared to others as shown in Fig. 5.2.

Considering the recent growth of the amount of data collected nowadays, we use distributed implementation of the proposed method LERS and ARAS using Hadoop Map Reduce framework by Tzacheva et al. [44]. We show that computation is much faster in the distributed framework than on single computer. The experiment results

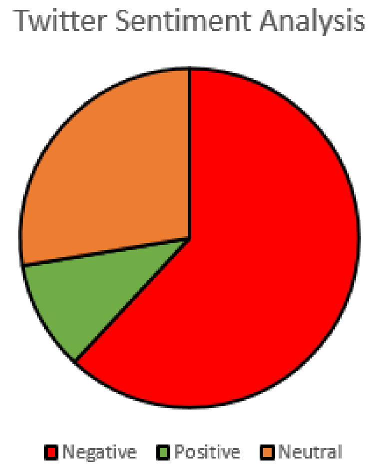

Figure 5.1: Tweets Sentiment Analysis.

are shown in Table 4. We can scale to large social media data sizes. It is considered that the workload can be spread across two nodes and can increase the number of mappers and scale to very large size data and handle it appropriately.

Let us consider AR1 from Fig. 5.3. If user's favorites count increases from 0-100 to 1001-5000 and user's followers count increases from 101-200 to 701-800 and user language is English, then tweet sentiment could be changed from neutral to positive. This rule is generated with a confidence of 100% and support 2 for the Twitter data with following attributes: RetweetCount, IsFavorited, UserID, UserFriendsCount, UserFavoritesCount, UserFollowersCount, UserLanguage and TweetSentiment. In future if more attributes relevant to the context of text like frequency of Part-of-Speech including adjectives is added then we anticipate that the action rules generated by our system would be more intuitive.

Considering the recent growth of the amount of data collected nowadays, we use distributed implementation of the proposed method LERS and ARAS using Hadoop Map Reduce framework by Tzacheva et al. [44]. We show that computation is much faster in the distributed framework than on single computer. The experiment results are shown in Table 5.2. We can scale to large social media data sizes. It is considered

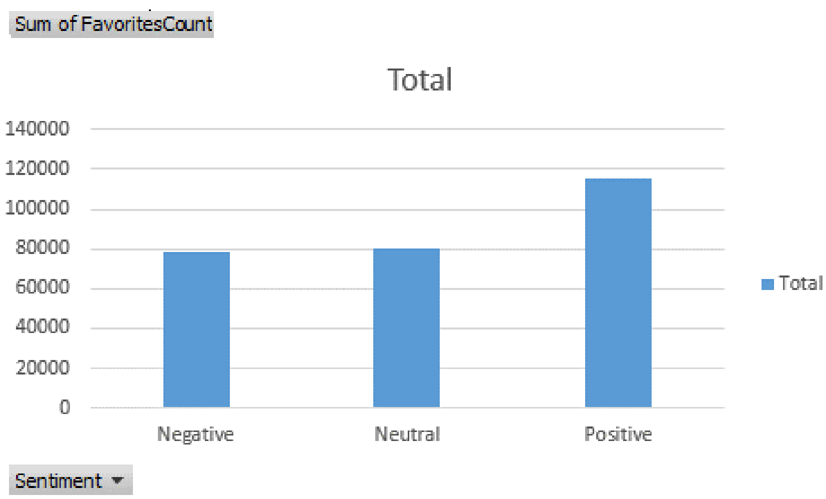

Figure 5.2: Favorite Counts For Various Sentiments.

Action Rule No.	Action Rule
AR1	(UserFavoritesCount, 0-100 → 1001-5000) ^ (UserFollowersCount, 101-200 → 701-800) ^ (UserLanguage = en-gb) ➔ (TweetSentiment, Neutral → Positive) [Support: 2, Old Confidence: 66%, New Confidence: 100%]
AR2	(UserFavoritesCount, 101-200 → 701-800) ^ (UserFollowersCount, 0-100 → 301-400) ^ (UserFriendsCount, 101-200 → 201-300) ^ (UserLanguage = en) ➔ (TweetSentiment, Neutral → Positive) [Support: 2, Old Confidence: 80%, New Confidence: 100%]
AR3	(UserFavoritesCount, 201-300 → 601-700) ^ (UserFollowersCount, 5001-10000 → 301-400) ^ (UserFriendsCount, 501-600 → 701-800) ➔ (TweetSentiment, Neutral → Positive) [Support: 2, Old Confidence: 100%, New Confidence: 100%]

Figure 5.3: Sample Action Rules.

that the workload can be spread across two nodes and can increase the number of mappers and scale to very large size data and handle it appropriately.

Table 5.2: Single Node and Hadoop Cluster Run Time comparison.

Experiment	Time Taken Single Node	Time Taken Hadoop
Experiment1	432 seconds	258 seconds
Experiment2	270 seconds	180 seconds
Experiment3	273 seconds	192 seconds

Single Node Action Rules	Hadoop Action Rules
(TweetSentiment, Negative → Positive) ^ (UserFavoritesCount, 0-100 → 10001-15000) ^ (UserFollowersCount, 0-100 → 1001-5000) ^ (UserLanguage = pt) ➔ (UserFriendsCount, 0-100 → 1001-5000) [Support: 2, Old Confidence: 67%, New Confidence: 100%]	(TweetSentiment, Negative → Neutral) ^ (UserFavoritesCount, 0-100 → 5001-10000) ^ (UserLanguage = it) ➔ (UserFriendsCount, 0-100 → 1001-5000) [Support: 2, Old Confidence: 60%, New Confidence: 100%]
(TweetSentiment, Neutral → Positive) ^ (UserFavoritesCount, 0-100 → 10001-15000) ^ (UserFollowersCount, 0-100 → 1001-5000) ^ (UserLanguage = pt) ➔ (UserFriendsCount, 0-100 → 1001-5000) [Support: 2, Old Confidence: 76%, New Confidence: 100%]	(TweetSentiment, Neutral → Positive) ^ (UserFavoritesCount, 601-700 → 1001-5000) ^ (UserLanguage = de) ➔ (UserFriendsCount, 0-100 → 1001-5000) [Support: 2, Old Confidence: 100%, New Confidence: 100%]
(TweetSentiment, Negative → Neutral) ^ (UserFavoritesCount, 20001-25000 → 601-700) ^ (UserFollowersCount, 0-100 → 5001-10000) ^ (UserLanguage = en) ➔ (UserFriendsCount, 0-100 → 1001-5000) [Support: 4, Old Confidence: 96%, New Confidence: 100%]	(TweetSentiment, Very negative → Neutral) ^ (UserFavoritesCount, 0-100 → 601-700) ^ (UserFollowersCount, 0-100 → 5001-10000) ^ (UserLanguage = en) ➔ (UserFriendsCount, 0-100 → 1001-5000) [Support: 2, Old Confidence: 100%, New Confidence: 100%]

Figure 5.4: Example action rules - Experiment 1.

5.1.1.1 Experiment 1 - Change from class UserFriendsCount: Low to High number of friends

UserFriendsCount, following attributes were used to generate action rules: Decision attribute - UserFriendsCount, Stable attribute - UserLanguage, Support - 2, Confidence - 60%. Sample action rules generated for this experiment are given in Figure 5.4.

5.1.1.2 Experiment 2 - Change class from TweetSentiment: Negative to Positive

Experiment 2: Negative to Positive, following attributes were used to generate action rules: Decision attribute - TweetSentiment, Stable attribute - UserLanguage, Support - 2, Confidence - 60%. Sample action rules generated for this experiment are given in Figure 5.5.

Single Node Action Rules	Hadoop Action Rules
(UserFavoritesCount, 10001-15000 → 601-700) ^ (UserFollowersCount, 101-200 → 301-400) ^ (UserFriendsCount, 301-400 → 701-800) ➔ (TweetSentiment, Negative → Positive) [Support: 2, Old Confidence: 60%, New Confidence: 100%]	(UserFavoritesCount, 0-100 → 601-700) ^ (UserFollowersCount, 5001-10000 → 301-400) ^ (UserFriendsCount, 201-300 → 701-800) ➔ (TweetSentiment, Negative → Positive) [Support: 2, Old Confidence: 100%, New Confidence: 100%]
(UserFavoritesCount, 101-200 → 15001-20000) ^ (UserFollowersCount, 101-200 → 701-800) ^ (UserLanguage = de) ➔ (TweetSentiment, Negative → Positive) [Support: 2, Old Confidence: 100%, New Confidence: 100%]	(UserFavoritesCount, 0-100 → 601-700) ^ (UserFollowersCount, 30000-Above → 301-400) ^ (UserFriendsCount, 201-300 → 701-800) ➔ (TweetSentiment, Negative → Positive) [Support: 2, Old Confidence: 100%, New Confidence: 100%]
(UserFavoritesCount, 501-600 → 601-700) ^ (UserFollowersCount, 101-200 → 301-400) ^ (UserFriendsCount, 801-900 → 701-800) ➔ (TweetSentiment, Negative → Positive) [Support: 2, Old Confidence: 100%, New Confidence: 100%]	(UserFollowersCount, 5001-10000 → 201-300) ^ (UserFriendsCount, 5001-10000 → 0-100) ^ (UserLanguage = es) ➔ (TweetSentiment, Negative → Positive) [Support: 3, Old Confidence: 75%, New Confidence: 100%]

Figure 5.5: Example action rules - Experiment 2.

5.1.1.3 Experiment 3 - Change class from TweetSentiment: Neutral to Positive

Neutral to Positive, following attributes were used to generate action rules: Decision attribute - TweetSentiment, Stable attribute - UserLanguage, Support - 2, Confidence - 60%. Sample action rules generated for this experiment are given in Fig. 5.6.

5.1.2 Emotion Mining and Actionable Pattern Discovery

5.1.2.1 Data Split Method

In this experiment, we extract action rules to identify what changes in attributes lead to change in emotion to a more positive state. For example, change from 'sadness' to 'trust', 'sadness' to 'joy', 'fear' to 'trust'. The dataset consists of continuous attributes which are discretized into intervals. The intervals are determined with the help of WEKA data mining software using unsupervised attribute discretization. The Table. 5.3. shows the parameters set to discretize the data. We use the following attributes AngerScore, TrustScore, FearScore, SadnessScore, AnticipationScore, Dis-

Single Node Action Rules	Hadoop Action Rules
(UserFavoritesCount, 30000-Above → 601-700) ^ (UserFollowersCount, 20001-25000 → 301-400) ^ (UserFriendsCount, 101-200 → 701-800) ➔ (TweetSentiment, Neutral → Positive) [Support: 2, Old Confidence: 66%, New Confidence: 100%]	(UserFavoritesCount, 30000-Above → 601-700) ^ (UserFollowersCount, 20001-25000 → 301-400) ^ (UserFriendsCount, 101-200 → 701-800) ➔ (TweetSentiment, Neutral → Positive) [Support: 2, Old Confidence: 66%, New Confidence: 100%]
(UserFavoritesCount, 101-200 → 601-700) ^ (UserFollowersCount, 901-1000 → 301-400) ^ (UserFriendsCount, 201-300 → 701-800) ➔ (TweetSentiment, Neutral → Positive) [Support: 2, Old Confidence: 66%, New Confidence: 100%]	(UserFavoritesCount, 101-200 → 601-700) ^ (UserFollowersCount, 901-1000 → 301-400) ^ (UserFriendsCount, 201-300 → 701-800) ➔ (TweetSentiment, Neutral → Positive) [Support: 2, Old Confidence: 100%, New Confidence: 100%]
(UserFavoritesCount, 30000-Above → 601-700) ^ (UserFollowersCount, 30000-Above → 301-400) ^ (UserFriendsCount, 201-300 → 701-800) ➔ (TweetSentiment, Neutral → Positive) [Support: 2, Old Confidence: 100%, New Confidence: 100%]	(UserFavoritesCount, 5001-10000 → 701-800) ^ (UserFollowersCount, 1001-5000 → 301-400) ^ (UserFriendsCount, 101-200 → 201-300) ^ (UserLanguage = en) ➔ (TweetSentiment, Neutral → Positive) [Support: 2, Old Confidence: 77%, New Confidence: 100%]

Figure 5.6: Example action rules - Experiment 3.

gustScore, SurpriseScore, JoyScore, PositiveScore, NegativeScore, LoveScore, People-Score, MessageScore, UserFollowersCount, UserFavoritesCount, UserFriendsCount, TweetSource, FinalEmotion from the original dataset. Discretization for the numeric attributes are shown in Table 5.5. The dataset with 174688 instances is divided into 100 parts based on the target class attribute 'FinalEmotion'. Action rules are generated for one part of the dataset with 1439 instances and 18 attributes listed above. The Table 5.4. gives list of parameters used for action rule generation.

Table 5.3: Pre-Processing Parameters

Parameter	Values
Number of Instances	174688
Method	Weka Filter Unsupervised Discretize
Number of Bins	5
Binning Method	Equal Frequency

Figure 5.7 shows sample action rules generated. Let us consider the action rule AR1, this rule suggest possible changes to achieve a desirable emotional state of 'joy'.

S.NO.	Action Rule
AR1	(AnticipationScore, 2 → 0) ^ (DisgustScore, 1 → 0) ^ (JoyScore, 0 → 2) ^ (PositiveScore, 0 → 1) ^ (SadnessScore, 2 → 0) ⇒ (FinalEmotion, sadness → joy) [Support: 21, Old Confidence: 100%, New Confidence: 100%]
AR2	(AngerScore, 2 → 0) ^ (AnticipationScore, 2 → 0) ^ (JoyScore, 0 → 2) ^ (SadnessScore, 2 → 0) ^ (TrustScore, 2 → 0) ⇒ (FinalEmotion, sadness → joy) [Support: 23, Old Confidence: 75%, New Confidence: 100%]
AR3	(AngerScore, 4 → 0) ^ (AnticipationScore, 2 → 0) ^ (DisgustScore, 4 → 0) ^ (FearScore, 4 → 0) ^ (JoyScore, 2 → 0) ^ (SadnessScore, 4 → 0) ^ (SurpriseScore, 2 → 0) ⇒ (FinalEmotion, sadness → trust) [Support: 30, Old Confidence: 100%, New Confidence: 100%]
AR4	(AngerScore, 2 → 0) ^ (AnticipationScore, 2 → 0) ^ (DisgustScore, 2 → 0) ^ (FearScore, 3 → 0) ^ (JoyScore, 2 → 0) ^ (SadnessScore, 3 → 0) ^ (SurpriseScore, 2 → 0) ⇒ (FinalEmotion, sadness → trust) [Support: 33, Old Confidence: 97%, New Confidence: 97%]
AR5	(AngerScore, 3 -> 0) ^ (AnticipationScore, 2 -> 0) ^ (DisgustScore, 3 -> 0) ^ (FearScore, 4 -> 0) ^ (JoyScore, 3 -> 0) ^ (SadnessScore, 3 -> 0) ^ (SurpriseScore, 3 -> 0) ⇒ (FinalEmotion, fear -> trust) [Support: 33, Old Confidence: 97%, New Confidence: 97%]
AR6	(AnticipationScore, 2 -> 0) ^ (FearScore, 4 -> 0) ^ (JoyScore, 3 -> 0) ^ (NegativeScore, 3 -> 0) ^ (SurpriseScore, 3 -> 0) ⇒ (FinalEmotion, fear -> trust) [Support: 31, Old Confidence: 100%, New Confidence: 100%]

Figure 5.7: Emotion Mining - Sample Action Rules.

The action rule is interpreted as follows: If the user tends to use more positive words as denoted by (JoyScore, 0 →2) and (PositiveScore, 0 →1), and reduce the words related to negative emotions like disgust, sadness and anticipation as denoted by (DisgustScore, 1→0) and (SadnessScore, 2→0) and (AnticipationScore, 2→0), then it is possible to change the emotion from 'sadness' to 'joy'. In that case, the emotion associated with this user tweet can be classified as 'joy', and we expect that the user is feeling more positive.

Table 5.4: Action Rule Parameters

Parameter	Values
Stable Attributes	LoveScore, PeopleScore, MessageScore
Decision Attribute	FinalEmotion
Support	20
Confidence	30

Table 5.5: Discretization Parameters

Attribute	Bins	Value Set
AngerScore	-infinity, 0.002068, 0.997299, 1.007317, 2.0893, infinity	0,1,2,3,4
TrustScore	-infinity, 0.011484, 0.935696, 1.01071, 2.01071, infinity	0,1,2,3,4
FearScore	-infinity, 0.003022, 0.990587, 1.003746, 2.062638, infinity	0,1,2,3,4
SadnessScore	-infinity, 0.004326, 0.973808, 1.003069, 2.003069, infinity	0,1,2,3,4
AnticipationScore	-infinity, 0.324121, 0.992358, 1.006851, 2.005516, infinity	0,1,2,3,4
DisgustScore	-infinity, 0.000009, 0.997325, 1.000536, 2.000852, infinity	0,1,2,3,4
SurpriseScore	-infinity, 0.000056, 0.999872, 1.001413, 2.005456, infinity	0,1,2,3,4
JoyScore	-infinity, 0.001784, 0.999909, 1.005155, 2.005155, infinity	0,1,2,3,4
PositiveScore	-infinity, 0.5, 1.5, 2.5, 3.5,infinity	0,1,2,3,4
NegativeScore	-infinity, 0.5, 1.5, 2.5, 3.5,infinity	0,1,2,3,4
UserFollowersCount	-infinity, 105.5, 307.5, 656.5, 1662.5,infinity	0,1,2,3,4
UserFavoritesCount	-infinity, 575.5, 2570.5, 7123.5, 19418.5,infinity	0,1,2,3,4
UserFriendsCount	-infinity,146.5, 310.5, 574.5, 1253.5, infinity	0,1,2,3,4

5.1.2.2 Hybrid Action Rule Method

To experiment our proposed novel method of Hybrid Actionable Pattern Mining, we use the Twitter dataset [227] which is densely populated with values. The Twitter dataset consists of records describing the emotions in the tweet like the intensity of emotions in a tweet text and other tweet attributes. We choose Emotion as the decision attribute and collect Action Rules that help identify changes that are required for the emotion to be more positive. For instance, to change the emotions from "Sadness" to "Joy". Table.5.6 gives an overview of the dataset used for the experiments. The minimum number of Partitions minPartitions in Spark scala is given as 30 for this experiment.

Table 5.6: Properties of Dataset Used for Experiments.

Property	Values
Attributes	25 attributes
Decision Attribute Values	Emotions: Joy, Sadness, Anger,Anticipation, Trust, Disgust, Surprise, Fear
# of Instances	174890

The decision problem here is to suggest possible recommendations to the user on how to be more positive in terms of Twitter users. Some promising applications in this context include but not limited to the following: *Education*, to benefit students, institution and faculty in terms of Teaching models, learning environment and how to improve them based on student evaluations, *Customer Care Service* based on emotions from customer feedback, these actionable patterns can suggest what aspects of the service could be improved or changed for better customer satisfaction. Table.5.7 gives information about the stable, and decision attributes, Number of atributes and instances used for each of the experiments to extract Action Rules using the algorithm Fig.4.11.

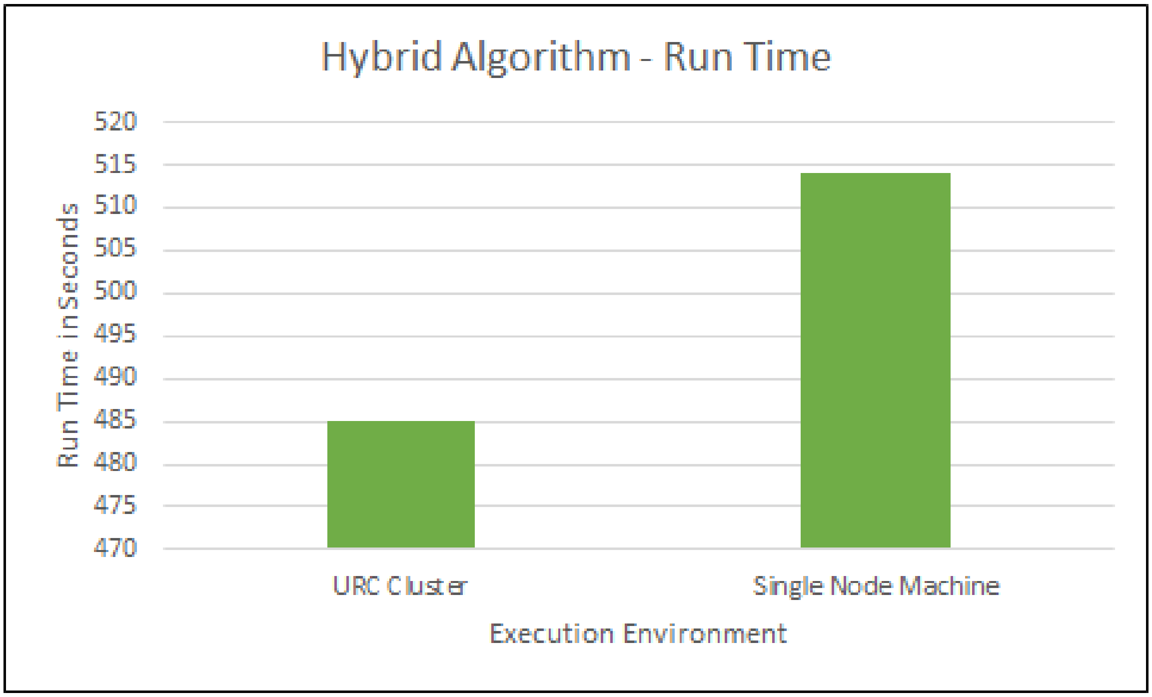

Figure 5.8: Hybrid Action Rule Algorithm - Run Time.

Table 5.7: Parameters Used for Action Rule Experiments.

Property	Experiment
# of Attributes	9 attributes
Stable Attributes	user language
Decision Attribute Values	Sadness → Joy
# of Instances	174888

We use the University Research Cluster - Taipan (Hadoop) for the experiments, which includes the following services: HBase, Hive, Hue, Impala, Kudu, Oozie, Spark and Spark2, Sqoop 2, and YARN. It has 16 nodes with dual Intel 2.93 GHz 6-core processors. Fig.5.8 gives run time analysis of the new hybrid Association Action Rule generation algoritrhm. The existing algorithm [228] does not span for the Twitter dataset and unable to complete extracting Action Rules. It fails due to the iterative overhead when run in the similar environment using the same parameters.

1. AR_{C1} : $(FearScore, Medium \rightarrow VeryLow) \wedge (SadnessScore, Medium \rightarrow VeryLow) \wedge (MediaEntities, 0) \Longrightarrow (Emotion, Sadness \rightarrow Joy)[Support : 526, Confidence : 96\%]$

2. AR_{C2} : $(AngerScore, VeryLow) \wedge (SadnessScore, Medium \rightarrow VeryLow) \wedge (UserLanguage, en) \Longrightarrow (Emotion, Sadness \rightarrow Joy)[Support : 553, Confidence : 96\%]$

3. AR_{C3} : $(SadnessScore, Medium \rightarrow VeryLow) \wedge (TweetSource, 1) \Longrightarrow (Emotion, Sadness \rightarrow Joy)[Support : 520, Confidence : 94\%]$

Let us consider the action rule AR_{C3} , this rule suggest possible changes to achieve a desirable emotional state of 'joy'. If user tends to reduce use of negative words as denoted by $(SadnessScore, Medium \rightarrow VeryLow)$ and if the TweetSource is iPhone then it is possible to change the emotion from 'sadness' to 'joy'. In that case, the emotion associated with this user tweet can be classified as 'joy'. This example is more intutive in case of applications where it is required to monitor people emotions in a particular city or county in order to understand the life satisfaction and help in public policy making and societal well-being measures.

5.1.3 Automatic Emotion Classification - Supervised Learning

5.1.3.1 Decision Tree

The decision tree classifier is built in both WEKA Data Mining Sofware [229] and Apache Spark [230] for comparison and scalability purpose. We perform several experiments with the feature set and select the following features for decision tree classification AngerScore, TrustScore, FearScore, SadnessScore, AnticipationScore, DisgustScore, SurpriseScore, JoyScore, PositiveScore, NegativeScore, LoveScore, PeopleScore, MessageScore, InstantScore, GetScore, KnowScore, GoingScore, Source, UserFollowers, UserFavorite, UserFriends, UserLanguage, isPossiblySensitive, MediaEntities. The dataset is split into train and test dataset with the ratio of 60 and 40 respectively. The decision tree model is trained with the train set, the model's accuracy is validated by using the test dataset.

5.1.3.1.1 WEKA

We build a decision tree classifier J48 model in WEKA Data Mining Sofware [229] for our Twitter emotion dataset. We achieve accuracy of 99.6% with WEKA's decision tree. The confusion matrix and evaluation measures are shown is shown in Table 5.8 and Table 5.9.

Table 5.8: Weka Decision tree - Confusion Matrix.

A	B	C	D	E	F	G	H	Class
15818	0	2	1	4	1	0	6	A-Sadness
7	10328	2	24	6	7	5	9	B-Joy
9	10	3050	3	4	2	6	6	C-Fear
1	1	0	20201	5	2	2	0	D-Anticipation
3	4	2	7	9130	5	1	3	E-Trust
3	9	8	2	3	2267	1	2	F-Surprise
7	6	14	5	5	5	4082	12	G-Anger
5	11	2	2	1	5	8	4734	H-Disgust

Table 5.9: Weka Decision Tree - Precision,Recall,F-Measure.

Measure	Sadness	Joy	Fear	Anicipation	Trust	Surprise	Anger	Disgust
Precision	0.998	0.996	0.990	0.998	0.997	0.988	0.994	0.992
Recall	0.999	0.994	0.987	0.999	0.997	0.988	0.987	0.993
F-Measure	0.998	0.995	0.989	0.999	0.997	0.988	0.991	0.992

5.1.3.1.2 Spark

In order to build the decision tree with Spark we use the Machine Learning Library MLLib - 'DecisionTreeClassifier' to train the model. We use scala programming language. We test with both Spark cluster single node instance, and Spark cluster with 6 nodes. The Spark cluster is installed over Hadoop YARN, and the 6 nodes are connected via 10 GigaBits per second Ehternet network. Visualization of the decision tree is shown is Fig. 5.9. and Fig. 5.10. With this model we achieve accuracy of 88.45% for emotion classification of Twitter dataset for both single node and 6 node cluster configuration . Table 5.10. shows the confusion matrix and the evaluation measures are shown in Table 5.11. The average execution time results for Spark single node and 6 nodes are shown in Table 5.12.

Table 5.10: Spark Decision tree - Confusion Matrix.

A	B	C	D	E	F	G	H	Class
20117	18	0	226	0	38	0	0	A-Anticipation
0	14930	23	77	0	101	436	0	B-Sadness
354	433	9326	271	12	25	13	0	C-Joy
9	29	172	8696	159	111	0	0	D-Trust
51	327	15	41	4106	153	38	0	E-Disgust
22	19	32	16	58	4058	0	0	F-Anger
204	94	36	1892	90	175	632	0	G-Fear
204	181	41	1784	41	38	15	0	H-Surprise

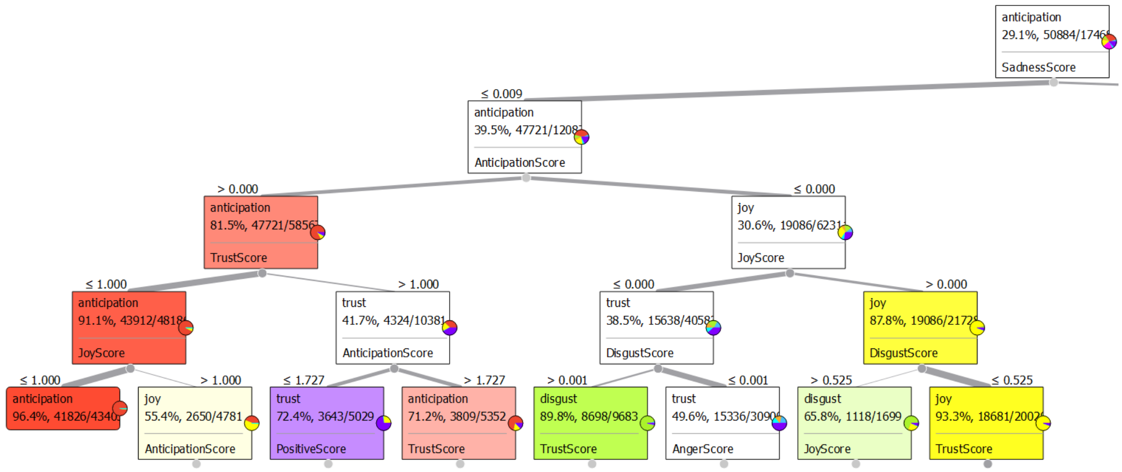

Figure 5.9: Decision Tree Left Side - Class Emotion - Twitter Dataset.

Table 5.11: Spark Decision tree - Precision,Recall,F-Measure.

Measure	Anticipation	Sadness	Joy	Trust	Disgust	Anger	Fear	Surprise
Precision	0.9597	0.9313	0.9669	0.6687	0.9193	0.8635	0.5573	0
Recall	0.9861	0.9590	0.8938	0.9476	0.8678	0.9650	0.2023	0
F-Measure	0.9727	0.9449	0.9289	0.7841	0.8928	0.9115	0.2969	0

Table 5.12: Decision Tree Execution Time in Seconds - Spark Single Node, Spark 6 Nodes.

Number of Instances	Spark Single Node (Secs)	Spark 6 Nodes (Secs)
174689	59.33	46.57

5.1.3.2 Decision Forest - Random Forest

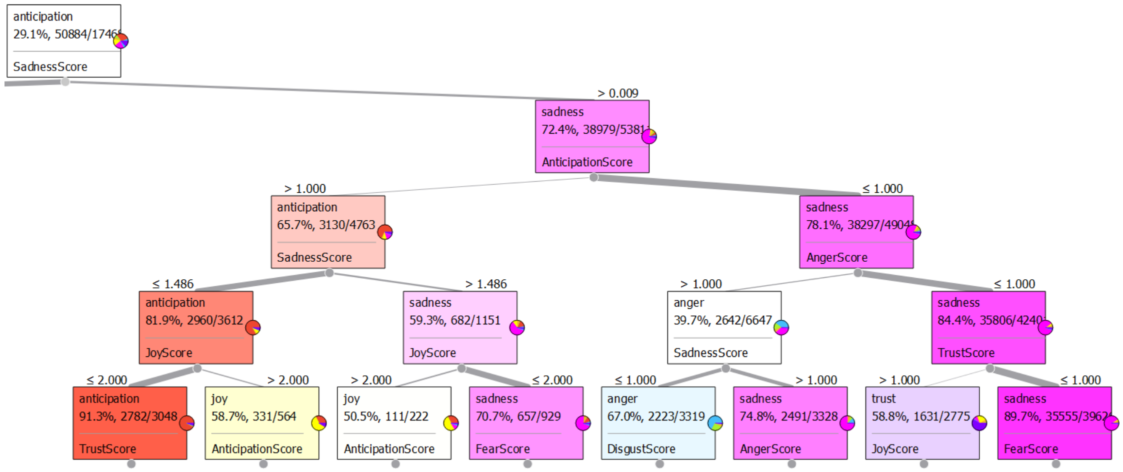

Figure 5.10: Decision Tree Right Side - Class Emotion - Twitter Dataset.

5.1.3.2.1 WEKA

We build a decision forest - random forest classifier model in WEKA Data Mining Sofware [229] for our Twitter emotion dataset. We achieve accuracy of 88.8% with WEKA's DecisionForest. The confusion matrix is shown in Table 5.13. The evaluation results with precision, recall and F-measure is given in Table 5.14.

A visualization of the decision forest - random forest - is the pythagorean forest, as shown on Fig.5.11.

Table 5.13: Weka Decision Forest - Confusion Matrix.

A	B	C	D	E	F	G	H	Class
15650	1	0	0	81	0	100	0	A-Sadness
34	9686	0	475	155	0	28	10	B-Joy
735	52	0	190	1860	0	173	80	C-Fear
19	0	0	20146	18	0	29	0	D-Anticipation
33	163	0	274	8452	0	88	145	E-Trust
203	55	0	193	1797	0	19	28	F-Surprise
20	22	0	19	20	0	3987	68	G-Anger
352	34	0	56	51	0	114	4161	H-Disgust

In Table 5.11 and Table 5.14, we see that precision and recall of ‘fear’ and ‘surprise’ emotion are lowest compared to ‘anticipation’ and ‘sadness’. We infer that the number of instances of training data for emotion ‘fear’ and ‘surprise’ is low, compared to the rest of the class labels, so the training model does not capture many correlations in the features. Fig.5.12. shows a tree map based on the number of instances in each emotion class. Therefore increasing the number of instances in the training set would improve the classifier accuracy.

Table 5.14: Precision,Recall,F-Measure - Weka Decision Forest tree.

Measure	Anticipation	Sadness	Joy	Trust	Surprise	Disgust	Anger	Fear
Precision	0.943	0.918	0.967	0.68	0.0	0.926	0.879	0.0
Recall	0.997	0.989	0.932	0.923	0.0	0.873	0.964	0.0
F-Measure	0.969	0.952	0.95	0.783	0.0	0.899	0.919	0.0

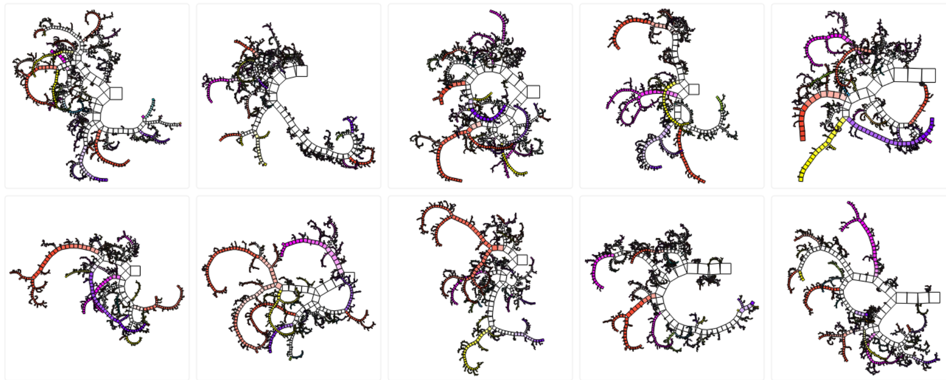

Figure 5.11: Decision Forest - Pythagorean - Class Emotion - Twitter Dataset.

5.1.3.3 Decision Table Majority

We implement the decision table majority as a rule-based classification method [231] for comparison purpose. We analyze several rule-based classification methods with our Twitter dataset for their accuracy, running time, and feasibility of implementation on a cloud clustered environment including: ZeroR, OneR [232], Decision Table [231]. The Accuracy results are shown in Table 5.15. Based on the results, we see the decision table majority classifier produces the best accuracy for our Twitter emotion dataset.

Table 5.15: Rule Based Classifier - Analysis.

Algorithm	Accuracy	Running Time (Seconds)
ZeroR	28.92	0.28
OneR	49.76	0.89
Decision Table	96.45	212.78

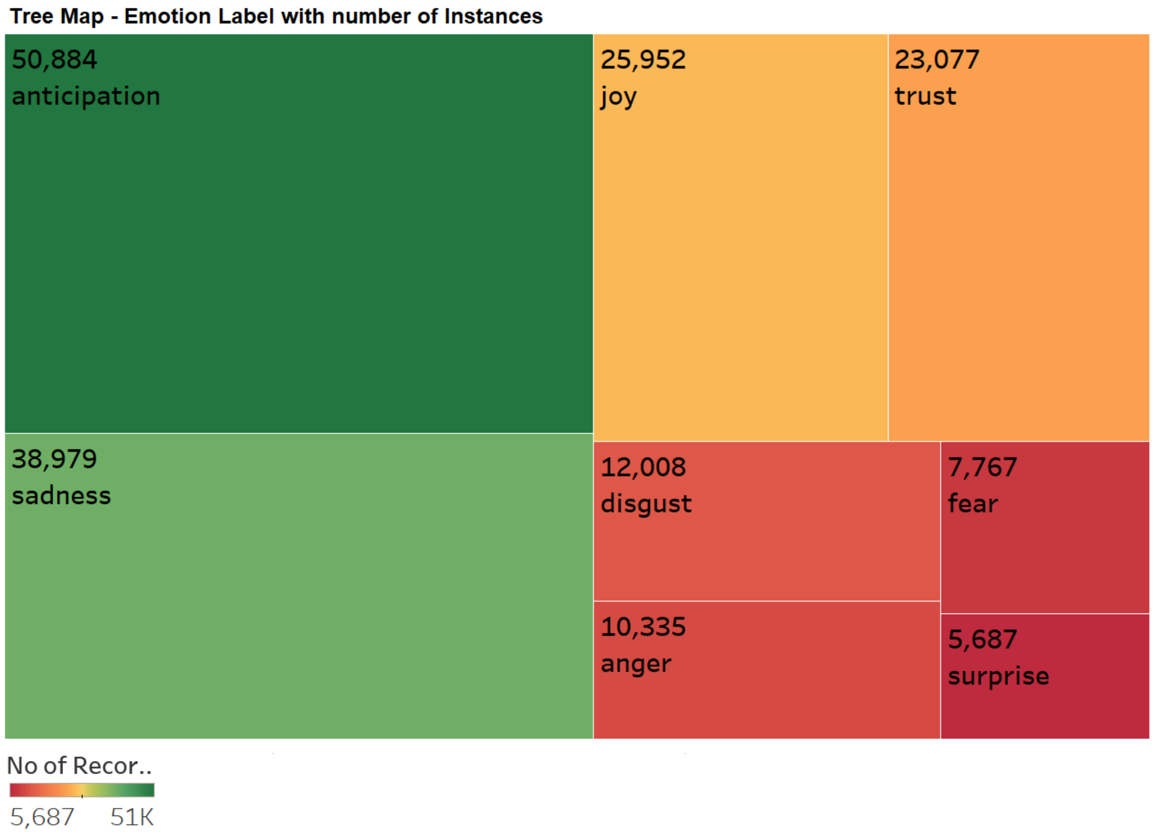

Figure 5.12: Tree Map - Corpus - Emotion Class Label Distribution.

5.1.3.3.1 WEKA

We build a decision table majority classifier model in WEKA Data Mining Software [229] for our Twitter emotion dataset. We achieve accuracy of 96.45% with WEKA's decision table majority. The confusion matrix is shown in Table 5.16. The evaluation measures are shown in Table 5.17

Table 5.16: Weka Decision Table Majority - Confusion Matrix.

A	B	C	D	E	F	G	H	Class
15498	0	9	302	0	0	17	6	A-Sadness
174	9861	0	332	0	8	0	13	B-Joy
134	0	2797	156	0	0	3	0	C-Fear
217	52	0	19942	1	0	0	0	D-Anticipation
118	8	0	221	8808	0	0	0	E-Trust
38	6	0	88	2	2161	0	0	F-Surprise
128	0	2	138	0	2	3861	5	G-Anger
110	0	0	180	0	4	5	4469	H-Disgust

Table 5.17: Weka Precision,Recall,F-Measure - Decision Table Majority.

Measure	Sadness	Joy	Fear	Anicipation	Trust	Surprise	Anger	Disgust
Precision	0.944	0.993	0.996	0.934	1	0.994	0.994	0.995
Recall	0.979	0.949	0.905	0.987	0.962	0.942	0.934	0.937
F-Measure	0.961	0.971	0.948	0.959	0.981	0.967	0.963	0.965

5.1.3.3.2 Spark

The schema of decision table is the features in the data which contribute to maximum accuracy. We use filter based feature selection algorithm in WEKA Data Mining software [229]. Some of the algorithms to extract the features for decision table Significance are: attribute evaluator, chi-squared attribute evaluator, Gain ratio attribute evaluator, greedy stepwise attribute evaluator, and filter attribute evaluator. Among the listed algorithms, gain ratio attribute evaluator is most appropriate for the given

(AngerScore=>0 AND TrustScore=>3 AND FearScore=>0 AND SadnessScore=>0 AND AnticipationScore=>1 AND DisgustScore=>0 AND SurpriseScore=>0 AND JoyScore=>4 AND PositiveScore=>4 AND NegativeScore=>0 AND LOVE_SCORE =>0→ **FinalEmotion=joy**)

Figure 5.13: Decision Table - Sample Rule.

dataset. We use the top 11 features from the entire Gain Ratio list. This selection is based on the accuracy yielded by using the selected features on the decision table majority algorithm.

The list of selected features for decision table majority algorithm using gain ratio feature selection algotithm is: AngerScore, TrustScore, FearScore, SadnessScore, AnticipationScore, DisgustScore, SurpriseScore, JoyScore, PositiveScore, NegativeScore,LoveScore and FinalEmotion.

In order to build the decision table majority classifier with Spark, we design the schema based on the above features. Train data with class labels are loaded as decision table based on the schema. First the decision table is loaded. Then the decision table induction matches the test data as per the decision table schema. A sample rule produced by the decision table is shown on in Fig.5.13. If matching records are identified then the algorithm returns the class with largest number of matching instances. Otherwise the algorithm returns the default class, which is usually the class with highest number of records in the Decision Table schema. According to Fig.5.12 we observe that the default class in our data is ‘anticipation’.

This method produces classification accuracy of 93.28% for our Twitter emotion Dataset. The confusion matrix is shown in Table 5.18, and the Table 5.19 shows the evaluation measures of precision, recall and F-measure for each of the emotion class labels.

Table 5.18: Spark Decision Table Majority - Confusion Matrix.

A	B	C	D	E	F	G	H	Class
14591	0	0	1120	0	0	0	0	A-Sadness
0	9160	0	963	0	0	0	0	B-Joy
0	0	2614	572	0	0	0	0	C-Fear
0	0	0	20090	0	0	0	0	D-Anticipation
0	0	694	8581	0	0	0	0	E-Trust
0	0	0	221	0	1980	0	0	F-Surprise
0	0	0	586	0	0	3783	0	G-Anger
0	0	0	536	0	0	0	4385	H-Disgust

Table 5.19: Spark Precision,Recall,F1-Score - Decision Table Majority.

Measure	Sadness	Joy	Fear	Anicipation	Trust	Surprise	Anger	Disgust
Precision	1	1	1	0.8106	1	1	1	1
Recall	0.9287	0.9048	0.8204	1	0.9251	0.8995	0.8656	0.8910
F-Measure	0.9630	0.95	0.9013	0.8954	0.9611	0.9471	0.9281	0.9424

Decision table majority is implemented in Apache Spark [230] using Scala programming language. We test in a single node cluster, and 6 nodes cluster configuration. Results show that the execution time is faster in 6 node cluster when compared to a single node. The average execution times are shown in Table 5.20.

Table 5.20: Decision Table Majority - Average Execution times in Seconds - WEKA, Spark Single Node, Spark 6 Node Cluster.

Number of Instances	Spark Single Node (Secs)	Spark 6 Nodes (Secs)
174689	62.42	37.39

5.1.3.4 Support Vector Machine

We use WEKA Data Mining Software [229] and Apache Spark [230] to develop the Support Vector Machine One Vs All Multi class classifier. Support Vector Machine classification model requires pre-processing of data which includes: normalization, categorical to numeric or binary, LIBSVM format. Based on the pre-processing three experiments are performed: Using only the numerical attributes in the data, Using all the attributes where the categorical fields are encoded as numeric values, Using all attributes where the categorical fields are encoded as binary.

5.1.3.4.1 WEKA

Experiment 1 - Using only Numeric attributes Experiment 1 is performed by selecting only the numerical attributes from the original dataset which includes: AngerScore, TrustScore, FearScore, SadnessScore, AnticipationScore, DisgustScore, SurpriseScore,JoyScore, PositiveScore, NegativeScore, LoveScore, PeopleScore, MessageScore, InstantScore,GetScore, KnowScore, GoingScore, UserFollowersCount, UserFavoritesCount, UserFriendsCount. We achieve accuracy of 84.92% with WEKA Data Mining software Multiclass Classifier with SVM Stochastic Gradient Descent(SGD). The confusion matrix and evaluation is shown in Table. 5.21 and Table. 5.22 respectively.

Table 5.21: WEKA - Confusion Matrix - Experiment 1.

A	B	C	D	E	F	G	H	Class
10133	0	0	1481	0	0	0	0	A - Sadness
144	5535	1	2080	0	0	57	0	B - Joy
152	6	1477	716	17	0	2	0	C - Fear
104	10	33	15150	3	0	4	0	D - Anticipation
127	42	22	339	6342	0	1	0	E - Trust
50	18	10	293	213	1124	0	1	F - Surprise
165	6	85	179	61	3	2600	0	G - Anger
112	2	27	1039	149	2	143	2146	H - Disgust

Table 5.22: WEKA - Precision, Recall, FMeasure - Experiment 1.

Measure	Sadness	Joy	Fear	Anticipation	Trust	Surprise	Anger	Disgust
Precision	0.922	0.985	0.892	0.712	0.935	0.996	0.926	1.000
Recall	0.872	0.708	0.623	0.990	0.923	0.658	0.839	0.593
F-Measure	0.897	0.824	0.734	0.828	0.929	0.792	0.880	0.744

Experiment 2 - Categorical Attributes Encoded as Numeric Experiment 2 is performed by selecting the numerical attributes and categorical attributes from the original dataset where the categorical attributes are encoded as numeric. The following are the numerical attributes: AngerScore, TrustScore, FearScore, SadnessScore, AnticipationScore, DisgustScore, SurpriseScore, JoyScore, PositiveScore, NegativeScore, LoveScore, PeopleScore,MessageScore, InstantScore, GetScore, KnowScore, GoingScore,UserFollowersCount,UserFavoritesCount,UserFriendsCount. UserLanguage, IsPossiblysensitive, MediaEntities, TweetSource are the categorical attributes which are encoded as numeric.We achieve accuracy of 84.94% with WEKA Data Mining software Multiclass Classifier with SVM Stochastic Gradient Descent(SGD).

Table 5.23: WEKA - Confusion Matrix - Experiment 2.

A	B	C	D	E	F	G	H	Class
10684	0	0	930	0	0	0	0	A - Sadness
189	7276	1	350	0	0	1	0	B - Joy
209	15	2005	139	0	0	2	0	C - Fear
171	22	427	14670	3	0	11	0	D - Anticipation
192	211	75	460	5933	0	2	0	E - Trust
67	28	121	239	155	1099	0	0	F - Surprise
232	12	180	162	28	10	2475	0	G - Anger
164	143	173	2584	17	8	156	375	H - Disgust

Table 5.24: WEKA - Precision, Recall, FMeasure - Experiment 2.

Measure	Sadness	Joy	Fear	Anticipation	Trust	Surprise	Anger	Disgust
Precision	0.897	0.944	0.672	0.751	0.967	0.984	0.935	1.000
Recall	0.920	0.931	0.846	0.959	0.863	0.643	0.799	0.104
F-Measure	0.908	0.937	0.749	0.842	0.912	0.778	0.861	0.188

Experiment 3 - Categorical Attributes Encoded as Binary Experiment 3 is performed by selecting the numerical attributes and categorical attributes from the original dataset where the categorical attributes are encoded as numeric. The following are the numerical attributes: AngerScore, TrustScore, FearScore, SadnessScore, AnticipationScore, DisgustScore, SurpriseScore, JoyScore, PositiveScore, NegativeScore, LoveScore, PeopleScore,MessageScore, InstantScore, GetScore, KnowScore, GoingScore,UserFollowersCount,UserFavoritesCount,UserFriendsCount. UserLanguage, IsPossiblysensitive, MediaEntities, TweetSource are the categorical attributes which are encoded as binary.

Table 5.25: WEKA - Confusion Matrix - Experiment 3.

A	B	C	D	E	F	G	H	Class
10969	0	0	644	0	0	1	0	A - Sadness
216	5452	17	2071	0	0	61	0	B - Joy
286	6	1895	177	0	0	6	0	C - Fear
235	9	79	14884	1	0	96	0	D - Anticipation
259	44	63	624	5879	0	3	1	E - Trust
92	12	80	325	50	1145	5	0	F - Surprise
274	5	108	168	25	7	2512	0	G - Anger
210	4	133	832	18	10	267	2146	H - Disgust

Table 5.26: WEKA - Precision, Recall, FMeasure - Experiment 3.

Measure	Sadness	Joy	Fear	Anticipation	Trust	Surprise	Anger	Disgust
Precision	0.875	0.986	0.798	0.755	0.984	0.985	0.851	1.000
Recall	0.944	0.697	0.800	0.973	0.855	0.670	0.811	0.593
F-Measure	0.908	0.817	0.799	0.850	0.915	0.798	0.830	0.744

5.1.3.4.2 Spark

Experiment 1 - Using only Numeric attributes Experiment 1 is performed by selecting only the numerical attributes from the original dataset which includes: AngerScore, TrustScore, FearScore, SadnessScore, AnticipationScore, DisgustScore, SurpriseScore, JoyScore, PositiveScore, NegativeScore, LoveScore, PeopleScore, MessageScore, InstantScore, GetScore, KnowScore, GoingScore, UserFollowersCount, UserFavoritesCount, UserFriendsCount along with the class label FinalEmotion.

We achieved almost similar accuracy with Spark single node and cluster as 88.16% and 88.01% respectively. The confusion matrix and classifier evaluation with precision, recall, and F1-score is shown in Table.5.27, Table. 5.29, Table.5.28, Table.5.30 respectively. But the Spark program runs faster in cluster when compared to Single Node machine. The results of average run time for execution is shown in Table.5.31.

Table 5.27: Spark Single Node - Confusion Matrix - Experiment 1.

A	B	C	D	E	F	G	H	Class
15246.0	22.0	92.0	0.0	9.0	3.0	0.0	0.0	A - Anticipation
13.0	11506.0	3.0	3.0	0.0	0.0	0.0	0.0	B - Sadness
688.0	222.0	6630.0	172.0	7.0	76.0	5.0	0.0	C - Joy
443.0	161.0	20.0	6277.0	12.0	3.0	0.0	0.0	D - Trust
47.0	432.0	122.0	34.0	2896.0	0.0	0.0	0.0	E - Disgust
102.0	570.0	5.0	51.0	8.0	2400.0	0.0	0.0	F - Anger
126.0	604.0	37.0	289.0	12.0	17.0	1176.0	0.0	G - Fear
843.0	304.0	234.0	243.0	41.0	19.0	100.0	0.0	H - Surprise

Table 5.28: Spark Single Node - Precision, Recall, F1 Score - Experiment 1

Measure	Anticipation	Sadness	Joy	Trust	Disgust	Anger	Fear	Surprise
Precision	0.8708	0.8325	0.9281	0.8879	0.9701	0.9531	0.9180	0.0
Recall	0.9918	0.9983	0.85	0.9076	0.8201	0.7653	0.5201	0.0
F1-Score	0.9273	0.9079	0.8873	0.8976	0.8888	0.8489	0.6640	0.0

Table 5.29: Spark Cluster - Confusion Matrix - Experiment 1.

A	B	C	D	E	F	G	H	Class
15199.0	24.0	83.0	0.0	12.0	3.0	0.0	0.0	A - Anticipation
13.0	11675.0	2.0	3.0	0.0	0.0	0.0	0.0	B - Sadness
657.0	234.0	6633.0	137.0	9.0	94.0	4.0	0.0	C - Joy
409.0	172.0	25.0	6262.0	12.0	3.0	1.0	0.0	D - Trust
41.0	436.0	120.0	25.0	2898.0	0.0	0.0	0.0	E - Disgust
79.0	580.0	3.0	48.0	8.0	2401.0	0.0	0.0	F - Anger
140.0	630.0	127.0	301.0	24.0	75.0	984.0	0.0	G - Fear
542.0	293.0	503.0	250.0	45.0	20.0	86.0	0.0	H - Surprise

Table 5.30: Spark Cluster - Precision, Recall, F1 Score - Experiment 1

Measure	Anticipation	Sadness	Joy	Trust	Disgust	Anger	Fear	Surprise
Precision	0.8898	0.8313	0.8848	0.8912	0.9634	0.9248	0.9153	0.0
Recall	0.9920	0.9984	0.8538	0.9096	0.8232	0.7697	0.4313	0.0
F1-Score	0.9381	0.9072	0.8691	0.9003	0.8878	0.8402	0.5864	0.0

Table 5.31: Experiment 1 - Average Run Time - Spark Single Node and Spark cluster.

Number of Instances	Spark Single Node Runtime (secs)	Spark 6 node Cluster Runtime (secs)
174688	256.75	207.70

Experiment 2 - Categorical Attributes Encoded as Numeric Experiment 2 is performed by selecting the numerical attributes and categorical attributes from the original dataset where the categorical attributes are encoded as numeric. The following are the numerical attributes: AngerScore, TrustScore, FearScore, SadnessScore, AnticipationScore, DisgustScore, SurpriseScore, JoyScore, PositiveScore, NegativeScore, LoveScore, PeopleScore,MessageScore, InstantScore, GetScore, KnowScore, GoingScore,UserFollowersCount,UserFavoritesCount,UserFriendsCount. UserLanguage, IsPossiblysensitive, MediaEntities, TweetSource are the categorical attributes which are encoded as numeric.

Table 5.32: Experiment 2 - Spark Single Node - Confusion Matrix.

A	B	C	D	E	F	G	H	Class
15179.0	25.0	95.0	0.0	16.0	4.0	0.0	0.0	A - Anticipation
10.0	11700.0	5.0	2.0	0.0	0.0	0.0	0.0	B - Sadness
642.0	172.0	6649.0	135.0	7.0	47.0	2.0	0.0	C - Joy
332.0	169.0	23.0	6398.0	12.0	3.0	1.0	0.0	D - Trust
37.0	470.0	119.0	17.0	2937.0	0.0	0.0	0.0	E - Disgust
104.0	599.0	2.0	77.0	7.0	2304.0	0.0	0.0	F - Anger
132.0	613.0	40.0	310.0	14.0	19.0	1156.0	0.0	G - Fear
656.0	212.0	397.0	245.0	122.0	12.0	102.0	0.0	H - Surprise

Table 5.33: Spark Single Node - Precision, Recall, F1 Score - Experiment 2.

Measure	Anticipation	Sadness	Joy	Trust	Disgust	Anger	Fear	Surprise
Precision	0.8880	0.8381	0.9070	0.8905	0.9428	0.9644	0.9167	0.0
Recall	0.9908	0.9985	0.8686	0.9221	0.8203	0.7449	0.5061	0.0
F1-Score	0.9366	0.9113	0.8874	0.9061	0.8773	0.8405	0.6521	0.0

Table 5.34: Spark 6 node Cluster - Confusion Matrix - Experiment 2.

A	B	C	D	E	F	G	H	Class
15198.0	24.0	82.0	0.0	14.0	3.0	0.0	0.0	A - Anticipation
16.0	11670.0	4.0	3.0	0.0	0.0	0.0	0.0	B - Sadness
662.0	190.0	6661.0	182.0	11.0	57.0	5.0	0.0	C - Joy
397.0	166.0	24.0	6281.0	13.0	2.0	1.0	0.0	D - Trust
38.0	479.0	95.0	19.0	2889.0	0.0	0.0	0.0	E - Disgust
91.0	617.0	3.0	127.0	8.0	2273.0	0.0	0.0	F - Anger
133.0	629.0	40.0	280.0	14.0	16.0	1169.0	0.0	G - Fear
822.0	169.0	227.0	249.0	155.0	18.0	99.0	0.0	H - Surprise

Table 5.35: Spark Cluster - Precision, Recall, F1 Score - Experiment 2.

Measure	Anticipation	Sadness	Joy	Trust	Disgust	Anger	Fear	Surprise
Precision	0.8756	0.8369	0.9334	0.8795	0.9307	0.9594	0.9175	0.0
Recall	0.9919	0.9980	0.8574	0.9124	0.8207	0.7287	0.5124	0.0
F1-Score	0.9301	0.9104	0.8938	0.8956	0.8722	0.8283	0.6576	0.0

Table 5.36: Experiment 2 - Average Run Time - Spark Single Node and Spark cluster.

Number of Instances	Spark Single Node Runtime (secs)	Spark 6 node Cluster Runtime (secs)
174688	258.58	228.20

We achieved almost similar accuracy with Spark single node and 6 node Cluster as 88.51% and 88.18% respectively. The confusion matrix and classifier evaluation with precision, recall, and F1-score is shown in Table.5.32, Table. 5.34, Table.5.33, Table.5.35 respectively. But the Spark program runs faster in 6 node Cluster when compared to Single Node machine. The results of average run time for execution is shown in Table 5.36.

Experiment 3 - Categorical Attributes Encoded as Binary Experiment 3 is performed by selecting the numerical attributes and categorical attributes from the original dataset where the categorical attributes are encoded as numeric. The following are the numerical attributes: AngerScore, TrustScore, FearScore, SadnessScore, AnticipationScore, DisgustScore, SurpriseScore, JoyScore, PositiveScore, NegativeScore, LoveScore, PeopleScore,MessageScore, InstantScore, GetScore, KnowScore, GoingScore,UserFollowersCount,UserFavoritesCount,UserFriendsCount. UserLanguage, IsPossiblysensitive, MediaEntities, TweetSource are the categorical attributes which are encoded as binary.

Table 5.37: Spark Single Node - Confusion Matrix - Experiment 3.

A	B	C	D	E	F	G	H	Class
15405.0	30.0	33.0	0.0	21.0	3.0	0.0	0.0	A - Anticipation
12.0	11566.0	4.0	1.0	0.0	0.0	0.0	0.0	B - Sadness
419.0	162.0	7027.0	25.0	9.0	21.0	2.0	0.0	C - Joy
300.0	168.0	33.0	6321.0	10.0	2.0	0.0	0.0	D - Trust
30.0	499.0	57.0	8.0	3005.0	0.0	0.0	0.0	E - Disgust
26.0	625.0	3.0	43.0	6.0	2458.0	0.0	0.0	F - Anger
114.0	628.0	19.0	273.0	19.0	14.0	1203.0	0.0	G - Fear
381.0	180.0	661.0	251.0	145.0	13.0	106.0	0.0	H - Surprise

Table 5.38: Spark Single Node - Precision, Recall, F1 Score - Experiment 3.

Measure	Anticipation	Sadness	Joy	Trust	Disgust	Anger	Fear	Surprise
Precision	0.9231	0.8346	0.8966	0.9131	0.9346	0.9788	0.9176	0.0
Recall	0.9943	0.9985	0.9167	0.9249	0.8349	0.7776	0.5299	0.0
F1-Score	0.9574	0.9092	0.9065	0.9190	0.8820	0.8667	0.6718	0.0

Table 5.39: Spark 6 node Cluster - Confusion Matrix - Experiment 3.

A	B	C	D	E	F	G	H	Class
15191.0	20.0	85.0	0.0	20.0	5.0	0.0	0.0	A - Anticipation
15.0	11669.0	8.0	1.0	0.0	0.0	0.0	0.0	B - Sadness
646.0	176.0	6666.0	173.0	13.0	90.0	4.0	0.0	C - Joy
303.0	168.0	25.0	6373.0	12.0	3.0	0.0	0.0	D - Trust
23.0	493.0	45.0	3.0	2956.0	0.0	0.0	0.0	E - Disgust
19.0	627.0	2.0	36.0	7.0	2428.0	0.0	0.0	F - Anger
128.0	633.0	36.0	276.0	15.0	16.0	1177.0	0.0	G - Fear
625.0	193.0	408.0	260.0	134.0	22.0	97.0	0.0	H - Surprise

Table 5.40: Spark 6 node Cluster - Precision, Recall, F1 Score - Experiment 3.

Measure	Anticipation	Sadness	Joy	Trust	Disgust	Anger	Fear	Surprise
Precision	0.8962	0.8347	0.9162	0.8948	0.9363	0.9469	0.9209	0.0
Recall	0.9915	0.9979	0.8581	0.9257	0.8397	0.7784	0.5160	0.0
F1-Score	0.9414	0.9090	0.8862	0.9100	0.8854	0.8544	0.6614	0.0

Table 5.41: Experiment 3 - Average Run Time - Spark Single Node and Spark cluster.

Number of Instances	Spark Single Node Runtime (secs)	Spark 6 node Cluster Runtime (secs)
174688	501.28	295.55

We achieved almost similar accuracy with Spark single node and 6 node Cluster as 89.76% and 88.79% respectively. The confusion matrix and classifier evaluation with precision, recall, and F1-score is shown in Table.5.37, Table. 5.39, Table.5.38, Table.5.40 respectively. But the Spark program runs faster in 6 node Cluster when compared to Single Node machine. The results of average run time for execution is shown in Table 5.41.

5.1.3.4.3 Tweet Feature Extraction and Classification using LibLinear Support Vector Machine

In the literature wide range of features have been explored in the task of tweet sentiment analysis including unigrams, bigrams, n-grams, part-of-speech (POS) tags, word embedding, word clusters [233], [234], [235], [236], [66], [237]. In this work we use TweetToSparseFeatureVector filter in Weka Affective tweets [63] package to extract word n-grams, character n-grams, brown word clusters and part-of-speech tags. The dataset with the following attributes is passed to the Weka data mining software: AngerScore, TrustScore, FearScore, SadnessScore, AnticipationScore, DisgustScore, SurpriseScore, JoyScore, PositiveScore, NegativeScore, TweetLanguage, TweetSource, UserFollowersCount, UserFavoritesCount, UserFriendsCount, UserLanguage, MediaEntities, FinalEmotion, TweetTokens.

LibLinear is a open source package that is considered to be efficient for training large-scale problems. We use the data mining software WEKA (Waikato Environment for Knowledge Analysis) [229] for emotion classification.

Table 5.42: Weka - Confusion Matrix - Tweet Feature Extraction and Classification.

A	B	C	D	E	F	G	H	Class
15449	16	0	0	0	1	7	0	A - Sadness
2	10259	1	127	7	4	1	7	B - Joy
44	4	3057	3	6	8	14	11	C - Fear
9	12	0	20411	4	0	12	1	D - Anticipation
77	43	2	49	8990	11	10	5	E - Trust
57	16	9	6	12	2155	1	7	F - Surprise
21	5	0	0	1	0	4100	26	G - Anger
16	22	4	1	1	2	8	4741	H - Disgust

Table 5.43: Weka - Precision, Recall, F1 Score - Tweet Feature Extraction and Classification.

Measure	Sadness	Joy	Fear	Anticipation	Trust	Surprise	Anger	Disgust
Precision	0.986	0.989	0.995	0.991	0.997	0.988	0.987	0.988
Recall	0.998	0.986	0.971	0.998	0.979	0.952	0.987	0.989
F1-Score	0.992	0.987	0.983	0.995	0.987	0.970	0.987	0.988

For experiments we use the processed dataset and classify using SVM LibLinear [238] classification model. The classifier excludes the string type attribute, in this case the TweetTokens which is the text of the tweet. This is because the text is processed in the feature extraction step to get additional attributes like word n-grams, brown word clusters, and part-of-speech tags. Thus we achieve an accuracy of 98%. The confusion matrix and precision, recall, f-measure is shown in Table. 5.42., and Table. 5.43 respectively. We achieve a 10% improved accuracy compared to our previous method [228].

5.1.3.5 Recurrent Neural Network - Gated Recurrent Unit (GRU)

The Twitter dataset is used for the text classification. The dataset is processed to remove discrepancies including newlines, the final dataset used for emotion classification contains 174090 instances.The input data to the GRU model is a text file with each tweet on single line labeled with the emotion. The tweet text is converted to vector for use by the neural network model. In this exepriment we use 200 dimensional vectors to build semantic word embeddings/feature vectors and computing the top list words.

We achieve an accuracy of 29.2% for the tweet emotion classification. The following hyper parameters are used for the model - Learning Rate (0.5), maximum Words Number (1000), Learning decay (0.0002).

Table 5.44: Comparison of Methods: Automatic Emotion Classification - Supervised Learning.

Method	Accuracy
Decision Tree	84.45% - 99.6%
Decision Forest	88.8%
Decision Table Majority	93.28% - 96.45%
Support Vector Machine - Numeric Attributes	84.92% - 88.16%
Support Vector Machine - Category as Numeric	84.92% - 88.51%
Support Vector Machine - Category as Binary	85.64% - 89.76%
Support Vector Machine - LibLinear(Feature Extraction)	98%
Recurrent Neural Network - GRU	29.2%

The table. 5.44 shows the overall comparison of all the supervised learning models for automatic emotion classification using Twitter dataset. We evaluated the following methods of Decision Tree, Decision Forest, Decision Table Majority, Support Vector Machine, Recurrent Neural Network - Gated Recurrent Unit. Comparing the performance of these methods, we find that Support Vector Machine performs the best, with accuracy in the range of 85 % to 98 % for Emotion Classification with Twitter dataset.

5.2 Student Evaluation Data

5.2.1 **Experiment 1: Emotion and Polarity**

In Experiment 1, the pre-processed data is passed to the system which finds the word associated with 8 basic emotions and the polarity for each of the student feedback response. After which the scores are calculated based on the frequency of each emotion and polarity related words. The sentiment that has highest score is assigned as overall emotion/polarity. The results are shown on a temporal basis from 2013 until 2017 on the X-axis and the count of each emotion on the Y-axis in Fig 5.14. It is observed that emotion 'trust' and polarity 'positive' has a growing trend through the time. Similarly, we see that 'anticipation' was high during the year 2014 which gradually decreased in the year 2017. These changes are attributed towards active learning methodology implemented in the year 2016 and 2017.

5.2.2 **Experiment 2: Basic Emotion**

In Experiment 2, the pre-processed data is passed to the system which finds the word associated with 8 basic emotions for each of the student feedback response. After which the scores are calculated based on the frequency of each emotion related words. The sentiment that has highest score is assigned as overall emotion. The results are shown on a temporal basis from 2013 until 2017 on the X-axis and the count of each emotion on the Y-axis in Fig 5.15. The results for this experiment is almost the same as Experiment 1, without the two polarities 'positive' and 'negative'. It is observed that emotion 'trust' has a growing trend through the time. Similarly,

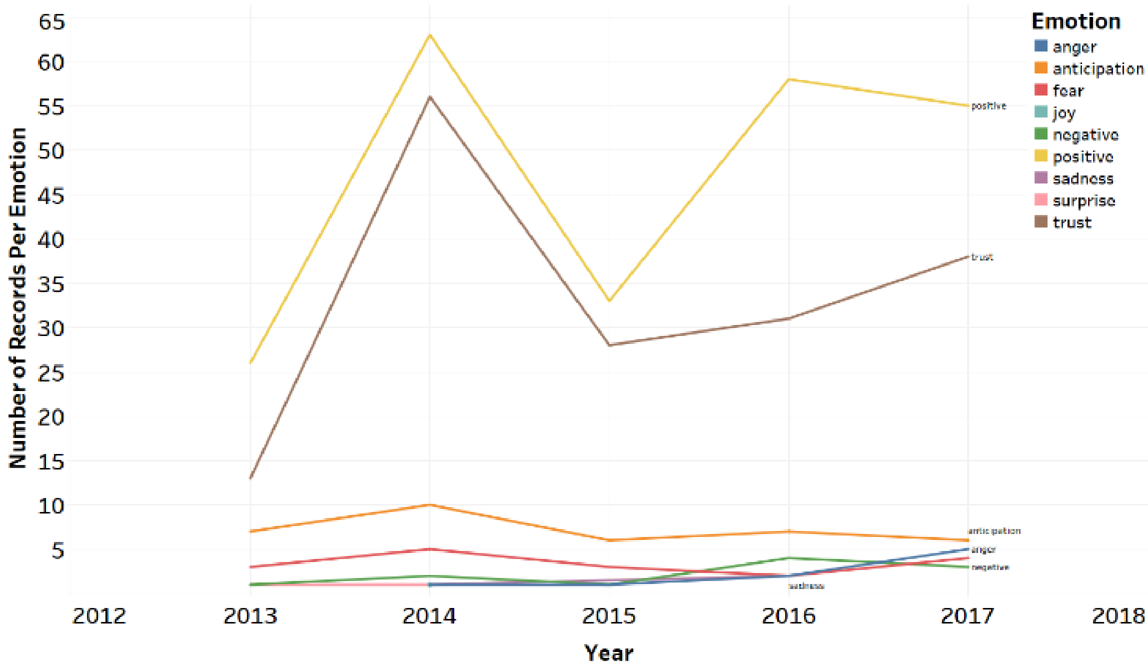

Figure 5.14: Experiment 1 Basic Emotion and Polarity.

we see that 'anticipation' was high during the year 2014 which gradually decreased in the year 2017. In Experiment we observe emotion 'joy' for the year 2016 when actually active learning methodology was started in the classes. But the count of the emotion 'joy' is low compared to others in the data.

5.2.3 Visualisation 1: Sentiment Analysis and Emotion Detection in Student Evaluations Word Cloud

Word Cloud is a text summarization, which shows the most frequently occurring words in a text, with the largest font. Word Cloud is helpful to learn about the number and kind of topics present in the text [239]. In this work we use the Word Cloud package in Python to create Word Clouds using the emotional words from the student evaluation data. During the emotion labeling step for each of the student feedback, the emotional words are recorded separately for each of the eight emotion

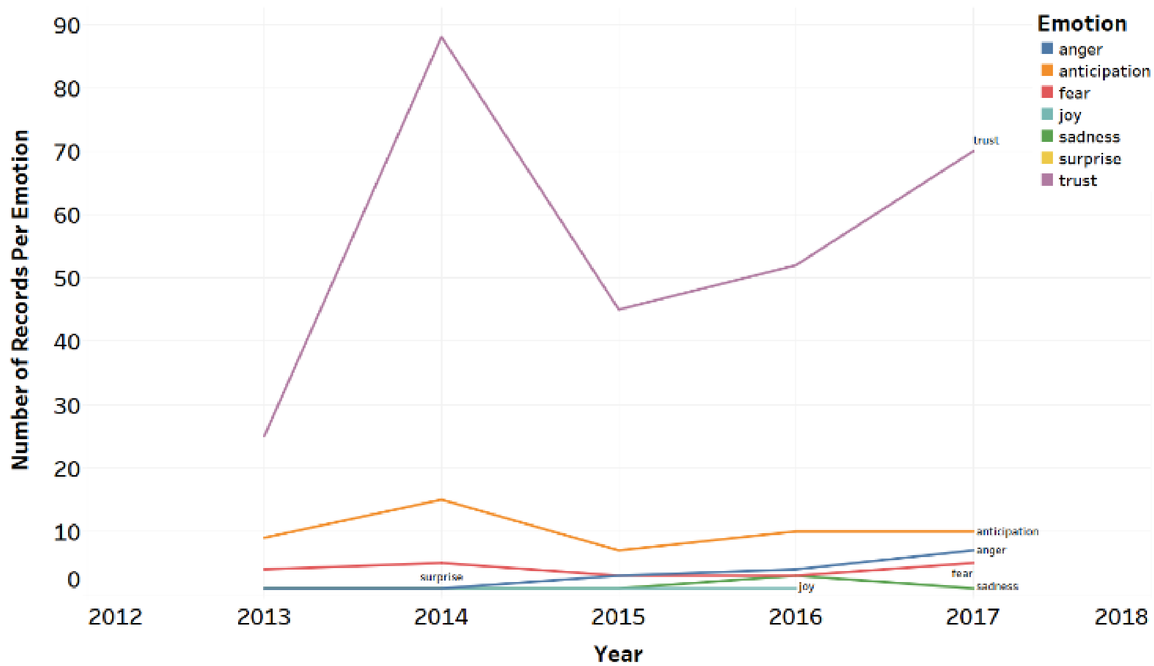

Figure 5.15: Experiment 2 Basic Emotion.

and the positive and negative polarities. To form word-cloud the list of words from the following emotions 'anger', 'fear', 'sadness', 'disgust', and 'negative' are taken as negative word list from the NRC Emotion Lexicon [79], [78]. These words appear in red color in the word cloud. The positive words are words that denote the following emotion 'joy', 'trust', 'anticipation', and 'positive' polarity appear in grey scale. The most frequently occurring positive Words are shown in green color.

We observe that the year 2014 and 2015 have more negative words including 'problem', 'waste', 'disappointed', 'awful', 'painful' and others as shown in 5.16. In 2017, more frequency of positive words like 'helpful', 'resources', 'good', 'information'. In 2017 Active Learning methods were implemented in the courses, including Light Weight Teams [87], [88], and Flipped Classroom [89]. We see that occurrences of negative emotion words like 'terrible' have decreased since 2017. Therefore, we claim that the implementation of Light Weight Teams and Flipped Classroom Active Learning methods increase positive emotions among students and improve their learning experience.

Figure 5.16: Temporal distribution of Word Cloud - Most frequest words appear with largest font. Positive words in Green and Negative words in Red.

5.2.4 Visualisation2: Multiple Emotion Label

Each student evaluation comment can contain multiple emotions, as a student can have emotions like 'trust' and 'anticipation' together in terms of course evaluation. For instance consider the following comment from the dataset *"The book which was chosen for this course is an amazing learning tool. There is a lot of very useful and necessary information covered in the textbook. This hardly applies to all classes. I hope that the instructor will continue using this book in the future"*. This shows that the student has trust that the book used for the course has good content and also anticipates that it will be used in future semesters. This kind of knowledge extracted from the student evaluation help the instructor gain better understanding the course delivery and student expectations.

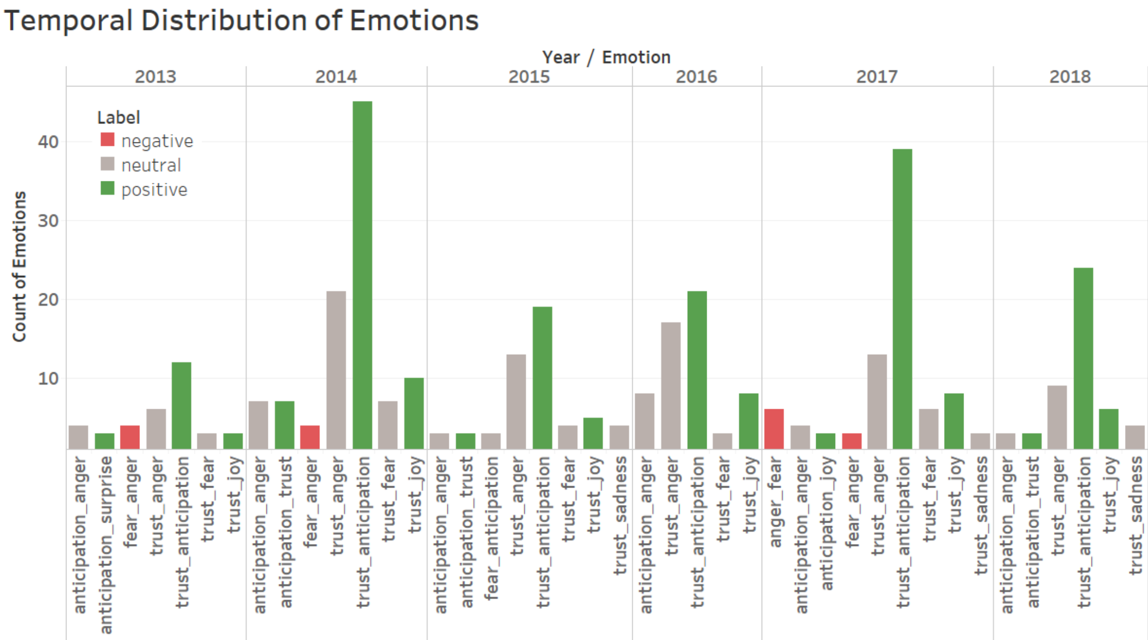

Figure 5.17: Temporal distribution Emotions.

Student comments are processed as tokens and calculate score with respect to each of the eight emotions ‘anger’, ‘fear’, ‘sadness’, ‘disgust’, ‘surprise’, ‘anticipation’, ‘trust’. After the entire comment is processed the emotion which has the highest score is assigned as the final label together with the second most frequent emotion with respect to that student comment. As part of Emotion labeling if the final emotion score is zero then those records are omitted from the dataset. This helps the instructor better understand how the emotions change over the years and what changes helped students. Fig. 5.17 shows the bar graph over the years ‘2013’ until ‘2018’ with the multi-emotions grouped into three sentiment classes of ‘positive’, ‘negative’, and ‘neutral’. The emotions ‘anticipation’, ‘trust’, ‘surprise’, ‘joy’ are marked as with a score of +1 and ‘sadness’, ‘disgust’, ‘fear’, ‘anger’ are marked with a score of -1. The final emotion class is determined to be positive if the overall score is greater than or equal to 1, negative if the overall score is less than or equal to -1 and neutral if it is 0. After the labeling the data is grouped based on the attribute year.

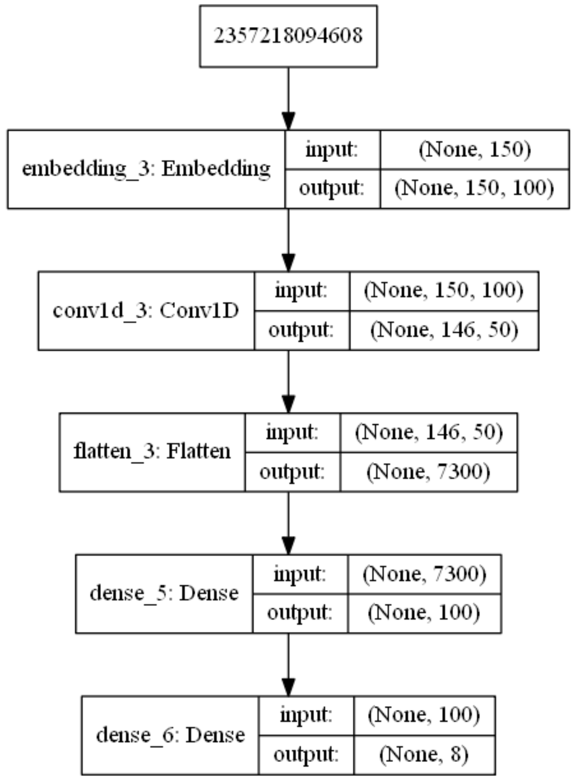

Figure 5.18: Neural Networks Model Summary.

5.2.5 Emotion Classification

We use Keras [240] a high-level neural network API in python for automatic classification of emotion from student evaluation data. The classification model is based on Keras sequential model, which is a linear stack of layers. We use the 1D convolutional kernal with dense (fully connected) layer compiled with Adaptive Moment Estimation (Adam) [240] optimizer and categorical crossentropy as loss function. Finally the model is trained using Epochs = 5 and Batch size = 2.

We use traditional Naive Bayes and Support Vector Classification methods as a baseline to compare the neural networks implementation.

5.2.5.1 Naive Bayes Classifier and Support Vector Machine Classifier

One of the popular use of text pre-processing in the traditional methods is use of TF-IDF (Term Frequency - Inverse Document Frequency) which is a popular weighting scheme used in information retrieval and text mining applications. It is a statistical measure to evaluate the importance of words in the document or corpus. TF-IDF is mainly composed of two terms: Term Frequency "(5.1)" and Inverse Document Frequency "(5.2)".

$$TF(t) = \frac{\text{Number of times term t appears in a document}}{\text{Total number of terms in the document}} \tag{5.1}$$

$$IDF(t) = \frac{\text{Total number of documents}}{\text{Number of documents with term t in it}} \tag{5.2}$$

The student evaluations dataset is processed with TF-IDF and given as input to the Naive Bayes and Support Vector classification. We achieve accuracy of approximately 74.79% with Naive Bayes and 77.97% with Support Vector Machine.

5.2.5.2 Neural Networks Classifier

In order for the text input to be understood by the neural network algorithm, it is required to process the text before passing to the classifier model to be trained. For this purpose words are replaced with unique numbers and combined with embedding vector to make it semantically meaningful. We achieve an accuracy of approximately 76.7% which is very much in close approximation with the traditional models

5.2.6 Actionable Pattern Discovery

The original dataset is replicated for scalabilty testing which include approximately 50,000 instances. We divided the dataset into two subsets or parts depending on the questions in the survey and conducted separate experiements for each of the data parts.

5.2.6.1 Student Survey Data 1 - Team Work and Student Emotion

This data consists of 8 attributes and the corresponding student emotion. These attributes are derived from the survey questions that focus on the Light-Weight team

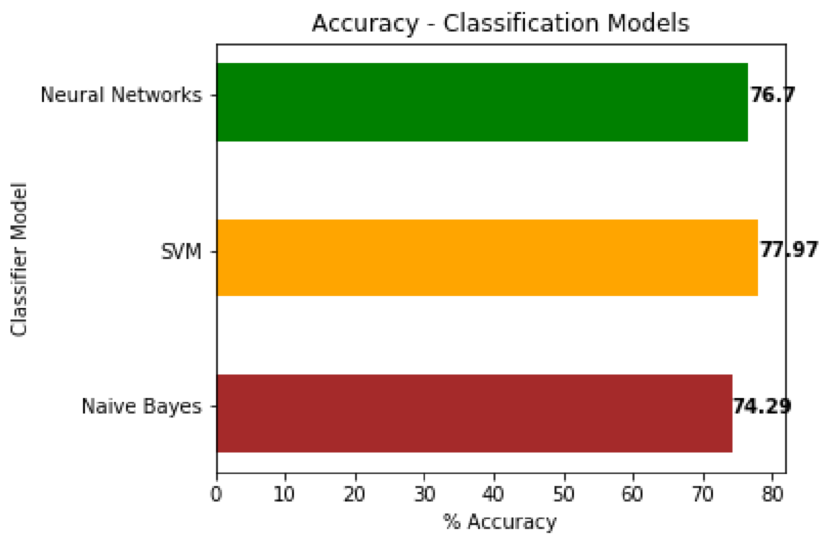

Figure 5.19: Classifier - Accuracy.

work activities and assignments.

The Table. 5.45 shows sample action rules extracted using this data.

Let us consider the action rule AR_{AT1}. This rule suggests that if the team members are technically effective, Number of members in the team are in the range of 5 to 7 students, and they feel a complete sense of belonging in the team then the student emotion could be changed from 'Anticipation' to 'Trust'. This provides an useful insight to the course instructor, that more attention is required when forming team members. The instructor should consider the technical complexity of the assignments or activities. Based on the complexity, the instructor could choose to have a poll at the beginning of the semester requesting students to state their level of expertise in each area required for the assignments in general. These results may provide a basic idea on the technical capability of students in the class, based on which the instructor could then form the teams. This way the Team Members formation would be more efficient and helpful to the students. Also the rule suggests that students feel better with 5 to 7 members in the team.

Table 5.45: Sample Action Rules - Student Survey Data 1 - Team Work and Student Emotion - Summer 2019.

Enhance Student Emotion - Anticipation $\rightarrow$ Trust

1. $AR_{AT1} : (LikeTeamWork, 4QuiteaBit \rightarrow 5VeryMuch) \wedge (TeamSenseofBelonging, 3AverageSenseofBelongingtotheTeam \rightarrow 4CompleteSenseofBelongingtotheTeam) \wedge (TeamMemberResponsibility, HelpfulMembers \rightarrow TechnicallyEffectiveMembers) \wedge (NumberofTeamMembers = 5to7) \Longrightarrow (StudentEmotion, Anticipation \rightarrow Trust)[Support : 108, Confidence : 100\%]$

2. $AR_{AT2} : (TeamMemberResponsibility, HelpfulMembers \rightarrow TechnicallyEffectiveMembers) \wedge (TeamWorkHelpedDiversity = 2Occasionally) \wedge (GroupAssignmentBenefit, SharedKnowledge \rightarrow AllofThem) \Longrightarrow (StudentEmotion, Anticipation \rightarrow Trust)[Support : 108, Confidence : 75.3\%]$

Enhance Student Emotion - Sadness $\rightarrow$ Joy

1. $AR_{SJ1} : (LikeTeamWork, 3Somewhat \rightarrow 5VeryMuch) \wedge (GroupAssignmentBenefit, EliminateStress \rightarrow AllofThem) \Longrightarrow (StudentEmotion, Sadness \rightarrow Joy)[Support : 36, Confidence : 100\%]$

2. $AR_{SJ2} : (TeamFormation = 3Average) \wedge (TeamSenseofBelonging = 3AverageSenseofBelongingtotheTeam) \wedge (TeamMemberResponsibility, ResponsibleMembers \rightarrow TechnicallyEffectiveMembers) \wedge (GroupAssignmentBenefit, EliminateStress \rightarrow AllofThem) \Longrightarrow (StudentEmotion, Sadness \rightarrow Joy)[Support : 36, Confidence : 50\%]$

Table 5.46: Sample Action Rules - Student Survey Data 1 - Team Work and Student Emotion - Summer and Fall 2019.

Enhance Student Emotion - Anticipation → Trust
1. AR_{AT3} : $(LikeTeamWork, 3Somewhat \rightarrow 5VeryMuch) \wedge (TeamFormation, 3Average \rightarrow 4Perfect) \wedge (TeamSenseofBelonging, 3AverageSenseofBelongingtotheTeam \rightarrow 4CompleteSenseofBelongingtotheTeam) \implies (StudentEmotion, Anticipation \rightarrow Trust)[Support : 163.0, Confidence : 80.0\%]$ 2. AR_{AT4} : $(TeamWorkHelpedDiversity, 2Occasionally \rightarrow 3Often) \wedge (GroupAssignmentBenefit, None \rightarrow SharedKnowledge) \implies (StudentEmotion, Anticipation \rightarrow Trust)[Support : 108, Confidence : 73.34\%]$
Enhance Student Emotion - Sadness → Joy
1. AR_{SJ3} : $(TeamFormation, 2BelowAverage \rightarrow 4Perfect) \wedge (TeamSenseofBelonging, 2BelowAverageSenseofBelongingtotheTeam \rightarrow 4CompleteSenseofBelongingtotheTeam) \implies (StudentEmotion, Sadness \rightarrow Joy)[Support : 28, Confidence : 100\%]$ 2. AR_{SJ4} : $(LikeTeamWork, 1Dont \rightarrow 3Somewhat) \wedge (TeamFormation, 2BelowAverage \rightarrow 3Average) \wedge (TeamWorkHelpedDiversity, 1Never \rightarrow 2Occasionally) \wedge (GroupAssignmentBenefit, None \rightarrow EliminateStress) \implies (StudentEmotion, Sadness \rightarrow Joy)[Support : 27, Confidence : 100\%]$

5.2.6.2 Student Survey Data 2 - Active Learning, Teaching Method, and Student Emotion

This data consists of 11 attributes and the corresponding student emotion. These attributes are derived from the survey questions that focus on the Active Learning method adopted for the courses and the teaching method.

The Table. 5.47 shows sample action rules extracted using this data. Let us consider the action rule AR_{AT5}. This rule suggests that there is needed some improvement in terms of exam preparation guides provided for the students. In other words if $(ExamPrepGuidesSampleQuestionsHelpfulness, 3 \rightarrow 5)$ and the Individual assignment $(IndividualAssignments, 3 \rightarrow 5)$ then there is a good chance that students feel better in the learning process, which would ultimately lead to better outcomes.

Table 5.47: Sample Action Rules - Student Survey Data 2 - Active Learning, Teaching Method, and Student Emotion - Summer 2019.

Enhance Student Emotion - Anticipation → Trust
1. AR_{AT5} : $(ExamPrepGuidesSampleQuestionsHelpfulness, 3 \rightarrow 5) \wedge (SyllabusAssignmentsPriorAvailabilityHelped = 5) \wedge (VideoCasesAssignmentsHelpful = 5) \wedge (IndividualAssignments, 3 \rightarrow 5) \Longrightarrow (StudentEmotion, Anticipation \rightarrow Trust)[Support: 148, Confidence: 66.4\%]$ 2. AR_{AT6} : $(ExamPrepGuidesSampleQuestionsHelpfulness, 3 \rightarrow 4) \wedge (ActiveLearningMethodologyVsTraditionalMethod, 3 \rightarrow 5) \Longrightarrow (StudentEmotion, Anticipation \rightarrow Trust)[Support: 110, Confidence: 60.99\%]$
Enhance Student Emotion - Sadness → Joy
1. AR_{SJ5} : $(OpenBookExamHelpsLowerAnxiety = 5) \wedge (ExamPrepGuidesSampleQuestionsHelpfulness, 3 \rightarrow 4) \wedge (SyllabusAssignmentsPriorAvailabilityHelped, 1 \rightarrow 5) \wedge (PeerTeachingInTeamsHelpSelfLearning = 4) \Longrightarrow (StudentEmotion, Sadness \rightarrow Joy)[Support: 37, Confidence: 100\%]$ 2. AR_{SJ6} : $(VideoCasesAssignmentsHelpful, 3 \rightarrow 5) \wedge (FlippedClassHelpsBetterLearning, 2 \rightarrow 4) \wedge (PeerTeachingHelpedUnderstanding = 2) \Longrightarrow (StudentEmotion, Sadness \rightarrow Joy)[Support: 36, Confidence: 100\%]$

5.3 Business Data

In this section we use the original raw data explained in section 3.3 apply the following methods: Emotion Labeling (section. 4.2) on the customer comment - text data, that labels the text into eight basic emotions ('joy', 'sadness', 'surprise', 'trust', 'anticipation', 'disgust', 'fear', 'anger'); attribute reduct (section. 4.6.3); and Actionable Pattern Discovery.

5.3.1 Emotion Labeling

After Emotion Labeling, the Net Promotor Score Business Data contains the following features as mentioned in Table. 5.48.

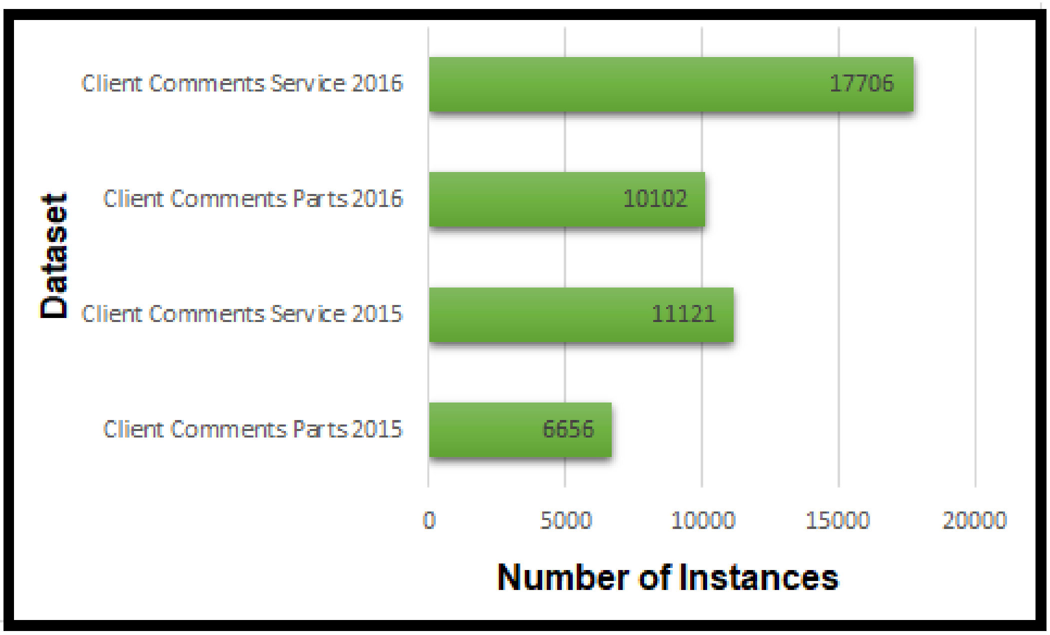

Figure 5.20: Business Data Distribution.

Table 5.48: Business Data - Features.

Dataset	# Instances	# Attributes
Client Comments Parts 2015	6656	23
Client Comments Service 2015	11121	22
Client Comments Parts 2016	10102	37
Client Comments Service 2016	17706	36

5.3.2 Reducts using Rough Sets

The experiments are performed in both Single node local machine and University Research Cluster (URC). The data in Table. 5.48 is used to run two reduct computaion algorithms using rough sets explained in section 4.6.3.1 and section 4.6.3.2.

Table 5.49: Business Data - Quick Reduct.

Dataset	# Reducts	Single Node - Time Taken	URC Cluster - Time Taken
Client Comments Parts 2015	21	36.3 seconds	32.74 seconds
Client Comments Service 2015	21	1.05 mins	1.102 mins
Client Comments Parts 2016	34	3.116 mins	3.092 mins
Client Comments Service 2016	33	6.557 mins	6.26 mins

Table 5.50: Business Data - Discerbility Matrix - Single Node.

Dataset	Single Node Time Taken (mins)		# Reducts
	Discernibility Matrix	Reduct Computation	
Client Comments Parts 2015	4.18	4.63	21
Client Comments Service 2015	50.85	8.93	21
Client Comments Parts 2016	23.18	12.70	32
Client Comments Service 2016	72	157.8	32

Table 5.51: Business Data - Discerbility Matrix - URC Cluster.

Dataset	URC Cluster		# Reducts
	Time Taken (mins)		
	Discernibility Matrix	Reduct Computation	
Client Comments Parts 2015	2.99	2.28	21
Client Comments Service 2015	59.05	8.75	21
Client Comments Parts 2016	7.6	981secs	32
Client Comments Service 2016	184.8	70.8	32

5.3.3 Action Rule - Vertical Data Distribution

We generate Action Rules for the Business Data labeled with emotions, generated in section 5.3.1. The dataset consists of records describing attributes of Machinery parts company sales, including the customer feedback text comments and their corresponding Emotion in one of the following category 'joy', 'trust', 'anticipation', 'fear', 'disgust', 'sadness', 'anger', 'surprise'. We choose Emotion as the decision attribute and generate Action Rules that help identify changes that are required for the Emotion to be more positive. For example, to change the emotion from 'anticipation' to 'trust'.

The decision problem here is to suggest possible recommendations to the Machinery parts companies sales, on how to make customers feel much better which ultimately helps improve customer loyalty and satisfaction.

Table 5.52: Business Data - Action Rules - Execution Time.

Dataset	Single Node - Time Taken (seconds)	URC Cluster - Time Taken (seconds)
Client Comments Parts 2015	part1 - 20, part2 - 43, Combine - 4	part1 - 12, part2 - 35, Combine - 1
Client Comments Service 2015	part1 - 98, part2 - 117, Combine - 13	part1 - 43, part2 - 59, Combine - 6
Client Comments Parts 2016	part1 - 535, part2 - 123, Combine - 121	part1 - 614, part2 - 65, Combine - 49
Client Comments Service 2016	part1 - 455, part2 - 121, Combine - 24	part1 - 200, part2 - 74, Combine - 11

Table. 5.52 shows the execution time for each of the Business datasets in seconds. The data is first processed to contain only attributes from the attribute reduction step. Later the data is divided into two parts with decision attribute 'Emotion' on each part of the data. The data was divided vertically into part1 and part2 for distributed processing. Then combined to show the final result. Extracted sample Action Rules are provided in Table. 5.53.

Table 5.53: Sample Action Rules - Business Data.

Enhance Customer Emotion - Anticipation $\rightarrow$ Trust
1. $ClientCommentParts2015_{AR1}$: $(BenchmarkAllOverallSatisfaction = 10) \wedge (BenchmarkPartsExplanationofDeliveryOptionsCosts = 10) \wedge (BenchmarkPartsHowOrdersArePlaced, 2 \rightarrow 3) \implies (Emotion, Anticipation \rightarrow Trust)[Support : 10, Confidence : 9.21\%]$ 2. $ClientCommentParts2015_{AR2}$: $(BenchmarkAllOverallSatisfaction = 10) \wedge (BenchmarkPartsOrderAccuracy = 10) \wedge (BenchmarkPartsTimeitTooktoPlaceOrder, 9 \rightarrow 10) \implies (Emotion, Anticipation \rightarrow Trust)[Support : 7, Confidence : 8.88\%]$ 3. $ClientCommentParts2016_{AR1}$: $(BenchmarkAllOverallSatisfaction = 10) \wedge (BenchmarkPartsPartsAvailability = 10) \wedge (BenchmarkPartsPromptNotificationofBackOrders = 10) \implies (Emotion, Anticipation \rightarrow Trust)[Support : 151, Confidence : 13.81\%]$

Let us consider the rule ClientCommentParts2015$_{\mathrm{AR2}}$ in Table. 5.53. According to the Action Rule, if the Benchmark value of time it took to place the order ($BenchmarkPartsTimeitTooktoPlaceOrder$) is changed from 9 to 10 and Overall Satisfaction and Order Accuracy values are maintained at 10, then it is possible to gain customer Trust.

CHAPTER 6: CONCLUSIONS

Emotions and feelings accompany us throughout the span of our lives and color the way we build and maintain the basis for interactions with people in a society [108], and through computer-based systems, including human-computer interaction. With rise of social media such as blogs, forums, social networking sites like Twitter, Facebook, and proliferation of online product reviews, the need for Sentiment Analysis techniques and Emotion Detection from text has been ever increasing. Additional applications include: customer care services, recommendation systems for online shopping, text messages, E-Learning, and student teaching evaluations, as well as the smart phones and technology of the future, which is able to detect and recognize human emotions. Mining for Actionable knowledge and providing Actionable Recommendations, which can alter emotions from negative to positive is a challenging and important subject, that benefits all emotion recognition systems.

6.1 Social Media Text - Sentiment Mining and Actionable Pattern Discovery

This work proposed a new approach to analyse sentiment of tweets through mining actionable patterns via action rules. We suggest actions that can be undertaken to reclassify user sentiment from negative to positive and neutral to positive using comments. We also suggest actions of how users can increase their friends count. We provide implementation on both single machine and cloud distributed environment for scalability purpose. We compare the results with single machine implementation and distributed Hadoop MapReduce framework. Our experiments show that the processing of the proposed algorithm runs faster on distributed environment than on single machine. The proposed method can scale to accommodate large social media data size.

6.2 Social Media Text - Emotion Mining and Actionable Pattern Discovery

In this work, we perform automatic detection of emotions in Twitter dataset. We utilize the National Research Council - NRC Emotion Lexicon to label the Emotion class for our data. We examine several classifiers and choose the decision tree and decision forest (random forest) as well as the decision table majority methods. These methods have not been used before for Twitter emotion classification. We report higher classification accuracy than any previous works. Our accuracy is 88.45% - 99%, compared to 60% - 90% for previous works, which mostly use the support vector machines and k-nearest neighbor classifiers. We implement the data collection, pre-processing, feature augmentation, and the proposed classifiers on both WEKA and Apache Spark system over Hadoop cluster for scalability purpose. Our Spark implementation is able to scale to BigData sets, as data is divided into partitions and is processed in parallel at each cluster node. Applications of this work include detection of emotions for: improving customer satisfaction, e-learning, psychological health care, and designing intelligent phones and devices which recognize user emotion.

In this work, we automatically detect user emotion from tweet data using the NRC Emotion Lexicon [78], [79] to label the Emotion class for our data. We use Support Vector Machine with Multiclass classification in particular ONE- AGAINST-ALL implementation in both WEKA data mining software [229] and Apache Spark system [40] over Hadoop 6 node Cluster for big data scalability. We achieve accuracy of 84.9% to 89.76%. The Spark system is able to scale to BigData with six node cluster, as the data is partitioned into several sets and processed in parallel at each cluster node. This is an extension of previous study [241] of finding user emotions from tweet data using the NRC emotion lexicon to label the emotion class for our data. In the previous work, we examined several classifiers including Decision Tree, Deci-sion Forest, and Decision Table Majority. In this work, we extract Action Rules to identify what factors that can be improved in order for a user to attain a more desir-able positive emotion. We suggest actions that can be undertaken to reclassify user emotion from a negative emotion to more positive emotion. For instance from ‘sadness’ to ‘joy’, ‘sadness’ to ‘trust’, and ‘fear’ to ‘trust’.

Action Rules are actionable recommendations that suggest possible transition from one state to another for benefit of the user. Emotion mining from text has its root in many application disciplines namely, Psychology, Neuroscience, Social Science, Computer Science, and others. Systems that can detect emotions and suggest actionable recommendations has many potential applications. In *Education*: to benefit students, institution and faculty in terms of Teaching Models, Learning Environment; in *Customer Care Service*: based on emotions from customer feedback, these actionable patterns can suggest what aspects of the service could be improved or changed for better customer satisfaction; in *Technology of Future*: like smart phones, to predict user Emotions and suggest suitable Movies, Music or call a Family member or friend. In this work we propose a novel approach of hybrid association action rule algorithm by combining the rule based and object based approach to reduce the overhead of the iterative procedure. We test our algorithm using Twitter dataset and compare with existing method of Association Action Rules mining. Twitter dataset is labeled with emotions based on the tweet text and is a densely populated data with more number of attributes. It is observed that the proposed algorithm is able to generate complete set of Action Rules given the entire dataset consisting of 174888 instances in less than 500 seconds. While the previously existing algorithm can not handle the entire dataset, it fails due to memory overhead of the iterative procedure.

6.3 Education Data - Emotion Mining

In this work we perform sentiment analysis, and emotion detection on the qualitative feedback provided by students in course evaluations. We identify eight basic human emotions: ‘anger’, ‘fear’, ‘joy’, ‘surprise’, ‘anticipation’, ‘disgust’, ‘sadness’, and ‘trust’ along with the two sentiment polarities ‘positive’ and ‘negative’. We use these emotions to analyze and assess the impact and effectiveness of Active Learning methods incorporated in the classroom during the years 2016 and 2017, compared to previous years. Active Learning methods were initiated in 2016, and implemented in 2017, in the courses including Light Weight Teams [87], [88], and Flipped Classroom [89]. Results show evidence that words associated with positive emotions, and trust have increased in the recent years compared to 2014. At the same time, occurrences of negative emotion words in the Fig. 5.16. have decreased. Therefore, we claim that the implementation of Light Weight Teams and Flipped Classroom Active Learning methods increase positive emotions among students and improve their learning experience.

We apply neural networks classifier for emotion detection in student evaluation of teaching. We use Keras Deep Learning API [240]. Using appropriate number of epochs for training on the source domain results in better performance. We also compare the neural networks model with the traditional text classification models like Naive Bayes and Support Vector Machine. We notice that neural networks yields (76.7%) similar performance to traditional text classification models like Naive Bayes (74.79%) and Support Vector Machine (77.97%). We are able to achieve good accuracy, even the size of the dataset is not big. Generally, neural networks for classification may required bigger dataset.

In this study we propose a new approach of identifying patterns and enhancing student emotions based on Student Survey Data. We propose to use Actionable Pattern Mining framework called Action Rule Mining. The data collected for this study is original data, with specifically designed questions, collected by the authors, from a public research university in United States. This kind of data collected in educational setting combined with the Action Rule Mining method provides intutive information on how student emotions can be altered from negative $\rightarrow$ postive , or neutral $\rightarrow$ positive, including classroom environment, teaching style, teamwork, and school facilites.

6.4 Net Promotor Score - Business Data - Emotion Mining

Most of the work to build intelligent data mining systems, machine learning algorithms, pattern mining tend to face the bottleneck at the point of the acquiring best possible knowledge to extract useful patterns. In this paper we use the rough-set attribute reduction algorithms, concept of sub systems and SparkR distributed framework to overcome the problem of high dimesion data. We choose the best possible attributes for pattern discovery. We apply our method on Business Data - Net Promoter Score - suggest ways of re-classifying customers from Detractor to Promoter and from Neutral to Promoter.

CHAPTER 7: FUTURE WORKS

7.1 Social Media Text - Sentiment Mining and Actionable Pattern Discovery

We proposed a new approach to analyse sentiment of tweets through mining actionable patterns via action rules. The proposed method can scale to accommodate large social media data size. In the future, we plan to augment the data set with more syntactical parts including nouns and adjectives and to build lexicons for specific subjects. For example, financial, medical, and industrial topics.

7.2 Social Media Text - Emotion Mining and Actionable Pattern Discovery

We performed automatic detection of emotions in Twitter dataset. We utilized the National Research Council - NRC Emotion Lexicon to label the Emotion class for our data. In the future, we plan to perform actionable pattern mining on our Twitter Emotion dataset to suggest ways to alter the user emotions from negative to positive sentiment.

We used Support Vector Machine with Multiclass classification in particular ONE-AGAINST-ALL implementation in both WEKA data mining software [229] and Apache Spark system [40] over Hadoop 6 node Cluster for big data scalability. We extracted Action Rules to identify what factors that can be improved in order for a user to attain a more desirable positive emotion. In the future, we plan further test with larger social networking data. We also plan to apply this system for customer surveys and education evaluations.

7.3 Education Data - Emotion Mining

We performed sentiment analysis, and emotion detection on the qualitative feedback provided by students in course evaluations. We use these emotions to analyze and assess the impact and effectiveness of Active Learning methods incorporated in the classroom. In the future, we plan to extend this work, by analyzing more Active Learning pedagogy methods such as gamification. We also plan to focus on women and minorities in computing discipline. It is evident from our collected survey data, that using the suggested methods of Action Rule Mining, we can extract meaningful insights in terms of the advatages, and improvements of teaching style, material provided, and learning methods adopted. We conducted this study, by specifically designing the survey questions to fit the Action Rules Mining Algorithm. We collected this Survey data in a single institution with 250 participants in courses offered in Computer Science discipline. The results as part of this study are exploratory and provide approaches for such analysis that help in Innovation in Education. However this study does not provide conclusive evidence, due to the limited number of participants. In the future we plan to increase the number of participants by conducting this survey with higher number of classes and participants to identify emotions from student evaluations of teaching and extract actionable patterns to help improve the teaching models and learning performance.

7.4 Business Data - Emotion Mining, Attribute Selection, and Actionable Pattern Discovery

We use the Business Data with customer comments (text) and labeled the text with eight basic Emotions namely ‘joy’, ‘anticipation’, ‘trust’, ‘surprise’, ‘disgust’, ‘anger’, ‘fear’, ‘sadness’. Applied the rough sets method of finding the most importnat attributes in the dataset. These attributes are used to generate Action Rules that suggest how to enhance customer emotion (to better positive emotion) which ultimately leads to improved customer satisfaction and loyalty. In future we plan to identify and group customers by data fusion which improves the knowledge base by collating multiple data sources (in this case data across several years). We plan to extend the Business data Customer Surveys data mining, as well as the Education and Social Media mining, through improved Action Rules methods for more positive suggestions of Emotion re-classifcation.

REFERENCES

[1] M. Hossin and S. M.N, "A review on evaluation metrics for data classification evaluations," *International Journal of Data Mining & Knowledge Management Process*, vol. 5, no. 2, pp. 01–11, 2015.

[2] D. Reinsel, J. Gantz, and J. Rydning, "Data age 2025: the digitization of the world from edge to core," *Seagate https://www. seagate. com/files/www-content/our-story/trends/files/idc-seagate-dataage-whitepaper. pdf*, 2018.

[3] S. Venkataraman, Z. Yang, D. Liu, E. Liang, H. Falaki, X. Meng, R. Xin, A. Ghodsi, M. Franklin, I. Stoica, *et al.*, "Sparkr: Scaling r programs with spark," in *Proceedings of the 2016 International Conference on Management of Data*, pp. 1099–1104, ACM, 2016.

[4] J. Posner, J. A. Russell, and B. S. Peterson, "The circumplex model of affect: An integrative approach to affective neuroscience, cognitive development, and psychopathology," *Development and psychopathology*, vol. 17, no. 3, pp. 715–734, 2005.

[5] M. Faith and J. F. Thayer, "A dynamical systems interpretation of a dimensional model of emotion," *Scandinavian Journal of Psychology*, vol. 42, no. 2, pp. 121–133, 2001.

[6] O. Spindler and T. Fadrus, "Grimace project documentation," *Vienna University of Technology*, 2009.

[7] P. Ekman, "An argument for basic emotions," *Cognition & emotion*, vol. 6, no. 3-4, pp. 169–200, 1992.

[8] R. Plutchik and H. Kellerman, *Emotion, psychopathology, and psychotherapy*, vol. 5. Elsevier Science, 2013.

[9] P. Shaver, J. Schwartz, D. Kirson, and C. O'connor, "Emotion knowledge: Further exploration of a prototype approach.," *Journal of personality and social psychology*, vol. 52, no. 6, pp. 1061–1086, 1987.

[10] H. Lovheim, "A new three-dimensional model for emotions and monoamine neurotransmitters," *Medical hypotheses*, vol. 78, no. 2, pp. 341–348, 2012.

[11] S. Tomkins, *Affect imagery consciousness: Volume I: The positive affects.* Springer publishing company, 1962.

[12] S. S. Tomkins, "Affect imagery consciousness: Volume ii: The negative affects.," no. 2, 1963.

[13] D. C. Rubin and J. M. Talarico, "A comparison of dimensional models of emotion: Evidence from emotions, prototypical events, autobiographical memories, and words," *Memory*, vol. 17, no. 8, pp. 802–808, 2009.

[14] J. A. Russell, "A circumplex model of affect.," *Journal of personality and social psychology*, vol. 39, no. 6, pp. 1161–1178, 1980.

[15] M. M. Bradley, M. K. Greenwald, M. C. Petry, and P. J. Lang, "Remembering pictures: Pleasure and arousal in memory.," *Journal of experimental psychology: Learning, Memory, and Cognition*, vol. 18, no. 2, pp. 379–390, 1992.

[16] D. Watson and A. Tellegen, "Toward a consensual structure of mood.," *Journal of Psychological bulletin*, vol. 98, no. 2, pp. 219–235, 1985.

[17] T. Dixon, *From Passions to Emotions: The Creation of a Secular Psychological Category*. Cambridge University Press, 2003.

[18] N. Gupta, M. Gilbert, and G. Di Fabbrizio, "Emotion detection in email customer care," in *Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, CAAGET '10, (Stroudsburg, PA, USA), pp. 10–16, Association for Computational Linguistics, 2010.

[19] S. Voeffray, "Emotion-sensitive human-computer interaction (hci): State of the art-seminar paper," *Emotion Recognition*, pp. 1–4, 2011.

[20] M. De Choudhury, M. Gamon, S. Counts, and E. Horvitz, "Predicting depression via social media.," *7th International Conference on Web Logs and Social Media, ICWSM'13*, vol. 13, pp. 1–10, 2013.

[21] K. Luyckx, F. Vaassen, C. Peersman, and W. Daelemans, "Fine-grained emotion detection in suicide notes: A thresholding approach to multi-label classification," *Biomedical informatics insights*, vol. 5, pp. 61–69, 2012.

[22] J. Ranganathan, A. S. Irudayaraj, and A. A. Tzacheva, "Action rules for sentiment analysis on twitter data using spark," in *Data Mining Workshops (ICDMW), 2017 IEEE International Conference on*, pp. 51–60, IEEE, 2017.

[23] P. Spooren, B. Brockx, and D. Mortelmans, "On the validity of student evaluation of teaching: The state of the art," *Review of Educational Research*, vol. 83, no. 4, pp. 598–642, 2013.

[24] A. Tzacheva and J. Ranganathan, "Emotion mining from student comments a lexicon based approach for pedagogical innovation assessment," *European Journal of Education and Applied Psychology*, no. 3, pp. 3–13, 2018.

[25] S. Mac Kim and R. A. Calvo, "Sentiment analysis in student experiences of learning," in *Educational Data Mining*, 2010.

[26] C. K. Leong, Y. H. Lee, and W. K. Mak, "Mining sentiments in sms texts for teaching evaluation," *Expert Systems with Applications*, vol. 39, no. 3, pp. 2584–2589, 2012.

[27] N. Altrabsheh, M. Gaber, and M. Cocea, "Sa-e: sentiment analysis for education," in *International Conference on Intelligent Decision Technologies*, vol. 255, pp. 353–362, 2013.

[28] K. Wang, Y. Jiang, and A. Tuzhilin, "Mining actionable patterns by role models," in *22nd International Conference on Data Engineering (ICDE'06)*, pp. 16–16, April 2006.

[29] Z. W. Ras, A. Dardzinska, L.-S. Tsay, and H. Wasyluk, "Association action rules," in *Data Mining Workshops, 2008. ICDMW'08. IEEE International Conference on*, pp. 283–290, IEEE, 2008.

[30] Z. W. Ras and E. Wyrzykowska, "Arogs: Action rules discovery based on grabbing strategy and lers," in *Proceedings of 2007 ECML/PKDD Third International Workshop on Mining Complex Data (MCD 2007), Univ. of Warsaw, Poland*, pp. 95–105, 2007.

[31] Z. W. Ras, E. Wyrzykowska, and H. Wasyluk, "Aras: Action rules discovery based on agglomerative strategy," in *Mining Complex Data, Post-Proceedings of 2007 ECML/PKDD Third International Workshop (MCD 2007), LNAI*, vol. 4944, pp. 196–208, Springer, 2008.

[32] C. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. Bethard, and D. McClosky, "The stanford corenlp natural language processing toolkit," in *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, pp. 55–60, 2014.

[33] A. Tzacheva, J. Ranganathan, and A. Bagavathi, "Action rules for sentiment analysis using twitter," *International Journal of Social Network Mining*, 2018.

[34] Z. W. Ras and A. Wieczorkowska, "Action-rules: How to increase profit of a company," in *Principles of Data Mining and Knowledge Discovery, Proceedings of PKDD'00, Lyon, France, LNAI*, no. 1910, pp. 587–592, Springer, 2000.

[35] A. Dardzinska and Z. W. Ras, "Extracting rules from incomplete decision systems: System erid," in *Foundations and Novel Approaches in Data Mining, (Eds. T.Y. Lin, S. Ohsuga, C.J. Liau, X. Hu), Studies in Computational Intelligence*, vol. 9, pp. 143–154, Springer, 2005.

[36] S. Im and Z. W. Ras, "Action rule extraction from a decision table: Ared," in *Foundations of Intelligent Systems, Proceedings of ISMIS'08, A. An et al. (Eds.), Toronto, Canada, LNAI*, vol. 4994, pp. 160–168, Springer, 2008.

[37] Z. W. Ras and L.-S. Tsay, "Discovering extended action-rules (system dear)," in *Intelligent Information Systems 2003, Advances in Soft Computing, Proceedings of the IIS'2003 Symposium, Zakopane, Poland*, pp. 293–300, Springer, 2003.

[38] Z. W. Ras and A. Dardzinska, "Action rules discovery, a new simplified strategy," in *Foundations of Intelligent Systems, Proceedings of ISMIS'06, LNAI*, no. 4203, pp. 445–453, Springer, 2006.

[39] J. Dean and S. Ghemawat, "Mapreduce: simplified data processing on large clusters," *Communications of the ACM*, vol. 51, no. 1, pp. 107–113, 2008.

[40] M. Zaharia, M. Chowdhury, M. J. Franklin, S. Shenker, and I. Stoica, "Spark: Cluster computing with working sets," in *Proceedings of the 2Nd USENIX Conference on Hot Topics in Cloud Computing*, HotCloud'10, (Berkeley, CA, USA), pp. 10–10, USENIX Association, 2010.

[41] M. Zaharia, M. Chowdhury, T. Das, A. Dave, J. Ma, M. McCauley, M. J. Franklin, S. Shenker, and I. Stoica, "Resilient distributed datasets: A fault-tolerant abstraction for in-memory cluster computing," in *Proceedings of the 9th USENIX Conference on Networked Systems Design and Implementation*, NSDI'12, (Berkeley, CA, USA), pp. 2–2, USENIX Association, 2012.

[42] K. Shvachko, H. Kuang, S. Radia, and R. Chansler, "The hadoop distributed file system," in *2010 IEEE 26th Symposium on Mass Storage Systems and Technologies (MSST)*, pp. 1–10, May 2010.

[43] V. K. Vavilapalli, A. C. Murthy, C. Douglas, S. Agarwal, M. Konar, R. Evans, T. Graves, J. Lowe, H. Shah, S. Seth, B. Saha, C. Curino, O. O'Malley, S. Radia, B. Reed, and E. Baldeschwieler, "Apache hadoop yarn: Yet another resource negotiator," in *Proceedings of the 4th Annual Symposium on Cloud Computing*, SOCC '13, (New York, NY, USA), pp. 5:1–5:16, ACM, 2013.

[44] A. A. Tzacheva, A. Bagavathi, and P. D. Ganesan, "Mr-random forest algorithm for distributed action rules discovery," *International Journal of Data Mining & Knowledge Management Process (IJDKP)*, vol. 6, no. 5, pp. 15–30, 2016.

[45] E. Fox, *Emotion science cognitive and neuroscientific approaches to understanding human emotions*. Palgrave Macmillan, 2008.

[46] A. Yadollahi, A. G. Shahraki, and O. R. Zaiane, "Current state of text sentiment analysis from opinion to emotion mining," *ACM Computing Surveys (CSUR)*, vol. 50, no. 2, p. 25, 2017.

[47] F. Rangel and P. Rosso, "On the impact of emotions on author profiling," *Information processing & management*, vol. 52, no. 1, pp. 73–92, 2016.

[48] W. Wang, L. Chen, K. Thirunarayan, and A. P. Sheth, "Harnessing twitter" big data" for automatic emotion identification," in *Privacy, Security, Risk and Trust (PASSAT), 2012 International Conference on and 2012 International Confernece on Social Computing (SocialCom)*, pp. 587–592, IEEE, 2012.

[49] J. Bollen, H. Mao, and A. Pepe, "Modeling public mood and emotion: Twitter sentiment and socio-economic phenomena.," *In Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media*, vol. 11, pp. 450–453, 2011.

[50] M. Hasan, E. Agu, and E. Rundensteiner, "Using hashtags as labels for supervised learning of emotions in twitter messages," in *ACM SIGKDD Workshop on Health Informatics, New York, USA*, 2014.

[51] M. W. Staff, "Online dictionary: http://www. mw. com/home. htm, 2004. merriam–webster, inc., 47 federal street," *PO Box*, vol. 281.

[52] C. Honey and S. C. Herring, "Beyond microblogging: Conversation and collaboration via twitter," in *2009 42nd Hawaii International Conference on System Sciences*, pp. 1–10, Jan 2009.

[53] Y. Chang, L. Tang, Y. Inagaki, and Y. Liu, "What is tumblr: A statistical overview and comparison," *SIGKDD Explor. Newsl.*, vol. 16, no. 1, pp. 21–29, 2014.

[54] D. Sullivan, "comscore media metrix search engine ratings," *Search Engine Watch*, vol. 21, 2006.

[55] A. Java, X. Song, T. Finin, and B. Tseng, "Why we twitter: understanding microblogging usage and communities," in *Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 workshop on Web mining and social network analysis*, pp. 56–65, ACM, 2007.

[56] H.-C. Chang, "A new perspective on twitter hashtag use: Diffusion of innovation theory," *Proceedings of the Association for Information Science and Technology*, vol. 47, no. 1, pp. 1–4, 2010.

[57] S. D'Alfonso, O. Santesteban-Echarri, S. Rice, G. Wadley, R. Lederman, C. Miles, J. Gleeson, and M. Alvarez-Jimenez, "Artificial intelligence-assisted online social therapy for youth mental health," *Frontiers in psychology*, vol. 8, p. 796, 2017.

[58] D. Tantam, "The machine as psychotherapist: impersonal communication with a machine," *Advances in Psychiatric treatment*, vol. 12, no. 6, pp. 416–426, 2006.

[59] H. Kaur, "Actionable rules: Issues and new directions.," in *in Transactions on Engineering, Computing and Technology, World Informatica Society*, pp. 61–64, Citeseer, 2005.

[60] Z. He, X. Xu, S. Deng, and R. Ma, "Mining action rules from scratch," *Expert Systems with Applications*, vol. 29, no. 3, pp. 691–699, 2005.

[61] C. E. Izard, *Human emotions.* Springer Science & Business Media, 2013.

[62] E. Huffman and S. Prentice, "Social media's new role in emergency management," tech. rep., Idaho National Laboratory (INL), 2008.

[63] S. Mohammad and F. Bravo-Marquez, "Emotion intensities in tweets," in *Proceedings of the 6th Joint Conference on Lexical and Computational Semantics, *SEM @ACM 2017, Vancouver, Canada, August 3-4, 2017*, pp. 65–77, 2017.

[64] Z. Zhang, Q. Zhou, and H. Wu, "Nlpzzx at semeval-2018 task 1: Using ensemble method for emotion and sentiment intensity determination," in *Proceedings of The 12th International Workshop on Semantic Evaluation*, pp. 116–122, 2018.

[65] L. De Bruyne, O. De Clercq, and V. Hoste, "Lt3 at semeval-2018 task 1: A classifier chain to detect emotions in tweets," in *SemEval-2018*, pp. 123–127, Association for Computational Linguistics, 2018.

[66] M. Aono and S. Himeno, "Kde-affect at semeval-2018 task 1: Estimation of affects in tweet by using convolutional neural network for n-gram," in *Proceedings of The 12th International Workshop on Semantic Evaluation*, pp. 156–161, 2018.

[67] A. Dardzinska, *Action rules mining*, vol. 468. Springer, 2012.

[68] A. C. Weidman, C. M. Steckler, and J. L. Tracy, "The jingle and jangle of emotion assessment: Imprecise measurement, casual scale usage, and conceptual fuzziness in emotion research.," *Emotion*, vol. 17, no. 2, p. 267, 2017.

[69] M. Abdul-Mageed and L. Ungar, "Emonet: Fine-grained emotion detection with gated recurrent neural networks," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 718–728, 2017.

[70] H. K. Wachtel, "Student evaluation of college teaching effectiveness: A brief review," *Assessment & Evaluation in Higher Education*, vol. 23, no. 2, pp. 191–212, 1998.

[71] H. Remmers, "The relationship between students' marks and student attitude toward instructors.," *School & Society*, 1928.

[72] H. Remmers, "To what extent do grades influence student ratings of instructors?," *The Journal of Educational Research*, 1930.

[73] F. Zabaleta, "The use and misuse of student evaluations of teaching," *Teaching in Higher Education*, vol. 12, no. 1, pp. 55–76, 2007.

[74] D. E. Clayson, "Student evaluations of teaching: Are they related to what students learn? a meta-analysis and review of the literature," *Journal of Marketing Education*, vol. 31, no. 1, pp. 16–30, 2009.

[75] W. James, “What is an emotion?,” *in journal of Mind*, vol. 9, no. 34, pp. 188–205, 1884.

[76] R. Plutchik, “Emotions and psychotherapy: A psychoevolutionary perspective,” in *Emotion, psychopathology, and psychotherapy*, pp. 3–41, Elsevier, 1990.

[77] A. Ortony and T. J. Turner, “What’s basic about basic emotions?,” *Psychological review*, vol. 97, no. 3, p. 315, 1990.

[78] S. M. Mohammad and P. D. Turney, “Crowdsourcing a word–emotion association lexicon,” *Computational Intelligence*, vol. 29, no. 3, pp. 436–465, 2013.

[79] S. M. Mohammad and P. D. Turney, “Emotions evoked by common words and phrases: Using mechanical turk to create an emotion lexicon,” in *Proceedings of the NAACL HLT 2010 workshop on computational approaches to analysis and generation of emotion in text*, pp. 26–34, Association for Computational Linguistics, 2010.

[80] E. Santhanam, B. Lynch, and J. Jones, “Making sense of student feedback using text analysis–adapting and expanding a common lexicon,” *Quality Assurance in Education*, vol. 26, no. 1, pp. 60–69, 2018.

[81] L. Alderman, S. Towers, and S. Bannah, “Student feedback systems in higher education: A focused literature review and environmental scan,” *Quality in Higher Education*, vol. 18, no. 3, pp. 261–280, 2012.

[82] L. Harvey, “The nexus of feedback and improvement,” in *Student Feedback*, pp. 3–26, Elsevier, 2011.

[83] S. Palmer, “Student evaluation of teaching: keeping in touch with reality,” *Quality in higher education*, vol. 18, no. 3, pp. 297–311, 2012.

[84] V. Svenja, K. David, K. Eckhard, and B. Sonja, *TALIS teaching practices and pedagogical Innovations evidence from TALIS: Evidence from TALIS*. OECD Publishing, 2012.

[85] P. Serdyukov, “Innovation in education: what works, what doesnt́, and what to do about it?,” *Journal of Research in Innovative Teaching & Learning*, vol. 10, no. 1, pp. 4–33, 2017.

[86] M. B. Miles, *Innovation in education*. Bureau of Publication, Teachers College, Columbia University, 1964.

[87] C. Latulipe, N. B. Long, and C. E. Seminario, “Structuring flipped classes with lightweight teams and gamification,” in *Proceedings of the 46th ACM Technical Symposium on Computer Science Education*, pp. 392–397, ACM, 2015.

[88] S. MacNeil, C. Latulipe, B. Long, and A. Yadav, “Exploring lightweight teams in a distributed learning environment,” in *Proceedings of the 47th ACM Technical Symposium on Computing Science Education*, pp. 193–198, ACM, 2016.

[89] M. L. Maher, C. Latulipe, H. Lipford, and A. Rorrer, "Flipped classroom strategies for cs education," in *Proceedings of the 46th ACM Technical Symposium on Computer Science Education*, pp. 218–223, ACM, 2015.

[90] Y. Kim, H. Lee, and K. Jung, "Attnconvnet at semeval-2018 task 1: attention-based convolutional neural networks for multi-label emotion classification," *arXiv preprint arXiv:1804.00831*, 2018.

[91] A. Severyn and A. Moschitti, "Twitter sentiment analysis with deep convolutional neural networks," in *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 959–962, ACM, 2015.

[92] S. Lai, L. Xu, K. Liu, and J. Zhao, "Recurrent convolutional neural networks for text classification," in *Twenty-ninth AAAI conference on artificial intelligence*, 2015.

[93] M. K. Danthala and D. Ghosh, "Bigdata analysis: Streaming twitter data with apache hadoop and visualizing using biginsights," *International Journal of Engineering Research & Technology*, vol. 4, no. 05, 2015.

[94] M. Zaharia, R. S. Xin, P. Wendell, T. Das, M. Armbrust, A. Dave, X. Meng, J. Rosen, S. Venkataraman, M. J. Franklin, *et al.*, "Apache spark: a unified engine for big data processing," *Communications of the ACM*, vol. 59, no. 11, pp. 56–65, 2016.

[95] B. Raman and T. R. Ioerger, "Instance-based filter for feature selection," *Journal of Machine Learning Research*, vol. 1, no. 3, pp. 1–23, 2002.

[96] L. C. Molina, L. Belanche, and À. Nebot, "Feature selection algorithms: A survey and experimental evaluation," in *2002 IEEE International Conference on Data Mining, 2002. Proceedings.*, pp. 306–313, IEEE, 2002.

[97] Y. Xie, "An attribute reduction method based on boolean matrix," in *2008 International Conference on Computer Science and Software Engineering*, vol. 1, pp. 423–426, IEEE, 2008.

[98] W. Shu and H. Shen, "Incremental feature selection based on rough set in dynamic incomplete data," *Pattern Recognition*, vol. 47, no. 12, pp. 3890–3906, 2014.

[99] Z. Pawlak, "Rough set theory and its applications to data analysis," *Cybernetics & Systems*, vol. 29, no. 7, pp. 661–688, 1998.

[100] S. K. Pal and A. Skowron, *Rough-fuzzy hybridization: A new trend in decision making.* Springer-Verlag, 1999.

[101] A. Skowron, "Extracting laws from decision tables: a rough set approach," *Computational Intelligence*, vol. 11, no. 2, pp. 371–388, 1995.

[102] A. Balahur, "Sentiment analysis in social media texts," in *Proceedings of the 4th workshop on computational approaches to subjectivity, sentiment and social media analysis*, pp. 120–128, 2013.

[103] A. Agarwal, B. Xie, I. Vovsha, O. Rambow, and R. Passonneau, "Sentiment analysis of twitter data," in *Proceedings of the workshop on languages in social media*, pp. 30–38, Association for Computational Linguistics, 2011.

[104] A. Chellal, M. Boughanem, and B. Dousset, "Multi-criterion real time tweet summarization based upon adaptive threshold," in *Proceedings of 2016 IEEE/WIC/ACM International Conference on Web Intelligence (WI)*, pp. 264–271, IEEE, 2016.

[105] Y. Xu, D. Zhou, and S. Lawless, "Inferring your expertise from twitter: Integrating sentiment and topic relatedness," in *Proceedings of 2016 IEEE/WIC/ACM International Conference on Web Intelligence (WI)*, pp. 121–128, IEEE, 2016.

[106] F. Bravo-Marquez, E. Frank, and B. Pfahringer, "From opinion lexicons to sentiment classification of tweets and vice versa: a transfer learning approach," in *Proceedings of 2016 IEEE/WIC/ACM International Conference on Web Intelligence (WI)*, pp. 145–152, IEEE, 2016.

[107] M. Al-Ayyoub, S. B. Essa, and I. Alsmadi, "Lexicon-based sentiment analysis of arabic tweets," *International Journal of Social Network Mining*, vol. 2, no. 2, pp. 101–114, 2015.

[108] A. Neviarouskaya, H. Prendinger, and M. Ishizuka, "Affect analysis model: novel rule-based approach to affect sensing from text," *Natural Language Engineering*, vol. 17, no. 1, pp. 95–135, 2011.

[109] D. T. Ho and T. H. Cao, "A high-order hidden markov model for emotion detection from textual data," in *Pacific Rim Knowledge Acquisition Workshop*, pp. 94–105, Springer, 2012.

[110] G. Mishne *et al.*, "Experiments with mood classification in blog posts," in *Proceedings of ACM SIGIR 2005 workshop on stylistic analysis of text for information access*, vol. 19, pp. 321–327, 2005.

[111] C. Strapparava and R. Mihalcea, "Learning to identify emotions in text," in *Proceedings of the 2008 ACM symposium on Applied computing*, pp. 1556–1560, ACM, 2008.

[112] P. Katz, M. Singleton, and R. Wicentowski, "Swat-mp: the semeval-2007 systems for task 5 and task 14," in *Proceedings of the 4th international workshop on semantic evaluations*, pp. 308–313, Association for Computational Linguistics, 2007.

[113] F.-R. Chaumartin, "Upar7: A knowledge-based system for headline sentiment tagging," in *Proceedings of the 4th International Workshop on Semantic Evaluations*, pp. 422–425, Association for Computational Linguistics, 2007.

[114] Z. Kozareva, B. Navarro, S. Vázquez, and A. Montoyo, "Ua-zbsa: a headline emotion classification through web information," in *Proceedings of the 4th international workshop on semantic evaluations*, pp. 334–337, Association for Computational Linguistics, 2007.

[115] R. E. Schapire and Y. Singer, "Boostexter: A boosting-based system for text categorization," *Machine learning*, vol. 39, no. 2-3, pp. 135–168, 2000.

[116] T. Danisman and A. Alpkocak, "Feeler: Emotion classification of text using vector space model," in *Proceedings of AISB 2008 Convention Communication, Interaction and Social Intelligence*, vol. 1, p. 53, 2008.

[117] J. T. Hancock, C. Landrigan, and C. Silver, "Expressing emotion in text-based communication," in *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 929–932, ACM, 2007.

[118] J. B. Walther, "Social information processing theory," *Engaging theories in interpersonal communication: Multiple perspectives*, pp. 391–404, 2008.

[119] S. M. Kim, A. Valitutti, and R. A. Calvo, "Evaluation of unsupervised emotion models to textual affect recognition," in *Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, pp. 62–70, Association for Computational Linguistics, 2010.

[120] C. Strapparava and R. Mihalcea, "Semeval-2007 task 14: Affective text," in *Proceedings of the 4th international workshop on semantic evaluations*, pp. 70–74, Association for Computational Linguistics, 2007.

[121] K. R. Scherer and H. G. Wallbott, "Evidence for universality and cultural variation of differential emotion response patterning.," *Journal of personality and social psychology*, vol. 66, no. 2, p. 310, 1994.

[122] E. C. O. Alm, *Affect in* Text and Speech.* University of Illinois at Urbana-Champaign, 2008.

[123] E. C.-C. Kao, C.-C. Liu, T.-H. Yang, C.-T. Hsieh, and V.-W. Soo, "Towards text-based emotion detection a survey and possible improvements," in *ICIME'09. International Conference on Information Management and Engineering*, pp. 70–74, IEEE, 2009.

[124] M. C. Jain and V. Kulkarni, "Texemo: Conveying emotion from text-the study," *International Journal of Computer Applications*, vol. 86, no. 4, pp. 43–49, 2014.

[125] J. Lei, Y. Rao, Q. Li, X. Quan, and L. Wenyin, "Towards building a social emotion detection system for online news," *Future Generation Computer Systems*, vol. 37, pp. 438–448, 2014.

[126] S. Bao, S. Xu, L. Zhang, R. Yan, Z. Su, D. Han, and Y. Yu, "Joint emotion-topic modeling for social affective text mining," in *Ninth IEEE International Conference on Data Mining, ICDM'09.*, pp. 699–704, IEEE, 2009.

[127] S. Bao, S. Xu, L. Zhang, R. Yan, Z. Su, D. Han, and Y. Yu, "Mining social emotions from affective text," *IEEE transactions on knowledge and data engineering*, vol. 24, no. 9, pp. 1658–1670, 2012.

[128] G. Mishne, M. De Rijke, *et al.*, "Capturing global mood levels using blog posts.," in *AAAI spring symposium: computational approaches to analyzing weblogs*, vol. 6, pp. 145–152, 2006.

[129] S. M. Mohammad, "# emotional tweets," in *Proceedings of the First Joint Conference on Lexical and Computational Semantics-Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation*, pp. 246–255, Association for Computational Linguistics, 2012.

[130] K. Roberts, M. A. Roach, J. Johnson, J. Guthrie, and S. M. Harabagiu, "Empatweet: Annotating and detecting emotions on twitter.," in *Proceedings of Eighth International Conference on Language Resources and Evaluation (LREC)*, vol. 12, pp. 3806–3813, Citeseer, 2012.

[131] M. Purver and S. Battersby, "Experimenting with distant supervision for emotion classification," in *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 482–491, Association for Computational Linguistics, 2012.

[132] R.-A. Turcu, S. M. Amarandei, I.-A. Flescan-Lovin-Arseni, D. Gifu, and D. Trandabat, "Emointens tracker at semeval-2018 task 1: Emotional intensity levels in# tweets," in *Proceedings of The 12th International Workshop on Semantic Evaluation*, pp. 177–180, 2018.

[133] M. Yassine and H. Hajj, "A framework for emotion mining from text in online social networks," in *2010 IEEE International Conference on Data Mining Workshops*, pp. 1136–1142, IEEE, 2010.

[134] J. Sun, G. Wang, X. Cheng, and Y. Fu, "Mining affective text to improve social media item recommendation," *Information Processing & Management*, vol. 51, no. 4, pp. 444–457, 2015.

[135] Z. Jianqiang, G. Xiaolin, and Z. Xuejun, "Deep convolution neural networks for twitter sentiment analysis," *IEEE Access*, vol. 6, pp. 23253–23260, 2018.

[136] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 1532–1543, 2014.

[137] F. Krebs, B. Lubascher, T. Moers, P. Schaap, and G. Spanakis, "Social emotion mining techniques for facebook posts reaction prediction," *arXiv preprint arXiv:1712.03249*, 2017.

[138] E. Lakomkin, C. Bothe, and S. Wermter, "Gradascent at emoint-2017: character-and word-level recurrent neural network models for tweet emotion intensity detection," *arXiv preprint arXiv:1803.11509*, 2018.

[139] B. Jagtap and V. Dhotre, "Svm and hmm based hybrid approach of sentiment analysis for teacher feedback assessment," *International Journal of Emerging Trends & Technology in Computer Science (IJETTCS)*, vol. 3, no. 3, pp. 229–232, 2014.

[140] Q. Rajput, S. Haider, and S. Ghani, "Lexicon-based sentiment analysis of teachers evaluation," *Applied Computational Intelligence and Soft Computing*, vol. 2016, pp. 1–12, 2016.

[141] J. Wiebe, T. Wilson, and C. Cardie, "Annotating expressions of opinions and emotions in language," *Language resources and evaluation*, vol. 39, no. 2, pp. 165–210, 2005.

[142] J. Daniel, "Making sense of moocs: Musings in a maze of myth, paradox and possibility," *Journal of interactive Media in education*, vol. 2012, no. 3, 2012.

[143] R. Rivard, "Measuring the mooc dropout rate," *Inside Higher Ed*, vol. 8, p. 2013, 2013.

[144] M. Wen, D. Yang, and C. Rose, "Sentiment analysis in mooc discussion forums: What does it tell us?," in *Educational data mining 2014*, Citeseer, 2014.

[145] C. Tucker, B. K. Pursel, and A. Divinsky, "Mining student-generated textual data in moocs and quantifying their effects on student performance and learning outcomes," *The ASEE Computers in Education (CoED) Journal*, vol. 5, no. 4, p. 84, 2014.

[146] M. Nichols, "A theory for elearning," *Educational technology & society*, vol. 6, no. 2, pp. 1–10, 2003.

[147] J. Cross, "An informal history of elearning. on the horizon," *2004. 12 (3). pp. 103*, vol. 110, 2004.

[148] Z. Kechaou, M. B. Ammar, and A. M. Alimi, "Improving e-learning with sentiment analysis of users' opinions," in *2011 IEEE global engineering education conference (EDUCON)*, pp. 1032–1038, IEEE, 2011.

[149] J. M. Martín, A. Ortigosa, and R. M. Carro, "Sentbuk: Sentiment analysis for e-learning environments," in *2012 International Symposium on Computers in Education (SIIE)*, pp. 1–6, IEEE, 2012.

[150] D. Song, H. Lin, and Z. Yang, "Opinion mining in e-learning system," in *2007 IFIP International Conference on Network and Parallel Computing Workshops (NPC 2007)*, pp. 788–792, IEEE, 2007.

[151] F. Colace, M. De Santo, and L. Greco, "Safe: a sentiment analysis framework for e-learning," *International Journal of Emerging Technologies in Learning (iJET)*, vol. 9, no. 6, pp. 37–41, 2014.

[152] H. H. Binali, C. Wu, and V. Potdar, "A new significant area: Emotion detection in e-learning using opinion mining techniques," in *2009 3rd IEEE International Conference on Digital Ecosystems and Technologies*, pp. 259–264, IEEE, 2009.

[153] N. Altrabsheh, M. Cocea, and S. Fallahkhair, "Learning sentiment from students' feedback for real-time interventions in classrooms," in *Adaptive and Intelligent Systems* (A. Bouchachia, ed.), (Cham), pp. 40–49, Springer International Publishing, 2014.

[154] C. K. Leong, Y. H. Lee, and W. K. Mak, "Mining sentiments in sms texts for teaching evaluation," *Expert Syst. Appl.*, vol. 39, pp. 2584–2589, Feb. 2012.

[155] V. Dhanalakshmi, D. Bino, and A. M. Saravanan, "Opinion mining from student feedback data using supervised learning algorithms," in *2016 3rd MEC International Conference on Big Data and Smart City (ICBDSC)*, pp. 1–5, March 2016.

[156] V. Stoyanov, C. Cardie, and J. Wiebe, "Multi-perspective question answering using the opqa corpus," in *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pp. 923–930, Association for Computational Linguistics, 2005.

[157] B. Guo, R. Zhang, G. Xu, C. Shi, and L. Yang, "Predicting students performance in educational data mining," in *2015 International Symposium on Educational Technology (ISET)*, pp. 125–128, IEEE, 2015.

[158] M. F. Musso, E. Kyndt, E. C. Cascallar, and F. Dochy, "Predicting general academic performance and identifying the differential contribution of participating variables using artificial neural networks.," *Frontline Learning Research*, vol. 1, no. 1, pp. 42–71, 2013.

[159] X. Wei, H. Lin, L. Yang, and Y. Yu, "A convolution-lstm-based deep neural network for cross-domain mooc forum post classification," *Information*, vol. 8, no. 3, p. 92, 2017.

[160] A. Agrawal, J. Venkatraman, S. Leonard, and A. Paepcke, "Youedu: addressing confusion in mooc discussion forums by recommending instructional video clips," 2015.

[161] C. S. Galbraith, G. B. Merrill, and D. M. Kline, "Are student evaluations of teaching effectiveness valid for measuring student learning outcomes in business related classes? a neural network and bayesian analyses," *Research in Higher Education*, vol. 53, no. 3, pp. 353–374, 2012.

[162] B. Bakhshinategh, O. R. Zaiane, S. ElAtia, and D. Ipperciel, "Educational data mining applications and tasks: A survey of the last 10 years," *Education and Information Technologies*, vol. 23, no. 1, pp. 537–553, 2018.

[163] L. D. Miller, L.-K. Soh, A. Samal, K. Kupzyk, and G. Nugent, "A comparison of educational statistics and data mining approaches to identify characteristics that impact online learning.," *Journal of Educational Data Mining*, vol. 7, no. 3, pp. 117–150, 2015.

[164] M. Cocea and S. Weibelzahl, "Can log files analysis estimate learners' level of motivation?," in *LWA*, pp. 32–35, University of Hildesheim, Institute of Computer Science, 2006.

[165] J. Bravo and A. Ortigosa, "Detecting symptoms of low performance using production rules.," *International working group on educational data mining*, 2009.

[166] L. Cao, "Actionable knowledge discovery and delivery," in *Metasynthetic computing and engineering of complex systems*, pp. 287–312, Springer, 2015.

[167] J. Barrett, E. Caruthers, K. German, E. Hamby, R. Lofthus, S. Srinivas, and E. Ells, "From data to actionable knowledge: a collaborative effort with educators," in *Proceedings of KDD*, 2011.

[168] S. Greco, B. Matarazzo, N. Pappalardo, and R. Slowinski*, "Measuring expected effects of interventions based on decision rules," *Journal of Experimental & Theoretical Artificial Intelligence*, vol. 17, no. 1-2, pp. 103–118, 2005.

[169] A. Bagavathi and A. A. Tzacheva, "Rule based systems in distributed environment: Survey," in *Proceedings of International Conference on Cloud Computing and Applications (CCA17), 3rd World Congress on Electrical Engineering and Computer Systems and Science (EECSS'17)*, pp. 1–17, 2017.

[170] J. W. Grzymala-Busse, "A new version of the rule induction system lers," *Fundamenta Informaticae*, vol. 31, no. 1, pp. 27–39, 1997.

[171] A. Dardzinska and Z. W. Ras, "On rules discovery from incomplete information systems," in *the Proceedings of ICDM'03 Workshop on Foundations and New Directions of Data Mining, (Eds: T.Y. Lin, X. Hu, S. Ohsuga, C. Liau), Melbourne, Florida, IEEE Computer Society*, pp. 31–35, 2003.

[172] A. Dardzinska and Z. W. Ras, "Cooperative discovery of interesting action rules," in *the Proceedings of FQAS 2006 Conference, Milano, Italy, (Eds. H.L. Larsen et al.), LNAI*, vol. 4027, pp. 489–497, Springer, 2006.

[173] R. Agrawal, R. Srikant, *et al.*, "Fast algorithms for mining association rules," in *Proc. 20th int. conf. very large data bases, VLDB*, vol. 1215, pp. 487–499, 1994.

[174] L.-S. Tsay and Z. W. Ras, "Action rules discovery: system dear2, method and experiments," *in the Special Issue on Knowledge Discovery, Journal of Experimental & Theoretical Artificial Intelligence*, vol. 17, no. 1-2, pp. 119–128, 2005.

[175] A. Dardzinska and Z. W. Ras, "Extracting rules from incomplete decision systems: System erid," in *Foundations and Novel Approaches in Data Mining, (Eds. T.Y. Lin, S. Ohsuga, C.J. Liau, X. Hu), Studies in Computational Intelligence*, vol. 9, pp. 143–154, Springer, 2006.

[176] Z. W. Ras, K. A. Tarnowska, J. Kuang, L. Daniel, and D. Fowler, "User friendly nps-based recommender system for driving business revenue," in *Proceedings of 2017 International Joint Conference on Rough Sets (IJCRS'17), LNCS*, vol. 10313, pp. 34–48, Springer, 2017.

[177] M. Hu and B. Liu, "Mining and summarizing customer reviews," in *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 168–177, ACM, 2004.

[178] K. A. Tarnowska, Z. W. Ras, L. Daniel, and D. Fowler, "Visual analysis of relevant features in customer loyalty improvement recommendation," in *Advances in Feature Selection for Data and Pattern Recognition, Intelligent Systems Reference Library*, vol. 138, pp. 269–293, Springer, 2018.

[179] A. A. Tzacheva, A. Bagavathi, and L. Ayila, "Discovery of action rules at lowest cost in spark," in *Data Mining Workshops (ICDMW), 2017 IEEE International Conference on*, pp. 87–94, IEEE, 2017.

[180] R. Agrawal and J. C. Shafer, "Parallel mining of association rules," *IEEE Transactions on Knowledge & Data Engineering*, no. 6, pp. 962–969, 1996.

[181] M. J. Zaki, "Parallel and distributed association mining: A survey," *IEEE concurrency*, vol. 7, no. 4, pp. 14–25, 1999.

[182] T. Shintani and M. Kitsuregawa, "Hash based parallel algorithms for mining association rules," in *Fourth International Conference on Parallel and Distributed Information Systems*, pp. 19–30, IEEE, 1996.

[183] E.-H. Han, G. Karypis, and V. Kumar, *Scalable parallel data mining for association rules*, vol. 26. ACM, 1997.

[184] A. Bagavathi, P. Mummoju, K. Tarnowska, A. A. Tzacheva, and Z. W. Ras, "Sargs method for distributed actionable pattern mining using spark," in *2017 IEEE International Conference on Big Data (Big Data)*, pp. 4272–4281, IEEE, 2017.

[185] A. Bagavathi, A. Tripathi, A. A. Tzacheva, and Z. W. Ras, "Actionable pattern mining-a scalable data distribution method based on information granules," in *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pp. 32–39, IEEE, 2018.

[186] H. AbouEisha, "Finding optimal exact reducts," in *Proceedings of the International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management-Volume 1*, pp. 149–153, SCITEPRESS-Science and Technology Publications, Lda, 2014.

[187] R. Wang, D. Miao, and G. Hu, "Discernibility matrix based algorithm for reduction of attributes," in *Proceedings of the 2006 IEEE/WIC/ACM international conference on Web Intelligence and Intelligent Agent Technology*, pp. 477–480, IEEE Computer Society, 2006.

[188] Q. A. Al-Radaideh, M. N. Sulaiman, M. H. Selamat, and H. Ibrahim, "A heuristic reduct computation approach by attributes weighting for rough set based classification," *Journal of Computer Sciences (Special Issue)*, vol. 41, p. 47, 2005.

[189] F. Li, Z. Zhang, and C. Jin, "Feature selection with partition differentiation entropy for large-scale data sets," *Information Sciences*, vol. 329, pp. 690–700, 2016.

[190] Y. Gao, X. Chen, X. Yang, and P. Wang, "Neighborhood attribute reduction: A multicriterion strategy based on sample selection," *Information*, vol. 9, no. 11, p. 282, 2018.

[191] R. Vashist, "An algorithm for finding the reduct and core of the consistent dataset," in *2015 International Conference on Computational Intelligence and Communication Networks (CICN)*, pp. 741–745, IEEE, 2015.

[192] M. Korzen and S. Jaroszewicz, "Finding reducts without building the discernibility matrix," in *5th International Conference on Intelligent Systems Design and Applications (ISDA'05)*, pp. 450–455, IEEE, 2005.

[193] Z. Pawlak, "Rough sets," *International journal of computer & information sciences*, vol. 11, no. 5, pp. 341–356, 1982.

[194] J. Watada, S. C. Tan, Y. Matsumoto, and P. Vasant, "Rough set-based text mining from a large data repository of experts diagnoses for power systems," in *International Conference on Intelligent Decision Technologies*, pp. 136–144, Springer, 2017.

[195] G. Jothi *et al.*, "Hybrid tolerance rough set–firefly based supervised feature selection for mri brain tumor image classification," *Applied Soft Computing*, vol. 46, pp. 639–651, 2016.

[196] L. Tang and J. Wang, "Predicting enzyme class with rough sets," in *2016 9th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, pp. 1832–1836, IEEE, 2016.

[197] F. E. Tay and L. Shen, "Economic and financial prediction using rough sets model," *European Journal of Operational Research*, vol. 141, no. 3, pp. 641–659, 2002.

[198] K. Makice, *Twitter API: Up and running: Learn how to build applications with the Twitter API.* " O'Reilly Media, Inc.", 2009.

[199] J. Hedley, "jsoup: Java html parser, url http://jsoup. org," 2009.

[200] S. Bird, E. Klein, and E. Loper, *Natural language processing with Python: analyzing text with the natural language toolkit.* " O'Reilly Media, Inc.", 2009.

[201] S. M. Mohammad and S. Kiritchenko, "Using hashtags to capture fine emotion categories from tweets," *Computational Intelligence*, vol. 31, pp. 301–326, May 2015.

[202] A. Khan, B. Baharudin, L. H. Lee, K. Khan, and U. T. P. Tronoh, "A review of machine learning algorithms for text-documents classification," in *Journal of Advances In Information Technology*, vol. 1, pp. 4–20, 2010.

[203] V. N. Vapnik, "An overview of statistical learning theory," *IEEE transactions on neural networks*, vol. 10, no. 5, pp. 988–999, 1999.

[204] H. Brücher, G. Knolmayer, and M.-A. Mittermayer, "Document classification methods for organizing explicit knowledge," *Proceedings of the Third European Conference on Organizational Knowledge, Learning, and Capabilities, Athens, Greece*, 2002.

[205] D. Meyer and F. T. Wien, "Support vector machines," *R News*, vol. 1, no. 3, pp. 23–26, 2001.

[206] T. Joachims, "Text categorization with support vector machines: Learning with many relevant features," in *European conference on machine learning*, pp. 137–142, Springer, 1998.

[207] S. Dumais, J. Platt, D. Heckerman, and M. Sahami, "Inductive learning algorithms and representations for text categorization," in *Proceedings of the seventh international conference on Information and knowledge management*, pp. 148–155, ACM, 1998.

[208] Y. Yang and X. Liu, "A re-examination of text categorization methods," in *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 42–49, ACM, 1999.

[209] G. Siolas and F. d'Alché Buc, "Support vector machines based on a semantic kernel for text categorization," in *Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks, IJCNN*, vol. 5, pp. 205–209, IEEE, 2000.

[210] K. P. Murphy *et al.*, "Naive bayes classifiers," *University of British Columbia*, vol. 18, 2006.

[211] H. Schütze, C. D. Manning, and P. Raghavan, *Introduction to information retrieval*, vol. 39. Cambridge University Press, 2008.

[212] N. Ansari and E. Hou, *Hopfield Neural Networks*, pp. 27–45. Boston, MA: Springer US, 1997.

[213] Wikipedia contributors, "Gated recurrent unit — Wikipedia, the free encyclopedia," 2020.

[214] J. J. Dai, Y. Wang, X. Qiu, D. Ding, Y. Zhang, Y. Wang, X. Jia, C. L. Zhang, Y. Wan, Z. Li, *et al.*, "Bigdl: A distributed deep learning framework for big data," in *Proceedings of the ACM Symposium on Cloud Computing*, pp. 50–60, ACM, 2019.

[215] M. Hossin, M. Sulaiman, A. Mustapha, N. Mustapha, and R. Rahmat, "A hybrid evaluation metric for optimizing classifier," in *2011 3rd Conference on Data Mining and Optimization (DMO)*, pp. 165–170, IEEE, 2011.

[216] M. Sokolova and G. Lapalme, "A systematic analysis of performance measures for classification tasks," *Information Processing & Management*, vol. 45, no. 4, pp. 427–437, 2009.

[217] R. Caruana and A. Niculescu-Mizil, "Data mining in metric space: an empirical analysis of supervised learning performance criteria," in *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 69–78, ACM, 2004.

[218] A. Bagavathi, V. Rao, and A. A. Tzacheva, "Data distribution method for scalable actionable pattern mining," in *Proceedings of the First International Conference on Data Science, E-learning and Information Systems*, p. 3, ACM, 2018.

[219] A. Skowron and C. Rauszer, "The discernibility matrices and functions in information systems," in *Intelligent decision support*, pp. 331–362, Springer, 1992.

[220] J. G. Bazan, H. S. Nguyen, S. H. Nguyen, P. Synak, and J. Wróblewski, "Rough set algorithms in classification problem," in *Rough set methods and applications*, pp. 49–88, Springer, 2000.

[221] Q. Shen and A. Chouchoulas, "A modular approach to generating fuzzy rules with reduced attributes for the monitoring of complex systems," *Engineering Applications of Artificial Intelligence*, vol. 13, no. 3, pp. 263–278, 2000.

[222] L. S. Riza, A. Janusz, C. Bergmeir, C. Cornelis, F. Herrera, D. Śle, J. M. Benítez, *et al.*, "Implementing algorithms of rough set theory and fuzzy rough set theory in the r package roughsets," *Information Sciences*, vol. 287, pp. 68–89, 2014.

[223] P. Hanrahan and C. Stolte, "Computer systems and methods for visualizing data with generation of marks," Sept. 21 2010. US Patent 7,800,613.

[224] P. Hanrahan, "Tableau software white paper-visual thinking for business intelligence," *Tableau Software, Seattle, WA*, 2003.

[225] S. Ribecca, "The data visualisation catalogue," *URL https://www. datavizcatalogue. com. Accessed*, pp. 06–19, 2018.

[226] P. Law, P. McAndrew, A. Law, K. Warner, J. Runyon, D. Lascu, A. Arundel, and B. Muramatsu, "A bridge to success," 2012.

[227] J. Ranganathan, H. Nikhil, A. Irudayaraj, and A. Tzacheva, "Automatic detection of emotions in twitter data - a scalable decision tree classification method.," pp. 1–10, 2018.

[228] A. Tzacheva, J. Ranganathan, and S. Y. Mylavarapu, "Actionable pattern discovery for tweet emotions," in *International Conference on Applied Human Factors and Ergonomics*, pp. 46–57, Springer, 2019.

[229] I. H. Witten, E. Frank, M. A. Hall, and C. J. Pal, *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, 2016.

[230] X. Meng, J. Bradley, B. Yavuz, E. Sparks, S. Venkataraman, D. Liu, J. Freeman, D. Tsai, M. Amde, S. Owen, *et al.*, "Mllib: Machine learning in apache spark," *The Journal of Machine Learning Research*, vol. 17, no. 1, pp. 1235–1241, 2016.

[231] R. Kohavi, "The power of decision tables," in *European conference on machine learning*, pp. 174–189, Springer, 1995.

[232] R. C. Holte, "Very simple classification rules perform well on most commonly used datasets," *Machine learning*, vol. 11, no. 1, pp. 63–90, 1993.

[233] B. Pang and L. 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*, p. 271, Association for Computational Linguistics, 2004.

[234] B. Pang, L. Lee, and S. 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*, pp. 79–86, Association for Computational Linguistics, 2002.

[235] J. M. Wiebe, R. F. Bruce, and T. P. O'Hara, "Development and use of a gold-standard data set for subjectivity classifications," in *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*, pp. 246–253, Association for Computational Linguistics, 1999.

[236] P. Přibáň, T. Hercig, and L. Lenc, "Uwb at semeval-2018 task 1: Emotion intensity detection in tweets," in *Proceedings of The 12th International Workshop on Semantic Evaluation*, pp. 133–140, 2018.

[237] R. K. Gupta and Y. Yang, "Crystalfeel at semeval-2018 task 1: Understanding and detecting emotion intensity using affective lexicons," in *Proceedings of The 12th International Workshop on Semantic Evaluation*, pp. 256–263, 2018.

[238] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, "Liblinear - a library for large linear classification," 2008. The Weka classifier works with version 1.33 of LIBLINEAR.

[239] F. Heimerl, S. Lohmann, S. Lange, and T. Ertl, "Word cloud explorer: Text analytics based on word clouds," in *2014 47th Hawaii International Conference on System Sciences (HICSS)*, pp. 1833–1842, IEEE, 2014.

[240] A. Gulli and S. Pal, *Deep Learning with Keras.* Packt Publishing Ltd, 2017.

[241] J. Ranganathan, N. Hedge, A. Irudayaraj, and A. A. Tzacheva, "Automatic detection of emotions in twitter data - a scalable decision tree classi-fication method," in *Proceedings of the RevOpID 2018 workshop on Opinion Mining, Summariza-tion and Diversification in 29th ACM Conference on Hypertext and Social Media*, p. 10, ACM, 2018.