Fall 2021

Discovering High-Profit Product Feature Groups by mining High Utility Sequential Patterns from Feature-Based Opinions

Priyanka Motwani
University of Windsor

Discovering High-Profit Product Feature Groups by mining High Utility Sequential Patterns from Feature-Based Opinions

by

Priyanka Motwani

A Thesis
Submitted to the Faculty of Graduate Studies
through the School of Computer Science
in Partial Fulfillment of the Requirements for
the Degree of Master of Science
at the University of Windsor

Windsor, Ontario, Canada

2021

Discovering High-Profit Product Feature Groups by mining High Utility Sequential Patterns from Feature-Based Opinions

by

Priyanka Motwani

APPROVED BY:

M. Belalia

Department of Mathematics and Statistics

H. Fani

School of Computer Science

C. Ezeife, Advisor

School of Computer Science

July 27, 2021

DECLARATION OF ORIGINALITY

I hereby certify that I am the sole author of this thesis and that no part of this thesis has been published or submitted for publication.

I certify that, to the best of my knowledge, my thesis does not infringe upon anyone's copyright nor violate any proprietary rights. Any ideas, techniques, quotations, or any other material from the work of other people included in my thesis, published or otherwise, are fully acknowledged in accordance with the standard referencing practices. Furthermore, to the extent that I have included copyrighted material that surpasses the bounds of fair dealing within the meaning of the Canada Copyright Act, I certify that I have obtained a written permission from the copyright owner(s) to include such material(s) in my thesis and have included copies of such copyright clearances to my appendix.

I declare that this is a true copy of my thesis, including any final revisions, as approved by my thesis committee and the Graduate Studies office and that this thesis has not been submitted for a higher degree to any other University or Institution.

ABSTRACT

Extracting a group of features together instead of a single feature from the mined opinions, such as *"{battery, camera, design} of a smartphone*," may yield higher profit to the manufactures and higher customer satisfaction, and these can be called High Profit Feature Groups (HPFG). The accuracy of Opinion-Feature Extraction can be improved if more complex sequential patterns of customer reviews are learned and included in the user-behavior analysis to obtain relevant frequent feature groups. Existing Opinion-Feature Extraction systems that use Data Mining techniques with some sequences include those referred to in this thesis as Rashid13OFExt, Rana18OFExt, and HPFG19_HU. Rashid13OFExt and Rana18OFExt systems use Sequential Pattern Mining, Association Rule Mining, and Class Sequential Rules to obtain frequent product features and opinion words from reviews. However, these systems do not discover the frequent high profit features considering utility values (internal and external) such as cost, profit, quantity, or other user preferences. HPFG19_HU system uses High Utility Itemset Mining and Aspect-Based Sentiment Analysis to extract High Utility Aspect groups based on feature-opinion sets. It works on transaction databases of itemsets formed using aspects by considering the high utility values (e.g., are more profitable to the seller?) from the extracted frequent patterns from a set of opinion sentences. However, the HPFG19_HU system does not consider the order of occurrences (sequences) of product features formed in customer opinion sentences that help distinguish similar users and identifying more relevant and related high profit product features.

This thesis proposes a system called High Profit Sequential Feature Group based on High Utility Sequences (HPSFG_HUS), which is an extension to the HPFG19_HU system. The proposed system combines Feature-Based Opinion Mining and High Utility Sequential Pattern Mining to extract High Profit Feature Groups from product reviews. The input to the proposed system is the product reviews corpus. The output is the High Profit Sequential Feature Groups in sequence databases that identify sequential patterns in the features extracted from opinions by considering the order of occurrences of features in the review. This method improves on existing system's accuracy in extracting relevant frequent feature groups. The results on retailer's graphs of extracted High Profit Sequential Feature Groups show that the proposed HPSFG_HUS system provides more accurate high feature groups, sales profit, and user satisfaction. Experimental results evaluating execution time, accuracy, precision, and comparison show higher revenue than the tested existing systems.

KEYWORDS: Sentiment Matching, Opinion mining, High Utility Sequential Pattern Mining, Feature Extraction

DEDICATION

To my parents, family, friends, and my love…

ACKNOWLEDGEMENTS

My sincere appreciation goes to my parents Mr. Anilkumar Motwani and Mrs. Mamta Motwani. Your love, faith and words of encouragement gave me the extra energy to see this work through.

I would like to express my sincere gratitude to my advisor Prof. Dr. Christie Ezeife for her continuous support throughout my graduate studies. Thank you for keeping your patience with me and investing your valuable time in reading all my thesis updates, feedbacks, and providing me with continuous feedbacks on my work and financial support through Research Assistantship (R.A.) positions supported from her grants from funding agencies such as NSERC.

Besides my advisor, I would like to thank my thesis committee members: Dr. Mohamed Belalia (my external reader), Dr. Hossein Fani (my internal reader) and Dr. Dima Alhadidi (thesis defense chair) for accepting to be my thesis committee, despite their tight schedules and their insightful comments and encouragement is highly appreciated.

Finally, I would express my appreciation to all my friends and colleagues at the University of Windsor, for their advice, motivation, and support throughout my graduation.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1: INTRODUCTION

"*Do you like this product? Is this worth purchasing? What other people think of this product?"* Customers and manufacturers often rely on opinions. The reviews are text sentences which contain opinions about the features/aspects of the product that serve as an essential piece of information for most of us to ease the decision-making process and significantly influence the public's behavior. The way customers communicate their ideas has changed drastically as a result of the Internet. Web has become a hub of online review websites. The enormous popularity of such social media platforms has led to fundamental changes in how humans share and form their opinions. Social networking websites like Amazon, Twitter, IMDB, Epinions, Facebook, etc., have a significant impact on their users when sharing their thoughts, reviews, ratings, likes, and dislikes about a particular subject, area, or an item.

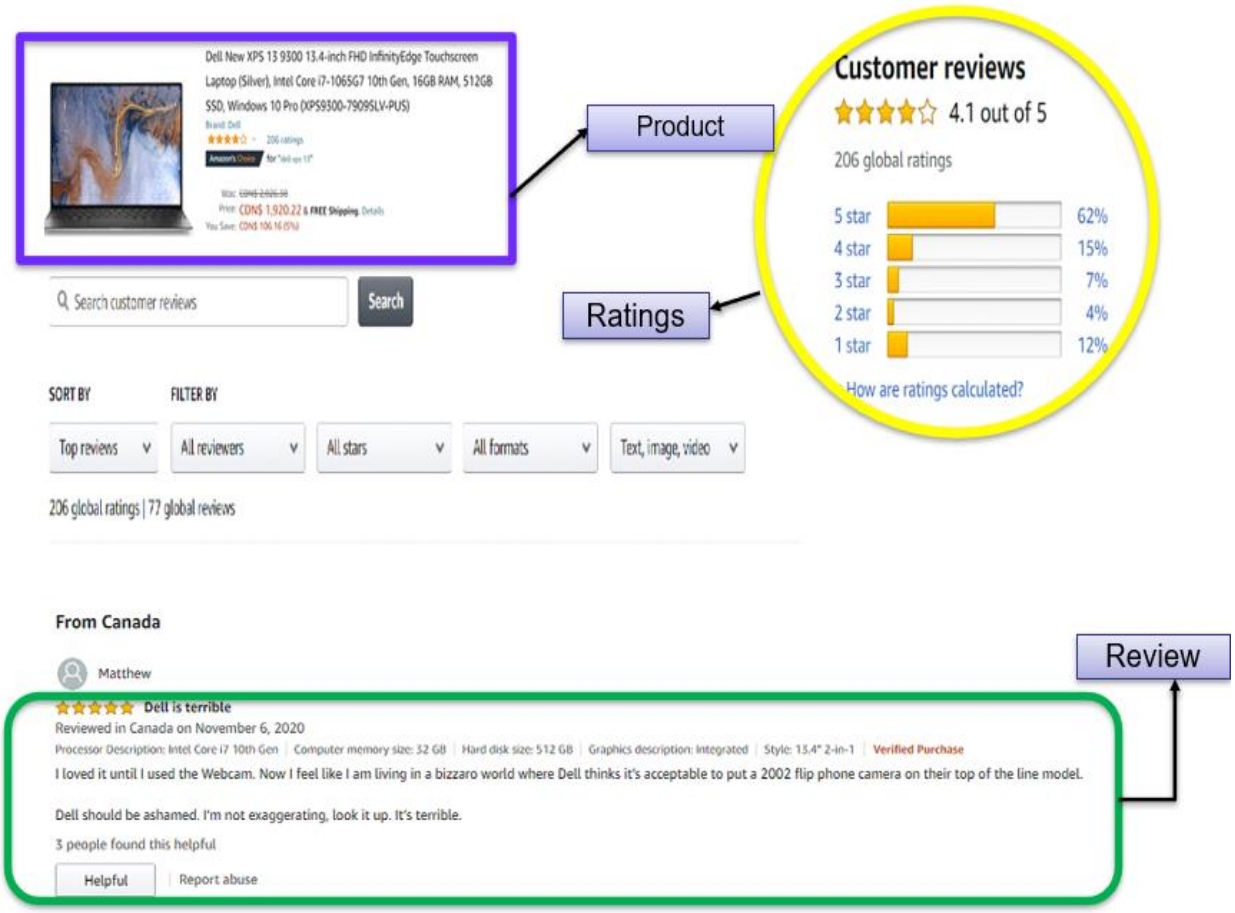


Figure 1: Example of Customer Reviews on Product – "Dell XPS Laptop"

These reviews provide excellent sources of consumer opinions on products that are very beneficial to prospective buyers and producers of products alike. The social networking sites enable the product manufacturers to gather customers' feedback, including opinions and concerns about the products. That can be done by maintaining product pages for consumers to post product reviews. These reviews, written by consumers or product end-users, may somehow reveal their expectations of the products (Li et al., 2014). Therefore, manufacturers can obtain some reflection for the product's redesign according to consumer feedback (Khalid & Helander, 2006). Hence, gathering opinions from User Generated Content (UGC) contributes significantly to the core processes of product design and development, which are critical in the value chain of consumer products. Opinions expressed in social networks play a significant role in influencing public opinion's behavior across areas as diverse as buying products, capturing the "pulse" of stock markets, voting for the president, etc. (Bai, 2011, Eirinaki et al., 2012).

If a product manufacturer wants to know what customers think about his or her product in order to determine whether or not they like it, they may conduct opinion polls, surveys, or form focus groups. These are often costly, time-consuming, and labor-intensive. These circumstances highlight the need for an automated method of gathering opinions — Opinion Mining. As a result, it is worth noting that Opinion Mining has emerged as a promising research area for improving customer experiences and recommendations.

Multiple approaches are proposed, and extensive research is done in the field of Social Network Opinion Mining. People may share their opinions online on the available product, and those reviews may be positive, negative, or neutral. Opinion Mining and Feature Extraction have been the subject of a variety of studies, including Sentiment Classification using machine learning techniques (Pang et al., 2002). In addition, (Turney, 2002) has proposed an unsupervised learning algorithm for classifying reviews, (Dave et al., 2003) proposed Opinion Extraction and Semantic Classification of product reviews. (Hu & Liu 2004) the proposed method to mine the features of the product about which the customers have voiced their opinions and whether those opinions are positive or negative and summarize all the customer reviews of a product, (Ding et al., 2008) proposed a holistic lexicon-based approach to opinion mining. Much study has been done in Opinion Mining and Feature Extraction and Sentiment Analysis, and it is still a vast explorable area.

Data Mining techniques play a significant role in customer behavior analysis and can be incorporated with the extraction of product features and opinions mined from review sentences obtained from online review websites. These approaches include Association Rule Mining(Agarwal & Srikant, 1994), finding associations/relations(probability that particular items are purchased together) between variables in large databases. Sequential Pattern Mining(Aggarwal & Srikant, 1995) discovers frequent patterns and subsequences in the sequence databases. Furthermore, Class Sequential Rules(Hu & Liu, 2006) discovers the sequential rules consisting of a sequence of ordered tokens having class labels. High Utility Itemset Mining(Yao et al., 2004) finds frequent and infrequent patterns from a transaction database of itemsets(Set of items that occur together) based on utility values(internal and external) like cost, profit, user preferences, etc. If the utility of an itemset is more than or equal to a user-specified minimum utility threshold, it is referred to as a High Utility itemset; otherwise, it is referred to as a low utility itemset.

Over here, the utility values (Yin, Zheng & Cao, 2012) can be defined as:

Internal Utility: Every item in the itemsets has an additional value known as internal utility, which is the item's "quantity" (i.e., count). This is a variable value.

External Utility: An external utility is attached to an item, showing the quality (e.g., unit profit) of the item. This is a fixed value.

Utility is a quantitative representation of user preference and can be termed as "*A measure of how 'useful' (i.e., profitable) an itemset is*". It is defined as the sum of the product of its external and internal utility of all items.

Existing algorithms, which include but not limited to (Hu & Liu, 2006), used Opinion Feature Extraction Using Class Sequential Rules (Ghorashi et al., 2012) used Frequent Pattern Mining for feature extraction. Rashid13OFExt(Rashid et al., 2013) compared two important and renowned algorithms of Association Rule Mining(Agarwal & Srikant, 1994) and Sequential Pattern Mining(Aggarwal & Srikant, 1995) for frequent features and opinion words extraction from customers' opinions obtained from a social networking website. (Nurrahmi, Maharani & Saadah ,2016) proposed a system that was able to automatically extract product features and opinions from the reviews using Class Sequential Rule (CSR) method. This method was initially used by (Hu & Liu, 2006) for opinion feature extraction. Rana18OFExt (Rana & Cheah, 2018) performed Feature-

Based Opinion Mining to obtain only the frequent features or important features using Sequential Pattern Mining and Sequential Rules. HPFG19_HU (Demir et al., 2019) extracted Feature-Opinion sets and High Profit Feature Groups using High Utility Itemset Mining and Aspect-Based Sentiment Analysis. Such approaches have been presented that try to incorporate Data Mining approaches mentioned above with Opinion Mining wherein the integration of High Utility Mining has been proven to be a recent enhancement over classical itemset mining.

Utility values can have multiple considerations while incorporating Data Mining approaches with Opinion Mining. Internal utility refers to a variable value like the number(quantity) of items, term frequency(TF), sentiment score, etc. In contrast, external utility refers to a fixed value for a particular product like profit, feature-importance, inverse document frequency (IDF), overall rating, etc.

This thesis aims to combine Social Network Opinion Mining and High Utility Sequential Pattern Mining to see the importance and understand the impact of high utility/opinion utility values on the opinions mined from social media. For this work, a product reviews dataset is considered, obtained from an online review website like Amazon.com, where the customers/reviewers have shared their opinions in the form of comments or ratings on a particular product. Given a set of product reviews, we aim we present an approach named High Profit Sequential Feature Groups based on High Utility Sequences (HPSFG_HUS) to extract High Profit Feature Sets from the product opinions mined from an e-commerce website using High Utility Sequential Patterns.

The input to the system is a set of product reviews extracted from Amazon datasets. The output is the High Profit Sequential Feature Groups obtained because of High Utility Sequential Pattern Mining. Firstly, we have taken a product reviews dataset from Amazon.com for a particular category. Secondly, we try to extract important features/aspects of the product reviews corpus. We calculate sentiment score using a sentiment lexicon like SentiStrength for internal utility, keep external utility constant, and convert the extracted feature words and score into a quantity sequence database. Lastly, by applying USpan algorithm, a High Utility Sequential Pattern Mining algorithm, we obtain High Utility Sequential Patterns to obtain the frequent and top feature sets that will serve as a High Profit Sequential Feature Groups. These feature-sets will have the highest customer satisfaction (or the highest utility) and highest customer dissatisfaction (or the lowest utility), respectively, depending on the whole dataset.

Existing HPFG19_HU System: Our focus is to compare the Feature Groups obtained from our HPSFG_HUS system with the current existing system, HPFG19_HU (Demir et al., 2019), which used High Utility Itemset Mining and Aspect-Based Sentiment Analysis for obtaining High Utility Aspect(HUA) Groups by considering only internal utility values. The existing algorithm considers feature words as itemsets of a transaction and obtains High Profit Aspects. It has considered sentiment score as internal utility values and assumed external utility values as an identical value(= 1) because they say that this value is not available in the review data. According to the authors(Demir et al., 2019), external utility is a domain-dependent value, such as a customer or producer's preference due to its low production cost. The detailed description of this system with an example is shown in ***Section 2.6*****,** ***sub-section 2.6.3***. Hence, we try to improve this system by considering the *Q-sequences* formed in the features to form high utility sequential patterns instead of taking itemsets.

Why are Sequential Patterns important in obtaining High Profit feature groups?

Opinion sentences that correspond to itemsets of aspects in a transaction as done in the HPFG19_HU system(Demir et al., 2019) will provide high profit feature groups, but these may not be necessarily enough if we consider a further step where these feature-sets can be given as an input to Recommender Systems. Also, we do not have any relation between the features or Feature-Sets obtained.

In our proposed HPSFG_HUS system, we compile sequences of features from opinion sentences which stand a better chance of identifying product features because the sequences allow us to know the feature-groups that are related to one another in terms of price, preferences, etc. For example: Feature of a smartphone product: battery. A feature-group might have frequent occurrences of the feature '*battery*' or next upcoming sequences have multiple occurrences of the feature '*battery*'. If price of one feature goes up, there can be a possibility that the importance of the feature-groups containing battery may have higher customer-preference and thus we can say that these aspects are related. Consumers might be attracted to those related features which can help manufacturers in the product redesign. Moreover, after finding sequences of high profit product feature groups/feature terms from a group of users, we can say these users are alike, we can consider those users similar from tweets or opinions, identify those users. This can serve as a better input to the Recommender Systems based on the preferences of similar users. Our approach shows how the

accuracy of the existing technique, and the relevancy of the obtained featured groups is improved. We tend to include external utility values and form sequences in the extracted features and opinions.

The remaining part of Chapter 1 provides a brief description with examples of Social Network Analysis, Opinion Mining, Data Mining background, challenges and approaches for mining social network websites, the problem addressed, and contribution for the thesis.

1.1 Social Network Analysis

Social Network is the chaining of organizations or individuals in the real world. It can be described as a network of social interactions and personal relationships. According to comScore, a marketing research firm that delivers marketing data and services to many of the Internet's major firms, 738 million people use social networking sites on a regular basis – roughly 67 percent of the 1.1 billion people who actively use the Internet throughout the world (Eirinaki et al., 2012). It further claims that when regular users of other social computing activities like blogging are included, the percentage jumps to 76%. Hence, there exists a vast amount of information in social networking sites such as blogs, review sites, social networking applications, etc.

Social networking revolves allows like-minded individuals to be in touch with each other using websites and web-based applications. Facebook, MySpace, Twitter, and LinkedIn are examples of social networking sites. (Include about opinions, reviews, amazon). To summarise it at a higher level, social networking is an area or field where users and customers can interact with each other by posting or sharing content, comments, feedback, messages, photos , videos, etc. on a website or an application.

According to (Barbier et al. 2013) “it is a corporation of variety of social media sites, including social networking (e.g., Facebook, LinkedIn, etc.), blogging (e.g., Huffington Post, Business Insider, Engadget, etc.), micro-blogging (e.g., Twitter, Tumblr, Plurk, etc.), wikis (e.g., Wikipedia, Wikitravel, Wikihow, etc.), social news (e.g., Digg, Slashdot, Reddit, etc.), social bookmarking (e.g., Delicious, StumbleUpon, etc.), media sharing (e.g., Youtube, Flickr, UstreamTV, etc.), opinion, reviews and ratings (e.g., Epinions, Yelp, Amazon, Cnet, etc.), and community Q&A (e.g., Yahoo Answers, WikiAnswers, etc.)”.

1.2 Opinion Mining or Sentiment Analysis

Why Opinions? - Capturing consumers' opinions and gaining knowledge about consumer preferences has long been a major concern for marketing researchers. Opinions help in understanding the thinking of the customers and their expectations which help the manufacturers in releasing the future versions of the product.

Opinion Mining (OM) is defined as processing unstructured data and text data to characterise it into results such as positive, negative, and neutral or good, bad, and average so that we can evaluate any product or item. It is becoming increasingly popular in modern culture, but before the emergence of web 2.0, people could only access information; now, they can also contribute material on the web in the form of comments and reviews. The User Generated Content (UGC) has compelled the organisation to pay attention to the analysis of this content for better visualisation of public's opinion. Hence, with the increasing availability and popularity of opinion-rich resources such as online review sites and personal blogs, social networking websites, etc., new opportunities and challenges emerge as people now can, and do, actively use information technologies for seeking out and understanding opinions of others.

Sentiment Analysis (SA) is the computational analysis of opinions, sentiments, emotions, and attitudes expressed in texts toward a specific entity. The history of the phrase Sentiment Analysis parallels that of "opinion mining" in certain respects. The term "sentiment" used in reference to the automatic analysis of evaluative text and tracking of the predictive judgments. As a result, when wide definitions are used, the terms "Sentiment Analysis" and "Opinion Mining" refer to the same field of study (which itself can be considered a sub-area of subjectivity analysis). Sentiment Analysis (also called opinion mining, review mining or appraisal extraction, attitude analysis) is the task of detecting, extracting, and classifying opinions, sentiments and attitudes concerning different topics, as expressed in textual input. Opinion Mining serves in reaching a variety of objectives, including monitoring public sentiment on political movements, market intelligence, customer satisfaction evaluation, and movie sales prediction and many more. Sentiments, evaluations, and reviews are becoming very much evident due to growing interest in e-commerce, which is also a prominent source of expressing and analyzing opinions. Nowadays, customers on e-commerce site mostly rely on reviews posted by existing customers, and producers and service providers, in turn, analyze customers' opinions to improve the quality and standards of their

products and services. For example, opinions given on e-commerce sites like Amazon, IMDb, epinions.com, etc. can influence the customers' decision in buying products and subscribing services(Ravi, & Ravi, 2015).

So, it can be said that Opinion Mining or Sentiment Analysis is an autonomous text analysis and description method for reviews available on Web(Golande, Kamble, & Waghere, 2016). It is a combination of Natural Language Processing and Text Mining. The main objective of Opinion Mining is Sentiment Classification (i.e., to classify opinion into positive or negative) and obtain a sentiment score corresponding to the opinion word.

Example: Consider following set of reviews from *Amazon.com* for a particular smartphone

UserID	**ReviewID**	**Product Review**
U1	R1	The iphone11 Pro has an amazing batterylife. It has an outstanding camera quality.
U2	R2	It has a horrible voice quality!!!. Not worth of a purchase.
U3	R3	No doubt colour accuracy is good, Touch response is good, but it not that sharp.

Table 1:Product Reviews of 'iphone 11 Pro'

An opinion has following five components:

Opinion Target	iphone 11 Pro
Opinion Polarity	R1: Positive; R2: Negative; R3:Neutral
Features	R1: batterylife, camera quality; R2: voice quality; R3: colour accuracy, touch response
Opinion Words	R1: amazing, outstanding; R2: horrible; R3:good, sharp
Opinion Source	U1, U2, U3

Table 2:Components of Opinion

Hence, to conclude, we can say that Opinion Mining is the problem of recognizing the expressed opinion on a particular subject and determining the polarity of opinion. It is a procedure to extricate information from client assessment, surveys, emotions, and musings(Parashar & Sharma, 2016). It provides a broad view of the sentiments expressed via text and to classify and summarize the opinions, which enable further processing of the data.

Note: *In this thesis, features or aspects and opinions or sentiments will be used interchangeably.*

1.3 Basics of Feature-Based Opinion Mining:

Feature-Based Opinion Mining performs fine-grain analysis by recognizing individual features of an object upon which the user has expressed his/her opinion(Golande, Kamble, & Waghere, 2016). From the point of view of e-commerce, receiving customers' opinions can significantly improve its policies to maximize its sales. Generally, each product includes thousands of opinions, so it is challenging for the consumer to analyze all the reviews. It may also be a very time-consuming job to find feedback on the specific features of a product that is usually desired by a routine consumer. Feature-Based Opinion Mining is useful for feature-level opinion extraction, which forms a comprehensive summary of views that assist clients in decision-making.

Feature Extraction: Features are aspects of the subject of the text. For instance, if the subject of the text is a mobile phone, the possible aspects are screen, battery, price, size, and weight. Depending on the application, the set of aspects may be available or extracted from the text, generally through applying an unsupervised method. This subproblem can be defined as follows. Given a sequence of terms <t_1,……, t_q>that corresponds to a sentence, feature extraction constructs a set of features{f_i, …. , f_n}, for the subject under consideration, where each feature either corresponds to a term in a given sentence or can be inferred from a sentence. Considering the running example, <*battery life, is, long, but, the price, is, high*>the extracted features are {*battery life, price*}.

Sentiment Extraction: This step is about extracting the sentiment terms in a sentence, which uses conventional Sentiment Analysis techniques to detect the sentiment terms. Hence, the sentiment extraction problem can formally define as follows. Given a sequence of terms <t_1,……, t_q> that corresponds to a sentence, sentiment extraction constructs a set of sentiment terms{s_i, …. , s_n}. For example, for the same sentence, extracted sentiments are {*long, high*}.

Feature-Sentiment Matching: Once the features and sentiment terms in each sentence are identified, the next step is to match extracted features and sentiment terms. We can formulate this sub problem as follows. Given sequence of terms T = <t_1,……, t_q> that corresponds to a sentence, a set of features F = {f_1, …. , f_n}, and a set of sentiment terms S ={ s_1, …. , s_k} extracted from T, feature-sentiment matching generates tuples(f_i, s_j, sc), such that $s_j$is the sentiment of feature f_i with sentiment score *sc*. Note that this score can be either positive or negative as in conventional

Sentiment Analysis. The base score is basically associated with the sentiment term, but it may be modified due to enhancer or negators in the sentence. For the running example, the output of the feature-sentiment matching process is *{(batterylife, long, 2), (price, high, −1)}*. If the sentence is changed as "*Battery life is long, but the price is very high"*, then the score of price changes to a more negative value due to enhancer word "*very*".

There are several further challenges in Feature-Based Opinion Mining. For the running example, the sentiment terms *long* and *high* may have different sentiment polarities due to context. Consider the sentence "*Screen resolution is high*.", where the same term *high* has a positive polarity this time. As another challenge, consider the sentence "*Battery life is long, but it is expensive.*". The sentence includes the features *battery life* and *price*. Note that the second feature is not explicitly mentioned, it should be inferred from the sentiment term. If the sentiment term's polarity is context dependent, then the inference gets even more complicated.

1.4 Data Mining

Data Mining has become one of the important aspects since a long time wherein important data is extracted from a huge data. It refers to Knowledge Discovery of Data (KDD) and the process includes (i) data selection – retrieving important information from the data (ii) data pre-processing – this includes data cleaning and removing unwanted and noisy data before processing (iii) data transformation – that transforms the pre-processed data into an appropriate form of data and (iv) pattern evaluation and knowledge interpretation – which identifies interesting patterns. Common Data Mining tasks include classification, clustering, association rule mining, frequent pattern mining, and sequential pattern mining.

Association Rule Mining aims to discover the co-occurrence relationships called associations in a customer transaction database among the attribute values of tuples (Liu & Wang, 2007). A transaction database is a collection of records (transactions) that track what customers have bought at various times. The most well-known use of association rule mining is the study of the market basket using frequent pattern mining (which is to discover frequent itemsets, a group of values/items that have occurred at least as frequently in the database as the given minimum support) algorithm such as Apriori (Agrawal & Srikant, 1994), which aims to discover how items purchased by customers in a supermarket are associated.

Association rule mining aims to discover rules from a given set of items to obtain the simultaneous occurrences of different items. It has been widely used in data mining research where transactions are maintained in a structured database and rules of form x -> y are created where x, y are items in the database and x is not equal to y (Ejieh, Ezeife, & Chaturvedi, 2019). For example, consider the Table 3 below showing a sample transaction history of customers in a grocery store.

Transaction ID	Purchased Items
1	Milk, Bread, Butter
2	Milk, Bread
3	Milk, Butter
4	Butter, Bread, Egg, Tea

Table 3: Customer Transaction Table

A rule that goes Milk -> Bread means customers who purchased Milk also purchased Bread. Some algorithms have been proposed to discover these rules, such as the Apriori algorithm (Agrawal & Srikant, 1994) and Frequent Pattern algorithm (Han et al., 2000). The sets {Milk, Bread}, {Milk, Bread, Butter} are all termed as ***itemsets***.

Apriori algorithm (Aggarwal & Srikant, 1995) finds the set of frequent patterns iteratively by computing the support of each itemset in the candidate set. Using the Apriori algorithm to mine association rules in the table above with minimum support of 50%, the 1-itemset is first found. This consists of the items *MILK, BREAD, EGG, TEA, and BUTTER*. On scanning the database above, *MILK, BREAD and BUTTER* occur in two or more transactions. Because the minimum support is 50% and the number of transactions is 4, their support count fulfills the minimum support, so they form the first large itemset, L1. Therefore, L1 = {*Milk, Bread, Butter}*. Further, the 2-itemset is created by using the apriori-gen join operator. The apriori-gen join of Li with Li joins every itemset k of first Li with every itemset n of second Li where n > k and first (i-1) members of itemsets k and n are the same. Using this example, applying the apriori-gen join to L1 yields *{Milk-Bread, Milk-Butter, Bread-Butter}.* This is the 2-itemset. Since the 3 items meet the minimum support of 50%, they form the second large itemset, L2. Applying the apriori-gen join again to L2 gives {*milk-bread-butter*} which is our 3-itemset. Since the minimum support for *milk-bread-butter* is lesser than the minimum support, the algorithm terminates.

Frequent Patterns are itemsets, subsequences, or substructures that appear in a data set with frequency no less than a user-specified threshold. For example, a set of items, such as milk and bread that appear frequently together in a transaction data set is a frequent itemset.

Frequent Pattern Mining aims to discover how items purchased by customers in a supermarket with a frequency no less than a user-specified threshold. Over here, the threshold is the minimum value that is considered for the percentage of transactions that occur in the dataset for itemsets.

Sequential Pattern Mining discovers frequent subsequences as patterns in a sequence database (Aggarwal & Srikant, 1995). Sequential Pattern Mining is one of the topics that has drawn attention of many researchers because of its high applicability to mine patterns and sequences from databases and web access sequences. Sequential pattern mining is a popular technique that can be applied to trend analysis from a set of long-term event sequence data since a sequential pattern with high frequency can provide the order of events (items) in the pattern in the sequence database. In real-world applications, the transaction time of each transaction is usually recorded in databases. If these transactions can be listed as a time-series data (called sequence data) in their occurring time order, then buying behaviour patterns can be found from the sequence data. Frequent sequential patterns are those patterns that occurred in the database at least as often as the minimum support given.

Support of a set of items defined as the number of tuples or the percentage of the database tuples in the table that contains these set of items. Support (itemset) = number of tuples in the itemset/total number of tuples in the database. A sequence database D store a number of records, where all records are sequences of ordered events, without any time order.

Sequential Pattern Mining using GSP (Generalized Sequential Patterns) Mining algorithm (Srikant & Agrawal, 1996**)**

Given a set of k unique items or events I = $\{i_1, i_2, \ldots\ldots, i_k\}$, the problem of mining sequential patterns can be addressed with GSP algorithm (Srikant & Agrawal, 1996) for the given sequence database D of items I provided the minimum support.

Example: Following table describes retail customer transactions or purchase sequences in a store showing for each customer, collection of store items they purchased every week for one month

SID	Sequences
01	< (Bread, Milk), (Bread, Milk, Sugar), (Milk)>
02	< (Bread), (Bread, Milk, Sugar)>
03	< (Eggs, Milk), (Bread, Milk) >
04	< (Milk, Sugar), (Milk) >

Table 4: Sequence Database of Items

Input: Sequence Database (Table 4), min_sup=2 and candidate set (C1) = {Bread, Milk, Sugar, Eggs}.

Output: Frequent sequential patterns.

Step 1: Find the minimum support of every 1-frequent sequence and remove the candidate sequence which have count less than minimum support.

Bread	*5*
Milk	8
Sugar	3
Eggs	1

Table 5: Support of each item

(L1) = {<Bread: 5>, <Milk:8>, <Sugar: 3>}

Step 2: Form candidate sequence (C_k=2) using L1 $GSPjoin$ L1 and use 1-frequent sequence (L1) to generate larger candidate set 2 and find 2- frequent sequences (L2) by counting the occurrence of 2-sequences in candidate sequence (C2).

(L2) = {<(Bread, Milk) :4>,<(Bread, Sugar) :2>,<(Milk, Bread):2>,<(Milk, Sugar) :3>, <(Sugar, Milk):2>}

Step 3: Repeat candidate generation and pruning process until the result of candidate generation (C_k) and prune (L_k) for finding frequent sequence is an empty set.

1- Frequent Sequences	**2- Frequent Sequences**	**3- Frequent Sequences**
<(Bread)>, <(Milk)>, <(Sugar)>	<(Bread, Milk)>, <(Bread, Sugar) >, <(Milk, Bread)>, <(Milk, Sugar)>, <(Sugar, Milk)>	<(Bread, Milk, Sugar)>

Table 6: Frequent Sequences Table

A sequence S is a frequent sequence or a sequential pattern if and only if sup(s) >= *minsup*.

Limitations:

- Sometimes, frequent patterns may only contribute a small portion of overall profits.
- All items are considered equally important in the sequential pattern mining (same weight is assigned)

- Frequent Sequential Pattern Mining techniques identify many patterns; however, they may not be useful for corporate decision-making because they do not reveal the business value and impact.

1.5 High Utility Sequential Patterns

High Utility Sequential Pattern Mining is a rising topic in the data mining community and extends to Sequential Pattern Mining (Yin et.al, 2013). wherein different constraints like quantity and quality of the sequences are taken into consideration which is called utilities. The "utility" is introduced into pattern mining to mine for patterns of high utility by considering the quality (such as profit) and quantity (such as number of items purchased) of itemsets. The utility framework delivers more useful and actionable knowledge than the standard frequent sequence mining since the utility of a sequence implies business worth and impact. Unlike classic SPM, HUSPM considers that each item is assigned a weight to represent its relative importance (e.g., weight, unit profit, or interestingness), and that each item has non-binary purchasing quantities in a sequential order. A sequence is considered a High Utility Sequential Pattern (HUSP) if its utility exceeds a user-defined minimum utility threshold (count) (Zhang, Lin, Fournier-Viger, & Li, 2017).

The Downward Closure Property of frequent patterns states that any subset of a frequent itemset must be frequent. For example, if {beer, diaper, nuts} is frequent, so is {beer, diaper} i.e., every transaction having {beer, diaper, nuts} also contains {beer, diaper}. This is an important property that must be maintained while mining High Utility Sequential Patters. In the traditional sequential pattern mining algorithms, the downward closure property (also known as Apriori property) (Agrawal et al., 1995) plays a fundamental role for varieties of algorithms designed to search for frequent sequential patterns.

Importance of Utility: The utility is used in pattern mining to find high utility patterns by considering the quality (such as profit) and quantity (such as the number of items purchased) of itemsets. This has resulted in High Utility Pattern Mining (Yao et al., 2004), in which interesting patterns are selected based on minimal utility rather than minimum support. Later sequential pattern mining is introduced in the High Utility Mining. A sequence is a high utility sequence only if its utility value is no less than a user specified minimum utility. Highly Profitable Sequential Patterns are retrieved using the High Utility Pattern Mining approach, which are more informative for retailers in selecting their marketing strategy. First, as with high utility itemset mining, the

downward closure property does not hold in utility-based sequence mining. This clarifies that most of the existing algorithms cannot be directly transferred, from frequent sequential pattern mining to High Utility Sequential Pattern Mining (Yin, Zheng & Cao, 2012). Later with the advent of the sequence weighted utility, the Apriori property issue is resolved as the normal sequence utility does not hold the property, but the weighted sequence utilities follow the Apriori property from which the High Utility Sequential Patterns are generated.

The task of High Utility Sequential Pattern Mining is to extract sequential patterns from a sequential database given the utility measures which gives the importance of each item in the sequence. Formally, a Sequential Database D is defined as follows. Let there be the set I of all items I = $\{i_1, i_2, \ldots\ldots, i_m\}$. A quantitative transaction database D is a set of sequences, denoted as D = $\{S_1, S_2, \ldots\ldots, S_n\}$ where each transaction, $S_q$is a set of items (i.e., $S_q \subseteq$ I), and has a unique identifier q called its SID (Sequence Identifier). Every item i ∈ I is associated with a positive number p(i), which is called its external utility. The external utility of an item is a positive number representing its relative importance to the user. Every item i appearing in a transaction Sq has a positive number q (i, S_q) called its internal utility, which represents quantity of i in sequence S_c.

Preliminary and Key Properties of the Problem of High Utility Sequential Pattern Mining:

Let a sequence *S*, denoted by $\{s_1, s_2, \ldots\ldots, s_r\}$, be an ordered list of patterns, that is, each s_q ($1 \leq q \leq r$) is a pattern *P*, and each pattern appearing in a sequence is called an element of the sequence. A Sequence Database (SDB) contains several transaction sequences (TS), where TS_s: $\{TS_1, TS_2, \ldots\ldots, TS_m\}$. TS_k, ($1 \leq k \leq m$) contains a tuple <SID_k, S_k,>, where SID_k is the Sequence ID, and $S_k$is the sequence of the TS_k. TS_k is said to contain a sequence, *X*, if *X* is a subsequence of S_k.

Sequence ID	Sequence with internal utility	Sequence utility ($)
S1	a(3) {a(2) b(6) d(2)} f(1) a(5) d(1)	130
S2	e(3) {a(2) b(5)} d(1) c(4)	85
S3	{c(1) f(2)} b(3) {d(1) e(4)}	74
S4	a(2) {b(7) d(4) } {a(6) b(3)} e(5)	180
S5	{d(1) f(3)} c(5) g(2)	67
S6	d(2) e(1) {a(7) b(8)} d(3) b(6) e(3)	207

Table 7:A Q-Sequence Database

Item	Profit per unit ($)
a	5
b	7
c	3
d	10
e	6
f	8
g	9

Table 8: An external utility(profit) table

Definition 1. Table 7 shows an example SDB with internal and external utility values. Here, the *internal utility* values represent the quantities of items in sequences, and the *external utility* value of each item represents profit ($) per unit of that item. For example, in *Table 7*, $iu(b, S1) = 6$, and $eu(b) = 7$. However, an item may appear multiple times in a TS. In that case, $iu(ij, S_k)$ is the addition of all the quantities of *ij* in sequence S_k. For example, in *Table 7*,$iu(a, S1) = 10$.

Definition 2. Sequence utility, $su(ij, S_k)$, is the quantitative measure of utility for item *ij* in TS_k, defined by:

$$su(ij, S_k) = iu(ij, S_k) * eu(ij,)$$

Equation 1: Sequence Utility formula for an item

For example, $su(b, S1) = 6{\times}7{=}42$ in Table 7.

Definition 3. A sequence, for example, $X = \{x_1, x_2, \ldots\ldots, x_m\}$, is called an *m*-sequence, where $X \subseteq S_k$, $x_p \subseteq I$, and $1 \leq p \leq m$. To calculate the internal utility of an item, *ij*, in a sequence *X* ($X \subseteq S_k$), we must take only the internal utility of *ij* in *X*. For example, *iu*(*d*, *de*(*ab*), *S*6)=2 (where *X*=*de*(*ab*)). Hence, as with an item, a sequence *X* may have multiple distinct occurrences in TS_k. Accordingly, for sequence utility of *X* in S_k, *su*(*X*, S_k) is defined by:

$$su(X, S_k) = \sum\sum su(ij, X, S_k)\ for\ all\ X\ \in\ S_k\ \forall\ X_{ij} \in X$$

Equation 2: Sequence Utility formula for sequence

However, in the above equation, we refer to only all distinct occurrences of *X*. For example, sequence *de* has two distinct occurrences in *S*6. Hence, *su*(*de*, *S*6) = (2×10+1×6) + (3×10+3×6) = 26+48 = 74 in *Table 7*.

Definition 4. The sequence utility of a transaction is the sum of products of internal (*iu*) and external (*eu*) utilities of each item in a transaction. The sequence utility of TS_k is sum of utility of all the items in the transaction defined by:

$$su(TS_k) = \sum su(ij, S_k)\ for\ ij \in S_k$$

Equation 3: Sequence Utility formula for a transaction (k)

For example, $su(TS_1)$ = *su*(*a*, *S*1)+ *su*(*b*, *S*1)+ *su*(*d*, *S*1)+ *su*(*f*, *S*1) =50+42+30+8=130

Definition 5. The sequence utility of a sequence say X in SDB is the sum of sequence utility of X in all the transactions of SDB. The sequence utility of a sequence *X* in an SDB is defined by:

$$su(X, SDB) = \sum\sum su(X, S_k)\ for\ TS_k\ \in SDB\ and\ X\ is\ a\ subset\ of\ S_k$$

Equation 4: Sequence utility formula for sequence in SDB

For example, *su*(*a*(*bd*)*a*, *SDB*) =*su*(*a*(*bd*)*a*, TS_1) +*su*(*a*(*bd*)*a*, TS_4) =102+129=231 in *Table 7*.

Definition 6. The sequence utility of the whole Sequential Database is the summation of all the transaction utilities in the database. The sequence utility value of an SDB is defined as:

$$su(SDB) = \sum su(TS_k) for\ TS_k\ \in SDB$$

Equation 5: Sequence utility value of SDB

For example, *su*(*SDB*)=743 in *Table 7*.

Definition 7. The minimum sequence utility threshold, δ, is given by the percentage of sequence utility value of the database. In *Table 7*, if δ is 30% or can be expressed as 0.3, then the minimum sequence utility value can be defined as:

$$minSeqUtil = \delta * su(SDB)$$

Equation 6: Formula to compute Minimum Sequence Utility threshold

Hence, in this example, *minSeqUtil* = 0.3×743=223 in *Table 7*.

Definition 8. A sequence *X* is a high utility sequential pattern if *su*(*X*) ≥ *minSeqUtil*. Mining High Utility Sequential Pattern means discovering all the sequences *X* having criteria *su*(*X*) ≥ *minSeqUtil*. For *minSeqUtil*=223, *a*(*bd*)*a* is a High Utility Sequential Pattern as *su*(*a*(*bd*)*a*) = 231. The sequential pattern mining does not satisfy the downward closure property. To maintain the downward closure property in High Utility Sequential Pattern Mining, we use a new measure called *sequence-weighted utility* (*swu*). The *swu* value of a sequence *X* is defined by:

$$swu(X) = \sum su(TS_k)\ for\ X\ is\ a\ subset\ of\ S_k\ and\ TS_k \in\ SDB$$

Equation 7: Formula to compute Sequence weighted utility

Definition 9. *X* is a high *swu* sequence if s*wu*(*X*) ≥ *minSeqUtil*.

High Utility Sequential Pattern: The problem of High Utility Sequential Pattern Mining is defined as follows: A sequence S is a high utility sequence if its utility u(S) is no less than a user-specified

minimum utility threshold minutil set by the user (i.e., u(S) ≥ minutil). Otherwise, S is a low-utility sequence pattern. For a given sequential database and minimum utility threshold, the problem of high utility sequential mining is to enumerate all patterns that have a utility greater than or equal to the user-specified minimum utility threshold. The problem of High Utility Sequential Pattern Mining is challenging because the number of patterns that have a high utility can be huge. Generally, if a database contains n distinct sequences, there can be 2n – 1 possible patterns (excluding the empty set) formed. Thus, the search space's size (the number of possible sequences) can be considerably more.

1.6 Text Mining

The overwhelming amount of information accessible to us due to the growth of the World Wide Web has contributed to a change in focus from mining and extracting meaningful information from structured data sources, such as relational and transactional databases. Knowledge discovery from semi-structured or unstructured data sources such as online news feeds, social media, medical records, email messages and review sites have become a significant focus(Ejieh, Ezeife, & Chaturvedi, 2019). Text mining extracts the relevant information or knowledge, or patterns from different sources available in an unstructured form(Sukanya & Biruntha, 2012). Text mining uses natural language processing (NLP) to interpret and process human language automatically.

One of the most important elements of text mining is document collection. This document collected from any group of text-based documents such as social media reviews and posts, comments, news reports is called as corpus. Text mining systems take corpus as an input. The second most important factor of Text Mining is the representation of words(text). Machines cannot process a raw text, and they break the text into numerical form which is easily readable by the system. The most widely used word representation method is the standard representation of words where words are interpreted as vectors, the length of vectors is the number of documents in the corpus, and the vector values correspond to the frequency of the occurrence of each word in the text (Ejieh, Ezeife, & Chaturvedi, 2019).

1.7 Mining the social network websites

Online media and Social Networking Sites (SNS) are used to share and describe public experiences in product reviews, blogs, and discussion forums. Collectively, these media contain highly

unstructured data combining text, images, animations, and videos useful in making the public aware of various issues(Ravi & Ravi, 2015).

Some of the research areas for Social Networks include community detection and analysis, Opinion Mining and Sentiment Analysis, social recommendation, influence maximization, and modeling, information diffusion and provenance and privacy, security, and trust (Gundecha & Liu, 2012). Unlike community detection and analysis (Mumu & Ezeife, 2014), we are not interested in discovering the social networks formed by a specific set of people, unlike influence modeling (Ahmed & Ezeife, 2013), we are not interested in the links and "friends" formed in social media sites and how they influence one another, and unlike information diffusion and provenance (Barbier et al., 2013), we are not interested in the origin of the user-generated content, social media.

In this thesis, we will majorly focus on Opinion Mining, Opinion-Feature Extraction, and obtaining Feature Groups. We will work on collecting the product reviews from social media website like Amazon.com, which is also an e-commerce website and how we can mine features of the products to know the opinions expressed on each of the aspects regardless of who(user/reviewer) posted them. These reviews dataset collected from Amazon.com will serve as our document collection(corpus).

Product Reviews: According to (Barthwal, 2020), customers' opinions or feedback on a product are referred to as product reviews. Many online businesses have a review section on their website where customers can rate and review the products they bought. A product review assists other consumers in gaining a comprehensive understanding of the product prior to purchase. They can read the reviews to make up their minds about whether or not the product is worth buying.

According to (Rajeev & Rekha, 2015), customers look at the following features while deciding:

- Number of star ratings
- Positive and Negative tone of reviews
- Various features of products (e.g., Battery life, RAM, screen resolution with respect to mobile phones) discussed in reviews
- Helpfulness factor of reviews
- Authenticity of reviews
- Number and age of reviews

Mining of the customer reviews will involve automating the extraction of reviews and ratings. Besides, cleaning the data, quantitatively analyzing the ratings, qualitatively analyzing the reviews through Opinion Mining or Sentiment Analysis, and arriving at a specific product score will help customers differentiate several products based on customer reviews.

Figure 2: An online sample review about a camera (Moghaddam & Ester 2012)

Figure 2 shows a product review about a camera from the product reviews website. Some product features (below the product rating), like *battery life,* have been explicitly mentioned, along with the camera's pros and cons. Apart from the explicit aspects which can be extracted without any challenges, the following information is not yet extracted, which pose fundamental challenges while mining features from opinions :

1. **Implicit Features:** An implicit feature is a feature that is not explicitly mentioned in the sentence, and it can be implied(Ghorashi et al., 2012). Users do not use any specific word to express their views(Rana & Cheah, 2018). Semantic understanding is required to find such implicit features. Such features often occur less than the explicit ones(Hu & Liu, 2004).

   Review 1: This camera is not easy to carry.

   "Weight" is an implicit feature of the camera, which is implied from the sentence.

   Review 2: While light, it will not easily fit in pockets.

   "size" is an implicit feature of camera, but the term is not explicitly mentioned in sentence.

2. **Frequent Features:** A feature f is frequent if it appears in a majority of the review sentences. These features are the ones that people are more interested in or talk about more. The term "frequent features" refers to features that are frequently mentioned by reviewers. Frequent features are detected from sentences with at least one feature word and its opinion(Rashid et al., 2013). Frequent features are also called explicit features. *Example:* The feature *battery life* and *durability* might appear in many review sentences, and such terms can be identified as frequent features.
3. **Infrequent Features:** A feature f is called infrequent if it only appears in a few reviews. These are some features that only a small number of people talk about. Some potential customers may be interested in these features (Hu & Liu, 2004). Let us take the following examples:

   R1: "Red eye is very easy to correct."

   R2: "The camera comes with an excellent easy to install software."

   R3: "The pictures are absolutely amazing"

   R4: "The software that comes with it is amazing"

   Here R1 and R2 share the same opinion word *easy* but describe different features. R1 is about *red eye*, R2 is about the *software*. Let us consider that *software* is a frequent feature in our digital camera review database. *red eye* is infrequent but also interesting. Similarly, *amazing* appears in both R3 and R4, but R3 is about *picture* while R4 is about the *software.*

4. **Noisy text:** Product reviews contain a lot of noise that requires cleaning of text. If any URL links, HTML Tags, XML Tags are present in the text data, we need to remove such text from the product reviews in order to obtain a clean text.
5. **Preprocessing the unstructured text:** Product reviews contain the data/text in an unstructured format where there can be many special characters, punctuations, stop words present. This preprocessing should be done which includes tokenization, stop words removal, punctuation removal, stemming, white space removal etc., in order to obtain a structured format of text in the reviews.
6. **Opinion Word Extraction:** Opinion words are terms that users/reviewers use to convey a positive or negative opinion. Observing that people frequently express their views on a product feature using opinion words that are located around the feature in the sentence, we can use all

the remaining frequent features to extract opinion words from the review database(Hu and Liu, 2004). For example, let us look at the two sentences:

R1:"The strap is horrible and gets in the way of parts of the camera you need access to."

R2:"After nearly 800 pictures I have found that this camera takes incredible pictures."

In the first sentence, *strap*, the feature, is near the opinion word *horrible*. And in the second example, feature *picture* is close to the opinion word *incredible*.

7. **High Utility Feature Groups:** According to **(**Demir et al., 2019), customer satisfaction or the utility expressed in terms of sentiments for an aspect/feature of a product or service can be determined in Feature-Based Opinion Mining. In addition, once these feature-based sentiment values are determined, a review can be considered a collection of utility values such that each one is assigned to the mentioned feature. Hence, HUFG are feature-sets (group of aspects as a whole) obtained from the product reviews that bring the highest consumer satisfaction and manufacturer's profit(or the highest utility) and highest customer dissatisfaction and manufacturer's loss(or the lowest utility). These utility values contribute to the profit/loss but will not be a part of the feature-groups. For example, <*battery life, camera quality*> together as a feature group can provide highest customer satisfaction and prove to be valuable factor for retailer's sales profit.

Thesis Motivation: As mentioned earlier, it can be seen that Feature-Based Opinion Mining has emerged as an explorable research area. Considering a business point of view, it is important for the manufacturers of the product to monitor their brand, product, and services. It is important for businesspeople to understand what features of their product, brand customers are interested in and what are their opinions at a particular time and their expectations. This will help them in the product redesigns and will increase their sales-profit and thereby increasing the overall rating of their brand. Moreover, customer satisfaction is also an important constraint for product selling and product improvements. This will also help to build further recommendation systems as per the customer's preference. Previous studies have proposed multiple approaches dealing with the problem of Opinion Mining to extract features and opinions from social networking or e-commerce websites. A number of algorithms have been proposed that aim to incorporate several approaches like Data Mining, Machine Learning, Artificial Intelligence etc. to increase the accuracy and relevancy of feature and opinion words. Furthermore, the main idea behind this thesis is to

incorporate Feature-Based Opinion Mining and High Utility Sequential Pattern Mining in order to obtain frequent sequence of features and opinions of products and to obtain high utility feature groups that have number of product features as a whole that tend to bring high profit to the sellers, understand the improvements in product redesigns, increase customer satisfaction and for recommendation systems. We tend to form sequences of features which will further help understand the learning patterns of any related customers that tend to like similar kind of features in a particular fashion and hence will help producers build up the marketing trends and suggestions according to the category of users.

Amazon.com Product Reviews: E-Commerce sites are gaining popularity across the world**.** People visit them not just to shop for products but also to know the opinion of other buyers and users of products. Online customer reviews are helping consumers decide which products to buy and companies to understand consumers' buying behavior**.** As of 2018, there were 233.1 million reviews collected by Amazon on their product, with over 60 million users making Amazon a suitable platform for gathering opinions about products. Amazon shows average ratings for products based on consumer feedback at the top of the product page, with customer reviews at the bottom. When a customer views a product page, they are immediately presented with product ratings and the amount of people who have given that product a rating.

Why Product Reviews?

1. Product Reviews provide better insights into the product: Hearing from people's previous purchasing experience allows prospects to assess whether the product has met customer expectations.
2. Product Reviews rectify the product's issues: If most customers are pointing out the same problem in the product, it is for a retailer to rectify the defect to get resolved. Customer reviews help find the loopholes in the product and provide an opportunity to improvise in those areas.
3. Product Reviews increase business, sales-profit, and customer satisfaction. Reviews are available on many prominent social media platforms like Twitter, Facebook, Amazon, Yelp, and other e-commerce and social networking websites.

1.8 Thesis Problem Definition

Table 9 below shows the huge volume of product reviews that are collected for various categories so we can see that it is impractical to read each of these reviews manually to get what features they are talking about, and the opinions expressed on each feature. Hence the problem arises: Can we build a system to automatically mine all the features from these reviews and the opinions expressed on each of the aspects? Can we build an algorithm that can extract High Profit Sequential Feature Groups and frequent features that will enhance customer satisfaction and increase sales-profit of retailers?

Amazon Products Categories	**Number of Reviews**
Cell Phones and Accessories	194439
Musical Instruments	10261

Table 9: Conservative estimates of reviews that mention the products dataset.

Existing system that attempts to address this similar problem is HPFG19_HU (Demir et al., 2019) that extracts High Profit Aspect Groups. Other systems like (Rashid et al., 2013, Rana & Cheah, 2018) perform Feature-Based Opinion Mining to obtain only the frequent features or important features.

Existing Systems	**Research Goal**	**Technique to Obtain Relevant Aspects**	**Limitations**
Data Mining Approaches – SPM and ARM (Rashid et al., 2013)	To obtain frequent product features and opinion words extraction from customers' opinions obtained from a social networking website.	Apriori algorithm and GSP algorithm are used to mine these patterns and comparison is made	Only the frequent features are extracted without getting high profit features
Sequential patterns rule-based approach by (Rana & Cheah, 2018)	To extract product features and opinions from the reviews using Class Sequential Rule (CSR) method	Class Sequential Rules and Opinion Lexicon application on free format reviews	The features were obtained, and rules were formed from them. Does not deal

			with extracting feature sets.
HPFG19_HU (Demir et al., 2019)	To extract feature-opinion sets and high profit feature groups	Aspect-Based Sentiment Analysis and High Utility Itemset Mining were used to extract high profit features	The feature sets obtained are itemsets and do not consider the order of occurrences and hence frequent features and accuracy and relevancy of features are compromised

Table 10: Existing Systems That Perform Feature-Based Opinion Mining

Shortcoming of existing algorithms include:

1. In order to obtain High Profit Feature Groups, that yield features which increase sales-profit, profit table or utility values are required. These utility values serve as an important factor as it represents the importance of feature word in the entire corpus of product reviews(external utility) as well the importance of word in each review sentence(internal utility). Out of the algorithms mentioned above, only HPFG19_HU(Demir et al., 2019) System tries to obtain itemsets of features which give single features or multiple features in a set as output. The HPFG19_HU system does not deal with the order of occurrences(sequences) formed in the features.
2. The algorithms(Rashid13OFExt and Rana18OFExt) tend to obtain frequent features and important features using Sequential Pattern Mining and Class Sequential Rules. These systems try to extract implicit and explicit features, frequent and infrequent features but do not yield high utility features and do not contribute to profit values because ‘utility’ values are not considered.

In this thesis, we try to enhance the work of HPFG19_HU (Demir et al., 2019) that extracts High Utility Aspect(HUA) Groups by obtaining High Utility Itemsets of aspects. We try to obtain sequences of features which are called High Profit Sequential Feature Groups based on High

Utility Sequential Patterns. To the best of our knowledge, none of the systems have tried to combine High Utility Sequential Pattern Mining with Opinion Mining to extract High Profit Sequential Feature Groups which will use sequential patterns and utility values to extract High Profit Feature Groups based on sequences and increase customer satisfaction and manufacturer's sales-profit in order to improve their brands product redesigns by considering the high utility feature groups.

Problem Statement:

For a social media or an online review website of e-commerce system having Product Reviews dataset R, we aim to extract product features and opinions mined which will be further given as an input along with a given minimum threshold utility to discover frequent High Utility Sequential Patterns over a dataset to get all frequent sequences whose utility is no less than threshold utility, which will serve as High Utility Sequential Feature Groups(HUSFG) yielding High Profit Sequential Feature Groups(HPSFG) that will be profitable for the retailer and also increase customer satisfaction.

1.9 Thesis Contributions

For this thesis, we focus on combining two areas to perform Opinion Mining from Social Networking Websites. We discover High Utility Sequential Patterns from the mined features and opinions. This thesis proposes a system called High Profit Sequential Feature Groups based on High Utility Sequences (HPSFG_HUS) (**Figure 9**) and aims to improve the work done in HPFG19_HU by (Demir et al., 2019) on the product reviews in mining features and opinions. We obtain sequences of features and frequent patterns along with High Profit Feature Groups(itemsets) using High Utility Sequential Patterns. This will enhance the existing system in terms of relevancy and accuracy of the feature groups required for customer satisfaction.

1.9.1 Thesis Feature Contributions:

1. **Modifying the sentiment score to get positive values which will be used as utility values:**

- **HPFG19_HU** System (Demir et al.,2019) has considered sentiment score obtained from SentiStrength library. This sentiment score is considered as internal utility values and external utility values are considered as identical values (=1).

- **Proposed HPSFG_HUS** System modifies the sentiment score obtained from SentiStrength library to get positive values for all the sentiment scores. This score will be considered as internal utility value and external utility value will be considered as 1. This positive score will be further useful to calculate the sequence utilities of each sequence transaction in the database. The detailed process is provided in **section 3.3**

2. **Forming Q-Sequence Database from itemsets of transactions**

- **HPFG19_HU** System forms itemsets of features and forms a *transaction database* using Triples by grouping the triples by adding/averaging the internal utility values if they have similar features together in same or different sentences.

- **HPSFG_HUS** forms Quantitative-Sequence (*Q-Sequences Database)* (Yin, Zheng & Cao, 2012) from the itemsets of triples (<feature, opinion, utility_value>) without creating any groups and by using all the occurrences of features with their utility values. These Q-Sequences of features are formed with respect to the order of their occurrences(sequences) in the review sentences. Each sequence has its own Sequence ID in the Q-Sequence database and Sequence Utility values for each transaction. The detailed process is provided in **section 3.3**

3. **High Utility Sequential Pattern Mining to get High Profit Sequential Feature Groups**

- **HPFG19_HU** System has considered *itemsets of features* and hence performs the task of High Utility Itemset Mining for discovering High Utility Feature Groups which will be considered as High Profit Feature Groups.

- **HPSFG_HUS** system mines the High Utility Sequential Patterns which provides *high profit sequences of features* greater than or equal to the threshold sequential utility called High Profit Sequential Feature Groups.The detailed process is provided in **section 3.3**

1.9.2 Thesis Procedures Contributions:

We propose an algorithm called High Profit Sequential Feature Groups based on High Utility Sequences (HPSFG_HUS) which is built on the top of existing High Profit Feature Groups based on High Utility (HPFG19_HU) (Demir et al.,2019) System by making the following modifications and additions:

1. In the **HPFG19_HU** System(Demir et al.,2019), the authors perform Aspect-Based Sentiment Analysis to extract aspects and sentiments and sentiment score triples(aspect, sentiment, sentimentscore) using SentiStrength library. These scores contain positive and negative values, and the scores are considered as internal utility values whereas external utility value is considered constant(=1)while forming itemsets. In our **proposed HPSFG_HUS** System, we modify the sentiment score by adding '**+5'** to get all positive values which will be considered as utility values(**section 3.4** and **section 3.5**)
2. In the **HPFG19_HU** System, *transactions database* is constructed from itemsets of features with the triples and grouped triples, whereas in our **proposed HPSFG_HUS** System, we form Q-sequences of features from triples(**section 3.5**) by considering the order of occurrences of features in the review sentences. Then the *Q-Sequence* Database is constructed from the q-sequences of features and sequence utility values of each sequence is calculated which is the addition of utility values of each sequence (**section 3.6**)
3. In the **HPFG19_HU** System, *High Utility Itemset Mining* is performed for itemsets of features using ***FHN(Faster High Utility itemset miner with Negative unit profits)*** **(**Lin, Fournier-Viger, & Gan,2016**)** algorithm and the output is High Profit Feature Groups. In our **proposed HPSFG_HUS** System, *High Utility Sequential Pattern Mining* is performed on Q-Sequence Database using ***USpan*** (Yin, Zheng & Cao, 2012) algorithm to obtain High Utility Sequential Patterns of features(**section 3.7**). These are High Profit Sequential Feature Groups having high utility sequences of significant importance as they are of potentially high profit from a business perspective.

1.10 Thesis Outline

In Chapter 2, we provide detailed related work on the two areas. Chapter 3 provides a proposed solution framework with running examples. Chapter 4 provides various experimental results, including comparisons between the existing and the proposed approach. Finally, Chapter 5 provides some concluding remarks.

CHAPTER 2: RELATED WORKS

The presented approach is mainly related to two areas of research which are *Opinion Mining* and *High Utility Sequential Pattern Mining*, therefore the related state of art algorithms on these areas are presented in the following subsections.

2.1 Text Preprocessing Methods (Mayo, 2017)

The online reviews consist of raw texts and these texts available are in an unstructured format, hence, text preprocessing methods help "clean up" the text so they can be fed into the text mining systems. These preprocessing operations in NLP include:

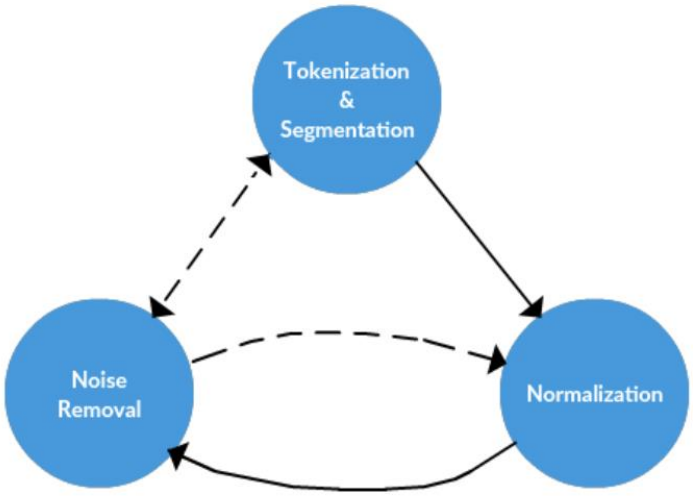


Figure 3: The basic Text Data Preprocessing Framework in NLP

[1] Tokenization and Segmentation

The method of breaking up text-data into tokens is tokenization. These tokens, most frequently, are usually words or phrases. Larger sections of text can be tokenized into sentences, then sentences into words, and so forth. Text segmentation or lexical analysis are other terms for tokenization. The breakdown of a significant chunk of text into portions larger than words (e.g., paragraphs or sentences) is referred to as segmentation, whereas tokenization refers to the breakdown process that results entirely in words.

For example, after tokenizing the sentence, "I love the new smart phone that was released by Samsung" the result is:

['I' 'love' 'the' 'new' 'smart' 'phone' 'that' 'was' 'released' 'by' 'Samsung'.]

Systems used for tokenization are called tokenizers. An example of a tokenizer is Natural Language Toolkit Tokenizer (NLTK, 2015).

[2] Normalization

Normalization typically refers to a set of similar activities that are intended to place all text on a level playing field: converting all text to the same (upper or lower) case, eliminating punctuation, converting numbers to their word equivalents, etc. Normalization places all terms on an equal footing and makes it possible to process them uniformly. The tasks involved in Normalization process are:

- **Stemming:** Stemming is the process of dropping affixes (suffixed, prefixes, infixes, circumfixes) from a word to obtain a stem word.

  chopping → chop

- **Lemmatization:** Lemmatization is the process of transforming a word to its base form.

  caring→care

  Lemmatization is associated with stemming, which differs in that lemmatization can grab canonical forms based on the lemma of a word.

  For example, stemming the word "better" would fail to return its citation form (another word for lemma); however, lemmatization would result in the following:

  better → good

- **Lowercasing:** Converting a word to lower case.

  NLP → nlp; Book→book

  Words such as Book and book imply the same, but these two are interpreted as two separate words in the vector space model when not converted to the smaller case (resulting in more dimensions)

- **Removing numbers:** Converting numbers to textual representations. For Sentiment Analysis, removing numbers may make sense because numbers offer no information about sentiments. This is an optional step that is dependent on the dataset type.
- **Removing whitespaces and punctuation:** This step is a part of tokenization process and can be done explicitly as well.

- **Stop words removal:** Stop words are those words that are filtered out before further text processing because, although being the most prevalent terms in a language, they contribute little to total meaning. These terms have no real meaning because they don't assist distinguish between two publications. For instance, "the," "and," and "a" are all needed words in a sentence, they don't usually contribute much to one's understanding of the content. As a simple example, the following panagram is just as understandable if the stop words are removed:

  ~~The~~ quick brown fox jumps over ~~the~~ lazy dog.

- **Parts-of-Speech Tagging (POS Tagging):** This is the process of assigning parts of speech (e.g., noun, adjective, adverb etc.) to words in a sentence. For example, POS tagging the sentence, "They have always been refusing us to obtain a refuse permit" gives the following output:

  [('They', 'PRP'), (have, 'VBP'), ('always', 'RB'), ('been', 'VBN'), (refusing, 'VBG'), ('us', 'PRP'), ('to', 'TO'), ('obtain', 'VB'), ('a', 'DT'), ('refuse', 'NN'), ('permit', 'NN')]

  Where 'NN' is the tag for noun and 'VB' is the tag for verbs. The POS Tags are named according to a naming convention proposed by Santorini (1990) and the complete list of tags, and the description is shown in the table below:

POS TAGS	**Description**
CC	Coordinating conjunction
CD	Cardinal number
DT	Determiner
EX	Existential there
FW	Foreign word
IN	Preposition or subordinating conjunction
JJ	Adjective
JJR	Adjective, comparative
JJS	Adjective, superlative
LS	List item marker
MD	Modal
NN	Noun, singular or mass
NNS	Noun, plural
NP	Proper noun, singular
NPS	Proper noun, plural
PDT	Predeterminer
POS	Possessive ending
PP	Personal pronoun

PP$	Possessive pronoun
RB	Adverb
RBR	Adverb, comparative
RBS	Adverb, superlative
RP	Particle
SYM	Symbol
TO	to
UH	Interjection
VB	Verb, base form
VBD	Verb, past tense
VBG	Verb, gerund or present participle
VBN	Verb, past participle
VBP	Verb, non-3rd person singular present
VBZ	Verb, 3rd person singular present
WDT	Wh-determiner
WP	Wh-pronoun
WP$	Possessive wh-pronoun
WRB	Wh-adverb

Table 11: POS Tags and their Descriptions

An example of a POS Tagger built for microblog posts is Tokenizer (Owoputi et al., 2013).

[3] Noise Removal: This part cleans up the text. This is a much more task-specific section of the framework. We are not dealing with a linear process in which the steps must all be performed in a specific order. As a result, noise removal can take place before or after the previously specified stages, or at any time in between. The tasks involved in Noise Removal process are:

- **Remove HTML Tags:** If the reviews or texts were scraped from the internet, there's a good probability they'll contain HTML tags. These tags aren't relevant for our NLP activities, thus it's best to get rid of them.
  Example: The people do not understand.
</br>This is not a good quality phone.
- **Convert Accented Characters:** Words containing accent marks, such as “latté” and “café”, can be converted and standardised to just "latte" and "café," or our NLP model will treat "latté" and "latte" as separate words, despite the fact that they refer to the same thing.
- **Expand Contractions: “**Don't” and “can't” are two examples of contractions. Text can be standardised by expanding such terms to “do not” and “cannot.”
- **remove text file headers, footers:** Removing any text headers or footers of file which are not required in the file

- **extract valuable data from other formats, such as JSON, or from within databases:** Since the line between noise removal and data collection and assembly is fuzzy, some noise removal must occur before other preprocessing steps. Any text necessary from a JSON structure, for example, would need to be removed before tokenization.

2.2 Social Network Opinion Mining on Product Reviews Domain

The World Wide Web's online social networks are becoming increasingly interactive and networked. Web 2.0 technologies enable a variety of platforms, such as blogs, wikis, and forums, where users can share information about products and manufacturers. This data provides an abundance of information on personal experiences and opinions which are incredibly useful to businesses and sales groups. Opinion Mining has emerged as a very exploratory area for manufactures when it comes to customers' thinking and requirements. This subsection consists of review of the research in this area.

2.2.1 Association Rule Mining Approach by (Kim et al.,2009)

The authors (Kim et al., 2009) proposed an approach for Opinion Mining of product reviews using Association Rule Mining. In their proposed methodology, they do the POS Tagging on each review and then extract features and opinion words in the form of transaction data. Then association rules(Liu & Wang, 2007) are discovered, and then PMI-IR algorithm is used for obtaining the summarized information. This research is carried out to study the problem of opinion summarization using association rule of online product reviews. Through Opinion Mining customers can easily find out other people's summarized opinions without reading all the product reviews.

The scheme for a transaction T is defined as follows.

T = (product, [$feature_1$, $opinion_1$], …, [$feature_n$, $opinion_n$], $opinion_1$, …, $opinion_i$)

product: name of product

feature: feature of product on product reviews

opinion: thinking of customer about product or feature

[feature, opinion]: feature-opinion set

The transaction T is comprised of extracted feature and opinion from each product review.

Proposed Methodology:

Three steps are followed in order to achieve the results. These steps along with a walk-through example are shown below:

Input: a product review dataset

Output: summarization of reviews

Step 1: Preprocessing

In this step, a phase-structure tree on each sentence of reviews is made using Stanford Parser. Then the feature and opinion words are extracted from the parsed tree. Extracted feature and opinion words are stored in Transaction T. The features of product are usually nouns or noun phrases in review sentences and the opinions of product feature are usually adjective phrases. Hence, the extraction algorithm extract opinion and feature via adjective and noun phrase.

Input: a sentence tree from review text

Output: T = (product, [$feature_1$, $opinion_1$], …, [$feature_n$, $opinion_n$], $opinion_1$, …, $opinion_i$)

Example: This camera has a solid body and excellent quality

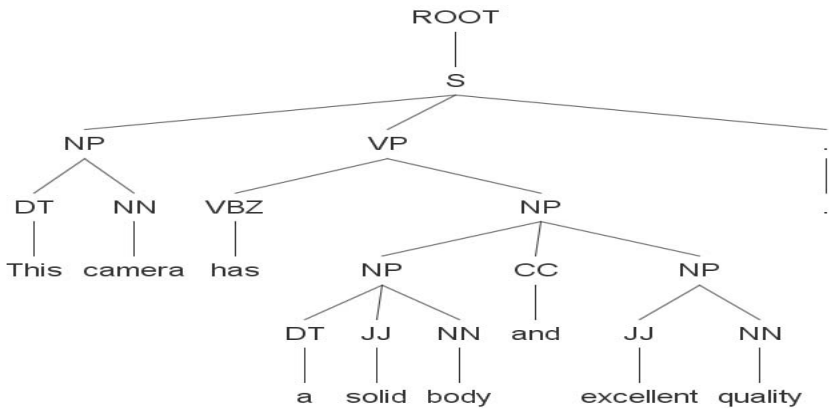


Figure 4: phrase-structure tree

T=(camera, [body, solid], [quality, excellent], solid, excellent)

Step 2: Association Rule Mining

Let I = I1, I2, …, In be a set of n distinct attributes, T be transaction that contains a set of items such that T ⊆ I, D be a database that consists of transaction Ts. An association rule is an implication of the form X ⇒ Y, where X, Y ⊂ I are sets of items called itemsets, and X ∩ Y = ϕ

$$\text{Support } (X \Rightarrow Y) = \frac{\sigma(X \cup Y)}{N}$$

$$\text{Confidence}(X \Rightarrow Y) = \frac{\sigma(X \cup Y)}{\sigma(X)}$$

To find association rules using association rule mining, Apriori principle is used. Apriori Principle(Srikant & Agarwal, 1994)is defined as "If an itemset is frequent, then all of its subsets must also be frequent." Thus, if feature-opinion set is frequent, then all of its subsets must also be frequent. The following four types of rules are obtained after this step that satisfies minimum support:

Type 1: product ⇒ opinion – overall opinion of customers for product

Type 2: product ⇒ feature -- features that appear frequently in product reviews

Type 3: feature ⇒ opinion -- opinion by each feature.

Type 4: product ⇒ [feature ⇒ opinion] – combination of Type 2 rule and Type 3 rule.

Step 3: Summarization

The opinion of a product which has high support value indicates a rating of the product, because it is important that some opinion was how many referred in transaction data. The rules of Type 4 indicate an opinion of product.

$$\text{Support (Type 4)} = \frac{(\text{product} \Rightarrow [\text{feature} \Rightarrow \text{opinion}])}{N}$$

$$\text{Confidence(Type 4)} = \frac{(\text{feature} \Rightarrow \text{opinion})}{(\text{Type 4})}$$

Whether the customer feels positive or negative based on calculating semantic orientation using PMI-IR algorithm about high confidence value of feature-opinion set. The semantic orientation(SO) of opinion is used to classify reviews as positive or negative. The Pointwise

Mutual Information(PMI) between two words, word1 and word2, is defined as follows (Church and Hanks, 1990):

$$\mathrm{PMI}(word_1, word_2) = \log 2\left[\frac{\mathrm{p}(word_1 \,\&\, word_2)}{\mathrm{p}(word_1)\,\mathrm{p}(word_2)}\right]$$

Summary of reviews of Type 1 looks like the following:

Product: product name
⇒ opinoin$_1$, opinion$_2$, …
Adv: feature1 ⇒ opinion
feature2 ⇒ opinion
…
Disadv: feature1 ⇒ opinion
feature2 ⇒ opinion
…

Advantage : Association Rule Mining & opinion summarization extract all the features

Limitation : Only explicit features are getting extracted. This work can be extended to implicit features of opinions.

2.2.2 Twitter Data to mine Opinions by (Hridoy et al., 2015)

The approach of Sentiment Analysis is a good way to find out what people think. One of the most effective and accurate measures of public sentiment is social network data. The authors (Hridoy et al., 2015) have proposed a methodology that allows utilization and implementation of twitter data to determine public opinions. The authors in the paper have examined a large data set extracted from Twitter as tweets from which they tried to assess the popularity of a given product in many locations. The research deals with outcome prediction and explores localized outcomes.

Proposed Methodology:

Step 1: The data is extracted from Twitter using the "twitteroauth" Twitter public API by Williams(2012). This data was stored into a MySQL database for further use. Each record or sample contains username, tweet id, text, etc. The geographical location is not available with the tweet. So, the major focus is done on the cities of USA. Each major city has a city center, the

latitude and longitude that was used to define the city itself. The product chosen was iPhone 6 which was trending at that moment.

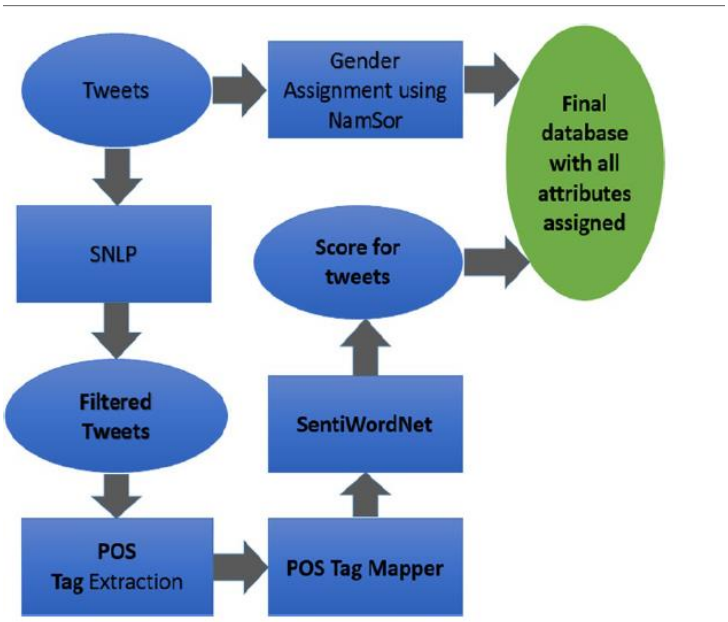


Figure 5: The overall architecture (Hridoy et. al, 2015)

The key and important features considered for iphone 6 were “battery”, “camera”, “iOS”, “iTunes”, “screen”, “sound”, and “touch”. For each tweet, the username, tweet text, location was extracted.

Step 2: Irrelevant data contained in the tweets was removed through data preprocessing step. For filtering out the useless data, The Stanford NLP Group (SNLP Group 2015) which is an open-source natural language processing tool developed by Stanford University was used. There are 50 predefined relations called dependences available in the tool, out of which only 3 are used which include nsubj, amod, dobj.

nsubj: This relation is used to find relations between nouns and adjectives or verbs which are complementing the noun in a sentence.

Example: My iPhone 6 camera is awesome!–*camera* noun linked with *awesome* adjective

amod: This is used to find any adjectives that are used in a sentence to modify noun phrase.

Example: Got a new gold iPhone 6, feeling great!!-gold is modifying noun phrase iPhone6.

dobj: This relation is the direct object, which is used to identify direct objects that a verb is referring to in a sentence.

Example: Love the camera of iPhone 6! – dobj(love, camera); dobj(love, iPhone6)

So, in order for a tweet to be valid and pass the preprocessing phase, it must have at least one of the three dependencies listed above, as well as at least one keyword from the list of prefixed keywords.

Let t1 be a tweet from the set of tweets T. If t1 contains **nsubj (n1, n2) V amod (n3, n4) V dobj (n5, n6)** Where at least one parameter, n_i, of the valid relations contains a keyword from the predefined list, then that tweet is said to be valid and is moved to the set of filtered tweets.

Step 3: POS Tagger is used to analyze the tweet and separate out the tweet into individual words and assign a part of speech to it. SentiWordNet(Baccianella, Esuli, & Sebastiani, 2010) is used to assess the sentiment of the tweets. SentiWordNet only address nouns, adjectives, adverbs, and verbs. So, for any other part of speech a mapping convention is applied. An example of the mapping convention would be that if a word is assigned the VBZ tag, which stands for verb in present tense, it will be assigned the Verb tag by the mapper. This set of words along with their normalized POS tags are then sent to SentiWord and the sentiment for each word is calculated and then the individual numeric sentiments are added to obtain a final score for the tweet.

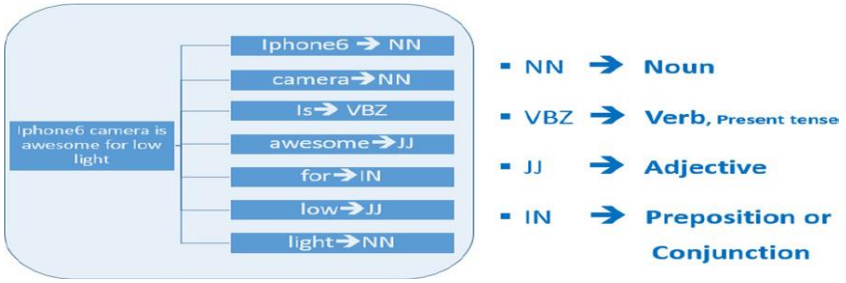


Figure 6: POS-Tagging of the example

Example: iphone6 camera is awesome for low light

Word	POSTag	Normalized POS	Score
iphone6	JJ	a	0.0
Camera	NN	n	0.0
Is	VBZ	v	0.0
Awesome	JJ	a	0.75
For	IN	null	0.0

Low	JJ	a	-0.253
Light	NN	n	0.056
Total Score			0.552

Table 12: The example and its described values

The total score is the sum of all the individual scores and is normalized within −1 to 1.Score is calculated using the following formula:

Score($location_j$) = $\frac{\sum_{i=1}^{n} SentiScore_i}{n}$ where, n = total number of tweets, $SentiScore_i$= SentiWord score for each tweet, $location_j$ = refers to one particular city

Step 4: As the scores obtained in this way do not follow any scale or are not within a given range it was necessary to normalize these scores to obtain fixed sentiment grades for tweets

Sentiment Score Range	**Assigned Sentiment**
Score ≤ -0.5	Worst
-0.5 < Score ≤ 0	Bad
Score=0	Neutral
0<Score≤ 0.5	Good
Score ≥ 0.5	Excellent

Table 13: Sentiment Score Range

Step 5: Once the filtered tweets were scored and placed into MySQL database, the database was exported into Rapid Miner and then the NamSor(2015) – a data mining tool and an extension to Rapid Miner was applied to the database. The set of genders returned by NamSor was then inserted into the database for each corresponding tweet.

Results: To properly understand trends and variations in sentiments various comparisons were made. The comparisons started at a national level and then became more detailed by the introduction of cities and genders. These are as follows:

1. *National Average Sentiment* Sentiments inclusive of all cities and genders. It gives a general overview.

2. *National Feature Average Score* Average score inclusive of all cities but grouped by features. It gives general view of sentiment towards iPhone 6 features.

3. *National Male/Female Average Score* Average scores inclusive of all cities and features grouped by gender.

4. *National Male/Female Feature Average Score* Average scores inclusive of all cities grouped by gender and features individually.

5. *Average Score per City* Average sentiment score for the individual city.

6. *Male/Female Sentiment* per city Sentiment for each city grouped by gender.

7. *Feature Average Score per city* Average score per city grouped by feature.

8. *Male/Female Feature Average per city* Sentiment score for each city grouped by gender for each individual feature. This is a very important comparison because it involves all the variables, specific location, gender, and feature.

Advantage: A method to determine popularity/opinion/sentiment of a product in different locations across male and female users is proposed. This bifurcation helps in categorization.

Limitation: Lesser number of tweets are used, and quality of tweets was also low.

Other Studies:

(Turney, 2002) is one of the early studies that addresses the problem of Opinion Mining that uses an unsupervised learning algorithm to classify reviews. In this system, a part-of-speech tagger is used to identify phrases in the text that contains adjectives or adverbs. Two consecutive words are extracted from the reviews if their tags conform to any of the patterns of POS Tagger. Then the semantic orientation of each extracted phrase is estimated using PMI-IR algorithm which uses Pointwise Mutual Information as a measure of the strength of semantic association between two words. PMI-IR(Church & Hanks, 1990) estimates PMI using Information Retrieval (IR) techniques and noting the number of matching documents (hits). Lastly, the given review is assigned to a class "$recommended$" or "$not\ recommended$" based on the average semantic orientation(SO) of the phrases. If average SO is positive, classify the review as $recommended$, and otherwise $not\ recommended$.

Dave et al. (2003) proposed an opinion extraction and mining method based on features and scoring matrices. This approach takes structured reviews and identifies appropriate features and scoring formula to determine whether reviews are positive or negative. The results perform machine learning method called Transductive learning to classify review sentences from the web. Firstly, users' text reviews, title, thumbs-up or thumbs-down rating are collected from the large

web sites. Then they separate the document into sentences, then split sentences into single-word token by substituting numerical tokens with $NUMBER$, product's name token with $\_productname$. Then pass the document sentence by sentence through Lin's MINIPAR linguistic parser to yield part of speech of each word and the relationships between parts of the sentence. Later, Pass the resulted words through WordNet, a database for finding synonyms and identify negative phrases and mark all words following the phrases as negated. Combine sets of n adjacent tokens into $n{-}grams$ and count frequencies of the extracted features i.e., the number of times each term occurs, the number of documents each term occurs in, and the number of categories a term occurs in. then set upper and lower limits for each of these measures, constraining the number of features looking for to determine a threshold for the classifier. After selecting a set of features $f1\ldots\ fn$, assign them scores. These scores are used to place the test documents in the set of positive reviews C or negative reviews C.

(Jain & Katkar, 2015) have proposed an approach of analyzing users' sentiments using Data Mining classifiers. This method is also able to compare the performance of single classifier over ensemble of classifiers for Sentiment Analysis. Twitter is Social Networking website that allow users to send and read 140-character messages called tweets. Users of Twitter can read and post tweets. The authors in their paper have tried to present a mechanism to predict the overall sentiments inclination of Indian people towards political situation and issue. For that, they followed data collection from twitter, data preprocessing to remove noise and cleaning, forming training and testing dataset and classification by splitting each tweet into into words and polarity of each word is then calculated using SentiWordNet.

2.3 Sequential Pattern Mining

Sequential pattern mining (SPM) discovers frequent subsequences as patterns (sequential patterns) in a sequence database. SPM is an important problem with broad applications, including the analysis of customer purchase behavior, web access patterns, scientific experiments, disease treatment, natural disasters, and protein formations. A SPM algorithm extracts frequent sequential patterns from a sequential database as sequences with support greater than or equal to a given minimum support, which can then be used by end users or management to discover associations between different items or events in their data for marketing campaigns, business reorganisation, prediction, and planning. In this section, we will be discussing about General Sequential Pattern Mining (GSP).

2.3.1 GSP (Generalized Sequential Pattern) Algorithm by (Srikant & Agrawal, 1996)

GSP is an Apriori-based sequential pattern mining algorithm introduced by (Srikant & Agrawal, 1996). The main step in the GSP algorithm is candidate generation (Ck) and pruning (Lk). To generate a candidate, we can use pair found in K-1th pass by merging. According to the algorithm, first sequence W1 and second sequence W2 can be merged if subsequences obtained by removal of the first element of sequence W1 and last element of sequence W2 are same. In the second step, we need to prune candidate that contains a subsequence which is infrequent in K-1 pass. We need to iterate the process of candidate generation (Ck) and pruning (Lk) until a candidate set is empty. Finally, frequent sequences are the union of the entire list obtained so far.

Example of GSP algorithm:

Input: sequence database (Table 14), minimum support=2 and candidate set (C1) = {A, B, C, D, E, F, G} and algorithm=GSP

Output: Frequent Sequential Patterns

SID	Sequences
1	<(A),(B),(FG),(C),(D)>
2	<(B),(G),(D)>
3	<(B),(F),(G),(A,B)>
4	<(F),(A,B),(C),(D)>
5	<(A),(B,C),(G),(F),(D,E)>

Table 14: Sequence Database

Step 1: Find 1- frequent sequence (L1) satisfying minimum support: Check the minimum support threshold of each singleton item and keep only sequences with occurrence or support count in the database that are greater than or equal to the minimum support count of 2. For example, (L1) = {< (A):4>, <(B):4>, <(C):3>, <(D):4>, <(F):4>, <(G):4>}.

Step 2: Generate candidate sequence (Ck=2) using L 1 $GSPjoin$ L1. To generate larger candidate set 2, use 1-frequent sequence (L1) found in step 1 to join itself using GSP join way, which can be written as L (k-1) $GSPjoin$ L (k-1) and it requires every sequence (W1) found in first L (k-1) joins with other sequence (W2) in the second if subsequences obtained by removal of the first element of W1 and last element of W2 are same. In our case, we are generating sequences with candidate 2, (Ck=2), which can generate 51 types of 2-length candidate set using Apriori algorithm(Agrawal & Srikant, 1994) as present in Table 15.

<(A),(A)>	<(A),(B)>	<(A),(C)>	<(A),(D)>	<(A),(F)>	<(A),(G)>
<(B),(A)>	<(B),(B)>	<(B),(C)>	<(B),(D)>	<(B),(F)>	<(B),(G)>
<(C),(A)>	<(C),(B)>	< (C),(C)>	<(C),(D)>	<(C),(F)>	<(C),(G)>
<(D),(A)>	<(D),(B)>	<(D),(C)>	<(D),(D)>	<(D),(F)>	<(D),(G)>
<(F),(A)>	<(F),(B)>	<(F),(C)>	<(F), (D)>	<(F),(F)>	<(F),(G)>
<(G),(A)>	<(G),(B)>	<(G),(C)>	<(G),(D)>	<(G),(F)>	<(G),(G)>
<(A,B)>	<(A,C)>	<(A,D)>	<(A,F)>	<(A,G)>	<(B,C)>
<(B,D)>	<(B,F)>	<(B,G)>	<(C,D)>	<(C,F)>	<(C,G)>
<(D,F)>	<(D,G)>	<(F,G)>			

Table 15: Candidate Generation Table

Step 3: Find 2- frequent sequences (L2) by counting the occurrence of 2-sequences in candidate sequence (C2) to keep the only sequence with occurrence or support count in the database greater than or equal to the minimum support.

For example, L2= {<(A), (B)>, <(A, B)>, <(A), (C)>, <(A), (D)>, <(A), (F)>, <(A), (G)>, <(B), (C)>, <(B), (D)>, <(B), (F)>, <(B), (G)>, <(C), (D)>, <(F), (A)>, <(F), (B)>, <(F), (C)>, <(F), (C)>, <(F), (D)>, <(G), (D)>}.

Step 4: Repeat process of candidate generation and pruning until the result of candidate generate (Ck) and prune (Lk) for finding frequent sequence is an empty set.

Output: The output frequent sequences as union of L1 U L2 U L3 U L4 U … Lk.

1-Frequent Sequences	**2-Frequent Sequences**	**3-Frequent Sequences**	**4-Frequent Sequences**
<(A)>, <(B)>, <(C)>, <(D)>, <(F)>, <(G)>	<(A), (B)>, <(A, B)>, <(A), (C)>, <(A), (D)>, <(A), (F)>, <(A), (G)>, <(B), (C)>, <(B), (D)>, <(B), (F)>, <(B), (G)>, <(C), (D)>, <(F), (A)>, <(F), (B)>, <(F), (C)>, <(F), (D)>, <(G), (D)>	<(F), (C), (D)> , <(F), (B, A)>, <(F), (A, B)> , <(B), (G), (D)> , <(B), (F), (D)> , <(B), (C), (D)> , <(A), (G), (D)> , <(A), (F), (D)> , <(A), (C), (D)> , <(A), (B), (G)> , <(A), (B), (F)> , <(A), (B), (D)>	<(A), (B), (G), (D)> <(A), (B), (F), (D)>

Table 16: Frequent Sequences Table using GSP.

2.4 High Utility Itemset Mining

Mining high utility itemsets from databases aims to find the itemsets which can bear high profits. High Utility Itemset Mining deals with mining patterns without any order of their occurrences.

2.4.1. Foundational approach of HUIM by (Yao et al., 2004)

Sequential Pattern Mining has emerged as an important topic in Data Mining. The utility is introduced into pattern mining to mine for patterns of high utility by considering the quality (such as profit) and quantity (such as several items purchased) of itemsets. This has led to high utility pattern mining (Yao et al., 2004), which selects interesting patterns based on minimum utility rather than minimum support. Later Sequential Pattern Mining is introduced in the High Utility Mining. A sequence is considered to be of high utility only if its utility value is no less than a user specified minimum utility. Following the High Utility Pattern Mining approach, highly profitable sequential patterns are retrieved, that are considered more informative for retailers in determining their marketing strategy.

Utility is introduced into Sequential Pattern Mining to mine for patterns of high utility by considering the quality (such as profit) of itemsets. This has led to high utility pattern mining (Yao, Hamilton & Butz 2004), which selects interesting patterns based on minimum utility rather than minimum support. The (Yao, Hamilton & Butz 2004) is widely believed that this was the first and foundational paper of High Utility Pattern Mining. The authors first defined the problem of mining high utility itemsets, and a theoretical model of utility mining was proposed. Specifically, two types of utilities for items, namely internal utility and external utility were first proposed (Tseng et al., 2013).

Example:

Input: Table 17 is the transaction table (input database D) where the items in each transaction are associated with an internal utility. The quality table in the Table 18, which contains the external utilities of all the items, namely I = {a, b, c, d, e, f} and a user specified minimum utility threshold ξ. itemset = {a, b, c, d, e, f}.

Output: The High Utility Itemset Patterns.

The problem of mining high utility itemset is to discover all the itemsets whose utility is no less than ξ.

TID	Transactions	Transaction Utility(TU)
T1	(a,2) (d,4) (e,1)	15
T2	(e,2) (f,2)	4
T3	(a,1) (b,1) (c,4) (d,5)	34
T4	(b,2) (d,5) (e,3)	23
T5	(a,1) (c,2) (d,5) (e,3)	24

Table 17:Transaction Database

Item	a	b	c	d	e	f
Weight/Quality(EU)	3	5	4	2	1	1

Table 18: Quality Table

From example, (a, 2) in T1 means the quantity of 'a' is 2. Therefore, the utility of (a, 2) in T1 is u (a, T1) = 3 × 2 = 6, which indicates the profit/price of a is 6. Furthermore, the utility of T1 is u (T1) = u (a, T1) +u (d, T1) +u (e, T1) = 6+8+1 = 15. It is also called the transaction utility of T2. The utility of the whole database is u (D) = u (T1) + u (T2) + ... + u (T5) = 15+4+... + 24 = 100. The utility of itemset {ad} in T1 is u ({ad}, T1) = 6 + 8 = 14, and the utility in the database is u ({ad}) = 14+13+13 = 40. Assume ξ = 35, then {ad} is a high utility itemset. Other high utility itemsets are {acd}, {bd}, {cd}, {d} and {de} with the utilities of 50, 35, 44, 38 and 35 respectively. The downward closure property is not satisfied in High Utility Pattern Mining. The property states that a pattern's support is no less than that of its super-pattern. However, when it comes to the utility framework as in the examples above, the utility of {d} is 38, which is bigger than 35 (the utility of {de}) and smaller than 50 (the utility of {acd}). Both {acd} and {de} are the super-patterns of {d}, but the utilities could be either bigger or smaller. It obviously does not hold the downward closure property anymore.

Advantages: A utility upper bound called Expected Utility for the itemset is introduced, which can be used to prune unpromising candidates.

Limitations: It suffers from the large candidate generation process with more memory consumption and execution time. It fails to follow the downward closure property.

2.5 High Utility Sequential Pattern Mining

Even though Sequential Pattern Mining recognises all items as having the same importance/utility and implies that an item appears only once at a time point, this does not reflect the characteristics of multiple real-life applications, and thus the valuable information of sequences with high utilities

(high profits) is lost. High Utility Sequential Pattern considers the external utility (e.g., unit profits) and internal utility (e.g., quantity) of items such that it can provide users with patterns having a high utility (e.g., profit).

2.5.1 USpan Algorithm by (Yin, Zheng & Cao, 2012)

USpan (Yin, Zheng & Cao, 2012) is one of the High Utility Sequential Pattern Mining algorithms composed of lexicographic q-sequence tree, 2 concatenation mechanisms and 2 pruning strategies.

Input: A sequence database, Profit table, Minimum Utility threshold.

Output: High Utility Sequential Patterns

Step 1: For utility-based sequences, the concept of the Lexicographic Sequence Tree is utilized for determining the characteristics of q-sequences and the authors come up with the concept of Lexicographic Q Sequence Tree (LQS-Tree) to construct and organize utility-based q-sequences.

Step 2: Suppose for a k-sequence t, the operation of appending a new item at the end of t is said to be forming a (k+1)-sequence concatenation. If the size of t does not change, the operation I-Concatenation will occur. Otherwise, if the size increases by one, S-Concatenation is occurred. For example, <ea>'s I Concatenate and S-Concatenate with b would result in <e(ab)> and <eab>, respectively. Let's say two k-sequences ta and tb are concatenated from sequence t, then ta < tb if

i) ta is I-Concatenated from t, and tb is S-Concatenated from t, or

ii) both the sequences ta and tb are I-Concatenated or S-Concatenated from t, but the concatenated item in ta is alphabetically smaller than that of tb.

For example, <(ab)>, < ((ab)b)>, <(abc)> < (ab)b>, <(ab)c> < <(ab)d> .

Step 3: A lexicographic q-sequence tree (LQS-Tree) T is a tree structure satisfying the following rules: Rule1: Each node in T is a sequence along with the utility of sequence, while the root is empty and Rule 2: Any node's child is either an I-Concatenated or S-Concatenated sequence node of the node itself. Rule 3: All the children of any node in T are listed in an incremental and alphabetical order.

Step 4: Additionally, if minimum utility threshold = 0, then the complete set of the identified High Utility Sequential Patterns forms a complete LQS-Tree, with complete search space. USpan uses a depth-first search strategy to traverse tree to search for high utility patterns.

Step 5: The I-Concatenation and the S-Concatenations are applied to the LQS Tree.

Step 6: The depth and width pruning techniques are further applied to remove the unpromising candidates from the tree.

Example :

Input : A sequence database shown in Table 19, shows five sequences listed with the itemsets associated with quantity, i.e., a number of items purchased in each sequence (in SID = 1 is e=5). In the Profit table from Table 20, each item's price is given, which represents quality (Price) of the item in a transaction. The minimum utility threshold $\xi = 0$;

Output: High Utility Sequential Patterns.

Sid	Q-Sequence
1	<(e, 5)[(c, 2)(f, 1)](b, 2)>
2	<[(a, 2)(e, 6)][(a, 1)(b, 1)(c, 2)][(a, 2)(d, 3)(e, 3)]>
3	<(c, 1)[(a, 6)(d, 3)(e, 2)]>
4	<[(b, 2)(e, 2)][(a, 7)(d, 3)][(a, 4)(b, 1)(e, 2)]>
5	<[(b, 2)(e, 3)][(a, 6)(e, 3)][(a, 2)(b, 1)]>

Table 19: Q-Sequence Database (Yin, Zheng & Cao, 2012)

Item	***a***	***b***	***c***	***d***	***e***	***f***
Price	2	5	4	3	1	1

Table 20: Profit Table

Step 1: The utility of a single item can be defined as the multiplication of its purchased quantity and its profit. The utility of an itemset can be stated as the sum of the utilities of all its items. For example, for sequence s1, the utility of q-item (e, 5) can be calculated as $u(e, 5) = 5 \times 1 = 5$, which is also the utility of the first itemset's utility. Similarly, the utility of s1 and S can be calculated as $u(s1) = u(e, 5) + u(c, 2) + u(f, 1) + u(b, 2) = 5 \times 1 + 2 \times 4 + 1 \times 1 + 2 \times 5 = 24$ and $u(S) = u(s1) + u(s2) + u(s3) + u(s4) + u(s5) = 24 + 41+27+50+37 = 179$ respectively. The utility of sequence *ea* is umax (<ea>) = 10 + 16 + 15 = 41. If the specified minimum utility is $\xi = 40$, then sequence <ea> is a High Utility Sequential Pattern because $umax(s) = 41 \geq \xi$. In frequent sequential pattern mining, the downward closure property serves as the foundation of pattern mining algorithms. However, this property does not satisfy in the High Utility Pattern Mining problem. Over here, umax (<ea>) = 41, but umax(<e>) = 5 + 6 + 2 + 2 + 3 = 18, which is comparatively lower than its super-pattern. The utility values of the sequential patterns <(ae)>, <(ae)a>, <(ae)(ab)>, <(ae)(abc)> and <(ae)(abc)a> are 49, 33, 41, 25 and 29 respectively. In the maximum utilities, there is no such thing as anti-monotonicity. As a result, given a value of $\xi > 0$, the high utility sequences are unlikely to construct a complete LQS-Tree. For example, for $\xi = 60$, the High Utility Sequential Patterns are {(be)a(ab)}, {ba(ab)}, {(be)aa} and {(be)ab}. Obviously, these four patterns cannot form a complete-LQS-Tree.

Step 2: USpan consequently uses a depth-first search strategy to traverse the LQS-Tree to search for high utility patterns. USpan generates the root's children first, as seen in Figure 7. It then uses <a> as the current node, determines whether '<a >' is a high utility pattern, and searches for <a>'s possible children. If the first children of ' <a >, which are <(ab) >, are not chosen as the current node, the similar procedures will be applied to <(ab)>.

This procedure will be called recursively until there are no more LQS-Tree nodes to visit. It then uses <a> as the current node, checks to see if <a> is a high utility pattern, then searches for <a>'s possible children. If <a>'s first children, i.e., < (ab) >, are not chosen as the current node, the similar operations will be applied to (ab) >. This operation will be called recursively until there are no more LQS-Tree nodes to visit. Sample LQS tree was given in Figure 7.

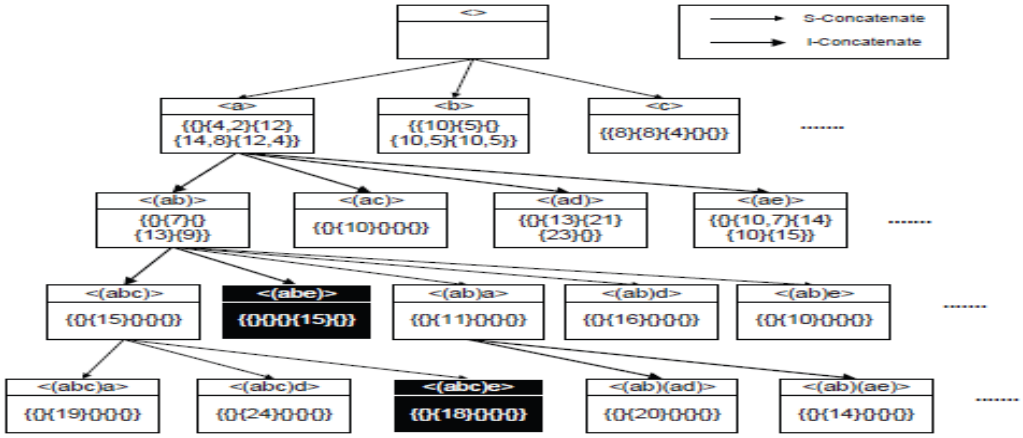


Figure 7: The Complete-LQS-Tree for the Example

Step 3: The I-Concatenation and the S-Concatenations are applied to the LQS Tree.

Step 4: The depth and width pruning techniques are further applied to remove the unpromising candidates from the tree.

Advantage: It follows the bitmap representation like in SPAM algorithm which is suitable for larger datasets.

Limitation: It follows the lexicographic tree construction for generating High Utility Sequential Patterns which are more time consuming.

Other Studies:

IPA: To eliminate the unpromising subsequences and for High Utility Sequential Pattern Mining, (Lan, Hong, Tseng & Wang, 2012) have proposed an Improved Projection-based Algorithm (IPA) with an effective pruning strategy to discover high sequential utility patterns in a quantitative

sequence database. The pruning strategy's main concept is to achieve more accurate upper bounds of sequence utility values of mining patterns after uncompromising items are extracted in the recursive process from sequences. To obtain more accurate upper bounds of sequences, the sequence utility of each modified sequence can be re-calculated.

TUS: For mining Top-k High Utility Sequences (TUS), (Yin et.al, 2013) have proposed a baseline algorithm called TUSNaive Algorithm. Furthermore, three effective strategies are introduced to handle the efficiency problem, including two strategies for raising the utility threshold and one pruning strategy for filtering unpromising items. A sequence t is called a top-k high utility sequence if there are less than k sequences whose utilities are no less than $umax(t)$[maximum utility of a sequence t]. The optimal minimum utility is denoted and defined as $\xi* = \{(t)|t \in \mathcal{T}\}$, where $\mathcal{T}$ means the set of top-k high utility sequences. Given a u-sequence database $\mathcal{S}$and a number k, the problem of finding the complete set of top-k High Utility Sequential Patterns in $\mathcal{S}$ is to discover all the itemsets whose utilities are no less than $\xi*$ in $\mathcal{S}$.

HUSP-NIV: Some sequences of High Utility Sequential Patterns do contain a negative item/utility value (NIV) (e.g. profit). For instance, a retailer sells a cartridge with negative profit at a higher positive return in a package with a printer. While a few techniques have been suggested to mine NIV high utility itemsets (HUI), they are not proper for NIV HUSP mining because an item can occur more than once in a sequence and its utility may have multiple values. The authors(Xu et.al, 2017) have proposed a novel method High Utility Sequential Patterns with Negative Item Values (HUSP-NIV) to efficiently mine HUSP with NIV from sequential utility-based databases. HUSP-NIV works as follows: (1) using the lexicographic quantitative sequence tree (LQS-tree) to extract the complete set of high utility sequences and using I-Concatenation and S-Concatenation mechanisms to generate newly concatenated sequences; (2) using three pruning methods to reduce the search space in the LQStree; (3) traversing LQS-tree and outputting all High Utility Sequential Patterns.

2.6 Studies involving combination of Opinion Mining and Data Mining

Recent studies include integrating Data Mining approaches like Association Rule Mining(Agarwal & Srikant, 1994), Sequential Pattern Mining(Agarwal & Srikant, 1995), Sequential Rule Mining(Fournier-Viger, Nkambou, & Tseng, 2011), High Utility Itemset Mining(Yao et al., 2004), etc., with Opinion Mining of customer reviews in order to achieve good accuracy for extracting

relevant product features from customer opinions/reviews. This section will discuss in detail such systems that have been proposed, which include Data Mining algorithms to obtain relevant and frequent product features.

2.6.1 RashidOFExt: Data Mining Approaches – SPM and ARM by (Rashid et al., 2013)

The authors(Rashid et al., 2013) tried to compare two important and renowned algorithms of Association Rule Mining and Sequential Pattern Mining for frequent features and opinion words extraction from customers' opinions obtained from a social networking website. For this comparison, they used the Apriori algorithm and the Generalized Sequential Pattern(GSP) algorithm on the review's dataset. The dataset used in this experiment is educational student feedback data obtained from an online survey of universities to extract the frequently commented features along with their opinion words. Sentence level sentiment classification is used, which is one level deeper to document level Opinion Mining. It extracts such sentences from review documents that contain an object, noun (just feature words), and adjectives.

Methodology:

Step1: Crawl Reviews: The online Teacher Evaluation surveys conducted by universities are collected. Students give their reviews in comments in free textual format about each professor separately. Such files are considered as reviews dataset in the experiments.

Step 2: Data Preprocessing: Since data is in free format and retrieved from the internet, much irrelevant information such as HTML tags, special characters, false reviews, spelling errors, student's data is omitted from review documents to make it easier to use information further.

Step 3: POS Tagging: As the features must be defined along with their words of opinion, such phrases containing feature and their corresponding adjectives are required. The best option is to perform this function as part of speech tagging.

Example:

Sentence: His teaching methodology is excellent

POS Tagging: His_ PRP teaching_ NN methodology_ NN is_ VBZ excellent _JJ

Replacing feature word and adjective: his_ PRP F_ NN is_ VBZ A_JJ

Step 4: N-gram modeling: An n-gram is a sub-sequence of n items from a given sequence". N-gram modeling is used to convert unstructured data into structured data. Trigram modeling is applied to tagged data to split sentences in a meaningful form.

Step 5: Apply SPM algorithms: Extracted rules are applied to testing data to check whether the rules are applicable or not. Then the best rules are implemented on a pre-processed dataset to extract feature words and opinion words. Both algorithms are applied to selected data to determine which one is the best to achieve the goal.

Apriori Algorithm: Apriori(Agrawal & Srikant, 1994) is an Association Rule Mining algorithm used to extract the valid rules based on the association among attributes.

Support = P (X U Y) / N

Where P (X U Y) = number of times X and Y appear together

N= total number of items

Confidence = P (X U Y) / P (X)

Where P (X U Y) = number of times X and Y appear together

P (X) = number of times X appears in the dataset

Prepared data files are uploaded in the machine learning WEKA tool one by one to mine the best rules. The set parameters are Lower bound min support = 0.04 Metric type =lift Num rule =100 Upper bound min support=0.9 The said parameters applied on all training files to get rules.

F_NNS, IN	NN_F,IN ,DT
DT ,F_NNP,F_NNP	DT ,NN_F,VBZ
IN, F_NNS	DT ,JJ_A,NN_F
F_NNP ,VBZ	DT ,NN_F ,NN_F
F_NNP ,PRP	NN_F ,VBP
F_NNP,F_NNP	NN_F, RB
VBD ,F_NN	DT ,PRP,VBP
F_NNS ,PRP	A_JJ ,TO ,VB
IN ,F_NNP	F_NN ,TO ,VB
IN ,DT,f_NN	A_JJR ,F_NNS ,IN
DT ,f_NNP	CD ,VBG ,A_JJR
JJ ,f_NN	A_JJR ,TO ,VB
f_NN ,VBZ	RBR ,A_JJ ,F_NN
DT ,f_NN	F_NNS,POS ,PRP

f_NN ,CC	F_NNS,TO ,VB
f_NN ,PRP	NN ,A_JJR ,TO
f_NN ,NN	WRB ,DT ,F_NN
A_JJ ,f_NNS	WRB ,PRP$,F_NN
f_NN ,IN	F_NNS ,JJ ,IN
NN ,f_NNS	A_JJ,F_NN ,VBN
f_NN ,DT	CD ,DT ,F_NN
IN ,f_NN	CD ,A_JJS ,RB
DT ,F_NNP ,NNP	CC ,F_NNS ,F_NNS
F_NN ,NN, PRP	CC ,A_JJS,F_NNS
F_NNP ,NNP , IN ,DT	DT ,PRP$,F_NNS

Table 21: Apriori Best Extracted Rules

GSP Algorithm: GSP scans the database several times; all of the frequent itemsets called candidate 1 (C-1) sequence generation are extracted in the first scan. The sequence generation set is built from C-1 candidate 2, and the C-3 sequence is generated from frequent itemsets. Until no frequent item remains, this process is repeated.

1-sequences	**2-sequences**	**3-sequences**
{DT}	{VBN,f_NN}	{f_NNP}{NN,PRP}
{NN}	{VBD,f_NN}	{f_NN}{NN,DT}
{VBZ}	{f_NNS,WRB}	{f_NN}{NN,VBZ}
{IN}	{A_JJ,NN}	{f_NN,NN}{DT}
{f_NNS}	{DT,A_JJ}	{NN_F}{PRP,VBP}
{f_NN}	{NN,A_JJ}	{NN_F}{WDT,VBZ}
{A_JJ}	{VBZ,A_JJ}	{NN_F}{WDT}{WDT}
{DT}	{F_NNS,A_JJ}	{A_JJ,PRP}{PRP}
{NN}	{F_NN,VBZ}	{NN_F}{A_JJ,VBZ}
{VBZ}	{VB,A_JJS}	{NN_F}{PRP,VBZ}
{IN}	{NNS,A_JJ}	{A_JJ}{WDT,VBZ}

Table 22: GSP Best-Extracted Rules

Step 6: Implication of best rules on testing files: On testing files, the best-extracted combinations of feature extraction and adjective rules are applied to extract feature and opinion terms, respectively. The rules are applied to test files one by one, and the confusion matrix parameter is determined for each rule to verify the accuracy of the applied combination. The results are compared, and it is proved that GSP outperforms Apriori while extracting implicit features from textual data.

Advantage: Sentence level Opinion Mining is used to extract the commented frequent features and opinion words from students' feedback dataset in textual free format about faculty evaluation.

The complete cycle of Apriori and GSP is executed to find out an efficient algorithm for extracting features and opinions.

Limitation: Opinion Classification is not done, and other machine learning algorithms can be used to get better results.

2.6.2 Rana18OFExt: Sequential patterns rule-based approach by (Rana & Cheah, 2018)

The research to study the impact of Sequential Pattern Mining in the context of Opinion and Feature extraction was done by (Rana & Cheah, 2018) and proposed a method that yielded better results than other state-of-the-art approaches. Their methodology focused on the features(aspects) that are present in the opinions of customers' reviews. To understand this concept better, the authors in their paper are taking reference of ABOS(Aspect-Based Opinion Summarisation) which is proposed by (Hu & Liu, 2004) where they mine opinion features from customer reviews. In ABOS, three important steps were involved: (1) to identify the aspects (product aspect/feature, in this work, the term 'aspect' is used) for which the customers have expressed their opinions; (2) to identify sentences from within the reviews which give positive or negative opinions about each aspect; and (3) to generate an overall summary based on the extracted information.

In their work, the authors (Rana & Cheah, 2018) have proposed sequential pattern rules-based approach that exploits sequential patterns to find out the association among the aspect and opinion terms and to extract explicit features. The work discussed in this paper focuses only on explicit features and consists of three sections: (1) in the first section, using the PrefixSpan(Pei. et al., 2001) algorithm, sequential patterns are generated; (2) in the second part, certain rules are specified by analyzing sequential patterns produced during the first step on the basis of the correlation between aspect and opinion terms; (3) the explicit aspects are extracted in the third part using the sequential rules described in the second step. In this paper, for the extraction of product aspects, the use of sequential patterns has been proposed. The sequential patterns not only generate direct association patterns, but also generate indirect association patterns between aspects and terms of opinion. The algorithm PrefixSpan is used because only certain patterns in which opinions or features appear as a prefix which are basically important for the experiments.

This implementation is applied on a set of free-format product reviews dataset which is taken from Amazon.com. The major focus of this research is that instead of using any dependency parser, the authors have used *Sequential Pattern Mining algorithm, PrefixSpan* to find association among the

opinions and aspects. The important reason for not using dependency parser is that the dependency parser–based approaches are highly dependent upon the generated parse trees which are generated using some language rules, but in the customer reviews, users do not follow the grammatical rules and neither the language constraints.

Proposed methodology: The proposed methodology carries out feature extraction in three stages: (1) pre-processing and mining sequential patterns, (2) generating sequential rules using mined sequential patterns, and (3) extracting explicit aspects using sequential rules.

Example:

The steps involved for the opinion-target extraction from customer reviews accompanied with a walkthrough example are as follows:

Input: Product Reviews Dataset from any social networking website(in this case, Amazon)

Output: Explicit features or aspects from customer opinions

Review ID	Product review
R1	It's very sleek looking with a very good front panel button layout, and it has a great feature set.
R2	The player usually plays dvds but has occasional problems.
R3	I bought this DVD player and I am using this player from the last 3 months. I am very pleased with this product
R4	I have not even used my new dvd player and already i am disapointed !
R5	This player is perfect for dvds with high pixels and quality.

Table 23: Product Reviews Dataset from Amazon (Rana & Cheah, 2018)

Step 1: The customer reviews dataset for a product is collected and every sentence in a review is preprocessed and tagged using Stanford POS Tagger. Each sentence in the dataset has been marked whether it contains any feature or not. The sentence, which contains at least one aspect, is labelled with all the aspects within the respective sentence and whether they are explicit or implicit. For review id R5, we can annotate sentence as,

Dvdplayer[+2]###This player is perfect for dvds with high pixels and quality

The sentence starts after the symbol '##' while 'Dvdplayer[+2]' represents the annotation used in the datasets. The annotation represents the explicit aspect in the sentence which is 'player' and the polarity of users' opinion against the aspect which is '+2'. It means that the user has expressed his or her positive opinion against aspect 'player'.

Step2: There are many different words in the review sentences that are used by the customers for features and to express their opinions. Hence, much frequent features cannot be generated in this case. Therefore, all the aspect words have been replaced by the word 'aspect' and opinion words with the term 'opinion'. This helped understand association between the features and the opinions through a variety of reviews. For review id R1, three aspects are represented by words '*looking'*, '*front panel layout button*' and '*feature set*' followed by the words '*very sleek'*, '*very good'* and '*great'* which represent the opinions for each aspect, respectively. After replacing the words, following is the resultant tagged sentence using Stanford parser:

/PRP/VBZ Opinion/RB Opinion/JJ Aspect/VBG /IN /DT Opinion/RB Opinion/JJ Aspect/JJ Aspect/NN Aspect/NN Aspect/NN ,/,/CC /PRP /VBZ /DT Opinion/JJ Aspect/NN Aspect/VBN ./.

Step 3: Once all the sentences have been changed accordingly after Step2, PrefixSpan Algorithm is applied with a support of 0.3 and an input of 50 subjective sentences from DVD player review datasets. PrefixSpan results in 537,645 possible sequential patterns and amongst which only 31,350 patterns contained aspect and opinion words. PrefixSpan algorithm generates a huge number of sequential patterns but not all the patterns are relevant. Therefore, patterns are pruned and selected automatically through the following three sub steps:

Patterns Pruning: Patterns which do not contain both aspect and opinion words are irrelevant and are eliminated.

Patterns selection: Only those patterns which have prefix or postfix either aspect or opinion are selected.

Patterns confirmation: In an association among aspect and opinion, there are several possible patterns with the prefix or postfix as aspect and opinion word. From these possible patterns, only one pattern can represent the true relationship of aspect and opinion terms.

Step 4: Sequential rules are generated based on the sequential patterns obtained. These patterns provide an association between features and opinions. Following rules are generated and are distributed into four different classes:

a. Noun/noun phrase association : A=noun/noun phrase, O=Opinion, C=Copula (connecting subject and predicate) PRP=Pronoun, J=subordinating conjunctions, D=conjunctions

IF A∼O THEN aspect=A

e.g., 'very bad quality'== bad-opinion, quality-aspect

IF A∼C AND C∼O THEN aspect=A

Copula contains words like is, are, has, have, etc. These are the auxiliary verbs.

e.g., 'The audio is excellent' == excellent-opinion, audio-aspect

IF not A∼C AND C∼O AND A∼PRP THEN aspect=A

e.g., 'The software you get with the camera is perfect' == perfect-opinion, software-aspect

IF A1∼C AND O∼C AND A2∼O THEN aspect=A1+A2

e.g., 'Apple is a great phone' == great-opinion, Apple phone-aspect

IF A1∼J AND A2∼J THEN aspect=A1+J+A2

e.g., ''I like the quality of the pictures' == like-opinion, quality and pictures-aspect

IF A1∼D AND A2∼D AND(A1 OR A2)∼O THEN aspect1=A1 AND aspect2=A2

e.g., 'its fast-forward and rewind work much more smoothly and consistently than those of other models I've had' == smoothly and consistently-opinion, fast-forward and rewind-aspect

b. Pronoun and demonstrative association: DT=demonstrative

IF (PRP∼O OR DT∼O) THEN search aspect(A)

e.g., 'I've had the player for about 2 years now and it still performs nicely' == nicely-opinion, player-aspect

c. Pronoun 'I' association: PP=Personal Pronoun

IF PP∼O THEN search aspect(A)

e.g., I am very pleased with this product. In this sentence, 'pleased' is the opinion and 'product' is the aspect which appears after the opinion word but consider the sentence, 'I have not even used my new dvd player and already i am disapointed !'. The 'disappointed' (misspelled by the user in the review) is the opinion which occurs after the aspect 'dvd player'. Therefore, if no aspect is identified and the sentence ended, then search the sentence before the personal pronoun 'I' for any potential aspect.

d. Conjunction association: BT=but

IF BT∼O AND search(A) before BT=TRUE THEN aspect=A

e.g., 'the player usually plays dvds but has occasional problems' == problems-opinion, player-aspect

e. Cue phrase association: CP=Cue Phrase such as pros and cons, positive and negative

IF NOT A∼O AND CP=TRUE THEN aspect=A

Step 5: The sequential rules, defined in the previous section, are for nouns/noun phrases. To extract an aspect from the sentence, we first search for any noun in the sentence based on the nouns. If any noun is identified, then we check that either it is a single noun or a noun phrase. The noun phrases are extracted by identifying compound nouns as produced by the Stanford Parser. For example, in the following tagged sentence, the compound nouns are 'picture' and 'quality' and hence both are extracted as a single noun phrase.

Picture/NN quality/NN is/VBZ excellent/JJ

Once any noun/noun phrase is identified with the sentence, then we used the sequential rules to identify any opinion word. If any opinion word is identified, then the noun/noun phrase is collected as an aspect, otherwise the extracted noun/noun phrase is discarded.

Advantage: This research specifically focused on explicit aspects/features. It used Sequential Pattern Mining and Sequential Rules generation for feature extraction.

Limitation: This approach using the sequential patterns can be extended to identify all possible aspect and opinion associations and to use these patterns to identify implicit aspects.

2.6.3 HPFG19_HU by (Demir et al., 2019)

The authors (Demir et al., 2019) proposed a method to extract feature groups. In this method, they have tried to combine Aspect-Based(Feature Based) Sentiment Analysis, triples-to-transactions transformation, and high utility itemset mining. The input to the system is a set of product reviews and the output is set of feature groups that yield high profit considering the utility factor.

In this work, the authors present an application of High Utility Itemset Mining using Sentiment Analysis. The HPFG19_HU system mines customer reviews to identify the most important aspect groups for a service or product. The system used aspect-targeted sentiment scores as utility, with an aim of identifying the top aspect sets that can lead to the highest levels of customer satisfaction. In this system, the authors mine itemsets of opinion sentences in a transaction database. They consider sentiment score obtained from SentiStrength library and considers external utility values as identical values(=1).

Problem Statement: Given a product, or a product family, such as a mobile phone of a particular make and model, HPFG19_HU system considers a set of features, such as shape, weight, color, or price. The number and the nature of the features may vary depending on the product. A group of features may be more preferable by either customer or producer. The sentiments expressed by users about features are important signals of preference (i.e., profit), and this can be considered as utilities of the features.

In this respect, the authors (Demir et al., 2019) proposed a system called HPFG19_HU to extract such feature groups. The approach combines Aspect-Based Sentiment Analysis and High Utility Pattern Mining. The method consists of three main steps: Aspect-Based Sentiment Analysis, triples-to-transactions transformation, and high utility pattern mining.

Aspect-Based Sentiment Analysis: In this step, aspects and sentiments are extracted from review sentences and triples are formed. Sentiment Score is calculated using SentiStrength lexicon and a triple of <review, aspect, sentiment score> is created in the form (r_i, a_j, sc) such that in review r_i, aspect a_j has sentiment score *sc*.

Following steps have been followed to achieve considerable results:

Step1: Each product review is parsed into sentences. The opinion and features of the sentences is extracted with a feature(aspect)-based sentiment analyzer.

Step 2: On each sentence, NLP Tokenizer has been employed in order to get tokens. Each noun and noun phrase obtained has been considered as a candidate feature.

Step 3: Sentiment words have been annotated with the help of a sentiment lexicon.

Step 4: Annotated nouns, noun phrases and opinion words have been arranged in the order of their occurrence.

Step 5: A triplet has been created for each co-occurring noun and sentiment word pairs. Sentiment score has been assigned using a sentiment lexicon.

Step 6: Union of triplet sets has been created for each of the sentences present in the review.

Triples-to-transaction transformation: A review is equivalent to a transaction, and aspects derived from a review in the first phase are equivalent to transaction objects in the third. Sentiment score of each feature in a review correlates to the item's internal utility in the transaction. For each item, external utility is also required. According to the authors (Demir et al., 2019), this value is not available in the review data and is instead a domain-specific value, such as a customer's preference or a producer's preference due to low production costs. Hence it is considered as identical(=1).

High Utility Itemset Mining: The system uses FHN (Faster High-Utility itemset miner with Negative unit profits) algorithm(Lin, Fournier-Viger, & Gan,2016) to extract high utility itemsets which will serve as potentially High Profit Utility Aspects(HUA). FHN algorithm is used as in this case, utility values can be either positive or negative, corresponding to positive or negative polarity in the sentiment

Example:

Input: Samples of product reviews of a smart phone

Output: High Utility Feature Groups

Sample	Review Text
1	Good looking cover and fits perfect. Seems to be of good quality and really protects the phone at a great price
2	They look good and stick good! I just dont like the rounded shape because I was always bumping it and Siri kept popping up and it was irritating.
3	This *product* is great. I like the kickstand on the back. The power indicator is very convenient to know charge pack status

Table 24: Product Reviews Dataset

Step 1: Aspect-Based Sentiment Analysis: Sample reviews are presented in Table 24. Corresponding triples, which are extracted through applying the above-mentioned process, are given in Table 25.

Review	Triples
1	*{quality, good, 2}, {price, great, 3}*
2	*{shape, like,−2}, {shape, irritating,−3}*
3	*{kickstand, like, 2}*

Table 25: Sample triples extracted from reviews in Table 25

Step 2: Triples to Transaction Transformation: review id is now considered as transaction id, where each aspect is considered as an item in the itemset of the transaction.

Review	Transaction
1	*{quality : 2, price : 3}*
2	*{shape : −5}*
3	*{kickstand : 2}*

Table 26:Sample transactions corresponding to reviews in Table 25

Step 3: High Utility Itemset Mining: FHN algorithm(Lin, Fournier-Viger, & Gan,2016) is used to extract High Utility Aspect(HUA) Groups for the reviews in Table 25.

Positive aspect sets obtained from the above transactions after high utility itemset mining for the product cover are <*quality*, *price*, *kickstand*>.

Negative aspect sets obtained from the above transactions after high utility itemset mining for the product cover are <*shape*>.

Advantage: combines Aspect-Based Sentiment Analysis and high utility itemset mining.

Limitation: Forming sequences or sequential patterns instead of itemsets can help in getting more relevant feature-sets. Sequential Feature Groups might help in getting a better input for recommendation and identify similar types of users. Also including external utility values can get accurate results.

Other Studies:

(Ghorashi, et al., 2012) proposed a frequent pattern mining algorithm to extract product features from a bunch of reviews available from social media. They proposed this method to outperform the old pattern mining techniques. According to the authors, Opinion Mining or Sentiment Analysis helps to assess whether a positive, negative, or neutral orientation is delivered by the review sentences. The extraction of product features is important for Sentiment Analysis since the recognition of opinion orientation is significantly influenced by the target features. The major focus in this research is on the features that have received more opinions from the reviewers. H-Mine algorithm(Pei et.al., 2001) is applied for frequent feature extraction that outperforms the work of (Hu & Liu, 2010) who used Association Rule Mining for the same. This has enhanced the precision and performance of the system simultaneously.

(Nurrahmi, Maharani, & Saadah, 2016) proposed a system that was able to automatically extract product features and opinions from the reviews using Class Sequential Rule (CSR) method. This method was initially used by (Hu & Liu, 2006) for opinion feature extraction. In this study, a high accuracy for feature extraction was achieved but the product reviews used were already separated as positive and negative. (Wen & Wan, 2014) used Opinion Lexicon Method for Opinion Classification from extracted features. They used it for emotion classification on microblog texts from Twitter and achieved high accuracy as well. Hence, in this paper, the authors(Nurrahmi, Maharani, & Saadah, 2016) have used CSR method and Opinion Lexicon method to extract the features from product reviews in free format and tried to improve the accuracy of opinion classification.

CHAPTER 3: THE PROPOSED HIGH PROFIT SEQUENTIAL FEATURE GROUPS BASED ON HIGH UTILITY SEQUENCES (HPSFG_HUS) SYSTEM FOR OPINIONS MINED FROM PRODUCT FEATURES

High Profit Sequential Feature Groups are a set of product features grouped as a whole in order of their occurrences(sequences) that yield high profit in the market to the manufactures of the product and are responsible for higher consumer satisfaction.

3.1 Problem Definition

Given a set of reviews R of a product P as an input, the problem being addressed in this thesis is to identify P's features (*shape, size, color, camera quality, or price)* and their opinions (*positive, negative*) (Feature-Based Opinion Mining). Then, these features are grouped to form feature-sets and generate potentially high utility/profit sequential feature groups (High Utility Sequential Pattern Mining) from the extracted features.

Feature-Based Opinion Mining: The goal of this task is to ***extract features and its opinions*** of the reviewed item e.g., long batterylife, good camera, etc.

High Utility Sequential Pattern Mining: This task aims at determining the High Profit Feature Groups (set of features as a whole) by forming ***high utility/profit sequences*** e.g., < [batterylife, camera, price], [batterylife, camera]>, < [batterylife, camera], [batterylife]>

3.2 Proposed HPSFG_HUS System

The major goal of the proposed High Profit Sequential Feature Groups based on High Utility Sequences System is to form sequences of features that yield high profit based on High Utility Sequential Pattern Mining. This approach is an enhancement of the existing system called HPFG19_HU (Demir et al., 2019) that used High Utility Itemset Mining(HUIM) using FHN algorithm (Lin, Fournier-Viger, & Gan,2016) to mine frequent high utility patterns with positive and negative unit profit values. This existing system combined Aspect-Based Sentiment Analysis and HUIM to discover high profit feature sets by taking a transaction database. This system considers the internal utility values as sentiment scores (positive or negative) and considers identical external utility values(=1). The approach uses itemsets of aspects to generate feature groups. For example, {*shape, size, color, camera-quality*} of a smartphone. These feature groups

obtained from customer reviews will help the retailers to know about the users' preferences. Since this system performs high utility itemset mining in a transaction database of features, it does not take the order of occurrences of aspects words into account. This means that the sequential ordering of feature words of reviews is not considered. There may be frequent occurrences of these feature-groups or individual aspects that can form sequences or patterns.

Use- Case: Sequential patterns stand a better chance to identify important product features that can be related to other aspects in the form of price, importance, customer preference, etc. Customers might be attracted to such related feature groups and moreover it can help the retailer to identify similar users from these patterns. Finding sequences of high profit product feature groups/aspects terms from a group of users, we can say these users are alike, we consider those users similar from opinions. Finding similar users will help the retailers to identify the pattern of features-sets in which they are interested. Learning patterns of similar users also help in identifying what suggestions to give to that group of users. For example, User A might like sequence of (feature-group 1, feature-group 2, feature-group 3) as mentioned in their product reviews. This can help to determine that User B who is like User A in choice or preferences might at least like sequence of (feature-group 1, feature-group 2). Hence such sequences of feature sets can serve as an input to Recommendation Systems and help businesspeople understand the relative high profit feature groups which will increase revenue and customer satisfaction.

HPSFG_HUS System Architecture:

In the **Figure 8**, we can understand how proposed HPSFG_HUS System is different from existing HPFG19_HU System(Demir et al., 2019). The overall architecture of the proposed HPSFG_HUS system is presented in the following **Figure 9**. As seen in the flowchart, the input to the overall methodology is a dataset of product reviews under consideration available on the social network websites like Amazon(www.amazon.com). The output is a collection of High Utility Sequential Patterns such that each pattern is considered as a set of sequential features that have the potential to generate profit if preferred together. In the figure, each box corresponds to one of the necessary steps.

Figure to show how the HPFG19_HU system is different from the proposed High Profit Sequential Feature Groups based on High Utility Sequences (HPSFG_HUS) System.

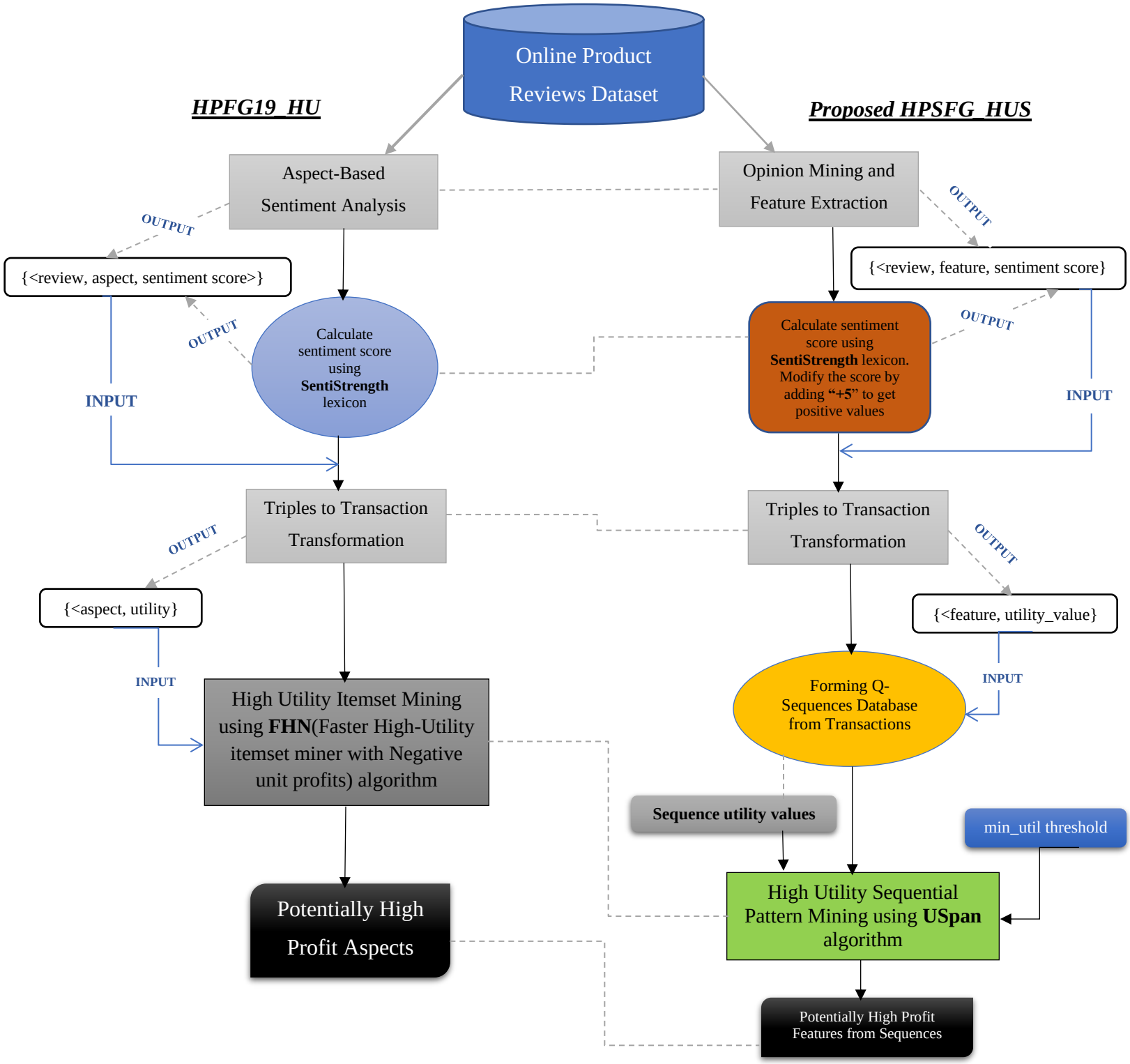


Figure 8: The major differences between the HPFG19_HU and proposed HPSFG_HUS systems

Flowchart of the proposed High Profit Sequential Feature Groups based on High Utility Sequences (proposed HPSFG_HUS) System

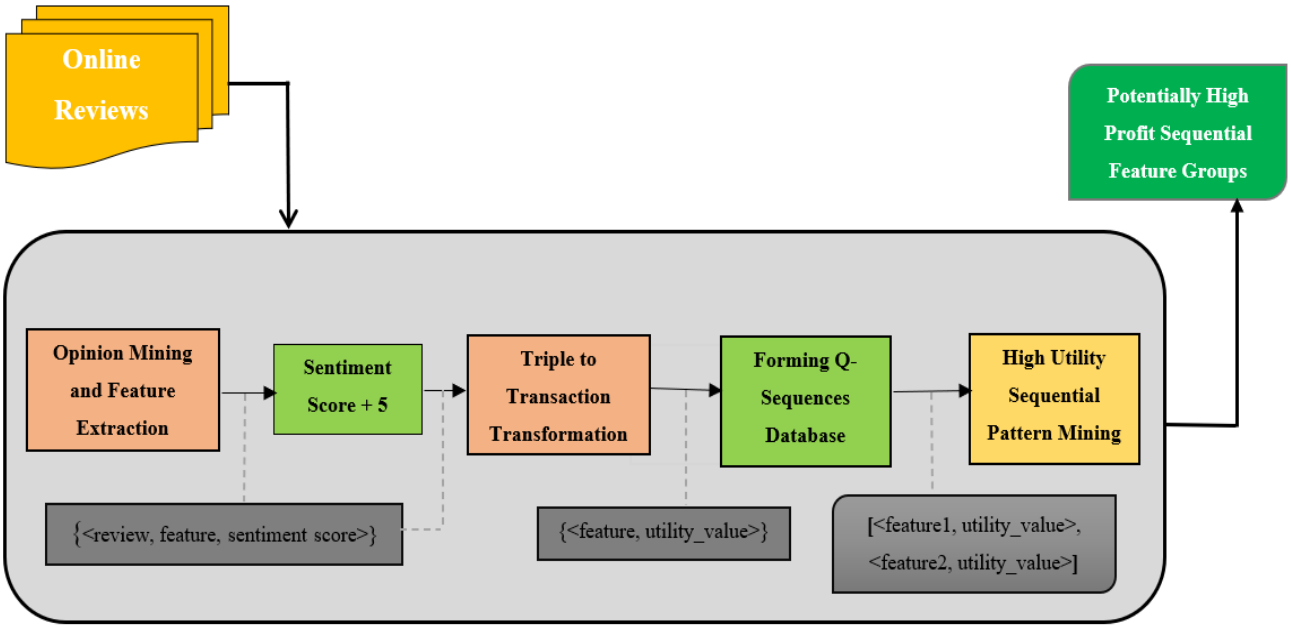


Figure 9: Flowchart of the proposed HPSFG_HUS System

Proposed Methodology Outline: The main algorithm of the proposed HPSFG_HUS System is presented in **Algorithm 1** in section 3.3, with 4 different stages. In the rest of this section, we are explaining the internals of each stage. Algorithm 2, 3, 4 in the respective sections 3.4, 3.5, 3.6 explains the working of each stage with examples.

3.3 Proposed *HPSFG_HUS* System's Main Algorithm

Algorithm 1: High Profit Sequential Feature Groups based on High Utility Sequences (HPSFG_HUS)

Input:	Online text reviews dataset for Products obtained from the social networking websites
Intermediate Inputs:	Features f_i, Opinions o_j, Sentiment Scores *sc*, Q-Sequences, High Utility Sequential Rules (HUSR), utility values(uv), minimum utility threshold(min_util), sequence utility values(su)
Output:	High Profit Sequential Feature Groups

Procedure:

BEGIN:
STAGE 1. FEATURE-BASED OPINION MINING

1: ***Collect product reviews*** dataset of a Product P from an online reviews' website: $R \leftarrow Reviews$. *Initialize* T $\leftarrow \emptyset$

2: ***for*** $r \in R$ ***do***

i. Perform data cleaning and preprocessing steps on the product reviews dataset R using **Algorithm 2** in **section 3.4**

ii. Extract features f_i, Opinions o_i, Sentiment Scores *sc* using **Algorithm 2** in **section 3.4** and form triples: $TR \leftarrow ExtractTriples(r)$

iii. Triples formed for each review r in R are unioned in T : $T \leftarrow T \cup TR$

End

STAGE 2: TRIPLES-TO-TRANSACTION TRANSFORMATION:

1: Modify each sentiment score for each triple by adding "**+5**" to convert it into a positive value

2: Construct a Transaction Database D of itemsets with the modified triples**.**

$D \leftarrow ConstructTransactionDatabase(TR)$

STAGE 3: FORMING Q-SEQUENCES FROM TRANSACTION DATABASE

1: Construct a Q-Sequence database from the sequence of itemsets and calculate sequence utility for each sequence using **Algorithm 3** in **section 3.5**

Given D, group transactions based on the occurrence of sentences in a review.

S ← ConstructQ-SequenceDatabase(D)

STAGE 4. HIGH UTILITY SEQUENTIAL PATTERN MINING:

Set the minimum utility threshold using total Sequence utility value of Q-Sequence database and obtain High Profit Sequential Feature Groups, HPSFG, by applying the USpan Algorithm(Yin, Zheng & Cao, 2012), which gives High Utility Sequential Patterns using **Algorithm 4** in **section 3.6**

HPSFG ← ApplyUspan(S)

Algorithm 1: Main Algorithm for HPSFG_HUS

Steps in the proposed HPSFG_HUS system:

Input: Set of Reviews Dataset for Product P (**Table 27**)

Output: High Profit Feature Groups based on High Utility Sequences

ReviewID	Review Text
1	The iphone11 pro has an amazing batterylife. It has a good quality. For such an outstanding battery life, the price is great! 😊
2	The phone comes with 3 lens and has beautiful camera quality. The charger is fast. It makes battery life longer in a good price.
3	I just dont like the shape because I was always bumping it and Siri kept popping up and it was irritating.
4	People who speak with me say voice quality is great. Battery life is good as well and the price is good.
5	These make using the home button easy. I like the longer battery life. Well worth the price

Table 27:Product Reviews Dataset

STAGE 1: FEATURE-BASED OPINION MINING

Step 1.1: Data Preprocessing of each Review: Parse each review present in the product reviews dataset (**Table 27**) and perform the data cleaning and preprocessing steps as mentioned in **Algorithm 2** (**Section 3.4**). At the end of this step, we get cleaned and preprocessed reviews

without any unwanted special characters, stopwords(is, the was, etc), whitespaces, punctuations, and emoticons. Lemmatization, Stemming and Tokenization as explained in (**Section 2.1**) is performed and we get preprocessed reviews as shown in **Table 28.**

ReviewID	Preprocessed Review Text
1	pro amazing batterylife good quality outstanding battery life price great
2	phone come lens beautiful camera quality charger fast make battery life longer good price
3	dont like shape always bump siri keep pop irritate
4	people speak say voice quality great battery life good well price good
5	make use home button easy like long battery life well worth price

Table 28: Cleaned and Preprocessed Reviews

Step 1.2 : Extracting Features, Opinions and calculating Sentiment Score:

In this step, as shown in **Algorithm 2** in **section 3.4,** extract nouns(e.g., quality), noun phrases(e.g., camera quality), and nouns having possessive forms(e.g., phone's charger) as features. Extract the corresponding sentiment words(adjectives, adverbs) with the features as sentiments using a sentiment lexicon(SentiStrength). Calculate sentiment score for each review using SentiStrength. Form Feature-Opinion pairs for each co-occurring noun and sentiment pairs as shown in **Table 29.**

ReviewID	Feature-Opinions
1	{batterylife, amazing}, {quality, good}, {batterylife, outstanding}, {price, great}
2	{quality, beautiful}, {cameraquality, beautiful}, {batterylife, long}, {price, good}
3	{shape, like}, {shape, irritating}
4	{quality, great}, {voicequality, great}, {batterylife, good}, {price, good},
5	{button, easy}, {homebutton, easy}, {batterylife, long}, {price, well}

Table 29: Feature-Opinion Pairs

Step 1.3: Forming Triples: In this sub-step, as shown in **Algorithm 2** in **section 3.4,** triples(feature, opinion, sentimentscore) are formed with noun/noun phrases, sentiments and sentiment score extracted in the previous step. Triples are formed as shown in **Table 30.**

ReviewID	Features-Opinions
1	{batterylife, amazing, 1}, {quality, good, 2}, {batterylife, outstanding, 4}, {price, great, 4}

2	{quality, beautiful, 2}, {cameraquality, beautiful, 2}, {batterylife, long, 3}, {price, good, 2}
3	{shape, like, -2}, {shape, irritating,-3}
4	{quality,great, 4}, {voicequality,great, 4}, {batterylife, good, 2}, {price, good, 2},
5	{button, easy, 1}, {homebutton, easy, 1}, {batterylife, long, 3}, , {price, well, 1}

Table 30*:* Triples

STAGE 2: TRIPLES-TO-TRANSACTION TRANSFORMATION:

Step 2.1: In our HPSFG_HUS system, we modify the sentiment score by adding **'+5'** to each of the sentiment score of each review in order to normalize the score and get a positive value because **'+5'** is the highest sentiment score value. This positive value will be helpful in further stages in order to get the high profit/utility sequences.

ReviewID	Features-Opinions
1	{batterylife, amazing, 6}, {quality, good, 7}, {batterylife, outstanding, 9}, {price, great, 9}
2	{quality, beautiful, 7}, {cameraquality, beautiful, 7}, {batterylife, long, 8}, {price, good, 7}
3	{shape, like, 3}, {shape, irritating, 2}
4	{quality,great, 9}, {voicequality,great, 9}, {batterylife, good, 7}, {price, good, 7},
5	{button, easy, 6}, {homebutton, easy, 6}, {batterylife, long, 8}, , {price,well, 6}

Table 31: Triples with modified sentiment score

Step 2.2: Further construct Transaction Database of itemsets with the triples considering the feature and sentiment score(as utility value) as shown in **Table 32** with the transformation table as mentioned in **Table 34** in **section 3.5.**

TransactionID	Features-Opinions
1	{(batterylife: 6), (quality:7), (batterylife: 9), (price:9)}
2	{(quality:7), (cameraquality:7), (batterylife:8), (price: 7)}
3	{(shape:3), (shape:2)}
4	{(quality:9), (voicequality:9), (batterylife:7), (price: 7)}
5	{(button:6), (homebutton:6), (batterylife:8), (price: 6)}

Table 32: Transaction Database D of itemsets

STAGE 3: FORMING Q-SEQUENCE DATABASE

Construct a Q-Sequence database as shown in **Table 33** from the sequence of itemsets(occurring in the order of sentences) and calculate sequence utility(sum of utility values of all q-sequences of each review) for each sequence using **Algorithm 3** in **section 3.6** for each review. Calculate Total Sequence Utility value at the end(sum of all sequence utilities of reviews)

For example, for ReviewID 1: sequence utility = 6+7+9+9 = 31

SequenceID	Features-Opinions	Sequence Utility
1	<(batterylife: 6), (quality:7), [(batterylife: 9), (price:9)]>	31
2	<[(quality:7), (cameraquality:7)], [(batterylife:8), (price: 7)]>	29
3	<(shape:3), (shape:2)>	5
4	<[(quality:9),(voicequality:9)], [(batterylife:7), (price: 7)]>	32
5	<[(button:6), (homebutton:6)], (batterylife:8), (price: 6)>	26

Table 33: Q-Sequence Database

Total Sequence Utility = 31+29+5+32+26 **=** 123

STAGE 4: HIGH UTILITY SEQUENTIAL PATTERN MINING

Set the minimum utility threshold with respect to Total Sequence utility value of Q-Sequence database and obtain High Profit Sequential Feature Groups HPSFG, by applying the USpan Algorithm(Yin, Zheng & Cao, 2012) as mentioned in **Algorithm 4** in **section 3.7**, which gives High Utility Sequential Patterns.

For example, ***min_util =*** *10% = 0.1*123 = 12.3*

High Profit Feature Groups based on High Utility Sequential Patterns: All the Q-Sequences having Q-Sequence utility > 12.3 are extracted. For example, for Q-Sequence, *<[(batterylife), (price)]>*, the utility values of this sequence = 9+9 = 18 >12.3

Final Output of High Utility Sequences: *<[(quality), (cameraquality)]>, <[(batterylife), (price)] > , <[(quality),(voicequality)]>*

3.4 Feature-Based Opinion Mining Module

As per our ***Algorithm 1***, ***in Stage 1***, we mine opinions from the product reviews, extract the features, sentiments and obtain the corresponding sentiment score values and form triples. It can be demonstrated as follows:

"*Given a set of text reviews of a product, we obtain triples of* (f_i o_j, *sc*), *where,* o_j *is the opinion associated with feature* f_i *having sentiment score sc."*

For our approach, we only rely on the tuple (f_i, *sc*) containing the feature and its corresponding sentiment score. On the other hand, we need to retain the information from which product review this tuple is extracted in the following steps.

Aspect-Based Sentiment Analyzer (Demir et al., 2019) is used in this stage to extract the product features. SentiStrength library is used for sentiment *sc.*

Algorithm : To obtain the triples of feature, opinion(sentiment), and sentiment score

Input : Product Reviews Dataset

Variables : r – reviews and T – Transactions

Output : Triples (r_i o_j, *sc*)

START

1. Initialize R=Set of Product Reviews, T=∅
2. **FOR** each review r ∈ R **DO**:
3. *TR* ← *ExtractTriplets(r)*
4. *T* ← *T* ∪ *TR*
5. ***End***
6. ***ExtractTriples(r)***
7. Parse review *r* into sentences
8. **FOR** each *sentence* in *review r* by applying NLP Tokenizer **DO**:

- Annotate each noun, noun phrase and possessive nouns as candidate aspect
- Transform each word of noun or noun phrase to its lemmatized form
- Concatenate the lemmatized words
- Annotate sentiment words using sentiment lexicon
- Order the annotated noun, noun phrases and sentiments on their occurrence

9. **FOR** each co-occurring noun-sentiment pair **DO**:

10. **IF** sentiment is between nouns or noun phrases **DO**:

○ Match sentiment with both noun and noun phrases

• Calculate sentiment score *sc* using SentiStrength library

➢ Form Triple TR = Triple(f_i o_j, *sc*)

11. ***return TR***

STOP

Algorithm 2: Algorithm for Aspect-Based Sentiment Analyzer

Example of Feature-Based Opinion Mining and Forming Triples

To understand the working of **Algorithm 2**, we let us consider Product Reviews Dataset (**Table 27**) as input. We will get triples(f_i o_j, *sc*) as an output of this module

Step 1: Data Preprocessing of reviews

Parsing the dataset and preprocessing and cleaning the dataset using the Stanford CoreNLP library. We clean the dataset by executing the following preprocessing steps. The detailed explanation, working and examples of the data cleaning and preprocessing tasks is shown in **Section 2.1**. Then we extract the opinion and feature words from the cleaned reviews.

Step 1.1: Clean all the sentences in each of the reviews by lowercasing the text and removing whitespaces, punctuations, stopwords, emoticons and special characters. Tokenization, stemming, and lemmatization steps are performed on the given text. These cleaning and preprocessing steps are performed using the Stanford CoreNLP library(https://stanfordnlp.github.io/CoreNLP/).

Input: Set of Reviews (Review R1, for instance)

R1: The iphone11 pro has an amazing batterylife. It has a good quality. For such an outstanding battery life, the price is great! 😊

Output: Cleaned Reviews (Tokens of extracted nouns and adjectives)

R1: ['pro', 'amazing', 'batterylife', 'good', 'quality', 'outstanding', 'battery', 'life', 'price', 'great'**]**

Step 1.2 Apply POS(Part of Speech) Tags to the cleaned reviews. All the types of POS Tags with their abbreviations are shown in **Table 11** of **Section 2.1**

Input: Cleaned Reviews from Step 1.2 (Review R1, for instance)

Output: POSTagged Reviews

R1: [('iphone', 'JJ'), ('amazing', 'JJ'), ('batterylife', 'NP'), ('good', 'JJ'), ('quality', 'NN'), ('outstanding', 'JJ'), ('battery', 'NN'), ('life', 'NN'), ('price', 'NN'), ('great', 'JJ')]

Step 1.3: The POSTags with Nouns(NN) and Noun Phrases(NP) will serve as *features* and the POSTags having adjectives(JJ) and adverbs(RB) will serve as *opinions/sentiments*. This sentiment extraction is done using sentiment lexicon called SentiStrength(http://sentistrength.wlv.ac.uk/). ***SentiStrength*** is a sentiment lexicon that helps to identify strength of positive or negative sentiment word and hence assign sentiment scores.

***-1** (not negative) to -**5** (extremely negative)*

***1** (not positive) to **5** (extremely positive)*

Order the annotated noun, noun phrases and sentiments on their occurrence. For each co-occurring noun-sentiment word pairs, match sentiment with both noun and noun phrases, if sentiment is between nouns or noun phrases.

Input: POSTagged Reviews from Step 1.2 (Review R1, for instance)

Output: Noun-Sentiment Pair

R1: {batterylife, amazing}, {quality, good}, {batterylife, outstanding}, {price, great}

Step 2: Forming Triples

To form the triples, using a sentiment lexicon, the corresponding sentiment score of the *sentiment word* is calculated and allocated to the feature. If sentiment word is annotated as negated, then score assignment is adjusted accordingly. Triples of *feature, opinion, sentimentscore* (f_i o_j, sc) are formed by noun-sentiment pairs(**Step 1.3**), sentiment score for each sentence of each review.

Input: Noun-Sentiment Pair from Step 1.3 (Review R1, for instance)

Output: Triples (f_i o_j, sc)

R1: {batterylife, amazing, 1}, {quality, good, 2}, {batterylife, outstanding, 4}, {price, great, 4}

3.5 Triples-to-Transaction Transformation Module:

Triples extracted from **Stage 1** are now transformed into transactions in this module. Since we have to perform Utility Based Pattern Mining, it is important to have utility values and hence these transactions formed will have them.

Internal Utility: Internal utility of the item corresponds to sentiment score of each feature

External Utility: In the product reviews, external utility is considered constant(=1) in the case as it is a domain-dependent value like customer or manufacturer preference

In our proposed HPSFG_HUS system, we are going to deal with only profit values, so in contrast to the existing HPFG19_HU system, we add **"+5"** to each of the sentiment score of each review in order to normalize the score and get a positive value because **'+5'** is the highest sentiment score value. This positive value will be useful in further **Stage 4** in order to get the high profit/utility sequences.

Input: Triples formed in Stage 1(Review R1, for instance)

Output: Triples with modified sentiment score

R1: {batterylife, amazing, 6}, {quality, good, 7}, {batterylife, outstanding, 9}, {price, great, 9}

Since our ultimate goal is to yield High Utility Feature Groups with profit values, we consider the product's "**rating**" as our deciding factor. The product ratings range from 1 to 5, and lower values of the product's rating will not give profit. So, we consider the rating > 3 as positive and discard the transactions having overall rating < 3 as negative. The transformation model is taken according to the model proposed by (Demir et al., 2019). Below is the transformation model for the transaction of itemsets with utility values:

Utility-Based Pattern Mining	**Feature-Based Opinion Mining**
Item	Feature
Transaction	Review
Utility	Feature's sentiment score * domain dependent utility value
External Utility	Domain dependent utility value
Internal Utility	Feature's sentiment score

Table 34: Triples to Transaction Transformation Model (Demir et al., 2019)

Forming Transaction Database(D) of itemsets:

To form a transaction with the triples, the review id is now considered transaction id, and each feature is considered an item in the itemset of transaction. Transaction Database formed with the itemsets after the transformation module is shown in **Table 32.**

3.6 Forming Q-Sequence Database

A review R of every product may/may not consist of multiple sentences. Therefore, each itemset of transaction obtained from the previous stage 2 is considered as a q-itemset and transaction id is now considered as sequence id. Hence a q-sequence database will have q-itemsets grouped according to their order of occurrence in the sentences. Once a Q-Sequence database is formed containing q-itemsets, Sequence Utility value is calculated which is the sum of all the utility values of each q-itemset in the database. The database will have q-sequences constructed from q-itemsets that are obtained from transactions.

Algorithm: To form a Q-Sequence Database

Input: Transaction Database(D) of itemsets(I) of reviews, Product Reviews Dataset(R)

Variables: Seq: Q-Sequence, Sub_Seq: Q-SubSequence, SU: Sequence Utility, TSU: Total Sequence Utility

Output: Q-Sequence Database

```
START
Initialize Seq ← ∅, Sub_Seq ← ∅, SU ←0, TSU←0
for each sentence s in review r € R do
    for each itemset I in Transaction T of Database D do
        if item i occurs in a sentence
            Sub_Seq = Sub_Seq.append(itemset I)
            Seq= Seq.add(Sub_Seq)
        else
            Seq = Seq.add(itemset I)
    SU = Sum of utility values of each itemset I in Sequence Seq
TSU = Sum(SU)
```

Algorithm 3: Algorithm to form Q-Sequence Database S

Q-Sequence Database(S) with the q-sequences formed from Transaction Database D having itemsets is shown in **Table 33.**

Example: Sequence Utility(SU), let us consider for **SequenceID 1:**

<(batterylife: 6), (quality:7), [(batterylife: 9), (price:9)]>. Hence, SU(S1): 6+7+9+9 = 31

Total Sequence Utility(TSU) for **Table 33**: TSU(S) = 31+29+5+32+26 = 123

3.7 High Utility Sequential Pattern Mining:

Once the q-sequences database is constructed, and we have positive utility values of features; we use USpan (Yin, Zheng & Cao, 2012) algorithm that will give High Utility Sequential Patterns. The Q-sequence Database is constructed in format that is applicable for USpan algorithm available at SPMF library (https://www.philippe-fournier-viger.com/spmf/USpan.php). For minimum sequence utility threshold, δ, it is a user-defined threshold value to obtain the desired number of Sequential Patterns and is given by the percentage of sequence utility value of the database.

Algorithm: To extract High Utility Sequential Patterns **Input:** A Q-sequence Database(S), TSU(Total Sequence Utility), min_util **Variables:** min_util, δ*: user-defined threshold* **Output:** High Utility Sequential Patterns
min_util = $\delta \times TSU$ High Utility Sequential Patterns will be obtained based on the minimum utility threshold *HPSFG* ← *ApplyUSpan(S)*

Algorithm 4: Extracting High Utility Sequential Patterns

Hence, High Profit Sequential Feature Groups(HPSFG) are extracted by applying the USpan algorithm on Q-sequence database S based on the factor that the Sequence Utility of the given sequences should be greater than the minimum utility threshold provided.

Example: let us consider the sequence: **<[*(batterylife), (price)]*>,** in **Table 33**, it occurs in **SequenceID 1, 2 and 4.** Let's say, the user-defined threshold ***δ = 10%***

So, min_util = δ * *TSU* = 0.10*123 = 12.3

Hence for sequence **<[*(batterylife), (price)]*>,**

SequenceID 1: [(batterylife: 9), (price:9)], utility value = 9+9=18

SequenceID 2: [(batterylife:8), (price: 7)], utility value = 8+7=15

SequenceID 4: [(batterylife:7), (price: 7)], utility value = 7+7 = 14

Max Utility Value: 18, 18 > 12.3(min_util), hence **<[*(batterylife), (price)]*>** is a High Utility Sequential Pattern.

Overall Profit of Sequence **<[*(batterylife), (price)]*> =** 18+15+14 = 47. Hence, it can be said as High Profit Sequential Feature.

3.8 Extracting Potential High Profit Sequential Feature Groups:

The High Utility Sequential Patterns that are obtained from the Q-sequence database S will yield High Profit Sequential Feature Groups. These feature groups will contain the sequential patterns that are of more importance from a consumer's perspective. Such High Profit Sequential Feature Groups will help to decide the upcoming product releases. Customers or end-users can identify multiple brands or services in terms of their best and worst feature sets and use the data to determine which one to choose. This comparison can also be made by their overall utility concerning one or more features under interest using the item or itemset utilities in the sequence database. The appropriate decision can be made by rating these feature sets. In terms of interesting features or complaints, i.e., features, the latter could address the question as to which is the best choice that brings out the highest customer satisfaction. Through the proposed High Profit Sequential Feature Groups approach, producers or service providers may discover their strong sequential features and get to know the interestingness of those features or the features that are mostly talked about. In other words, they will understand what to continue to do and what to enhance. They will direct their potential investments by taking advantage of this research.

***Use Cases**:*

- Let's consider a high utility sequence: <[(price), (batterylife)], (batterylife)> Considering a given time frame, if market value of one feature ***batterylife*** goes up, there can be a possibility that the importance of all the feature-groups containing ***'batterylife'*** may have higher customer-preference and thus we can say that these features are related. *Hence, we obtain such related feature groups because of sequences.* These related Sequential Feature Groups can

identify interested similar users and can serve as a better input to the Recommender Systems based on the preferences of similar users. They help identify any learning patterns of such similar users.

- Let's say there are three promotion positions available on the shelf, then HUSPM can be used to discover the patterns with the highest utility. Assume one of the patterns is <[(batterylife, sound)], [(batterylife, sound, camera)], (camera)>, decision-makers can put *batterylife* and *sound* on sale and then arrange camera into promotion position for cross-marketing based on the mining results.
- For example, In Udemy, a course learning website, we can say that data science students might be interested in learning some particular topics of courses that are in patterns based on the level of their education.

A walk-through example for comparing HPFG19_HU and proposed HPSFG_HUS Systems

For comparison, we will use the similar example as given in paper of HPFG19_HU System by (Demir et al., 2019). We will compare the results of both the systems(HPFG19_HU and proposed HPSFG_HUS) and show the output generated by each in a table. We will also show how our system outperforms in terms of accuracy and relevancy in extracting High Profit Feature Groups.

Input: Product Reviews Dataset(Table 35)

Output: High Profit Feature Groups

Sample	Review Text
1	Good looking cover and fits perfect. Seems to be of good quality and really protects the phone at a great price
2	I use this with a Motorola Android phone. It works very well. I have no connection problems. People who speak with me say voice quality is great. People complain about some other headsets I have, so this one is good. Battery life is good as well. One of the best features of this headset, which I have not seen in others, is that it tells you with a womans voice that when it turns on, off, establishes connection, and gives you updates on battery life (just says "Battery high, medium, or low"). I really like this headset
3	They look good and stick good! I just dont like the rounded shape because I was always bumping it and Siri kept popping up and it was irritating. I just wont buy a product like this again

4	This product is great. I like the kickstand on the back. The power indicator is very convenient to know charge pack status
5	These make using the home button easy. My daughter and I both like them. I would purchase them again. Well worth the price

Table 35: Online Product Reviews (Demir et al.,2019)

Comparison table of HPFG19_HU and proposed HPSFG_HUS Systems:

<table>
<tr><th>Steps of HPFG19_HU</th><th>Steps of proposed HPSFG_HUS</th></tr>
<tr><td>Step 1: Extract opinion and aspect words from the review sentences.
Input: Table 35 Product reviews dataset
Output: <aspect, sentiment>

<table>
<tr><td>ReviewID</td><td>Aspects-Sentiments</td></tr>
<tr><td>1</td><td>{quality, good}, {price, great}</td></tr>
<tr><td>2</td><td>{quality, great}, {voicequality, great}, {batterylife, good}, {feature, best}</td></tr>
<tr><td>3</td><td>{shape, like}, {shape, irritating}</td></tr>
<tr><td>4</td><td>{kickstand, like}</td></tr>
<tr><td>5</td><td>{button, easy}, {homebutton, easy}, {price,well}</td></tr>
</table>
Table 36: Aspects-Sentiments Table after Step 1</td><td>Step 1: Extract opinion and feature from the review sentences. Preserve the order of the feature words while extraction.
Input: Table 35 Product reviews dataset
Output: <feature, opinion>

<table>
<tr><td>ReviewID</td><td>Features-Opinions</td></tr>
<tr><td>1</td><td>{quality, good}, {price, great}</td></tr>
<tr><td>2</td><td>{quality, great}, {voicequality, great}, {batterylife, good}, {feature, best}</td></tr>
<tr><td>3</td><td>{shape, like}, {shape, irritating}</td></tr>
<tr><td>4</td><td>{kickstand, like}</td></tr>
<tr><td>5</td><td>{button, easy}, {homebutton, easy}, {price,well}</td></tr>
</table>
Table 37: Features-Opinions Table after Step 1</td></tr>
<tr><td colspan="2">Output is same after Step 1</td></tr>
<tr><td>Step 2: Calculate sentiment score for each aspect using SentiStrength. Assign the score with each pair and form a triple.
Input: Table 36 Aspect-Sentiment Pairs
Output: Triples for each pair of Aspect-Sentiment

<table>
<tr><td>ReviewID</td><td>Aspects-Sentiments</td></tr>
<tr><td>1</td><td>{quality, good, 2}, {price, great, 3}</td></tr>
<tr><td>2</td><td>{quality, great, 3}, {voicequality, great, 3}, {batterylife, good, 2}, {feature, best, 2}</td></tr>
</table></td><td>Step 2: Calculate sentiment score for each feature using SentiStrength. Assign the score with each pair and form a triple. Add +5 to the score of each triple
Input: Table 37 Feature-Opinion Pairs
Output: Triples for each pair of Feature-Opinion

<table>
<tr><td>ReviewID</td><td>Features-Opinions</td></tr>
<tr><td>1</td><td>{quality, good, 7}, {price, great, 8}</td></tr>
<tr><td>2</td><td>{quality, great, 8}, {voicequality, great, 8}, {batterylife, good, 7}, {feature, best, 7}</td></tr>
</table></td></tr>
</table>

3	{shape, like,−2}, {shape, irritating,−3}
4	{kickstand, like, 2}
5	{button, easy, 1}, {homebutton, easy, 1}, {price,well, 1}

Table 38: Triples formed after Step 2 for review and aspects

3	{shape, like, 3}, {shape, irritating,2}
4	{kickstand, like, 7 }
5	{button, easy, 6}, {homebutton, easy, 6}, {price,well, 6}

Table 39: Triples formed after Step 2 for review and features

Step 3: Convert the triples obtained after Step 2 to transaction by forming itemsets in a transaction database. Follow the conversion from Table 28. Consider the sentiment score as internal utility and external utility =1. The triples are grouped together if a feature occurs more than once in the same review and the sentiment score is adjusted accordingly.

Input: Table 38

Output: Itemsets in a transaction database

TID	Transactions
1	{quality : 2, price : 3}
2	{quality : 3, voicequality : 3, batterylife : 2, feature : 2}
3	{shape : −5}
4	{kickstand : 2}
5	{button : 1, homebutton : 1, price : 1}

Table 40: Triples to Transaction by forming itemsets of aspects

Step 3: Convert the triples obtained after Step 2 to transaction by forming itemsets in a transaction database. Follow the conversion from Table 28. Consider the sentiment score as internal utility and external utility =1. Do not group the triples and consider each item as individual itemsets.

Input: Table 39

Output: Itemsets in a transaction database

TID	Transactions
1	{quality : 7}, {price : 8}
2	{quality : 8}, {voicequality : 8}, {batterylife : 7}, {feature : 7}
3	{shape, 5}
4	{kickstand : 7}
5	{button : 6}, {homebutton : 6}, {price : 6}

Table 41: Triples to Transaction by forming itemsets of features

Step 4: High Utility Aspect Groups are extracted from the itemsets formed using FHN algorithm.

Input: Table 40

Output: High Utility Aspect Groups(HUA)

Top Aspect Set with positive utility: *<quality, price, voicequality>*

Top Aspect Set with negative utility: *<shape>*

Step 4: Form Q-sequences from itemsets and calculate sequence utility for each Q-Sequence. Form the Q-sequence database.

Input: Table 41

Output: Q-sequence database

	<table><tr><th>SID</th><th>Opinion-Features</th><th>Sequence Utility</th></tr><tr><td>1</td><td>< [(quality :7), (price: 8)]></td><td>15</td></tr><tr><td>2</td><td><(quality: 8), (voicequality: 8), [(batterylife: 7), (feature :7)]></td><td>30</td></tr><tr><td>3</td><td><(shape : 5)></td><td>5</td></tr><tr><td>4</td><td><(kickstand : 7)></td><td>7</td></tr><tr><td>5</td><td><[(button : 6) , (homebutton : 6)], (price : 6)></td><td>18</td></tr></table> Table 42: Q-Sequence database of opinion features Total Sequence Utility: 75
	Step 5: High Profit Feature Groups are extracted from the Q-sequences using USpan(Yin, Zheng & Cao, 2012) algorithm. All the sequences having utility values > min_util value as considered as High Utility Sequences. Specify the minimum utility threshold value to extract High Utility Sequential Patterns. Note that this threshold value is selected by user of the program to yield High Profit patterns. In this case, we consider threshold value as 10%. ***Input:*** Table 42, *min_util = 0.1*75 = 7.5* ***Output:***High Profit Sequential Feature Groups(HPSFG) **High Profit Sequential Feature Groups based on High Utility Sequences:** *<[(quality), (price)]>, <(quality)>, <(voicequality)>, <[(batterylife), (feature)]>, <[(button) , (homebutton)]>*

CHAPTER 4: EXPERIMENTS AND ANALYSIS

Evaluation Analysis: This chapter discusses the implementation details and experiments performed to evaluate our proposed HPSFG_HUS system's effectiveness in terms of Precision, Accuracy, Recall and F1-Score in mining the high utility features of the product with respect to the different minimum utility threshold values. We also compare the execution times of the working of the proposed algorithm with respect to different minimum utility threshold values in finding HPSFG(High Profit Sequential Feature Groups). The details of how the experiments are conducted and results are obtained is discussed in the **section 4.2**

Comparison Analysis: This chapter also shows analysis of how the proposed HPSFG_HUS system is more efficient than previously existing HPFG19_HU approach and the baseline approaches. The existing HPFG19_HU algorithm works on the itemsets of transaction databases, not with the sequential databases. Also, the HPFG19_HU system generates the non-sequential high utility patterns. The other baseline algorithms (**section 4.3**) used for comparison also generate itemsets as features. So, it quite difficult to compare with the proposed framework because in the proposed system we generate sequences of ordered features that are occurring in the reviews. Hence, we mainly compare ***High Utility Itemset Patterns*** generated by HPFG19_HU system and ***High Utility Sequential Patterns*** generated by the proposed HPSFG_HUS system.

Implementation Details:

To implement the proposed HPSFG_HUS system, we have used the following tools and infrastructure:

i) System Configuration: Windows 10, with 16 GB RAM and 64-bit Operating System, x64 based processor.
ii) Integrated Development Environment, such as Eclipse Java EE IDE for Web Developers, Jupyter Notebook
iii) Programming Languages: Java SE Development Kit (13.0 version) and Python (3.7.0)

4.1Datasets Selection and Information:

We will use the Amazon Product Reviews data extracted from Amazon (www.Amazon.com). The datasets are used for the evaluation and comparison analysis of the proposed solution as shown in the following Table 43.

Dataset Name	Source	Number of reviews
Cellphones and Accessories	Amazon	194439
Musical Instruments	Amazon	10261

Table 43: Dataset Table

Dataset Description: Cellphones and Accessories; Musical Instruments from Amazon: https://nijianmo.github.io/amazon/index.html (Ni et al., 2019)

- reviewerID - ID of the reviewer, e.g. A2SUAM1J3GNN3B
- asin - ID of the product, e.g., 0000013714
- reviewerName - name of the reviewer
- vote - helpful votes of the review
- style - a dictionary of the product metadata, e.g., "Format" is "Hardcover"
- reviewText - text of the review
- overall - rating of the product
- summary - summary of the review
- unixReviewTime - time of the review (unix time)
- reviewTime - time of the review (raw)
- image - images that users post after they have received the product

Note: We classified the reviews of our datasets based on the field "overall" and considered the reviews having rating > 3 as positive reviews and the reviews having rating < 3 are considered negative. The negative reviews were discarded as for this research we are only interested in obtaining High Profit Sequential Feature Groups values which will show the interesting of sequences of features that the customers/reviewers have talked about the most.

4.2Evaluation Analysis of HPSFG_HUS System

We use the following baseline algorithms and HPFG19_HU algorithm to compare the results obtained by our HPSFG_HUS System.

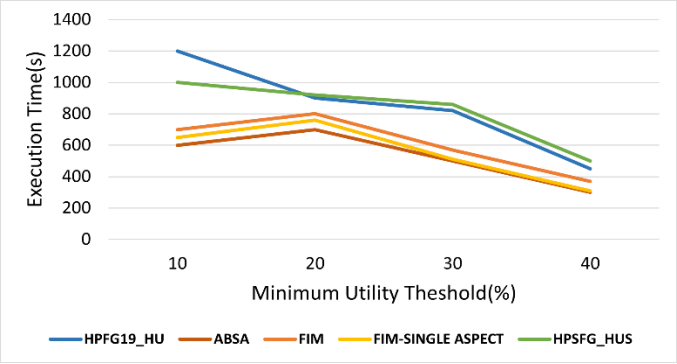
- ***Aspect-Based Sentiment Analysis (ABSA):*** We obtain the execution time required to obtain the aspects/features and sentiment scores and forming feature groups for the given datasets along with calculating the evaluation metrics.

- ***Frequent Itemset Mining (FIM):*** We obtain the results of evaluation metrics and execution time required to obtain the frequent itemsets of features for the given datasets
- ***One-item Frequent Itemset:*** We obtain the execution time required to obtain the one item frequent features and forming feature groups for the given datasets along with calculating the evaluation metrics.
- ***Extracting Feature Groups HPFG19_HU System:*** We obtain the execution time required to obtain the High Utility Feature Groups of itemsets for the given datasets along with calculating the evaluation metrics.
- ***Extracting Sequential Feature Groups HPSFG_HUS (Proposed System):*** We obtain the execution time required to obtain the High Profit Sequential Feature Groups of sequences of features for the given datasets along with calculating the evaluation metrics.

Effect of Minimum Sequential Utility Threshold on Execution Time: In this section, we will evaluate the performance of HPSFG_HUS system in terms of execution time with respect to different minimum utility thresholds. Since there is no execution time provided by the existing algorithm HPFG19_HUS, we evaluate the performance of our proposed system, HPSFG_HUS using multiple utility values on datasets. The total number of transactions for Cellphones and Accessories dataset is 117894 and the total number of unique features are 411, whereas the total number of transactions for Musical instruments dataset is 8367 and the total number of unique features are 461. These generated results of execution time can be further used as a baseline in future work.

Results and Discussion:

Cellphones and Accessories Dataset



Musical Instruments Dataset

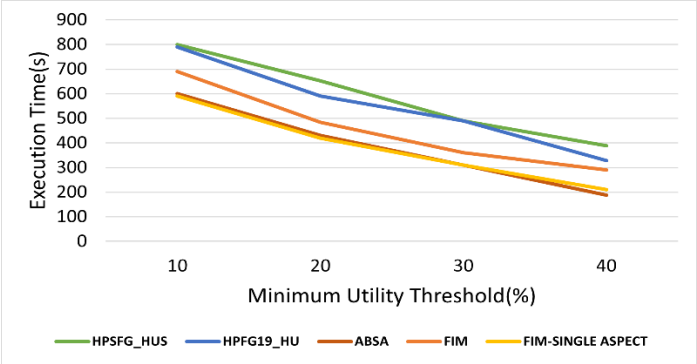


Figure 10: Execution Time v/s Minimum Utility Threshold

A. The graphs show that the execution time for generating high profit sequences drops as the minimum utility threshold increases, and that when the minimum utility threshold decreases, more execution time is required because we may generate many more High Utility Sequential Patterns. The findings also suggest that USpan may extract high utility sequences with a low minimum utility.
B. From the graphs, it can also be seen that in comparison to other existing and baseline algorithms, the execution time of our proposed system HPSFG_HUS is more for all the datasets. Extra work is required in forming Q-Sequences. And the major performance time is required by USpan to generate high profit sequences in comparison to the time required for extracting the features/aspects.

Evaluation Metrics for HPSFG_HUS System: In this section, we will evaluate the performance of HPSFG_HUS system with respect to the existing HPFG19_HU System and other baseline algorithms in terms of accuracy, precision recall and F1-Score. The datasets are divided in the ratio of 80:20 for training and testing, respectively. Since there are no evaluation metrics of the models provided by the existing algorithm HPFG19_HU, we measure the performance of all the models including the proposed HPSFG_HUS system on all datasets in our system configurations, which can be further used as a baseline in future work.

The formulas for the evaluation methods are given below:

True Positives: It means when the model predicted YES, and the actual output was also YES(Powers, 2020).

True Negatives: It means when the model predicted NO, and the actual output was NO(Powers, 2020).

False Positives: It means when the model predicted YES, and the actual output was NO(Powers, 2020).

False Negatives: It means when the model predicted NO, and the actual output was YES(Powers, 2020).

Accuracy: It measures all the correctly identified cases (Goutte & Gaussier, 2005).

$$Accuracy(Acc) = \frac{True\ Positive + True\ Negative}{Total\ Input}$$

Precision: It measures the correctly identified positive cases from all the predicted positive cases. It is important when the costs of False Positives are high(Goutte & Gaussier, 2005).

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

Recall: It measures the correctly identified positive cases from all the actual positive cases. It is important when the cost of False Negatives is high (Goutte & Gaussier, 2005).

$$Recall = \frac{True\ Positive}{True\ Positive\ +\ False\ Negative}$$

F1-Score: It is the harmonic mean of Precision and Recall (Goutte & Gaussier, 2005).

$$F1 - Score = \frac{2 * Precision * Recall}{Precision + Recall}$$

Results and Discussion:

Cellphones and Accessories				
Algorithms	**Accuracy (in %)**	**Precision (in %)**	**Recall (in %)**	**F1-Score (in %)**
Aspect-Based Sentiment Analysis (ABSA):	79.123	78.657	74.967	76.306
Frequent Itemset Mining (FIM):	77.532	76.122	73.124	76.989
One-item Frequent Itemset:	78.980	77.456	75.145	75.989
HPFG19_HU System	75.673	74.547	75.222	74.695
Proposed HPSFG_HUS System	77.672	76.129	75.489	75.807
Musical Instruments				
Algorithms	**Accuracy (in %)**	**Precision (in %)**	**Recall (in %)**	**F1-Score (in %)**
Aspect-Based Sentiment Analysis (ABSA):	84.563	83.123	82.784	82.345
Frequent Itemset Mining (FIM):	82.895	81.023	81.234	81.322
One-item Frequent Itemset:	83.989	82.783	81.673	81.234
HPFG19_HU System	81.524	81.012	79.306	78.123
Proposed HPSFG_HUS System	83.234	81.481	80.456	80.965

Table 44: Results of Evaluation Metrics

From the table 44, it is clear that the values of accuracy and precision for obtaining High Profit Sequential Feature Groups is higher than the existing close HPFG19_HU System that forms itemsets of features. However, the evaluation metrics show a slightly good results for the baselines ABSA and FIM-Single Aspect because only single features/aspects are obtained as a result instead of High Profit Feature Groups. FIM shows considerable results. From these results, it can be seen that our HPSFG_HUS system outperforms the existing HPFG19_HU System by giving High Profit Sequences of Features instead of High Profit Itemsets for all the datasets.

4.3Comparison Analysis of HPSFG_HUS System

Experiments Evaluation and Results Discussion:

- ***Aspect-Based Sentiment Analysis (ABSA):*** We analyze the utilities provided by the aspects' sentiment scores. Note that, this basically corresponds to one item high utility patterns.
- ***Frequent Itemset Mining (FIM):*** We analyze the utilities provided by the frequent feature sets.
- ***One-item Frequent Itemset:*** We analyze the performance of frequent single features in terms of utility gain.
- ***Extracting Feature Groups HPFG19_HU System:*** We analyze the itemsets obtained as feature sets in terms of utility gain.
- ***Extracting Sequential Feature Groups HPSFG_HUS (Proposed System):*** We analyze the sequential patterns obtained as high profit feature sets in terms of utility gain

1) **Analyzing the Accumulated Utility Performances:** In this experiment, we compare the accumulated utilities' values under increasing number of top patterns for each algorithm.

- **HPSFG_HUS:** These patterns are set of sequences of features with high utility values i.e., high sentiment value having sequences in the feature groups. These groups are called potentially High Profit Sequential Feature Groups
- **HPFG19_HU:** These patterns are itemsets of aspect sets with high utility values having high sentiment values.
- **ABSA:** The extracted patterns are single aspects with high sentiment scores.
- **FIM and FIM-Single Aspect: The** extracted patterns are frequent aspect sets that frequently appear together in review database. In FIM-Single Aspect, we particularly analyze the utility performance under single-item sets.

The experimental results are displayed in Figure 11. From the graphs, it is clear that the proposed HPSFG_HUS produces top High Utility Sequences with USpan algorithm which identifies high utility patterns with increasing number of accumulated utility values. These patterns are top high utility sequences of features in contrast with HPFG19_HU and other algorithms that produces itemsets of features. The number of high utility sequences exhibit an almost similar behaviour to

itemset patterns for top 25 positive utility patterns extracted as seen from the graphs because the transaction count is same for all the reviews. These extracted sequences have a maximum length of 4 which means they can also have one feature sequence to 4 features in each sequence with the increasing value of utility values.

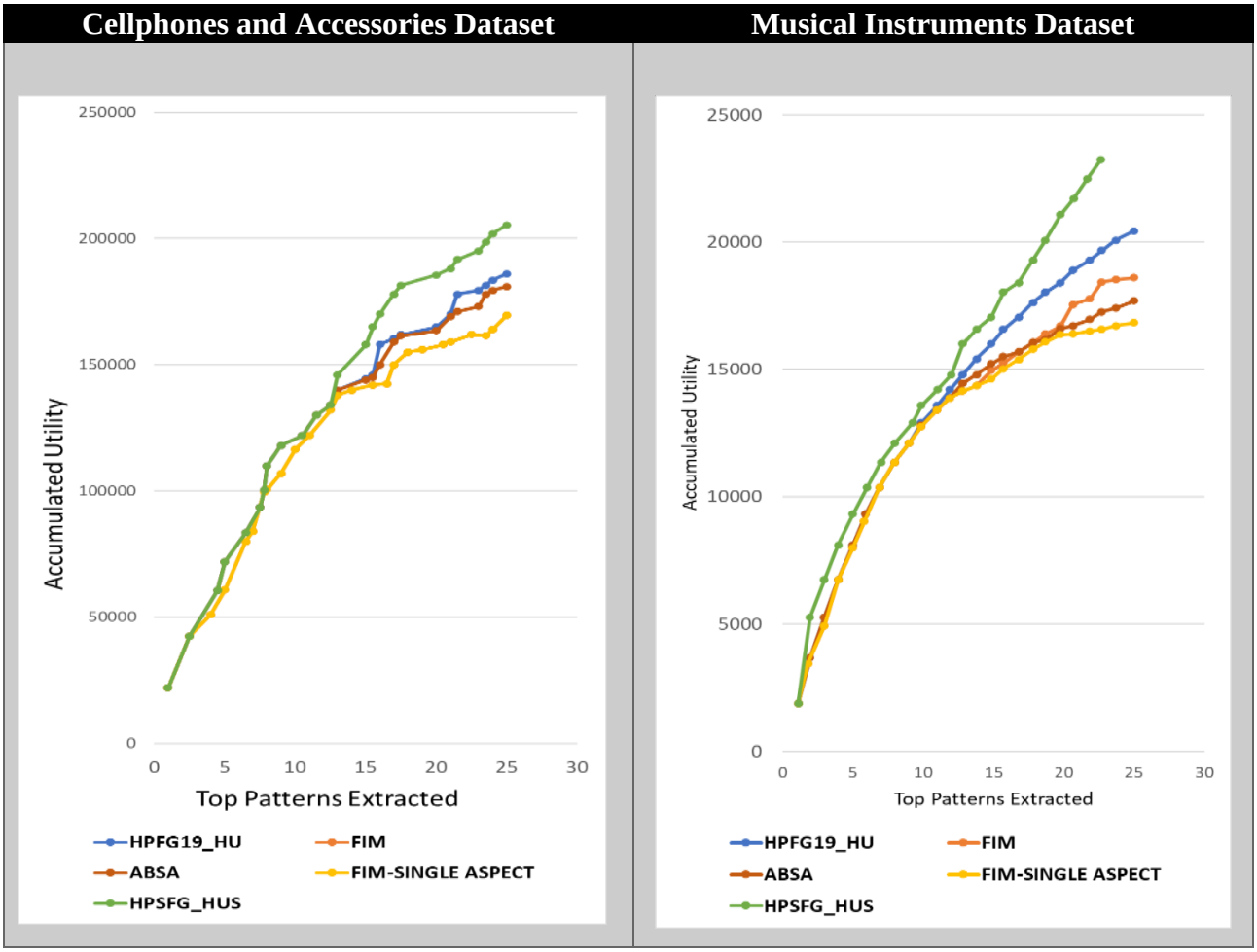


Figure 11: Top Patterns Extracted with Accumulated Utility Values

The top sequential patterns extracted are with the higher values of accumulated utility values have individual sequence feature and group of multiple sequential features which clarify the interestingness(people have talked most about in the reviews) of the feature and hence denotes high profit values of the top feature sets in terms of sequences instead of multiple individual features. From the experiments, it is also observed that accumulation of utility values gets higher by the proposed method than that of baselines.

2) **Support v/s Utility correlation:** We will analyze the correlation between utility and the support of the sequential patterns generated by the proposed method. This is performed for top

25 sequential feature groups. Utility values of sequential feature sets have been identified. Later, we also calculated support values for each feature in those feature sets. Then we dump feature sets onto figures with their Support values on y axis and Utility values on x axis.

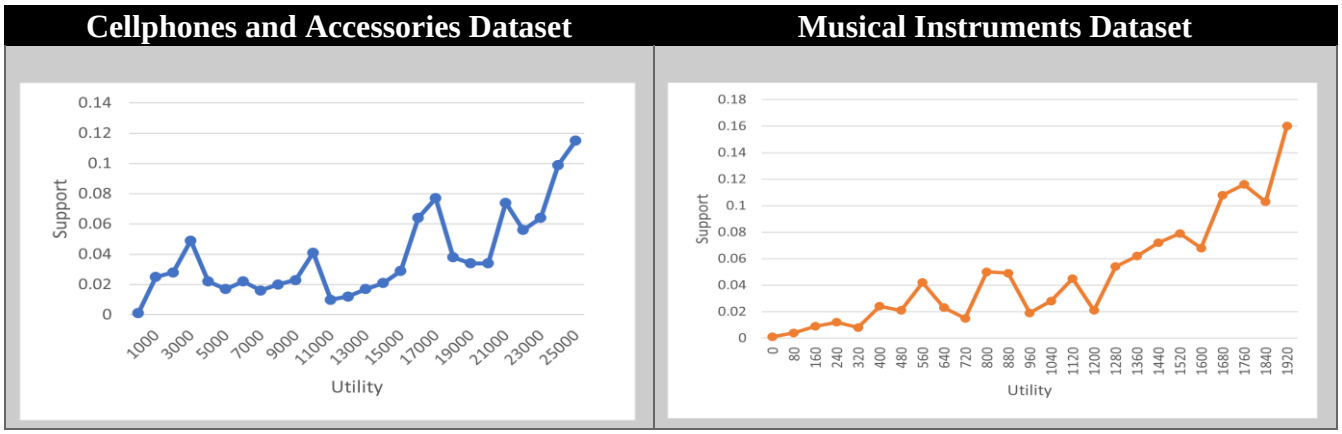


Figure 12: Sequential Feature Sets plotted for Support v/s Utility values Correlation

As can be seen from the figures, a clear correlation between support and utility cannot be observed from the experiment results of all two datasets. Hence, we observe that the pattern extraction through support does not guarantee finding high profit patterns. Any statistic correlation cannot be particularly identified from the figures. Feature sets lie on the figures arbitrarily. So, we can say that utilities can add some value over talking about supports, because they form independently.

3) **Support v/s utility values for top sequential feature groups:** In this experiment, we plot the top 15 high utility sequences of features with the support and utility values. We compare results with the existing algorithm HPFG19_HU that produces high utility itemsets of features. In the figures, the patterns are displayed on the x-axis and the bars show support and utility.

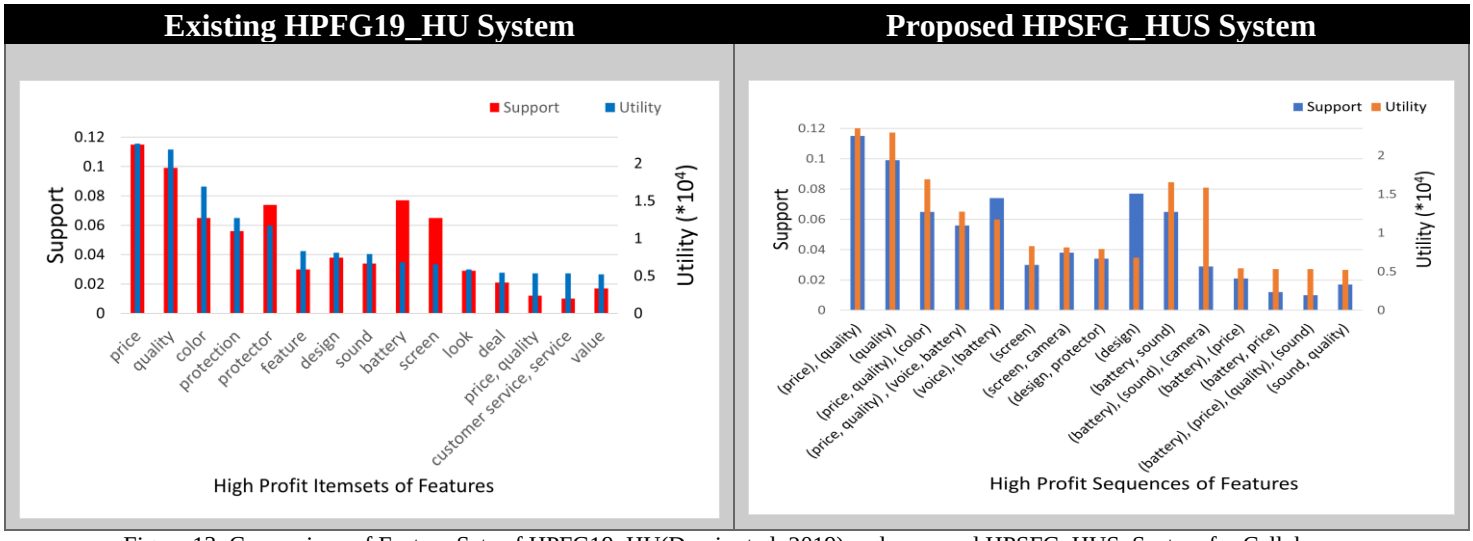


Figure 13: Comparison of Feature Sets of HPFG19_HU(Demir et.al, 2019) and proposed HPSFG_HUS System for Cellphones and Accessories dataset

From the figures 13 and 14, it can be observed that there is no regular trend for support values, and it is consistent with the results. From the graphs, we can see that for the existing HPFG19_HU system, top patterns extracted are the itemsets of features and these are mostly single items. For our proposed system, the sequences of features are obtained as results. We can see that the features will be similar, but the only difference is that in our case we are extracting **sequential patterns of features** instead of single items or multiple feature items in one itemset. This shows that High Utility Aspects do not necessarily show High Utility Sequences of Features.

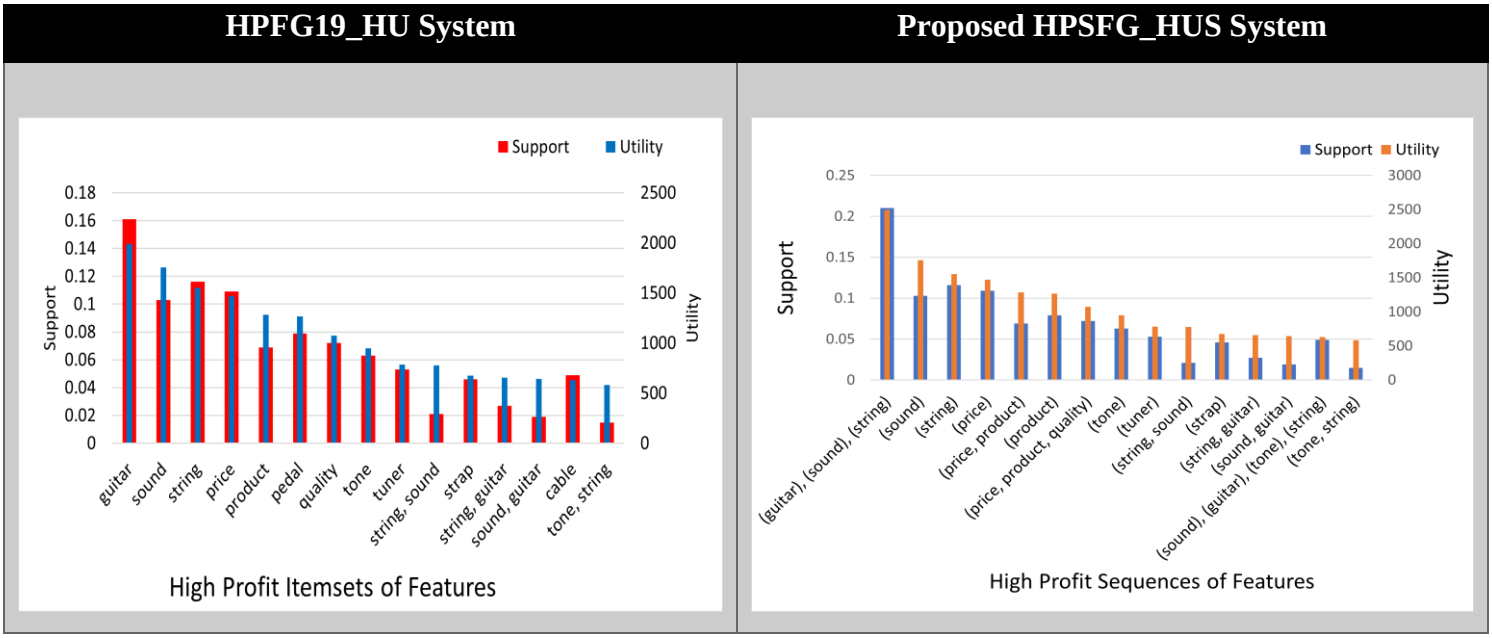


Figure 14: Comparison of Feature Sets of HPFG19_HU(Demir et.al, 2019) and proposed HPSFG_HUS System for Musical Instruments dataset

So, for feature groups that are mostly single items indicate that there are particular features that provide high customer satisfaction. For the proposed HPSFG_HUS System, some of the features have high support value, as well. This is an expected result since an item's total utility increases with the increase in support. On the other hand, the features/feature groups with high support value, but comparatively lower utility value may indicate that they have been mentioned frequently, but the expressed sentiments are either not very strong, or not very consistent (i.e., there are both positive and negative sentiment expressions). As the reverse case, feature groups with comparatively lower support but having high utility are those that have higher potential for focusing on. Such feature groups are not mentioned very frequently, but they carry strong sentiment expressions.

CHAPTER 5: CONCLUSION AND FUTURE WORK

With the increase in the use of social media, we have opinions available online on the e-commerce or social networking websites like Amazon, Twitter, Epinions, etc. for all the available products. Opinions play a major role in influencing customers as well as manufactures. By following the comments posted by users, one can get invaluable information on products. Recently, Data Mining approaches have been incorporated to extract such opinions and features of the product. So, on the basis of this idea, we proposed a method to combine Social Network Opinion Mining and High Utility Sequential Pattern Mining to extract high profit sequential feature groups for a given product or product family. Given a set of product reviews, the output is a set of preferable(and hence potentially high profit) set of sequential features called High Profit Sequential Feature Groups(HPSFG) that are extracted on the basis of high utility sequences(HPSFG_HUS). The system will provide feature-sets which will increase customer satisfaction rather than individual aspects or multiple aspect groups. Further we get frequent high utility sequences in the patterns and hence frequent features as well as high profit sequential features are extracted. The extracted feature groups have utility values more than the minimum threshold sequential utility which ensures that the proposed HPSFG_HUS system suggests feature-sets that could help product sellers to increase their revenue generation by making profit sales. We have compared our proposed HPSFG_HUS system with the existing systems like HPFG19_HU system and baseline algorithms of Frequent Itemset Mining on the same dataset. We try to improvise the existing system HPFG19_HU (Demir et al., 2019), by getting relevant high profit sequences and frequent features instead of high profit itemsets that serve as high profit features which increase sales-profit. Furthermore, we have evaluated our system on the basis of Precision, Accuracy, recall and F1 score. This will serve as a better input in recommendation systems. Even the number of feature-sets suggested are more in the proposed HPSFG_HUS system. Therefore, the proposed HPSFG_HUS system gives better results with a High Utility Sequential Pattern Mining based e-commerce recommendations.

Below are some interesting extensions of this study and some avenues to explore for future works:

1. The current approach, HPSFG_HUS only deals with positive utility values and moreover is constrained to find only high profit feature groups. We can enhance this method to deal with both

positive and negative utility values and hence obtain both high profit and high loss sequential feature groups.

2. Since we are using USpan, we provide min_util threshold value by trial-and-error method. This value can be obtained dynamically or through parameter tuning methods. We can also explore other High Utility Sequential Pattern Mining algorithms with this approach and compare the results.

3. In addition, building a recommender on top of extracted feature groups enables to generate recommendations according to the feedback users have provided through reviews and analyzed through the proposed technique. Extracted features that can be potentially preferred by the users can be recommended in addition to recommending an item on its own.

4. Multiple large data sources can be incorporated based on the High Utility Sequential Pattern Mining algorithms which have different data schemas and also make recommendations based on the overall dataset.

REFERENCES

- Aggarwal, C. C., & Zhai, C. (Eds.). (2012). *Mining text data*. Springer Science & Business Media.
- Agarwal, R., & Srikant, R. (1994). Fast algorithms for mining association rules. In Proc. of the 20th VLDB Conference, 487-499.
- Baccianella, S., Esuli, A., & Sebastiani, F. (2010). *Sentiwordnet 3.0: an enhanced lexical resource for Sentiment Analysis and opinion mining*. In Lrec (Vol. 10, No. 2010, 2200-2204).
- Bai, X. (2011). *Predicting consumer sentiments from online text*. Decision Support Systems, 50(4), 732-742.
- Barthwal, M. (2020*). Why Product Reviews are Important in eCommerce?* Retrieved January 31, 2021, from https://www.knowband.com/blog/ecommerce-blog/product-reviews-importance/#:~:text=Product%20reviews%20are%20the%20opinions,the%20product%20before%20purchasing%20it.
- Clean and ready to use structured datasets(2020). Retrieved June 17, 2021, from https://crawlfeeds.com/.
- Das, B., & Chakraborty, S. (2018). *An improved text sentiment classification model using TF-IDF and next word negation*. arXiv preprint arXiv:1806.06407.
- Dave, K., Lawrence, S., & Pennock, D. M. (2003). *Mining the peanut gallery: Opinion extraction and semantic classification of product reviews*. In Proceedings of the 12th international conference on World Wide Web, 519-528.
- Demir, S., Alkan, O., Cekinel, F., & Karagoz, P. (2019). *Extracting Potentially High Profit Product Feature Groups by Using High Utility Pattern Mining and Aspect Based Sentiment Analysis*. Studies in Big Data High-Utility Pattern Mining, 233-260.
- Ding, X., Liu, B., & Yu, P. S. (2008). *A holistic lexicon-based approach to opinion mining*. Proceedings of the International Conference on Web Search and Web Data Mining - WSDM '08.
- Eirinaki, M., Pisal, S., & Singh, J. (2012). *Feature-based opinion mining and ranking.* Journal of Computer and System Sciences, 78(4), 1175-1184.

- Ejieh, C., Ezeife, C. I., & Chaturvedi, R. (2019). *Mining product opinions with most frequent clusters of aspect terms*. Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing.
- Fournier-Viger, P., Nkambou, R., & Tseng, V. S. M. (2011). *RuleGrowth: mining sequential rules common to several sequences by pattern-growth*. In Proceedings of the 2011 ACM symposium on applied computing, 956-961.
- Fournier-Viger, P. (2012). *Example: Mining High-utility sequential patterns from a Sequence database with Utility information using The USpan Algorithm (SPMF - Java)*. Retrieved April 12, 2021, from https://www.philippe-fournier-viger.com/spmf/USpan.php.
- Gan, W., Lin, J. C. W., Zhang, J., Chao, H. C., Fujita, H., & Philip, S. Y. (2020). *ProUM: Projection-based utility mining on sequence data*. Information Sciences, 513, 222-240.
- Golande, A., Kamble, R., & Waghere, S. (2016). *An Overview of Feature Based Opinion Mining*. Advances in Intelligent Systems and Computing Intelligent Systems Technologies and Applications 2016, 633-645.
- Ghorashi, S. H., Ibrahim, R., Noekhah, S., & Dastjerdi, N. S. (2012). *A frequent pattern mining algorithm for feature extraction of customer reviews*. International Journal of Computer Science Issues (IJCSI), 9(4), 29.
- Han, J., Pei, J., Mortazavi-Asl, B., Pinto, H., Chen, Q., Dayal, U., & Hsu, M. (2001). *Prefixspan: Mining sequential patterns efficiently by prefix-projected pattern growth*. In proceedings of the 17th international conference on data engineering, 215-224.
- He, R., & McAuley, J. (2016). *Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering.* In proceedings of the 25th international conference on world wide web, 507-517.
- Hridoy, S. A., Ekram, M. T., Islam, M. S., Ahmed, F., & Rahman, R. M. (2015). *Localized twitter opinion mining using sentiment analysis*. Decision Analytics, 2(1).
- Hu, M., & Liu, B. (2004). *Mining and summarizing customer reviews*. Proceedings of the 2004 ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '04.
- Hu, M., & Liu, B. (2004). *Mining opinion features in customer reviews*. In AAAI (Vol. 4, No. 4, 755-760).

- Hu, M., & Liu, B. (2006). *Opinion Feature Extraction Using Class Sequential Rules*. In AAAI Spring Symposium: Computational Approaches to Analyzing Weblogs, 61-66.
- Jain, A. P., & Katkar, V. D. (2015). *Sentiments analysis of Twitter data using Data Mining*. In 2015 International Conference on Information Processing (ICIP), 807-810.
- Jedrzejewski, K., &; Morzy, M. (2011). *Opinion Mining and Social Networks: A Promising Match*. 2011 International Conference on Advances in Social Networks Analysis and Mining.
- Jin, W., Ho, H. H., & Srihari, R. K. (2009). *OpinionMiner: a novel machine learning system for web opinion mining and extraction*. In Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining,1195-1204.
- Kadri, O., & Ezeife, C. I. (2011). *Mining uncertain web log sequences with access history probabilities*. In Proceedings of the 2011 ACM Symposium on Applied Computing, 1059-1060.
- Khalid, H. M., & Helander, M. G. (2006). *Customer emotional needs in product design*. Concurrent Engineering, 14(3), 197-206.
- Kim, S. M., & Hovy, E. (2004). *Determining the sentiment of opinions*. In COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics, 1367-1373.
- Kim, W. Y., Ryu, J. S., & Kim, U. M. (2009). *Opinion Mining of Product Reviews using Association Rules*. In Proceedings of the Korea Information Processing Society Conference, 747-748.
- Lan G., Hong T., Tseng V. S. & Wang S. (2012). *An improved approach for sequential utility pattern mining*. 2012 IEEE International Conference on Granular Computing, Hangzhou, 226-230.
- Li, Y. M., Chen, H. M., Liou, J. H., & Lin, L. F. (2014). *Creating social intelligence for product portfolio design*. Decision Support Systems, 66, 123-134.
- Lin, L., Li, J., Zhang, R., Yu, W., & Sun, C. (2014). *Opinion mining and sentiment analysis in social networks: a retweeting structure-aware approach*. 2014 IEEE/ACM 7th International Conference on Utility and Cloud Computing, 890-895.
- Lin, J. C. W., Gan, W., Fournier-Viger, P., Hong, T. P., & Tseng, V. S. (2016). *Efficient algorithms for mining high utility itemsets in uncertain databases*. Knowledge-Based Systems, 96, 171-187.

- Liu, B., Hsu, W., & Ma, Y. (1998). *Integrating classification and association rule mining*. In KDD (Vol. 98, 80-86).
- Liu, B., Hu, M., & Cheng, J. (2005). *Opinion observer*. Proceedings of the 14th International Conference on World Wide Web - WWW '05.
- Liu, H., & Wang, B. (2007). *An association rule mining algorithm based on a Boolean matrix.* Data Science Journal, 6, S559-S565.
- Lin, J. C. W., Fournier-Viger, P., & Gan, W. (2016). *FHN: An efficient algorithm for mining high-utility itemsets with negative unit profits*. Knowledge-Based Systems, 111, 283-298.
- Sukanya, M., & Biruntha, S. (2012). *Techniques on text mining*. IEEE International Conference on Advanced Communication Control and Computing Technologies (ICACCCT), Ramanathapuram, 269-271.
- Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J. R., Bethard, S., & McClosky, D. (2014). *The Stanford CoreNLP natural language processing toolkit*. In Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations, 55-60.
- Mayo, M. (2017). *A General Approach to Preprocessing Text Data*. Retrieved February 01, 2021, from https://www.kdnuggets.com/2017/12/general-approach-preprocessing-text-data.html.
- McAuley, J. (2016). *Amazon product data*. Retrieved February 01, 2021, from https://jmcauley.ucsd.edu/data/amazon/.
- Mewari, R., Singh, A., & Srivastava, A. (2015). *Opinion Mining Techniques on Social Media Data*. International Journal of Computer Applications, 118(6), 39-44.
- Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). *Distributed representations of words and phrases and their compositionality*. In Advances in neural information processing systems, 3111-3119.
- Moghaddam, S., & Martin, E. (2012). *Aspect-Based Opinion Mining from Product Reviews*. In Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval, 1184–1184.
- Mumu, T. S., & Ezeife, C. I. (2014). *Discovering Community Preference Influence Network by Social Network Opinion Posts Mining*. Data Warehousing and Knowledge Discovery Lecture Notes in Computer Science, 136-145.

- Ni, J., Li, J., & McAuley, J. (2019). *Amazon review Data (2018)*. Retrieved April 12, 2021, from https://nijianmo.github.io/amazon/index.html.
- Nurrahmi, H., Maharani, W., & Saadah, S. (2016). *Feature extraction and opinion classification using class sequential rule on customer product review*. In 2016 4th International Conference on Information and Communication Technology (ICoICT),1-5.
- Owoputi, O., O'Connor, B., Dyer, C., Gimpel, K., Schneider, N., & Smith, N. A. (2013, June). *Improved part-of-speech tagging for online conversational text with word clusters*. In Proceedings of the 2013 conference of the North American chapter of the association for computational linguistics: human language technologies,380-390.
- Pang, B., Lee, L., & Vaithyanathan, S. (2002). *Thumbs up? Sentiment classification using machine learning techniques*. arXiv preprint cs/0205070.
- Parashar, P., and Sharma, S. (2016). *A Literature Review on Architecture Classification Technique and Challenges of Sentiment Analysis*. International Journal of Engineering Research 5, no. 5.
- Pei, J., Han, J., Lu, H., Nishio, S., Tang, S., & Yang, D. (2001). *H-mine: Hyper-structure mining of frequent patterns in large databases*. In proceedings 2001 IEEE international conference on data mining, 441-448.
- Pennington, J., Socher, R., & Manning, C. D. (2014). *Glove: Global vectors for word representation*. In Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP),1532-1543.
- Powers, D. M. (2020). *Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation*. arXiv preprint arXiv:2010.16061
- Rabiner, L., & Juang, B. (1986). *An introduction to hidden Markov models*. IEEE assp magazine, 3(1), 4-16
- Rajeev, P. V., & Rekha, V. S. (2015). *Recommending products to customers using opinion mining of online product reviews and features*. In 2015 International Conference on Circuits, Power and Computing Technologies [ICCPCT-2015], 1-5.
- Rana, T. A., & Cheah, Y. N. (2019). *Sequential patterns rule-based approach for opinion target extraction from customer reviews*. Journal of Information Science, 45(5), 643-655.

- Rashid, A., Asif, S., Butt, N. A., & Ashraf, I. (2013). *Feature level opinion mining of educational student feedback data using sequential pattern mining and association rule mining*. International Journal of Computer Applications, 81(10).
- Ravi, K., & Ravi, V. (2015). *A survey on opinion mining and sentiment analysis: Tasks, approaches and applications*. Knowledge-Based Systems, 89, 14-46.
- Sentistrength. Retrieved April 12, 2021, from http://sentistrength.wlv.ac.uk/
- Srikant, R., & Agrawal, R. (1996). *Mining sequential patterns: Generalizations and performance improvements*. In International Conference on Extending Database Technology,1-17.
- Toutanova, K., Klein, D., Manning, C. D., & Singer, Y. (2003). *Feature-rich part-of-speech tagging with a cyclic dependency network*. In Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics, 252-259.
- Turney, P. D. (2002). *Thumbs up or thumbs down? Semantic Orientation Applied to Unsupervised Classification of Reviews* Proceedings of the 40th Annual Meeting on Association for Computational Linguistics - ACL '02.
- Virk, K. (2021). *Improving E-Commerce Recommendations using High Utility Sequential Patterns of Historical Purchase and Click Stream Data* (Doctoral dissertation, University of Windsor (Canada)).
- Xu, T., Dong, X., Xu, J., & Dong, X. (2017). *Mining high utility sequential patterns with negative item values*. International Journal of Pattern Recognition and Artificial Intelligence, 31(10), 1750035.
- Yao, H., Hamilton, H. J., & Butz, C. J. (2004*). A Foundational Approach to Mining Itemset Utilities from Databases*. Proceedings of the 2004 SIAM International Conference on Data Mining.
- Yin, J., Zheng, Z., & Cao, L. (2012). *USpan: an efficient algorithm for mining high utility sequential patterns*. In Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining, 660-668.
- Yin, J., Zheng, Z., Cao, L., Song, Y., & Wei, W. (2013). *Efficiently mining top-k high utility sequential patterns*. In 2013 IEEE 13th International conference on data mining,1259-1264.

VITA AUCTORIS

NAME	Priyanka Motwani
PLACE OF BIRTH	Ahmedabad, Gujarat, India
YEAR OF BIRTH	1995
EDUCATION	Nelson's High School, Ahmedabad, Gujarat India (2011)
	Best Higher Secondary School, Ahmedabad, Gujarat, India (2011 - 2013)
	Dharmsinh Desai University, Nadiad, Gujarat, India (2013 - 2017)
	University of Windsor, Ontario, Canada (September 2019 – July 2020)