Politecnico di Torino Master Thesis Department ofEngineering Management & Production (DIGEP)

Graduation Session APRIL-2022

Application of anomaly detection tools for digital VoC analysis

Relatori:	Candidati:
Prof. BARRAVECCHIA FEDERICO Prof. MASTROGIACOMO LUCA	Aamir Hasany S289678

DIPARTIMENTO DI INGEGNERIA GESTIONALE

E DELLA PRODUZIONE (DIGEP)

Declaration and acknowledgement

The work would not have been possible without the constant support and guidance from my supervisors, Prof. BARRAVECCHIA FEDERICO (DIGEP) and Prof. MASTROGIACOMO LUCA (DIGEP). Their ideas, vision and assistance paved the way towards the completion of this thesis. For this, I would like to express my deep appreciation and forever indebtedness towards them.
Secondly, I have acknowledged all sources used in the preparation of this thesis. I would like to express my sincere appreciation to all the authors, researchers and individuals who have contributed to the completion of this thesis. Finally, I pay heartiest regards to my parents, Mr. and Mrs. Lais Hasany for their love, care and around the clock support. Thank you for letting me study abroad and chase my dreams. To my siblings, relatives, and friends, Thank you all.

Table of Contents

LIST OF FIGURE

ABSTRACT

The fourth industrial revolution, or Industry 4.0, has brought significant advancements in the field of technology, including the implementation of Artificial Intelligence, Automation, machine learning and Internet of Things. These technologies have had a significant impact on the manufacturing industry, leading to the emergence of Quality 4.0. Quality 4.0 focuses on improving the quality and management of products and services through the use of data processing techniques, providing continuous process improvement. One of the key drivers of Quality 4.0 is the incorporation and analysis of big data.

Big data is comprised of feedback and reviews gathered from customers and clients, which is then processed using machine learning methods to gain insight into customer demand. The use of data mining techniques is critical in analyzing and explaining customer sentiment. To achieve this, Topic Modeling algorithms are used to identify key topics discussed in the review data. By using different data analytics tools, key information can be extracted, and the required reviews can be filtered out.

The process of detecting anomalies in customer reviews has been a challenging task for a long time, and it has always been an active field of research in quality assurance. The reason behind this is that traditional methods have often failed to capture the complexities of human language. Therefore, it is essential to have more advanced techniques that can accurately analyze and interpret customer feedback to identify any underlying quality defects.

In recent years, deep learning methodologies have emerged as one of the most promising approaches to address this issue. Deep learning algorithms can learn from large datasets, extract relevant features, and make predictions based on the learned patterns. In the context of customer feedback, deep learning algorithms can analyze large volumes of unstructured data, such as customer reviews, to detect and classify quality defects in real-time.

One of the key techniques used in deep learning for anomaly detection in customer feedback is Topic Modeling algorithms. Topic Modeling algorithms are statistical models that identify the underlying topics in a set of documents, such as customer reviews. By identifying the most relevant topics, the algorithm can

analyze the frequency and distribution of words in each topic to determine whether a particular review is anomalous or not.

Behavior patterns of words and topic distribution probabilities are two critical factors in detecting anomalies in customer feedback. For instance, if a customer review mentions a product defect that has not been previously reported, it may indicate a quality issue that needs to be addressed. Similarly, if multiple reviews mention similar problems, it may be an indicator of a larger quality issue that requires immediate attention.

Another benefit of using deep learning methodologies for anomaly detection in customer feedback is that it allows for real-time monitoring and rectification of quality defects. By analyzing customer feedback in real-time, manufacturers can quickly identify and rectify any quality defects, improving customer satisfaction and reducing the risk of negative reviews.

Anomaly detection is a crucial aspect of quality assurance, and the use of deep learning methodologies has revolutionized the field. The adoption of Topic Modeling algorithms in anomaly detection provides a unique advantage, enabling the identification of quality defects in real-time. By detecting anomalies in customer feedback, manufacturers can identify and address quality issues before they become widespread.

The benefits of Quality 4.0 are numerous. It provides manufacturers with the tools and techniques necessary to improve product and service quality, reducing defects and improving customer satisfaction. The use of data analytics tools, including machine learning and data mining techniques, enables manufacturers to gain a deep understanding of customer feedback and sentiment, identifying areas for improvement.

INTRODUCTION

Background

In today's digital age, feedback analysis has become increasingly important for businesses and organizations of all sizes. With the growing use of social media, online reviews, and other digital platforms, companies have access to an enormous amount of data about their customers and their preferences. This feedback data is crucial for identifying areas of improvement, understanding customer sentiment and opinions, and ultimately improving customer satisfaction.

One of the primary sources of feedback data is customer surveys. These can be conducted in a variety of ways, including online surveys, phone surveys, and in-person surveys. Companies can use survey data to gain insight into customer preferences, satisfaction levels, and pain points. This information can be used to improve product or service offerings, enhance customer experiences, and ultimately drive business growth.

Another source of feedback data is public reviews. These can include online reviews on websites such as Amazon or Yelp, as well as reviews left on social media platforms like Facebook and Twitter. Public reviews can provide valuable insights into the strengths and weaknesses of a business, as well as feedback on specific products or services. Companies can use this information to improve their offerings and address any issues that customers may be experiencing.

User interviews are another valuable source of feedback data. Companies can conduct one-on-one interviews with customers to gain deeper insights into their preferences, needs, and pain points. User interviews can provide more detailed and nuanced feedback than surveys or public reviews, as they allow companies to ask follow-up questions and clarify any ambiguities.

Social media posts and tweets are also a valuable source of feedback data. Companies can use social media listening tools to monitor conversations about their brand, products, or services. By analyzing social media posts and tweets, companies can gain insights into customer sentiment, preferences, and pain points in real-time.

In order to make sense of this vast amount of feedback data, companies are turning to deep learning and artificial intelligence (AI) tools. These tools can help to identify patterns and trends in feedback data, and provide insights that would be difficult to obtain through traditional analysis methods. By using deep learning and AI, companies can gain a more comprehensive understanding of their customers, and use this information to drive business growth and improve customer satisfaction.

Problem Statement

Topic modeling is a machine learning technique that allows us to discover the hidden structure in a large dataset by identifying the topics that are being discussed in the data. It can be used for a wide range of applications, including text analysis, data mining, and anomaly detection. In anomaly detection, topic modeling can be particularly useful because it can identify unusual patterns in the data that may indicate the presence of an anomaly.

One of the main advantages of topic modeling for anomaly detection is its ability to handle complex and dynamic data distributions. Unlike traditional rule-based methods, topic modeling can capture the underlying structure of the data without relying on pre-defined rules. This makes it particularly useful for identifying anomalies in large datasets that may have complex and dynamic distributions.

Another advantage of topic modeling for anomaly detection is its ability to identify subtle anomalies that may be difficult to detect using traditional methods. By analyzing the topics and the distribution of words in the data, topic modeling can identify anomalies that may not be immediately apparent to the human eye.

Topic modeling for anomaly detection has been applied in various fields, including computer security, financial fraud detection, and quality control. For example, in computer security, topic modeling can be used to identify unusual patterns in network traffic that may indicate the presence of a malware infection. In financial fraud detection, topic modeling can be used to identify unusual patterns in financial transactions that may indicate the presence of fraudulent activity.

In quality control, topic modeling can be used to identify unusual patterns in customer feedback that may indicate the presence of a quality defect. By analyzing the topics and the distribution of words in customer feedback, topic modeling can identify anomalies that may indicate a problem with a product or service. This information can be used to make informed decisions and improvements to a company's products, services, and overall customer experience.

Scope

The rise of Web 2.0 has transformed the way users interact with websites and the internet as a whole. The focus has shifted towards creating a more interactive and engaging user experience, with greater emphasis on sharing information and communicating with others. This has led to the emergence of Digital Voice of Customer (VoC), which has become increasingly popular due to its accessibility, trustworthiness, and cost-effectiveness. By leveraging the power of digitalization and data analysis tools, companies can now gain valuable insights into customer needs and preferences.

VoC interpretation involves the analysis and understanding of customer feedback, opinions, and sentiments, with the ultimate goal of improving a company's products, services, and overall customer experience. This process can be challenging, as it requires the identification of patterns and trends in large volumes of customer data. However, recent advances in data mining and deep learning tools have made this process much more efficient and effective.

One of the most widely used methods for VoC interpretation is the application of topic modeling algorithms to customer review data. This technique involves identifying key topics and themes within the data, and analyzing the relationships between them. By identifying the key attributes that customers care about, companies can gain valuable insights into how to improve their products and services.

Supervised and unsupervised learning methods can also be used to interpret customer review data. These methods involve the use of machine learning algorithms to analyze the data and identify patterns and trends. Supervised learning involves the use of labeled data to train the algorithm, while unsupervised learning uses unlabeled data to identify patterns and trends.

Overall, the use of digital Voice of Customer interpretation has greatly reduced the mystery surrounding customer behavior and preferences. By leveraging the power of data analysis tools and machine learning algorithms, companies can gain valuable insights into how to improve their products and services, and ultimately provide a better customer experience.

Significance of the Study

In this thesis, we propose to use topic modeling for anomaly detection and explore its capability to capture the underlying structure of the data. We aim to demonstrate how topic modeling can be used to identify anomalies in real-world datasets, comparing its performance with traditional methods.

The thesis is structured to provide a comprehensive overview of topic modeling and its applications, with a focus on anomaly detection. We start by introducing the concepts of anomaly detection and topic modeling, providing a literature review of the various methods used for topic modeling and its applications in anomaly detection. We then describe the dataset used in the study and the experimental design, including the various methods used for anomaly detection.

We evaluate the performance of topic modeling-based methods for anomaly detection on different datasets, comparing them with traditional methods. Finally, we draw conclusions about the effectiveness of topic modeling for anomaly detection, highlighting its potential advantages and limitations. We discuss the implications of the study and provide recommendations for future research in this area. We also provide suggestions for the practical implementation of topic modeling-based methods for anomaly detection, highlighting the potential applications in various domains.

In conclusion, this thesis proposes to use topic modeling for anomaly detection and explores its capability to capture the underlying structure of the data. The study provides a comprehensive evaluation of the performance of topic modeling-based methods for anomaly detection, comparing them with traditional methods. The results of the study highlight the potential advantages of using topic modeling for anomaly detection, and provide suggestions for future research in this area. Overall, the study contributes to the growing body of research on anomaly detection and provides insights into the potential applications of topic modeling in this domain.

Literature Review

Quality 4.0

Quality 4.0 is the name attributed to the industrial revolution 4.0 the term defines their digital implementation of the quality management system procedures including advancement in data, analytics, machine learning, big data, Internet of Things, blockchain and artificial intelligence this allows greater efficiency and cost reduction. [1]

The evolution of the quality 4.0 began around the end of 19th century, due to the large-scale production of goods, the quality at that time monitored by visual inspection. This evolved with the passage of time from the generation of electric and computer power to the use of AI to self-regulate and monitor the process of quality management. The concept of Quality 4.0 first developed in Germany in 2011, aims to increase and monitor quality enhancement methods more efficiently by fusing conventional procedures and Industry.

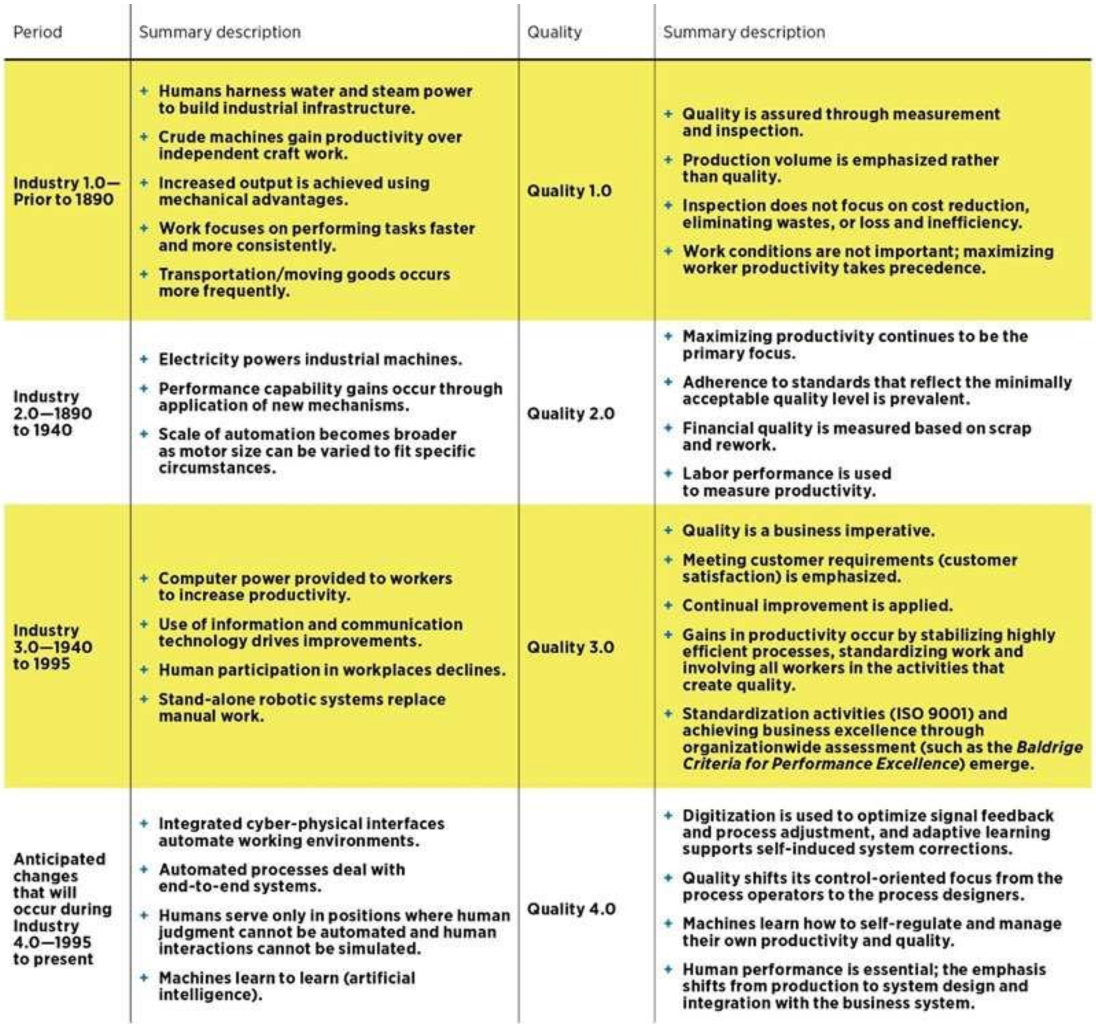
4.0 technologies.

In the recent years, the market has grown extremely competitive. Businesses are fighting to manufacture premium product lines and offer their clients top-notch services to survive.

Traditional quality management techniques have a number of problems and may not be able to maintain firm’s competitiveness in today's digitally driven environment when consumer tastes are constantly shifting. As a result, businesses are developing the greatest creative practices, like Quality 4.0, for the future.

Industry 4.0 aims to transform how businesses operate using ground-breaking technologies including cloud computing, big data, artificial intelligence, machine learning, virtual reality, augmented reality, the internet of things, and 3D printing. These cutting-edge technologies change how people interact with and use various technologies throughout the value chain.

Since, every employee in a firm must accept the new habits, adjusting to the technology driven transformation necessitates a significant culture revolution which incorporates the usefulness of quality4.0. [2]



Period	Summary description	Quality	Summary description
Industry 1.0—Prior to 1890	+ Humans harness water and steam power to build industrial infrastructure. + Crude machines gain productivity over independent craft work. + Increased output is achieved using mechanical advantages. + Work focuses on performing tasks faster and more consistently. + Transportation/moving goods occurs more frequently.	Quality 1.0	+ Quality is assured through measurement and inspection. + Production volume is emphasized rather than quality. + Inspection does not focus on cost reduction, eliminating wastes, or loss and inefficiency. + Work conditions are not important; maximizing worker productivity takes precedence.
Industry 2.0—1890 to 1940	+ Electricity powers industrial machines. + Performance capability gains occur through application of new mechanisms. + Scale of automation becomes broader as motor size can be varied to fit specific circumstances.	Quality 2.0	+ Maximizing productivity continues to be the primary focus. + Adherence to standards that reflect the minimally acceptable quality level is prevalent. + Financial quality is measured based on scrap and rework. + Labor performance is used to measure productivity.
Industry 3.0—1940 to 1995	+ Computer power provided to workers to increase productivity. + Use of information and communication technology drives improvements. + Human participation in workplaces declines. + Stand-alone robotic systems replace manual work.	Quality 3.0	+ Quality is a business imperative. + Meeting customer requirements (customer satisfaction) is emphasized. + Continual improvement is applied. + Gains in productivity occur by stabilizing highly efficient processes, standardizing work and involving all workers in the activities that create quality. + Standardization activities (ISO 9001) and achieving business excellence through organizationwide assessment (such as the *Baldrige Criteria for Performance Excellence*) emerge.
Anticipated changes that will occur during Industry 4.0—1995 to present	+ Integrated cyber-physical interfaces automate working environments. + Automated processes deal with end-to-end systems. + Humans serve only in positions where human judgment cannot be automated and human interactions cannot be simulated. + Machines learn to learn (artificial intelligence).	Quality 4.0	+ Digitization is used to optimize signal feedback and process adjustment, and adaptive learning supports self-induced system corrections. + Quality shifts its control-oriented focus from the process operators to the process designers. + Machines learn how to self-regulate and manage their own productivity and quality. + Human performance is essential; the emphasis shifts from production to system design and integration with the business system.

Figure i Evolution of Quality 4.0 and Industrial Advancement [3]

However, previous research has shown that only 16% of the companies have a clear connection between their corporate strategies and quality. The companies that have adopted have been successful in overcoming obstacles like cross functionality insufficiency, lacking in data-driven decision making and lower data transparency.

Voice Of Customer

The term "voice of the customer" describes a consumer's opinions, expectations, preferences, remarks, or observations regarding a service or product. Customer feedback is essentially what a customer says about a good or service. Therefore, an analytics program is required that includes customer feedback as an strategic system that collects feedback, examines data, and takes necessary action.
Analyses of the "voice of the customer" seek to discover how consumers feel about particular products, services, and brands. Voice of Customer can give you an unmatched grasp of what your customers desire from your company,

products, etc. through detailed awareness of the customer's needs. A company can make highly informed judgments based on highly informed business data by utilizing the voice of the consumer. The sources for obtaining the customer feedback are various as can be seen from the figure below.

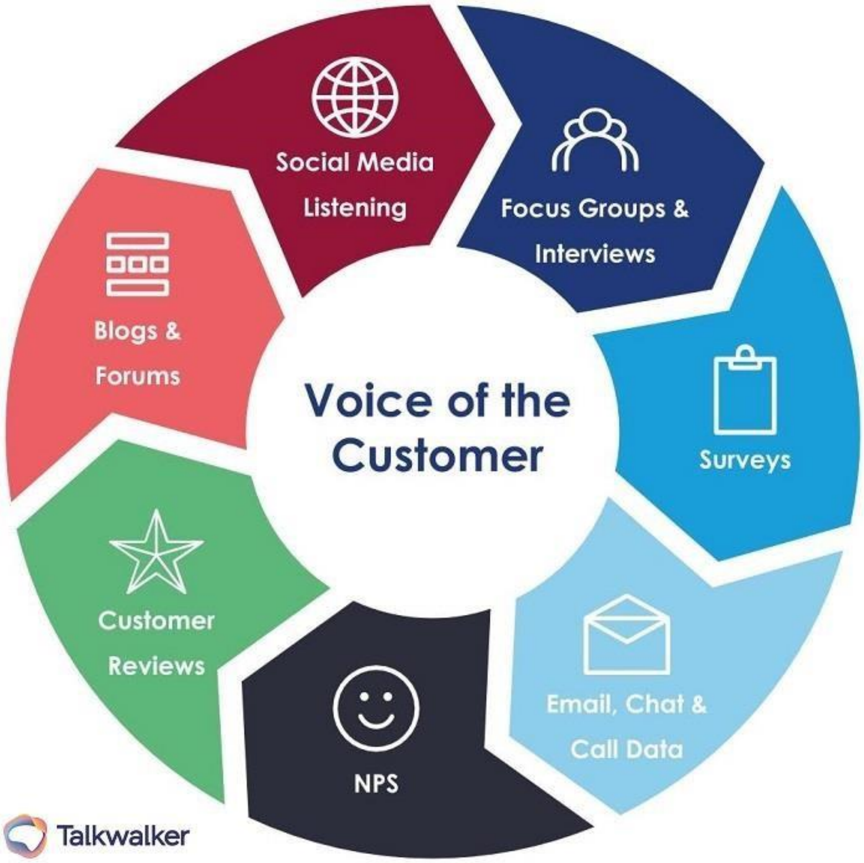


Figure ii Some examples of data sources you can integrate to complete your voice of customer analysis. [4]

Customer reviews or posts on social media platforms not only give businesses helpful feedback, but they can also help identify the biggest issues that certain types of customers could be dealing with. The business can then take the necessary remedial action to address the issues raised by customers. For example, imagine that many consumers of a mobile operator are complaining on social media about problems with the network's coverage (e.g. Twitter). If the mobile operator has the ability to automatically track such user remarks in real-time, it can cooperate with the troubled customer to resolve their issues.

Online customer reviews can be helpful in providing a low-cost information source for determining client needs and their expectations [3]. However, despite some platforms (such as Facebook and Google) are making efforts to restrict the download of sizable volumes of digital VoC, there are numerous software tools available for online scraping. Additionally, web scraping libraries are frequently included in text mining software. These tools frequently permit the collecting of pertinent metadata, such as author, title, date, country, and rating, in addition to the reviews' textual content [4]. There are numerous ways to get insights from digital VoC at the moment. To determine the most popular topics, most people employ topic modeling algorithms [4]. Online product reviews enable businesses to conduct comprehensive consumer environment studies quickly and affordably. Researchers and practitioners have gradually brought attention to this market data in recent years [5] [6]. Although online voices of the customer (VOC) are free text, they have shown to represent the basic market characteristics and to apply advances in conventional marketing operations [7]However, there is a roadblock that keeps online product reviews from reaching their full potential. Due to the size and high quality of user-generated online material, it is challenging to quantify the data and produce meaningful insights. [8]

Businesses replace quality ratings for product reviews. For instance, [9] used product ratings to explore the connection between consumer feedback and movie industry sales, while [10] used them to study the book industry. Based on several methodologies, such as setup, brand- switching data [11], and brand-associative networks, market structure shows a partnership between brands [12]. Researchers have started using text mining techniques based on natural language processing (NLP) to get standardized and quantitative market knowledge from online product evaluations [13]. For instance, [14] created a text mining method for online product reviews. Additionally, [15] developed a hybrid platform for semantic network analysis and text mining for market structure surveillance.

Ways to analyze voice of customer data

Sentiment Analysis

With the help of AI, sentiment analysis can determine whether a reference is being made in a positive, negative, or neutral way. Then, you may easily separate your analysis into two parts.

i. What people enjoy, you should emphasize
ii. What people dislike, you should fix

Scaling up sentiment is harder than it seems. To assure accuracy, you need a system that recognizes sarcasm and context. Additionally, the analysis may be tailored to your use case so that it can learn as you evaluate the data over time, enhancing your workflow and time- efficiency. To analyze the feedback of the brand, sentiment analysis can be used to identify the positive and negative mentions, as can be viewed from the figure below.

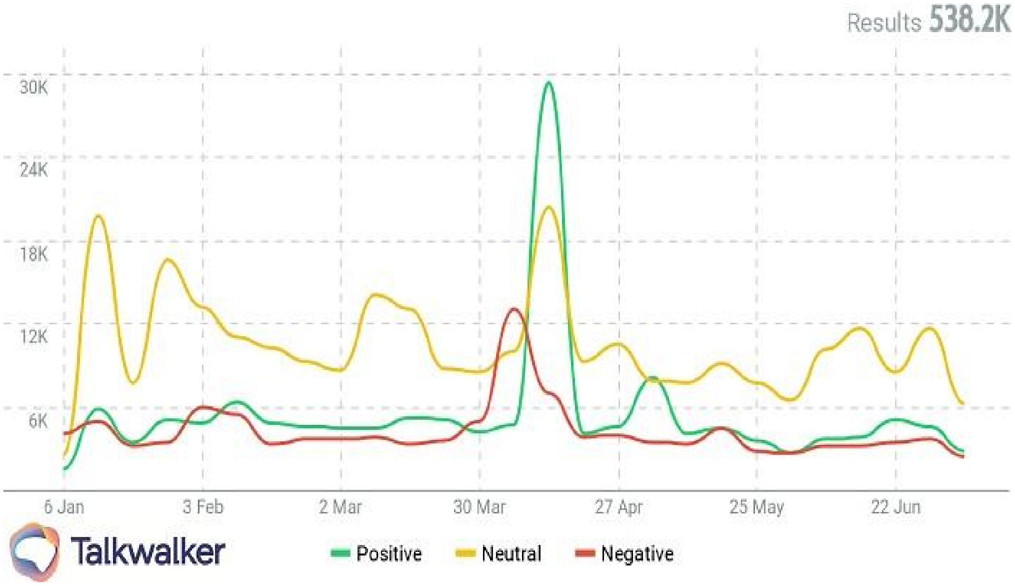


Figure iii sentiment analysis assisted a brand in identifying a spike in unfavorable mentions.

Natural Language Processing

The goal of natural language processing (NLP) is to use tools, techniques, and algorithms to process and comprehend data that uses natural language data which typically is unstructured such as text, speech, and so on. Here are a few examples of effective Natural Language Processing (NLP) applications:

- Google, Yahoo, and other search engines Because Google recognizes your preferences, and it displays results that are relevant to you.
- Social media websites provide news feeds similar to Facebook's. The news feed algorithm uses natural language processing to recognize your interests and displays related advertisements and content to you more frequently than other posts.
- speech recognition software like Apple Siri.
- spam filters like those on Google. Spam filters today consider the content of emails when determining whether or not they are spam. This goes beyond the standard spam filtering.

Many techniques for natural language processing tasks, like text mining, use supervised learning, where you provide an input along with a label and allow the computer identify the pattern. Unsupervised learning is another option, though, if you simply have data and want to identify groups, say, based on similarities. [16]

A technique for revealing a collection of documents' latent structure is topic modeling. It entails recognizing the topics or themes that are present in the text

and putting together the
topics that cover related subjects.

Topic Modeling

A topic model is a probabilistic model that contains data on the subjects in our text. But in this case, two crucial queries come up, which are stated as:

I. How to define a topic?

The word "topic" refers to the overall concepts or themes that run through our writing. As an illustration, a corpus of newspaper articles might cover subjects like money, the environment, politics, sports, news from different states, and so on.

II. Secondly, What role do topic models have in text processing?

We are aware that we can use information retrieval and searching strategies to find similarities in text. However, topic models can now help in searching and organizing our text files.

We can define subjects as the probability distribution of words in this way. By utilizing topic models, the documents can be categorized as probabilistic distribution of different topics. [17]

The emphasis of this phenomena is on the concepts and themes, as was previously discussed. Its primary objectives are as follows:

- It is possible to summarize text using topic models.
- Since the subjects are determined by evaluating the texts, topic modeling algorithms do not need any prior annotations or tagging of the documents [18].
- The documents can be organized using them. Topic modeling, for instance, can be used to organize news stories into sections that are connected and organized, such as grouping all cricket-related news pieces together.
- They can boost search engine results. How? In response to a search query, topic models can provide the content that contains a variety of distinct keywords but is focused on the same subject.
- For marketing, the idea of recommendations is highly helpful. Numerous online stores, news websites, and many others use it. Making suggestions on what to buy, what to read next, etc. is made easier with the use of topic models. They search materials with a common topic in lists to accomplish this.

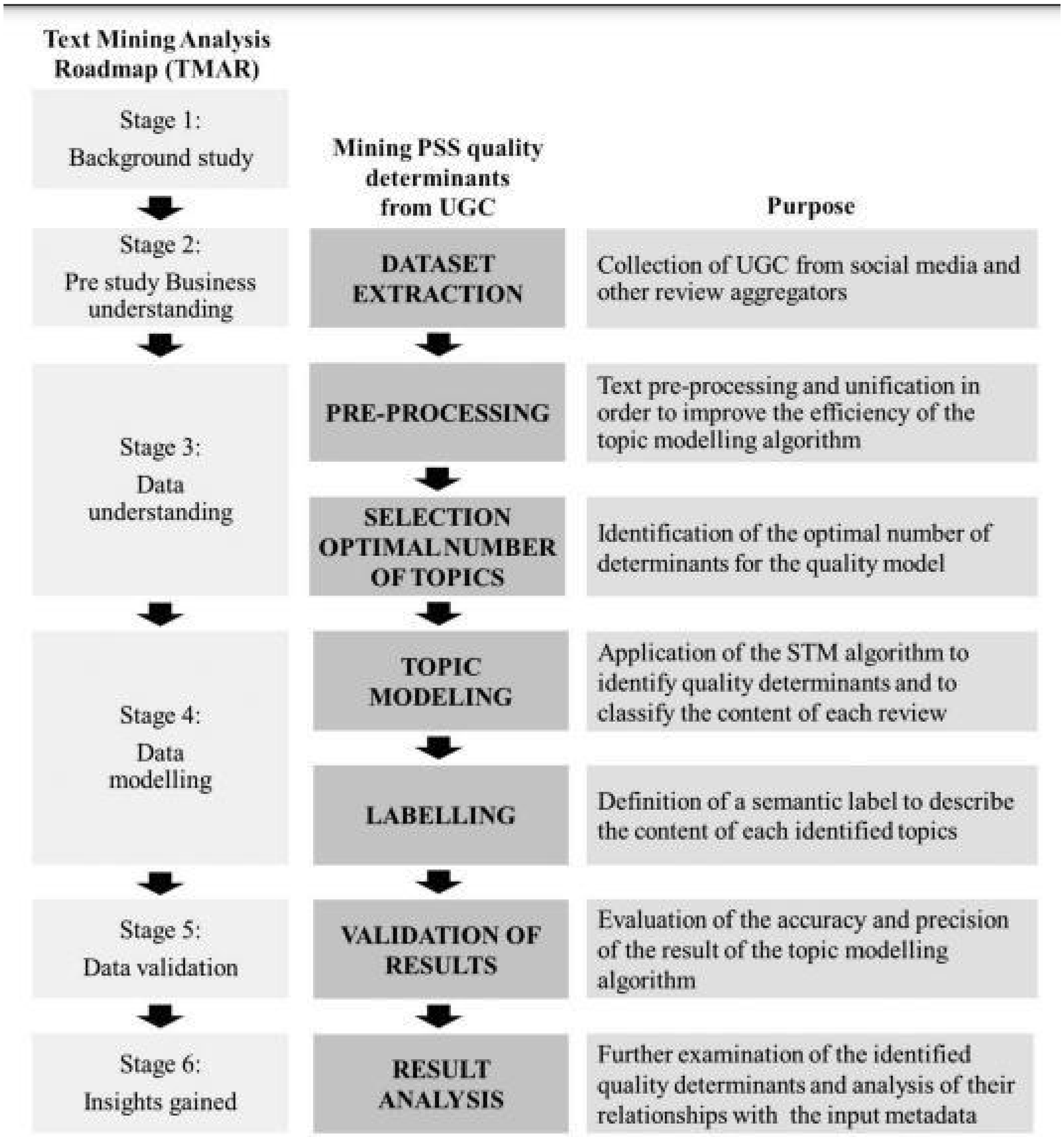

Text Mining Analysis Roadmap (TMAR)
Stage 1: Background study
Stage 2: Pre study Business understanding
Stage 3: Data understanding
Stage 4: Data modelling
Stage 5: Data validation
Stage 6: Insights gained
Mining PSS quality determinants from UGC
DATASET EXTRACTION
PRE-PROCESSING
SELECTION OPTIMAL NUMBER OF TOPICS
TOPIC MODELING
LABELLING
VALIDATION OF RESULTS
RESULT ANALYSIS
Purpose
Collection of UGC from social media and other review aggregators
Text pre-processing and unification in order to improve the efficiency of the topic modelling algorithm
Identification of the optimal number of determinants for the quality model
Application of the STM algorithm to identify quality determinants and to classify the content of each review
Definition of a semantic label to describe the content of each identified topics
Evaluation of the accuracy and precision of the result of the topic modelling algorithm
Further examination of the identified quality determinants and analysis of their relationships with the input metadata

Figure iv Activity flow of the proposed methodology and comparison with the Text Mining Analysis Roadmap [21]

The topic modeling phonomenon as can be viewed from the above figure, is a part of process flow of text mining analysis of the data. Natural language processing, information retrieval, and text categorization are just a few of the areas where topic modeling has been used. By locating the underlying subjects in text data, topic modeling has been used to the field of text classification to increase the accuracy of text classification. By putting comparable documents together, topic modeling has been used to enhance the performance of search engines in information retrieval. Topic modeling has been used in natural language processing to extract information from big text corpora, such as social media data.

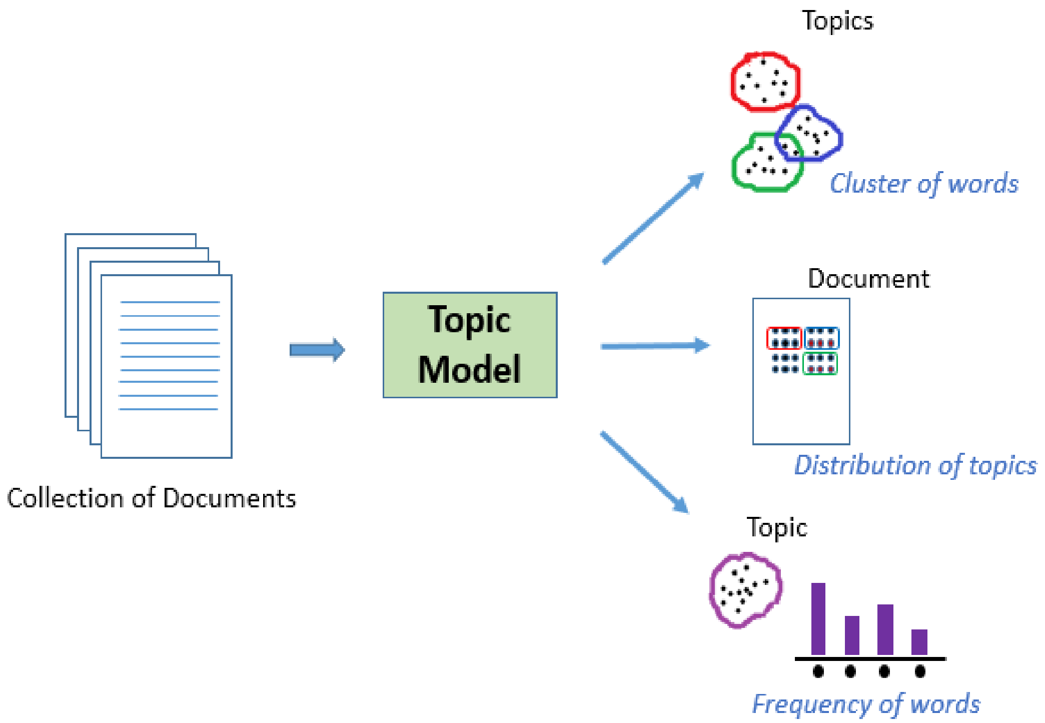


Figure v Flowchart of Topic Modelling Using LDA All topic models are based on the same basic assumption that 1. Each document consists of a mixture of topics. 2. Each topic consists of a collection of words [22]

A document's latent semantic structure can be discovered using topic modeling as can be seen from the figure above. The document with the help of topic modeling can be break down into its respective clusters, distribution of topics and frequency of words. The fundamental premise is that each document is made up of a variety of topics, and each topic is made up of a collection of words.

Scholars have been working on Topic Modeling for a long time. Numerous research have been made on topic modeling using Gensim and scikit-learn. The most popular algorithms among researchers include Latent Dirichlet Allocation, Non-Matrix Factorization, Latent Dirichlet Allocation, Latent Semantic Analysis, and pLSA, some of which are implemented in libraries. Following are some notable researches in terms of practical implementation of Topic modeling:

Manufacturing

The paper “analyzing scientific research topics in manufacturing field using a topic model by Hui Xiong, Yi Cheng, Wenhao Zhao, Jianhua Liu” deals with the study to identify growing trends and areas of manufacturing research. The research scope of manufacturing is becoming more interdisciplinary, which means that fields like management and meteorology have also contributed to manufacturing. It gathers a large number of abstract articles from different nations and niche areas.

Additionally, it is no longer just restricted to heavy machinery or mechanical transmissions but also includes advanced technologies. The study also demonstrates that some regions exclusively concentrate on a few fields, which means that some regions might experience issues that other regions do not. [19]

Black Markets online

The article “analyzing the DarkNetMarkets subreddit for evolutions of tools and trends using LDA topic modeling by Kyle Porter” examines DarkNetMarkets, a subreddit forum with a focused audience. To find the pattern, the objective is to extract a term and create a trend from it. Latent Dirichlet allocation was used to analyze the data, which was gathered between November 2016 and October 2017 and the results show that users are being more cautious as a result of recent police enforcement. [20]

Topics For Scientific Research

Similar to the manufacturing research, the article “Towards Predicting Trend of Scientific Research Topics using Topic Modeling by Tesfamariam M.Abuhay, Yemisrach G.Nigatie, Sergey V.Kovalchuka” examined conference abstracts from 2001 to 2017 from an event called ICCS. 5982 papers were gathered as a result of the authors' experiments with non-negative matrix factorization. In order to forecast the trend of the research issue, the outcome was additionally projected to a time series chart using ARIMA. [21]

Article - United States Presidential Elections 2016

The 2016 US Presidential Election data was collected and analyzed in the article tracking geographical locations using a geo-aware topic model for analyzing social media data by Marianela García Lozano, Jonah Schreiber, Joel Brynielsson. They created two unique datasets. One, 3200 of the most recent tweets were obtained on April 27, 2016, using Twitter4J to gather the data. Another, known as USE2016, collected data from 1 February to 1 May 2016 and pooled 144 million tweets using popular hashtags for the US 2016 primary election. Without defining the location of the tweets, the paper's goal was to determine whether any given tweet was identical to that location. It did this by employing streaming latent Dirichlet allocation. [22]

By extracting relevant information from unstructured text data, topic modeling aims to make such information more understandable to humans. Latent Dirichlet Allocation is one of the most extensively used topic modeling methods (LDA). LDA is a probabilistic generative model that assumes that each document consists of a fixed number of topics and that each topic consists of a variety of words.

Discovering the hidden topic structure for a collection of connected documents is the primary objective of probabilistic topic modeling. A topic structure typically includes the following three elements:

1. Topics
2. Statistics about the issues that appear in the papers.
3. The topic's words are found throughout the paper.

The Hierarchical Dirichlet Process (HDP) and the Correlated Topic Model (CTM) are two additional models that can be suggested to enhance the effectiveness of topic modeling.

Latent Semantic Analysis (LSA), Structural Topic Model (STM), Probabilistic Latent Semantic Analysis (PLSA), and Latent Dirichlet Allocation (LDA) are just a few of the topic modeling methods that have been developed during the past three decades [23]. Most suitable algorithms for assessing digital Voice of Customer from the large family of topic modeling approaches are probabilistic modeling topic algorithms [20]. Particularly, it was found that STM outperformed LDA when information which is covariate (i.e., metadata connected with each textual article) was present [24]. For the examination of digital VoC, this factor is viewed as crucial.

Textual comments about the customer experience frequently include extra details like the product or service's rating, the kind of product or service utilized, the user's country of origin, etc. Probabilistic topic modeling algorithms address the issues of

1. identifying a set of topics that describe a text corpus (i.e., a collection of text documents from various sources)
2. associating a set of keywords to each topic
3. defining a specific mixture of these topics for each document when faced with a large number of documents [25].

Recent data indicates that digital Voice of Customer analysis can be used to classify items and services based on user perceptions as well as identify qualities of products and services. Various attempts have been undertaken to classify automatically product and service features in accordance with the initial categorical attributes provided by Kano [26].

Distinction among other methods

There are various ways in which topic modeling differs from other anomaly detection techniques, including:

Approach

While other approaches may concentrate on numerical data or other types of input, topic modeling is a text-based approach that aims to uncover underlying themes or topics in a collection of documents.

Modeling

To identify subjects, topic modeling employs probabilistic models like Latent Dirichlet Allocation (LDA), although other approaches may make use of statistical, machine learning, or deep learning methodologies.

Representation

Topic modeling displays data in a higher-dimensional space, where each document is represented as a combination of themes, whereas other methods may do it in a lower- dimensional space or in a more condensed fashion.

Anomaly Detection

Topic modeling uses topic representations to spot anomalies, like documents that stray greatly from the norm, whereas other approaches may make use of statistical metrics, distance measurements, or other techniques.

Interpretability

While other methods may produce more abstract representations that are harder to grasp, topic modeling provides a human-readable representation of the data, making it simpler to understand and analyze the results.

Python for Machine Learning

Python is a preferred programming language because of its extensive capabilities, applicability, and simplicity. Due to its independent platform and widespread use in the programming community, the Python programming language is the most suitable for machine learning. [27]

A component of artificial intelligence (AI) called machine learning tries to make a machine learn from experience and carry out tasks automatically without necessarily having to be programmed to do so. Contrarily, Artificial Intelligence (AI) is a more general term for machine learning in which computers are made to be sensitive to the human level by perceiving visually, by speaking, by language translation, and thereafter making important decisions.

Due to the demand for automation, machine learning and AI are rapidly expanding in utilization. Innovative solutions to everyday issues can be developed thanks to artificial intelligence, including fraud detection, personal assistants, spam filters, search engines, and recommendation systems.

The requirement for intelligent answers to practical issues needs the further development of AI to automate laborious processes that would be difficult to program without AI. The Python programming language is thought to be the ideal technique for automating these processes since it is more straightforward and consistent than other programming languages.

Additionally, having a vibrant Python community makes it simple for developers to discuss projects and offer suggestions for improving their code.

Natural Language Toolkit

The NLTK toolkit was created for Python users to deal with Natural Language Processing. It offers us numerous test datasets and different text processing libraries. Using NLTK, a range of activities as can be seen in the below process diagram, can be carried out, including tokenizing and visualizing parse trees.

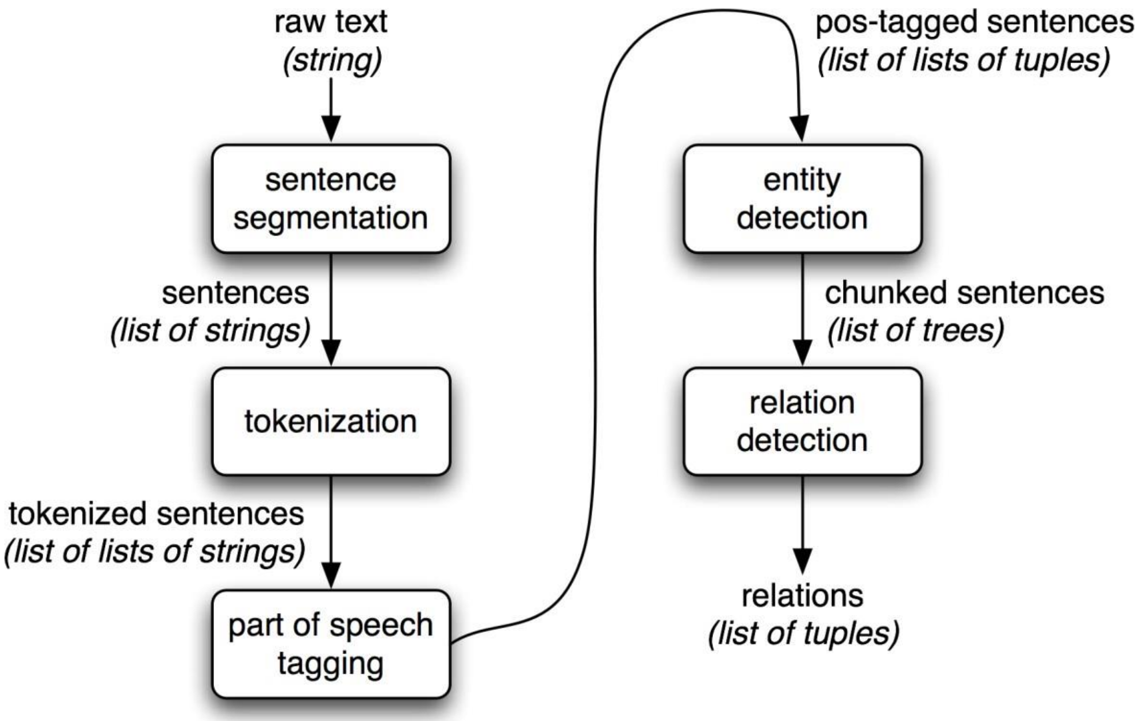


Figure vi Typical NLTK pipeline for information extraction [33]

Some common uses of data processing tasks of NLTK library includes

1. Tokenization
2. Lower case conversion
3. Stop Words removal
4. Stemming
5. Lemmatization
6. Parse tree or Syntax Tree generation
7. POS Tagging

To install NLTK in our system we can use the pip command method as follows

```
pip install nltk
```

To download the datasets from nltk we can use the following command:

```
import

nltk
```

Gensim topic Modeling- Python

Gensim, which stands for "Generate Similar", is a natural language open source library which is used for unsupervised topic modeling. It performs a variety of complex tasks such as building document or word vectors, corpora, performing topic identification, performing document comparison (retrieving semantically similar documents), and analyzing plain-text documents for semantic structure using the best academic models and contemporary statistical machine learning. Gensim is without a doubt the most well-liked subject modeling toolkit. It is more well-liked because it is free and written in Python. We'll talk about a few of the most well-liked subject modeling methods in this part. Because Gensim abstracts them so well for us, we shall concentrate on "what" in this case rather than "how." Over a thousand academic and commercial applications have used and cited Gensim. Numerous research publications and student thesis have also referenced it. It includes the following streamed parallelized implementations. [20]

fastText

FastText is a text classification and word embedding library that uses a neural network for word embedding. It was created by the Facebook AI Research (FAIR) lab. Using this strategy, we can easily create a supervised or unsupervised technique to produce vector representations of words.

Word2vec

Word2vec is a collection of two-layer neural network models. It is used in word embedding, this helps the models to be taught to rebuild the language contexts of words.

LSA (Latent Semantic Analysis)

Latent Semantic Analysis (LSA), a method for extracting meaning from words by examining connections between a group of texts and the terms that are present in it. It is an NLP or Natural Language Processing tool that enables us to examine connections between a group of documents and the terms they include.

LDA (Latent Dirichlet Allocation)

The method used most frequently and widely for topic modeling right now is latent Dirichlet allocation (LDA). It is the same one which was employed by the Facebook researchers in the 2013 research article. In 2003, David Blei, Andrew Ng, and Michael Jordan made the initial suggestion. In their article, simply named Latent Dirichlet Allocation, they proposed LDA.

It is an NLP approach that allows unobserved groups to explain sets of observations. These unobserved groupings provide an explanation for why some data elements are similar. It is a generative statistical model for this reason. A probabilistic topic modeling method is LDA. As was said above, when topic modeling, we make the assumption that each document in any collection of related documents—which could include academic papers, news stories, Facebook posts, Tweets, emails, and so on—contains a certain combination of themes.

LDA considered to be unsupervised, because conditional probabilities are used by LDA to uncover the underlying topic structure. It is assumed that the subjects are unevenly spread out throughout the group of connected papers.

LDA models may be easily created in Gensim. Just the corpus, dictionary mapping, and number of topics that we want to include in our model need to be specified.

It can be visualized from the below figure vii, that LDA model works by distributing topics over words. LDA has the computational issue of calculating the likelihood of each potential topic structure. .It is difficult because it must determine the likelihood of each observed word under every conceivable topic structure. When there are many subjects and words, LDA may run into computationally impossible problems.

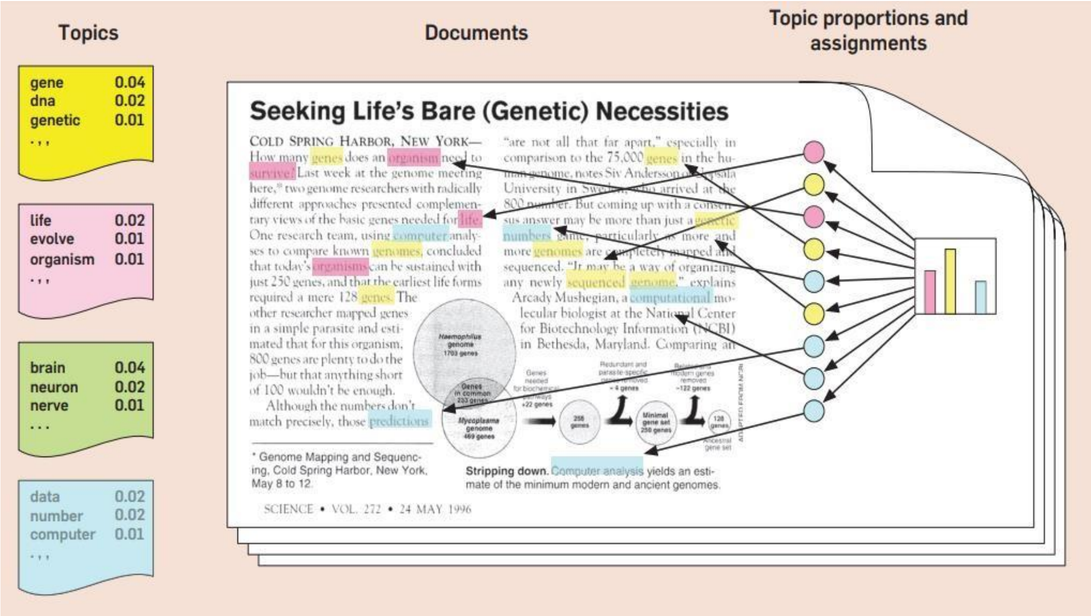

Topics
Documents
Topic proportions and assignments
gene 0.04
dna 0.02
genetic 0.01
life 0.02
evolve 0.01
organism 0.01
brain 0.04
neuron 0.02
nerve 0.01
data 0.02
number 0.02
computer 0.01
Seeking Life's Bare (Genetic) Necessities
Stripping down. Computer analysis yields an estimate of the minimum modern and ancient genomes.
SCIENCE • VOL. 272 • 24 MAY 1996

Figure vii The intuitions behind latent Dirichlet allocation assuming that some number of "topics," which are distribution over words [34]

tf-idf

A numerical statistic used for retrieval of information called tf-idf. It measures a word's significance to a corpus of documents. Search engines frequently utilize it to rank and score a document's relevancy to a user query. This can also help in filtering out stop words while classifying and summarizing material.

Advantages of Gensim

The following are some of Gensim's key benefits:

- Although other software packages like "scikit-learn" and "R" may offer topic modeling and word embedding capabilities, Gensim's capabilities are unmatched in these areas. Additionally, it offers text processing facilities that are more practical.
- Gensim's ability to handle enormous text files even without putting the entire file into memory is one of its biggest advantages.
- Gensim employs unsupervised models instead of expensive annotations or manual document labeling.

Gensim can be install using pip install method

```
pip install gensim
```

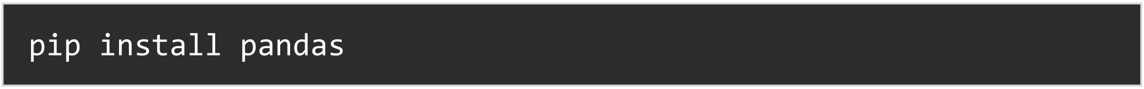
Python Pandas

The most crucial resource available to data scientists and analysts using Python today is the pandas module. Although the eye-catching visuals and strong machine learning technologies may get all the attention, pandas remains the foundation of most data initiatives.

By cleaning, manipulating, and analyzing the data with pandas, we can analyze it for the respective usage. As been shown in below figure viii, different operations including slicing, merging, joining etc. can be done using this library.

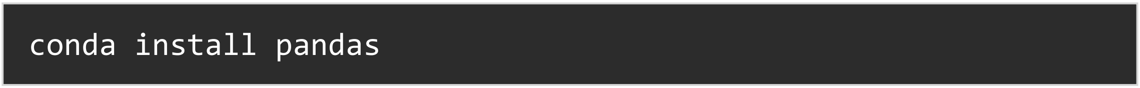
Since Pandas is built on top of the NumPy package, it uses and replicates a lot of NumPy's structure. Data from pandas is frequently fed into SciPy's statistical analysis, Matplotlib's graphing capabilities, and Scikit-machine Learn's learning algorithms. [28]

Installing the Pandas package is simple. Use one of the following commands to install it after your terminal program (for Mac users) or command line (for PC users) is open:



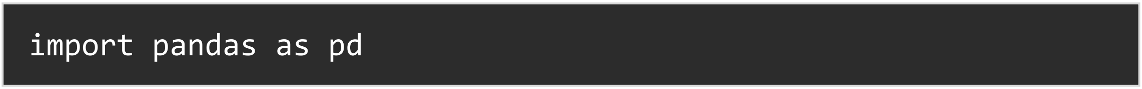
```
pip install pandas
```

or



```
conda install pandas
```

It can then be imported using the following command



```
import pandas as pd
```

We can perform a variety of operations with series, data frames, missing data, group by, etc. with Python pandas. The following diagram represents some typical data manipulation operations:

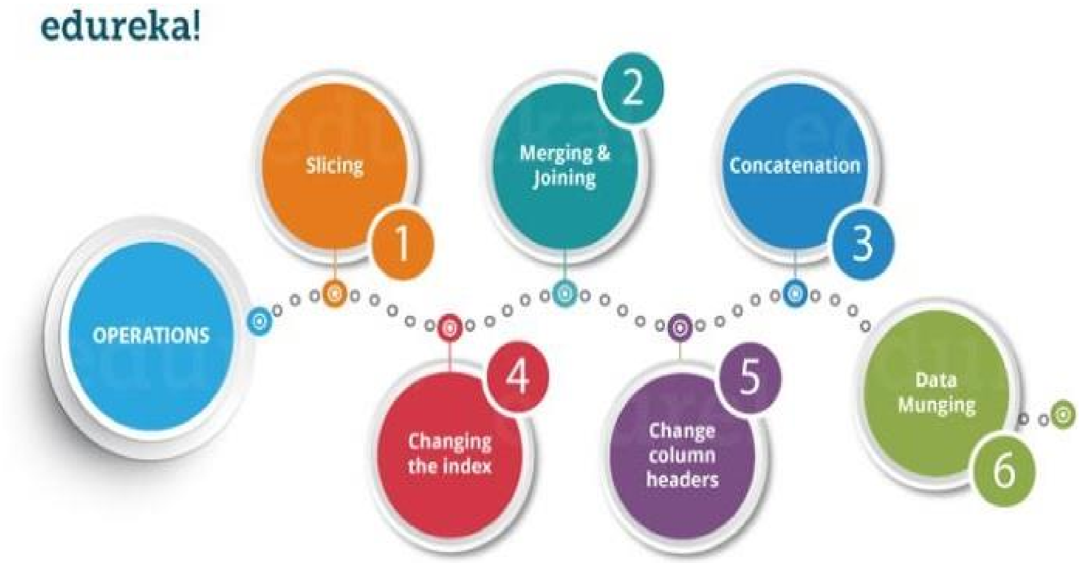


Figure viii List of Python Pandas Operations - Python Pandas Tutorial [36]

Matplotlib

For Python and its numerical extension NumPy, Matplotlib is a cross-platform data visualization and graphical representation package. As a result, it presents a strong open-source substitute for MATLAB. The APIs (Application Programming Interfaces) for matplotlib allow programmers to incorporate graphs into GUI applications. [29]

A state-based interface to matplotlib is pyplot. It offers an implicit plotting method similar to MATLAB. Additionally, it opens figures on your screen and manages the figure GUI. The primary uses of pyplot are interactive graphs and straightforward instances of programmatic plot generation.

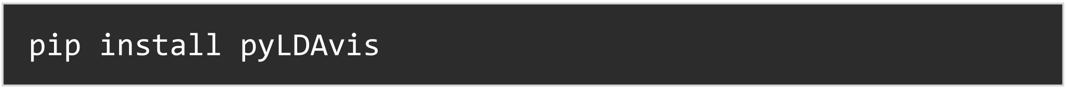
Pip method can be used to install Matplotlib. The command prompt is used to perform the following command to install Matplotlib.

```
pip install matplotlib
```

pyLDAvis

The purpose of pyLDAvis is to aid users in understanding the topics contained in a topic model that has been fitted to a corpus of text data. A fitted LDA topic model is used to extract data that the software then used to inform an interactive web-based visualization.

Similarly, as above mentioned python libraries, it can be installed simply by using pip command method.



```
pip install pyLDAvis
```

Classification of Anomaly Detection

The technique of finding irregular facts or events in datasets that deviate from the usual can be termed as anomaly detection—also known as unsupervised anomaly detection—is frequently used on data which is unlabeled. Two fundamental presumptions govern anomaly detection:

1. Anomalies are the rare defects present in the set of data.
2. Their characteristics are very different from those of typical situations.

Finding anomaly is a major business issue, which enables the discovery of aberrant occurrences and their subsequent investigation and correction. Hawkins defined an anomaly as an observation that differs so significantly from previous observations that it raises questions about whether it was produced by a separate mechanism [30]. Anomalies are also known as abnormalities, outliers, novelties, or discordant in data mining and statistics literature.

Anomalies frequently lead to some sort of issue, such as data problems, medical issues, or bank fraud, depending on the domain. Such events can be quickly responded to when they are detected early and accurately, which may be necessary to prevent potentially harmful effects. [31]

Univariate Anomaly Detection

The type of anomaly detection in which outliers are detected from a distribution of values in a single feature space.

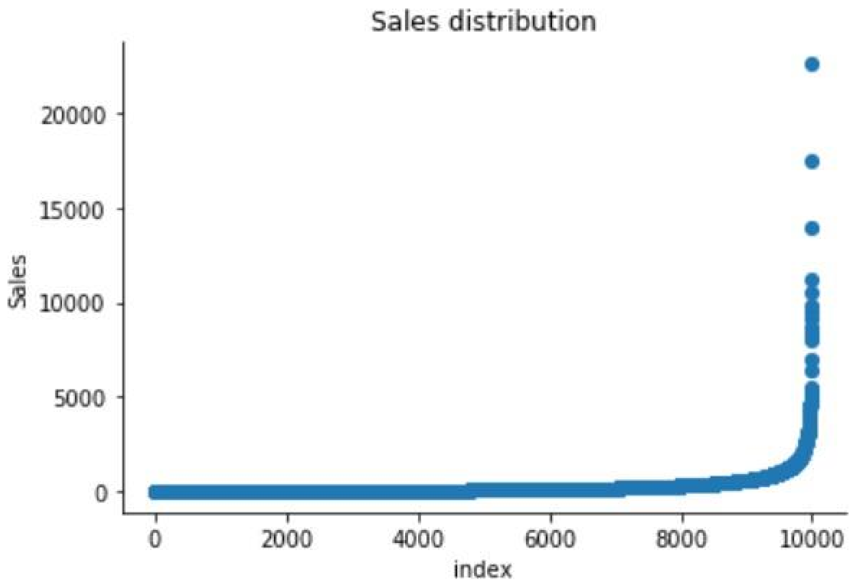


Figure ix The figure represents the exponential growth in sales with respect to the index value

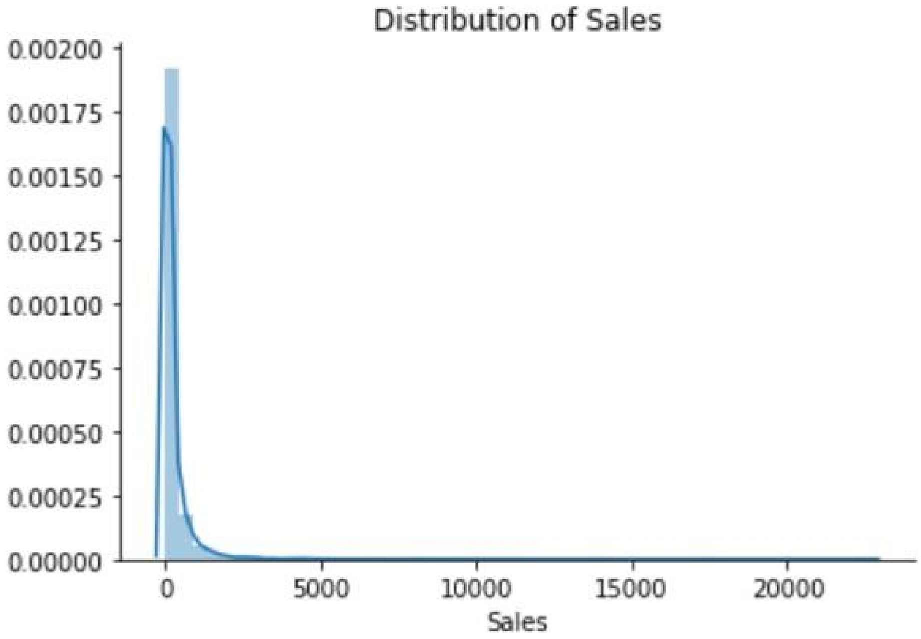


Figure x The graph between the distribution of sales with respect to Sales of a super store

The above set of graphs represents the sales distribution of a super store and is an example of univariate anomaly detection. The sales distribution of the Superstore deviates significantly from the usual distribution; it has a long thin tail which is positive, and majority of the distribution is centered to the left of the picture. The tails of the normal distribution are significantly outnumbered by the tails of the sales distribution. While on the other side of the distribution, there is one area where there is a low possibility that the data will show up.

Isolation Forest

The Isolation Forest algorithm, which is based on the idea that anomalies are data points are few and unusual, yields the anomaly score for each of the sample when used to find outliers. A model based on trees is isolation forest. In these data trees, distributions are produced by randomly choosing a feature, followed by a value which is split between that feature's max and min value.

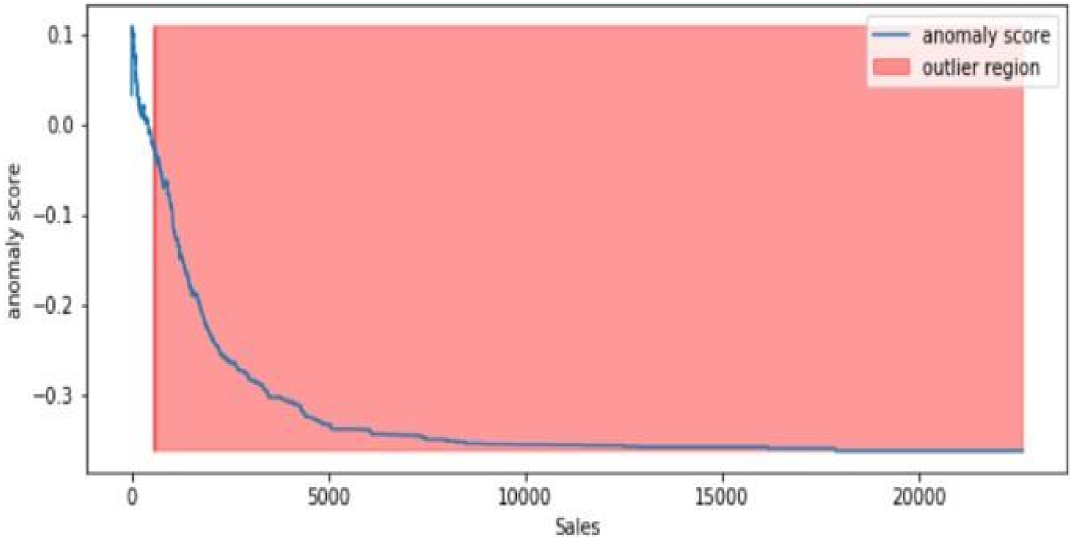


Figure xi Isolation forest model representing The anomaly score with respect to the sales, thus identifying the region in the sales which lies in the outer region

The results and graphics presented above suggest that sales exceeding 1,000 would unquestionably be regarded as an anomaly.

Anomaly Detection in Multivariate data

An outlier is an overall exceptional score for a minimum set of two variables in multivariate anomaly detection. Using the Sales and Profit variables as an example, an unsupervised multivariate anomaly detection method is built based on several models. The need is to develop a detection system which is both unsupervised and multivariate based on various models employing the profit and sales variables. It has usually been anticipated a positive correlation between sales and profit while in business. If there is a negative correlation between some of the profit data points and sales data points, such data points would be regarded as outliers and require additional research.

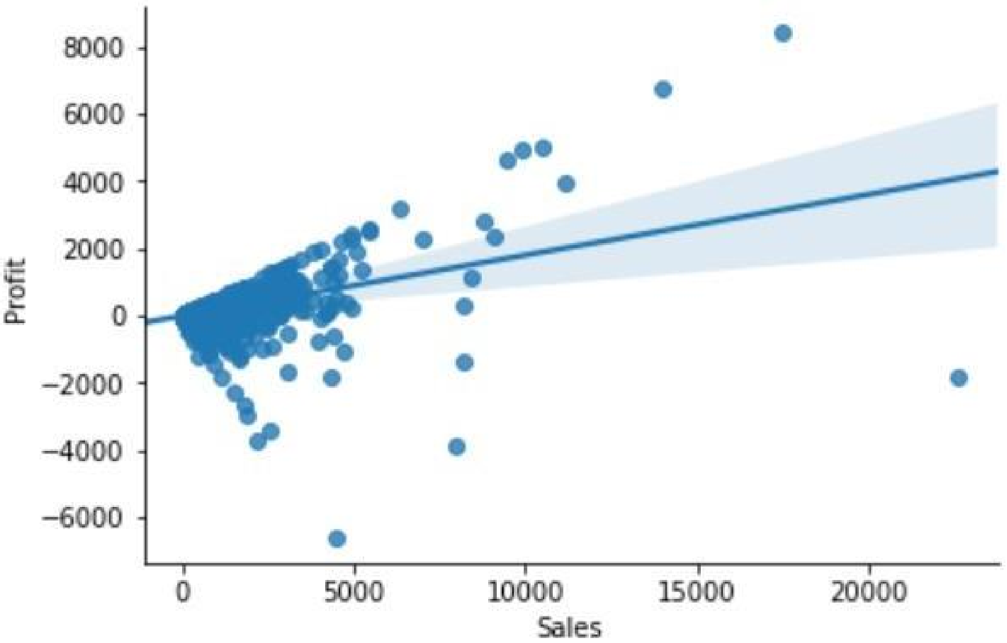


Figure xii It can be seen from the correlation graph above, some of these data points are outliers i.e., very low and high values.

Cluster-based Local Outlier Factor (CBLOF)

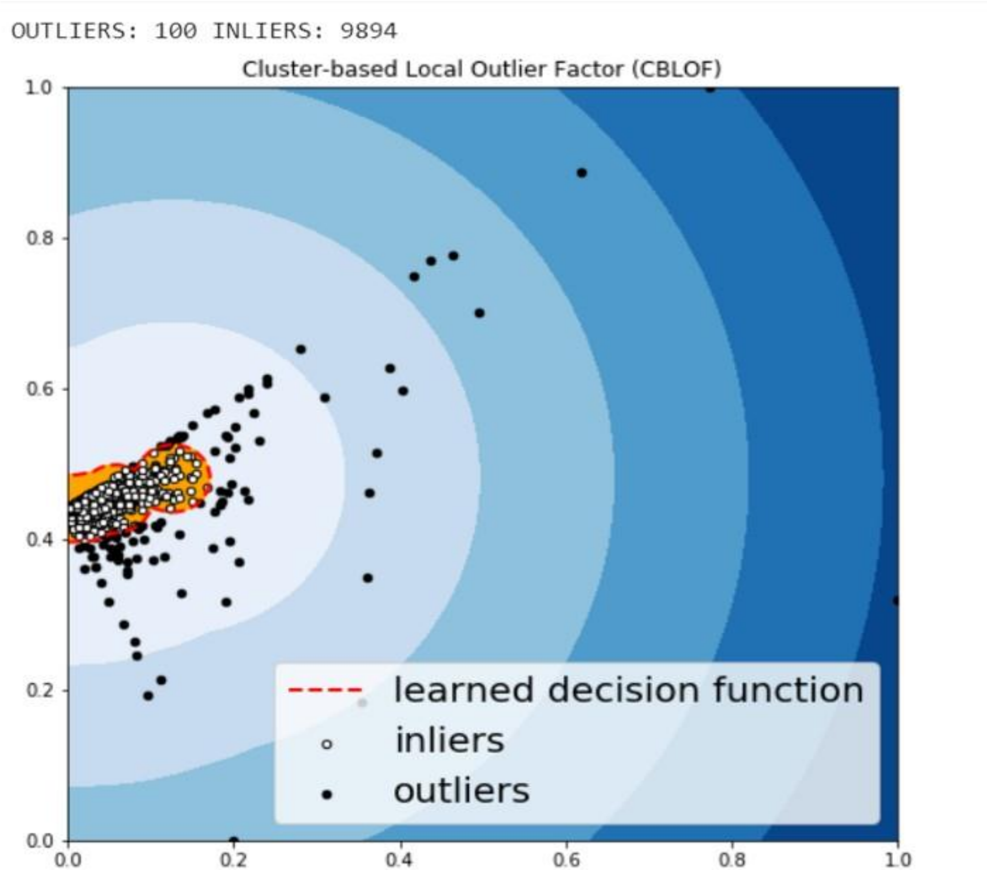
Cluster-based Outlier Detection is a technique used to identify outliers or anomalies in a dataset. This method involves dividing the data into clusters based on similarity and then identifying the instances that do not fit well with the other instances in the same cluster. These instances are considered outliers.

Cluster-based outlier detection can be performed using various clustering algorithms such as K- Means, Hierarchical Clustering, or Density-Based Clustering. The basic idea behind these algorithms is to partition the data into clusters such that similar instances are assigned to the same cluster, and dissimilar instances are assigned to different clusters.

Once the data is divided into clusters, instances that are significantly different from the other instances in the same cluster can be identified as outliers. This can be done using various metrics such as the Euclidean distance, Mahalanobis distance, or the density of the instance compared to the other instances in the cluster.

The score of an anomaly can be computed by the distance of every instance from its cluster center which is then multiplied by the instances that are belonging to its cluster. One advantage of cluster-based outlier detection is that it can handle high-dimensional data, as the data is divided into clusters based on similarity, rather than distance. Additionally, it can handle non-linear relationships between variables, which can be difficult for other outlier detection methods, such as Z-score or the modified Z-score. The process involves scaling sales and profit data between zero and one, setting a 1% outlier fraction based on trial and best guess, fitting the data to the CBLOF model, and using a threshold value to determine if a data point is an inlier or outlier. An anomaly score is calculated for each data point using a decision function.

However, the choice of clustering algorithm and the number of clusters can greatly affect the results of the outlier detection, so careful consideration should be given to the choice of method and parameters used.



```
OUTLIERS: 100 INLIERS: 9894
```

Figure xiii The data as examined from decision function, representing the possible outliers and inliers

(HBOS) Histogram-based Outlier Detection

Histogram-based Outlier Detection is a method used to identify outliers or anomalies in a dataset by constructing a histogram of the data and then determining which instances are significantly different from the majority of the data.

The basic idea behind histogram-based outlier detection is to divide the range of the data into a set of equal-width bins, and then count the number of instances that fall into each bin. The resulting histogram can be used to identify which instances are significantly different from the majority of the data. For example, instances that fall into bins with very low frequency can be considered outliers, as they are significantly different from the majority of the data. Additionally, the width of the bins can be adjusted to increase or decrease the sensitivity of the outlier detection.

One advantage of histogram-based outlier detection is that it is easy to implement and computationally efficient, making it suitable for large datasets. Additionally, it can handle non- normal distributions, which can be difficult for other outlier detection methods, such as Z-score or the modified Z-score.

However, the choice of bin width can greatly affect the results of the outlier detection, so careful consideration should be given to the choice of bin width used. Additionally, histogram- based outlier detection may not be suitable for high-dimensional data, as the data may not be well represented in a histogram.

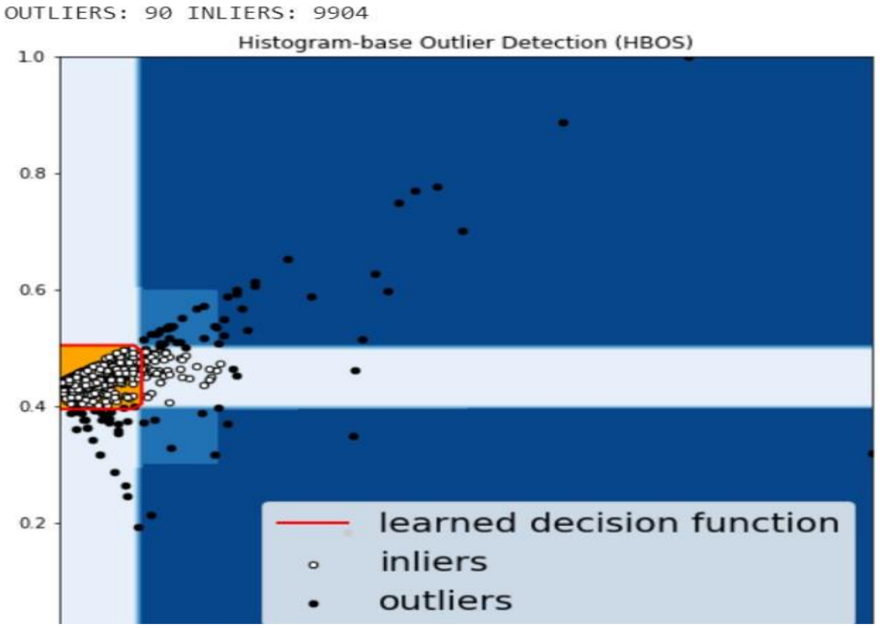


Figure xiv represents the clustered learned decision function along with possible inliers and outliers

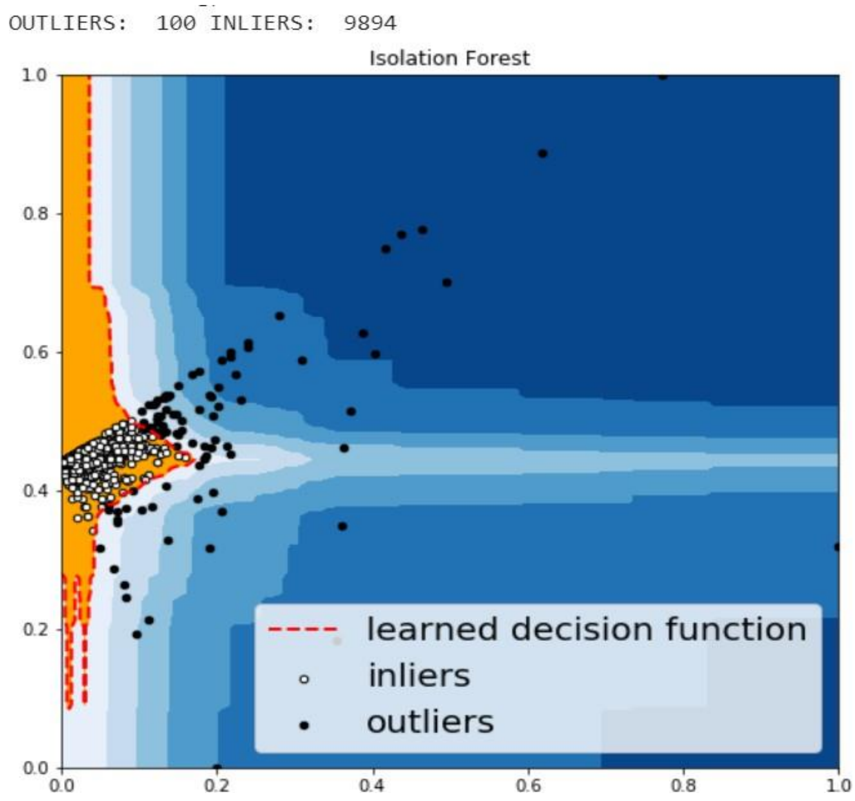
Isolation Forest

Isolation Forest is an unsupervised machine learning algorithm used for anomaly detection. It is based on the principle of isolating individual observations to identify anomalies. The algorithm works by randomly selecting a feature and then randomly selecting a split value between the minimum and maximum values of the selected feature. The process is repeated recursively on the resulting sub-samples until a certain stopping criterion is met, such as a maximum tree depth or a minimum number of samples in a leaf node.

Anomalies are defined as observations that are isolated from the rest of the data. In the Isolation Forest, the length of the path from the root node to the leaf node for an observation is used as a measure of its anomalousness. The shorter the path length, the higher the anomalousness score.

The main advantage of the Isolation Forest is that it is fast and scalable, making it suitable for large datasets. Additionally, it does not require any prior knowledge of the distribution of the data, making it suitable for data with unknown or complex distributions.

Overall, Isolation Forest is a powerful and efficient algorithm for anomaly detection, particularly when working with large and complex datasets.



```
OUTLIERS:  100 INLIERS:  9894
```

Figure xv The isolation forest model representing the clustered forest along with possible anomalies

(KNN) K - Nearest Neighbors

K-Nearest Neighbor (KNN) Anomaly Detection is a method used to identify anomalies ouutliers in a dataset. It is based on the idea that instances that are close to many other instances in the dataset are likely to be normal, while instances that are far from other instances are likely to be anomalies.

The basic idea behind KNN anomaly detection is to calculate the distances between an instance and its K nearest neighbors, and then use these distances to determine whether the instance is an anomaly. For example, if the distances between an instance and its K nearest neighbors are significantly larger than the distances between the K nearest neighbors and other instances in the dataset, then the instance is considered an anomaly.

The choice of K can greatly affect the results of the KNN anomaly detection. A large K value will result in a more conservative approach, as it will take into account a larger number of neighbors, while a small K value will result in a more sensitive approach, as it will take into account a smaller number of neighbors.

One advantage of KNN anomaly detection is that it can handle non-linear relationships between variables, which can be difficult for other outlier detection methods, such as Z-score or the modified Z-score. Additionally, it can handle high-dimensional data, as it only considers the distances between instances, rather than the actual values of the variables.

However, KNN anomaly detection can be computationally expensive, as it requires the calculation of distances between all instances, which can be time-consuming for large datasets. Additionally, the choice of distance metric can greatly affect the results of the KNN anomaly detection, so careful consideration should be given to the choice of distance metric used.

```
OUTLIERS:  91 INLIERS:  9903
```

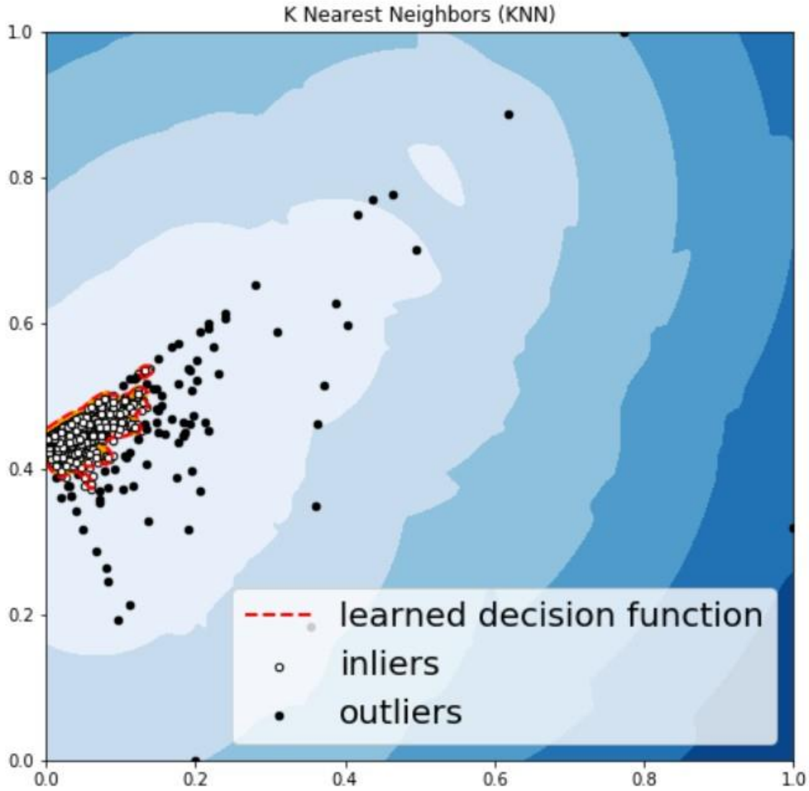


Figure xvi The clustered learned decision function represented by the red outline along with outliers and inliers

Methodology

Machine Learning

Machine learning is a subfield of artificial intelligence (AI) concerned with the design and development of algorithms that can learn patterns and knowledge from data and make predictions or decisions without being explicitly programmed to do so. It involves using statistical techniques to enable a computer to identify patterns and trends in data and make predictions based on that data.

Machine learning algorithms can be classified into three main categories: supervised learning, unsupervised learning, and reinforcement learning.

Supervised Learning

Supervised machine learning is a type of machine learning in which a model is trained on a labeled dataset, where the desired output labels are provided. The goal of supervised learning is to learn a mapping between the input features and the output labels, such that given a new, unseen instance, the model can predict its label accurately.

Examples of supervised machine learning tasks include:

a. Regression (predicting a continuous value)
b. Classification (predicting a categorical label)

Supervised learning algorithms include linear regression, logistic regression, decision trees, random forests, support vector machines (SVMs), and neural networks.

Unsupervised Learning

Unsupervised machine learning is a type of machine learning where the algorithm is not provided with labeled training data. Instead, the goal is to discover hidden patterns or relationships in the data, without any prior knowledge or guidance.

Examples of unsupervised machine learning tasks include:

a) Clustering (grouping similar data points together)
b) Dimensionality reduction (reducing the number of features while preserving the underlying structure of the data)

Unsupervised learning algorithms include K-means, hierarchical clustering, principal component analysis (PCA), and auto encoders

Reinforcement Learning

Reinforcement learning is a type of machine learning where an agent learns to make decisions in an environment by performing actions and receiving rewards or penalties based on the outcome of those actions. The goal of reinforcement learning is for the agent to learn a policy that maximizes the cumulative reward over time.

In reinforcement learning, the learning process involves trial and error. The agent takes actions, receives feedback in the form of rewards or penalties, and updates its policy based on the feedback. Over time, the agent's policy should converge to an optimal policy that maximizes the reward.

Examples of reinforcement learning tasks include:

- Game playing (such as chess or Go)
- Robotics (control of physical robots)
- Resource allocation (such as energy management in a smart grid)
  Reinforcement learning algorithms include Q-learning, SARSA, and deep reinforcement learning.

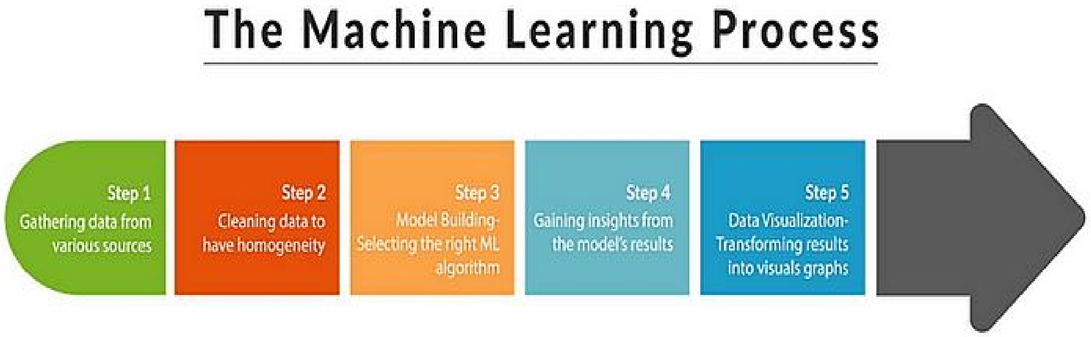
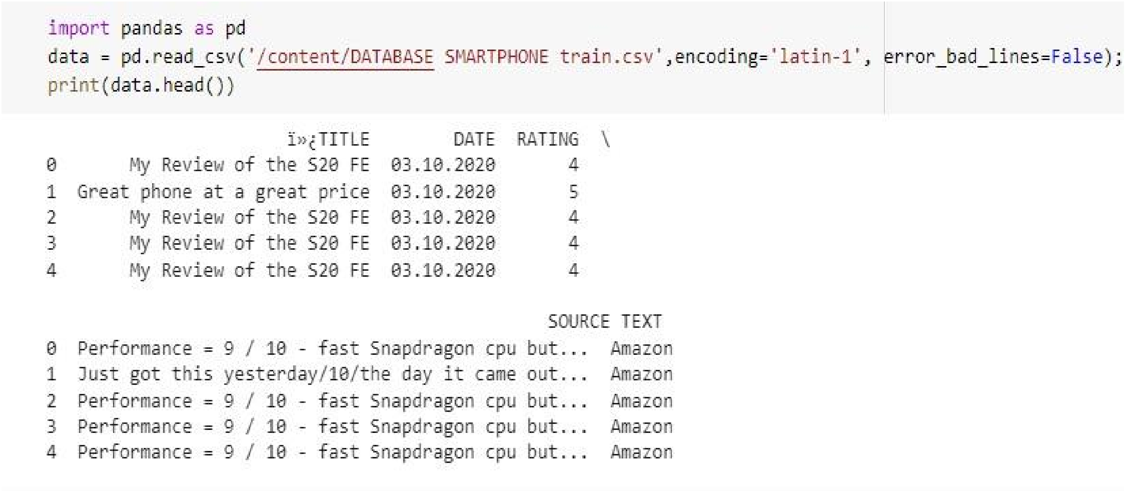

The Machine Learning Process
Step 1
Gathering data from various sources
Step 2
Cleaning data to have homogeneity
Step 3
Model Building-Selecting the right ML algorithm
Step 4
Gaining insights from the model's results
Step 5
Data Visualization-Transforming results into visuals graphs

Figure xvii The flow of machine learning process. Initiating from gathering of the required data, after cleaning and modeling, it transform results into visual graph

Loading and Cleaning of Data

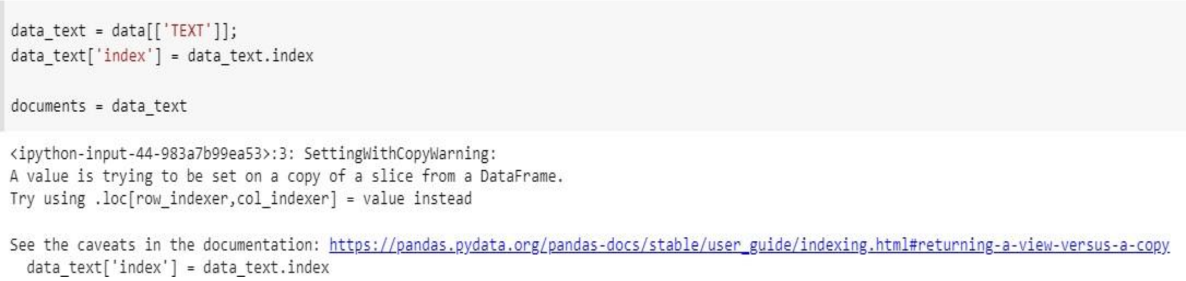
The data we gathered is user feedback about smartphones. The data has been split with major portion of it attributed for training the model and remaining is used to test the model and Identifying the anomalies. This process of splitting of data has been done manually with almost half of the dataset has been allocated to training and the remaining half for testing.



```
import pandas as pd
data = pd.read_csv('/content/DATABASE SMARTPHONE train.csv',encoding='latin-1', error_bad_lines=False);
print(data.head())
```

```
                      ï»¿TITLE        DATE  RATING  \
0       My Review of the S20 FE  03.10.2020       4
1  Great phone at a great price  03.10.2020       5
2       My Review of the S20 FE  03.10.2020       4
3       My Review of the S20 FE  03.10.2020       4
4       My Review of the S20 FE  03.10.2020       4

                                          SOURCE TEXT
0  Performance = 9 / 10 - fast Snapdragon cpu but...  Amazon
1  Just got this yesterday/10/the day it came out...  Amazon
2  Performance = 9 / 10 - fast Snapdragon cpu but...  Amazon
3  Performance = 9 / 10 - fast Snapdragon cpu but...  Amazon
4  Performance = 9 / 10 - fast Snapdragon cpu but...  Amazon
```

Figure xviii represents importing Pandas library to read the csv file of training dataset and then printing the header part



```
data_text = data[['TEXT']];
data_text['index'] = data_text.index

documents = data_text
```

```
<ipython-input-44-983a7b99ea53>:3: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  data_text['index'] = data_text.index
```

Figure xix The text column of the data is passed to data_text variable which then stored in documents variable

Data Preprocessing

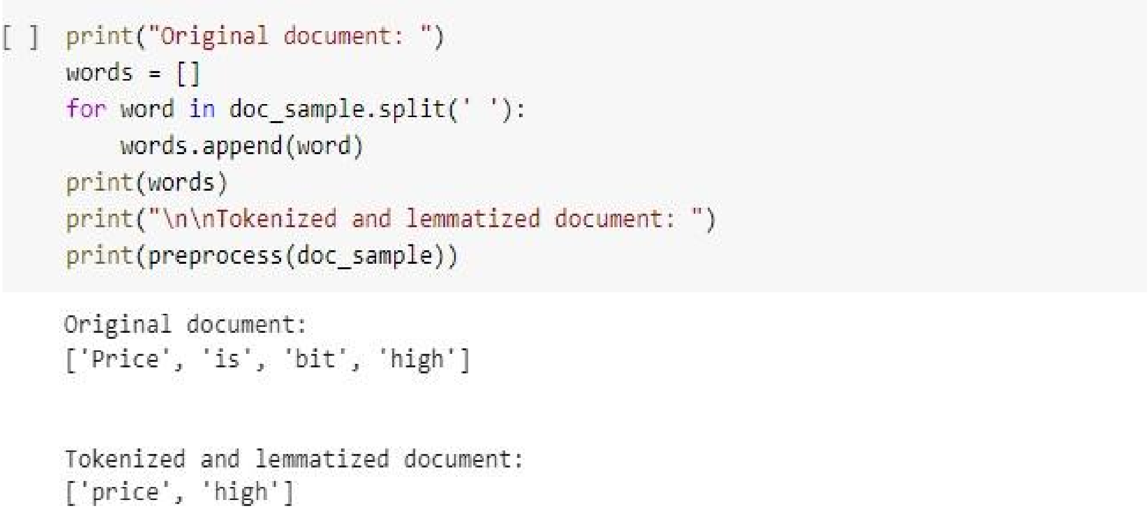
The output of the preceding cells shows that the data is not yet ready for the topic modeling assignment. Punctuation, content words, and other features are among the numerous components that cause "noise" in the data. The usual processes for the cleaning can be consist of stemming, bigrams, trigrams and removal of stop words as shown in figure XXiii. As the input utilized for the model is essential to the effectiveness of a language model, the preparation of textual data can be time-consuming. The following pre-processing procedures are applied based on the data used in this notebook:

Tokenization

In a sentence, a token is the word or punctuation mark that appears there. The technique of tokenization involves separating the phrases into their component words and punctuation. This procedure is advantageous because it separates the text data into manageable chunks, making it simpler for a language model for distinguishing.

Lemmatization

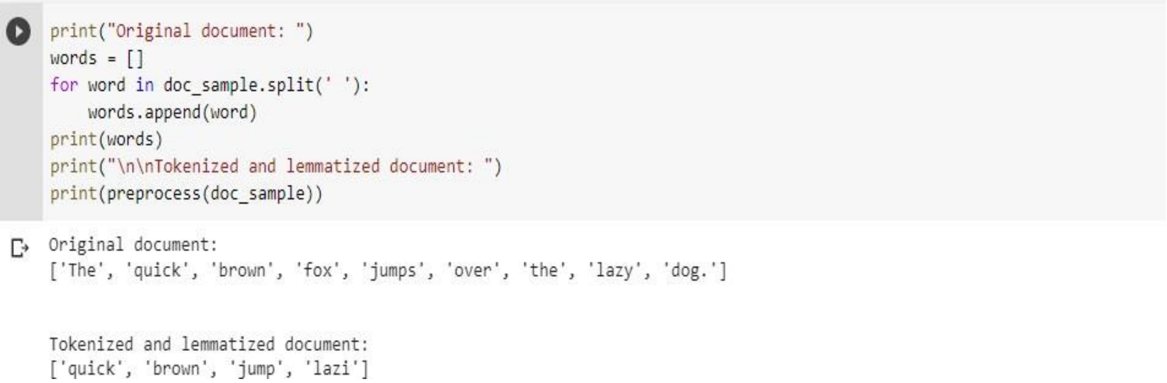
A lemma is a token's root form; for example, the word "undivided" in a sentence is a token, and the word "divide" would be the corresponding lemma. In this instance, lemmatization is being used to avoid topics like "books" and "book" being used twice.



```
[ ] print("Original document: ")
    words = []
    for word in doc_sample.split(' '):
        words.append(word)
    print(words)
    print("\n\nTokenized and lemmatized document: ")
    print(preprocess(doc_sample))
```

```
Original document:
['Price', 'is', 'bit', 'high']


Tokenized and lemmatized document:
['price', 'high']
```

Figure xx The code used to tokenized and lemmatized the required document



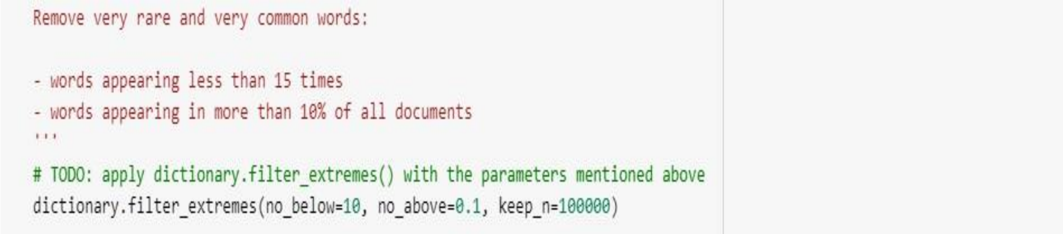
```
print("Original document: ")
words = []
for word in doc_sample.split(' '):
    words.append(word)
print(words)
print("\n\nTokenized and lemmatized document: ")
print(preprocess(doc_sample))
```

```
Original document:
['The', 'quick', 'brown', 'fox', 'jumps', 'over', 'the', 'lazy', 'dog.']


Tokenized and lemmatized document:
['quick', 'brown', 'jump', 'lazi']
```

Figure xxi Example of lemmatization and tokenization representing inclusion of important words while rejecting the common repeated words

Filtering

Eliminating terms like pronouns, determiners, and conjunctions that have no semantic value. This lessens the data's "noise" and aids in the training of the language model.



```
Remove very rare and very common words:

- words appearing less than 15 times
- words appearing in more than 10% of all documents
'''
# TODO: apply dictionary.filter_extremes() with the parameters mentioned above
dictionary.filter_extremes(no_below=10, no_above=0.1, keep_n=100000)
```

Figure xxii The code to filter the extremes with respect to the described parameters

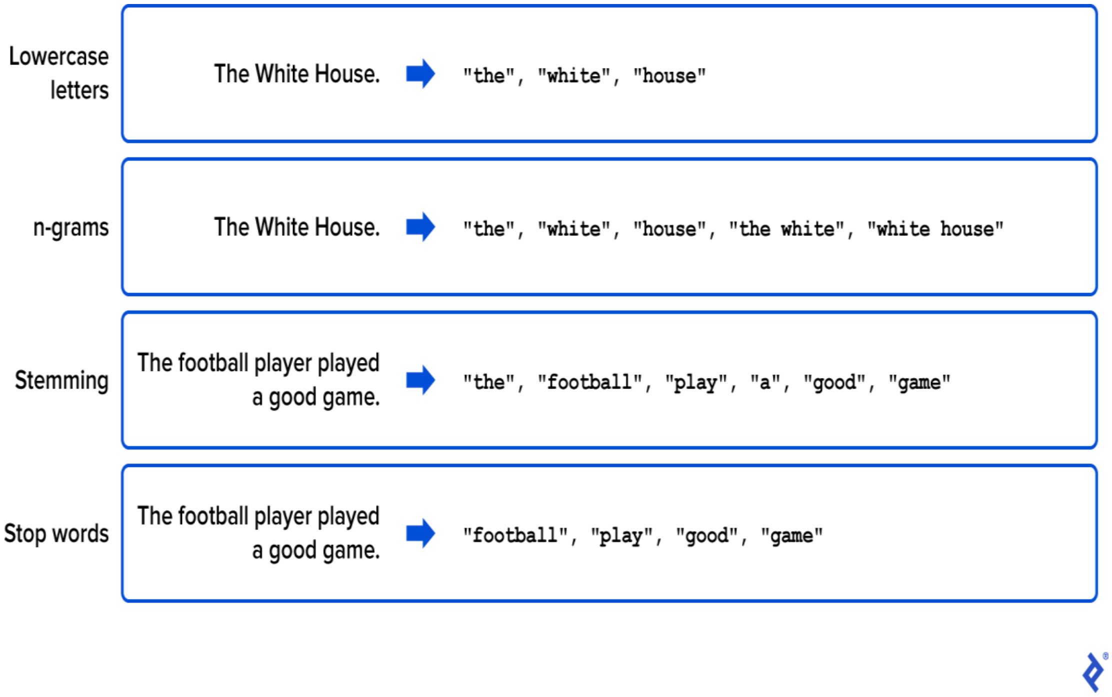
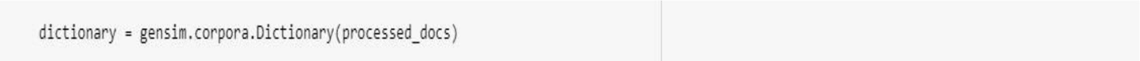

Lowercase letters
The White House.
"the", "white", "house"
n-grams
The White House.
"the", "white", "house", "the white", "white house"
Stemming
The football player played a good game.
"the", "football", "play", "a", "good", "game"
Stop words
The football player played a good game.
"football", "play", "good", "game"

Figure xxiv The diagram representing the types of data preprocessing techniques

Creating a dictionary

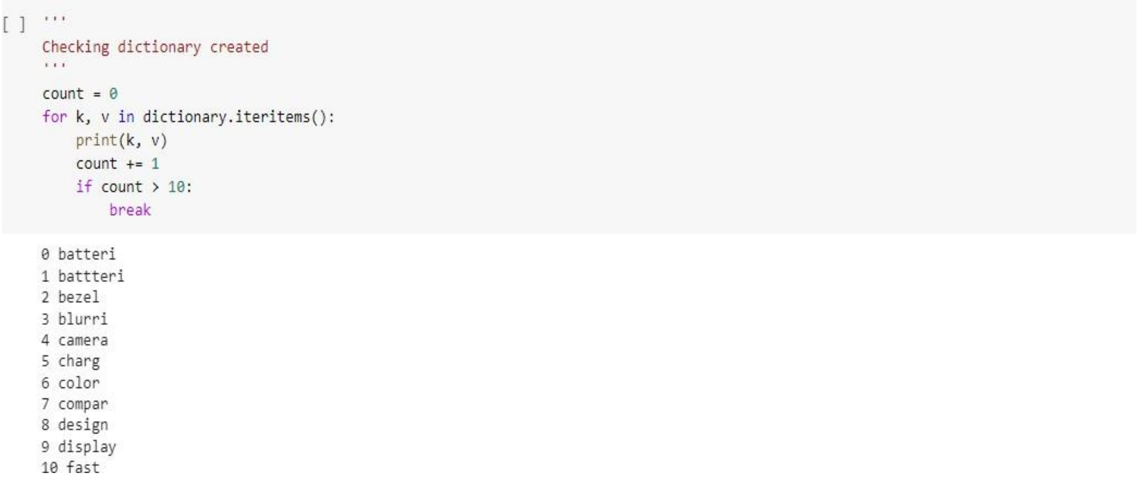
After preprocessing of data, the next step consists of forming a dictionary using Gensim doc2bow.

The dictionary is created from the "processed docs" that includes the frequency of each term in the training set. After that, a dictionary reporting the number of words and the frequency with which each word appears should be created for each document.



```
dictionary = gensim.corpora.Dictionary(processed_docs)
```

Figure xxvi The code to create a dictionary using the preprocessed data



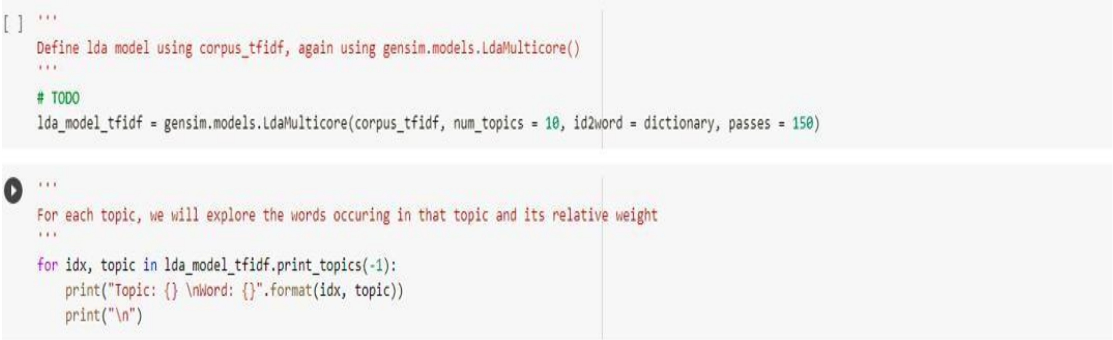
```
[ ] '''
    Checking dictionary created
    '''
    count = 0
    for k, v in dictionary.iteritems():
        print(k, v)
        count += 1
        if count > 10:
            break
```

```
0 batteri
1 battteri
2 bezel
3 blurri
4 camera
5 charg
6 color
7 compar
8 design
9 display
10 fast
```

Figure xxvii The code to check the items created inside the dictionary using for loop

TF-IDF (term frequency-inverse document frequency)

A technique for weighing the significance of terms in a document in the context of a corpus of texts is called TF-IDF (term frequency-inverse document frequency). It is frequently used in text mining and information retrieval applications to increase the efficiency of text representations that are in the form of bag-of-words.



```
'''
Define lda model using corpus_tfidf, again using gensim.models.LdaMulticore()
'''
# TODO
lda_model_tfidf = gensim.models.LdaMulticore(corpus_tfidf, num_topics = 10, id2word = dictionary, passes = 150)
```

```
'''
For each topic, we will explore the words occuring in that topic and its relative weight
'''
for idx, topic in lda_model_tfidf.print_topics(-1):
    print("Topic: {} \nWord: {}".format(idx, topic))
    print("\n")
```

Figure xxviii The code by which LDA-TFIDF model is created and by using for loop it generates a list of 10 most discussed topics in the dataset

The fundamental principle underlying TF-IDF is to weigh the significance of each word in a text by taking into account both its term frequency (TF) and its overall corpus of document frequency (inverse document frequency, IDF). This is done by determining the TF-IDF values: multiplying term frequency (TF) of a word in a document by inverse document frequency (IDF), the process is done over the entire corpus of documents.

This method can decrease the dimensionality of data thus, increasing the weight of significant words over less important ones by employing TF-IDF values as inputs. Additionally, it helps lowering the drawback of bag-of-words approach i.e. assigning a large weight for a common word across all documents. The bag of words representation of matrix model can be viewed as below. Each sentence is break down into different words with each representation of the word is accounted into the relevant matrix model.

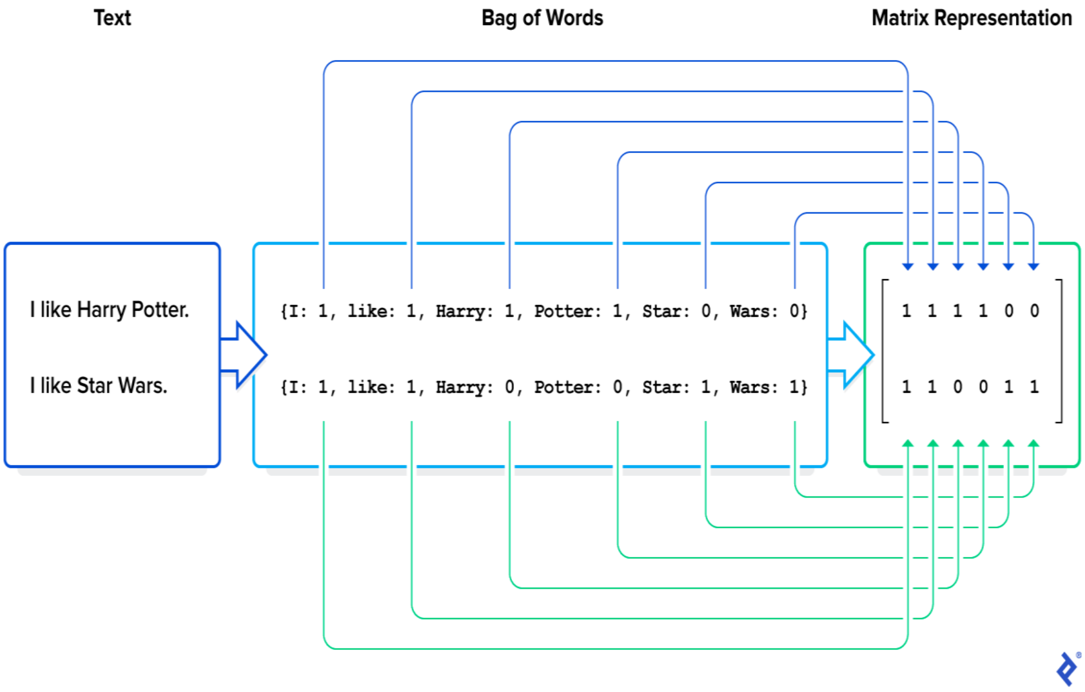

Text
Bag of Words
Matrix Representation
I like Harry Potter.
I like Star Wars.
{I: 1, like: 1, Harry: 1, Potter: 1, Star: 0, Wars: 0}
{I: 1, like: 1, Harry: 0, Potter: 0, Star: 1, Wars: 1}
1 1 1 1 0 0
1 1 0 0 1 1

Figure xxix The matrix representation of the bag of words

The Latent Dirichlet Allocation

The next step consist of training the Latent Dirichlet Allocation model using bag-of-words and TF-IDF corpus respectively. The parameters for the given model are as follows:

num_topics - It represents the number of latent topics requested that the training corpus extracts.

Id2word- It is the mapping or converting the ids of words from integers to strings. This helps in determining the size of vocabulary along with the use for topic printing and debugging.

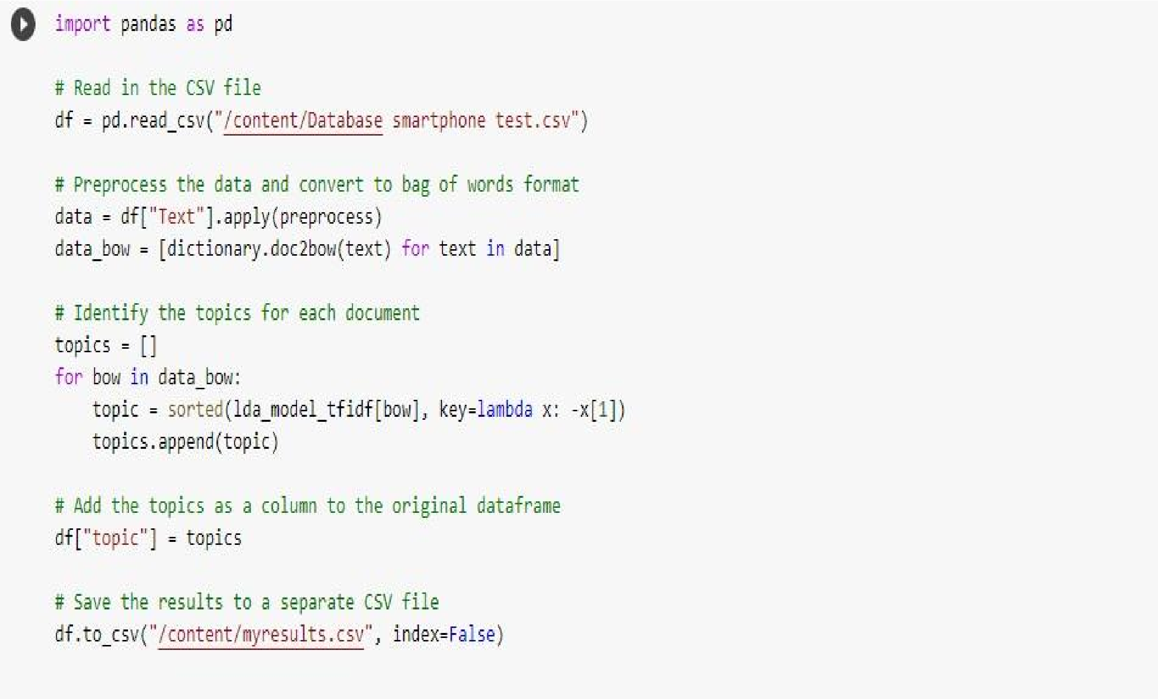
Apha and eta – These hyperparameters that influence the sparsity of distributions of topic- document and topic word. For the sake of simplicity these can be left thus making them default (1/num_topics).

Passes- It represents the number of times the whole algorithm passes over the training dataset. The more passes and thus, more accurate the model will become and will take long time to process.

For increasing the efficiency, LDA multicore is employed which uses all the ports of the CPU and so increases the time to model. The model will provide the desired number of topics from the training dataset. These topics consist of words and their respective percentage weightage in that particular topic.

Using the Model on Testing Dataset

The model is now tested on a training data set. This represent the probability of the dataset of being part of a particular topic among num_topics. With the help of python code every single customer review is checked on every single topic given by the Topic modeling algorithm. The minimum threshold is set to be 0.1(10%) which means that any review having a probability of less than 10 percent in the given review is neglected.

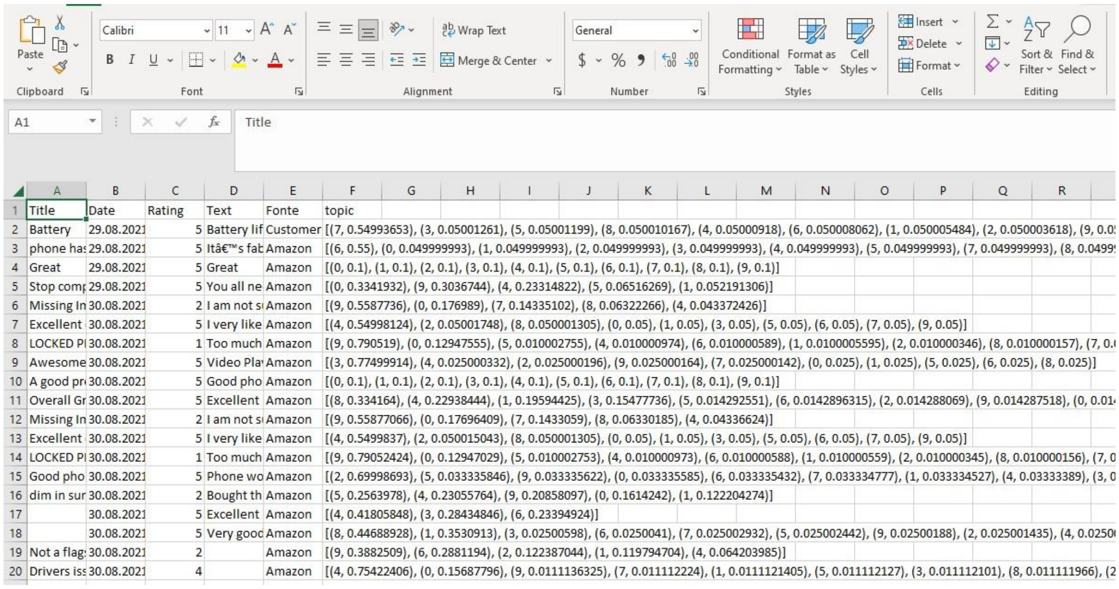


```
import pandas as pd

# Read in the CSV file
df = pd.read_csv("/content/Database smartphone test.csv")

# Preprocess the data and convert to bag of words format
data = df["Text"].apply(preprocess)
data_bow = [dictionary.doc2bow(text) for text in data]

# Identify the topics for each document
topics = []
for bow in data_bow:
    topic = sorted(lda_model_tfidf[bow], key=lambda x: -x[1])
    topics.append(topic)

# Add the topics as a column to the original dataframe
df["topic"] = topics

# Save the results to a separate CSV file
df.to_csv("/content/myresults.csv", index=False)
```

Figure xxx The code used to check the testing dataset on the basis of the created LDA model and then saving the scoring results to a csv file

Scraping the Results

The testing results are scraped and transfer into an excel sheet in comma separated value CSV format. This is done to provide data analysis using Power BI, Tableu and MS Excel.



	A	B	C	D	E	F
1	Title	Date	Rating	Text	Fonte	topic
2	Battery	29.08.2021	5	Battery lif	Customer	[(7, 0.54993653), (3, 0.05001261), (5, 0.05001199), (8, 0.050010167), (4, 0.05000918), (6, 0.050008062), (1, 0.050005484), (2, 0.050003618), (9, 0.05
3	phone has	29.08.2021	5	Itâ€™s fab	Amazon	[(6, 0.55), (0, 0.049999993), (1, 0.049999993), (2, 0.049999993), (3, 0.049999993), (4, 0.049999993), (5, 0.049999993), (7, 0.049999993), (8, 0.04999
4	Great	29.08.2021	5	Great	Amazon	[(0, 0.1), (1, 0.1), (2, 0.1), (3, 0.1), (4, 0.1), (5, 0.1), (6, 0.1), (7, 0.1), (8, 0.1), (9, 0.1)]
5	Stop comp	29.08.2021	5	You all ne	Amazon	[(0, 0.3341932), (9, 0.3036744), (4, 0.23314822), (5, 0.06516269), (1, 0.052191306)]
6	Missing In	30.08.2021	2	I am not s	Amazon	[(9, 0.5587736), (0, 0.176989), (7, 0.14335102), (8, 0.06322266), (4, 0.043372426)]
7	Excellent	30.08.2021	5	I very like	Amazon	[(4, 0.54998124), (2, 0.05001748), (8, 0.050001305), (0, 0.05), (1, 0.05), (3, 0.05), (5, 0.05), (6, 0.05), (7, 0.05), (9, 0.05)]
8	LOCKED Pl	30.08.2021	1	Too much	Amazon	[(9, 0.790519), (0, 0.12947555), (5, 0.010002755), (4, 0.010000974), (6, 0.010000589), (1, 0.0100005595), (2, 0.010000346), (8, 0.010000157), (7, 0.0
9	Awesome	30.08.2021	5	Video Pla	Amazon	[(3, 0.77499914), (4, 0.025000332), (2, 0.025000196), (9, 0.025000164), (7, 0.025000142), (0, 0.025), (1, 0.025), (5, 0.025), (6, 0.025), (8, 0.025)]
10	A good pr	30.08.2021	5	Good pho	Amazon	[(0, 0.1), (1, 0.1), (2, 0.1), (3, 0.1), (4, 0.1), (5, 0.1), (6, 0.1), (7, 0.1), (8, 0.1), (9, 0.1)]
11	Overall Gr	30.08.2021	5	Excellent	Amazon	[(8, 0.334164), (4, 0.22938444), (1, 0.19594425), (3, 0.15477736), (5, 0.014292551), (6, 0.0142896315), (2, 0.014288069), (9, 0.014287518), (0, 0.01
12	Missing In	30.08.2021	2	I am not s	Amazon	[(9, 0.55877066), (0, 0.17696409), (7, 0.1433059), (8, 0.06330185), (4, 0.04336624)]
13	Excellent	30.08.2021	5	I very like	Amazon	[(4, 0.5499837), (2, 0.050015043), (8, 0.050001305), (0, 0.05), (1, 0.05), (3, 0.05), (5, 0.05), (6, 0.05), (7, 0.05), (9, 0.05)]
14	LOCKED Pl	30.08.2021	1	Too much	Amazon	[(9, 0.79052424), (0, 0.12947029), (5, 0.010002753), (4, 0.010000973), (6, 0.010000588), (1, 0.010000559), (2, 0.010000345), (8, 0.010000156), (7, 0
15	Good pho	30.08.2021	5	Phone wo	Amazon	[(2, 0.69998693), (5, 0.033335846), (9, 0.033335622), (0, 0.033335585), (6, 0.033335432), (7, 0.033334777), (1, 0.033334527), (4, 0.03333389), (3, 0
16	dim in sur	30.08.2021	2	Bought th	Amazon	[(5, 0.2563978), (4, 0.23055764), (9, 0.20858097), (0, 0.1614242), (1, 0.122204274)]
17		30.08.2021	5	Excellent	Amazon	[(4, 0.41805848), (3, 0.28434846), (6, 0.23394924)]
18		30.08.2021	5	Very good	Amazon	[(8, 0.44688928), (1, 0.3530913), (3, 0.02500598), (6, 0.0250041), (7, 0.025002932), (5, 0.025002442), (9, 0.02500188), (2, 0.025001435), (4, 0.0250
19	Not a flag	30.08.2021	2		Amazon	[(9, 0.3882509), (6, 0.2881194), (2, 0.122387044), (1, 0.119794704), (4, 0.064203985)]
20	Drivers iss	30.08.2021	4		Amazon	[(4, 0.75422406), (0, 0.15687796), (9, 0.0111136325), (7, 0.011112224), (1, 0.0111121405), (5, 0.011112127), (3, 0.011112101), (8, 0.011111966), (2

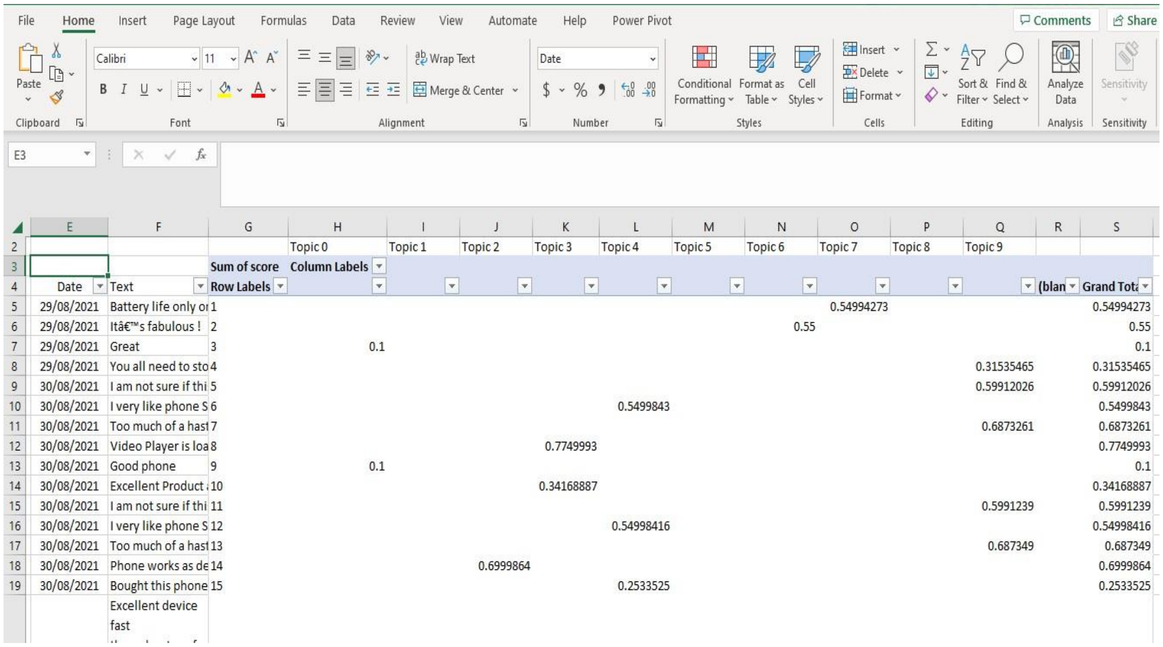
Figure xxxi The data as saved after scoring the testing dataset. Notice a separate column of Topic is created representing the score of that particular review on the list of topics

Results and Analysis

The extracted results are analyzed to check any possible anomaly present in the customer reviews. This includes first arranging the data properly in the excel and formatting it according to the Tab spaces. The topics that are presented in word combination form with percentages should be changed into a label form. The labeling can be done based on idea of discussion hinted from the words in the topic model.

PIVOT TABLE

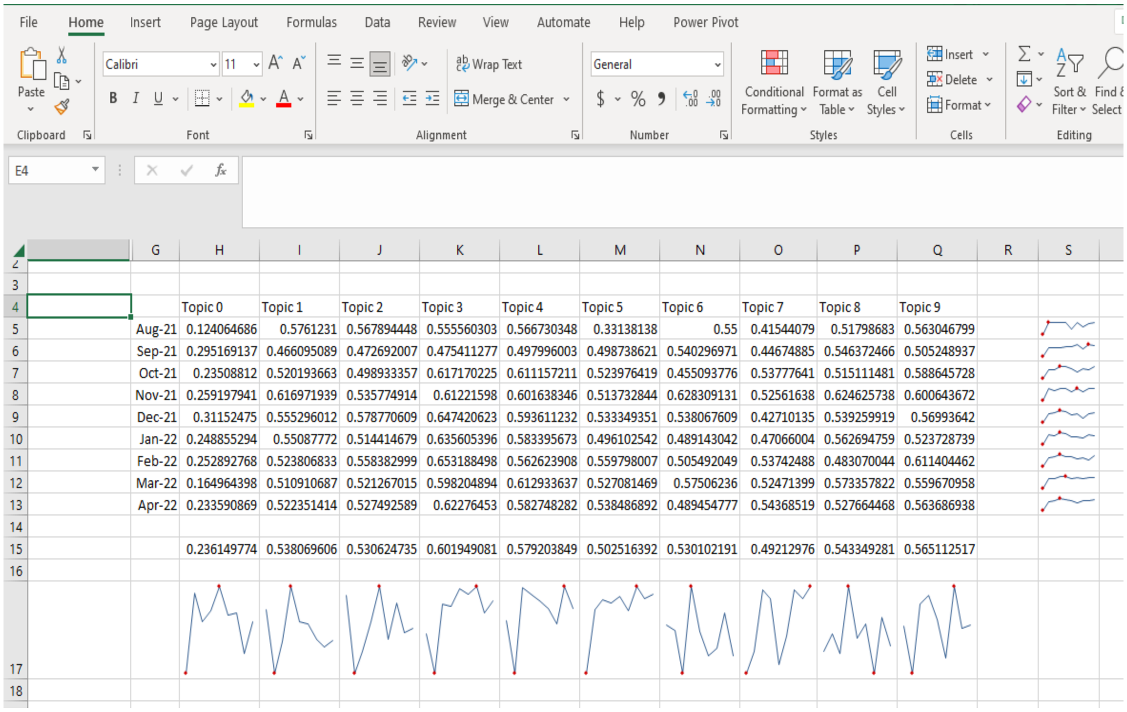
The data is then classified on monthly basis with the help of using pivot table. This helps in categorizing the data according to the respective scores and Document numbers.



	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
2				Topic 0	Topic 1	Topic 2	Topic 3	Topic 4	Topic 5	Topic 6	Topic 7	Topic 8	Topic 9		
3			Sum of score	Column Labels											
4	Date	Text	Row Labels											(blan	Grand Tota
5	29/08/2021	Battery life only or	1								0.54994273				0.54994273
6	29/08/2021	Itâ€™s fabulous !	2							0.55					0.55
7	29/08/2021	Great	3	0.1											0.1
8	29/08/2021	You all need to sto	4										0.31535465		0.31535465
9	30/08/2021	I am not sure if thi	5										0.59912026		0.59912026
10	30/08/2021	I very like phone S	6					0.5499843							0.5499843
11	30/08/2021	Too much of a hast	7										0.6873261		0.6873261
12	30/08/2021	Video Player is loa	8				0.7749993								0.7749993
13	30/08/2021	Good phone	9	0.1											0.1
14	30/08/2021	Excellent Product	10				0.34168887								0.34168887
15	30/08/2021	I am not sure if thi	11										0.5991239		0.5991239
16	30/08/2021	I very like phone S	12					0.54998416							0.54998416
17	30/08/2021	Too much of a hast	13										0.687349		0.687349
18	30/08/2021	Phone works as de	14			0.6999864									0.6999864
19	30/08/2021	Bought this phone	15					0.2533525							0.2533525
		Excellent device fast													

Figure xxxii The Pivot table is created which separates customer reviews with respect to the topics and their relative score

The average of topics in every single month is calculated and is then plotted on the graph. The data is segregated on the monthly basis to examine the changes in the Topics with respect to the months. The data is distributed so that the pattern of discussion of every single topic can be visualized by the average score of the topic in that particular month. Anomalous behavior of the data can be observed by looking at the extremities as represented by the graph.



	Topic 0	Topic 1	Topic 2	Topic 3	Topic 4	Topic 5	Topic 6	Topic 7	Topic 8	Topic 9
Aug-21	0.124064686	0.5761231	0.567894448	0.555560303	0.566730348	0.33138138	0.55	0.41544079	0.51798683	0.563046799
Sep-21	0.295169137	0.466095089	0.472692007	0.475411277	0.497996003	0.498738621	0.540296971	0.44674885	0.546372466	0.505248937
Oct-21	0.23508812	0.520193663	0.498933357	0.617170225	0.611157211	0.523976419	0.455093776	0.53777641	0.515111481	0.588645728
Nov-21	0.259197941	0.616971939	0.535774914	0.61221598	0.601638346	0.513732844	0.628309131	0.52561638	0.624625738	0.600643672
Dec-21	0.31152475	0.555296012	0.578770609	0.647420623	0.593611232	0.533349351	0.538067609	0.42710135	0.539259919	0.56993642
Jan-22	0.248855294	0.55087772	0.514414679	0.635605396	0.583395673	0.496102542	0.489143042	0.47066004	0.562694759	0.523728739
Feb-22	0.252892768	0.523806833	0.558382999	0.653188498	0.562623908	0.559798007	0.505492049	0.53742488	0.483070044	0.611404462
Mar-22	0.164964398	0.510910687	0.521267015	0.598204894	0.612933637	0.527081469	0.57506236	0.52471399	0.573357822	0.559670958
Apr-22	0.233590869	0.522351414	0.527492589	0.62276453	0.582748282	0.538486892	0.489454777	0.54368519	0.527664468	0.563686938
	0.236149774	0.538069606	0.530624735	0.601949081	0.579203849	0.502516392	0.530102191	0.49212976	0.543349281	0.565112517

Figure xxxiii The data is gathered on the monthly basis of the 10 topics which is then used to identify anomalies

The final graph consisting of monthly average of every single topic discussed in the form of graph. The extreme points can be noted in the graph to look for the possible anomalies in the dataset.

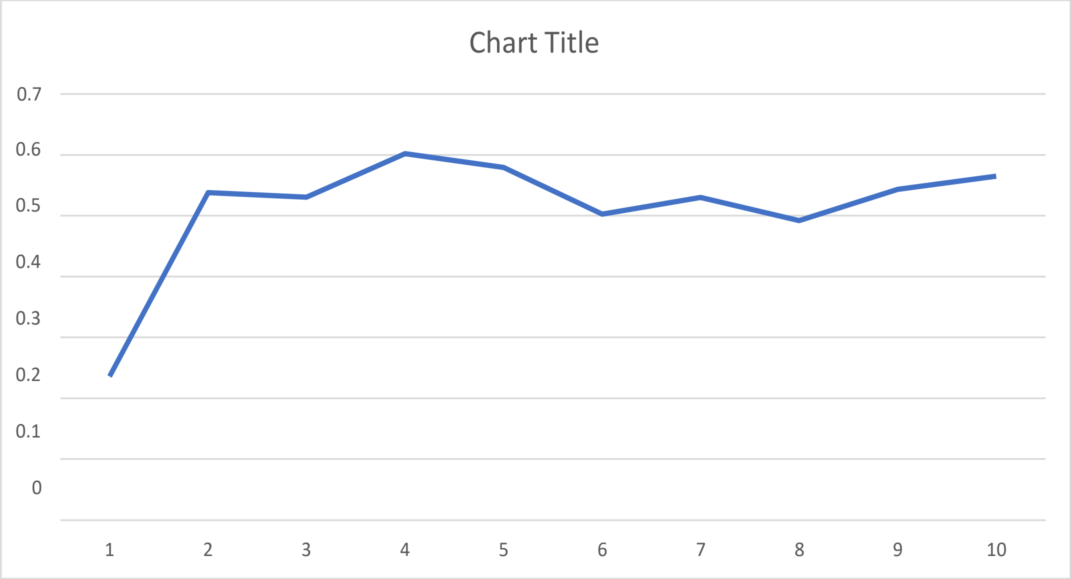


Figure xxxiv The final graph of the 10 topics representing the monthly average of each and every single topic

Anomaly has been detected by using the the following formulas in excel.

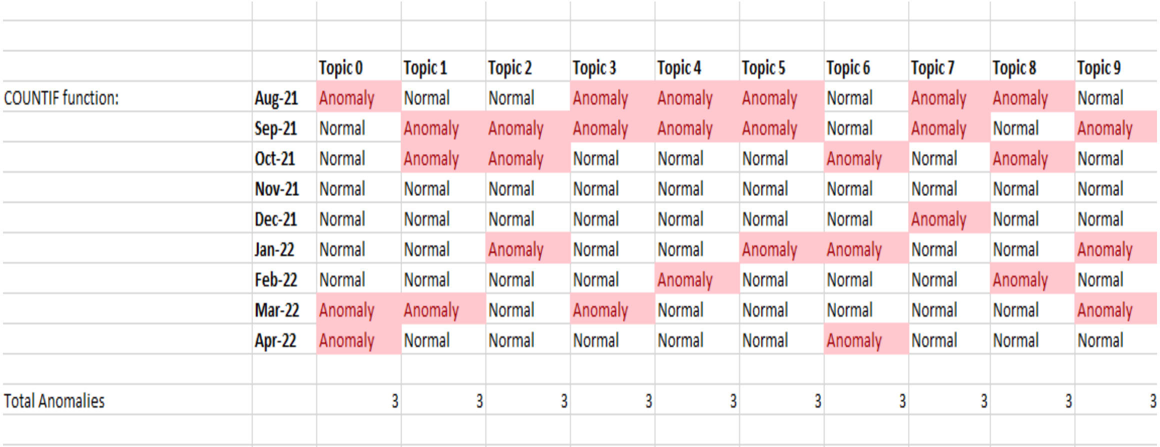
1. Count-IF Method

=IF(COUNTIF($H$5:$H$13,"<"&H5)<3,"Anomaly","Normal")

This formula is using the COUNTIF function to count the number of values in the range H5:H13 that are less than the value in H5.

If the count is less than 3, then the formula returns "Anomaly". Otherwise, it returns "Normal".

In other words, if there are less than 3 values in the range that are smaller than the value in H5, it is considered an anomaly and is flagged as such.



		Topic 0	Topic 1	Topic 2	Topic 3	Topic 4	Topic 5	Topic 6	Topic 7	Topic 8	Topic 9
COUNTIF function:	Aug-21	Anomaly	Normal	Normal	Anomaly	Anomaly	Anomaly	Normal	Anomaly	Anomaly	Normal
	Sep-21	Normal	Anomaly	Anomaly	Anomaly	Anomaly	Anomaly	Normal	Anomaly	Normal	Anomaly
	Oct-21	Normal	Anomaly	Anomaly	Normal	Normal	Normal	Anomaly	Normal	Anomaly	Normal
	Nov-21	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal
	Dec-21	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Anomaly	Normal	Normal
	Jan-22	Normal	Normal	Anomaly	Normal	Normal	Anomaly	Anomaly	Normal	Normal	Anomaly
	Feb-22	Normal	Normal	Normal	Normal	Anomaly	Normal	Normal	Normal	Anomaly	Normal
	Mar-22	Anomaly	Anomaly	Normal	Anomaly	Normal	Normal	Normal	Normal	Normal	Anomaly
	Apr-22	Anomaly	Normal	Normal	Normal	Normal	Normal	Anomaly	Normal	Normal	Normal
Total Anomalies		3	3	3	3	3	3	3	3	3	3

Figure xxxv The formula for the anomaly detection is used such as to determine atleast 3 anomalies in a particular topic

2. Z-Score Method

=IF(ABS((H5-AVERAGE($H$5:$H$13))/STDEV($H$5:$H$13))>3,"Anomaly","Normal")

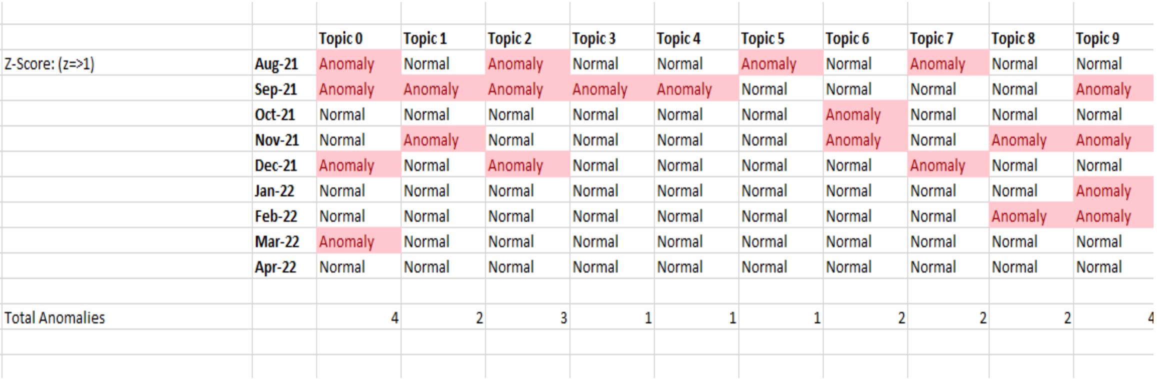
This formula in Microsoft Excel calculates whether the value in cell H5 is an anomaly based on the values in the range H5:H13.

It first calculates the difference between the value in cell H5 and the average of the values in the range H5:H13.

Then it divides that difference by the standard deviation of the values in the range H5:H13.

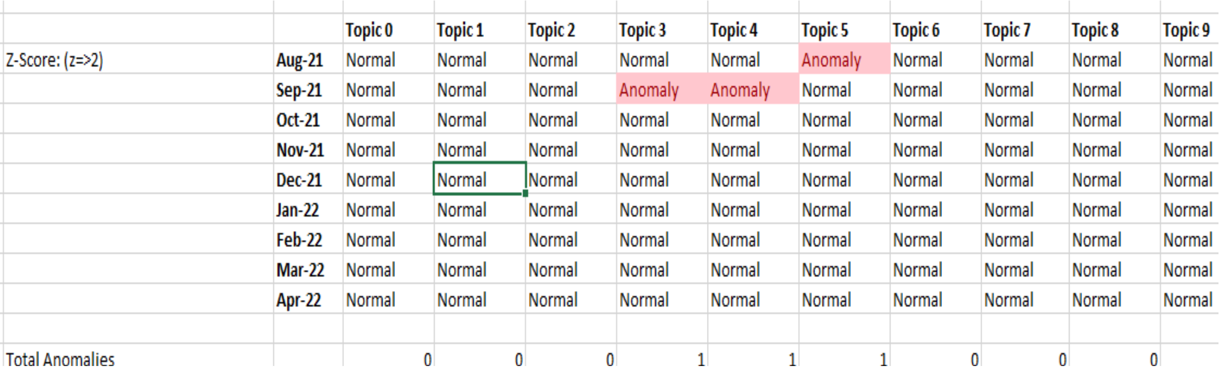
Finally, it checks if the result of that calculation is greater than 3. If it is, the formula returns "Anomaly", otherwise it returns "Normal".

This calculation assumes that the data in the range H5:H13 follows a normal distribution and considers values that are 3 or more standard deviations from the mean to be anomalies.



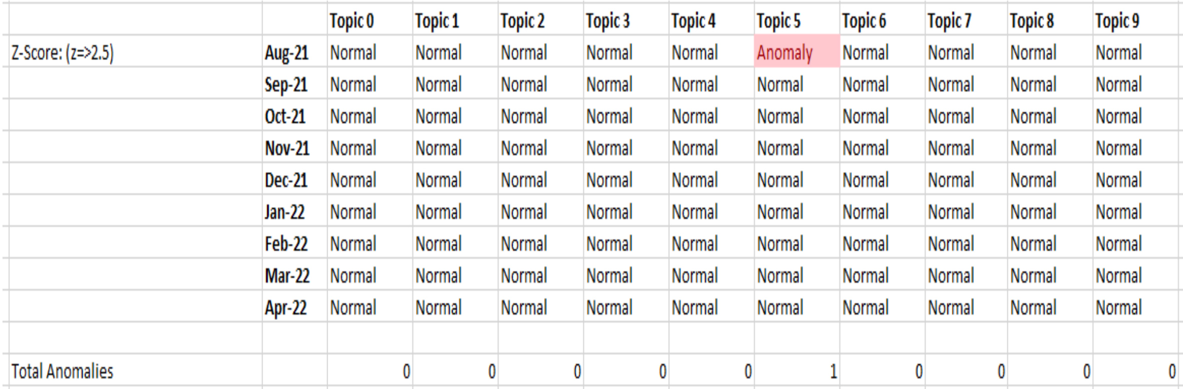
		Topic 0	Topic 1	Topic 2	Topic 3	Topic 4	Topic 5	Topic 6	Topic 7	Topic 8	Topic 9
Z-Score: (z=>1)	Aug-21	Anomaly	Normal	Anomaly	Normal	Normal	Anomaly	Normal	Anomaly	Normal	Normal
	Sep-21	Anomaly	Anomaly	Anomaly	Anomaly	Anomaly	Normal	Normal	Normal	Normal	Anomaly
	Oct-21	Normal	Normal	Normal	Normal	Normal	Normal	Anomaly	Normal	Normal	Normal
	Nov-21	Normal	Anomaly	Normal	Normal	Normal	Normal	Anomaly	Normal	Anomaly	Anomaly
	Dec-21	Anomaly	Normal	Anomaly	Normal	Normal	Normal	Normal	Anomaly	Normal	Normal
	Jan-22	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Anomaly
	Feb-22	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Anomaly	Anomaly
	Mar-22	Anomaly	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal
	Apr-22	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal
Total Anomalies		4	2	3	1	1	1	2	2	2	4

Figure xxxvi The Anomaly detected using Z score method when the Z score is kept at 1



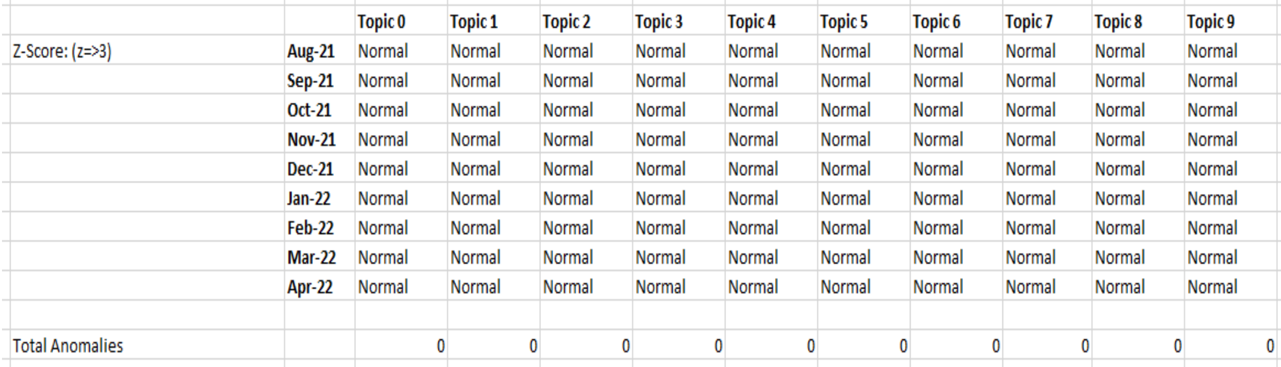
		Topic 0	Topic 1	Topic 2	Topic 3	Topic 4	Topic 5	Topic 6	Topic 7	Topic 8	Topic 9
Z-Score: (z=>2)	Aug-21	Normal	Normal	Normal	Normal	Normal	Anomaly	Normal	Normal	Normal	Normal
	Sep-21	Normal	Normal	Normal	Anomaly	Anomaly	Normal	Normal	Normal	Normal	Normal
	Oct-21	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal
	Nov-21	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal
	Dec-21	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal
	Jan-22	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal
	Feb-22	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal
	Mar-22	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal
	Apr-22	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal
Total Anomalies		0	0	0	1	1	1	0	0	0	

Figure xxxvii The Anomalies detected using Z score formula when Z is kept at 2. Notice the number of anomalies has been reduced to 3 in the entire dataset



		Topic 0	Topic 1	Topic 2	Topic 3	Topic 4	Topic 5	Topic 6	Topic 7	Topic 8	Topic 9
Z-Score: (z=>2.5)	Aug-21	Normal	Normal	Normal	Normal	Normal	Anomaly	Normal	Normal	Normal	Normal
	Sep-21	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal
	Oct-21	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal
	Nov-21	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal
	Dec-21	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal
	Jan-22	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal
	Feb-22	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal
	Mar-22	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal
	Apr-22	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal
Total Anomalies		0	0	0	0	0	1	0	0	0	0

Figure xxxviii The anomaly detected when the Z score is kept at 2.5. This represents only one anomaly is found when the parameters are relaxed.



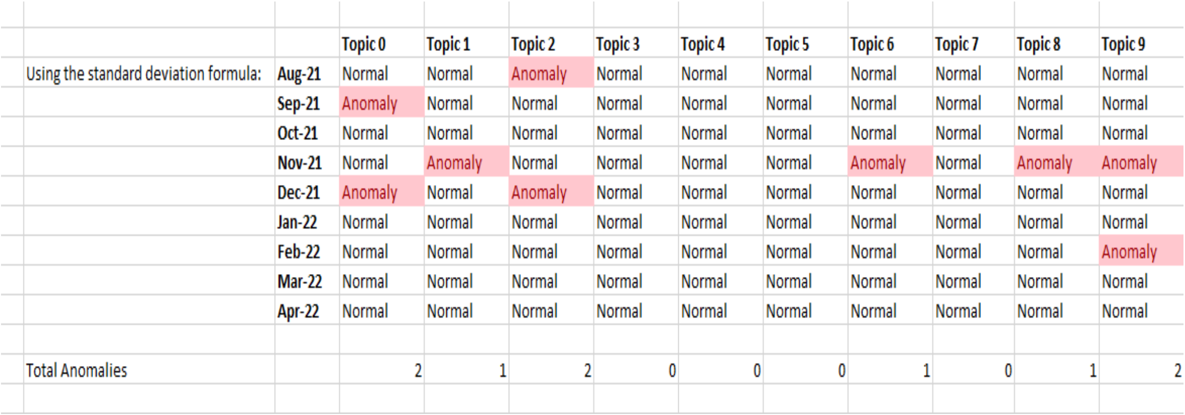
		Topic 0	Topic 1	Topic 2	Topic 3	Topic 4	Topic 5	Topic 6	Topic 7	Topic 8	Topic 9
Z-Score: (z=>3)	Aug-21	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal
	Sep-21	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal
	Oct-21	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal
	Nov-21	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal
	Dec-21	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal
	Jan-22	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal
	Feb-22	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal
	Mar-22	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal
	Apr-22	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal
Total Anomalies		0	0	0	0	0	0	0	0	0	0

Figure xxxix No anomaly has been found when the Z score is kept at 3. As all of the data lies inside 99.7% of the tolerance range

3. Standard Deviation Method

=IF(H5>(AVERAGE($H$5:$H$13)+STDEV($H$5:$H$13)),"Anomaly","Normal")

This formula uses the concept of statistical analysis to determine if a data point is an anomaly or not. The formula first calculates the average of the values in the range H5:H13, and then calculates the standard deviation of those values. Then, the value of H5 is compared with the average plus the standard deviation. If H5 is greater than this calculated value, the formula returns "Anomaly", indicating that the data point may be an outlier. Otherwise, the formula returns "Normal", indicating that the data point falls within the expected range of values based on the average and standard deviation.



		Topic 0	Topic 1	Topic 2	Topic 3	Topic 4	Topic 5	Topic 6	Topic 7	Topic 8	Topic 9
Using the standard deviation formula:	**Aug-21**	Normal	Normal	Anomaly	Normal	Normal	Normal	Normal	Normal	Normal	Normal
	Sep-21	Anomaly	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal
	Oct-21	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal
	Nov-21	Normal	Anomaly	Normal	Normal	Normal	Normal	Anomaly	Normal	Anomaly	Anomaly
	Dec-21	Anomaly	Normal	Anomaly	Normal	Normal	Normal	Normal	Normal	Normal	Normal
	Jan-22	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal
	Feb-22	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Anomaly
	Mar-22	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal
	Apr-22	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal	Normal
Total Anomalies		2	1	2	0	0	0	1	0	1	2

Figure xl The Anomaly detection using Standard Deviation formula

Analysis of the results

Visualization through WordCloud

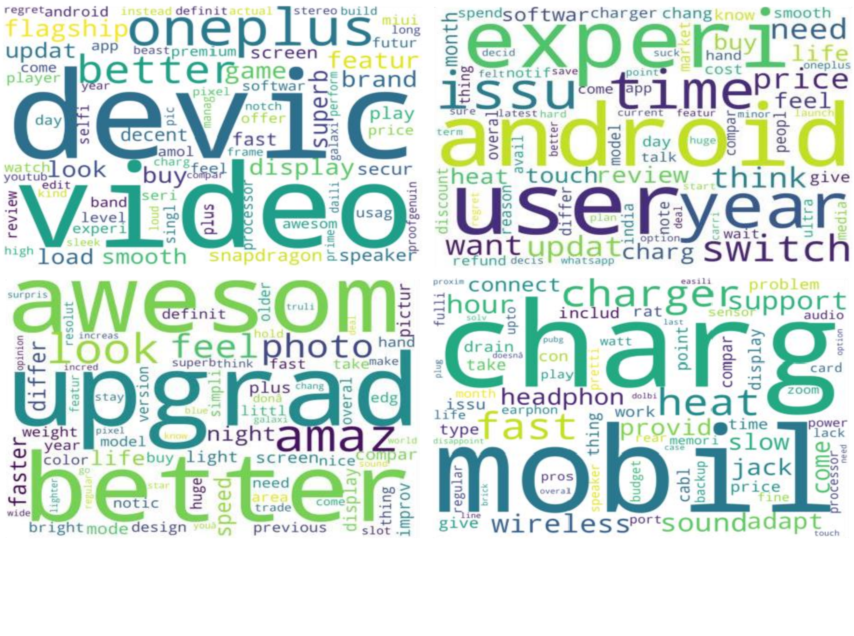
A word cloud is a popular way to visualize the most frequently occurring words in a text corpus. It is often used to represent the topics in a document or a set of documents by showing the size of the words proportional to their frequency of occurrence. The most frequent words are displayed in larger fonts and less frequent words in smaller fonts.

To create a word cloud, the text data needs to be preprocessed to remove stop words (common words such as "the" and "and" that don't carry much meaning), and the frequency of each word needs to be calculated. Then, the words are placed in a cloud-like shape, with the most frequent words appearing larger and the less frequent words appearing smaller. The size of the words can also be represented by color intensity or other visual cues.

Word clouds can be easily generated using various libraries and tools such as word cloud in Python or the word cloud generator on websites like wordclouds.com. These tools allow you to customize the appearance of the word cloud, including the font, color scheme, and shape.

In topic modeling, a word cloud can provide a quick and intuitive way to see what topics are covered in a large corpus of text data. By examining the words that appear most frequently, you can gain insight into the main themes and topics in the text data.

Word Cloud LDA Model



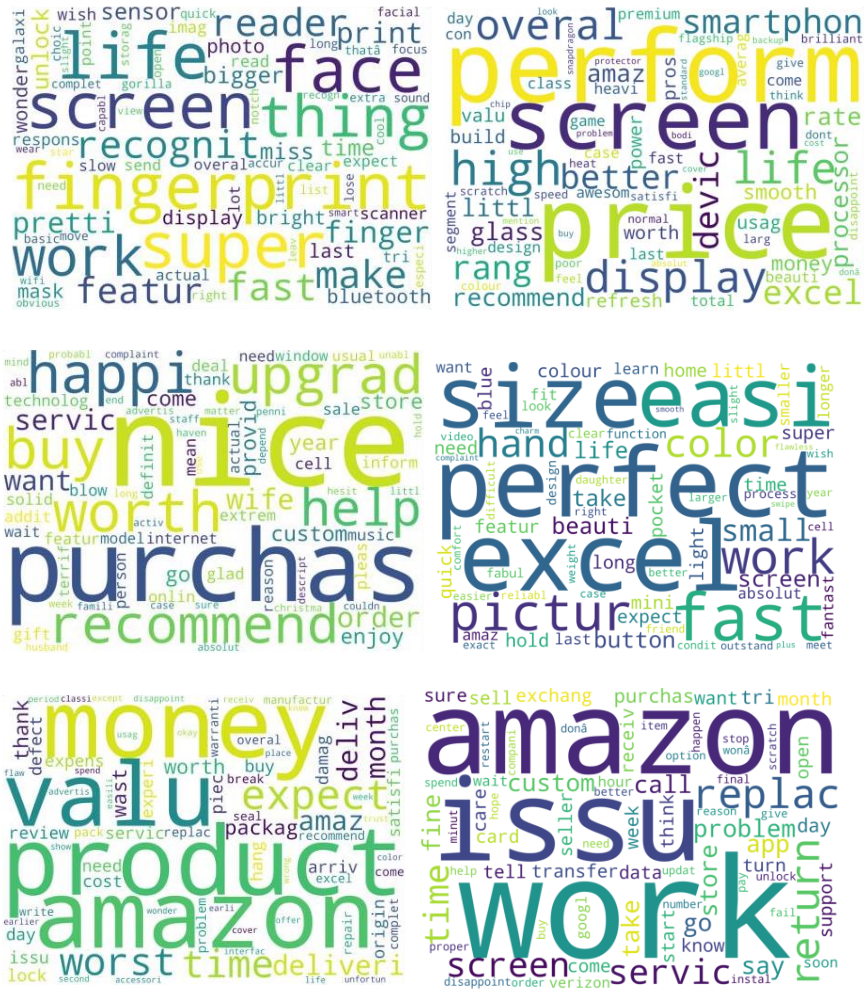
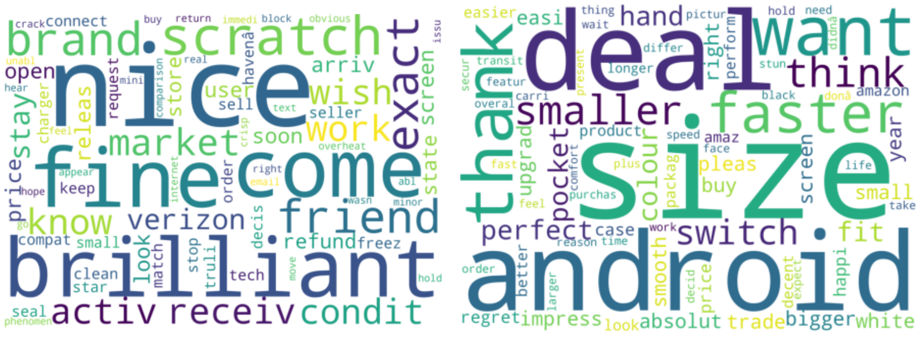


Figure xli Word cloud representing the LDA model distribution of words using in particular topic with size representing the probability of usage

Word Cloud LDA-TFIDF Model



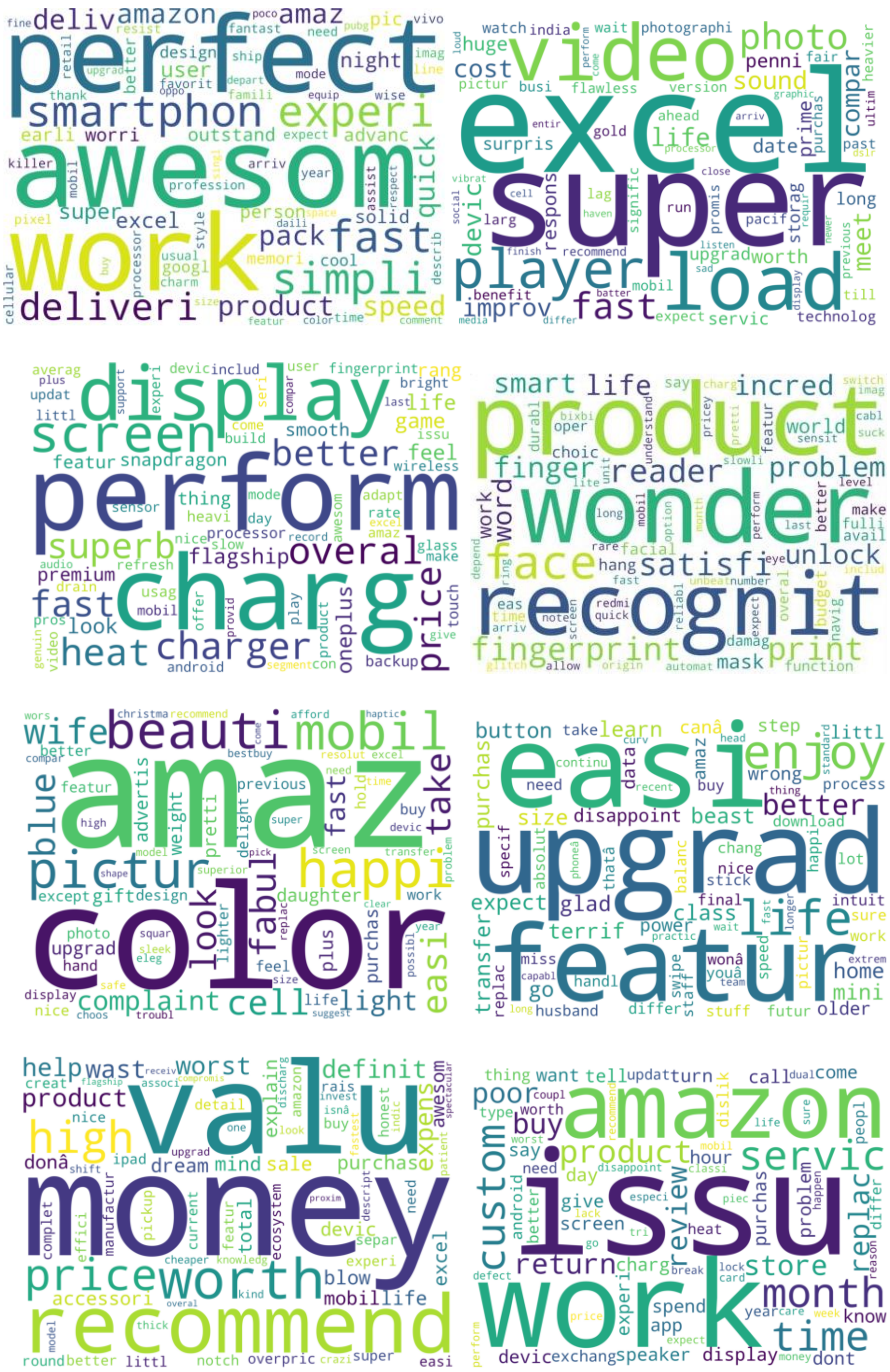


Figure xlii representation of LDA TF-IDF word cloud

The contrast between Standard LDA and LDA TF-IDF word cloud lies around the fact that prior considers the frequency of the word or number of times it appears in a document, this does not provide the unique topics or words that are important to distinguish the topics.

The main difference between the results of LDA and LDA-TFIDF topics is the type of words that dominate each topic. LDA-TFIDF tends to focus on more subjective or evaluative words, whereas LDA focuses more on specific features or characteristics of the product.

For example, in LDA topic 0, the words "device", "video", "feature", "high", and "better" suggest that the topic is about the technical specifications of the product. In contrast, in LDA-TFIDF topic 0, the words "nice", "brilliant", and "fine" suggest that the topic is about the overall subjective experience of using the product.

Similarly, in LDA topic 8, the words "product", "money", "value", and "delivery" suggest that the topic is about the business aspects of the product, such as pricing and delivery. In contrast, in LDA-TFIDF topic 2, the words "awesome", "perfect", "work", and "smartphone" suggest that the topic is about the overall experience of using the product, including its speed and simplicity.

Overall, LDA-TFIDF may be more useful for understanding the overall sentiment or evaluation of a product, while LDA may be more useful for understanding specific features or aspects of a product.

Analysis Through pyLDAvis

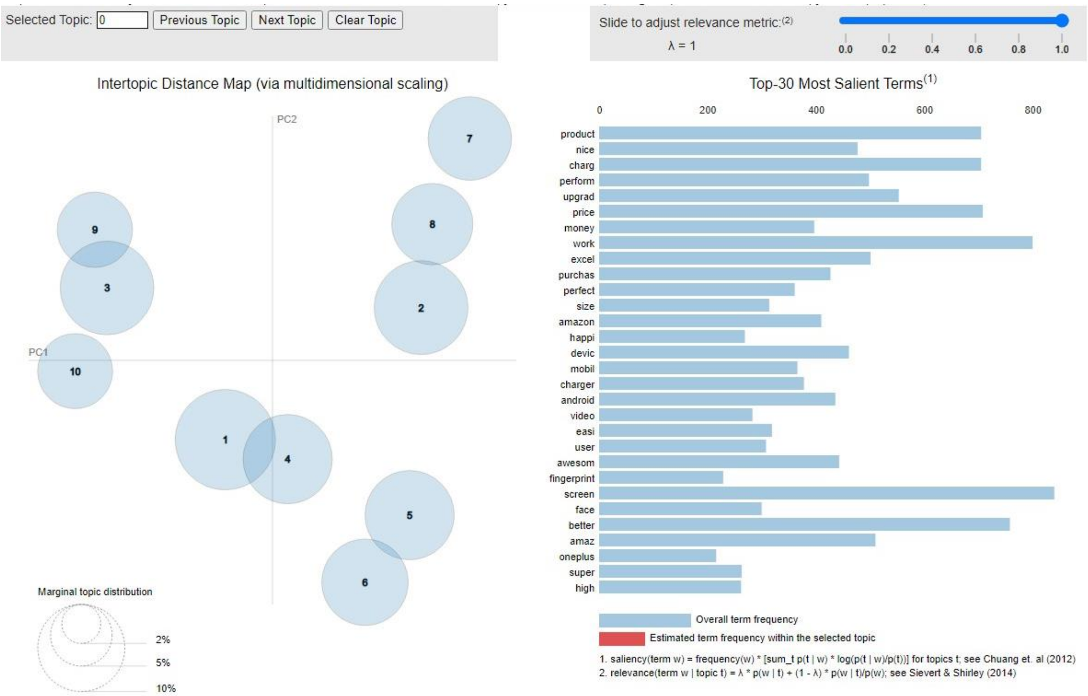
PyLDAvis is a Python library that provides interactive visualizations for topic models created using Latent Dirichlet Allocation (LDA). It helps to understand the relationships between topics, the words that make up each topic, and the distribution of topics in a corpus of text data.

The main visualization in pyLDAvis is an interactive scatterplot, which displays the topics on one axis and the terms on the other axis. The size of the points in the plot represents the relative frequency of the terms, and the color of the points represents the contribution of the terms to the topics.

In addition to the scatterplot, pyLDAvis also provides several other visualizations to help interpret the results of topic modeling. For example, it provides a bar chart that shows the frequency of each topic in the corpus, and a bar chart that shows the distribution of the most representative terms for each topic.

To use pyLDAvis, you first need to fit an LDA model to your text data using tools such as the gensim library in Python. Then, you can pass the LDA model to the pyLDAvis library, which will generate the visualizations. The visualizations can be displayed in a web browser and interacted with, allowing you to explore the relationships between topics and terms in more detail.

Overall, pyLDAvis provides a useful and user-friendly way to understand the results of topic modeling and to gain insights into the structure of large text datasets.



1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) = λ * p(w | t) + (1 - λ) * p(w | t)/p(w); see Sievert & Shirley (2014)

Figure xliii PyLDAvis General Topic Distribution, representing overall most used terms along with their frequency in the relative topic

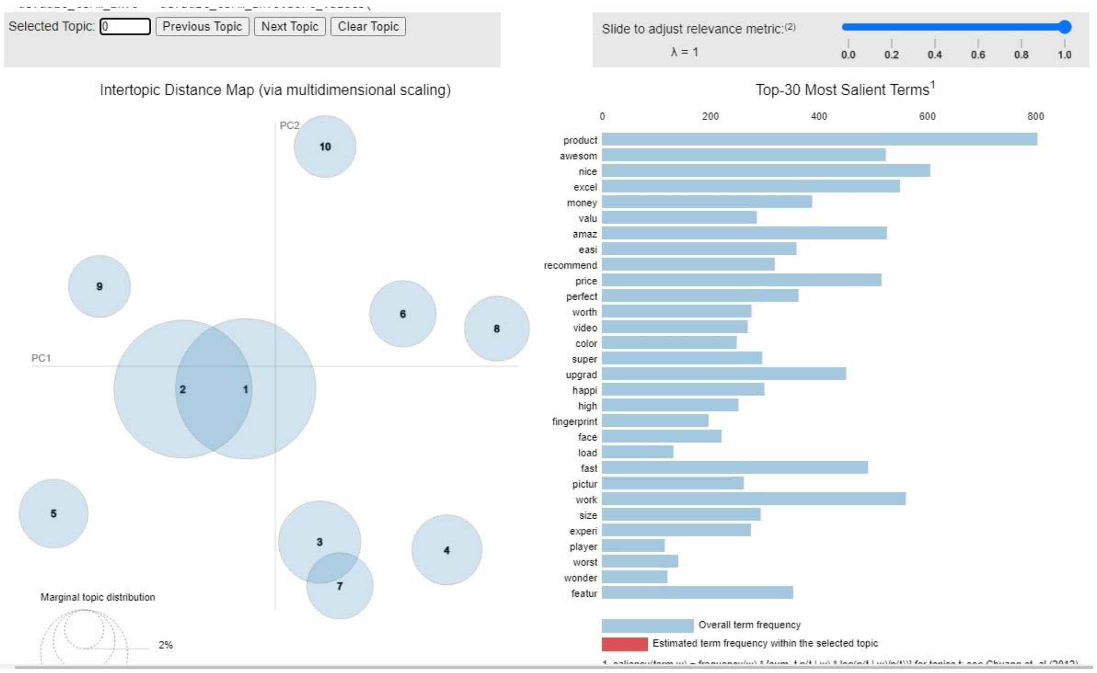


Figure xliv pyLDA General Topic Distribution according to Term Frequency Inverse Document Frequency

Differentiation in LDA and LDA TF-IDF Approach

The difference between pyLDAvis visualization of LDA and LDA-TFIDF lies in the way the text data is transformed and prepared for modeling.

LDA (Latent Dirichlet Allocation) is a generative probabilistic model that assumes that each document in a corpus is generated by a mixture of topics, where a topic is represented as a distribution over words. In LDA, the input to the model is a document-term matrix, where each row represents a document and each column represents a word, and the entries in the matrix represent the word count or frequency of the word in the document.

LDA-TFIDF (Latent Dirichlet Allocation with Term Frequency-Inverse Document Frequency) is an extension of LDA that takes into account the importance of each word in the corpus. In TFIDF, each word is weighted by its importance, which is measured by the frequency of the word in the document compared to its frequency in the entire corpus. This weighting helps to down- weight the influence of common words that appear frequently in most documents, and up- weight the influence of words that are specific to a particular document.

In pyLDAvis, you can visualize the results of both LDA and LDA-TFIDF using the same interface. The only difference is in the way the text data is transformed and prepared for modeling. If you use LDA, you would input the document-term matrix directly to the LDA model. If you use LDA-TFIDF, you would input the TFIDF-weighted document-term matrix to the LDA model. The pyLDAvis library then generates the same interactive visualizations for both models, allowing you to compare the results and gain insights into the structure of the text data.

It can be observed from the visualization data that the intertopic distance is greater in LDA-TFIDF model as compared to LDA because the prior focus on assigning weights to the words in the topic. Only Topic 1 and 2 are overlapping while the remaining topics can be seen far and wide from each other due to the more distinct and unique nature. It can also be observed that in the LDA model the prevalence of the topics are almost the same whereas in TF-IDF model topic 1 and 2 are in greater proportion as compared to the rest of the topics. Overall, the choice between PyLDAvis LDA and PyLDAvis LDA TFIDF depends on the specific needs and goals of the analysis. If the goal is to explore the general topics present in the corpus, then BoW may be sufficient. However, if the goal is to identify more nuanced topics that are specific to individual documents, then TFIDF may be more appropriate.

Analysis of the Results

The topic modeling results show the distribution of topics in 9 months, from Aug-21 to Apr-22. Each column represents a topic and each row represents a month. The values in the table are the probabilities of each topic being present in the corresponding month. The highest value in each row represents the most dominant topic in that month. The topics seem to change in dominance over the months, with no single topic consistently being the most dominant across all the months.

Anomalies in this data could include:

- Data points that are significantly higher or lower than the surrounding values.
- Abrupt changes in the trend of the data, for example if the values in one row are significantly different from the values in the previous row.
- Unexpected patterns or relationships in the data. It is important to note that an anomaly in the data does not necessarily indicate a problem, but it is something that should be further investigated to determine its significance and potential cause.

All three formulas aim to identify whether a data point is an anomaly or not based on a comparison with other data points in the same dataset.

Each method has its own assumptions and limitations, and the choice of which method to use depends on the characteristics of the dataset and the goals of the analysis.

The Count-IF method looks for the number of values in a range that are smaller than the current value and flags it as an anomaly if the count is less than 3. The formula by its nature bound to express the required number of anomalies i.e in the given case it will represent at least 3 anomalous months for the each respective topic. This method may not work well when the distribution of data is not symmetrical or when there are outliers in the data.

The Z-Score method calculates the difference between the current value and the average of the dataset, and then divides it by the standard deviation. It flags the current value as an anomaly if the result is greater than the provided one. This method assumes that the data follows a normal distribution, and may not be suitable for datasets with a non-normal distribution. The difference between the two formulas lies in the threshold used to determine whether a data point is considered an anomaly or not.

The first formula uses a threshold of 3 standard deviations from the mean as the cutoff point for an anomaly. This means that any data point that falls more than 3 standard deviations away from the mean is classified as an anomaly. This threshold is considered to be quite strict and may result in fewer anomalies being identified.

The second and third formula uses a threshold of 2.5, 2 and 1 standard deviations from the mean as the cutoff point for an anomaly. This means that any data point that falls more than provided standard deviations away from the mean is classified as an anomaly. These thresholds are considered to be less strict than the first one and may result in more anomalies being identified. It is also evident from the given results that the number of anomalies detected increased as the threshold decreases i.e threshold of 1 results with most number of anomalies detected.

However, the choice of threshold depends on the context of the data and the desired level of sensitivity in detecting anomalies. A more strict threshold may result in a lower false positive rate but may also result in missing some true anomalies, while a less strict threshold may increase the false positive rate but may also capture more true anomalies.

The Standard Deviation method calculates the average and standard deviation of the dataset, and flags the current value as an anomaly if it is greater than the average plus the standard deviation. This method assumes that the data follows a normal distribution and that the data points are independent of each other. The standard deviation is a measure of how spread out the values in a dataset are from the average (mean) value. Adding the standard deviation to the average value creates a range of values that is considered "normal" or expected for the dataset. In the formula provided, if the value in a given cell is greater than the upper end of this range (i.e., the average plus one standard deviation), then it is considered to be an "anomaly" or an outlier. This approach is often used in statistical process control and quality control to identify data points that fall outside of a normal range and may require further investigation. By using the average and standard deviation as reference points, the formula can quickly and easily identify potential outliers without requiring a detailed analysis of the entire dataset. It may not work well when the dataset contains dependent or correlated data points.

Future Work

Topic modeling and anomaly detection are two powerful techniques that can be used to improve product and service management. Topic modeling can help identify the topics that customers are discussing, while anomaly detection can help detect unusual patterns that may indicate issues with the product or service. By combining these techniques, businesses can gain insights into the strengths and weaknesses of their products and services and take action to improve them.

One way to use topic modeling and anomaly detection for product and service management is to analyze customer feedback. Customer feedback can be collected from various sources, such as online reviews, social media, customer service calls, and surveys. Once collected, the feedback can be preprocessed and fed into a topic modeling algorithm to identify the topics that customers are discussing. These topics can then be analyzed to identify areas where the product or service is performing well and areas where improvements can be made.

Anomaly detection can be used to identify unusual patterns in the feedback data that may indicate issues with the product or service. For example, if a sudden increase in negative reviews is detected for a particular product, it may indicate a quality issue that needs to be addressed. Anomaly detection can also be used to identify unusual patterns in customer behavior, such as a sudden decrease in usage or a change in purchase patterns.

Once issues are identified, businesses can take action to address them. For example, if a quality issue is detected, the business can investigate the issue and take steps to improve the quality of the product. If a change in customer behavior is detected, the business can investigate the cause and take steps to address the underlying issue.

1. Customer Feedback Analysis: Companies can use topic modeling to analyze customer feedback from various sources such as social media, surveys, and online reviews. This can help companies identify the most common issues or concerns that customers have with their products or services. For example, a hotel chain could use topic modeling to analyze online reviews and identify common topics like cleanliness, customer service, or amenities. This information can help the company prioritize improvements to its services and address common pain points for customers.
2. Quality Control: Anomaly detection can be used to identify issues with product quality by analyzing data from various sources, such as manufacturing sensors, customer complaints, or quality control tests. For example, a car manufacturer could use anomaly detection to identify defective parts or components that are causing quality issues in its products. This information can help the company take corrective action to improve product quality and reduce defects.
3. Market Research: Topic modeling can be used to analyze market research data to identify trends and consumer preferences. For example, a consumer goods

company could use topic modeling to analyze survey data to identify which product features are most important to consumers. This information can help the company develop new products or improve existing ones to better meet customer needs.

4. Fraud Detection: Anomaly detection can be used to identify fraudulent activity in financial transactions, such as credit card fraud or insurance fraud. For example, an insurance company could use anomaly detection to identify unusual patterns in insurance claims that may indicate fraud. This information can help the company take action to prevent fraud and protect its customers.
5. Supply Chain Optimization: Topic modeling can be used to analyze supply chain data to identify bottlenecks or inefficiencies. For example, a logistics company could use topic modeling to analyze data from shipping manifests to identify which routes or carriers are causing delays or quality issues. This information can help the company optimize its supply chain and improve delivery times and quality.

Overall, topic modeling and anomaly detection can be powerful tools for product and service management. By analyzing customer feedback and detecting unusual patterns, businesses can gain insights into the strengths and weaknesses of their products and services and take action to improve them. Topic modeling and anomaly detection are versatile techniques that can be used in a variety of ways to improve product and service management. By analyzing large datasets and identifying patterns and anomalies, companies can gain valuable insights that can help them make better decisions and improve their operations.

The Complete Programming Code

```
import pandas as pd
data = pd.read_csv('/content/DATABASE SMARTPHONE
train.csv',encoding='latin-1', error_bad_lines=False);
print(data.head())

# We only need the Headlines text column from the data
data_text = data[['TEXT']];
data_text['index'] = data_text.index

documents = data_text

!pip install gensim
import gensim
from gensim.utils import simple_preprocess
from gensim.parsing.preprocessing import STOPWORDS
from nltk.stem import WordNetLemmatizer, SnowballStemmer
from nltk.stem.porter import *
import numpy as np
np.random.seed(400)

import nltk
nltk.download('wordnet')
```

The code starts by importing the pandas library as pd, which is a data analysis and manipulation tool. Then, it reads a CSV file called "DATABASE SMARTPHONE train.csv" using the read_csv() function of pandas. The file path is specified as an argument to the function. The encoding parameter is set to 'latin-1' and error_bad_lines is set to False, which means that lines with errors are skipped instead of throwing an error. The data is then printed using the head() function, which displays the first few rows of the dataset.

Next, the code selects only the "TEXT" column from the data and creates a new DataFrame called data_text. A new column called "index" is added to data_text which contains the index of each row.

The code then installs gensim, a popular Python library for topic modeling and text analysis. It also imports several modules from gensim, nltk, and numpy libraries.

Finally, the code downloads the "wordnet" corpus from nltk, which is a lexical database for the English language. The wordnet corpus is used for lemmatization, which is the process of reducing words to their base or dictionary form.

```
nltk.download('omw-1.4')
print(WordNetLemmatizer().lemmatize('went', pos = 'v')) # past
tense to present tense

stemmer = SnowballStemmer("english")
original_words = ['caresses', 'flies', 'dies', 'mules',
'denied','died', 'agreed', 'owned',
           'humbled', 'sized','meeting', 'stating', 'siezing',
'itemization','sensational',
           'traditional', 'reference', 'colonizer','plotted']
singles = [stemmer.stem(plural) for plural in original_words]

pd.DataFrame(data={'original word':original_words,
'stemmed':singles })


def lemmatize_stemming(text):
    return stemmer.stem(WordNetLemmatizer().lemmatize(text,
pos='v'))


def preprocess(text):
    result=[]
    for token in gensim.utils.simple_preprocess(text) :
        if token not in gensim.parsing.preprocessing.STOPWORDS
and len(token) > 3:
            # TODO: Apply lemmatize_stemming() on the token,
then add to the results list
            result.append(lemmatize_stemming(token))

    return result

'''
Preview a document after preprocessing
'''
document_num = 551
doc_sample = documents[documents['index'] ==
document_num].values[0][0]

doc_sample
```

```
print("Original document: ")
words = []
for word in doc_sample.split(' '):
    words.append(word)
print(words)
print("\n\nTokenized and lemmatized document: ")
print(preprocess(doc_sample))

doc_sample = 'the doctor came to the village to visit some
patients yesterday.'
print("Original document: ")
words = []
for word in doc_sample.split(' '):
    words.append(word)
print(words)
print("\n\nTokenized and lemmatized document: ")
print(preprocess(doc_sample))

processed_docs = documents['TEXT'].astype(str).map(lambda x:
preprocess(x))

'''
Preview 'processed_docs'
'''
processed_docs[:10]
```

The code performs text preprocessing on a corpus of documents using NLTK and Gensim libraries in Python. It first downloads the Open Multilingual WordNet (OMW) and uses NLTK's WordNetLemmatizer class to lemmatize words. It also applies stemming to a list of original words using SnowballStemmer class and displays the original and stemmed versions using a pandas DataFrame.

The code defines two functions, "lemmatize_stemming" and "preprocess," to perform text preprocessing on a given text. The "preprocess" function tokenizes the text, removes stop words, filters out short words, applies stemming and lemmatization using "lemmatize_stemming" function and returns the preprocessed version.

The code uses the "preprocess" function to preprocess each document in a pandas DataFrame called "documents" and stores the preprocessed versions in a new pandas Series called "processed_docs". The code also prints the preprocessed versions of two sample documents.

```
dictionary = gensim.corpora.Dictionary(processed_docs)

count = 0
for k, v in dictionary.iteritems():
    print(k, v)
    count += 1
    if count > 10:
        break

print(dictionary)

from collections import Counter
count = Counter()
for doc in processed_docs:
    for word in doc:
        count[word]+=1
print(count)

print(count['communiti'])

len(count)


len(dictionary)
'''
bow_corpus = [dictionary.doc2bow(doc) for doc in
processed_docs]

bow_corpus[document_num]

bow_doc_4310 = bow_corpus[document_num]

for i in range(len(bow_doc_4310)):
    print("Word {} (\"{}\") appears {}
time.".format(bow_doc_4310[i][0],
                                                     dictionary
[bow_doc_4310[i][0]],
                                                     bow_doc_43
10[i][1]))
```

The code performs various text preprocessing tasks such as creating a dictionary of unique words with their corresponding IDs, counting the frequency of each word in the corpus, and creating a bag-of-words (BOW) representation of the preprocessed documents. The BOW representation is a list of tuples where each tuple contains the ID of a word and its frequency in the document. Finally, the code selects one of the documents, retrieves its BOW representation, and prints out the word and frequency for each element in the BOW representation.

```
from gensim import corpora, models

tfidf = models.TfidfModel(bow_corpus)
corpus_tfidf = tfidf[bow_corpus]

from pprint import pprint
for doc in corpus_tfidf:
    pprint(doc)
    break



lda_model = gensim.models.LdaMulticore(bow_corpus, num_topics =
10, id2word = dictionary, passes = 150)

for idx, topic in lda_model.print_topics(-1):
    print("Topic: {} \nWords: {}".format(idx, topic))
    print("\n")


lda_model_tfidf = gensim.models.LdaMulticore(corpus_tfidf,
num_topics = 10, id2word = dictionary, passes = 150)

for idx, topic in lda_model_tfidf.print_topics(-1):
    print("Topic: {} \nWord: {}".format(idx, topic))
    print("\n")
```

The code first creates a TF-IDF model from the bag-of-words representation of the corpus using models.TfidfModel. It then applies the TF-IDF transformation to the corpus using tfidf[bow_corpus] and prints the first document in the transformed corpus.

Next, the code creates an LDA model using gensim.models.LdaMulticore and fits it to the bag-of-words corpus. It prints the top words associated with each topic

in the model.
Finally, the code creates another LDA model using the TF-IDF-transformed corpus and prints the top words associated with each topic in this model.

```
from google.colab import drive
drive.mount('/content/drive')


from tkinter.constants import X
# Data preprocessing step for the unseen document
data2 = pd.read_csv('/content/Database smartphone
test.csv',encoding='latin-1', error_bad_lines=False)
data_text2 = data[['TEXT']];
data_text2['index'] = data_text2.index

documents2 = data_text2
processed_docs2 = documents2['TEXT'].astype(str).map(lambda x:
preprocess(x))
bow_corpus2 = [dictionary.doc2bow(doc2) for doc2 in
processed_docs2]
x=-1
threshold = 0.1
from google.colab import drive
drive.mount('/content/drive')
import os
os.chdir("/content/drive/")
with open('/content/thesis4.csv','w') as out:

  while x<4370:
    document_no=x
    x=x+1

    for index, score in
sorted(lda_model[bow_corpus2[document_no]], key=lambda tup: -
1*tup[1]):
      if score < threshold: break
      print("Score: {}\n Topic: {}".format(score,
lda_model.print_topic(index, 5)))
      print()
      out.write("\n{} \t{} \t{}".format(x,score,
lda_model.print_topic(index, 5)))



from wordcloud import WordCloud, STOPWORDS

def word_cloud(topic, model):
    plt.figure(figsize = (8,6))
    topic_words = [model.print_topic(topic, 75)]
    cloud = WordCloud(stopwords = STOPWORDS, background_color =
'white',
                      width=2500, height=1800).generate("
".join(topic_words))

    print('\nWordcloud for topic:', topic, '\n')
    plt.imshow(cloud)
    plt.axis('off')
    plt.show()

for topic in range(10):
    #plt.figure(figsize=(10,15))
    word_cloud(topic, lda_model)

for topic in range(10):
    plt.figure(figsize=(10,15))
    word_cloud(topic, lda_model_tfidf)

!pip install pyLDAvis
!pip install pyLDAvis.gensim_models
import pyLDAvis
import pyLDAvis.gensim_models
pyLDAvis.enable_notebook(local=True)
vis =
pyLDAvis.gensim_models.prepare(lda_model,bow_corpus,dictionary)
vis

import pyLDAvis
import pyLDAvis.gensim_models
pyLDAvis.enable_notebook(local=True)
vis =
pyLDAvis.gensim_models.prepare(lda_model_tfidf,bow_corpus,dicti
onary)
vis
```

The code imports necessary libraries for data manipulation, GUI programming, and topic modeling. It reads in a csv file and preprocesses the text, creates a bag-of-words corpus, and applies LDA topic modeling. It filters topics below a threshold and saves the top relevant topics to a csv file. The code also generates word clouds for the top 10 topics using the LDA model and the LDA model with tf-idf weighting. Finally, it uses pyLDAvis to create interactive visualizations of the LDA models.

Conclusion

The thesis proposes a novel approach to anomaly detection in large datasets using topic modeling-based techniques. Specifically, the method utilizes the Latent Dirichlet Allocation (LDA) algorithm to uncover the underlying topics in a dataset and then identifies anomalies based on the testing score of a particular document from the general topic distribution. The study demonstrates the effectiveness of this approach in identifying anomalies in a dataset of online customer reviews, suggesting its potential as a useful tool for detecting abnormalities in big datasets. The thesis emphasizes the importance of detecting anomalies in various fields, making the proposed approach a valuable contribution to anomaly detection. Furthermore, future research could focus on improving the accuracy and scalability of the method, as well as exploring its application in other fields. Overall, the study underscores the potential of topic modeling techniques for anomaly detection in large datasets and highlights the need for continued research in this area.

Application in the world of Digital VoC

it is important to understand the significance of the LDA and LDA-TFIDF models in uncovering the underlying themes and topics in the reviews. LDA is a generative probabilistic model that identifies the topics in a corpus of text by analyzing the distribution of words in each document. LDA-TFIDF is a variation of LDA that takes into account the importance of each word in the document based on its frequency and the overall frequency of the word in the corpus.

Once the topics are identified using these models, it is possible to analyze the distribution of these topics in the reviews to detect any anomalies or unusual patterns. For instance, if a particular topic such as "customer service" is present in the majority of reviews but is missing from a few reviews, this could indicate an anomaly. This anomaly could be further investigated to understand the reason behind the absence of this topic in the reviews.

Similarly, if a topic such as "product quality" is present in only a few reviews but is not present in the majority of the reviews, this could also be an indication of an anomaly. This could indicate a potential problem with the product quality that is not being reported by customers, which would require further investigation by the business.

The results of the topic modeling analysis can then be used to identify these anomalies and understand why they are present. For instance, if the absence of the "customer service" topic in a few reviews is due to a glitch in the system that prevented customers from reaching customer service representatives, this could be addressed by the business to improve their customer service experience.

Overall, the approach of using topic modeling results from customer reviews is a powerful tool for businesses to gain insights into customer behavior and preferences. By leveraging natural language processing techniques such as LDA and LDA-TFIDF, businesses can analyze customer feedback at scale and make data-driven decisions to improve their products and services.

Limitations For The Approach

There are several limitations to using topic modeling for anomaly detection, including:

1. Data Quality: The accuracy and effectiveness of topic modeling depends on the quality of the data. If the data is noisy or has missing values, the results of the topic modeling may not be reliable. to ensure the accuracy and effectiveness of topic modeling, it is essential to have high-quality data that is free of noise, missing values, and biases. Data cleaning and preprocessing techniques, such as text normalization, removal of stop words, and stemming, can help improve the quality of the data. Additionally, data validation and verification procedures can help ensure that the data is accurate and reliable. By using high-quality data, topic modeling can generate meaningful insights that can inform decision-making in various domains, such as business, healthcare, and social sciences.
2. Computational Complexity: Topic modeling algorithms can be computationally intensive, especially for large datasets. This can limit the ability to scale the analysis to very large datasets in real-time. In addition to the size of the dataset, the complexity of the topic modeling algorithm can also affect computational requirements. Some algorithms, such as Latent Dirichlet Allocation (LDA), require more computational resources than others, such as Latent Semantic Analysis (LSA). As a result, the choice of algorithm can impact the speed and scalability of the analysis.
3. Topic Ambiguity: The interpretation of topics generated by topic modeling algorithms can be subjective and open to interpretation. This can lead to ambiguity in the results and difficulty in drawing accurate conclusions. This ambiguity can be due to several reasons, such as the choice of algorithm, the number of topics selected, and the specific parameters used in the algorithm. Furthermore, the quality of the output generated by topic modeling is dependent on the quality of the input data. If the data is noisy, ambiguous, or incomplete, the resulting topics may also be ambiguous or incomplete. In such cases, domain knowledge or human expert analysis may be necessary to make sense of the topic modeling results and interpret them accurately
4. Domain-Specific Knowledge: Topic modeling algorithms may not always generate meaningful topics for specific domains, especially in industries where the language and terminology used is specialized. While domain-specific knowledge is critical for interpreting topic modeling results accurately in specialized industries, its use may be limited by the availability and accessibility of knowledge, subjectivity, the dynamic nature of terminology and concepts, limited scope, and interdisciplinary nature. It is essential to consider these limitations and use domain-specific knowledge in conjunction with other approaches to interpret topic modeling results accurately.
5. Model Selection: Choosing the right topic modeling algorithm and parameters can be challenging, as different algorithms and parameters may produce different

results. model selection is a critical step in topic modeling that requires careful consideration and experimentation to select the best algorithm and parameters for a given dataset. The choice of algorithm and parameters should be based on factors such as algorithm complexity, model performance, domain-specific considerations, and user requirements. Finally, the model should be trained and evaluated using appropriate techniques to ensure its robustness and generalizability. This requires careful consideration and experimentation to select the best model for the given dataset.

6. Anomaly Definition: Anomaly detection is the process of identifying rare or unusual patterns in a dataset. In the context of topic modeling, anomaly detection involves identifying topics that are significantly different from the other topics in the corpus. However, the definition of what constitutes an anomaly is not always clear and can vary depending on the specific use case. This can result in difficulties in evaluating the performance of the topic modeling approach for anomaly detection.

The following are some factors that contribute to the challenge of defining anomalies in topic modeling:

- Subjectivity: The definition of what constitutes an anomaly may be subjective and vary depending on the individual or organization's perspective. For example, a topic that is considered an anomaly in one industry may be a regular occurrence in another industry.
- Data Quality: Anomalies may be the result of data quality issues such as errors or noise in the data. Therefore, defining what constitutes an anomaly requires a clear understanding of the underlying data and its quality.
- Contextual Factors: Anomalies may be influenced by contextual factors such as time, location, or user behavior. Therefore, the definition of what constitutes an anomaly must consider these contextual factors.
- Rare Occurrences: Anomalies are, by definition, rare occurrences in a dataset. Therefore, the definition of what constitutes an anomaly must take into account the frequency of the topic and its relative occurrence compared to other topics.
- Use Case: The definition of what constitutes an anomaly may vary depending on the specific use case. For example, in a financial fraud detection scenario, an anomaly may be defined as a topic that is significantly different from the normal behavior of the user.

Final Remarks

In summary, while topic modeling can be a useful tool for anomaly detection in text data, it is crucial to be aware of its limitations and to approach the analysis with caution and critical thinking. By doing so, we can ensure that the results of the analysis are accurate and reliable, and we can make informed decisions based on the insights obtained from the analysis. To ensure accurate results, it is essential to approach the analysis with caution and critical thinking. Topic modeling is a statistical technique that identifies latent topics in a corpus of text.

Anomalies are often characterized as topics that are significantly different from other topics in the corpus. However, there are various factors to consider when using topic modeling for anomaly detection, such as data quality, model selection, domain-specific knowledge, and the definition of what constitutes an anomaly.

Therefore, it is important to recognize that topic modeling is not a perfect tool, and it has certain limitations that must be considered. For example, if the data quality is poor, the results of the analysis may not be reliable. Similarly, the choice of the model and parameters can significantly impact the results of the analysis. Furthermore, domain-specific knowledge is often necessary to interpret the results accurately. Finally, the definition of what constitutes an anomaly can be subjective and may depend on the specific use case.

Bibliography

[1] Jacob, D., & Murugesan, V. (2017). *What is quality 4.0?* LNS Research Blog. Retrieved March 6, 2023, from https://blog.lnsresearch.com/quality40

[2] *What is Industry 4.0 and how does it work?* IBM. (n.d.). Retrieved March 6, 2023, from https://www.ibm.com/it-it/topics/industry-4-0

[3] *Cos'è l'Industria 4.0 e come funziona?* IBM. (n.d.). Retrieved March 6, 2023, from https://www.ibm.com/it-it/topics/industry-4-0

[4] Talkwalker.com. (n.d.). *The #1 Consumer intelligence acceleration platform*. Talkwalker. Retrieved March 6, 2023, from https://www.talkwalker.com/

[5] Liu, M., Sethi, S. P., & Zhang, J. (2019). International Journal of Production Research.

[6] Mastrogiacomo, L., Barravecchia, F., Franceschini, F., & Marimon, F. (2021). Mining quality determinants of product-service systems from user-generated contents. *Quality Engineering*, *33*(3), 425-442.

[7] Chen, K., Kou, G., Shang, J., & Chen, Y. (2015). Visualizing market structure through online product reviews: Integrate topic modeling, TOPSIS, and multi-dimensional scaling approaches. *Electronic Commerce Research and Applications*, *14*(1), 58-74.

[8] Duan, W., Gu, B., Whinston, A.B., (2008). The dynamics of online word-of-mouth and product sales—an empirical investigation of the movie industry. Journal of Retailing 84 (2), 233–242.

[9] Shao, K.-H., (2012). The effects of controversial reviews on product sales performance: the mediating role of the volume of word of mouth. International Journal of Marketing Studies 4 (4), 32–38.

[10] Godes, D., Mayzlin, D., Chen, Y., Das, S., Dellarocas, C., Pfeiffer, B., Libai, B., Sen, S., Shi, M., Verlegh, P., (2005). The firm's management of social interactions. Marketing Letter 16 (3), 415–428.

[11] Chintagunta, P.K., Gopinath, S., Venkataraman, S., (2010). The effects of online user reviews on movie box office performance: accounting for sequential rollout and aggregation across local markets. Marketing Science 29 (5), 944–957.

[12] Chevalier, J.A., Mayzlin, D., (2006). The effect of word of mouth on sales: online book reviews. Journal of Marketing Research 43 (3), 345–354.

[13] Cooper, L.G., Inoue, A., (1996). Building market structures from consumer preferences. Journal of Marketing Research 33 (3), 296–306.

[14] John, D.R., Loken, B., Kim, K., Monga, A.B., (2006). Brand concept maps: a methodology for identifying brand association networks. Journal of Marketing Research 43 (4), 549–563.

[15] Feldman, R., Fresko, M., Goldenberg, J., Netzer, O., Ungar, L.H., (2007). Extracting product comparisons from discussion boards. In: Proceedings of the 2007 Seventh IEEE International Conference on Data Mining. Association for Computing Machinery, New York, pp. 469–474.

[16] Lee, T.Y., Bradlow, E.T., (2011). Automated marketing research using online customer reviews. Journal of Marketing Research 48 (5), 881–894.

[17] Netzer, O., Feldman, R., Goldenberg, J., Fresko, M., 2012. Mine your own business: market-structure surveillance through text mining. Marketing Science 31 (3), 521–543.

[18] Chandra, A. (2020). *Recent works in topic modeling*. Medium. Retrieved March 6, 2023, from https://medium.com/data-folks-indonesia/recent-works-in-topic-modeling-56c38da8dfc4

[19] Xiong, H., Cheng, Y., Zhao, W., & Liu, J. (2019). Analyzing scientific research topics in manufacturing field using a topic model. *Computers & Industrial Engineering*, *135*, 333-347.

[20] Porter, K. (2018). Analyzing the DarkNetMarkets subreddit for evolutions of tools and trends using LDA topic modeling. Digital Investigation, 26, S87-S97.

[21] Zaki, M., & McColl-Kennedy, J. R. (2020). Text mining analysis roadmap (TMAR) for service research. *Journal of Services Marketing*, *34*(1), 30-47.

[22] Kandukuri, M., & HaraGopal, V. V. (2021). Topic Modelling Extraction of "Mann Ki Baat". *European Journal of Mathematics and Statistics*, *2*(1), 1-12.

[23] Xiong, H., Cheng, Y., Zhao, W., & Liu, J. (2019). Analyzing scientific research topics in manufacturing field using a topic model. *Computers & Industrial Engineering*, *135*, 333-347.

[24] Abuhay, T. M., Nigatie, Y. G., & Kovalchuk, S. V. (2018). Towards predicting trend of scientific research topics using topic modeling. *Procedia Computer Science*, *136*, 304-310.

[25] Lozano, M. G., Schreiber, J., & Brynielsson, J. (2017). Tracking geographical locations using a geo-aware topic model for analyzing social media data. *Decision Support Systems*, *99*, 18-29.

[26] Hosseini, S. Y., & Ziaei Bideh, A. (2014). A data mining approach for segmentation-based importance-performance analysis (SOM–BPNN–IPA): a new framework for developing customer retention strategies. *Service Business*, *8*, 295-312.

[27] Saabith, A. S., Fareez, M. M. M., & Vinothraj, T. (2019). Python current trend applications-an overview. *International Journal of Advance Engineering and Research Development*, *6*(10).

[28] Johari, A. (2020). *Python pandas guide - learn pandas for data analysis*. Medium. Retrieved March 6, 2023, from https://medium.com/edureka/python-pandas-tutorial-c5055c61d12e

[29] Kashosi, A., & Nazarevych, T. (2021). Heart rate variability analysis toolkit for further analysis of human stres. *Матеріали IV Міжнародної студентської науково-технічної конференції „Природничі та гуманітарні науки. Актуальні питання "*, 10-11.

[30] Cook, A. A., Mısırlı, G., & Fan, Z. (2019). Anomaly detection for IoT time-series data: A survey. *IEEE Internet of Things Journal*, *7*(7), 6481-6494.

[31] Aggarwal, C. C., & Aggarwal, C. C. (2017). *An introduction to outlier analysis* (pp. 1-34). Springer International Publishing.

[32] CFI Team. (2023). *Python (in Machine Learning)*. Corporate Finance Institute. Retrieved March 6, 2023, from https://corporatefinanceinstitute.com/resources/data-science/python-in-machine-learning/

[33] Patel, S. M., Dabhi, V. K., & Prajapati, H. B. (2017). Extractive Based Automatic Text Summarization. *J. Comput.*, *12*(6), 550-563.

[34] Blei, D. M. (2012). Probabilistic topic models. *Communications of the ACM*, *55*(4), 77-84.

[35] Scientist, A. G. M. I. D., Author: Brendan Martin Founder of LearnDataSci, & Author: Lauren Washington Lead Data Scientist & ML Developer. (n.d.). *Python pandas tutorial: A complete introduction for beginners*. Learn Data Science - Tutorials, Books, Courses, and More. Retrieved March 6, 2023, from https://www.learndatasci.com/tutorials/python-pandas-tutorial-complete-introduction-for-beginners/

[36] Johari, A. (2020). *Python pandas guide - learn pandas for data analysis*. Medium. Retrieved March 6, 2023, from https://medium.com/edureka/python-pandas-tutorial-c5055c61d12e

[37] Aladakatti, S. S., & Senthil Kumar, S. (2022). Exploring natural language processing techniques to extract semantics from unstructured dataset which will aid in effective semantic interlinking. *International Journal of Modeling, Simulation, and Scientific Computing*, 2243004.

[38] Li, S. (2019). *Anomaly detection for dummies*. Medium. Retrieved March 6, 2023, from https://towardsdatascience.com/anomaly-detection-for-dummies-15f148e559c1