**Web and Social Media Analytics**

***7 BUIS025W.3***

**Hasan Enes Guray**

**Student ID: 19489124**

**Word Count:**

**SOCIAL MEDIA COURSEWORK  
ELON MUSK**

|  |
| --- |
| *Hasan E. Guray* |
|  |
| MSc Data Science and Analytics student |
| University of Westminster |
| w1948912@my.westminster.ac.uk |

ABSTRACT

This research paper presents a case study focused on analysing public perceptions of Elon Musk after his latest decision to change the Twitter as X. Leveraging a dataset comprising 2000 Reddit comments, the study delves into five primary discussion topics and employs three distinct sentiment analysis methods to gain comprehensive insights. The findings shed light on the sentiment surrounding the businessman’s portrayal and offer valuable implications for understanding general reception in the society.

**Keywords:** Elon Musk, Twitter, Reddit comments, Sentiment analysis, Topic modelling.

# PERSON SELECTED

Elon Musk, a prominent business magnate and investor, holds numerous distinguished roles, serving as the founder, chairman, CEO, and chief technology officer of SpaceX, CEO and product architect of Tesla, Inc., owner and CTO of Twitter, founder of the Boring Company, and co-founder of both Neuralink and OpenAI. Additionally, he assumes the presidency of the Musk Foundation. As of July 2023, his estimated net worth amounts to an astounding US$248.8 billion, rendering him the wealthiest individual globally.

Acknowledged for his enterprising vision and unwavering dedication to fostering societal betterment, Elon Musk remains a figure of considerable controversy. Nonetheless, he garners widespread admiration for his transformative initiatives. Throughout his lifetime, he has acquired a reputation for consistently astounding the public with both positively and negatively surprising decisions.

Recently, one of Musk's unexpected decisions involved rebranding the long-established and widely recognized Twitter platform to the new name "X." By announcing this change on Twitter itself, where he actively engages as a social media user, Musk demonstrated his propensity for unveiling innovations in unprecedented ways. Notably, as of August 2, 2023, Musk boasts a staggering follower count of 150.9 million on this platform, while he selectively follows a mere 396 individuals. Evidently, Musk holds the distinction of having the largest number of followers on the X platform (https://www.brandwatch.com/blog/most-twitter-followers/ accessed on 2 August 2023).

# Data COLLECTION

In the present era characterized by technological advancements, the rapid dissemination of information and ideas has become notably accelerated, with social media playing a significant role in facilitating this process. Numerous social media platforms offer a virtual space where users can freely exchange ideas and engage with one another. To conduct an in-depth analysis of Elon Musk's public perception and assess his recent professional accomplishments, Reddit, a platform known for its free API support and substantial user traffic, was employed to establish the dataset.

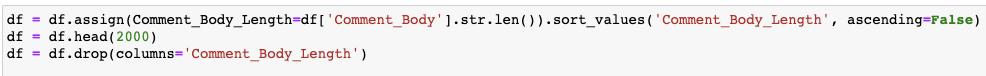
Reddit has garnered substantial global popularity, securing its position as the 10th most frequented website, according to June 2023 data (https://www.semrush.com/website/top/ accessed on 29 July 2023). To ensure the acquisition of pertinent data for comprehensive analysis, the date range spanning from 24th to 30th July 2023 was meticulously selected. This period encompasses the commencement of the change process, initiated by the Twitter CEO, and extends over a one-week interval, facilitating a thorough exploration of its societal ramifications.

Employing Python's pandas, praw, and datetime libraries, an extensive dataset comprising 5974 comments (Figure 1) was meticulously compiled by leveraging Reddit's API support. To enhance the meaningfulness of the results, a filtering process was executed, focusing on the 2000 comments boasting the longest character count (Figure 2).

A screenshot of a computer

Description automatically generated

**Figure 1** *This is a screenshot of Python code to scrape Reddit comments about Elon Musk.*



**Figure 2** *This is a screenshot of Python code to filter the longest 2000 Reddit comments about Elon Musk.*

By leveraging this dataset, researchers can delve into the perceptions and discussions surrounding Elon Musk recent work and gauge the sentiment and popularity of his projects within the Reddit community during the specified time frame. The abundance of user-generated content on Reddit offers valuable insights into the reception of her performances and provides a unique opportunity for data-driven analysis in the realm of data science.

In order to mitigate potential dataset ambiguity arising from user-to-user interactions, exclusively top-level comments were focused. Moreover, to ensure data integrity, certain comments, namely those auto-generated by the system and those intended for specific users, were excluded from the analysis. As a consequence of these filtering measures, a comprehensive dataset was constructed, encompassing pertinent attributes such as "Comment Body" to encapsulate the comment content, "Author" denoting the user responsible for the comment, "Upvotes" and "Awards" to account for interactions received by the comment, and "Date" indicating the precise timestamp of comment dissemination on the platform (Figure 3).

A screenshot of a computer

Description automatically generated

**Figure 3** *This is a screenshot of Pandas dataframe, consists Reddit comments about Elon Musk.*

Undoubtedly, given the unprocessed nature of the dataset, replete with raw user-generated comments, it is imperative to undertake diligent data cleaning and pre-processing before commencing the exploratory data analysis phase. Consequently, during the examination of the dataset, the initial pre-processing step involved the conversion of the "Date" column into the datetime format (Figure 4). This conversion rendered the temporal information amenable to subsequent analytical procedures.

A screenshot of a computer code

Description automatically generated

**Figure 3** *This is a screenshot of Pandas dataframe’s column details, consists Reddit comments about Elon Musk.*

To optimize the efficacy of topic model analysis and sentiment analysis procedures, it is essential to subject the raw comments to preparatory steps. In this context, Python's contractions library was employed to standardize the comments, effectively resolving contractions within the text and thereby facilitating a more coherent representation (Figure 5). By executing this pre-processing operation, the comments became suitably prepared for subsequent analytical operations, enhancing the overall success of the topic model and sentiment analysis processes.

A screenshot of a computer

Description automatically generated

**Figure 5** *This is a screenshot of a Python formula to fix contractions.*

Subsequently, to ensure uniformity in the analytical treatment of textual content, Python's "re" library was leveraged to standardize words encompassing price, size, volume, and weight attributes (Figure 6). By accomplishing this standardization, the potential disparity arising from disparate numerical representations was effectively mitigated, safeguarding the consistency and integrity of the subsequent analyses.

A screenshot of a computer program

Description automatically generated

**Figure 6** *This is a screenshot of a Python formula to substitute patterns.*

Within the corpus of comments, a notable presence of hyperlinks has been observed, serving the purpose of guiding users to external online resources. The inclusion of such links can serve various functions, ranging from explanatory or supportive references to instances of humorous commentary. In the interest of enhancing the efficacy of the analyses, it becomes imperative to standardize these hyperlinks, consolidating them under a singular category. To this end, the "re" library in Python is harnessed to effectively aggregate the links and establish a coherent representation (Figure 7). By undertaking this standardization process, the analyses stand to benefit significantly, ensuring greater consistency and interpretability throughout the investigative endeavours.

A screenshot of a computer

Description automatically generated

**Figure 7** *This is a screenshot of the Python formula to replace http links.*

In this phase of the investigation, comments comprising only a singular hyperlink were identified and subsequently omitted from the dataset, as such comments were deemed inconsequential to the analytical process (Figure 8).

A close-up of a white background

Description automatically generated

**Figure 8** *This is a screenshot of the Python formula to filter out the comments comprising only a singular hyperlink.*

While navigating the Reddit platform, the occurrence of the "\n" expression is not apparent. However, it has come to light that certain comments within the scraped dataset employ this expression as a representation of line transitions. To rectify this extraneous presence and foster a more streamlined data representation, Python's "replace" formula was employed (Figure 9). Through this process, these inconsequential expressions were effectively expunged, resulting in a more coherent and structured dataset, thus facilitating subsequent analyses with greater precision and clarity.

A screenshot of a computer error

Description automatically generated

**Figure 9** *This is a screenshot of the Python formula to remove line breaks.*

Various factors, such as hurried composition, social media-specific abbreviations, and other contributing sources, contribute to the prevalence of typographical errors in the comments. Left unaddressed, these inaccuracies have the potential to jeopardize the veracity of analytical outcomes. To rectify these issues, the utilization of Python's TextBlob library enables the correction of misspellings and the standardization of words sharing a common root. Nonetheless, it is crucial to exercise caution in processing entire comments collectively, as such an approach can inadvertently render certain accurate words erroneous (Figure 10).

A screenshot of a computer

Description automatically generated

**Figure 10** *This is a screenshot of the Python formula to correct the words.*

To mitigate this challenge, a more meticulous approach was adopted, wherein mistranslated words were identified before subjecting the words individually through the TextBlob library (Figure 11). This refined methodology yielded more precise and reliable results, circumventing the risk of false corrections and enhancing the overall accuracy of the analyses.

A screenshot of a computer program

Description automatically generated

**Figure 11** *This is a screenshot of the Python formula to correct the words.*

In pursuit of optimizing the efficacy of topic model analysis, comprehensive standardization assumes paramount importance. To this end, an essential pre-processing step involves the conversion of all letters in the comments to lowercase, thereby enhancing uniformity and consistency. Employing Python's "lower()" formula, this operation was effectively carried out (Figure 12).

A screenshot of a computer

Description automatically generated

**Figure 12** *This is a screenshot of the Python formula to convert to lowercase all of the letters.*

However, it is noteworthy that the distinction between uppercase and lowercase letters holds particular significance in sentiment analysis. This aspect serves to convey diverse emotional states or to accentuate emotional emphasis. Consequently, it is judicious to exercise caution, and, thus, the application of lowercase transformation and related pre-processing steps are judiciously withheld from sentiment analysis and subsequent stages.

An additional instrumental procedure that significantly enhances the efficacy of the topic model analysis pertains to tokenization. Tokenization represents a pivotal phase in the preparatory treatment of textual data for topic model analysis. This crucial process facilitates the conversion of unstructured text into a structured framework, thereby enabling topic modelling algorithms to discern underlying patterns and extract coherent topics of significance. Through the standardization of word representations and the removal of extraneous noise, tokenization substantially elevates the accuracy and interpretability of the topic modelling process.

In this context, the implementation of tokenization was accomplished utilizing Python's "nltk" library (Figure 13). The application of this natural language processing tool empowered the segmentation of textual content into discrete tokens, aligning with the data pre-processing requirements of topic modelling, and ultimately fostering a more robust and informative analytical outcome.

A screenshot of a computer

Description automatically generated

**Figure 13** *This is a screenshot of the Python formula to convert to tokenize the words.*

The exclusion of stopwords, which are function words devoid of substantive meaning on their own, constitutes an essential endeavour in topic model analysis. By eliminating these commonly occurring, non-informative words, the influence of such extraneous terms is mitigated, thereby enhancing the identification and comprehension of meaningful patterns within the text. This pre-processing step significantly augments the overall quality and interpretability of the derived topics by ameliorating noise levels, enhancing topic coherence, and channelling the model's attention towards more pertinent and distinctive terms.

To execute this pivotal undertaking, the "nltk" library within the Python programming language was judiciously employed (Figure 14). Notably, within this library, the term "not" has been designated as a stopword; however, deliberate discretion was exercised in retaining the term "not" given its capacity to exert transformative influence on the analytical outcomes. The deployment of this natural language processing resource facilitated a methodical eradication of stopwords, congruent with the imperatives of data purification essential for proficient topic model scrutiny. As a result, this procedure engendered the extraction of topics that were not only more refined and perceptive, but also characterized by enhanced cohesiveness, thereby heightening the perceptibility of latent structures and discernments inherent within the textual corpus.

A screenshot of a computer

Description automatically generated

**Figure 14** *This is a screenshot of a Python formula to remove stopwords.*

Of paramount significance within comments are emojis and punctuation marks. In the context of topic model analysis, their removal assumes indispensable importance as it serves to alleviate noise, enhance coherence, and elevate the interpretability of the resulting topics. By expunging emojis and punctuation marks, a refined and standardized textual representation is attained, focusing attention on the salient lexical content, mitigating dimensionality, and cultivating a cleaner data format conducive to efficient and precise topic modelling.

To effectuate this vital pre-processing task, the Python "re" and "string" libraries were judiciously employed (Figure 15). Through the strategic application of these libraries, emojis and punctuation marks were systematically deleted, aligning with the data refinement requisites essential for rigorous topic model analysis. Consequently, this preparatory operation not only fostered a more discerning identification of meaningful lexical elements but also contributed to the establishment of a robust and amenable foundation for the extraction of insightful and coherent topics from the textual corpus.

A screenshot of a computer program

Description automatically generated

**Figure 15** *This is a screenshot of a Python formula to remove punctuations and emojis.*

Lemmatization constitutes a pivotal pre-processing phase in topic model analysis, bearing remarkable value in the normalization of words to their base forms while consolidating various inflected word variations. This transformative process yields substantial improvements in the interpretability and coherence of the text, while concurrently reducing the dimensionality of the vocabulary. As a result, lemmatization fosters the generation of more meaningful and efficient topic modelling outcomes, faithfully capturing the essence of the data's core themes with heightened precision.

The implementation of lemmatization realized through Python's "nltk" library, embodied a potent means to accomplish this essential data refinement task (Figure 16). By capitalizing on the linguistic insights afforded by the "nltk" library, lemmatization profoundly influences the topic model analysis, facilitating a more sophisticated and nuanced understanding of the underlying semantic structures. In turn, this contributes significantly to the pursuit of more robust, informative, and insightful topic modelling outcomes that resonate coherently with the textual corpus at hand.

A screenshot of a computer

Description automatically generated

**Figure 16** *This is a screenshot of the Python formula for lemmatisation.*

To ensure analytical integrity and mitigate complexity arising from redundant data, it is imperative to expunge duplicate entries, particularly identical comments from the same users. This de-duplication process is essential to maintain data coherence and prevent undue influence on the analysis. Leveraging Python's "drop\_duplicates()" function, the issue of duplicates was effectively circumvented (Figure 17).

Through the implementation of this function, duplicate instances were efficiently identified and removed, thus fostering a more streamlined and unambiguous dataset. By undertaking this preparatory step, the risk of inflated influence or skewed representation of certain comments was averted, rendering the ensuing analysis more robust and trustworthy. The exclusion of duplicates engendered enhanced data quality and contributed to the generation of more accurate and credible insights in the context of the data science investigation.



**Figure 17** *This is a screenshot of the Python formula to remove duplicates.*

Following the comprehensive data cleaning and pre-processing procedures, certain comments were identified to be entirely devoid of meaningful content and were subsequently rendered as "na" values. To ensure the dataset's integrity and to obviate the influence of these empty entries on subsequent analyses, Python's "dropna()" function was diligently employed (Figure 18).

Through the application of this function, all instances containing "na" values were expunged, thereby refining the dataset and purging it of any remaining vestiges of uninformative or incomplete data. This judicious step contributed to the generation of a more robust and reliable dataset, free from any spurious or misleading information. By removing the "na" data, the analytical procedures stood to yield more accurate and insightful outcomes, unencumbered by the presence of extraneous or erroneous entries.

A close up of a sign

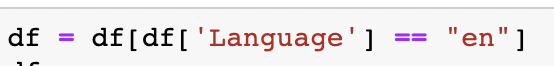
Description automatically generated

**Figure 18** *This is a screenshot of the Python formula to remove na records.*

The linguistic composition of comments plays a decisive role in the analysis, as the analytical procedures are tailored to a specific language context. Given the prevalence of English as the predominant language, the removal of non-English records proves imperative to engender more efficacious results. Consequently, the dataset was meticulously purged of non-English comments, a process executed via the Python "langdetect" library, resulting in the retention of 1899 rows of data (Figure 19).

A screenshot of a computer program

Description automatically generated



**Figure 19** *This is a screenshot of the Python formula to keep just English comments.*

By detecting and eliminating non-English content using the "langdetect" library, the dataset became more homogenous and amenable to the intended analysis, fostering enhanced analytical outcomes that align with the prescribed linguistic focus. This data refinement endeavour ensured that the subsequent analyses were conducted within a coherent language context, obviating the confounding effects of language diversity, and rendering the conclusions more robust and relevant within the purview of the data science exploration.

# EXPLORATORY ANALYSIS

Regarding data acquisition, a specific temporal window of one week was chosen, commencing from the date of Twitter's transition to X. The dataset encompassed the period from 24th to 30th July 2023, during which a noticeable augmentation has been observed (Figure 20). Nevertheless, it is important to acknowledge that the encompassed comments are not exclusively from the Reddit platform; rather, they consist of the 2000 lengthiest comments recorded between the specified dates. Thus, it can be inferred that an elevated number of users who have encountered application modifications on their mobile devices have contributed to the composition of lengthier comments.

A graph of a bar graph

Description automatically generated

**Figure 20** *This is a histogram of the comments count related to Elon Musk on Reddit.*

To enhance the efficacy of the analyses, it was deemed imperative to ascertain the presence of concentration among specific authors, as having diverse contributors for the comments would facilitate more robust conclusions. As part of this investigation, the distribution of comments across various authors was scrutinized. Remarkably, it was observed that the author with the highest comment count contributed merely 6 comments, thereby substantiating the heterogeneous nature of the dataset, emanating from distinct individuals' sentiments and viewpoints (Figure 21).

A graph of a number of authors

Description automatically generated

**Figure 21** *This is a bar graph of the comments count of the authors related to Elon Musk on Reddit.*

Utilizing the upvotes and awards system as an interactive mechanism among users, an analysis was conducted on comments that garnered the most upvotes and awards. Among these, the comment contains the phrase “I will never understand how someone posts a picture of a video and then is satisfied the message was conveyed” attained the highest number of upvotes, amassing a total of 7553 and the highest number of 13 awards.

Given Reddit's global outreach, the dataset encompassed comments not only in English but also in diverse languages. In the examination of the language distribution, it was observed that 1899 comments were in English, constituting 95.2% of the entire dataset and occupying the foremost position. Subsequently, French and Italian followed in prominence (Figure 22).

A white and purple graph

Description automatically generated

**Figure 22** *This is a bar graph of the comments count of the languages related to Elon Musk on Reddit.*

Analysing the average comment length, Italian comments emerged as the most extensive, with an average of 190.88 characters, while French ranked second with an average length of 164.76 characters and English ranked third with an average length of 162.78 characters (Figure 23).

A graph of blue bars

Description automatically generated with medium confidence

**Figure 23** *This is a bar graph of the average comment length of the languages related to Elon Musk on Reddit.*

Conducting a comprehensive examination of term frequency during the initial stages of exploratory data analysis before undertaking sentiment and topic model analyses emerges as a pivotal step in comprehending data attributes, discerning prospective themes, and unearthing valuable insights into prevailing lexical usage patterns and emotional tendencies. This endeavour not only facilitates judicious decision-making concerning text pre-processing, feature selection, and data cleansing but also expedites the generation of a concise data synopsis and facilitates timely anomaly detection.

The prominent revelations extracted from the comprehensive term frequency analysis of the comments have unveiled the lexicon's most frequently deployed terms. These preeminent terms include, in a descending order of prevalence, "twitter", "elon", "people", "size", "like", and "musk". A notable observation arises in the pertinence accorded to the term "size" a representation intrinsically linked to numerical manifestations. Additionally, within the encompassing corpus, the lexemes "x", "company", and "tesla" held positions among the upper echelon of the 20 most recurrently reiterated terms. Particularly within the confines of the dataset, where the transformation of Twitter into X assumes a central thematic focus, these findings incontrovertibly underscore the vigilant cognizance Reddit users exhibit toward the unfolding discourse.

Furthermore, predominant among the recurrently transpiring noun phrases were "elon musk", "free speech", "social medium", and "dark mode". This composite expression signifies the paradigm shift to X, emblematic of the shift from a white to a black visual background, serving as the default configuration. The “social media” phrase was changed as “social medium” in lemmatisation process. These phrases and words were associated with Elon Musk.

A close up of words

Description automatically generated

**Figure 24** *This is a wordcloud of the most common words in the comments related to Elon Musk on Reddit.*

# TOPIC MODEL ANALYSIS

Topic modelling represents a pivotal approach for unearthing primary thematic patterns within Reddit comments datasets. By adeptly categorizing comments into coherent topics, it facilitates effective data exploration and comparative analysis. Nonetheless, its application entails certain drawbacks, such as contextual deficiency, potential topic ambiguity, and reliance on pre-processing methodologies. Augmenting topic modelling with sentiment analysis, employing diverse algorithms, and exercising judicious interpretation can substantially augment its efficacy in generating profound insights.

Latent Dirichlet Allocation (LDA) emerges as a widely adopted topic modelling technique for text data, owing to its capability to autonomously discover topics, yield coherent and interpretable thematic representations, and scale adeptly to handle extensive datasets. LDA's adaptability in controlling the number of derived topics and its pervasive integration within the domain of natural language processing further reinforce its popularity.

To cater to the appropriateness of the dataset and ensure sufficient capacity, LDA topic modelling, complemented by the support of Python's nltk and gensim libraries, was executed, resulting in the identification of five distinct topics (Figure 25).

A screenshot of a computer program

Description automatically generated

A screenshot of a computer code

Description automatically generated

**Figure 25** *This is a Python code to build LDA topic modelling.*

Topic 0 was related with the “Opinions and Reactions”. This topic encompassed a range of opinions and reactions from Reddit users regarding Elon Musk's transformation of Twitter into X. People expressed their thoughts and preferences, discussing how they feel about this decision and its potential implications.

Topic 1 was related with the “Elon Musk's Role and Influence”. Centered on Elon Musk's influence, this topic delved into his pivotal role in the transformation. Commenters discuss Musk's impact on the change, considering its effects on both the platform's scale and its users.

Topic 2 was related with the “Company and Brand Impact”. Focusing on the business aspect, this topic explored the impact of the transformation on the company and its brand. Discussions revolved around the effects on Tesla, the scale of the changes, and how Elon Musk's decisions tied into the transformation.

Topic 3 was related with the “Thoughtful Analysis”. In this topic, Reddit users engaged in analytical conversations about the transformation. Speculative in nature, the discussions involved considerations about potential outcomes, Musk's intentions, and the significance of various elements, such as speeches.

Topic 4 was related with the “Desires and Expectations”. Centered on user desires, this topic captured discussions about what people want and expect from the transformation. The conversations revolved around users' expectations, desires for certain outcomes, and how the changes might align with their preferences.

According to the available dataset, the comments took place the most in the "Company and Brand Impact" topic with 32.5%. While the current dataset has allowed the determination of five distinct topics, it is essential to subject these topics and their corresponding keyword weights to rigorous testing with a more extensive dataset to ensure the establishment of more robust and effective thematic representations. However, the present dataset enabled the evaluation of user comments under five distinct thematic headings.

A pie chart with different colored sections with Crust in the background

Description automatically generated

**Figure 26** *This is a pie chart to show the topic distribution of the Reddit Comments about Elon Musk.*

# SENTIMENT ANALYSIS

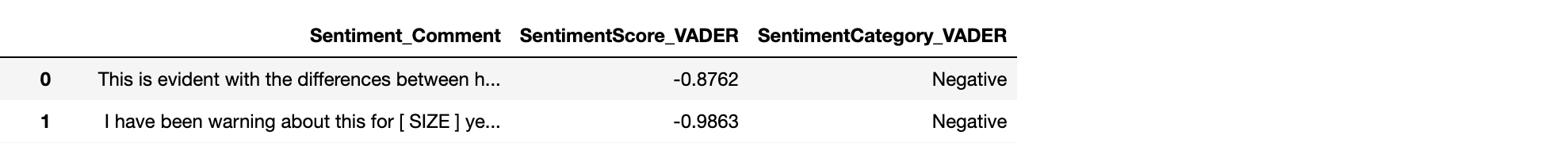
Sentiment analysis assumes a pivotal role in the domain of social media modelling, wherein it extracts and quantifies sentiments conveyed within user-generated content, such as comments on popular platforms like Reddit. By scrutinizing sentiments, one gains valuable insights into the public's perceptions, emotions, and responses concerning specific topics, such as Elon Musk's recent decision. These discernments prove instrumental for various stakeholders, including businesses, businessmen, and researchers, as they strive to gauge public reception, identify trends, and make data-driven decisions based on user feedback.

The utilization of diverse sentiment analysis methodologies across various disciplines necessitates a systematic approach to select the most appropriate method for a given dataset. This involves a rigorous process of trying different techniques, evaluating and comparing their performance, and integrating them with machine learning models. One prominent model employed for such analysis is VADER (Valence Aware Dictionary and Sentiment Reasoner), which operates as a lexicon and rule-based sentiment analysis framework specifically tailored for social media texts. By discerning the emotional tone of textual data, VADER effectively categorizes it as positive, negative, or neutral sentiment.

In the context of this research, the VADER model, bolstered by Python's nltk library, was implemented to analyse the provided dataset (Figure 27). Consequently, sentiment categories were established based on both sentiment scores and sentiments themselves. Specifically, comments with sentiment scores equal to or above 0.05 were deemed positive, while those with sentiment scores equal to or below -0.05 were considered negative. It was observed that a substantial portion of the dataset fell within the range of 0.0 to 0.1 (Figure 28).

A screenshot of a computer program

Description automatically generated



**Figure 27** *This is a Python code to build VADER sentiment analysis.*

A graph of blue bars

Description automatically generated

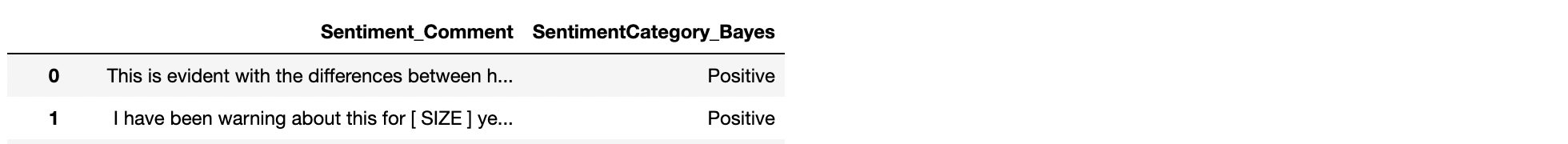
**Figure 28** *This is a histogram to show the VADER sentiment analysis score breakdown.*

Further investigation into the extreme ends of the sentiment spectrum revealed that VADER demonstrated remarkable success in precisely detecting positive and negative emotions. However, the predominance of dataset entries near the sentiment boundaries necessitates closer scrutiny and analysis. For instance, while the comment with the highest sentiment score was extensive, its most poignant sentence, "He's been able to communicate a vision of the future that everyone wants to believe in" contributed significantly to the overall score. On the other hand, the comment with the lowest sentiment score, despite its length, featured the striking sentence, "Now he is stuck with a company that doesn't really make any money" prominently influencing the overall negative sentiment. These findings suggest that VADER excels in discerning sentiments at the extremities but needs in-depth consideration when dealing with sentiments concentrated around the boundary values.

TextBlob's sentiment classifier represents a pre-trained machine learning model proficient in analysing and categorizing the sentiment of textual content as positive, negative, or neutral based on the presence of specific words and phrases. While this model, trained using the movie reviews dataset, may yield more accurate outcomes when applied directly to movie-related comments, its development involved the utilization of Python's textblob and nltk libraries (Figure 29).

A screenshot of a computer program

Description automatically generated



**Figure 29** *This is a Python code to build TextBlob’s sentiment classifier.*

Distinct from VADER, the TextBlob model generated separate sentiment scores for both negative and positive sentiments, and it is generally regarded as indicative of a particular emotion when it surpasses the threshold of 0.6. Consequently, precisely 1051 comments have been identified as different emotion when compared with VADER's categorization.

It is essential to highlight that, despite its ability to produce multiple sentiment scores, the TextBlob model's categorization approach contrasts with VADER's single-score output. Consequently, their utilization and interpretation demand careful consideration to ensure meaningful insights are drawn from the sentiment analysis process.

Stanford University's SocialSent analysis is a sentiment analysis approach that operates on a corpus of 1900-2000 words, employing predetermined scores (https://nlp.stanford.edu/projects/socialsent/ accessed on 31 July 2023). The sentiment score is derived by summing the preassigned coefficients and then dividing them by the total number of words in the text (Figure 30). Upon conducting this analysis, it was observed that a significant proportion of the dataset resulted in sentiment scores proximate to 0 (Figure 31).

A screen shot of a computer code

Description automatically generated

**Figure 30** *This is a Python code to build SocialSent sentiment analysis.*

A graph of a column

Description automatically generated with medium confidence

**Figure 31** *This is a histogram to show the SocialSent sentiment analysis score breakdown.*

As per the SocialSent analysis, the comment with the highest sentiment score contained the phrase "I honestly am glad Elon is destroying the platform." while the comment with the lowest sentiment score contained the phrase " HOW STUPID IS HE TWITTER IS DESTROYED FOR REAL THIS TIME". While it is seen that SocialSent could not fully grasp some of the ironies here, it could be concluded that the positiveness of most comments that it evaluated as positive should be approached with suspicion.

This implies that the two sentiment analysis methodologies, VADER and SocialSent, diverge in their response to different comments, potentially influencing the interpretation and effectiveness of sentiment analysis based on comment structure. Consequently, these findings underscore the significance of considering the specific characteristics and peculiarities of each sentiment analysis technique to derive comprehensive and accurate insights from user-generated content.

Upon comprehensive evaluation of all three sentiment analysis methodologies, it is seen that the sentiment categorization process is quite different from each other (Figure 32). Although positive comments dominate for all three analyses, the difference between positive comments and negative comments in Bayes is quite large, while in SocialSent it is quite close to each other. The number of comments classified as neutral is quite low compared to other categories. In general, the conclusion that there are many positive comments shows that sentiment analysis methodologies have difficulty in understanding some ironies.

A graph of different colored bars

Description automatically generated

**Figure 32** *This is a bar chart to compare the sentiment category of VADER, TextBlob, and SocialSent analysis.*

When juxtaposed against a machine learning-based sentiment analysis model established using the NaiveBayesClassifier, TextBlob with NaiveBayes emerged as the most efficacious technique, achieving an accuracy rate of 56%. Subsequently, VADER achieved a commendable accuracy rate of 51%, followed by SocialSent with 50%. These findings underscore the superiority of Bayes as a more suitable model for the dataset under consideration. However, it also shows that it is necessary to work with larger datasets in order to achieve more successful results.

The discernible disparity in accuracy rates between Bayes and the other methods signifies the former's enhanced capability in capturing the intricate nuances of sentiment expressed within the dataset. Consequently, these outcomes hold crucial implications for the selection of an appropriate sentiment analysis model to derive optimal insights from user-generated content in this context.

Sentiment analysis comment’leri sadece positive ya da negative olarak ayırmaktan ibaret değildir. Commentler ayrıca barındırdırdığı duygu unsuruna göre de sınıflandırılabilmektedir. LexMo kütüphanesiyle English texts duygulara göre classify edilebilmektedir (<https://betterprogramming.pub/unlocking-emotions-in-text-using-python-6d062b48d71f> accessed on 3 August 2023).

# CONCLUSION

In this case study, Reddit comments related to Elon Musk and his recent decision about transforming the “Twitter” into “X” were analysed. A dataset of 2000 comments was collected and underwent rigorous data cleaning and pre-processing. The exploratory analysis revealed a consistent daily influx of comments, and the dataset exhibited heterogeneity with diverse contributions from various authors.

The topic model analysis using Latent Dirichlet Allocation (LDA) identified five distinct topics, encompassing discussions about Elon Musk’s opinions and reactions, role and influence, company and brand impact, thoughtful analysis, and desires and expectations.

Three sentiment analysis methodologies, VADER, TextBlob, and SocialSent, were employed to assess the sentiments expressed in the comments. While all methods showed a higher proportion of positive sentiments, TextBlob emerged as the most accurate model for this dataset, achieving a notable accuracy rate of 56%.

However, improvements and further developments are recommended, such as increasing the dataset size for better topic modelling and sentiment analysis. Overall, the case study provided valuable insights into public perceptions and sentiments surrounding Elon Musk, but ongoing enhancements in data collection and analysis methodologies would lead to more robust and accurate conclusions.

references