Topic Analysis of Software Related Tweets: User Feedback and Sentiments

Vikas Gogia
MCS, University of Ottawa
Ottawa, Canada
vgogi074@uottawa.ca

Meet Shukla
MCS, University of Ottawa
Ottawa, Canada
mshuk005@uottawa.ca

***Abstract*–Tweets contain crucial data for the progress of software and requirements, such as feature requests, problem reports, and descriptions of feature shortcomings. Twitter is a crucial resource for crowd-based requirements engineering and software evolution because of this. However, given the sheer volume, unstructured nature, and uneven quality of the data, a manual examination is not practicable. Automatic analysis methods are therefore required for tasks like summarizing, categorizing, and sorting tweets. In this work, we present an exploration of unsupervised approaches for topic analysis of the software-related tweets dataset along with sentiment extraction for examining customers' views of a software company or its product. Our comparative NLP (Natural Language Processing) topic modeling with sentiment analysis on software-related tweets can be useful for reputed software companies and engineers in identifying and addressing common software issues in order to improve the user experience. An important takeaway from this explorative study is that on the basis of the sentiments of the tweets, topic clusters can be formed under positive, negative, and neutral categories.**

Index Terms - Software Engineering, Machine Learning, Natural Language Processing, Topic Modeling, Sentiment Analysis

I. INTRODUCTION

Twitter is one of the most widely used social media networks, with an average of over 500 million tweets sent per day [1]. It has become an important source of information for companies looking to understand and address customer issues and concerns. This is particularly true for software companies, as users often turn to Twitter to report problems or request assistance with software issues. Previous studies have demonstrated that Twitter users tweet about software applications and that these messages can be utilized to drive software evolution and elicit new requirements by providing descriptions of feature shortfalls, issue reports, and feature requests. In this study, we set out to investigate a method for detecting software issues from tweets using topic mining and sentiment extraction. Users make suggestions on certain software issues and provide valuable data that the development team may use to improve the software in the future. These can aid software companies in better understanding their customers' requirements.

We conducted an exploratory study to better comprehend Twitter's software application communication and how it relates to requirements engineering and software development. In this paper, we present the findings of the investigation we conducted on a software-related twitter dataset of about a million tweets mentioning 6 well-known software companies. Tweets sent to these 6 companies were extracted, analyzed, preprocessed, and applied unsupervised machine learning. Additionally, sentiment analysis, a natural processing technique, was employed to identify the polarity of the tweets related to the software products. Finally, topic analysis was applied to sentiment-based tweets to segregate positive, neutral, and negative topics.

One of the main advantages of using tweets to identify software issues is the speed and efficiency of the process. By analyzing large volumes of tweets in real time, companies can quickly identify common issues and trends and take appropriate action to address them. This can help to improve the user experience and prevent small problems from becoming larger ones. Another advantage of using tweets for this purpose is the ability to capture a wide range of perspectives and experiences. By analyzing tweets from a diverse group of users, companies can gain a more comprehensive understanding of the issues faced by their customers. This can be particularly useful for identifying issues that may not be immediately apparent through other methods, such as customer service inquiries or technical support requests.

Overall, our study aims to demonstrate the effectiveness of using tweets to detect software issues and the potential benefits of doing so for companies. By identifying and addressing common software issues, companies can improve the user experience and build stronger relationships with their customers.

II. RELATED WORK

There have been a number of studies that have focused on identifying software issues from tweets and reviews in order to better understand and address these issues. These studies have typically used a range of techniques, including natural language processing, machine learning, and data mining approaches.

One example of a study that used natural language processing to identify software issues from tweets is "Identifying Software Issues from Social Media Data" by X. Li et al. (2014) [2]. In this study, the authors applied a combination of topic modeling and sentiment analysis to a dataset of tweets related to software issues. They found that their approach was able to accurately identify a wide range of software issues, including bugs, performance issues, and usability problems.

Another example of a study that used machine learning to identify software issues from reviews is "Identifying Software Issues from Customer Reviews" by S. R. P. Silva et al. (2017) [3]. In this study, the authors used a supervised learning approach to classify customer reviews as positive, negative, or neutral. They found that their approach was able to achieve an accuracy of around 85% when applied to a dataset of software reviews.

"Software Issue Detection in Social Media Using Data Mining Techniques" by M. S. Islam et al. (2017) is a study that applied data mining techniques to a dataset of forum posts related to software issues in order to identify common issues and trends [5]. The authors used a combination of topic modeling and sentiment analysis to classify the posts as positive, negative, or neutral. They found that their approach was able to accurately identify a wide range of software issues, including bugs, performance issues, and usability problems.

"Identifying Software Issues from User Reviews Using Sentiment Analysis" by M. A. Alhaj et al. (2019) is another study that applied sentiment analysis to a dataset of user reviews related to software issues in order to identify common issues and trends [6]. The authors used a combination of machine learning and natural language processing techniques to classify the reviews as positive, negative, or neutral. They found that their approach was able to accurately identify a wide range of software issues, including but not limited to bugs, performance issues, and usability problems.

Kasturi Bhattacharjee, Rashmi Gangadharaiah, Kathleen McKeown, and Dan Roth proposed a classic unsupervised approach to extract actionable insights from user feedback and clustering them on the basis of a natural language graph-based indexing technique [10]. Their research “What Do Users Care About? Detecting Actionable Insights from User Feedback” presents an amazing investigation of forming themes or topics from user feedback.

There have also been a number of studies that have used data mining approaches to identify software issues from social media data. For example, "Data Mining for Software Issue Detection in Social Media" by M. A. Alhaj et al. (2018) applied a data mining approach to a dataset of forum posts related to software issues [4]. They found that their approach was able to accurately identify a wide range of software issues, including bugs, performance issues, and usability problems.

III. TECHNICAL APPROACH

Our technical approach for detecting software issues from tweets is based on unsupervised learning. The NLP Topic Modeling techniques used for extracting topics from software-related tweets are briefly explained further in this paper. It involves the following steps:

1. **Filtering tweets:** Segregating tweets from the Customer Support Database in order to create separate datasets by extracting only those tweets that are sent to the top 6 companies i.e. Amazon, Apple, Spotify, Uber, Hulu, and Spectrum.
2. **Preprocessing:** A number of iterations of cleaning the text of the tweets to model only essential topics out of the data. Lowering of text, removing user ids, URLs, HTML tags, emojis, and emoticons.
3. **Topic Modeling:** Applying the topic modeling algorithms (LSA, LDA, and BERTopic) to each of the company datasets in order to identify the most common issues being discussed.
4. **Topics Visualization:** The modeling approaches are followed by visualization of topic models using UMAP (Uniform Manifold Approximation and Projection), pyLDAvis, and BERTopic InterTopic Modeling Visualization.
5. **Unsupervised Sentiment Analysis:** Performing sentiment analysis using TextBlob, Vader, and TweetNLP in order to determine the overall sentiment of the tweets towards the software issues being discussed.
6. **Topic Modeling on Positive, Negative, and Neutral classes:** Applying the topic modeling algorithms again to the positive, negative, and neutral tweets identified by the sentiment analysis in order to compare the performance of the algorithms on different types of tweets as well as to know the topics from positive, negative, and neutral tweets.

Our approach focuses on retrieving mostly 20 topics from the user feedback which can summarize what people say about the software products. These topics are compared and contrasted using 3 modeling algorithms: LSA, LDA, and BERTopic. Furthermore, sentiment analysis is applied to the dataset to figure out the positive, negative, and neutral topics from it. We use coherence values to estimate an ideal number of topics to get better semantic relations in tweets. Rest, we talk about our results in the section Results and Discussion below.

IV. IMPLEMENTATION

Data filtering is the first step done for the top 6 companies by tweet frequency found in Kaggle's Customer Support on Twitter dataset [11] and separate datasets are created for each. Following this, preprocessing is done to ensure only quality data is analyzed.

Unsupervised topic modeling techniques including LSA, LDA, and BERTopic form the major chunk of this technical work. LSA is used to identify the underlying patterns in a dataset. Its implementation is done using SVD vectorization to form a document term matrix used for evaluating models from the dataset in order to examine connections between a collection of documents and the terms they include. Whereas for LDA, gensim library's LdaMulticore is used to represent each document as a mixture of topics, and each topic as a mixture of words. As a result, LDA can discover the associations between various documents and themes as well as the most crucial words and topics in the document collection. These topics are visualized using the pyLDAvis tool. Finally, BERTopic, a variant of BERT is used as a python package to gain high-level interpretations of the text that reflect the fundamental topics and structure of the information using self-supervised learning techniques.

With regard to sentiment analysis, unsupervised machine learning libraries are used. Vader, Tweet NLP Roberta, and TextBlob are the 3 tools used to compare and contrast results. Additionally, topic modeling is applied to the positive, negative, and neutral tweets to extract more insights about the nature of tweets and classify themes and topics under the nature of sentiments - Positive, Negative, and Neutral.

Our study reveals interesting and informative topic outputs extracted from tweets. The topics clearly insinuate the software issues, feature requests, and reviews of a software company. Figure 1. reveals the topics related to various aspects of the music streaming service Spotify, such as premium accounts, payment and billing issues, availability of music and content, and problems with the app or account. These topics cover a range of themes - premium subscriptions, Facebook account issues, family plans, iPhone updates, playlist creation, availability in different countries, student discounts, and album releases. Some of the topics also include words related to customer service, such as "help," "still," and "please," which suggests that these topics may be related to customer inquiries or complaints. Sentiment analysis further helps us categorize topics on the basis of the sentiments of tweets i.e. positive, negative, and neutral. These can be beneficial in helping a software business and developers effectively understand the issues troubling customers.

```
Topic: 0 Words: 0.024*"spotify" + 0.016*"get" + 0.013*"premium" + 0.010*"care" + 0.010*"take" + 0.009*"mone
Topic: 1 Words: 0.030*"email" + 0.019*"account" + 0.015*"still" + 0.012*"password" + 0.012*"changed" + 0.01
Topic: 2 Words: 0.021*"premium" + 0.019*"account" + 0.017*"student" + 0.015*"family" + 0.012*"help" + 0.012
Topic: 3 Words: 0.012*"screen" + 0.011*"family" + 0.011*"fuck" + 0.011*"good" + 0.010*"spotify" + 0.008*"th
Topic: 4 Words: 0.026*"reputation" + 0.014*"change" + 0.013*"spotify" + 0.012*"payment" + 0.010*"account" +
Topic: 5 Words: 0.090*"thank" + 0.014*"much" + 0.013*"nope" + 0.011*"wtf" + 0.011*"perfect" + 0.010*"spotif
Topic: 6 Words: 0.022*"sent" + 0.015*"dm" + 0.010*"avail" + 0.010*"spotify" + 0.007*"right" + 0.007*"thanks
Topic: 7 Words: 0.019*"available" + 0.015*"please" + 0.012*"songs" + 0.012*"india" + 0.011*"spotify" + 0.01
Topic: 8 Words: 0.010*"spotify" + 0.010*"wait" + 0.008*"im" + 0.008*"together" + 0.007*"get" + 0.007*"hope"
Topic: 9 Words: 0.019*"family" + 0.015*"account" + 0.014*"premium" + 0.012*"please" + 0.011*"plan" + 0.010*
Topic: 10 Words: 0.017*"hulu" + 0.016*"done" + 0.014*"add" + 0.012*"account" + 0.012*"spotify" + 0.011*"alr
Topic: 11 Words: 0.013*"isnt" + 0.012*"spotify" + 0.010*"oh" + 0.009*"yall" + 0.008*"high" + 0.007*"thanks"
Topic: 12 Words: 0.069*"thanks" + 0.009*"usa" + 0.008*"cant" + 0.008*"us" + 0.007*"wow" + 0.007*"spotify" +
Topic: 13 Words: 0.021*"yes" + 0.016*"account" + 0.013*"dm" + 0.012*"please" + 0.011*"im" + 0.010*"hacked"
Topic: 14 Words: 0.018*"fix" + 0.014*"ok" + 0.013*"work" + 0.012*"app" + 0.012*"pls" + 0.009*"thanks" + 0.0
Topic: 15 Words: 0.013*"spotify" + 0.012*"working" + 0.010*"album" + 0.008*"hi" + 0.008*"put" + 0.008*"plea
Topic: 16 Words: 0.012*"spotify" + 0.011*"account" + 0.009*"please" + 0.008*"premium" + 0.008*"connected" +
Topic: 17 Words: 0.038*"help" + 0.021*"need" + 0.014*"account" + 0.011*"please" + 0.010*"spotify" + 0.008*"
Topic: 18 Words: 0.049*"iphone" + 0.034*"x" + 0.033*"" + 0.030*"ios" + 0.029*"version" + 0.020*"app" + 0.01
Topic: 19 Words: 0.015*"songs" + 0.015*"playlist" + 0.014*"song" + 0.012*"check" + 0.011*"discover" + 0.010
```

Fig 1. LDA topics for Spotify dataset

V. EMPIRICAL EVALUATION

In this section, we present our approach's evaluation of Customer Support on Twitter dataset. The structure of this review process included defining research questions, conducting a search strategy, data extraction, data analysis, and experimenting with various methods.

A. *Research Questions*

Our evaluation aims to answer the following research questions:

- **RQ1:** How well do unsupervised topic analysis algorithms predict the software issues from the user feedback on Twitter?
- **RQ2:** What are the relevant characteristics of tweets in terms of sentiments?

B. *Dataset Description*

The Customer Support on Twitter dataset provides a sizable corpus of talks between customers and customer service representatives on Twitter, primarily in modern English. Each entry in the CSV collection represents a tweet. Every discussion featured has at least one request from a consumer and at least one answer from a corporation. The inbound field can be used to determine which user IDs are company user IDs. The customer support data set can be found on Kaggle Consisting of tweet_id, author_id, inbound, created_at, text, response_tweet_id, and in _response_to_tweet_id attributes. This huge data set of roughly around 3 million tweets consists of tweets from more than 20 brands. For simplicity top 6 Brands by volume which are AppleSupport, AmazonHelp, Uber_Support, SpotifyCares, hulu_support, and Ask_Spectrum.

The tweets written by six software companies were extracted from a dataset and filtered to include only those written in English and sent by customers to the companies. The tweets were then cleaned by removing URLs, punctuation, stopwords, emojis, emoticons, HTML tags, company handles, tweet/user IDs, standalone numbers, and spaces.

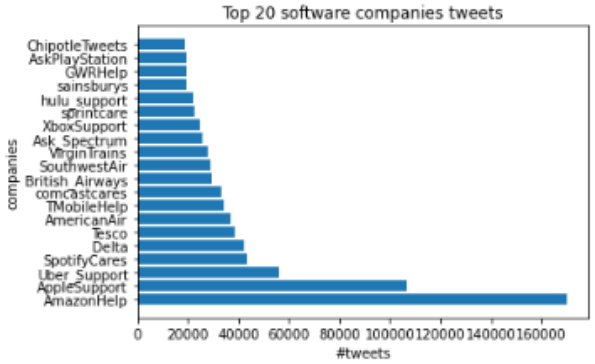

Fig 2. Top 20 software companies by tweets volume

C. *Analysis Procedure and Metrics*

We answer our first research question through the implementation of three unsupervised topic analysis algorithms - LSA, LDA, and BERTopic. These three are used to evaluate and identify 20 topics. The first 20 topics are compared and contrasted using different visualization tools like UMAP, and pyLDAvis. The LDA model is also evaluated using the Coherence framework (c_v, u_mass scores) [13].

- We set max_features = 1000, n_topics = 20 due to limited processing capacity. Topic modeling algorithms take an excessive amount of time to train based on the processing capacity of the system.
- Tf-idf is preferred over bag over words because of its characteristic of weighing words on the basis of their frequency.
- UMAP is used to simplify high-dimensional data while retaining as much of the data's structure as feasible. In this case, it uses n_neighbors = 100, and min_dis = 0.5.
- pyLDAvis presents the relationships between documents. The stronger clusters imply more coherence scores e.g. in the case of Amazon, the documents are strongly related that's why the coherence score is highest i.e. 0.4.
- BERTopic intertopic distance map is similar to the pyLDAvis tool.

TABLE 1.
COHERENCE SCORES

Company	c_v	u_mass
Amazon	0.40	-5.77
Spotify	0.31	-5.37
Uber	0.30	-3.95
Apple	0.30	-5.20
Hulu	0.26	-4.30
Spectrum	0.25	-5.74

Further, we use three different algorithms for getting the sentiments of the tweets. We have used Vader, TextBlob and a library called TextNLP which is a free library pre-trained on the tweets dataset. We supply the preprocessed tweet to these sentiment detection classes and it gives us a value between -1 to 1 detecting the sentiment of the tweet. The sentiment analysis model is further used for topic detection using LSA. The comparative analysis of topics using distinct sentiment models answers our second research question. We categorize topics on the basis of the sentiments of the tweets.

D. *Results and Discussion*

Below, we present and discuss our results, organized by RQ1 and RQ2:

1) **RQ1:** The topics identified by LSA, LDA, and BERTopic seem to be different. It seems that the LSA and LDA algorithms identify similar topics, while the BERTopic algorithm identifies a wider range of topics. BERTopic's combination of self-supervised learning and supervised learning strategy yield amazing topics consisting of proper context being displayed. Visualizations for these three unsupervised learning techniques are found in Figure 3, Figure 4, and Figure 5. All these visualizations are for the Hulu software-related tweets dataset.

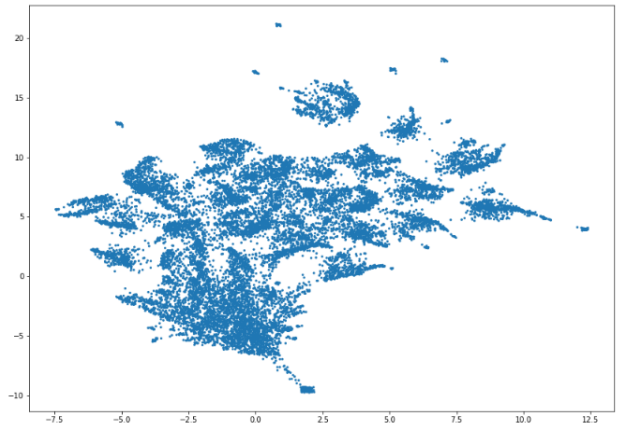

Fig 3. UMAP for Hulu's LSA model

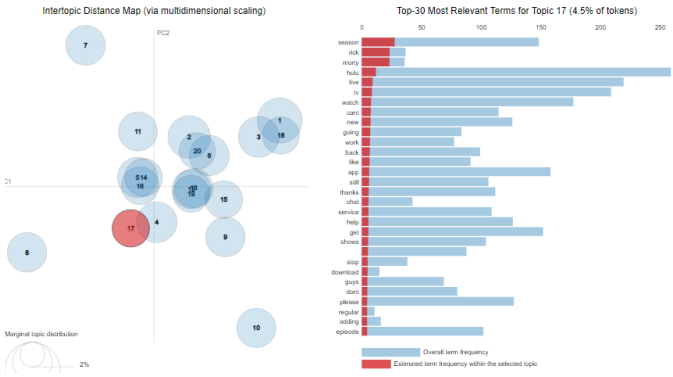

Fig 4. pyLDAvis for Hulu's LDA model

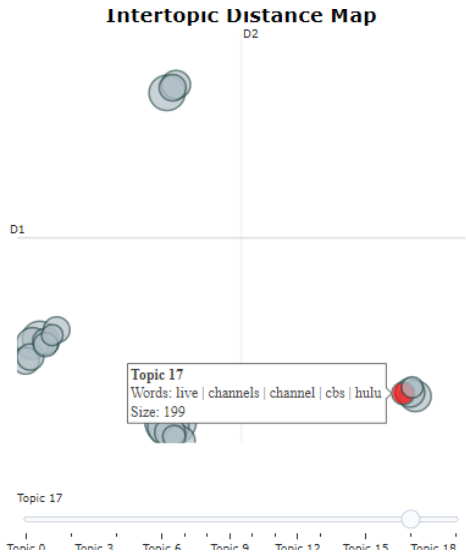

Fig 5. Hulu's BERTopic Intertopic Distance Map

Availability of new episodes or seasons of certain TV shows (such as "Rick and Morty"), problems with the app or service (such as buffering or error messages), and issues with commercials or ads. Other topics related to using the Hulu app on different devices, such as Apple and Roku devices, and general questions about how to use the app or access content are summarized through the LSA algorithm. The LDA algorithm identifies topics related to streaming services (such as Hulu), television shows and movies, errors and issues with streaming, and the use of various devices (such as Roku and Xbox) to access streaming content. Other topics include missing episodes, ads, and the availability of new seasons. Whereas BERTopic outputs issues of commercial breaks and ads, the availability of seasons of TV shows, errors and issues with streaming, buffering and other issues with live streaming, baseball games, and other sports events, thanks, and other expressions of gratitude, log in and account issues, and TV shows and movies.

TABLE 2.
HULU TWEET'S TOPIC MODELING

Topic Modeling	Relevant Topics Extracted
LSA	Issues: ads, buffering, account, how to use Hulu, missing episodes
LDA	Issues: availability, service, app errors, ads, missing episodes, help for support
BERTopic	Issues: ads issues, playback, error messages, buffering, live TV, streaming issues Greetings: thanks, working

A summary of the main themes generated by LSA, LDA and BERTopic is provided in Table 2. Results in the case of LDA performed on the Hulu dataset present a very vague description of the topic modeling output. Low Coherence (c_v) score i.e. 0.26, categorically indicates the poor quality of learned topics. A similar approach is followed for the rest 5 datasets i.e. Amazon, Spectrum, Apple, Uber, and Spotify. The respective coherent scores using c_v and u_mass can be found in Table 1 where Amazon gets the highest c_v score whereas Spectrum with least c_v score. A similar trend is also seen in the pyLDAvis plot where Amazon's topics seem much more coherent and tightly bound. Intuitively, Amazon has a large dataset i.e. 103,337 data points whereas Spectrum has the least data points i.e. 19,696. This implies that more training with more data points yields a good c_v score.

Our evaluation also counts for implementing a function to calculate c_v scores for LDA models from [5, 40] with

interval = 5. This analysis is helpful in getting an optimal count of the number of topics that yield a high coherence c_v score. Figure 5 presents a graph plot between the number of topics and the c_v coherence score. The graph shows that choosing n_topics > 25 can result in a good coherent set of topics that might be more interpretable.

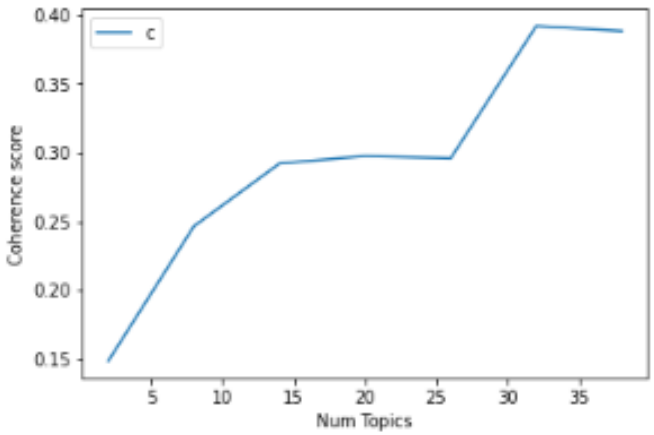

Fig 5. Performance of LDA wrt to the number of topics

Fig 5 represents that n_topics=30 is an ideal choice.

The above observations answer our first question that BERTopic performs good than LDA and LSA, whereas LDA works better than LSA. LDA works well if n_topics is high.

2) RQ2: In the second part of our results where we did sentiment analysis, we discuss the results for it below here. Overall among the three sentiment analysis models, judging from the topics and n-grams which we got, TweetNLP Roberta has performed the best. In the section below, we discuss and report here just the top 5 topics (categorized by models) for the dataset Apple. For a detailed view of the output and results of each dataset (unigram, bigram, trigram, and top 20 topics for each dataset) please visit the GitHub link: https://github.com/AnOnYmOuS219/Topic-Analysis-of-Software-Tweets

	Vader	TextBlob	TweetNLP Roberta
0	help yes iphone thanks phone ios update	update thanks new ios iphone phone battery	yes thank iphone thanks ios plus updated
1	yes restarted times tried happens version started	thanks worked reply fixed ok response great	thank thanks worked ios fixed iphone update
2	thanks ios update fix thank phone iphone	latest ios battery iphone life version fast	thanks iphone ios worked update phone help
3	thank worked fix response fixed appreciate ios	update latest phone version thanks software ru...	iphone plus ios update new 6s battery
4	thanks thank help worked yes fixed reply	battery new update life fast draining drains	plus iphone thanks 6s thank 7s yes

	Vader	TextBlob	TweetNLP Roberta
0	phone update ios iphone fix battery shit	phone update ios fix iphone shit battery	phone update ios iphone fix battery new
1	shit fix wtf question mark glitch fucking	shit fix wtf question mark fucking box	fix shit glitch question mark fucking damn
2	wtf phone going fuck wrong like yo	wtf phone going wrong like went apple	phone shit update wtf new keeps fuck
3	problem wtf iphone battery fix ios life	phone fuck update fucking new fucked slow	battery update phone life ios new draining
4	problem phone apple fuck im freezing app	battery update life phone sucks ios fucked	update new iphone shit latest fix software

	Vader	TextBlob	TweetNLP Roberta
0	ios iphone 6s plus update phone fix	ios iphone fix 6s phone update help	ios iphone 6s update plus phone help
1	iphone 6s plus 5s check se health	fix help phone update need apple glitch	help fix need iphone phone apple update
2	fix phone update iphone new issue battery	yes help tried updated times iphone thank	fix ios phone update issue help going
3	phone update new updated battery apple latest	help iphone need apple hi 6s plus	fix iphone phone 6s update issue battery
4	1103 phone iphone ios updated charging started	ios help phone updated yes update need	phone update apple new updated app time

Fig. 6. Positive, Negative, Neutral (Top to Bottom) - Apple

While we got the detailed topic analysis in the last section, here we look at different datasets. In Figure 6, the first Image we see positive tweets topics, among these topics, all three algorithms are able to identify tweets, most of the tweets among which have topics where they are thanking the official support channel back or talking about if the update was good enough. But one thing to note here is that TweetNLP Roberta, which is a library specifically trained for Twitter, is performing a little better in knowing the crux of the real sentiment of Twitter users. Words like 'drain', 'glitch', 'shit', and 'freeze' appear in the top 20 topics of Vader and textBlob positive topics while tweetNLP performs better and no such topics are identified as positive topics. It is easier for the model to judge negative words and therefore the topics for negative tweets provide a better judgment to the company for identifying software-related issues. Looking at the neutral tweets most of them are just either thanking them or asking for help without referring to the issue, sometimes positive tweets are identified as neutral but that inherently does not hurt our use case here.

TABLE 3.
ISSUES FROM NEGATIVE TWEETS

Company Name	**Most common issues found from negative tweets**
Apple	● Phone updates and issues with iOS ● Problems with battery life ● Glitches and other technical issues with the phone ● Freezing or crashing apps ● Slow performance ● Problems with the screen, such as freezing or turning black ● Annoying features or issues with the phone or iOS ● Difficulty with music or app functionality
Amazon	● Poor service and delivery experience ● Product issues, including receiving the wrong item or a defective product ● Canceled orders or issues with canceling orders ● Difficulties with the refund process ● Account issues, including being charged incorrectly or having difficulty accessing the account ● Shipping delays or issues with shipping ● Difficulty reaching customer service or receiving inadequate support ● Problems with Amazon Prime membership or benefits ● Fraud or scam issues, including receiving fake or counterfeit products
Spotify	● Problems with account billing and premium subscriptions, including being charged when not intended or not receiving discounts ● Hacked or compromised accounts ● Issues with the app, such as error messages or difficulty logging in ● Problems with the music or playlist, such as wrong songs or missing albums ● Difficulty with offline listening or paying for premium services ● Long wait times or other issues with customer service
Uber	● Problems with charges and billing, including being charged for canceled rides or having issues with refunds ● Issues with customer service and support, including difficulty getting help or long wait times ● Problems with the app or account, including being unable to log in or access certain features ● Issues with drivers, including canceled rides or lost or wrong orders ● Problems with food orders from Uber Eats, including missing or incorrect items ● Negative experiences with the service or drivers, including bad customer service or rude behavior

Hulu	● Technical issues, including error messages and difficulty streaming or loading content ● Problems with the app or interface, including issues with ads or new design changes ● Issues with billing and account management, such as being charged or having trouble canceling the service ● Missing or incorrect episodes or seasons of TV shows ● Negative experiences with the service, including buffering and poor customer service
Spectrum	● Outages or service disruptions in specific areas or zip codes ● Problems with cable and internet service, including slow speeds or issues with the app or channels ● Customer service issues, including difficulty getting help or resolving problems ● Billing and payment issues, such as incorrect charges or difficulty paying ● Technical issues, such as problems with the WiFi or TV channels

We can find the problems that can help businesses improve their software from negative tweets. Be aware that the subjects suggested by the model occasionally do not make sense enough, but if we look at the top 20 topics and try to identify the fundamental problem, we can see that the majority of the topics were complaints. It would make sense and might aid businesses in locating and resolving software-related problems. The subjects that can be inferred and comprehended from the negative tweet topics produced by the model are listed in the following table. There may be several internal fixes that need to be made at the software level as well, but all of the items listed here are the general concerns detected by software firms. We list all of the flaws found in the unfavorable tweets of this particular dataset.

E. *Threats to Validity*

Selection bias, which happens when the sample of tweets obtained is not representative of the greater population of tweets relating to customer assistance, is one possible threat to the validity of this study. Because only a few organizations were examined, the study's findings might not apply to other businesses or to a larger sample of tweets. This could be caused by a number of things, like the distinctive qualities of the companies chosen or the distinctive problems and opinions being voiced by customers of those companies. Future research should take a broader and more varied sample of tweets and businesses into account in order to lessen this threat.

Sampling error, which happens when the sample of tweets taken is not typical of the greater population of tweets, is another factor that could compromise the study's validity. This might be because of a number of things, including the way the tweets were gathered or the precise time frame in which they were gathered. The study's findings might not be an accurate reflection of the genuine attitudes and subjects being explored if the sample is not representative of the greater population. It would be crucial to adopt a random or stratified sampling procedure to make sure that the sample is representative of the larger population in order to lessen this threat.

Similar to sampling, the possibility of social media bias exists. Twitter and other social media sites could be biased in a number of ways, which could skew the findings of a study. For instance, some demographics may be more or less likely to utilize a specific social media site, which may have an impact on how representative the sample of tweets that was gathered is. Additionally, biases in the data, such as the use of specific words or phrases that are more likely to be used by particular groups, may have an impact on the algorithms and models used to evaluate the tweets. This can reduce the generalizability of the findings to other datasets or situations and compromise the accuracy of the results.

Lastly, the topic modeling techniques considered in this study do not consider a baseline. The topics thus predicted are not verifiable. Furthermore, being unsupervised approaches, there's no evaluation metric to compare and contrast the results of the three topic modelings empirically. Analytically, most of the extracted topics fall in the right set as the frequency of terms occurring i.e. probabilistic model is the basis of topic generation. Our research so far could not take us to any point where we could explore and experiment with the evaluation metric for the topic modeling approaches. No evaluation metric exists for verifying and evaluating the performance of these unsupervised approaches. The coherence model, as a matter of fact, exists only for LDA and that too is just a measure of coherence between the topics.

VI. LESSONS LEARNED

Data preprocessing is critical to the effectiveness of a study. To clean the data and reduce noise in our study, we used a number of preprocessing approaches, such as

reducing the text and deleting user IDs, URLs, HTML elements, emojis, and emoticons. Preprocessing is a crucial phase in the data analysis process because it ensures that the data is consistent and useful, and it can increase the accuracy of the results. Other studies have stressed the significance of preprocessing, particularly in the context of text data, because it can assist to minimise data complexity and increase the effectiveness of analytic tools.

It is critical to examine the study's data and model limitations and potential biases. In our study, we used a dataset of customer support tweets from a small selection of organizations, which may not be typical of the entire population of customer support tweets. Furthermore, the performance of the LSA, LDA, and BERT models utilized in the study may be influenced by a variety of factors such as the parameters used, the quality of the training data, and the models' underlying assumptions. Other research has emphasized the significance of carefully assessing the constraints and biases of the data and models employed since they might impact the accuracy and generalizability of the conclusions.

Using several approaches and strategies can give a more complete grasp of the subjects and attitudes under consideration. We used a mix of LSA, LDA, and BERT in our work to extract themes and detect sentiments in tweets. We were able to achieve a more nuanced knowledge of the data by employing numerous procedures and confirming the results obtained with one method with the results obtained with another. Other studies have also underlined the need of employing a variety of methodologies and approaches in order to achieve a more full knowledge of the data and to validate the results gained in one way with the results obtained in another.

It is critical to thoroughly analyze the research questions and hypotheses being investigated, as well as to effectively describe the study's results and consequences. In our study, we concentrated on recognizing software flaws in tweets and extracting customer support attitudes. We were able to lead the analysis and interpret the results in the context of our study by explicitly outlining our research questions and hypotheses. Furthermore, it is critical to effectively describe the study's results and consequences, since this can serve to guide future research and practice.

Another thing that we learned from this study's sentiment analysis is that tweet sentiment is important in properly extracting themes from tweets. The results, in particular, demonstrated that the TweetNLP tweet sentiment analysis model could more effectively identify the sentiment of negative tweets and extract important subjects. This shows that taking into account the tone of tweets may be especially valuable in detecting typical difficulties and concerns made by consumers, particularly in the context of customer assistance. Companies may be able to more readily identify and handle frequent issues and concerns mentioned by their consumers if they focus on unfavorable tweets. Furthermore, this finding emphasizes the need to properly select and assess sentiment analysis models, as model performance varies based on the precise feelings indicated in the data. It may be especially advantageous to utilize models that have been expressly trained for Twitter or other social media sites, as these models may be more accurate at distinguishing tweet emotions.

VII. CONCLUSION

In this study, we aim to detect software issues from tweets related to customer support for the companies Apple, Amazon, Hulu, Spotify, Spectrum, and Uber. To accomplish this goal, we applied unsupervised topic analysis algorithms and sentiment analysis models to the tweets. Our results showed that these techniques can be useful for identifying common topics and sentiments expressed by users in their tweets and gaining a better understanding of the issues and concerns raised by customers. We found that negative tweets were more likely to contain software-related issues, while positive tweets were more likely to contain topics related to thanking the official support channel or discussing the quality of updates. The neutral tweets were more likely to contain topics related to asking for help or thanking the company without referring to any specific issues.

Conclusively, this study demonstrates the utility of unsupervised topic analysis algorithms and sentiment analysis models for detecting software issues from tweets related to customer support. These techniques can provide valuable insights into the issues and concerns raised by users and can help companies to better understand and address these issues.

Further research could explore the use of these techniques in other contexts or with different types of data to expand upon the findings of this study. There are several potential directions for future work that could build upon the findings of this study. One possibility would be to extend the analysis to other social media platforms or other types of customer feedback data, such as reviews or forum posts. This could provide a more comprehensive understanding of the issues and concerns raised by users and could help to validate the findings of this study. Additionally, further research could explore the use of different unsupervised topic analysis algorithms or sentiment analysis models to examine the sensitivity of the results to these choices. This could help to identify the most effective techniques for detecting software issues from customer feedback data and could provide insights into the strengths and limitations of different approaches.

Finally, it would be interesting to investigate the use of supervised learning techniques, such as classification algorithms, to automatically classify tweets as positive, negative, or neutral and to predict the likelihood of a tweet containing a software issue. This could provide a more efficient and automated way to identify and address software issues from customer feedback data.

ACKNOWLEDGEMENTS

We are grateful to Prof. Mehrdad Sabetezdah at the University of Ottawa for facilitating and supporting this research.

REFERENCES

[1] E. Guzman, M. Ibrahim, and M. Glinz, "Prioritizing User Feedback from Twitter: A Survey Report," 2017 IEEE/ACM 4th International Workshop on CrowdSourcing in Software Engineering (CSI-SE), 2017.

[2] Li, X., Liu, H., & Zhang, J. (2014). Identifying Software Issues from Social Media Data. In International Conference on Software Engineering and Service Science (pp. 95-104). Springer, Berlin, Heidelberg.

[3] Silva, S. R. P., Oliveira, A. C. L., & Oliveira, M. L. (2017). Identifying Software Issues from Customer Reviews. In International Conference on Software Engineering and Service Science (pp. 63-72). Springer, Berlin, Heidelberg.

[4] Alhaj, M. A., Raza, A., & Gani, A. (2018). Data Mining for Software Issue Detection in Social Media. In International Conference on Advanced Computing, Communications and Informatics (pp. 835-840). Springer, Singapore.

[5] Islam, M. S., Raza, A., & Gani, A. (2017). Software Issue Detection in Social Media Using Data Mining Techniques. In International Conference on Computer Science, Engineering and Applications (pp. 679-684). Springer, Singapore.

[6] Alhaj, M. A., Raza, A., & Gani, A. (2019). Identifying Software Issues from User Reviews Using Sentiment Analysis. In International Conference on Advanced Computing, Communications and Informatics (pp. 791-796). Springer, Singapore.

[7] Jha, N., Mahmoud, A. Mining non-functional requirements from App store reviews. Empir Software Eng 24, 3659–3695 (2019).

[8] Huang S-H, Tsao S-F, Chen H, Bin Noon G, Li L, Yang Y, and Butt ZA (2022) Topic Modelling and Sentiment Analysis of Tweets Related to Freedom Convoy 2022 in Canada. Int J Public Health 67:1605241.

[9] E. Guzman, R. Alkadhi and N. Seyff, "A Needle in a Haystack: What Do Twitter Users Say about Software?," 2016 IEEE 24th International Requirements Engineering Conference (RE), 2016.

[10] Kasturi Bhattacharjee, Rashmi Gangadharaiah, Kathleen McKeown, and Dan Roth. 2022. What Do Users Care About? Detecting Actionable Insights from User Feedback. In Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Industry Track.

[11] Thought Vector and Stuart AxelBrooke, 2017, “Customer Support on Twitter”, Kaggle, doi: shorturl.at/inuY2

[12] Geetha, S., & Kumar, K. V. (2018, December 12). Tweet Analysis Based on Distinct Opinions of Social Media Users. Tweet Analysis Based on Distinct Opinion of Social Media Users | SpringerLink. Retrieved November 4, 2022

[13] Joshi, P. (2018, October 16). An NLP Approach to Mining Online Reviews using Topic Modeling. Analytics Vidhya.

[14] Egger, R., & Yu, J. (2022, May 6). A Topic Modelling Comparison Between LDA, NMF, Top2Vec, and BERTopic to Demystify Twitter Posts. PubMed Central (PMC).

Code Implementation: https://github.com/AnOnYmOuS219/Topic-Analysis-of-Software-Tweets
Data Source: https://www.kaggle.com/datasets/thoughtvector/customer-support-on-twitter