

Extracting Insights from Reviews using Cluster Analysis

Ayush Hans¹, Nihar Khera¹

¹National Institute of Technology, Kurukshetra, India

Abstract

The top operating organizations understand an essential role that customer feedback plays in the business industry. These businesses then consistently listen to the feedbacks of the consumers to stay ahead in the competition. Customer feedback gives crucial insights into the workings of the product, services, and what could be done within the company's domain to make the experiences of the consumers better. Customer's opinions help the companies ensure that the final product actually shall suffice their expectations, solve their problems and meet their needs.

Hence, the customer feedback is one of the most reliable and easy to get sources for tangible data that can also be used in making wise business decisions. The proposed approach provides a method to make effective use of this feedback and generate insights for the Product Team. Since it is not feasible to go through all reviews to find out what the customers are talking about, the reviews are clubbed together by Topic Modelling approach. The Business Team is presented with top keywords corresponding to each group of reviews which makes it easy for them to find out the actionable areas. The way the results are presented to the team guides them in the right direction so as to improve their products and services. A model is generated once the reviews have been labelled with topics. This is helpful to classify the new reviews which keep on coming from the customers' end. The Topic Modelling algorithm is again followed once the team has good number of new reviews which will further help in improving the model.

Keywords

Reviews, Natural Language Processing, Machine Learning, LDA, Topic Modelling, BERT

1. Introduction

The existing customers' reviews are not only helpful for the new customers to find the right product but they also serve as a means for the product teams to improve their products and services. In this era of digitization, organizations use customer reviews and other feedback information from various sources and generate insights out of those reviews. Machine Learning and Natural Language Processing both are used to process these wide varieties and a huge volume of reviews. Different approaches such as Topic Modeling, Text Clustering are used in Natural Language Processing for Customer Feedback Analysis.

Data Preprocessing: It is an important step to preprocess textual data before performing Natural Language Processing tasks. NLP involves text/data processing to convert the available data into more usable and convenient form. It helps to get rid of the redundant and irrelevant data present in the dataset and also plays a role in maintaining the standard of the text [1].

International Conference on Smart Systems and Advanced Computing (Syscom-2021), December 25–26, 2021

✉ ayushhans2011@gmail.com (A. Hans); nkniarkhera@gmail.com (N. Khera)



© 2021 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).


 CEUR Workshop Proceedings (CEUR-WS.org)



Figure 1: Flow Diagram of the model

Topic Modeling: Topic Modeling is used for finding different topics from documents (basically some form of textual data) without having any knowledge in advance.

LDA (Latent Dirichlet Allocation): This topic modeling approach makes use of each document as a different set of topics and every word is considered to be drawn from those topics. A good LDA Model involves tuning of hyperparameters such as word topic density, document topic density etc. In order to get good quality of topics, a suitable number of topics has to be selected which can be done by measuring the Topic Coherence, which measures the degree of semantic similarity between the words which scored highest in the topic[2].

Text Classification: Text Classification is a good choice to get familiar with textual data processing. It finds a lot of interesting applications in daily life. There have been a significant amount of researches in this field. One of such research is Bert Model. BERT stands for “Bidirectional Encoder Representations from Transformers”.

The remainder of this paper is structured as follows: Section 2 provides the proposed approach followed throughout the paper. In Section 3, we present the related works. In Section 4, we present the implementation of the proposed method. In Sections 5 and 6, we discuss the results, provide a conclusion and propose recommendations for some future work.

2. Proposed approach

The proposed approach basically combines two aspects of Machine Learning algorithms- Clustering and Classification.

Clustering helps to avoid the manual task of labeling the product reviews by dividing them into topics or clusters. The labels then serve as the basis of classifying the new reviews. Topic Modeling finds a theme across reviews and discovers hidden topics. It can be interpreted as creating some buckets and putting each review into these buckets. First, the reviews are split

into positive or negative depending on the rating value given by the customer. Then LDA Topic Modeling is used to find themes across these two categories.

The output of Topic Modeling is visualized on a webpage that displays the top Bigrams (two words frequently occurring together) corresponding to each topic or cluster identified by the LDA Topic Modeling. This type of visualization is really helpful from the perspective of the Business or Product Team as they get a clear picture of what the customers are talking about in the reviews. The team also gets the list of actual customer reviews to read them as and when needed. It also displays an Inter-topic Distance Map which reflects the clusters formed where each cluster is represented in the form of a bubble. This is very helpful for Data Scientists for analysis of the topics or clusters formed. The webpage shows a list of the most relevant words corresponding to each topic along with their frequency in the selected topic and overall frequency.

Now, we have the clusters[3][4], but new reviews still keep on coming up from the customers. These reviews are classified into the clusters formed with the help of a classification model which is built using the topics or clusters from the topic modeling algorithm.

The topic modeling algorithm can again be followed after a specified time (for example, after two or three months) when the Product Team has quite a reasonable quantity of new reviews. This will in result improve the quality of new topics or clusters formed.

3. Related works

Many researches have been done in the text summarizations and terminology identifications [5]. This technique requires designing templates by adequately identifying and extracting primary elements and significant facts in a document. Researchers still are working on the information extraction processes from texts. The main focus is on the machine learning and NLP methods for proper extraction or classification of entities and relations. Continuing on the same, the other area of research in this field is the opinion and review extraction from online web pages and the opinion summarizations based on product features with the help of edge[6] and cloud computing [7][8]. The central problem with the existing studies on the work of reviews is that they consider all the reviews with the same significance, which may not give relevant and accurate results. That is why the classification of reviews based on importance is a significant task. Hiremath proposed a system to automatically assess the review's quality using quartile measure and identify a customer review as Most Significant review, More Significant review, Significant review, and Insignificant review.

Other approaches include Topic Modeling algorithms like Latent Dirichlet Allocation, Latent Semantic Analysis etc. which enables us to discover topics from set of documents. In Topic modeling using LDA, different topic groups are created. It is the role of the researcher to decide the number of groups in the final output. Since there is no prior knowledge about what is the best number of groups, we generate models with different numbers of groups and then analyze and compare different topic modeling, and then the decision is made to select the topic model which is most meaningful and sensible out of all the models generated with different hyperparameters.

Topic Modeling is an approach which is useful in finding out the themes across the data, hence

this is quite effective when we are dealing with customer reviews. Each review is assigned one of the themes to which it belongs with highest proportion making it easy for the businesses to figure out the difficulties being faced by the customers in regard to their products and services.

4. Implementation

4.1. NLP Preprocessing

- Contractions Expansion: Contractions are quite common in English Language. The contractions of words are created by removing specific letters and sounds. This step expands each and every contraction to its original form to maintain the standard of the text[9].

```
# expanding contractions
def expand_contractions(text):
    word = []
    for words in text.split():
        words.append(contractions.fix(word))
    return ''.join(words)
```

- Removal of URLs: There is a chance that the review may have some URL in it. Therefore, we need to remove it to continue with further processing.

```
# removing url's
def remove_urls(text):
    pattern = re.compile(r'https?://\S+|www\.\S+')
    return pattern.sub(r'',text)
```

- Removal of HTML Tags: This step is useful when the reviews have been extracted from a website because there is a chance that some HTML specific code has become a part of the review during scrapping.

```
# removing html tags
def remove_html_tags(text):
    pattern = re.compile('<.*?>')
    return pattern.sub(r'',text)
```

- Lower Casing: Lower Casing is a text preprocessing technique. It is done to convert the text into same casing format, so that the words are not considered as different.

```
# lower casing  
review_df[processed_reviews] = review_df[processed_reviews].str.lower()  
review_df.head()
```

- Removal of Punctuation: This step is performed to maintain the standard of the text. The list of punctuations to exclude should be chosen after taking into consideration the task for which preprocessing is done.

```
# removing punctuation  
punctuation = string.punctuation  
def remove_punctuation(text):  
    return text.translate(str.maketrans('', '', punctuation))
```

- Tokenization: This text preprocessing step splits textual strings into smaller pieces which are referred to as “tokens”. It involves splitting textual data into sentences which are then split into words. This is a necessary step in almost all of the textual data processing tasks. Tokenization is also known as Text Segmentation.

```
# tokenization  
def tokenization(text):  
    return nltk.word_tokenize(text)
```

- Lemmatization: This is one of the most important NLP preprocessing steps. Lemmatization aims at reducing a word to its base or dictionary form, which is called as the “lemma”. It really transforms words to their true root form, instead of just chopping them. For example, the words “playing”, “plays”, “played” are mapped to “play”. It can be done with the help of Python “nltk” package and makes use of a dictionary such as “WordNet” for producing the mappings. Lemmatization plays a significant role in Natural Language Processing and Artificial Intelligence tasks. In languages other than English, lemmatization can be quite complicated.

```

# Lemmatization
def get_tag(tag):
    if tag.startswith('N') or tag.startswith('J'):
        return wordnet.NOUN
    elif tag.startswith('V'):
        return wordnet.VERB
    elif tag.startswith('R'):
        return wordnet.ADV
    else:
        return wordnet.NOUN

def lemmatization(tokens):
    pos_tags = nltk.pos_tag(tokens)
    wordnet_lemmatizer = WordNetLemmatizer()
    doc_words = [wordnet_lemmatizer.lemmatize(word, pos=get_tag(tag)) for
word, tag in pos_tags]
    return doc_words

```

4.2. Topic Modelling

- LDA (Latent Dirichlet Allocation): LDA (Latent Dirichlet Allocation): Topic Modeling is an approach that is used to find themes across the reviews and discover hidden topics. It is based on extracting a certain number of groups consisting of specific words from the reviews. These groups represent the topics that are useful from the perspective of the Business or Product Team to find out what the customers are talking about in the reviews. LDA (Latent Dirichlet Allocation) is one of the most popular methods of Topic Modeling. LDA takes two hyperparameters into consideration, the “alpha parameter” and the “beta parameter”. The “alpha parameter” controls the mixture of topics for any given document. If it is low, the documents will have less of a mixture of topics and if it is high, the documents will have more of a mixture of topics. The “beta parameter” controls the distribution of words per topic. If it is low, the topics will likely have fewer words. If it is high, the topics will likely have more words. Another factor that LDA takes into account is K, the number of topics or groups to form.

```
#LDA model
ldamodel = LdaModel(corpus, id2word = dictionary,
                    num_topics = n_topics,
                    chunksize = 100,
                    minimum_probability = 0.001,
                    random_state = 100,
                    iterations = 50,
                    passes = passes)
```

- Topic Modeling using Nouns and Adjectives: The topics generated by LDA can be a mixture of nouns, verbs, adjectives, etc. The LDA algorithm treats all tokens equally with the same importance. When we are dealing with the reviews, removing all words except nouns and adjectives helps to improve the semantic coherence of the topics.

```
# topics modelling using nouns and adjectives
def nouns_and_adjs(series):
    # POS(part of speech) tagging
    pos_tags = nltk.pos_tag(series)
    noun_adjs = [word for (word, tag) in pos_tags if (tag=="NN" or
    tags=="NNS" or tag=="JJ")]
    return nouns_adjs
```

- Bigrams Formation: Bigrams refer to two words frequently occurring together in the text. Applying LDA Topic Modeling after taking into account bigrams (or in general, n-grams) helps to improve the quality of topic models. In Python, Gensim's Phrases model can build and implement the bigrams, trigrams, etc.

```
#forming bigrams
def bigrams(bigram_model, text):
    return [bigram_model[review_doc] for review_doc in text]
reviews_docs = list(reviews)
phrases = gensim.models.Phrases(review_docs, min_count = 15, threshold =20)
bigram_model = gensim.models.phrases.Phraser(phrases)
bigrams_list = bigrams(bigram_model, review_docs)
```


4.3. Evaluation of LDA Topic Modeling: Topic Coherence

The probabilistic topic models (such as LDA) are popular approaches for textual processing and analysis. They provide predictive and latent topic representation of the corpus. It is assumed that the latent space discovered by these models is generally meaningful and useful, and evaluating such assumptions is challenging due to its unsupervised training process. Topic Coherence is a method that can be used to evaluate the LDA topics. It is based on the concept of combining a number of measures into a framework to evaluate the coherence between topics that have been generated by the model. If a set of sentences or facts support each other, they are said to be coherent. Topic Coherence measures score of a single topic by measuring the degree of semantic similarity between high importance words in the topic. Higher the value of Topic Coherence for a model, better is the quality of topics formed by the model.

4.4. Visualization

Visualizing clusters makes it convenient for the Business or Product Team to evaluate, explore and interpret the results of Cluster Analysis[10]. It lists out the top Bigrams corresponding to each topic or cluster identified by the LDA Topic Modeling which gives the Product team a clear picture of what the customers are talking about in the reviews about their product. The webpage also displays the list of actual customer reviews for deep analysis. It has an Inter-topic Distance Map which is helpful for Data Scientists to evaluate the clusters formed. Hence, we have both unigrams and bigrams for each cluster, which is useful for the Product Team to find out the areas to focus upon to improve their product.

4.5. Classification of New Reviews

We've labeled the reviews after we've finished clustering. Each review now has a label that corresponds to the topic number to which it belongs. To perform text token processing, the BERT employs the Transformer encoder architecture. This processing is done in the full context of all tokens before and after it. Such models are pre-trained on a large corpus of text before being fine-tuned for specific NLP tasks.

BERT is an encoder stack of transformer architecture[11], which is an encoder-decoder network that makes use of self-attention on the encoder side and attention on the decoder side. BERT Models also have large feed forward networks, 768 hidden units in case of Base Bert and 1024 hidden units in case of Large Bert. During training process, the Bert model takes pairs of sentences and learns to predict if the second sentence is the subsequent sentence of the first sentence in the original text. 50 percent of the inputs are a pair in which the second sentence is the subsequent sentence in the original text. For the other 50 percent of the inputs, a random sentence from the corpus is chosen as the second sentence[12].

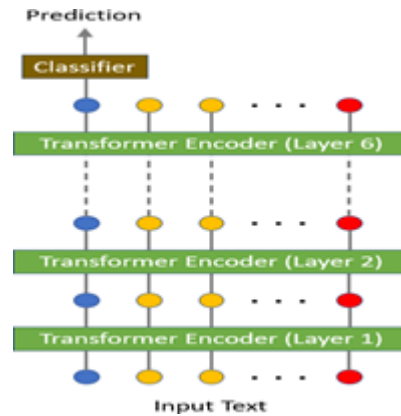


Figure 2: Using BERT for Classification

```
# bert
modelClass , tokenizerClass, pretrainedWeights = (DistilBertModel,
DistilBertTokenizer, 'distilbert-base-uncased')
tokenizer = tokenizerClass.from_pretrained(pretrainedWeights)
model = modelClass.from_pretrained(pretrainedWeights)
```

5. Results

Finally, after combining the two aspects of Machine Learning algorithms, Clustering and Classification, we visualize the insights to see if we can have some meaningful results from them.

6. Conclusion and Future plans

The customers' reviews are of utmost importance for any firm or organization. The organizations that look into the feedback given by the customers always excel in their domain. It is not possible to go through each and every piece of customer feedback manually. Clustering the reviews is a better way to get insights from them. Topic Modeling can be used to find themes across the reviews and discover hidden topics. LDA (Latent Dirichlet Allocation) is one of the most popular methods of Topic Modeling. It is a "generative probabilistic model". After applying the LDA model, we have the topic or cluster for each customer review to which it belongs with highest probability value. As a result, we have labelled reviews, each of which belongs to one of the topics or clusters. These clusters are visualized to present them to the Product Team in an easy to interpret and analyze form.

Once the organization has a significant number of fresh reviews, it may use the clustering technique to improve the quality of topics or clusters, since we know that the more data there



Figure 3: Cluster Reviews

is, the better the model performs. This approach is quite effective from the perspective of an organization and helps them to improve the quality of their products and services by making it easy to identify actionable areas.

Another improvement that could be made in the future is to incorporate sentences or embeddings from a model like Bert into the Topic Modeling technique. The vectors from the model and LDA can be combined with some weight or hyperparameter to improve the results.

Acknowledgments

We would like to thank our college, National Institute of Technology, Kurukshetra for giving us the platform to express ourselves. Also, we would like to thank our mentor Dr. B.B. Gupta, Asst. Professor, NIT Kurukshetra.

References

- [1] S. Kapadia, "Towards Data Science," 19 08 2019. [Online]. Available: <https://towardsdatascience.com/evaluate-topic-model-in-python-latent-dirichlet-allocation-lda-7d57484bb5d0>. [Accessed 07 02 2021].
- [2] Kaggle, "Clustering with Topic Modeling using LDA," Kaggle, 01 09 2020. [Online]. Available: <https://www.kaggle.com/panks03/clustering-with-topic-modeling-using-lda>. [Accessed 19 03 2021].

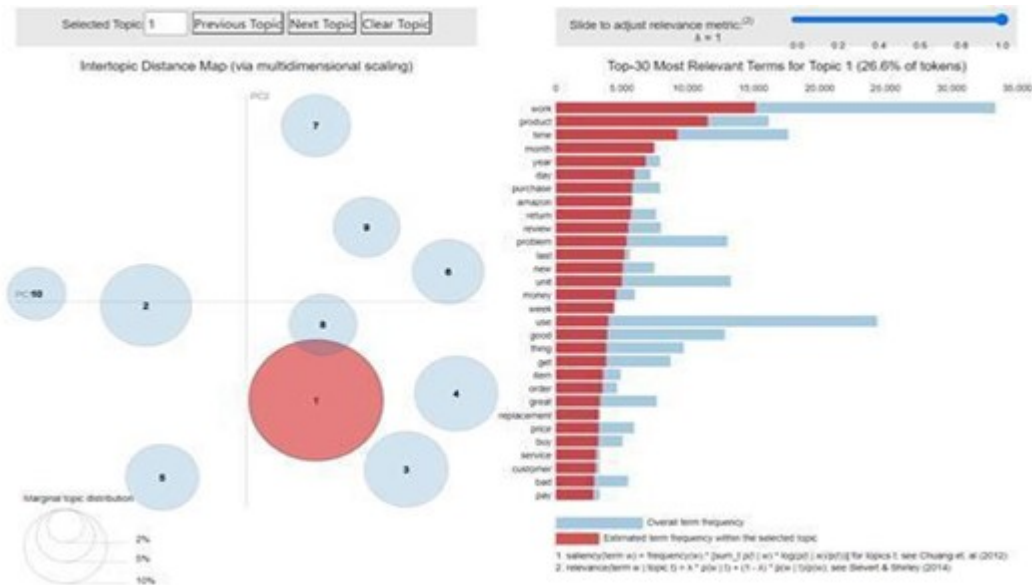


Figure 4: Classification results in form of clusters and relevance metrics

- [3] Shahabadi, M. S. E., Tabrizchi, H., Rafsanjani, M. K., Gupta, B. B., Palmieri, F. (2021). A combination of clustering-based under-sampling with ensemble methods for solving imbalanced class problem in intelligent systems. *Technological Forecasting and Social Change*, 169, 120796.
- [4] Manasrah, A. M., Gupta, B. B. (2019). An optimized service broker routing policy based on differential evolution algorithm in fog/cloud environment. *Cluster Computing*, 22(1), 1639-1653.
- [5] Gou, Z., et al. (2017). Analysis of various security issues and challenges in cloud computing environment: a survey. In *Identity Theft: Breakthroughs in Research and Practice* (pp. 221-247). IGI global.
- [6] A Dahiya, B. Gupta (2021), Edge Intelligence: A New Emerging Era, *Insights2Techinfo*, pp.1
- [7] A. Dahiya (2021), Integration of Cloud and Fog Computing for Energy Efficient and Scalable Services, *Insights2Techinfo*, pp.1
- [8] Mirsadeghi, F., Rafsanjani, M. K., Gupta, B. B. (2020). A trust infrastructure based authentication method for clustered vehicular ad hoc networks. *Peer-to-Peer Networking and Applications*, 1-17.
- [9] Kaggle, "Getting started with Text Preprocessing," Kaggle, 25 03 2019. [Online]. Available: <https://www.kaggle.com/sudalairajkumar/getting-started-with-text-preprocessing>. [Accessed 10 02 2021].
- [10] S. A. S. S. Prakash Hiremath, "Cluster Analysis of Customer Reviews Extracted from Web Pages," *Journal of Applied Computer Science & Mathematics*, 24 07 2014. [Online]. Available: https://www.researchgate.net/publication/47807593_Cluster_Analysis_of_Customer_Reviews_Extracted_from_Web_Pages. [Accessed 06 02 2021].
- [11] "A Visual Notebook to Using BERT for the First Time," Google Colab, 28 01 2020. [Online].

Available: https://colab.research.google.com/github/jalammar/jalammar.github.io/blob/master/notebooks/bert/A_Visual_Notebook_to_Using_BERT_for_the_First_Time.ipynb. [Accessed 08 02 2021].

- [12] H. E. BOUKKOURI, "Text Classification: The First Step Toward NLP Mastery," Medium, 18 06 2018. [Online]. Available: <https://medium.com/data-from-the-trenches/text-classification-the-first-step-toward-nlp-mastery-f5f95d525d73> . [Accessed 03 08 2021].