Module: Text analysis

Title: Topic modelling on customer reviews

Student: Sujatha Ramesh Matric Number:30006034

Executive summary:

For a frequently buying product in Amazon, the number of reviews can be in hundreds or even thousands. However, since these online reviews are quite often overwhelming in terms of numbers and information, an intelligent system, capable of finding key insights (topics) from these reviews. Topic modelling is the method used to extract topics from these reviews. It will be of great help for both the consumers and the sellers. This system will serve two purposes:

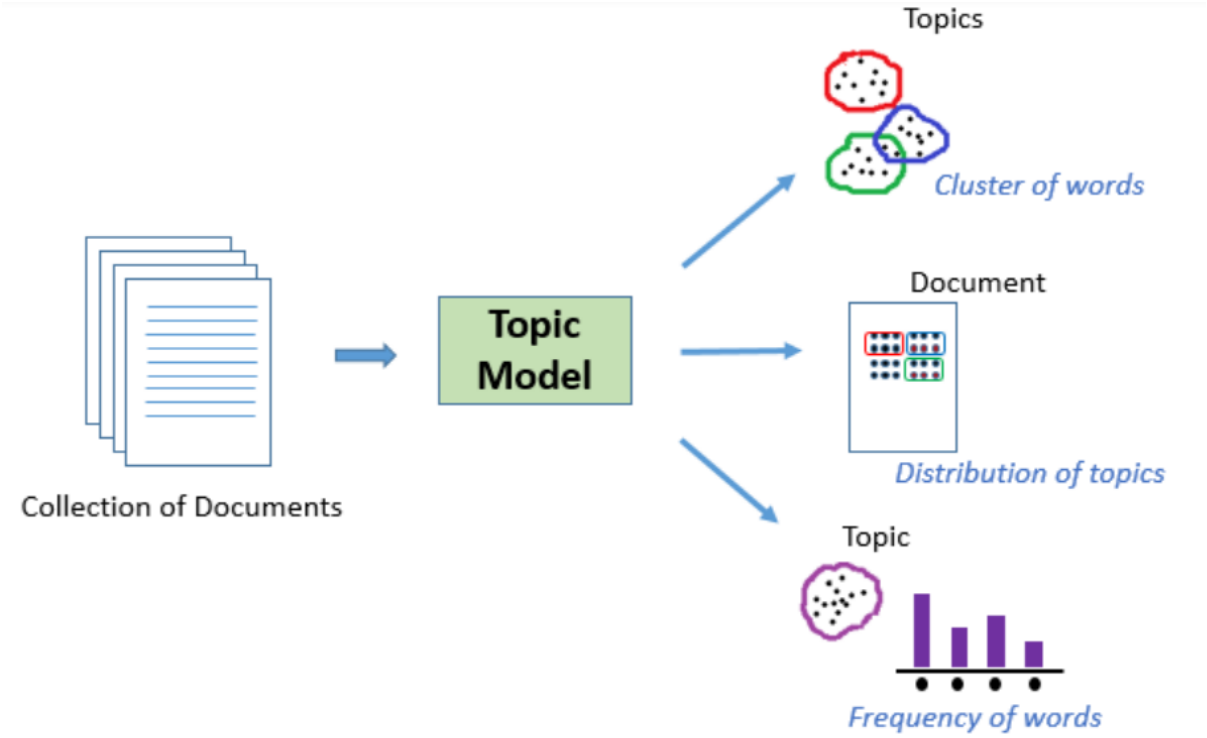
- Enable consumers to quickly extract the key topics covered by the reviews without having to go through all of them.
- Help the sellers/retailers get consumer feedback in the form of topics (extracted from the consumer reviews).

From a business standpoint, it is very important to understand how customer feedback is on the products/services they offer to improvise the same for the customer satisfaction.

Methodology:

Topic Modelling is a process to automatically identify topics present in a text object and to derive hidden patterns exhibited by a text corpus. Topic Models are very useful for multiple purposes, including:

- Document clustering.
- Organizing large blocks of textual data.
- Information retrieval from unstructured text.
- Feature selection. A good topic model, when trained on some text about the stock market, should result in topics like “bid”, “trading”, “dividend”, “exchange”, etc. The below image illustrates how a typical topic model works:



In our case, instead of text documents, we have thousands of online product reviews. Our aim here is to extract a certain number of groups of important words from the reviews. These groups of words are basically the topics which would help in ascertaining what the consumers are actually talking about in the reviews.

Conclusion:

The reviews from the dataset are pre-processed and retrieve the most frequently repeated words from the reviews to extract the topics using LDA model with libraries like gensim and pyLDAvis.

PROJECT

1 Introduction:

This is the part 2 of the project for the Spring semester module Text analysis and NLP. This project covers,

1. Introduction and Motivation.
2. Detailed view of Data including Data pre-processing, descriptive statistics and Analysis plan.
3. Methodology.
4. Limitations.
5. Conclusion.

1.1 Motivation:

Analytics Industry is all about obtaining the “Information” from the data. With the growing amount of data in recent years, that too mostly unstructured, it’s difficult to obtain the relevant and desired

information. But technology has developed some powerful methods which can be used to mine through the data and fetch the information that we are looking for.

One such technique in the field of text mining is Topic Modelling. As the name suggests, it is a process to automatically identify topics present in a text object and to derive hidden patterns exhibited by a text corpus. Thus, assisting better decision making.

Topic Modelling is different from rule-based text mining approaches that use regular expressions or dictionary-based keyword searching techniques. It is an unsupervised approach used for finding and observing the bunch of words (called “topics”) in large clusters of texts.

Topic Models are very useful for the purpose for document clustering, organizing large blocks of textual data, information retrieval from unstructured text and feature selection. For Example – New York Times are using topic models to boost their user – article recommendation engines. Various professionals are using topic models for recruitment industries where they aim to extract latent features of job descriptions and map them to right candidates. They are being used to organize large datasets of emails, customer reviews, and user social media profiles.

Aim:

In this project, topic modelling technique is done to extract the useful topics from Amazon face mask reviews using the concept of Latent Dirichlet Allocation (LDA).

Analysis Plan:

1. Import the data and other libraries.
2. Data pre-processing: Converting upper to lower case, normalising, tokenising, Lemmatization and so on.
3. Extracting topic from the customer reviews using Latent Dirichlet Allocation (LDA).

2 Data

I choose Amazon face mask’s customer reviews for Topic modelling.

2.1 DATA RETRIEVAL AND UNDERSTANDING:

Importing all necessary libraries:

```
import numpy as np
import gensim
import os
import nltk
from nltk import FreqDist
nltk.download('stopwords')
import pandas as pd
import numpy as np
import re
import gzip
import spacy
import gensim
from gensim import corpora
import pyLDAvis
import pyLDAvis.gensim
import matplotlib.pyplot as plt
import seaborn as sns
```

The dataset is extracted from the Amazon product page using the extension called Amazon reviews Exporter. And the dataset looks like,

	id	profileName	text	date	title	rating	images	helpful
0	R12R75KL44J0TR	Abhishek Raj	\n I found the product more than expected. Ve...	Reviewed in India on 1 March 2021	Good Product	5	https://images-na.ssl-images-amazon.com/images...	254
1	R1JBVRPYUAB9IL	tejas	\n Masks are too comfortable especially for a...	Reviewed in India on 20 January 2021	Comfortable and high quality N95 masks	5	https://images-na.ssl-images-amazon.com/images...	205
2	R24A67ANCH72S	Arpit	\n This CERTIFIED N95 mask gives a sense of a...	Reviewed in India on 2 April 2021	awesome N95 mask and made in India....	5	https://images-na.ssl-images-amazon.com/images...	107
3	R2XEP5P47XSB2S	Rajni gupta	\n This is a nice mask. My father goes to off...	Reviewed in India on 2 May 2021	Great mask	5	https://images-na.ssl-images-amazon.com/images...	88
4	RE73LQOE8T73	Sunil jangir	\n Â Mask bahut hi comfort he or mask Ki flee...	Reviewed in India on 15 May 2021	Acha he or comfortable bi he	5	https://images-na.ssl-images-amazon.com/images...	85

2.2 Dataset size:

I collected 10,067 reviews totally. The shape of the dataset shows 10,067 rows and 8 columns.

```
df.shape
```

```
(10067, 8)
```

2.3 Data Exploration:

This dataset has 8 columns namely id, profilename, text, date, title, rating, images and helpful.

Id – It contains the identification number of the customer. Profilename – It is the name of the customer.

Text – It contains reviews of the product given by the customer.

Date – This column contains the reviewed date of the product.

Title – This title column contains the short description of the reviews.

Rating – It contains the rating for the product ranges from 1 – 5 given by customer.

Images – This column contains the images of the product which is brought by the customer.

Helpful – This column contains the number of the people who find that the reviews are helpful.

2.4 Descriptive statistics:

```
df.describe()
```

	rating	helpful
count	10067.000000	10067.000000
mean	3.631767	1.017384
std	1.604062	11.441200
min	1.000000	0.000000
25%	2.000000	0.000000
50%	4.000000	0.000000
75%	5.000000	0.000000
max	5.000000	521.000000

Rating:

From the above description of the dataset, the rating has count and the average as well as median value nearly 4. The minimum value is 1 and the maximum value is 5. As the rating ranges from 1 - 5.

Helpful:

The minimum of zero customer finds the reviews helpful and the maximum of 521 customers find helpful with the average of 1.017.

2.5 Data Pre-processing:

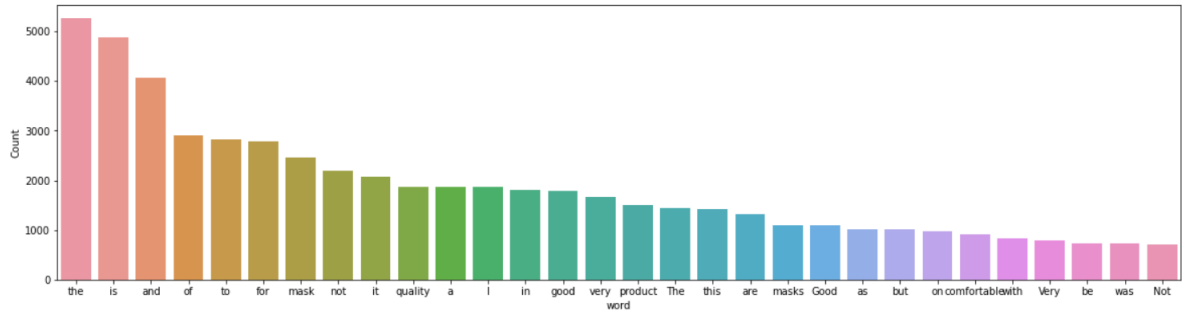
Data pre-processing is the process of transforming raw data into an understandable format prior to processing and analysis. It involves in reformatting data, making corrections to data and the combining of datasets to enrich data.

Missing values:

This dataset contains 106 missing values in the text attribute and 8815 missing values in the images attribute. The images attribute is not important for the topic modelling but the text attribute is important.

Data pre-processing and cleaning is an important step before any text mining task, in this step, I removed the punctuations, stopwords and normalize the reviews as much as possible. After every pre-processing step, it is a good practice to check the most frequent words in the data. Therefore, let's define a function that would plot a bar graph of n most frequent words in the data.

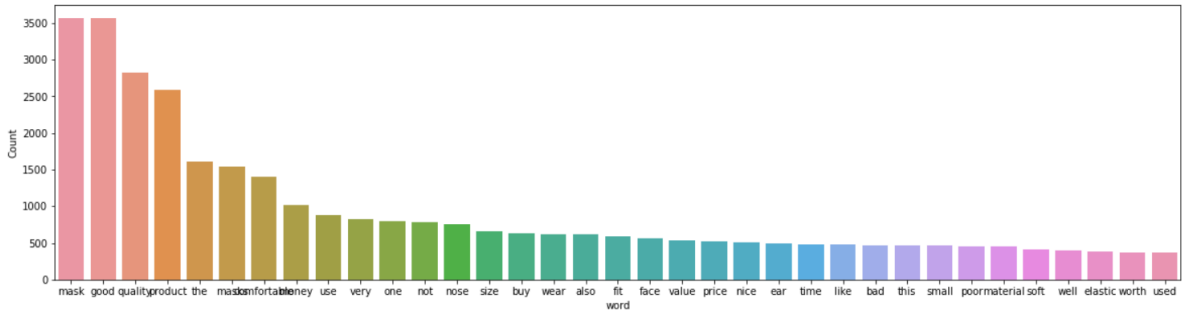
```
freq_words(df['text'])
```



From the above visualization, the most common words are 'the', 'and', 'to', 'a', 'is', so on and so forth. These words are not so important for my task and they do not tell any story. I have to get rid of these kinds of words. Before that let's remove the punctuations and numbers from the text data.

After removing stopwords, turning the entire text to lowercase, remove short words etc

```
freq_words(reviews, 35)
```

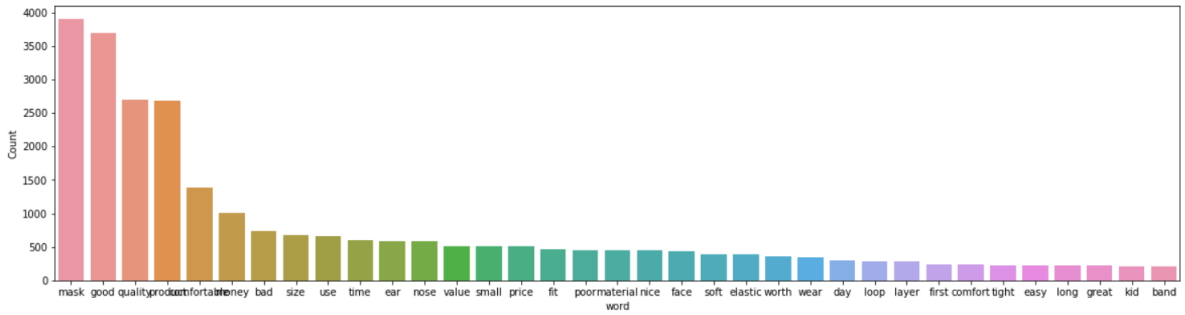


From the above plot, can see some improvement. Term like ‘comfortable’, ‘quality’, ‘poor’, ‘soft’ have come up which are quite relevant for the mask category. However, we still have neutral terms like ‘the’, ‘this’, ‘not’, ‘like’ which are not that relevant.

To further remove noise from the text I can use lemmatization from the spacy library. It reduces any given word to its base form thereby reducing multiple forms of a word to a single word.

After de-tokenize, lemmatization and filtering only nouns and adjectives, plotted the most common words.

```
freq_words(df['reviews'], 35)
```



From the above visuals, it seems that now most frequent terms in our data are relevant. I can now go ahead and start building our topic model.

3 Methodology:

LDA for Topic Modelling:

Latent Dirichlet Allocation is the most popular topic modelling technique. It is a matrix factorization technique. LDA produces the document from the mixture of topics. These topics generates words based on their probability distribution. When the dataset of text or documents are given, LDA tries to track and figure out what topics could create those documents in the first place. In vector space, any corpus can be represented as a document-term matrix. The main aim of LDA is to improve the document topic distribution and topic word. LDA makes use of sampling techniques in order to improve these matrices.

Using LDA model, here created a 7 topics from the text corpus.

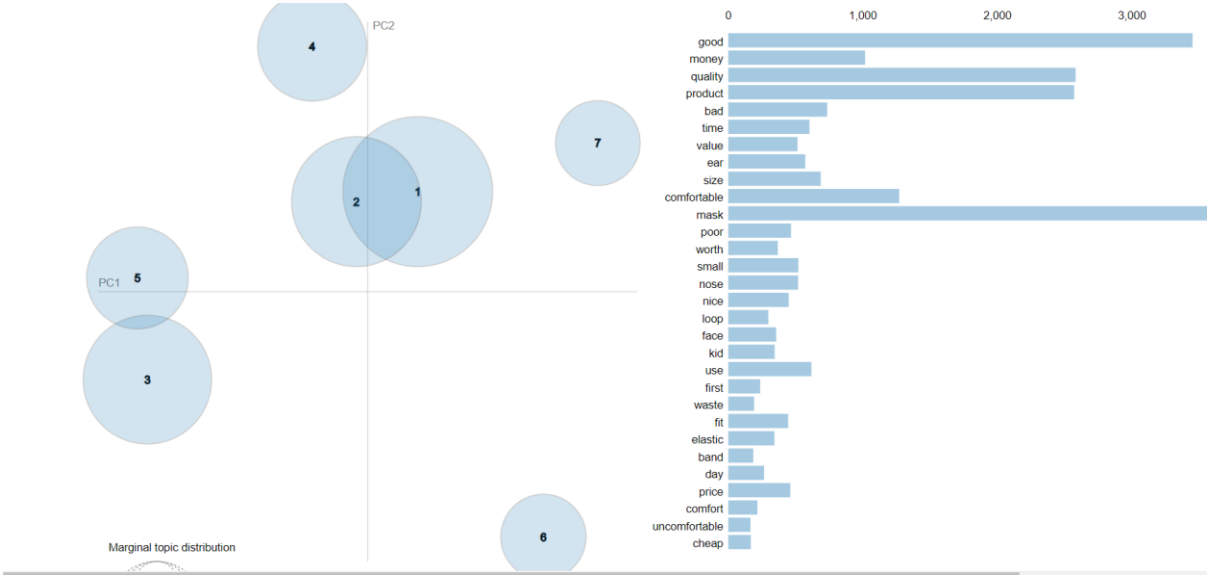
```
lda_model.print_topics()
```

```
[(0,
  '0.259*"good" + 0.080*"quality" + 0.080*"comfortable" + 0.072*"product" + 0.069*"mask" + 0.021*"price" + 0.017*"soft" + 0.015
*"wear" + 0.015*"easy" + 0.013*"material"'),
 (1,
  '0.235*"money" + 0.119*"value" + 0.085*"worth" + 0.046*"product" + 0.044*"waste" + 0.027*"comfort" + 0.020*"comfortable" + 0.
016*"piece" + 0.015*"useful" + 0.015*"price"'),
 (2,
  '0.139*"mask" + 0.069*"product" + 0.045*"nice" + 0.022*"layer" + 0.017*"pack" + 0.017*"different" + 0.016*"colour" + 0.015*"c
olor" + 0.014*"box" + 0.011*"review"'),
 (3,
  '0.098*"time" + 0.093*"ear" + 0.050*"mask" + 0.049*"loop" + 0.041*"use" + 0.039*"first" + 0.034*"elastic" + 0.030*"band" + 0.
027*"uncomfortable" + 0.017*"pain"'),
 (4,
  '0.158*"quality" + 0.077*"mask" + 0.066*"poor" + 0.028*"day" + 0.024*"cheap" + 0.024*"use" + 0.022*"year" + 0.022*"material"
+ 0.021*"strap" + 0.021*"string"'),
 (5,
  '0.172*"bad" + 0.170*"product" + 0.094*"quality" + 0.031*"item" + 0.026*"excellent" + 0.025*"child" + 0.021*"average" + 0.019
*"buy" + 0.017*"expensive" + 0.015*"packing"'),
 (6,
  '0.070*"size" + 0.053*"small" + 0.053*"nose" + 0.041*"mask" + 0.036*"face" + 0.035*"kid" + 0.032*"fit" + 0.019*"tight" + 0.01
8*"old" + 0.016*"big"')]
```

From the printed topics in the LDA model, the first topic 0 has terms like ‘good’, ‘quality’, ‘comfortable’, ‘wear' 'price’, indicating that the topic is very much related to Mask. Similarly, Topic 4 seems to be about the overall value of the product as it has terms like ‘cheap’, ‘material’, and ‘strap’ and also topic 1 and 6.

Topics Visualization:

To visualize the topics in a 2-dimensional space I can use the pyLDAvis library. This visualization is interactive in nature and displays topics along with the most relevant words.



The model above shows that it extracted 7 topics from the customer text reviews. Some of topics are overlapped, it means that it has relevant text within it. But the model can be evaluated whether it is a good or bad by calculating perplexity and coherence score.

Perplexity:

Perplexity is a statistical measure of how well a probability model predicts a sample. As applied to LDA, for a given value estimate the LDA model. Then it gives the theoretical word distributions represented by the topics, compare that to the actual topic mixtures, or distribution of words in your documents.

Coherence:

Topic Coherence measures score a single topic by measuring the degree of semantic similarity between high scoring words in the topic. These measurements help distinguish between topics that are semantically interpretable topics and topics that are artifacts of statistical inference

```
print('\nPerplexity: ', lda_model.log_perplexity(doc_term_matrix,total_docs=10000))  # a measure of how good the model is. lower the better.

# Compute Coherence Score
from gensim.models.coherencemodel import CoherenceModel
coherence_model_lda = CoherenceModel(model=lda_model, texts=tokenized_reviews, dictionary=dictionary , coherence='c_v')
coherence_lda = coherence_model_lda.get_coherence()
print('\nCoherence Score: ', coherence_lda)
```

Then, the Perplexity and coherence value is calculated. The lower perplexity score and higher coherence score estimates how good the model is.

```
Perplexity:  -5.944328118556108

Coherence Score:  0.4235288348748996
```

The perplexity value is -5.94 and the coherence score is 42. The Lower the perplexity and higher the coherence score shows how the model is good.

4 Limitations:

- Only the reviews of the face mask are used.
- The number of topics is given as 7 in the LDA model. Can also give more than or less than 7.
- The coherence and perplexity are calculated to see the betterness of the LDA model. If there are many overlaps in the topic model, it means that the model is not good. So, we can change the number of topics and for each topic we can generate various LDA model to find which number of topic shows better the model.

5 Conclusion:

The aim of this project is now built and is completed successfully with the descriptive statistics, main analysis and methodology as planned. As the result, topic model technique is done on customer reviews and extract about 7 topics respectively. Finally, perplexity and coherence score are observed to see how the model is good. This model is pretty good as it has lower perplexity and higher coherence values.

6 References:

https://www.machinelearningplus.com/nlp/topic-modeling-gensim-python/

https://ai.stanford.edu/~ang/papers/jair03-lda.pdf

https://www.analyticsvidhya.com/blog/2016/08/beginners-guide-to-topic-modeling-in-python/

https://www.researchgate.net/publication/306285022_Summarization_of_Customer_Reviews_for_a_Product_on_a_website_using_Natural_Language_Processing

https://www.analyticsvidhya.com/blog/2021/07/topic-modelling-with-lda-a-hands-on-introduction/

https://towardsdatascience.com/evaluate-topic-model-in-python-latent-dirichlet-allocation-lda-7d57484bb5d0

https://stats.stackexchange.com/questions/375062/how-does-topic-coherence-score-in-lda-intuitively-makes-sense

https://www.machinelearningplus.com/nlp/topic-modeling-gensim-python/#:~:text=Topic%20Modeling%20is%20a%20technique,in%20the%20Python's%20Gensim%20package.