Yuan Wang | ywang340
CSC447 Natural Language Processing – Final Project
University of Rochester
Instructor: Prof. Aaron Steven White
05/02/2022

Topic Modeling & Text Summarization

1. Introduction

Customer review is a crucial information for company to understand their customers feedback for the products and services. So that the company could adjust its strategies to better serve their customers. However, before reading the huge text, it would be great if we can get an overview of topics and the summarized opinion before dive into the details.
The purpose of this project is to use **semantic analysis** (NLTK, genism), **unsupervised topic model (LDA)**, and **abstractive text summarization** methods(T5) to better understand customer review. The methods will be used for my future work in an ML company that helps retailers improve their online shopping revenue and customer perception. I use this public data (Amazon food review) from Kaggle for this project to prevent the company's data leakage. The data can be found on: Kaggle.

2. Methods

There is total four steps for this analysis:

- I use traditional methods to clean text data and lemmatize, vector the words.
- I check whether that review summary is enough to represent whole opinion from customer review by using cosine similarity. I also extract the sentiment polarity score from review to compute Cosine similarity with customer review score.
- I use latent Dirichlet allocation (LDA) and Coherence score to find out the optimal topic numbers, then label each review with topics. These two will give a general idea of each review text before dive into the details.
- I use the text summarization methods to summary review texts by topics and scores.

The workflow shows in Figure-1 as below:

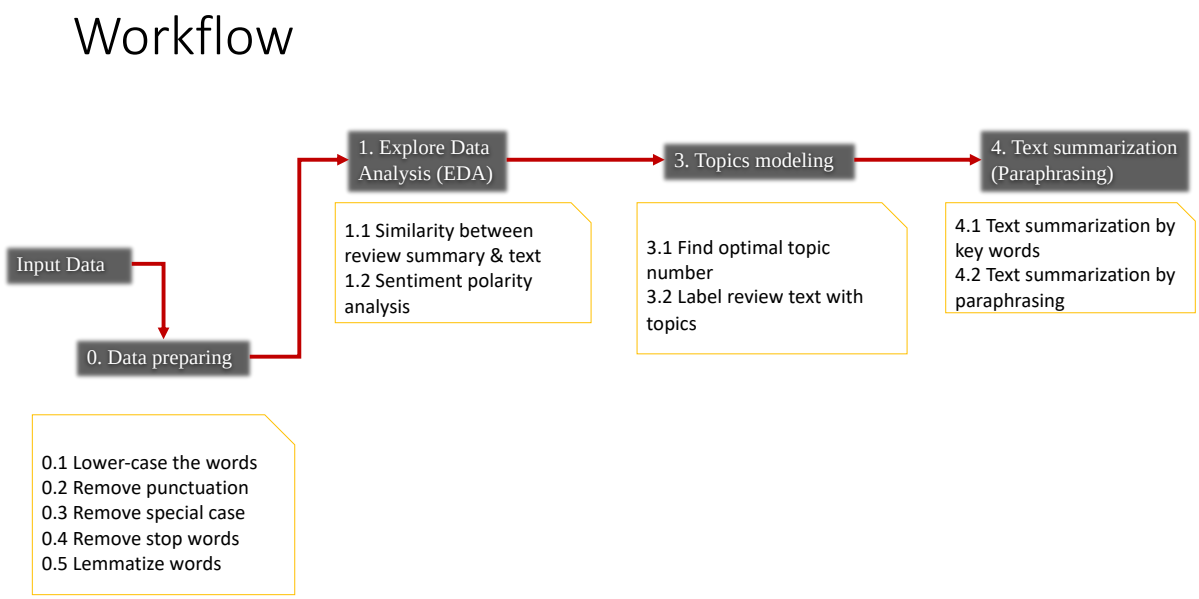

Figure-1. Project workflow

3. Findings

There are two columns of texts from data, can we use review summary instead of both review summary and text? The answer is no.

3.1 Similarity between review summary and text

Figure-2 shows the similarity value between text and summary are not very high. Most of the score stays below 0.6. In fact, the density is at its highest when cosine similarity is 0. The density hit to 9 when the cosine similarity ranges between -0.05 to 0.03. When cosine similarity is between 0.04 to 0.6, the density stays below 2. As the conclusion, summary cannot represent the whole customer review, only very few summaries describe what the text says. Thus, we need to analyze the review summary and text together. In next, I combined cleaned review summary and text into one column for further anaylsis.

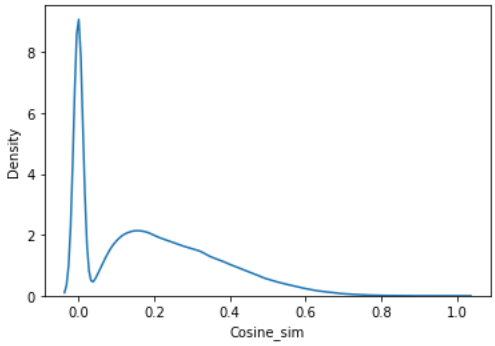

Figure-2. Cosine similarity between review summary & text

What is the general sentiment for the customer review? Does sentiment score indicate the customer review score? The answer is yes.

3.2 Text Sentiment polarity analysis

I use the *SentimentIntensityAnalyzer* to get the sentiment score. The similarity between Score and Sentiments compound score is 0.92 which means that text sentiment score can represent customer review score.

```
[11] from sklearn.metrics.pairwise import cosine_similarity
     cosine_similarity(amazon["Score"].values.reshape(1, -1), amazon["compound"].values.reshape(1, -1))

     array([[0.91688415]])
```

Figure-3. Cosine similarity between sentiment score & customer review score

3.3 Topic modeling (LDA)

The next step is to find out the topics for the whole review data so that we will know what the texts are talking about. I choose Latent Dirichlet allocation (LDA) for topic modeling because it assumes each document as a collection of topics in a certain proportion, so I can "soft" cluster the texts into different topics.
Next, I use *CoherenceMode*l algorithm to evaluate the optimal topics number. Figure-4 shows that the optimal number is 5.

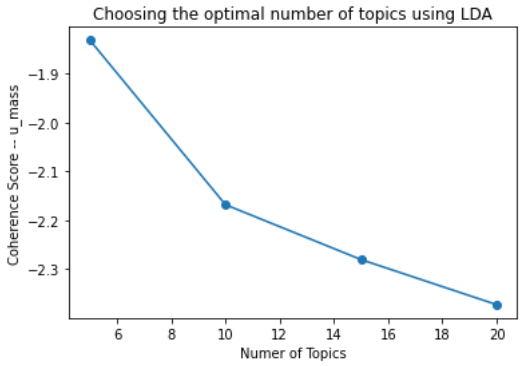

Figure-4 Choosing optimal # of topics

Figure-5 shows the 5 topics that LDA generates, and the potential topics could be:

0. Amazon's price
1. Snack
2. Pet food
3. Amazon product & price
4. Coffee/Tea

```
[(0,
  '0.021*"use" + 0.014*"make" + 0.013*"like" + 0.012*"taste" + 0.011*"good" + 0.010*"product" + 0.010*"water" + 0.008*"great" + 0.008*"add" + 0.008*"oil"'),
 (1,
  '0.024*"taste" + 0.018*"like" + 0.015*"good" + 0.015*"flavor" + 0.011*"great" + 0.011*"chip" + 0.010*"love" + 0.010*"chocolate" + 0.010*"sugar" + 0.009*"eat"'),
 (2,
  '0.030*"food" + 0.028*"dog" + 0.019*"treat" + 0.018*"love" + 0.014*"cat" + 0.011*"get" + 0.010*"like" + 0.010*"eat" + 0.010*"one" + 0.007*"give"'),
 (3,
  '0.019*"product" + 0.018*"price" + 0.018*"amazon" + 0.017*"buy" + 0.017*"order" + 0.015*"great" + 0.013*"store" + 0.013*"find" + 0.013*"get" + 0.011*"good"'),
 (4,
  '0.043*"coffee" + 0.038*"tea" + 0.022*"cup" + 0.019*"flavor" + 0.018*"taste" + 0.015*"like" + 0.013*"drink" + 0.013*"good" + 0.009*"try" + 0.009*"one"')]
```

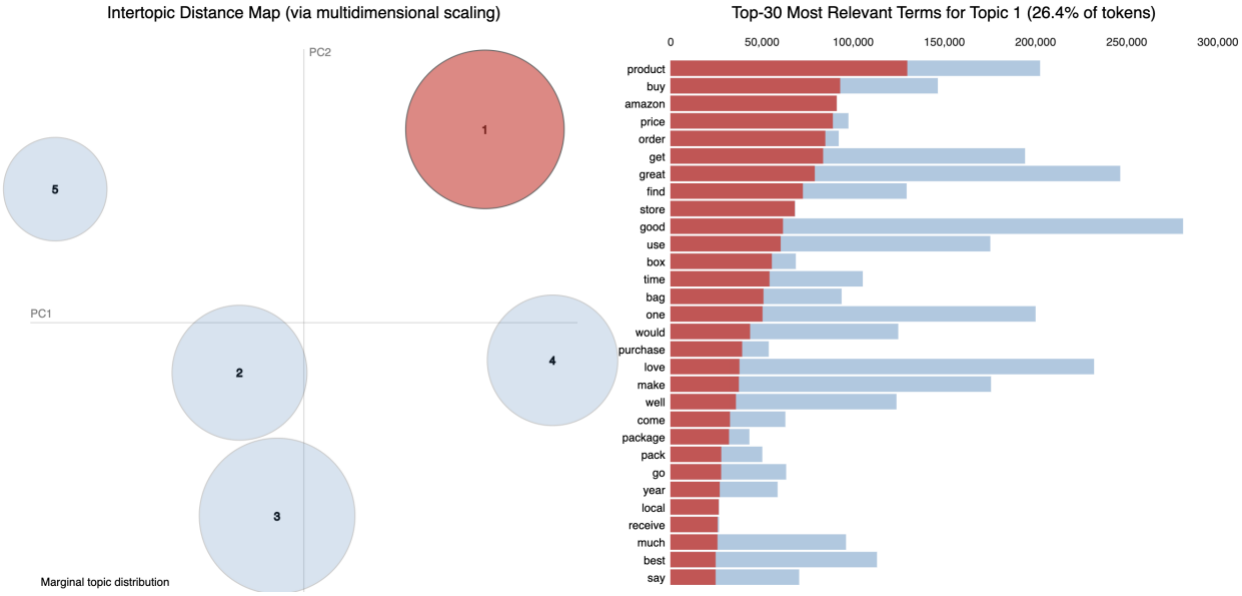

Figure-5 5 topics

Then, I label each text with a topic number and name, so that we can drill down the details based on the topics. Figure-6 shows the example of the output:

	Id	dominant_topic	perc_topic	keywords	review
0	1	3	0.545052	food, dog, treat, love, cat, like, eat, get, o...	good quality dog food buy several vitality can...
1	2	0	0.637893	product, buy, amazon, price, order, get, great...	advertised product arrive labeled jumbo salt p...
2	3	2	0.329799	taste, like, good, flavor, great, love, eat, c...	delight say confection around century light pi...
3	4	1	0.876109	tea, taste, like, flavor, water, drink, use, g...	cough medicine look secret ingredient robituss...
4	5	2	0.519893	taste, like, good, flavor, great, love, eat, c...	great taffy great taffy great price wide assor...
...	...	...	...	...	...
568449	568450	2	0.841936	taste, like, good, flavor, great, love, eat, c...	without great sesame chicken this good good re...
568450	568451	4	0.586835	coffee, cup, flavor, taste, like, good, great,...	disappointed disappointed flavor chocolate not...
568451	568452	3	0.893097	food, dog, treat, love, cat, like, eat, get, o...	perfect maltipoo star small give one training ...
568452	568453	3	0.677976	food, dog, treat, love, cat, like, eat, get, o...	favorite train reward treat best treat train r...
568453	568454	2	0.504178	taste, like, good, flavor, great, love, eat, c...	great honey satisfied product advertise use ce...

568454 rows × 5 columns

Figure-6 Add dominated topic number to each review text

3.4 Text summarization by topics

In general, there are two types of text summarization, **Abstractive** and **Extractive** summarization. Since abstractive methods select words based on **semantic**

understanding, even those words did not appear in the source documents. So, I use this method for this project to adapt the course criteria. T5 is a new transformer model from Google that is trained in an end-to-end manner with text as input and modified text as output. The minimum length of rephrased sentence is 5 and maximum is 10. Figure-7 shows the paraphrasing summary from each review text.

Id	dominant_topic	perc_topic	keywords	review	t5_summary
1	2	0.582149	food, dog, treat, love, cat, get, like, eat, o...	good quality dog food buy several vitality can...	labrador finicky appreciate product well
2	3	0.64506	product, price, amazon, buy, order, great, sto...	advertised product arrive labeled jumbo salt p...	advertised product arrive labeled jumbo
3	1	0.63771	taste, like, good, flavor, great, chip, love, ...	delight say confection around century light pi...	delight say confection around century light pi...
4	1	0.490769	taste, like, good, flavor, great, chip, love, ...	cough medicine look secret ingredient robituss...	medicine look secret ingredient robitussin
5	3	0.952346	product, price, amazon, buy, order, great, sto...	great taffy great taffy great price wide assor...	great taffy great price great
...	...	...	...	...	...
96	2	0.982775	food, dog, treat, love, cat, get, like, eat, o...	good healthy dog food pleased natural balance ...	good healthy dog food pleased natural balance ...
97	2	0.81849	food, dog, treat, love, cat, get, like, eat, o...	great dog food year old basenji jack russell m...	basenji jack russell mix love dog
98	2	0.933713	food, dog, treat, love, cat, get, like, eat, o...	great allergy sensitive dog food dog love pup ...	great allergy sensitive dog food dog love food
99	2	0.982939	food, dog, treat, love, cat, get, like, eat, o...	perfect english bulldog allergy english bulldo...	english bulldog skin allergy vet recommend wean
100	2	0.945864	food, dog, treat, love, cat, get, like, eat, o...	bad feed golden retriever hat eat do give terr...	golden retriever hat eat do give

Figure-7 Using google T5 to paraphrase review text

Let's check the first example. The cleaned review summary & text is like this
"*good quality dog food buy several vitality can dog food product find good quality product look like stew process meat smell well labrador finicky appreciate product well most*"
LDA model label it as **topic 2** for **pet food**, and the T-5 summary is
"*labrador finicky appreciate product well*". The result makes sense and matching to the original text.

4. Conclusion and future work

We proof that the similarity between review summary and text is quite low (3.1). So, we cannot get the main opinion from customer review summary only. However, the customer review text is so big and it's impossible to quickly get main idea by reading text line by line. Thus, I use LDA model to "softly" (probability) cluster each cleaned text into different topics. Then, I use Coherence score to find out the optimal topic number is 5. I extract the key word of each text and label it with topic number. Therefore, it is convenience to understand the main idea of the whole review by different topics. Last, I use abstractive summarization (google T5) to paraphrase the review text into 5-10 words. This will shorten the time for human check each review's gist.

There are a lot of work can be done in the future. First, evaluated the performance for different NLP models, such as NMF, K-means for the topic modeling. Second, expand the text summarization models using different methods (extractive models) and compare the performance. Third, I could detect the fake reviews to improve the helpfulness rate for customer.

References:

1. Lewis, Mike, et al. “Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension.” *arXiv preprint arXiv:1910.13461* (2019).
2. Raffel, Colin, et al. “Exploring the limits of transfer learning with a unified text-to-text transformer.” *arXiv preprint arXiv:1910.10683* (2019).
3. Raymond Cheng, Abstractive Summarization Using Pytorch: https://towardsdatascience.com/abstractive-summarization-using-pytorch-f5063e67510
4. Praveen Dubey, Understand Text Summarization and create your own summarizer in python: https://towardsdatascience.com/understand-text-summarization-and-create-your-own-summarizer-in-python-b26a9f09fc70
5. Topic Modeling in Python: Latent Dirichlet Allocation (LDA) https://towardsdatascience.com/endto-end-topic-modeling-in-python-latent-dirichletallocation-lda-35ce4ed6b3e0