Review-Summarization-using-NLTK

Motivation

****What is Review Summarization?****

Summarization is the task of condensing a piece of text to a shorter version, reducing the size of the initial text while at the same time preserving key informational elements and the meaning of content. There are important applications for text summarization in various NLP related tasks such as text classification, question answering, legal texts summarization, news summarization, and headline generation. Moreover, the generation of summaries can be integrated into these systems as an intermediate stage which helps to reduce the length of the document.

****Why we need Data Science approach for Review Summarization?****

Since manual text summarization is a time expensive and generally laborious task, the automatization of the task is gaining increasing popularity and therefore constitutes a strong motivation for academic research. In the big data era, there has been an explosion in the amount of text data from a variety of sources. This volume of text is an inestimable source of information and knowledge which needs to be effectively summarized to be useful. This increasing availability of documents has demanded exhaustive research in the NLP area for automatic text summarization. Automatic text summarization is the task of producing a concise and fluent summary without any human help while preserving the meaning of the original text document.
It is very challenging, because when we as humans summarize a piece of text, we usually read it entirely to develop our understanding, and then write a summary highlighting its main points. Since computers lack human knowledge and language capability, it makes automatic text summarization a very difficult and non-trivial task. Various models based on machine learning have been proposed for this task. Most of these approaches model this problem as a classification problem which outputs whether to include a sentence in the summary or not. Other approaches have used topic information, Latent Semantic Analysis (LSA), Sequence to Sequence models, Reinforcement Learning and Adversarial processes.
In general, there are two different approaches for automatic summarization: extraction and abstraction. First, extractive summarization systems form summaries by copying and rearranging passages from the original text.
Second, abstractive summarization systems generate new phrases, rephrasing or using words that were not in the original text. Due to the difficulty of abstractive summarization, the great majority of past work has been extractive.
For our purpose we will be using extractive approach." Our main aim is to develop an automatic system that takes all the positive and negative reviews for a particular product and generates an overall summary of a few lines that is easy to read and make final decision regarding the overall user experience of product . "

****1) The Data****

"Amazon Consumer Reviews of Amazon Products" dataset contains the data of 5000 records and 24 feature variables.the Data is hosted as part of a competition by Kaggle. Here we take from kaggle and summarize reviews for a particular product Amazon Echo Show Alexa-enabled Bluetooth Speaker.

2) Data Cleaning

Once we have all the Data. We performed Data cleaning on it. Data Cleaning is necessary because the more clean the data is better is the model predictions. Garbage data results in inaccurate models that cannot make good predictions. So First we checked for missing values. We also checked for duplicate data and removed them. For the purpose of consistency in text data we converted everything to lowercase characters. The title text has several punctuations. Punctuations are often unnecessary as it doesn't add value or meaning to the NLP model. So we can remove them.
Now, we have a list of words without any punctuation. Next we removed the stop words. Stop words are irrelevant words that won't help in identifying a text. We will use "nltk" library for stop-words and some of the stop words in this library are :('I', 'me', 'we','are'). There are 179 stop words in this library.
Stemming and Lemmatizing is the process of reducing a word to its root form. The main purpose is to reduce variations of the same word, thereby reducing the corpus of words we include in the model. The difference between stemming and lemmatizing is that, stemming chops off the end of the word without taking into consideration the context of the word. Whereas, Lemmatizing considers the context of the word and shortens the word into its root form based on the dictionary definition. Stemming is a faster process compared to Lemmantizing. Hence, it a trade-off between speed and accuracy.
Other cleaning steps can be performed based on the data ex: Removing URLs, HTML tags, emoji, numbers etc., doing Contraction mapping i.e. removing words or combinations of words that are shortened by dropping letters and replaced by an apostrophe. We can also remove any text inside the parenthesis or short words i.e words less than certain length (ex word>=3 like the, be, as , an).

3) Data Modelling:

We will start by tokenizing the sentences here instead of words and we'll give weight to these sentences. Then we can create the Frequency matrix of the words in each sentence. Here, each sentence is the key and the value is a dictionary of word frequency. Once we have the frequency matrix we will calculate and build a Term-frequency matrix. Term Frequency (TF) is how often a word appears in a document, divided by how many words there are. We'll find the TermFrequency for each word in a product review.

TF(t) = (Number of times term t appears in a document) / (Total number of terms in the document)

Next we create a table for idf matrix. IDF is Inverse document frequency, Term frequency is how common a word is, inverse document frequency (IDF) is how unique or rare a word is.

IDF(t) = log_e(Total number of documents / Number of documents with term t in it)

Here, the document is a paragraph, the term is a word in a paragraph. Once we have both the Tf and Idf matrix we will calculate TF-IDF, which is nothing but the product of 2 matrices and generate a final matrix. Next we will score the sentences, Scoring a sentence is differs with different algorithms. Here, we are using Tf-IDF score of words in a sentence to give weight to the paragraph. Similar to any summarization algorithms, there can be different ways to calculate a threshold value. We’re calculating the average sentence score. Finally to generate the summary we select a sentence for summary if the sentence score is more than the average score. Here For the threshold, we’ve used 1.3x of the average score,

4) Conclusion:

Online customer reviews is considered as a significant informative resource which is useful for both potential customers and product manufacturers. In web pages, the reviews are written in natural language and are unstructured-free-texts scheme. The task of manually scanning through large amounts of review one by one is computational burden and is not practically implemented with respect to businesses and customer perspectives. Therefore it is more efficient to automatically process the various reviews and provide the necessary information in a suitable form.This Data science approach addreses this problem and gives a convenient solution for summary generation.

Summary of 5000 alexa reviews for a threshold of 1.4

coexist with all iot devices. alexa is awesome! highly recommended. would recommend it to anyone. Overall product is excellent. worth every penny. waiting for more update for access to apps in the future. bought a second for my dad. it's a waste of money for us. so quick and easy. alexa can do it all! these are convenient and fun! easy to set up. i use it everyday. great product! overall works well. screen is clear. precise,accurate,and knowledgeable.

Summary of 5000 alexa reviews for a threshold of 2.0

alexa is awesome! worth every penny. so quick and easy. alexa can do it all! these are convenient and fun!