Seminar 4: Designing Real World Big Data Projects - Topic Modelling

Group: Kunal Ajaykumar Jeshang (300328339) & Blossom Goyal (300347123)

GitHub Repository: https://github.com/kjeshang/CSIS4260-Seminar4-StoryTopicModeling

Introduction

As per the instructions for this seminar, we selected a project from the online article, 13 Ultimate Big Data Project Ideas & Topics for Beginners [2022], published by UpGrad. The project of choice is a text mining project that is considered a beginner level big data project according to the online article. The text mining project discussed in the UpGrad article involves performing text analysis & visualization of a Hollywood Movie Script/s (i.e., Star Wars) using the R programming language. For the purpose of not replicating the project, it was decided to work on a text mining project utilizing a different programming language on a novel rather than a movie script. Our term project involves performing text analysis of various streaming services data using Term Frequency and Inverse Document Frequency and N Grams methods to recommend potential titles, and in turn, streaming services. However, the textual data we acquired was not sufficient enough to utilize Entity Extraction or Topic Modeling to make recommendations. Novels contain more text that contribute to breathing depth into the story in terms of themes, imagery, motifs, etc. Therefore, it was decided to explore the Topic Modeling as a method for text mining on the textual data of a novel. Python was the technology of choice for this project. The novel chosen was *The Great Gastby* by F. Scott Fitzgerald. The novel was retrieved from Project Gutenberg, which is an online library of free ebooks, in a text file format (i.e, .txt) format.

Word Frequency in Text

We first had to construct functions that appropriately count the words whilst accommodating for basic punctuation and symbols. The counter sub-package from the Collections package was helpful to do this. Another function needed to be created that reads in the contents of the novel from the text file that accommodates for spaces and new lines. Lastly, a function is created that can calculate the total number of unique words as well as the total cumulative count of all unique words. After applying the aforementioned functions, it was discovered that the novel contained 11,627 unique words, and a cumulative total of the unique words is 41,600 words.

Pre-Processing for Natural Language Processing

Despite the contents of the novel already being imported from the text file, the novel's textual data is not in a state where topic modeling is possible. Thus, a function was created to take the novel's textual data and prepare it for exploratory analysis & unsupervised learning in the form of Topic Modeling. The following packages & sub-packages were used to create this function: nltk (word_tokenize) nltk.corpus (stopwords), nltk.stem (WordNetLemmatizer), string (punctuation), and re. This function performed the following steps.

1. Tokenize the textual data into elements of a list of words, and make each word to be lower case.
2. Created another list of words that were not stop words or punctuation from the initial list of tokenized-lower cased words.
3. Created a new list of normalized words from the above mentioned list by looping through it, and appending the word to the new list if it meets the conditions discussed below.

Each word is cleaned such that punctuation and special symbols are removed from all words. Lemmatization also occurs in the loop so that action/extended words revert to their base form.

   a. The word is not a digit, not an apostrophe, and not blank.
   b. The word is not invalid such as being a single letter (other than “a” or “I”), “nt”, or “said” (which is commonly used in novels for character dialogue).

4. All words in the normalized list are combined together into a single string variable. Each word has a comma in-between each of them.
5. The pre-processed textual data of the novel is then returned by the function.

After applying the function explained in the aforementioned steps to the textual data of the novel by instantiating it to another a new variable, the pre-processed textual data is now ready for further analysis.

Word Cloud

To get a general idea on the prevalence of common words in the novel, with the help of the word cloud package and the pre-processed textual data, we created a Word Cloud to display a visual representation of the most common words. The word cloud visualization can be seen below. Larger the size of the word, the more common it is. It is clear that the most common words are “gatsby”, “daisy”, and “tom”. This makes sense as these are names of the main characters of the novel. Specifically, Jay Gatsby is the main character, whilst the characters of Daisy and Tom contribute to the internal conflict of the main character as well as progression of the plot.

Latent Dirichlet Allocation (LDA)

“LDA is a generative probabilistic model that assumes each topic is a mixture over an underlying set of words, and each document is a mixture of over a set of topic probabilities” (Kapadia, 2019). The mathematics of the LDA model is very complex in nature, but in order to train an LDA model, a set of parameters are required. Below are some of the more important ones in the context of training an LDA model programmatically using Python.

- Number of topics: the number of topics extracted from the corpus (i.e., the textual data) (Bansal, 2016).
- Number of topic terms: the number of terms composed in a single topic. If the intent is to extract themes or concepts, a high number of topic terms should be used (Bansal, 2016).

- Number of iterations/passes: “Maximum number of iterations to LDA algorithm for convergence” (Bansal, 2016).

Using the aforementioned parameters and the respective Python packages, a predetermined number of topics will be generated with each topic having the predetermined number of topic terms. Each topic would technically be differentiated numerically. It would then be the job of the human to make a decision on a descriptive word for each topic based on the associated words for the respective topic. An Intertopic Distance Map visualization can be helpful to understand how topics relate to each other (Chibueze, 2018). Also, the Intertopic Distance Map can help determine the most salient terms across all the topics as well as per topic. Specifically, a salience score “a prediction of what a human would consider to be the most important entities in the same text” (Garry, 2022).

The application of the LDA method for Topic Modelling will be discussed in the following sections in the context of the novel’s textual data. The LDA method was attempted twice using different approaches and packages, however the predetermined parameters remained consistent for each attempt. Thus, the number of topics and number of words are 5 and 10, respectively.

Topic Modelling - Attempt 1

In our first attempt at topic modelling with LDA, we used the following packages & sub-packages: gensim (corpora) and gensim.models.ldamodel.LdaModel. First, we had to create a Term Dictionary of our corpus (i.e., pre-processed textual data split into list form via comma), where every unique term is assigned an index. Second, we then created a Document Term Matrix out of the pre-processed textual data (that is in list form split via comma) using the aforementioned term dictionary. Then, the LDA model is applied using the Document Term Matrix, predetermined number of topics, and Term Dictionary. After the LDA model is applied, we printed the LDA model’s results using the predetermined number of topics and number of words. A screenshot of the results can be seen below. The results made sense but to a certain degree inconclusive as it was harder to determine a descriptive name for each topic.

```
[(0,
  '0.007*"gatsby" + 0.006*"tom" + 0.005*"daisy" + 0.004*"one" + 0.003*"little" + 0.003*"man" + 0.003*"like" + 0.003*"came" + 0.003*"get" + 0.003*"looked"'),
 (1,
  '0.006*"gatsby" + 0.005*"tom" + 0.004*"daisy" + 0.004*"one" + 0.004*"man" + 0.004*"back" + 0.003*"came" + 0.003*"like" + 0.003*"new" + 0.003*"know"'),
 (2,
  '0.008*"gatsby" + 0.005*"tom" + 0.005*"daisy" + 0.004*"like" + 0.004*"one" + 0.003*"house" + 0.003*"man" + 0.003*"car" + 0.003*"came" + 0.003*"went"'),
 (3,
  '0.008*"gatsby" + 0.007*"daisy" + 0.006*"tom" + 0.004*"one" + 0.004*"man" + 0.004*"like" + 0.004*"back" + 0.003*"went" + 0.003*"little" + 0.003*"came"'),
 (4,
  '0.007*"gatsby" + 0.006*"tom" + 0.005*"one" + 0.004*"daisy" + 0.004*"came" + 0.004*"back" + 0.004*"like" + 0.003*"little" + 0.003*"eyes" + 0.003*"old"')]
```

Topic Modelling - Attempt 2

In our second attempt at topic modelling with LDA, we used the following packages & sub-packages: pandas, numpy, sklearn.feature_extraction.text (CountVectorizer), sklearn.decomposition (LatentDirichletAllocation), and mglearn. Firstly, the CountVectorizer module was used to create a Document Term Matrix out of the novel’s textual data (that is in list form split via comma). The LDA model is then applied using the Document Term Matrix just created using the predetermined number of topics. After applying the LDA model, the predetermined number of ‘top’ words are displayed per the predetermined number of topics. A screenshot of the results can be seen below. The results made a little more sense compared to

the first attempt. Having read the story, it was easier to recognize elements of the story to the words associated with each topic. Although, not all the words in a respective topic are logical enough to assist in determining a descriptive name for a topic.

topic 0	topic 1	topic 2	topic 3	topic 4
daisy	gatsby	came	tom	like
wilson	man	little	come	house
night	went	know	mr	long
way	got	old	car	door
turned	eyes	moment	away	voice
hand	looked	new	look	people
sport	time	going	face	good
young	jordan	want	began	asked
oh	think	knew	right	took
white	saw	girl	room	head

Analyzing the words of each Topic

Topic 0	*TOPIC 1*	• Daisy: She is the love interest of the main character. She was born rich (i.e., old money) • Wilson: Not rich at all and struggling financially. • Night: The time when Gatsby hosts his parties, but is also when the disparity of those that are risk and those that are not as those that are rich appear at Gatsby's parties. • Sport: A term of endearment used to a friend which is used by Gatsby. • Hand: Could refer to Daisy's bruised finger that is caused by her husband Tom. The bruised finger symbolizes the damaged marriage of Tom and Daisy. • Young: Refers to the age of Daisy who is young and frivolous. • White: The color white serves as a symbol for purity, innocence, and honesty. Although, a lot of the characters' persona contradict what the color white stands for due to them lying about their true intentions, past, and not showing their true selves, as well as making questionable moral decisions at times. • Inconclusive words: Way, Turned, Oh This topic heavily refers to the character of Daisy, her diverging persona, and unclear intentions.
Topic 1	*TOPIC 2*	• Gatsby: The main character of the story. • Man: This is quite literal as Gatsby is a man, but the word could infer more on the societal perception of a successful man that has money, charisma, and respect. The tragedy of Gatsby that he had money and charisma, but he did not have the true

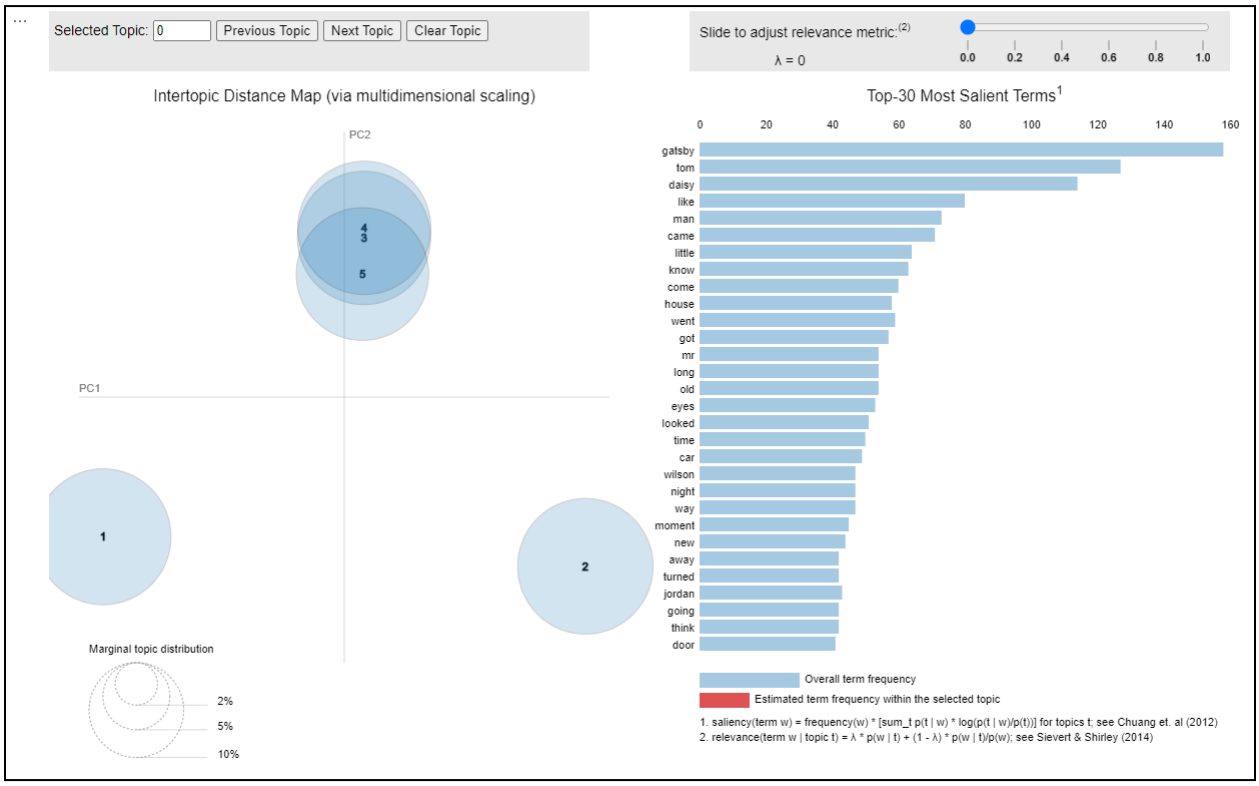
		respect of others as he represented the “New Money” socio-economic class. Specifically, he was someone that worked his way to having riches and a lavish lifestyle. Although, he did not always follow the straight-and-narrow path towards all that he has by the time we are first introduced to him in the book. ● Eyes, Looked, Saw: These words quite literally refer to the visual sense of looking at something. However, the ‘eyes’ serve as a symbol that is prevalent throughout the story. Specifically, the eyes of Doctor T.J. Eckleburg are a pair of bespectacled eyes painted on an old advertising billboard over the valley of ashes, which is a prominent location in the book. These eyes represent God staring down upon the American society, and judging it. It also represents the utter meaningless-ness of arbitrary objects, and how us as human beings derive meaning from them. ● Jordan: Is a female supporting character of the story. She is perceived as a distant and aloof person. By nature, she is cynical and self-centered; this is almost like how the world is in contemporary world. According to the narrator, Jordan tends to bend the truth to keep the world at a distance to protect herself from the world’s cruelty. Interestingly, she perceives the world in an omniscient way like Doctor T.J. Eckleberg (i.e., the eyes), as the latter is distant like Jordan yet is all knowing in a similar to the eyes which is a representation of God. ● Time: This is a recurring aspect of the story, and imperative to understanding Gatsby as a character. Gatsby is constantly trying to recapture the relationship he had with Daisy before he left Daisy five years ago to go on his journey. He also expects Daisy to remain the same girl that she was. This is not the case as she has moved on by getting married and living a luxurious socialite life. ● Inconclusive words: Went, Got, Think This topic seems to focus on the character of Gatsby and the various omniscenient aspects of the story that are ever present and that look upon him.
Topic 2	*TOPIC 3*	● Old: Refers to the ‘Old Money’ socio-economic class. Those that are born with riches through their family-lineage. ● New: Refers to the ‘New Money’ socio-economic class. Those that have gained riches through actions taken in their lives, but were not born with it. ● Inconclusive words: Came, Little, Know, Moment, Going, Want, Knew, Girl Most of the others words in this topic are not as conconclusive however, it does consist of the words that refer to the ongooing contrast between the Old Money and New Money socio-economic statuses.

Topic 3	*TOPIC 4*	Most of the words in this topic could refer to the accident that takes place at the Valley of Ashes, which is where the 'eyes' are situated. The accident is that of Myrtle who is Tom's mistress. Tom indirectly accused gatsby of running Myrtle over with the 'yellow' car to the authorities. In actuality, it was Daisy driving the car with Gatsby at the passenger seat.
Topic 4	*TOPIC 5*	I am not precisely sure what the words are referring to in this topic.

As can be seen in the table above, not all topics have words that are entirely conclusive. If I had not read 'The Great Gastby' in the past, I may not have been able to draw some of the conclusions and connections mentioned above. Thus, the NLP Pre-Processing and LDA could definitely be improved.

Result Visualization

After applying the LDA model, we used the following packages & sub-packages to create a visualization of the results: pyLDAvis and pyLDAvis.sklearn. The visualization is an Intertopic Distance Map (via multidimensional scaling) of the topics. According to the visualization that topics 3, 4, & 5 are related to each other, whereas topics 1 and 2 are distinctly different.



Reflection

If we were to perform a similar study again, we would construct term frequency bar charts to find out the top words before performing LDA on the novel's textual data. Also, to better pre-process the novel's textual data for LDA, we would try to perform POS tagging to remove more unnecessary words, such as pronouns & conjunctions, for further LDA topic result accuracy.

Technology stacks related to Big Data

When performing text mining in the context of Big Data, the Python programming language would still be used. Although, topic modelling would be performed on a whole collection of documents rather than just one. If the size of each document is large as well as the size of the collection of documents itself, performing topic modelling on a local machine may not suffice. Thus, utilizing a cloud service to spin up an ephemeral cluster would be needed. Thus, pyspark would be required to implement Python along with the packages needed for LDA. A data scientist would ideally use the ephemeral cluster to import, perform the unsupervised learning job, save the results, and then shut down the cluster. This would be the case provided the task they want to use the ephemeral cluster for is predefined and planned.

Although, research suggests that there are other versions of LDA that may be more appropriate for a real-world big data scenario. There is a version of LDA called 'Clustered Latent Dirichlet Allocation (CLDA)'. It is "a method for extracting dynamic latent topics from a collection of documents." CLDA is advantageous as it uses data decomposition strategy to partition data. It is also advantageous due to its ability train a model of very large datasets that contain a large number of topics. "LDA is used to infer topics of local segments". The results of CLDA are combined together using clustering to bring together topics from different segments into global topics (Gropp, Herzog, Safro, Wilson & Apon, 2022).

References

Bansal, S. (2016). Beginners Guide to Topic Modeling in Python and Feature Selection. Analytics Vidhya. Retrieved 12 July 2022, from https://www.analyticsvidhya.com/blog/2016/08/beginners-guide-to-topic-modeling-in-python/.

Chibueze, O. (2018). NLP For Topic Modeling & Summarization Of Legal Documents.. Medium. Retrieved 12 July 2022, from https://towardsdatascience.com/nlp-for-topic-modeling-summarization-of-legal-documents-8c89393b1534.

Garry, B. (2022). Entity Salience & Its Implications for SEO | Impression. Impression. Retrieved 12 July 2022, from https://www.impression.co.uk/blog/entity-salience-seo/.

Gropp, C., Herzog, A., Safro, I., Wilson, P., & Apon, A. (2022). Scalable Dynamic Topic Modeling with Clustered Latent Dirichlet Allocation (CLDA). Retrieved 12 July 2022, from https://www.arxiv-vanity.com/papers/1610.07703/

Kapadia, S. (2019). Topic Modeling in Python: Latent Dirichlet Allocation (LDA). Medium. Retrieved 12 July 2022, from https://towardsdatascience.com/end-to-end-topic-modeling-in-python-latent-dirichlet-allocation-lda-35ce4ed6b3e0.

Martinez, J. (2021). The Great Gatsby I Summary, Context, Reception, & Analysis. Retrieved 12 July 2022, from https://www.britannica.com/topic/The-Great-Gatsby

Field, W. (2022). On what page of Fitzgerald's The Great Gatsby does Tom tell Wilson that Gatsby was the one who killed Myrtle?. enotes. Retrieved 19 August 2022, from https://www.enotes.com/homework-help/what-page-fitzgeralds-great-gatsby-tom-tell-wilson-646312.

The Great Gatsby: Jordan Baker | SparkNotes. SparkNotes. (2022). Retrieved 19 August 2022, from https://www.sparknotes.com/lit/gatsby/character/jordan-baker/.

The Great Gatsby: Symbols | SparkNotes. SparkNotes. (2022). Retrieved 19 August 2022, from https://www.sparknotes.com/lit/gatsby/symbols/#:~:text=The%20eyes%20of%20Doctor%20T.%20J.%20Eckleburg%20are%20a%20pair%20of,never%20makes%20this%20point%20explicitly.

The Role Of Time In The Great Gatsby. CRAM. (2022). Retrieved 19 August 2022, from https://www.cram.com/essay/The-Theme-Of-Time-In-The-Great/P3RG2FPNBXYQ.

Helpful Resources

- https://www.geeksforgeeks.org/text-analysis-in-python-3/
- https://www.youtube.com/watch?v=7WfoYI-EPtI&list=LL&index=7
- https://towardsdatascience.com/end-to-end-topic-modeling-in-python-latent-dirichlet-allocation-lda-35ce4ed6b3e0
- https://www.geeksforgeeks.org/python-lemmatization-with-nltk/?ref=lbp
- https://datagy.io/python-remove-punctuation-from-string/
- https://stackoverflow.com/questions/354038/how-do-i-check-if-a-string-is-a-number-float
- https://towardsdatascience.com/end-to-end-topic-modeling-in-python-latent-dirichlet-allocation-lda-35ce4ed6b3e0
- https://www.analyticsvidhya.com/blog/2016/08/beginners-guide-to-topic-modeling-in-python/
- https://stackoverflow.com/questions/33229360/gensim-typeerror-doc2bow-expects-an-array-of-unicode-tokens-on-input-not-a-si
- https://towardsdatascience.com/nlp-for-topic-modeling-summarization-of-legal-documents-8c89393b1534
- https://github.com/chibueze-oguejiofor/Machine-Learning-In-Law/blob/master/project.ipynb