**CMPE 257 - Machine Learning -**

**Natural Language Processing**

**Submitted By**

**Team Name: Team\_Duos**

**Team Members**

**Sakshi Kekre**

**Poojan Shah**

**Project Title:**

Decoding Kipling: NLP Assignments in Language Wizardry

**Project Description:**

This document is a synthesis of the work done for Natural Language Processing (NLP) assignments performed throughout the second half of our course. The tasks involved scraping poems and articles authored by and related to multiple poets (obtaining natural language artifacts), and processing and visualizing them to uncover facts and insights from these.

**Keywords**

Part-Of-Speech tagging, Tokenization, Topic Modeling, Fine-Tuning, Retrieval Augmentation, Knowledge Graph, Poem

**Assignments performed (with links to colab notebook):**

Following sections will describe the input, objective and insights from each of these assignments:

1. NLP POS Substitutions and Tones for Poets (Week 7)

Link:<https://drive.google.com/file/d/1IdHvpawsecXgbS38NC2kDg7HiOw02ch_/view?usp=drive_link>

1. Gold Standard identification and comparison for Pushcart Vs Sample Short Text using sentiments, topics, POS and KG's (Week 9)

Link:<https://drive.google.com/file/d/17e_3UU8b2oV5wQC8_PdUgjoVKx_m4OSL/view?usp=drive_link>

1. Fine-tune a LLM for your Poet (Week 10)

Link:<https://drive.google.com/file/d/14WqCO9iWqprZkM-g5GPxtUah6jXygn0_/view?usp=drive_link>

1. Retrieval Augmented Generation (Week 11)

Link: <https://colab.research.google.com/drive/12NO2P_keaKfkzKtCtQKde62i7Ps3tng2?usp=drive_link>

1. Mining News Articles and Assembling a Knowledge Graph (Week 13)

Link:<https://drive.google.com/file/d/1nbaJ8m3m9SWub5i4ix-zdFimGmC938Vu/view?usp=drive_link>

# #1 NLP POS Substitutions and Tones for Poets

## **Objective:**

To explore the beauty of poetry and dive into its linguistic nuance through NLP with the help of techniques like Part-Of-Speech tags, swap verb, adjective, and noun between poets. Then, unveil concise summaries and unravel the poetic tapestry with some topic modeling magic.

## **Input:**

## Natural Language Processing (NLP) techniques were applied into the poetry of Rudyard Kipling and Alfred Lord Tennyson.

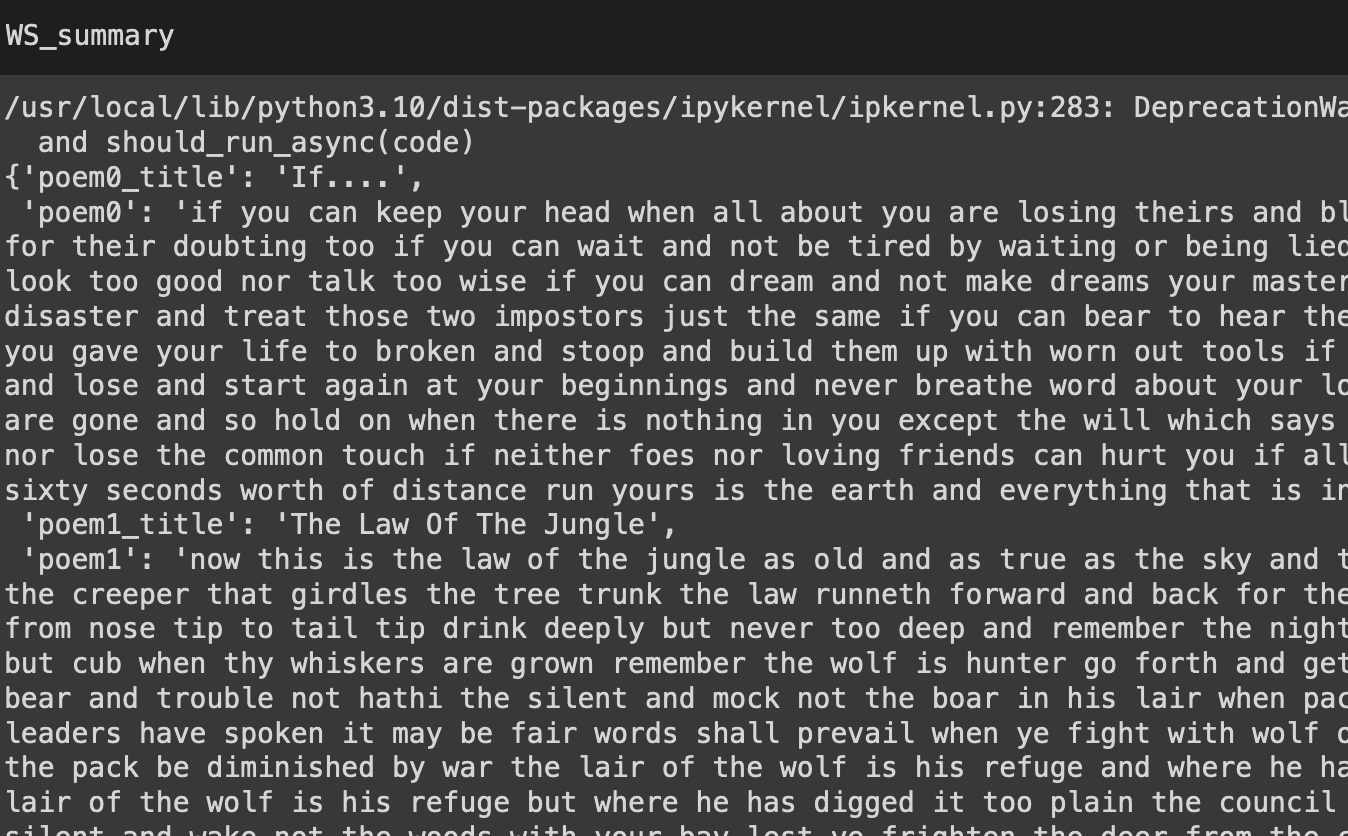
1. Performing Summarization and Topic Modeling on the poem to unveil primary subjects or themes in the poem

## **Experiments:**

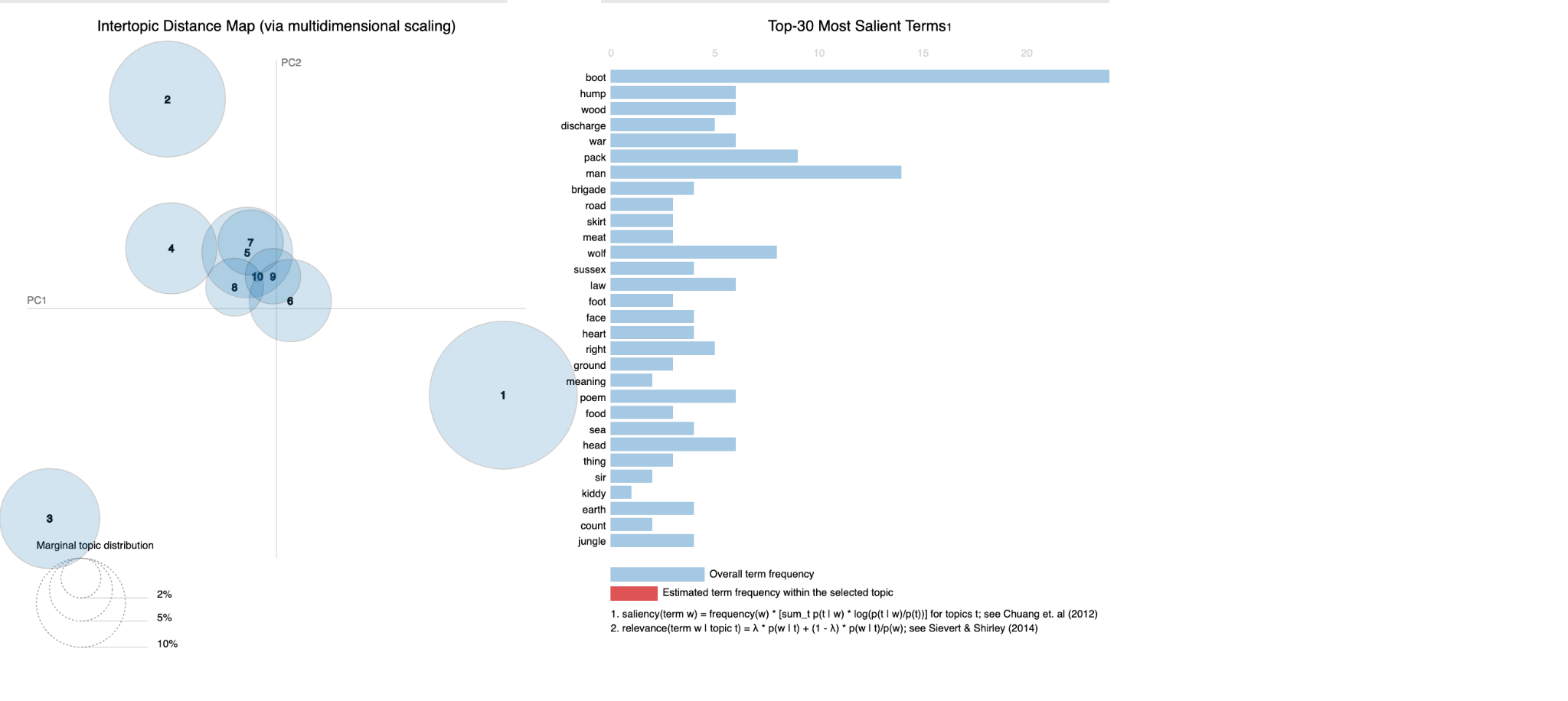
1. We gathered 2 poems each from Rudyard Kipling and Alfred Lord Tennyson from allpoetry.com using the beautifulSoup library. We saved structured data in a JSON file, saving poet names, poem names, and texts in a team folder.
2. We conducted two distillations exploring POS and embeddings to identify stylistic similarities between poets. Extracted and stored POS, adjectives, nouns, and verbs in a JSON data structure for comprehensive analysis.
3. We used NLP techniques, first, to assess the likeness between newly crafted poems and original works by Kipling and Tennyson. Then, transposed POS Based on Semantic Similarity: Swapped POS of Poet A with Poet B using semantic similarity. Saved transposed poems (Poet1-Poet2-PoemName.txt) for both poems of each poet pair.
4. Applied summarization techniques to distill the core themes and emotions from the poems. Evaluated the effectiveness of summarization in capturing the essence of each poem concisely, providing valuable insights.
5. We applied LDA to uncover primary subjects or themes in the poems. Utilized BERTopic to capture nuanced topics and context, enhancing the understanding of underlying themes.

## **Output:**

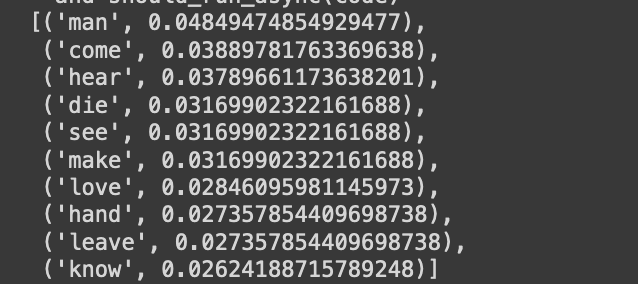
Output of summarization:



Topic modeling:



Output of BERT:



## **Insights/Conclusion:**

## Exploring Poetic Similarity: Evaluating the likeness between newly crafted poems and original works by other poets is key. This analysis was vital to ensure the new poems match the style and themes of the chosen poet's compositions.

## Condensing Text for Clarity: Summarizing each poem distilled its core themes and emotions, providing a quick peek at the poem's essence without going through the entire text.

## Unveiling Main Themes: Utilizing methods such as Latent Dirichlet Allocation (LDA) for topic modeling revealed the primary subjects or themes in the poems. This technique uncovered underlying themes that might not be immediately obvious.

## 

## 

## 

# #2 Gold Standard for Pushcart Poems

## **Objective:**

Utilize NLP techniques, such as sentiment analysis, topic modeling, and POS tagging, to compare and analyze both sets. Generate knowledge graphs, statistical plots, and narrative insights to understand distinctions and establish a baseline for ranking new poems against the gold standard.

## **Input:**

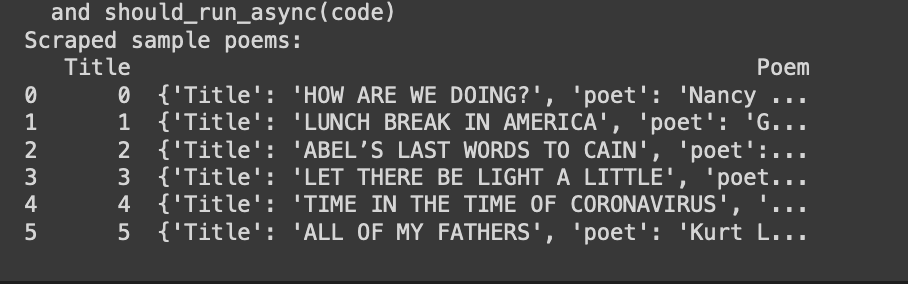
1. Two sets of poetry datasets - one comprising Pushcart-nominated or awarded poems sourced from various websites and the other consisting of non-nominated poems from different platforms.
2. JSON repositories are created for each poet, capturing poems, POS, and topics.

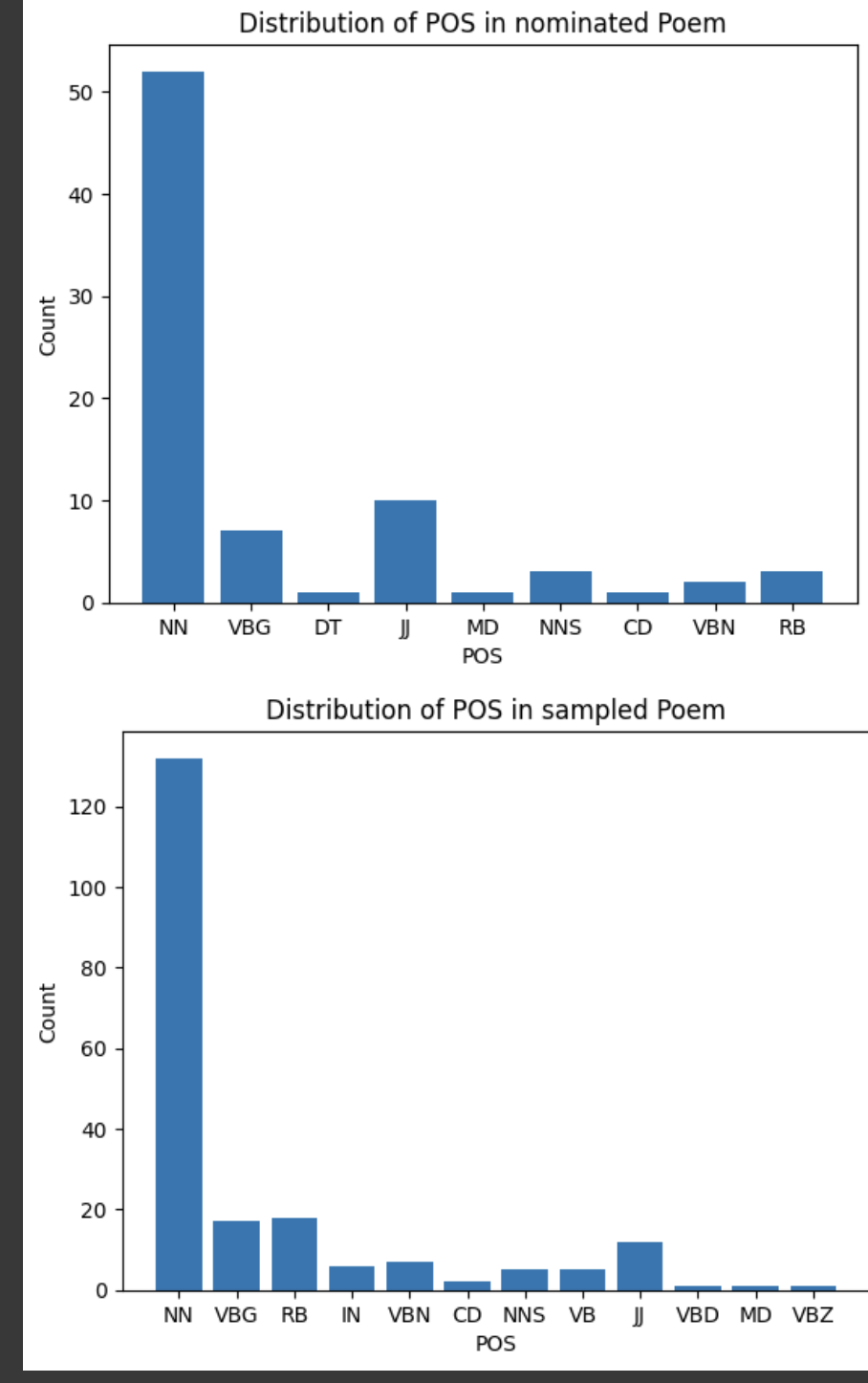
**Experiments:**

1. Collect Pushcart and sample Data (Web scraping)
2. Plot POS distributions: Plotting POS distribution involves graphically depicting the frequency of different parts of speech like nouns, verbs, adjectives in a text. In this analysis, it helps compare linguistic patterns between Pushcart-nominated (gold standard) and sampled poems, revealing distinctions that contribute to a nuanced evaluation of poetry.
3. Gold std generation for PK poems: POS, KGs
4. Plot POS Distribution: preprocesses poems by tokenizing, removing stop words, and lemmatizing. It extracts part-of-speech (POS) tags, counts their occurrences, and stores distributions for Pushcart and sample poems.
5. Knowledge Graph Generation: spaCy to extract nouns and verbs from poems, creating triples that represent relationships. It then constructs knowledge graphs for the first 3 poems by adding nodes for nouns and connecting them with verbs, producing a visual representation of the semantic structure and flow within the text.

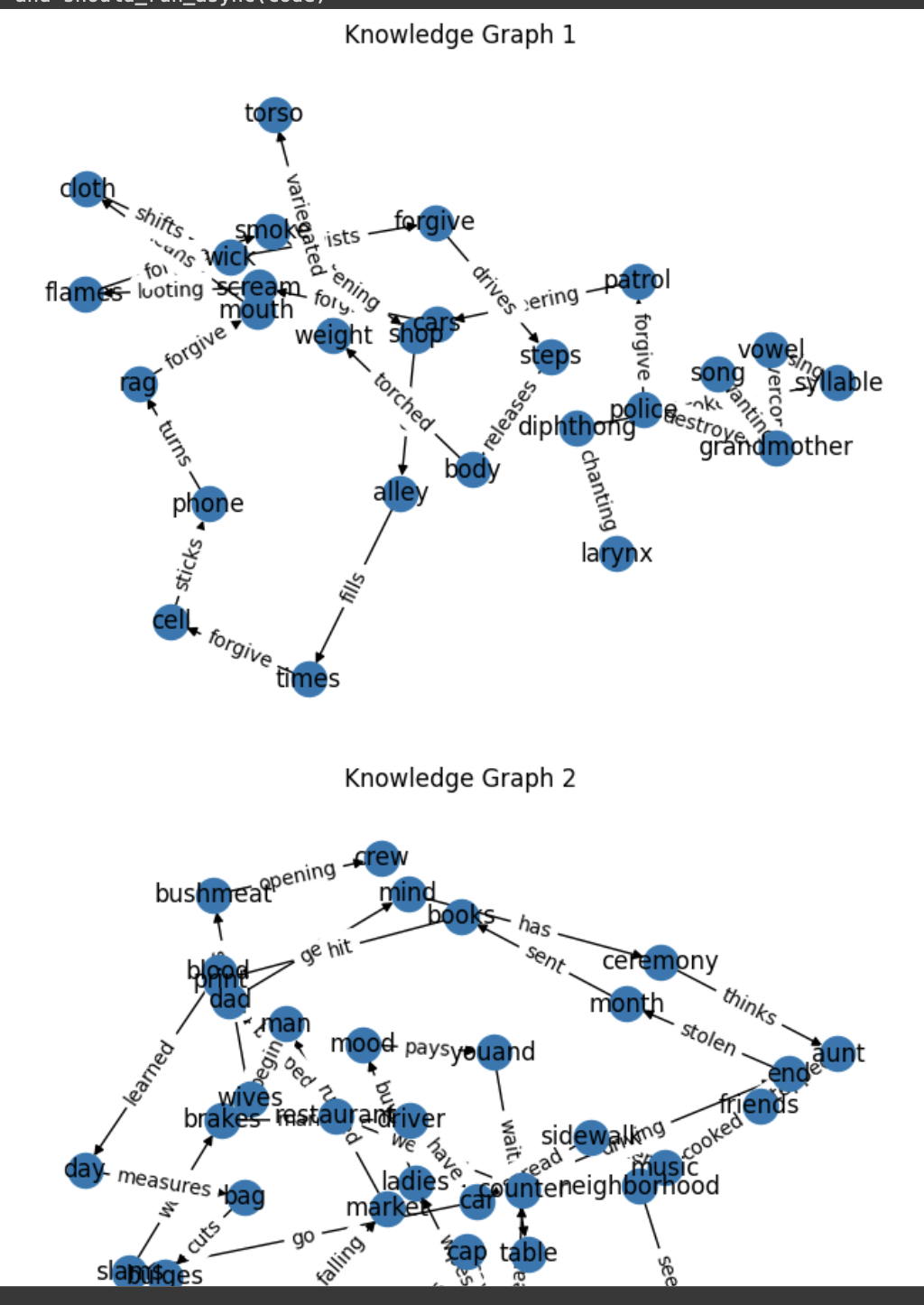
## **Output:**

Below is the list of collected Pushcart and sample Data (Web scraping)



Result of POS distribution:  


Knowledge Graph generation:



## 

## **Insights/Conclusion:**

1. From observing noun POS and performing topic analysis, we determined that the pushcart nominated poems, specifically the established gold standard, are/is rightly so because they mostly speak to the emotional quotient of the reader and they cover deeper topics that compel reader to be in touch with their emotions.
2. On the other hand, sampled poems have a more generic topic base such as day to day things. It may be so that the 'machine' fails to identify the metaphorical relationship between these day-to-day things and human emotions, but that is beyond our scope of work for now.

## 

## 

# #3 Fine-tune a LLM for your Poet

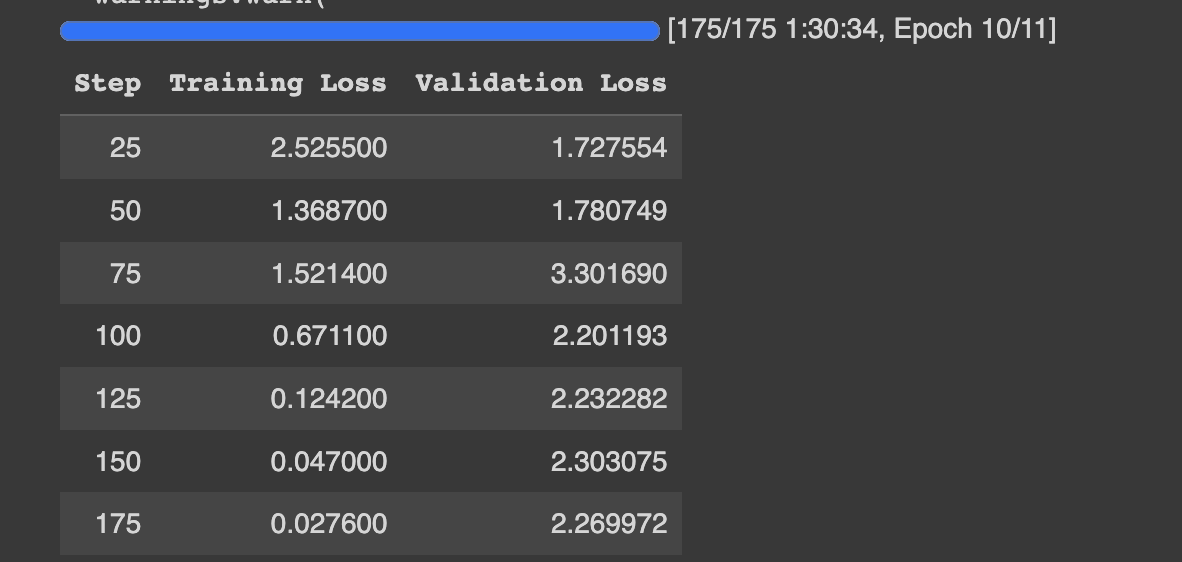
## Objective:

The objective of this assignment was to understand the fine-tuning process of a Language Model and to tune the model to generate content like our chosen poet.

## Input:

1. We selected Rudyard Kipling as our poet of choice, providing us with the chance to delve into the works of a Nobel Prize-winning poet.
2. We scraped around 40 poems from AllPoetry.com using BeautifulSoup library
3. For fine-tuning, we opted for the Casual Language model from Mistral AI, as it met the processing requirements for our specific use case.
4. We also utilized sharding to enhance the speed of model loading and training.

## Experiments:

1. For the fine-tuning of the model, the scraped poems were converted int the required json lines format.
2. The lines were then split into training and validation dataset.
3. After the data pre-processing, we loaded the base model ‘bn22/Mistral-7B-Instruct-v0.1-sharded’ in session for the fine-tuning.
4. We tokenized the dataset and observed the distribution of input
5. We then prompted the base model with a string we would use later to compare the outputs before and after training.
6. We then trained the model with the training dataset of poems.
7. We saved the model locally to google drive.
8. Later, using huggingface cli, we uploaded the model and tokenizer to HuggingFace model hub which got committed at below location:  
   <https://huggingface.co/7pjshah/rudyard_mistral-sharded/commit/72556df24d8d0b72058745d2065e359453ecb893>

## Output:

We prompted the fine-tuned model with several similar and different prompts to see if the model generated the poems in the style of our poet, Rudyard Kipling.

Here is one of the results:

Prompt:

| The following is a Poem by Rudyard Kipling the Poet, about the story of the origin of Vampires : # |
| --- |

Output:

| The following is a Poem by Rudyard Kipling the Poet, about the story of the origin of Vampires : #  There was a man who lived in the Caucasus,  Where the peaks are Killarney-like and blue,  And the valleys are Babadag-like and red.  He had a son who was born at midnight,  And a mother who died when he was nine. |
| --- |

And below is the result from the base model before the tuning:

Prompt:

| The following is a Poem by Rudyard Kipling the Poet: # |
| --- |

Output:

| The following is a Poem by Rudyard Kipling the Poet: # 1  The light that failed me in my youth,  Had nothing to do with the stars;  It was the darkness of my father's heart  That shut my eyes to all his charms. |
| --- |

## Insights and Conclusion:

1. While the fine-tuned model generated coherent text, the poems do not semantically make complete sense. They also seem to be cut short before giving out all of the sentences.
2. The model sometimes provided an empty output, which could signify that it needed more training and more data to do the training on.
3. One interesting thing that we observed was that the model did pick up on the poet’s writing style as seen in things like the names used for places or adjectives (see highlighted words below):

| There was a man who lived in the Caucasus,  Where the peaks are Killarney-like and blue,  And the valleys are Babadag-like and red.  He had a son who was born at midnight,  And a mother who died when he was nine. |
| --- |

1. To conclude, upon comparing the results before and after the fine-tuning, it can be inferred that training the model on as little as 40 poems is not sufficient to generate human-like poems.

One way to confirm this could be to try out with a larger training and evaluation dataset to see the responses of the model.

# #4 Retrieval Augmented Generation

## Objective:

The objective of this assignment was to understand the functionality of Retrieval-augmented generation (RAG) AI framework to improve the quality of responses of our fine-tuned Rudyard model.

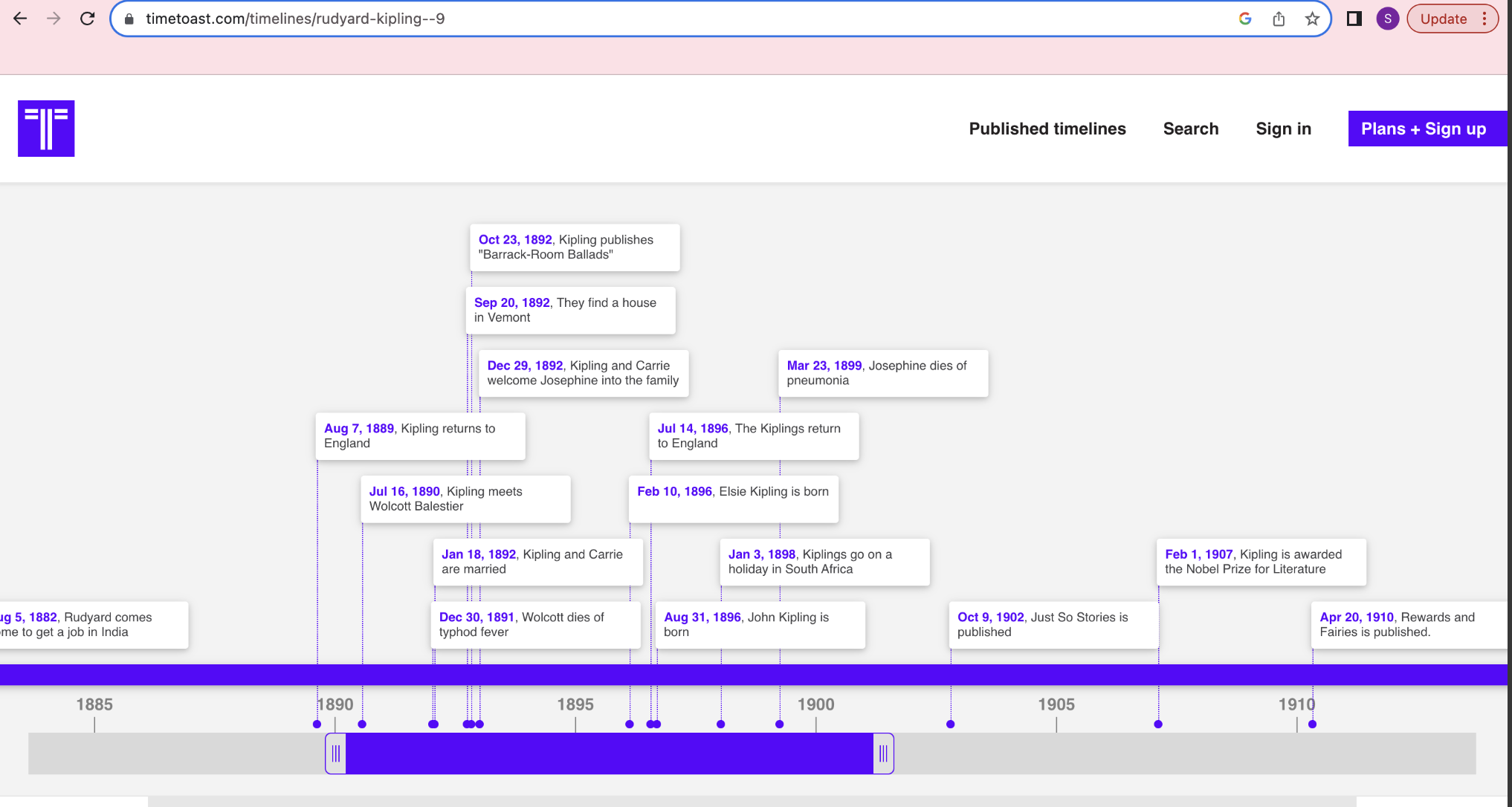
With the use of RAG, we provided our model with a supplemental source of information to augment its internal information.

## Input:

1. The model that was fine-tuned in the previous (Fine-tune LLM) assignment
2. Weaviate cluster for implementing RAG on our model
3. The input supplemented to the model in the form of Weaviate vectorDB were:
   1. Poems generated by our Rudyard model
   2. News articles that spoke about the topics or issues mentioned in the poems

## Experiments:

1. We installed all the required libraries and frameworks such as transformers, peft accelerate, bitsandbytes, safetensors, sentencepiece, streamlit, weaviate-client, langchain, sentence-transformers, tiktoken and more.
2. We loaded our tuned model and tokenizer from the HuggingFace model hub.
3. We created a cluster on Weaviate and used the API key and cluster URL in our code to create a Weaviate session for the following processes.
4. To enable our model to capture semantic relationships and context of our poem and article text, we created embeddings using an existing HuggingFace model ‘sentence-transformers/all-mpnet-base-v2’.
5. Looking at Rudyard's timeline, years 1880-1900 seem most impactful given the major life experiences (moving to India, moving to England, birth of progeny and death of multiple loved ones) that happened in his life. These experiences must certainly have had an influence on his writing. Therefor, we shall scrape news articles from this time period and created a Weaviate VectorDB using these articles.



1. We passed this vector to a conversation chain along with our fine-tuned model and prompted the chain with a question about the article.

1. We prompted the same question to another conversation chain that did not have the news article vector and compared the results.

## Output:

Below are the outputs from both the conversation chains:

| | Without RAG | | --- |   The poem Mandalay is a collection of verses which reflect Kipling's experiences and observations about British colonialism. The poem's setting is Mandalay, a city in Myanmar (formerly known as Burma). The poem is narrated by a first-person speaker who recounts his experiences as a British soldier stationed in Mandalay. The speaker is nostalgic for his past life as "a first-class fightin' man" and regrets the loss of his certificate, a document given to soldiers who prove themselves in battle. The poem also reflects Kipling's critique of colonialism and the negative impact it has on both the colonized and the colonizing powers. |
| --- | --- |

| | With RAG (news) | | --- |   The poem "Mandalay" by RudyardKipling is not a direct commentary on the events that occurred in his life or in India between 1865 and 1890. However, the events and experiences Kipling had in India and his personal life in the mid to late 19th century did significantly influence the themes and language of Mandalay. The poem is a nostalgic reflection on a time of youth and adventure in a fantasy, idyllic, and romanticized Burma (now Myanmar) which is a contrast to the reality of the time, where Kipling was dealing with malaria and dysentery. The poem also reflects the British Raj and the Orientalism prevalent in the Victorian era, where the British viewed the people of the East as "savage" and "heathen". The poem also shows Kipling's fascination with the Orient, the allure and danger of the "Other", and the dangers of imperial expansion. |
| --- | --- |

## Insights and Conclusion:

1. Looking at the QA chain responses in general, we could confirm our initial understanding that the events in the years 1880-1900 had a significant impact on Rudyard's life.
2. Reviewing the differences in outcomes from the previous section, when we implemented news RAG with our model, the prompts became more focused with the addition of factual information obtained from the scraped news articles.
3. We therefore concluded that RAG aids in improving the responses by grounding the information in confident sources like news articles.

## 

# #5 Mining News Articles and Assembling a Knowledge Graph

## Objective:

The objective of this assignment was to understand how NLP tools can help us determine the similarity of a poem and an article related to the poem using knowledge graphs.

## Input:

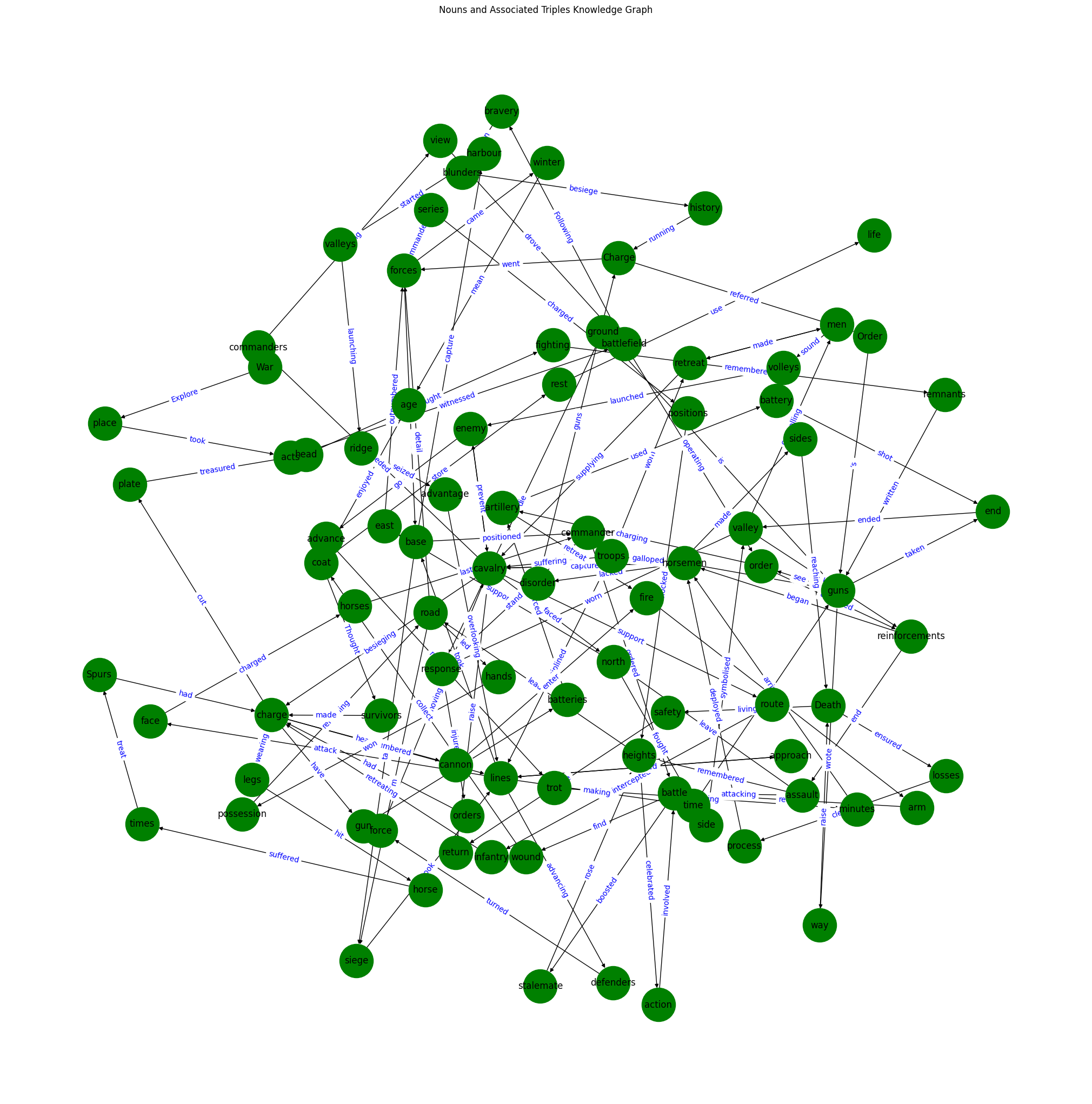
1. We used the following:
   1. the poem - The Last of the Light Brigade - by Rudrard Kipling (reference: allpoetry.com)
   2. the article - Battle of Balaklava, which gives information about the Balaklava battle that the poet talks about in his poem (reference: [link](https://www.nam.ac.uk/explore/battle-balaklava))

## Experiments:

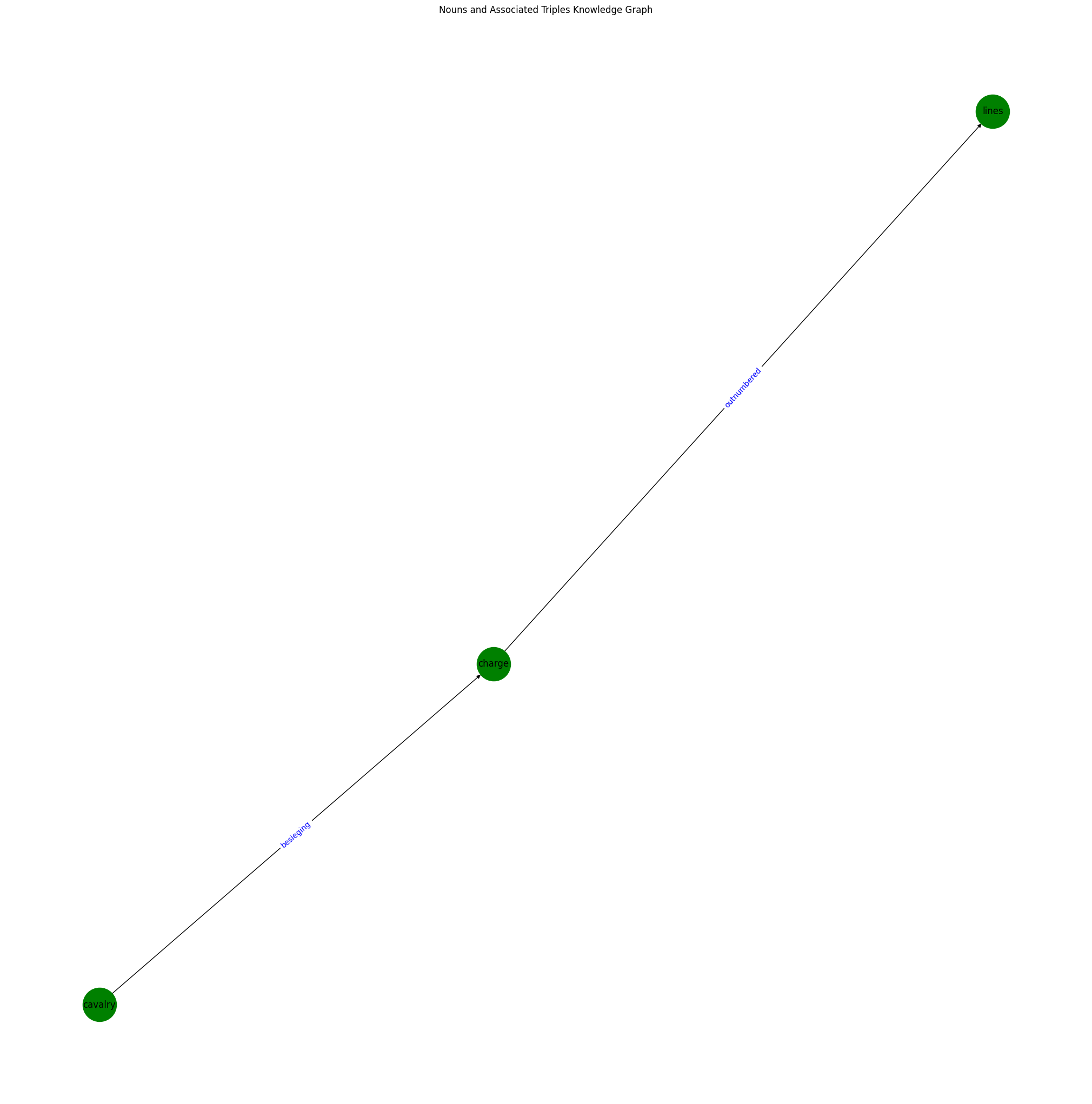
1. We installed the required libraries such as spacy, networkx and more.
2. We scraped the poem and article using the BeautifulSoup library.
3. We created utility functions to create and plot graphs and noun-verb-noun triples to keep the code in our notebook manageable.
4. We created triples for the poem and articles and plotted knowledge graphs (KG) for both of these.
5. We observed that the KG for the article was not legible due to overlapping nodes and labels.
6. Therefore, we calculated the degree centrality of all the nodes in our article KG and listed them in the order of ranking from highest to lowest.
7. We then re-plotted the KG using the most important nodes in the graph to observe the relation between POS of the poem text.
8. We calculated the cosine similarity of the initial article KG and the poem and compared it with that of the similarity of article KG of only important nodes and the poem

## Output:

Initial Article KG:



KG with 4 most important Degree Central nodes:



The cosine similarities are as follows:

| Cosine Similarity of poem with article: 0.010718661571406801  Cosine Similarity of poem with top degree central nouns of article: 0.08838834764831843 |
| --- |

## Insights/Conclusion:

1. The cosine similarity of the two knowledge graphs - Poem and Article - is not very high.
2. However, with trivial human analysis, we can see that the article most certainly speaks about the topics (Battle of Balaklava) which the poet has written about in the poem. This indicates that knowledge graphs are not the most appropriate tool in our use case; topic modeling might (or might not) have been the better alternative.
3. A point worth noting is that after eliminating the nodes with low degree centrality, the cosine similarity of the graphs increased. But, surprisingly, this behavior is not linearly associated. Increasing the number of top degree central nodes did not linearly decrease the similarity. Top 4 nouns yielded the second highest similarity of ~0.08.
4. It would be an interesting future scope to try out more combinations of triples or try out subject-verb-object triples instead of noun-verb-noun triples for the graphs.