Topic Modeling and Sentiment Analysis of Wikipedia-Style Text

# GitHub Page: <https://github.com/Zan5ki/TopiSent>

# Introduction

It’s useful to model topics that pertain to a random collection of documents, if not to accurately describe them then to simply probe for potential insights. Once classified, however, is it possible to apply sentiment analysis to these documents in order discover whether they pertain to positive, negative or neutral material? What does the “sentiment” surrounding encyclopedic/technical documentation look like in the first place, and is it useful to analyze the product of sentiment analysis on such documents?

# Literature Review

# In this review, we will be exploring the relevant literature surrounding the natural language processing (NLP) technique of text classification. For the purposes of this project, we will be focusing on two common areas (or “tasks”) related to text classification: topic modeling and sentiment analysis. For topic modeling, we will investigate Latent Dirichlet Allocation (LDA). For sentiment analysis, we will investigate the Valence Aware Dictionary and sEntiment Reasoner (VADER) tool.

Topic models are algorithms for discovering the main themes that pervade a large and otherwise unstructured collection of documents. “Probabilistic Topic Models: A focus on graphical model design and applications to document and image analysis” by David Blei, Lawrence Carin, and David Dunson, states: “Latent Dirichlet allocation (LDA) is a hierarchical probabilistic model used to decompose a collection of documents into its salient topics, where a “topic” for LDA is a probability distribution over a vocabulary [15]. LDA and its relatives are called probabilistic topic models.” [[1]](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4122269/)

VADER (Valence Aware Dictionary and sEntiment Reasoner) is a lexicon and rule-based sentiment analysis tool that is specifically attuned to sentiments expressed in social media. [[2]]((https:/medium.com/analytics-vidhya/simplifying-social-media-sentiment-analysis-using-vader-in-python-f9e6ec6fc52f) In this project we will assess its suitability for encyclopedic/technical material. According to its authors: “Interestingly, using our parsimonious rule-based model to assess the sentiment of tweets, we find that VADER outperforms individual human raters (F1 Classification Accuracy = 0.96 and 0.84, respectively), and generalizes more favorably across contexts than any of our benchmarks.” [[3]]((https:/www.aaai.org/ocs/index.php/ICWSM/ICWSM14/paper/download/8109/8122)

The following blog posts were also used as reference and to write the code for this analysis:

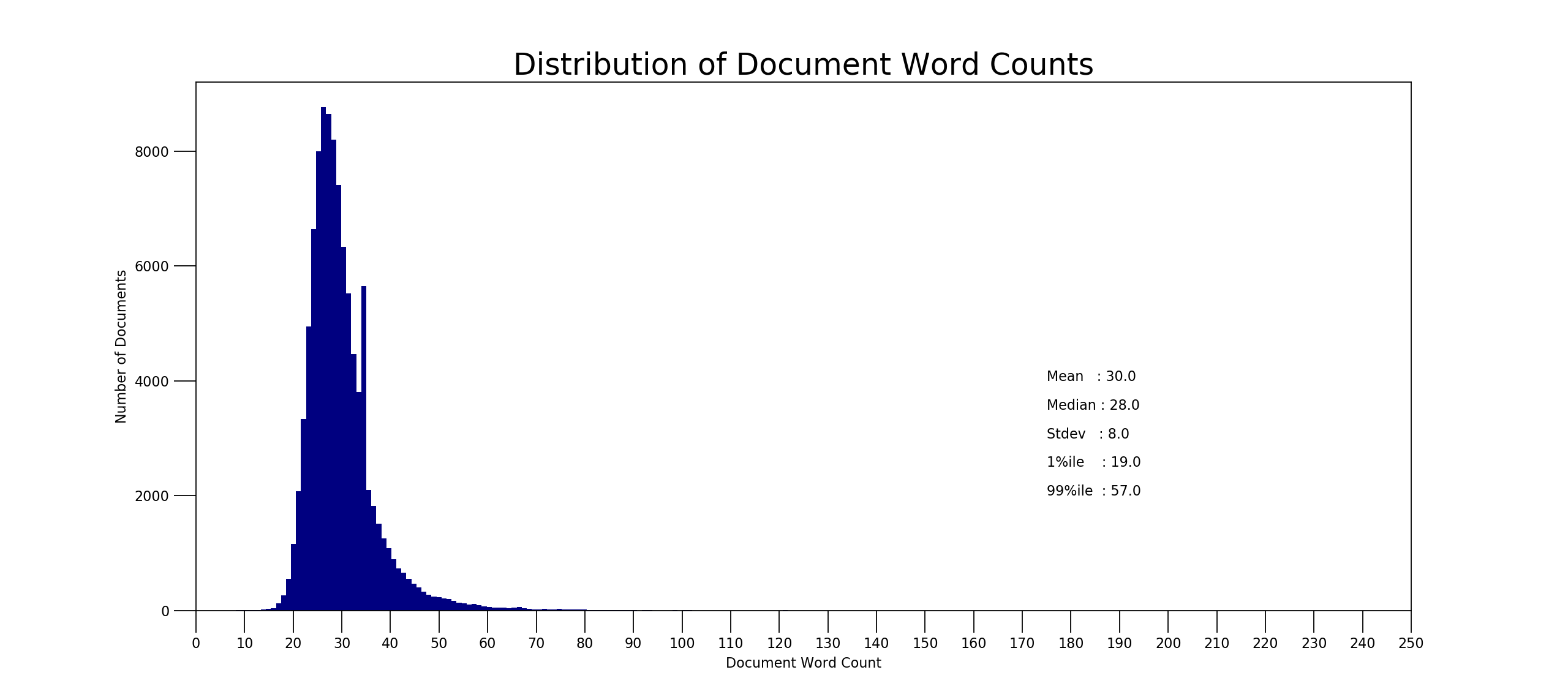
[Topic Modeling and Latent Dirichlet Allocation (LDA) in Python](https://towardsdatascience.com/topic-modeling-and-latent-dirichlet-allocation-in-python-9bf156893c24) [4]

[An Introduction to t-SNE with Python Example](https://towardsdatascience.com/an-introduction-to-t-sne-with-python-example-5a3a293108d1) [5]

[Topic modeling visualization – How to present the results of LDA models?](https://www.machinelearningplus.com/nlp/topic-modeling-visualization-how-to-present-results-lda-models/) [6]

# Dataset

The following dataset was used for this project: <https://www.kaggle.com/mikeortman/wikipedia-sentences>. The only curation performed was trimming the data by the character length of each sentence (only sentences with at least 300 characters were retained). This amounted to 102,100 total sentences (down from 7.8M). The word count of each sentence (post-processing) can be seen in the figure below.



Pre-processing involved removing all special characters, numbers, URLs, stop-words, and finally stemming and lemmatizing.

# Approach

**LDA Processing**  **VADER Sentiment Processing**

# LDA Processing – [GitHub Page](https://github.com/Zan5ki/TopiSent/blob/master/WikipediaTopics%20(gensimlemstem).py)

## Step 1: Pre-process data (lines 25-69)

Create required functions. Remove all special characters, numbers, URLs, and stop-words. Stem and lemmatize remaining text

## Step 2: Determine optimal number of topics (lines 70-165)

Create dictionary and bag of words corpus. Keep only words in 50% or less of total documents and appearing in at least 15

Measure coherence across increasing number of topics (start at 2 and increase by 6 until reaching 38 topics. Plot to determine where marginal coherence flattens

## Step 3: Set up and train LDA model (lines 166-191)

Train and save LDA model. Export pyLDAvis model visualization

## Step 4: Run model on training and test datasets (lines 192-244)

Create required functions and use to apply model to training and test datasets. Export results to file

## Step 5: Extract topic results metrics (lines 245-402)

Extract topic word count distributions, wordclouds, and dataset t-SNE distribution

# VADER sentiment processing – [GitHub Page](https://github.com/Zan5ki/TopiSent/blob/master/WikipediaTopicsVADER.py)

## Step 1: Pre-process data (lines 8-47)

Create required function. Remove special characters, URLs, and numbers from the training and test datasets created during first process

## Step 2: Run VADER Sentiment Analysis (lines 48-60)

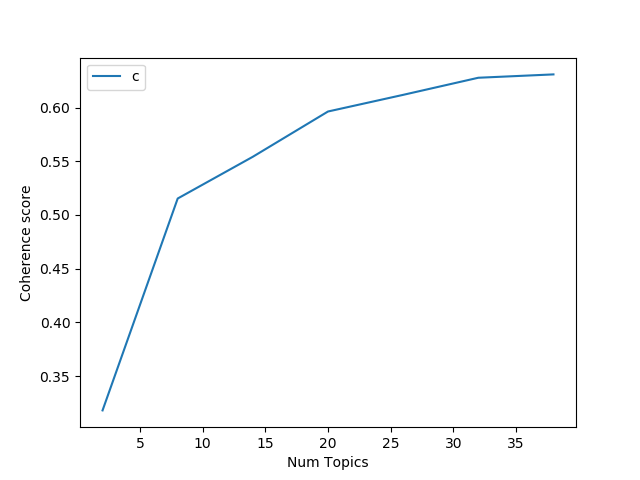
Create required function and run over training and test datasets

## Step 3: Combine VADER and LDA results (lines 61-97)

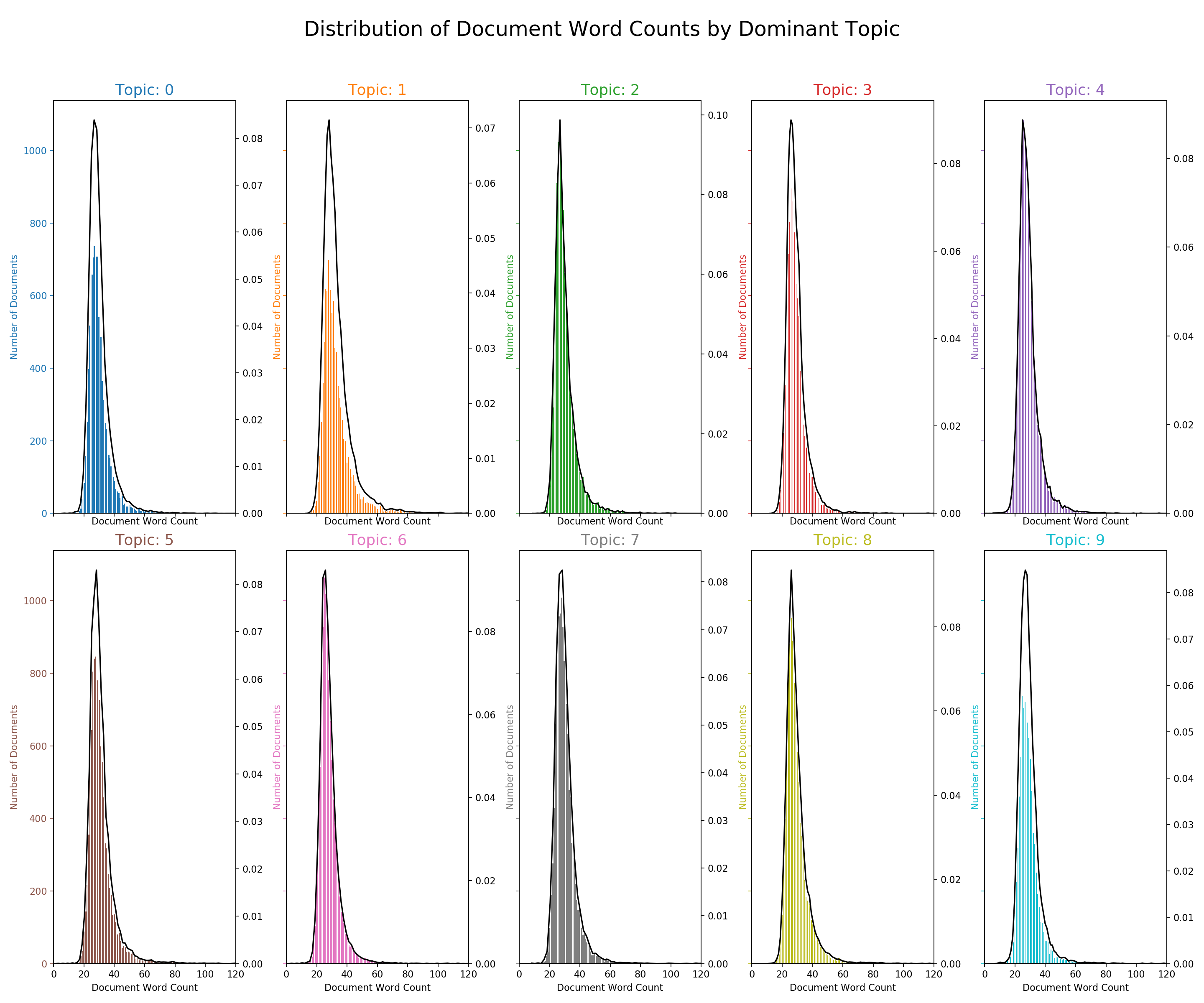
Stitch VADER results to LDA results for training and test datasets and export data

# Results

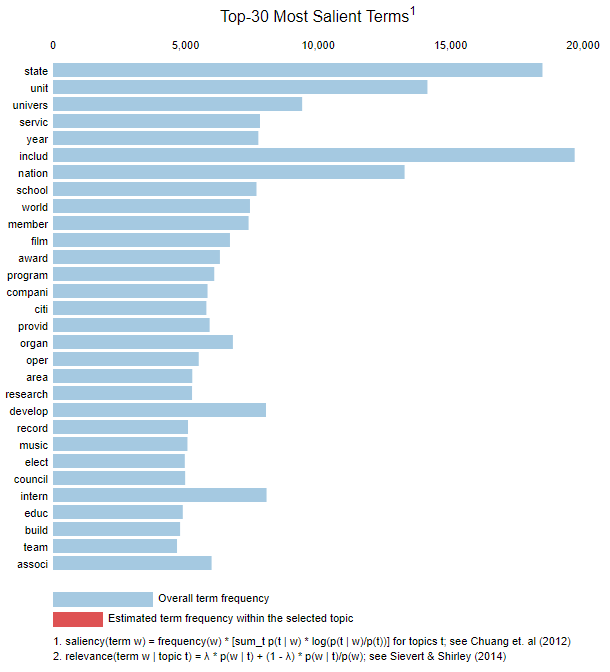
The results of modeling coherence scores across an increasing number of topics (see figure below) suggests that model coherence for our training dataset drops off after 8 topics.



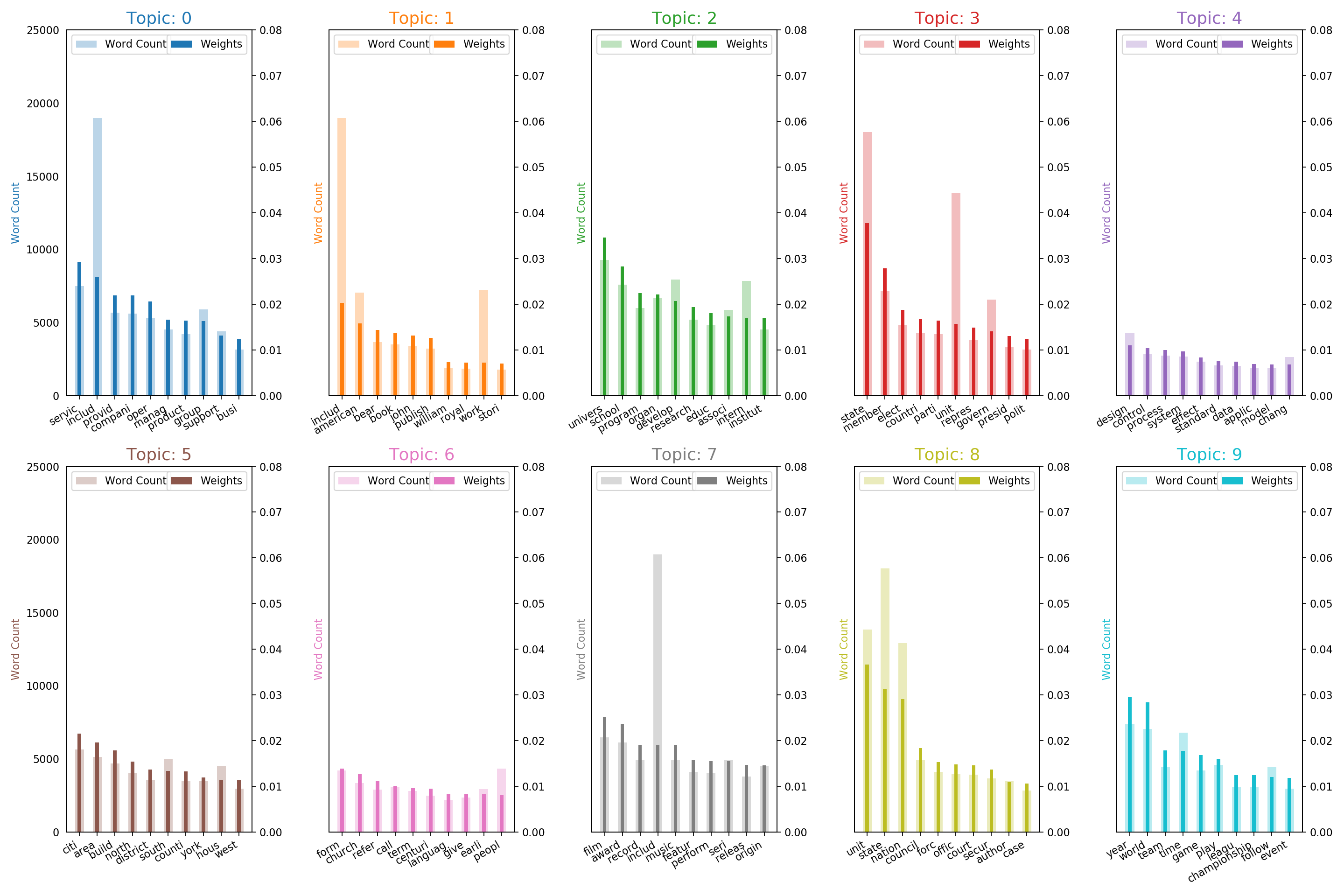
Considering the large number of documents and wide breadth of the material covered by Wikipedia, a 10-topic length was selected for the project model. The word count distributions for each model are pictured below and suggest a fairly similar word count profile.



The 30 most prominent and defining terms are pictured below. As we can see, the terms ‘include’, ‘state’, ‘unit’, and ‘nation’ are by far the most prevalent.



Next, we take a look at the profile of individual terms in each topic (see figure below). Here we can see that certain topics contain outliers in terms of the word count and weighting of individual terms, while others have more evenly distributed term counts and weightings. As already identified, ‘includ’, ‘state’, and ‘unit’ appear at a high rate across many topics, while ‘nation’ appears at a high rate in only one.



We can also visualize term frequency within topics using wordclouds (see below).



After some manual investigation, the word collections identified by the model seem to suggest the following topic categorizations (we refer to ‘Topic 0’ as ‘Topic 1’ and so on in this report going forward):

Topic 1: Commerce

Topic 2: Literature/People

Topic 3: Education/Research

Topic 4: Politics/Government

Topic 5: Science/Technology

Topic 6: Geography/Locations

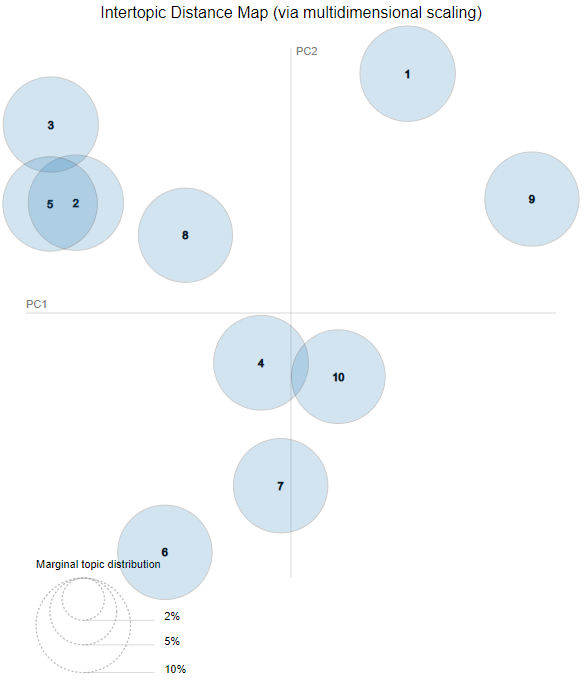
Topic 7: Religion/History/Language

Topic 8: Music/Film/Entertainment

Topic 9: Military/State Authority

Topic 10: Sports/Competition

Finally, we visualize our model using pyLDAvis and t-SNE clustering of the topics (see below).



Map Legend:

Topic 1: Military/State Authority

Topic 2: Education/Research

Topic 3: Science/Technology

Topic 4: Geography/Locations

Topic 5: Commerce

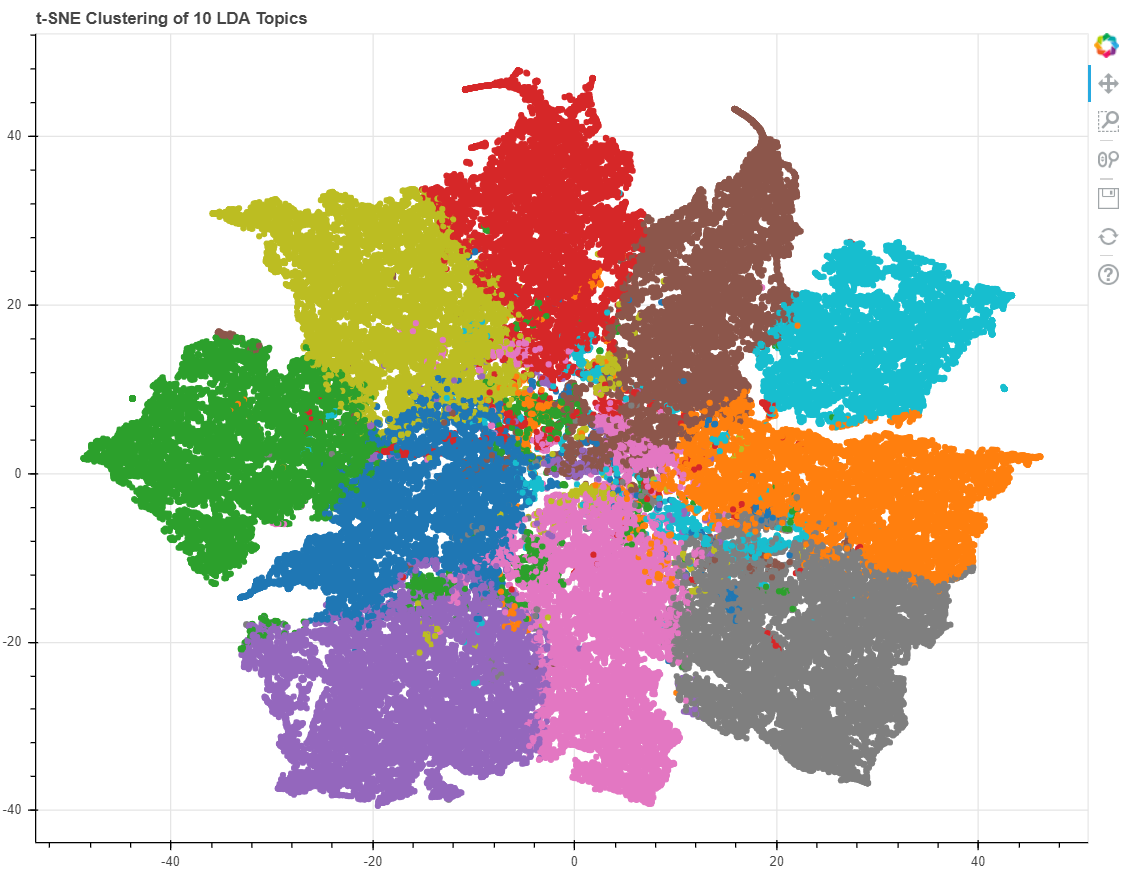
Topic 6: Music/Film/Entertainment

Topic 7: Literature/People

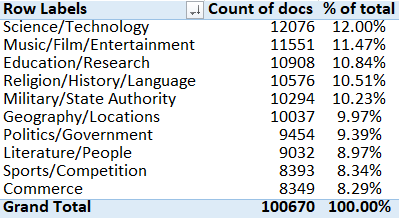
Topic 8: Religion/History/Language

Topic 9: Politics/Government

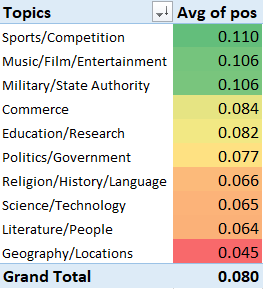
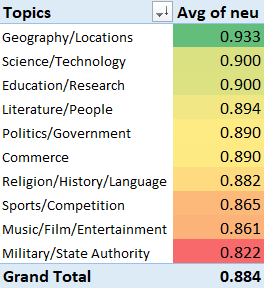
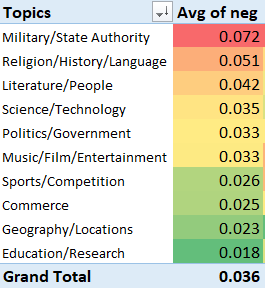
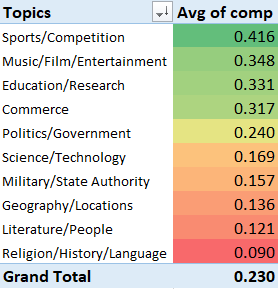
Topic 10: Sports/Competition



These two maps seem to suggest that our topics are well separated from a purely technical standpoint. They serve as fairly strong confirmation of the distinctiveness between topics we observed during manual investigation. The composition of our overall training corpus is as follows:

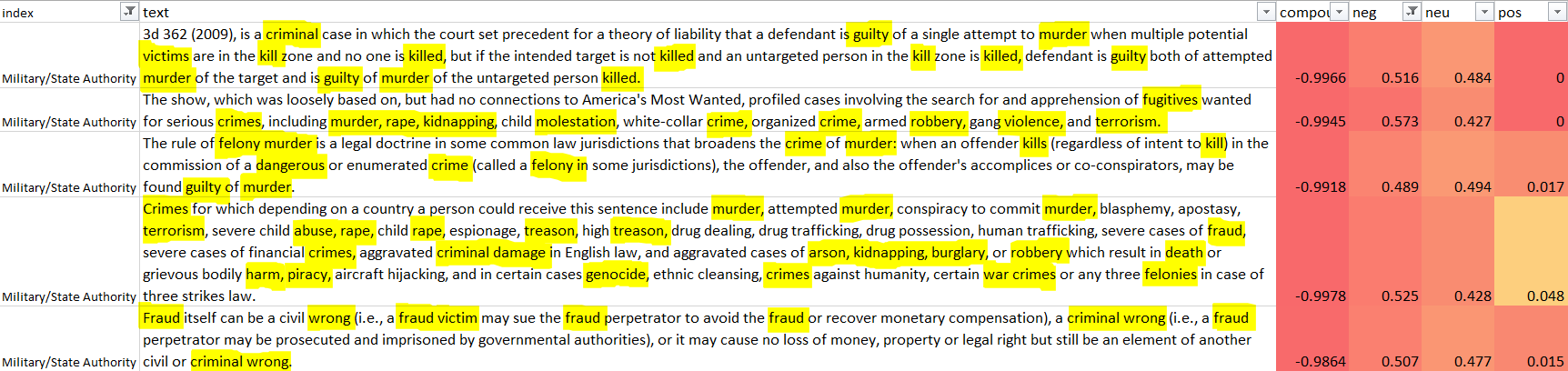


Now that we have identified the topics in our dataset, it’s time to analyze the sentiment of each document and look for insights. Our VADER results can be seen below, with each figure showing the ranked sentiment in each sentiment category (compound/overall, negative, neutral, and positive).

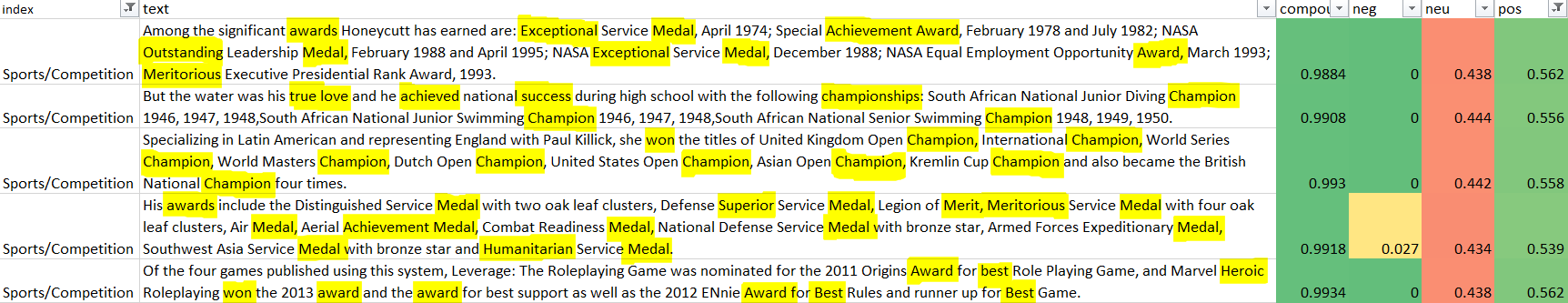


We can notice immediately that the overall composition of the corpus is extremely neutral (total avg = .884) with very low levels of positivity (total avg = 0.080) and even lower levels of negativity (total avg = 0.036). This is somewhat expected considering that we are dealing with encyclopedic/technical documents that are by their very nature intended to written without passion or bias. We do, however, see a few interesting outliers. For instance, the ‘Military/State Authority’ topics appears to be an outlier with respect to overall sentiment polarity, being overly negative relative to the rest of the corpus as well as overly positive, making it the least neutral by far. ‘Sports/Competition’ appears to be an outlier on the positive end, exhibiting high relative positivity and low relative negativity. ‘Geography/Locations’ appears to be the most neutral, exhibiting the highest neutrality rating to go with extremely low negativity and positivity ratings.

While sentiment classification at the topic level is somewhat interesting, it’s unclear what is actually being conveyed by this information and what it represents. We need to look at individual documents to get a better understanding of what the sentiment classifications really mean on this data. Below are the 5 most negative documents from the most negative category (‘Military/State Authority’) according to our sentiment analysis. As we can see, these documents appear to deal with subjects that are generally viewed as negative by most individuals. Murder and various other criminal offences as well as the term ‘crime’ and other negative subjects such as ‘victim’ and ‘guilt’ pop up very frequently.

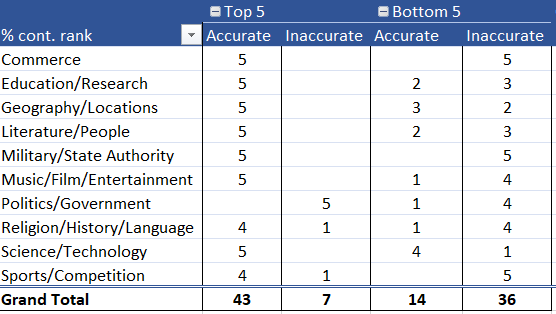


Next, we take a look at the 5 most positive documents from the most positive category according to our sentiment analysis (‘Sports/Competition’).



Here we see many terms that most individuals would deem positive, such as ‘award’, ‘merit’, ‘champion’, ‘won’, ‘best’, ‘medal’ etc.

While the document corpus was far too large to classify manually in order to perform a proper confusion matrix analysis, the ‘Topic\_Perc\_Contrib’ (Topic Percent Contribution) metric proved to be a fairly reliable indicator of how accurate the model was when classifying topics for each document. This is a trend that was noticed when manually analyzing the individual documents but further evidence of the usefulness of this metric was observed by taking the top and bottom 5 documents from each topic in terms of percent contribution score and assessing the accuracy of each classification for these records. As seen below, the top 5 of each category generally performed well and bottom 5 generally performed poorly, with some disparity among the topics. The overall coherence score of the model was .565, which is fairly high from a technical standpoint.



# Conclusions

According to this analysis, it appears that the sentiment analyzer picks up on positive and negative events or human behaviours in these mostly neutral documents. While the VADER technique is most attuned to social media data, we see evidence in this report that it can still be used to judge positivity/negativity of the general topic material in Wikipedia-style documents rather than simply relying on sentiment which stems from human expression (bias, emotion, passion etc.).

While there does appear to be some usefulness in using the sentiment metrics provided by the VADER analysis for analyzing these types of documents, it’s important to note that these metrics were only useful on a relative basis. If looked at in absolute terms, the VADER metrics conveyed very little positive or negative sentiment at all. Again though, this is to be expected from Wikipedia-style text that is meant to be encyclopedic/technical in nature and devoid of passion or bias. It’s interesting, however, that we can still gauge the positivity/negativity of such documents despite the absence of strong human expression.

Regarding the effectiveness of the LDA topic model on this material, while it was generally effective when set to 10 topics, it appears that the number of topics chosen may have been too low as some of the topics can likely be expanded to denote several more subtopics. Manual investigation of the test dataset showed that some of the topics assigned were quite inaccurate, although the correlation to the percent contribution scores remained starkly in place. A more accurate and comprehensive topic model would likely have increased the usefulness of our sentiment analysis and this would be a good area to explore further during future research. For the purposes of this cursory report, however, it was most feasible to use fewer topics to accommodate our visualizations and the code runtimes.

**Citations**

[1] Blei D, Carin L, Dunson D. Probabilistic Topic Models. IEEE Signal Processing Magazine 2010. doi:10.1109/msp.2010.938079

[2] Pandey, Parul. Simplifying Sentiment Analysis using VADER in Python (on Social Media Text). Available from: <https://medium.com/analytics-vidhya/simplifying-social-media-sentiment-analysis-using-vader-in-python-f9e6ec6fc52f>

[3] Hutto, C.J. & Gilbert, E.E. VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text. Eighth International Conference on Weblogs and Social Media (ICWSM-14). Ann Arbor, MI, June (2014)

[4] Li, S. (2018, June 1). Topic Modeling and Latent Dirichlet Allocation (LDA) in Python. Retrieved from https://towardsdatascience.com/topic-modeling-and-latent-dirichlet-allocation-in-python-9bf156893c24.

[5] Violante, A. (2018, August 30). An Introduction to t-SNE with Python Example. Retrieved from <https://towardsdatascience.com/an-introduction-to-t-sne-with-python-example-5a3a293108d1>

[6] Topic modeling visualization - How to present results of LDA model?: ML . (2018, December 4). Retrieved from <https://www.machinelearningplus.com/nlp/topic-modeling-visualization-how-to-present-results-lda-models/>