# Assignment 4: Topic Modeling

## Submission Deadline: June 21, 2020 (11:59 PM)

**Deliverables:** **Project Report** in BMVC format (template can be downloaded from https://www.overleaf.com/project/5e301ad33c2f38000171a776, Ensure you insert your name and Entry Number at the top of your solution) and **submit your code in Python (Solutions in other programming languages will invoke a penalty). Make a zip file containing the code and report, and upload the zip file with your name as title on Google classroom.**

**This assignment will is designed to be harder, requiring a significant amount of time and effort! Also, significantly delayed submissions will not be considered, that is, solutions submitted more than 12 hours after the deadline will receive a ZERO score. This assignment carries 20% weightage over your cumulative FDS score (all scripts will be graded out of 100 and then scaled by a factor of 0.2), so please give this assignment your best shot!**

**Topic Modeling (100 marks, adopted from compjournalism.com):** This assignment is designed to help you develop a feel for the way *topic modelling* works, the connection to the human meanings of documents, and common ways of handling a time dimension. You will analyse the **State of the Union** speeches corpus, and report on how the subjects have shifted over time in relation to historical events.

Download the source data file *state-of-the-union.csv* from Google Classroom. This is a standard CSV file with one speech per row. There are *two columns*: the year of the speech, and the text of the speech. You will write a Python program that reads this file and turns it into TF-IDF document vectors, then prints out some information. Here is how to read a CSV in Python. You may need to add the line: *csv.field_size_limit(1000000000)* to the top of your program to be able to read this large file.

***The file is a csv with columns year, text. Note: there are some years where there was more than one speech! Design your data structures accordingly.***

1. Feed the data into the **gensim** Python package. Now you need to load the documents into Python and feed them into the **gensim package** to generate tf-idf weighted document vectors. Check out **the gensim example** code here. You will need to go through the file *twice*: once to generate the dictionary, and then again to convert each document to what gensim calls the ***bag-of-words representation***, which is un-normalized term frequency (the code snippet involving *doc2bow*).

   Note that there is implicitly another step here, which is to tokenize the document text into individual word features — not as straightforward in practice as it seems at first, but the example code just does the simplest, stupidest thing, which is to lowercase the string and split on spaces. You may want to use a better *stopword* list, such as this one. Once you have your Corpus object, tell **gensim** to generate tf-idf scores for you.

2. **Do LSI topic modeling**. You can apply Latent Semantic Indexing (LSI) to the *tf-idf* vectors. You will have to supply the number of topics to generate. Figuring out a good number is part of the assignment. Print out the resulting topics, each topic as a *list of word coefficients*. Now, sample ten topics randomly from your set for closer analysis. Try to annotate each of these ten topics with a short descriptive name or phrase that captures what it is "about." You will likely have to refer to the original documents that contain high proportions of that topic, and you will likely find that some topics have

no clear concept. **Write in your report**: your annotated topics plus a comment on how well you feel each “topic” captured a real human concept.

3. Now do **LDA topic modeling**. Repeat the exercise of step 3 but with LDA instead, again trying to annotate ten randomly sampled topics. What is different? **Write in your report**: your annotated topics plus a comment on how LDA differed from LSI.

4. **Come up with a method to figure out how topics of speeches have changed over time**: The goal is to summarize changes in the State of the Union speech in each decade of the 20th and 21st century. There are many different ways to use topic modeling to do this. Possibilities include: visualizations, grouping speeches by decade after topic modeling, and grouping speeches by decade before topic modeling. You can base your algorithm on either LSI or LDA, whichever you feel gives the most insight. Choose a method, and then use your decade summarization algorithm to understand what the content of speeches was in each decade.
   **Write in your report**: a description of your decade summarization algorithm, and an analysis of how the topics of the State of the Union have changed over the decades of the 20th century. What patterns do you see? Can you connect the terms to major historical events? (wars, the great depression, assassinations, the civil rights movement, Watergate…)

5. **Analyze a different document set**: Try LDA on a different document set, this collection of AP wire stories. Repeat the process of choosing the number of topics, fitting a model, and interpreting a random sample of 10 of them. Are the topics any clearer on this document set? If so, why? You may wish to look at previous LDA results on these documents, the top 20 words from 100 topics. **Write in your report**: your annotated topic sample, plus a description of the differences between the output on these documents vs. the State of the Union documents. Does one work better than the other? If so, define “better.”