

A1: That's What I LIKE

This assignment focuses on creating a system to find similar context in natural language processing. The system, deployed on a website, should return the top paragraphs with the most similar context to a given query, such as "Harry Potter." This task will involve building upon existing code, understanding and implementing word embedding techniques, and creating a web interface for the system to deliver the results.

Note: You are ENCOURAGED to work with your friends, but DISCOURAGED to blindly copy other's work. Both parties will be given 0.

Note: Comments should be provided sufficiently so we know you understand. Failure to do so can raise suspicion of possible copying/plagiarism.

Note: You will be graded upon (1) documentation, (2) experiment, (3) implementation.

Note: This is a one-weeks assignment, but start early.

Deliverables: The GitHub link containing the jupyter notebook, a README.md of the github, and the folder of your web application called 'app'.

Task 1. Preparation and Training - Build upon the code discussed in class. Do not use pre-built solutions from the internet.

- 1) Read and understand the **Word2Vec**¹ and **GloVe**² papers.
- 2) Modify the Word2Vec (with & without negative sampling) and GloVe from the lab lecture (3 points)
 - Train using a **real-world corpus** (suggest to **categories news** from nltk dataset). Ensure to source this dataset from reputable public databases or repositories. It is **imperative to give proper credit to the dataset source in your documentation**.
 - **Create a function** that **allows dynamic modification of the window size during training**. Use a window size of **2 as default**.

Task 2. Model Comparison and Analysis

- 1) Compare Skip-gram, Skip-gram negative sampling, GloVe models on **training loss, training time**. (1 points)
- 2) Use Word analogies dataset ³ to calculate between **syntactic and semantic accuracy**, similar to the methods in the Word2Vec and GloVe paper. (1 points)

Note : using only capital-common-countries for semantic and past-tense for syntactic.

Note : Do not be surprised if you achieve 0% accuracy in these experiments, as this may be due to the limitations of our corpus. If you are curious, you can try the same experiments with a pre-trained GloVe model from the Gensim library for a comparison.

Model	Window Size	Training Loss	Training time	Syntactic Accuracy	Semantic accuracy
Skipgram					
Skipgram (NEG)					
Glove					
Glove (Gensim)		-	-		

- 3) Use the similarity dataset⁴ to find the **correlation** between your **models' dot product and the provided similarity metrics**. (from `scipy.stats import spearmanr`) Assess if your embeddings correlate with human judgment. (1 points)

¹<https://arxiv.org/pdf/1301.3781.pdf>

²<https://aclanthology.org/D14-1162.pdf>

³<https://www.fit.vutbr.cz/~imikolov/rnnlm/word-test.v1.txt>

⁴<http://alfonseca.org/eng/research/wordsim353.html>

Model	Skipgram	NEG	GloVe	GloVe (gensim)	Y_true
MSE					

TABLE 1. Swapped Columns and Rows Table

Task 3. Search similar context - Web Application Development - Develop a simple website with an input box for search queries. (2 points)

- 1) Implement a function to compute the dot product between the input query and your corpus and retrieve the top 10 most similar context.
- 2) You may need to learn web frameworks like Flask or Django for this task.

Best of luck in developing your search engine!