Carlos

# Capstone Project Proposal

The problem I want to solve is outdated linguistics. Solving this problem is conducive to a regenerative utopia.

## Data

I have three datasets that I am employing to solve this problem. The first is a list of exuberant synonyms I created to replace outdated terms. The second is a list of neologisms I generated. The last one is a lexical, semantic, comprehensive dataset that derives from WordNet.

## Approach to Solving the Problem

This shall likely be a supervised learning problem, as I shall be training the model utilizing labeled datasets (outdated terms and their corresponding replacements). It is a classification problem, as I shall classify outdated terms into their corresponding exuberant synonyms or neologisms. I am trying to predict the most suitable replacement for outdated terms based on context and connotation. I shall utilize features extracted from the WordNet dataset, as well as contextual information from the input text and the lists of exuberant synonyms and neologisms. I plan to explore both traditional machine learning approaches, such as decision trees and logistic regression, as well as deep learning techniques, like recurrent neural networks (RNNs) or transformers, depending on the complexity of the problem and the availability of data.

## Final Deliverable

My final deliverable shall be a web application that serves as an interactive tool for revitalizing outdated terms. The application shall include an interactor-friendly interface where discerners can input text containing and the system provides suggestions for enhancement based on the exuberant synonyms and neologisms in the datasets. This approach can take sentences, words or paragraphs as an input.

## Computational Resources Needed

To effectively execute the Squishifier project, the following minimum computational resources would be required:

a. Processing Power (CPU):

- A multi-core CPU (at least 4 cores) will be essential for handling the text processing tasks efficiently, especially when transforming outdated terms into exuberant synonyms. Depending on the complexity of the model training and the size of the datasets, a more powerful CPU (6-8 cores) could be beneficial for faster processing times.

b. Memory (RAM):

- A minimum of 8 GB of RAM is recommended to ensure smooth performance while running the application and processing text. However, for larger datasets and more complex models, 16 GB or more would be ideal to avoid memory bottlenecks during data processing and model training.

c. Specialized Hardware (GPUs):

- While the project can be executed on CPU alone, utilizing a GPU would significantly enhance the performance, especially if deep learning approaches are employed. A GPU with at least 4 GB of VRAM (e.g., NVIDIA GTX 1050 or equivalent) would facilitate faster training and inference times for models that handle natural dialect processing tasks.

## Additional Considerations

- Cloud Resources: If local resources are limited, levying cloud computing services (e.g., AWS, Google Cloud, Azure) can provide scalable processing power and memory, allowing for more flexibility in facilitating resource demands as the project progresses.
- Space: Adequate space (at least 50 GB) shall be necessary to handle datasets, model files, and any generated output.