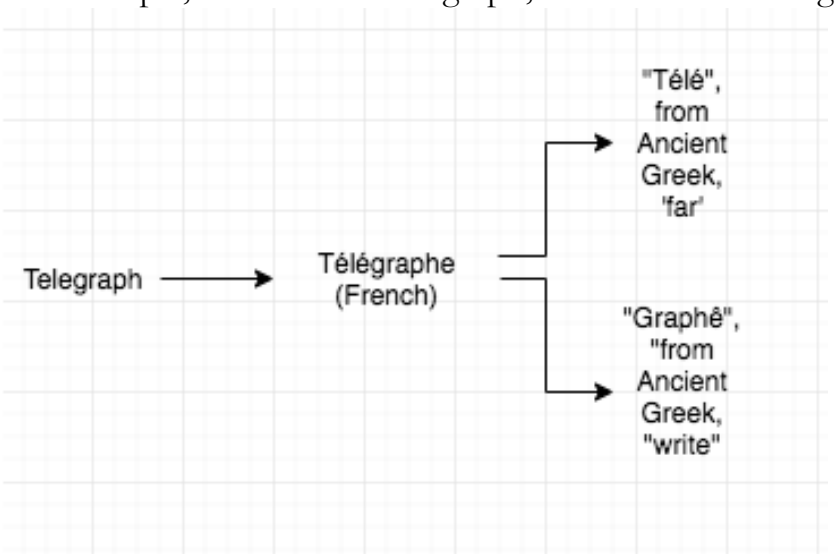
Description:
The main idea for my project is for any given word in English (maybe also in a few other languages), to output an etymological tree for that word.
For example, for the word telegraph, to obtain something like:



Additionally, for every word, I would like to give a few relevant cognates, in the input's language, but also other languages: for example, in this case: we could also output "television", "graphiste" in French and "Telekommunikation" in German.
Ultimately, once I've sorted out the word by word case, I'd like to use it to analyze full texts, and determine statistics about the main etymological origins of the text.
And finally, in a more absurd - "artistic" fashion, I was thinking of "teaching" it to create new words – a bit like how Lewis Carroll made up new words such as "vorpal" and "mimsy" whose morphology somehow fit in with their meaning. (That might be more complicated.)

1. Dataset:
   The dataset I'll be using is:
   http://www1.icsi.berkeley.edu/~demelo/etymwn/ which is a database of words and roots for several, mainly European, languages (English, but also French, Latin, Greek…) It's very extensive, almost too much, as it for instance has almost every word in English. It's a TSV (tab-separated values) file, so in principle not too difficult to parse in Python. For every word/component, it usually gives a derived form, its etymological origin in that language or another one, and another cognate or so. It's quite reliable, as it's mostly coming from Wiktionary, and concentrates all of the data I need for my project, so it's quite adapted to what I'm trying to do.

2. Methodology:
   i. Data Pre-Processing
   The data in the dataset is more or less already in the "form" I need it to be. I would like to cut it down, as it's already a bit too extensive, so it might be too heavy to handle in its full size (for instance, I have trouble simply opening the TSV file as it's so large.) Maybe, at first, I'll restrict it to simply the etymological origins for each word (by running "grep "rel:etymology" etymwn.tsv | less"), and, then, to find cognates, simply define a function to return the words with a common etymological origin and a certain degree of morphological/semantic similarity. From then on, I'd like to use "word embeddings" to represent each word by a vector, using, for example, Word2Vec.

   ii. Machine Learning model:
   To implement my project, I will use a neural network with a multi-layer perceptron which will run on the word embedding vectors.

   iii. Final Conceptualization
   For my final project, I would perhaps like to make a simple web application (where the user can input the word and obtain the etymological tree), but hopefully, depending on how the project evolves, also present a poster with my results. As I haven't found, so far, any "baseline" to my kind of output with machine learning, the "result" I'd like to obtain is to be able to, for instance, parse the English dictionary and determine percentages of most etymological origins with a certain amount of accuracy.

Sources:

- https://scikitlearn.org/stable/modules/neural_networks_supervised.html
- https://towardsdatascience.com/introduction-to-word-embedding-and-word2vec-652d0c2060fa