23 July 2020

Task1: implemented the relationship sentiment algorithm according to paper “Recognizing Sentiment of Relations between Entities in Text”.
There were multiple(3) methods described in the paper. Machine Learning approach based on CRF (accuracy 80%) could not be implemented because it required training the model with text. I believe we would prefer an unsupervised method as training the model for every single identity category would cost a lot of training data. I implemented an unsupervised method using the string of words in dependency tree region. According to the paper it had the lower accuracy of 40% but was the closest approach to what we need.

What I did was find all animated/living entities in text using parts of speech tagging and coreference resolution. All the entities were in three categories first person, second person and third person.
Marked all chains that refer to the same entity. Using a copy of the fundraising appeal text, hashed all mentions of all unique entities via keys. Used a dictionary to save the key- entity pair information.
Once we have a copy of appeal with keywords instead of each mention of the entities. We can create individual dictionaries for each entities with key as the sentence index and value as the list of all strings that reference to that entity in that sentence. We also create a dictionary with the sentence index as key and all the entities that are present in it.
When we do the dependency parsing of a sentence, we cannot use the edited text as it changes the dependency tree and hence the string of words that connect the two entities in a sentence. So, we need to do the dependency parsing of the original text itself. Now we use the dictionary which marks the entities keywords in that sentence. For every entity pair present in the sentence, we find the connecting string in the dependency parse tree using the string reference for those entities/nodes in that sentence index. The above defined dictionaries are needed to do this.

We repeat the same for all sentences in the appeal. To get the pair wise sentiment of the entity pairs we divide the total sentiment of all their connecting strings by the number of times that entity pair was found in all the sentences combined.
The library/utility that the authors used for extracting the connecting string using dependency parsing was only for Chinese language so, I had to code it myself from what was explained in the paper.
I used the Stanford NLP dependency parser. I have implemented this for one sample appeal. Please have a look at the implementation and results.

Task2: Implemented the unsupervised Text classification Algorithm from the paper “Towards Unsupervised Text Classification Leveraging Experts and Word Embeddings”. The steps were very descriptive and straightforward so their were no difficulties in the implementation. I followed all the steps exactly as mentioned in section 3 and section 4.3 of the paper.
This approach will give us a number(cosine similarity between the vector of the label keywords and input text appeal) between 0 and 1 for every input appeal and identity category keywords. 1 implies highly similar to the label category keywords in the semantic meaning of tokens constituting it and 0 implies highly dissimilar. After we develop identity category keywords for all the other categories- this method will give us a score between 0-1 implying how likely a text appeal is to be of that category.

You said we would want a count of number of tokens/words in the appeal that are similar to the identity category keywords say gender, religion, etc … all categories. Then proceed with some sort of regression. Do you think this cosine similarity score can be a proxy of such a count? There can be several complications with a simple count approach- number of words in the text appeal, number of words that are repeated, etc.

After we complete these two steps, we can have a score of how likely an appeal is to be of an identity category. And pairwise entity relationship sentiment score. It is like semantic graph with entities as nodes and their relationship sentiment as the edge.