Repo Link: https://github.com/Juan-Clay/NLP\_a1

NLP assignment 01 Report  
  
The objective of this assignment was to learn and figure out how different methods of tokenization allow for models’ accuracy on test cases to change substantially. The Jupiter notebook gives an example of a model that tokenizes characters, one that tokenizes words, and one that can take in both methods to get a hybrid result. The highest result came from tokenizing words, however with further testing and optimization, I know that optimizing the hyper features would allow for a much higher accuracy.   
  
Changing the tokenization from characters to words involves taking either words or characters as separate entities, the change allows the model to capture the meaning or context of the sentence more efficiently, and thus the vocab becoming very massive and computationally expensive. As for the hyper parameter optimization, essentially 50 randomly selected states from the possible pool of states were chosen for the created model, and by iterating and testing each version 3 times, it was possible to find out which one of the pool of 50 was the best. This strategy did find a peak however it wasn’t much higher than the average, meaning I likely did not create enough hyper parameters to test which one would yield a more significant change in the accuracy.

character level tokenization resulted in about a 64% accuracy when it comes to the test cases, however the word tokenization resulted In a much higher accuracy of 85%, The final model accuracy however was much lower, at 50% with an average loss of 69%.

The final results for the word tokenization model were very promising, however the resulting final model was not, looking forward, I will need to allocate more time in finding a more deep method to find a more accurate configuration to train my model.   
  
The biggest findings found during this project was that when it comes to more controlled text, word tokenization helps with accuracy a great deal and allows for a higher vocab count, however because of limitations of my hardware had to reduce the input size to 10,000 for this instance as google collab would crash due to not meeting the memory requirement.