T. Luke Banaszak
Tbanas2
CS410 FA22
Final – Progress Report

My final project will explore the use of text classification methods in helpdesk/IT support systems. The objectives have changed slightly from my proposal.

My proposal was to create a proof of concept that demonstrated the value of text classification within helpdesk systems. The purpose of the proposal to demonstrate how a text classifier could be used to automatically categorize IT support tickets so that they are routed to the proper technician without the submitter, or a lead, manually labeling the category. The objective was to calculate a value estimate for the application of text classifier with IT support systems - for example, based on the precision, a classifier within help desk can eliminate 10 hours of work spent reviewing and categorizing tickets.

Challenge 1 – Already been Done

I eventually found that this idea is not as novel as I'd originally thought. The exact idea was mentioned by the professor in the lecture for the week following our proposal submission date, and then I found many similar analyses and projects online as I began to work on my own.

Considering these findings, my project will instead build on some of the existing work. It will now focus on answering the question: "what specific classifier, including more complex ML methods, works best in the case of helpdesk tickets"? Time permitting, the project will also evaluate how well human-labeled support categories work by clustering tickets using an unsupervised method.

My review of the subject has found many demonstrations of a single, unigram discriminative or generative algorithm classifying tickets at an acceptable accuracy level. My project will perform this task using more complex ML classification methods. T**he results will be compared to the generative model method to determine if these methods improve accuracy and if it justifies the additional complexity and computing. Additionally, it will attempt to answer "why" helpdesk tickets work best with a certain model. For example, their narrow vocabulary, length, etc.**

Tasks Completed so far

My tasks completed so far include:

- Acquired a dataset of labeled helpdesk tickets
    - I found a dataset on Kaggle containing approximately 70,000 helpdesk tickets that are already labeled
- Chosen a baseline classification method
    - Naïve Bayes will be used as the baseline, single algorithm method for comparison to the more complex ML methods.
- Chosen the new ML methods that will compared be to a generative algorithm
    - My project will create a Multilayer Perceptron model and a CNN model for classifying

    - These methods were chosen based on Google documentation on text classifying that discussed these techniques as options depending on the sample size and average document length of the corpus
- Chosen an evaluation criteria
    - My project will use precision, recall, and F1 scores to compare the classifiers
- Chosen a technology stack and began configuring
    - This proposal includes the training of multiple types of ML models and, possibly, hyperparameter tuning. This computing workload surpasses the power of my laptop. As such, AWS Sagemaker will be used.
    - Models will be created using Scikit Learn and Tensorflow

Remaining Tasks:

- Tokenizing data
    - Lemmatizing, removing stop words, etc.
- Coding and training models
    - This will constitute the majority of my project's work
- Some hyperparameter tuning
    - Time permitting, multiple models may be trained to identify the best parameter values
- Compiling and comparing results
    - Production of visuals, charts, etc. comparing the methods
- Analyzing results
    - Summarizing key findings, identifying potential new hypotheses, etc.

Additional Challenge:

- I expect some of the models I am going to create will be large files with lengthy training. AWS is my plan to address these issues.