# Addi Wei’s Capstone 1 Project: Proposal

# Kaggle: Quora Insincere Questions Classification

Problem Statement: Insincere or toxic comments in an online community causes user frustration, wasted bandwidth and server resources to host useless conversation, and ultimately leads to increase in user churn rate. Quora is a company founded in June 2009, where questions are asked, answered, edited by its community of users. They are currently sponsoring a Kaggle competition to develop a model that identifies and flags insincere questions. With my help, Quora can develop a more scalable method to detect toxic and misleading content.

Decision Maker: Quora Manager

Other Stakeholders: N/A

Success Criteria: Since the project is a Kaggle competition, the success of the project is determined by the F1 Score of the model and directly related to the competition with other Kagglers. Since this is one of my first projects I will be utilizing more advanced machine learning models, and first project published on my GitHub, I would be satisfied with a result near the top 100 on the leaderboard.

Constraints: February 5,th 2019 is the final submission deadline.

Scope + Risks: N/A

Machine Learning Strategy (Subjective Rank color: green = strong, yellow = medium, red = weak:

1. Logistic regression: LR is till the most widely used in the industry. Since the training data is relatively large, logistic regression should perform well as a basic model.
2. Bayes Theorem / Naïve Bayes: Concern with using this simple model is due to the multi-dimensionality of data in dealing with English language, it might not produce a high accuracy. In other words, if we look at individual words, then the words by themselves may not be independently meaningful without the context of the entire sentence or question. E.g. “super” by itself vs. “super glue”, “super man”, “super fun”. NB is a bad estimator as a result.
3. KNN – since there is little understanding of the distribution of the data this algorithm could be a start but is computationally expensive and prediction stage might be too slow because the data set is so large
4. Tree Models – Gradient Boosted – generally regarded as best performing. Possible to improve the accuracy of each prediction
5. Support Vector Machine – Going to have decent accuracy, but performance will be an issue with such large data.

Initial Proposal / Desired Outcome: There are 20,000 total regularly used words in the English vocabulary and roughly 130,000 total words in the dictionary. My initial approach is to use wordnet or similar dictionary to treat each question as a vector, and every single word as a binary digit to visualize if there differences between good and bad questions. Perform a logistic regression on the two sets of data and assess the performance. Consider utilizing TF\*IDF to minimize impact of stop words which are neutral words that are not useful in determining good or bad questions. Examples of stop words are: A, I, I will, because, itself, these, them, etc. We don’t want the stop words to interefere in the scoring of good and bad questions. If logistic regression failed to produce a satisfactory model, move on to KNN model to look at similarity between good and bad questions. The assumption is there will be similarities between the good and bad questions. The score score of each question can be summed and compare the score of a good vs. bad question. The good questions should have consistently higher score and the our model will be able to predict the outcome accordingly. The third model I want to try is GAN.

Step-by-step Procedure:

1. Gather data from Kaggle website. The total data file is 6 GB.
2. Clean the data:
   1. Remove irrelevant, or non-alphanumeric characters
   2. Tokenize the text
   3. Convert all characters to lowercase
   4. Consider combining commonly misspelled words (e.g. ‘cool’/’kewl’)
   5. Consider applying stemming and lemmatization to achieve common base form of word

3) Data Exploratory Data Analysis:

1. Utilize Principle Component Analysis to visualize any difference between good and bad questions
2. Are there any unusual observations?
3. What do the features of the data look like? – shape of train and test data, how many null values,
4. Visualize

4) Classification

Logistic Regression Analysis:

1. This should be a straight forward process to analyze performance. Analyze the confusion matrix, and the impact of applying TF\*IDF or stem and lemmatization to the performance of the model.

Gradient Boosting:

1. Need to research more for examples on boosting process w/ examples.

5) Deliverables:

1. Proposal
2. Submission of results in Kaggle
3. Presentation on my project
4. Possible Youtube presentation