Answers for questions raised

1. Using Label Encoder and Rationale on encoding categorical features by encoding categories as integers rather than having one hot encoded features.
   1. One concern is that, if a one-hot encoding is used then the number of dimensions in all would be 34 + 75 + 40 + 2 + 2 + 22 + 4 = 179 features instead of 12. With the increase in the number of dimensions (and not necessarily new information), performance of machine learning models can potentially go down, (training time will certainly go up)
   2. Second, is the correlation between the one-hot variables. For example: consider the Theme Initial Segment feature which has 25 distinct values. If we create 25 distinct features, then these features will be highly correlated. Since all data points have unique values, so all but one will have zeroes.
   3. By labelling features, a potential bias is an ordering that is introduced with the different values of the features. I think this can be a potential problem because Random Forest, Gradient Boosting and Adaboost generally uses comparison operation and by introducing such labels, clearly that would get biased.
   4. A suggestion that I have received is to use the one hot encoding as well and see if there any potential benefits in doing that. I think that I must try out one hot encoding and will just take me a single day.
2. Correlation between features especially when we have an arbitrary labelling to the variables.
   1. First the Pearson correlation is defined as: where are
   2. Regarding the meaning of correlation, and a specific example in this regard would be positive high correlation between ‘semantic field’ and ‘start letter’ would mean that a high value of ‘semantic field’ corresponds to a high value of ‘start letter’. A negative high correlation would mean the inverse: high value of ‘semantic field’ corresponds to a low value of ‘start letter’.
   3. Now, from the correlation plot ‘sexgen’ is highly negatively correlated with ‘human’ and ‘animate’. This is because all inanimate objects (s\_lexiconAnimateYN == N) which are labelled as 1 don’t have the sexgen (s\_lexiconSexGender) specified which was labelled as 4. This is why they are negatively correlated. Same goes with humanness.
3. Dropping variables: dropping sexgen but not human or animate.
   1. The correlation between human and animate is 0.765 whereas correlation between sexgen and animate was -0.93; between sexgen and human it was -0.73. I considered a cutoff magnitude of 0.8, and so removed sexgen. Again this is not cast in stone, so ‘human’ or ‘animate’ can be removed as well and the result can be seen.
4. On the choice of models:
   1. The models cover a range of complexity with support vector machines being the simplest and the ensemble model being most comprehensible.
   2. These models can happen high dimensional datasets and extract complex relationships between target and features.
   3. We avoided Neural Nets or Deep Learning because the data was not enough to train neural nets.
   4. Some other machine learning models that we could have tried were: Logistic Regression, Categorical Naïve Bayes. (we worked with logistic regression but don’t report as results were no very good, (we can put this in supplementary)), I couldn’t work with Categorical Naïve Bayes due to technical issues.
5. On the choice of AUC-ROC score:
   1. AUC ROC score is said to handle class imbalance well. Though the dataset is only moderately imbalanced (75% masculine; 25% feminine), it is good to have a metric which can handle class imbalance.
   2. AUC ROC score is a summarization of confusion matrices at different cutoffs and so reports the model performance across cutoffs.
   3. We also report the Brier Loss Score which covers the limitation of AUC ROC score that it is independent of the probability magnitude values.
6. On ranking models:
   1. Here I mean that I am evaluating the models and then arranging them based on the AUC ROC score. A 5 fold CV repeated 20 times is basically the data is partitioned into 5 subsets. 4 subsets are used to train the model and then tested on the 5th set and the AUC ROC score is calculated. This is repeated 20 times and the mean AUC ROC score from the 20 runs is used to rank the models.
7. Feature Importance
   1. I have modified the code and also created new figures which are compatible. They are there in github.

Correlation Figure with new labels



