Social Computing Assignment 3

Sagarika Srishti (ssrishti3)

November 14, 2019

Part 1

In the first part of this assignment, we build 4 different Support Vector Machine models with a linear kernel, and try to predict whether a request for pizza on the Reddit subreddit Random Acts Of Pizza (RAOP) will be successful. We have to take 90% of the data as the training set, and the rest 10% of the data as the test set. To ensure that the training and test set remains constant across all the 4 models, I set the random state to the same number while dividing the dataset into training and test sets in all the 4 models. This makes sure that the random number generator gets the same seed, in turn dividing the dataset in the same way each time. I also use scikit-learn's Standard Scaler function to normalize all the features.

a) In the first model, we extract the top 500 unigrams and top 500 bigrams from the 'request_text' field of each post as features for training. This gives us a total of 1000 features. Given below are the performance metrics of this model:

Accuracy	0.73
Precision	0.41
Recall	0.28
F1 score	0.33
Specificity	0.88
AUC	0.5758477

b) In the second model, we use various fields from the data that give us information about the Activity and Reputation of the requester, such as requester account age, number of comments by the requester in RAOP,

number of subreddits, etc. While most of the features specified were numerical, two features, namely 'requester_subreddits_at_request' and 'requester_user_flair' were not. For the former, I use the number of subreddits of the user as a feature, and for the latter, I use one-hot encoding of the possible values as features. Given below are the performance metrics of this model:

Accuracy	1.00
Precision	1.00
Recall	0.99
F1 score	1.00
Specificity	1.00
AUC	0.9962686

c) In the third model, we extract features corresponding to various narrative dimensions from the 'request_text' field of each post. Given a set of words associated with each of the 5 narratives (desire, family, job, money, student), a feature value is the ratio of the number of words corresponding to each narrative dimension in the request text, and the total number of words. Given below are the performance metrics of this model:

Accuracy	0.76
Precision	0.00
Recall	0.00
F1 score	0.00
Specificity	1.00
AUC	0.5

d) In the fourth model, we extract features corresponding to various moral foundation dimensions from the 'request_text' field of each post. Given a set of words associated with each of the 11 moral foundations, a feature value is the ratio of the number of words corresponding to each moral foundation dimension in the request text, and the total number of words. Given below are the performance metrics of this model:

Accuracy	0.76
Precision	0.00
Recall	0.00
F1 score	0.00
Specificity	1.00
AUC	0.5

Part 2

a) The second model in which we take several Activity and Reputation fields as features performs the best, with an AUC of 0.99626. The first model, with top unigrams and bigrams as features performs second best with an AUC of 0.5758. Both the third and fourth model perform the worst and obtain an AUC of 0.5 (which is the minimum possible value).

b) We can see from the above tables that all 4 models achieve a very good accuracy (minimum being 0.72). But, as discussed in the last class, for skewed datasets such as the one we have, accuracy is not a good performance metric as a model which just predicts the dominant class as output will also obtain a high accuracy. This happens in models 3 and 4. In their cases, the model outputs False for all test points, which leads to an accuracy of 0.76 (it is given in the paper that the dataset has a success rate of 24.6%). This is also why these models get a precision, recall, and F1 score of 0.0. When we have skewed datasets, AUC is a good predictor of performance as it is not sensitive to imbalance, hence we compare our models based on their AUC values.

   As seen in the paper Althoff et al. (2014), a user's status within the community is strongly correlated to success (second most strongly correlated factor), hence I expected model 2 to work well. As for model 1, the authors develop baseline models based on unigrams, bigrams, and trigrams as features, and since they get a decent AUC of around 0.6 for each one of them, I expected model 1 to work somewhat similarly to this, which it did.

   I expected model 3 to perform better than it did as the paper suggested that the narratives are strongly correlated to success. I did not expect model 4 to give a good performance as I didn't find the moral foundations to be a relevant factor here.

c) Models 3 and 4 perform very poorly as compared to models 1 and 2. As both models 3 and 4 give an AUC of 0.5 and a precision/recall/f1-score of 0.0, it is hard to determine which of the two is better than the other. However, as given in the paper Althoff et al. (2014), narratives are significantly correlated with success. Hence, I predict that model 3 should perform better than model 4 after some changes are made in the features. The authors of the paper Althoff et al. (2014) find that narratives are strongly correlated to success. However, the model 3 still performs

poorly. I think that this happens because there are few words related to each narrative given. Also, the words are given in their absolute form, for example, a word in the desire narrative is 'drunk', and a match will be made only when the word 'drunk' is found in the post, not for other forms of the words such as 'drink' and 'drank'. Because of this, I think the model is missing a lot of relevant words, and hence performs poorly.

As for the fourth model, I think the category of moral foundations are not relevant to the task at hand, with the exception of care/harm Dobolyi (n.d.). The correlation of foundations like loyalty/betrayal, fairness/cheating, authority/subversion do not tend to be strongly correlated with success at getting a pizza, as a request for pizza is very unlikely to have any of these themes, while it is likely to have one of the 5 given narratives.

Another reason why I think neither of the models 3 and 4 perform well is that we're normalizing the feature value of a narrative/moral foundation by dividing it by the total number of words in a request text. Any sentence includes a lot of auxiliary verbs, articles, etc. which are not relevant to the meaning of the sentence but increase its word count. I think only nouns should be considered in order to have better feature values.

Model 1 trains on a total of 1000 features, and its performance tallies with the n-gram models developed by the authors in theAlthoff et al. (2014), hence it's performance is average. Model 2 trains on features that according to the paper are a strong predictor of success, and hence performs well.

d) I think language is able to predict the success of altruistic requests, but it is not a very strong predictor all by itself. When linguistic features are combined with other features such as social and temporal features, we're able to obtain a very strong predictor of success.

This thinking is evidenced by the performance of model 1. Model 1 uses only linguistic features, in fact it uses the most basic linguistic features, that is unigrams and bigrams, and it is still able to obtain an AUC of 0.576. I think the model would've obtained even better performance if word-embeddings were used. As given in the paper, narratives, which are also linguistic factors, are strong predictors of success. Hence, this

is another evidence that language is able to predict success of altruistic requests.

While model 2 uses features that are found to be good indicators of request success, it performs exceedingly well mainly because of the 'requester_user_flair' feature, which can be seen by calculating its correlation with the 'requester_received_pizza' field.

Part 3

a) In the paper Althoff et al. (2014), the authors extract social, linguistic and temporal features from the posts, and try to predict whether a request for pizza would be successful or not. The various textual factors include politeness, evidentiality, reciprocity, sentiment and length. The social factors include a user's status and the similarity between user profiles of the requester and the giver.

Our models differ from the models given in Table4 of Althoff et al. (2014) in that none of the models that we've developed consider temporal features. Another difference between the models that we've developed and the models given in Table 4 is that none of our models combine 2 or more types of features. While models 1, 3 and 4 are solely textual features based models, model 2 does not include textual features at all. None of our textual models (models 1, 3 and 4) achieve a performance close to that of the textual features model given in the table (AUC of 0.625).

Model 1 is similar to the the unigram, bigram, and trigram baselines given in Table 4. While model 1 combines both unigrams and bigrams into 1 baseline model, it's performance is also similar to those of these baseline models.

Another similarity between our models and the models given in the paper is that there is no significant difference between the performance of a model with only a few textual features (models 3 and 4 with and AUC of 0.5) and a unigram-bigram bsed model (AUC of 0.576), which has orders of magnitude more features.

b) Model 1 is similar to the unigram, bigram, and trigram baselines implemented in Althoff et al. (2014). While it's performance is similar to those

of these baseline models, it is still poorer than these models. While it is not given how many n-grams these baseline models use, one possible reason for the poorer performance of model 1 may be that we are only using the top 500 unigrams and bigrams as features, while the baseline models could be using more, and more information leads to a better performance.

Model 2 performs exceedingly better than any of the models given in Althoff et al. (2014). Model 2's performance can primarily be accredited to the one-hot encoded 'user_requester_flair' features, removing which its performance was observed to sharply drop. The significantly high correlation of this feature to the success of a request reaffirms the hypothesis of the authors that reciprocity is positively correlated to success.

Since models 3 and 4 always give False as their output, it can be said that their performance is the same as the random baseline model given in Althoff et al. (2014). I think this poor performance can be attributed to the poorly built features in model 3, and poorly selected features in model 4. I think the models would have performed better we'd combined 2 or more types of features together to take advantage of the maximum information that can be extracted from a post and its meta-data.

References

Althoff, T., Danescu-Niculescu-Mizil, C., & Jurafsky, D. (2014). How to ask for a favor: A case study on the success of altruistic requests. *CoRR*, *abs/1405.3282*. Retrieved from `http://arxiv.org/abs/1405.3282`

Dobolyi, D. (n.d.). *Moralfoundations.org.* `moralfoundations.org`.