MiniProject 1: Machine Learning 101

COMP 551

Submitted to

Professor William L. Hamilton

Submitted by

Marcos Cardenas-Zelaya

Viet Nguyen

Linck Wei

January 31, 2019

Abstract

In this paper we make use of a linear regression model using both closed form and gradient descent methods to estimate the popularity score of a Reddit post. We trained our model on shallow bag of common words (n-BOW), and other features. We start the analysis with varying lengths of bag of words as well as including interaction terms and transforming the feature space. Additionally, we explored various common text-processing practices such as bigrams, trigrams, character n-grams, and stopword removal. We compare these model's performance on the validation sets based on their mean squared error and adjusted R^2 coefficients. We remark that our model is not overfitting nor underfitting, because we had a lower MSE for our validation set than training set, while having a reasonable number of parameters. We find that using 67-BOW plus additional features and interactions helped us achieve the lowest MSE of 0.960 and highest adjusted R^2 of 0.264 under the closed form solution on the validation set; and had an MSE of 1.298, adjusted R^2 of 0.186 when run on the test set. Finally, we compare the performance of the gradient descent and closed form solutions and find that the gradient descent is less stable and slower on average than the closed form solution.

Introduction

The question of inferring a potential human reaction to a natural language text has been a subject of research and debate in the field of natural language processing/understanding (NLP/NLU) and in particular, sentiment analysis. Our project about estimating the popularity score of Reddit posts, given features such as the occurrence of frequent words, number of children posts, controversiality score, and a root indicator, falls into this category. Although more advanced NLP deep parsing techniques that infers the grammatical structure of sentences, or word preprocessing such as skip-gram and continuous bag of words are now widely used for similar tasks, we stick with shallow parsing using traditional n-bag-of-words (n-BOW) approach as we hypothesize that the relative frequency at which words appear in comments play a role in determining the likelihood at which a sample population would appreciate or not a comment. The closed-form and gradient descent methods of linear regression are used to optimize weights and form a hypothesis. Moreover, a timer is implemented to estimate the efficiency of the regression methods. We found that the closed-form regression has the lowest MSE and produced results in the shortest time. We compared our results with the findings of Segal et al. [1] which used linear regression, but differs in our features from ours, to predict reddit popularity score; their model achieved a MSE of 134.05, which is much higher than our training set MSE of 1.297 and validation MSE of 0.960.

Dataset

The data set used in this paper is drawn from Reddit's r/AskReddit forum. Each data points target variable is a numeric popularity score. It has features children, a count of how many responses a comment receives, a binary controversiality score, a binary root variable indicating whether it is the root of a string of comments or not, and a string of texts that is the comment of the post. The dataset contained 12,000 samples of text comments and the above-mentioned features, in which we split into 10,000 samples for training, and 1,000 samples each for validation and testing. We extracted the text features by transforming all words to lower cases,

removed apostrophes and punctuation marks, splitting each comment into its individual words and removing non-words; adding each word to a dictionary with its frequency count and sorted the comments by frequency from most to least frequent as shown in figure 1. Depending on the model employed, we used the top 60 or 160 most frequent words. We also tested various models with and without stopwords removal to qualitatively assess the contribution of stopwords to our model's performance.

Index	**Word**
0	the
1	i
2	a
3	to
…..	

Figure 1. The frequency of the most common words sorted from most to least frequent.

In addition to the core features of is_root, controversiality, children and text frequency. We included (1) max-normalized comment length, that is how many words are in each comment normalized to the longest comment length in the dataset, (2) an interaction term between is_root and children, and (3) an interaction term between comment length, is_root and children. Our reasoning for choosing these features are as follows. (1) Given the nature of reddit as being a leisurely, non-serious, forum we argue that readers prefer short comments, as such we propose that popularity should be a function of comment length that penalizes lengthy comments that will often go unread by many readers. (2) Given that popularity of a comment is a direct result of how many readers actually read and score your comment, then popularity is a function of "visibility"; in reddit a comment is more likely to be seen if it is a root comment as it is very hard to see responses to a comment that are deep in a "nested" conversation and thus have a low popularity regardless of its actual contents merit. Further, we argue that the number of children a comment has could entice a reader to score that comment more favourably as a reader could associate number of children with a comments merit. (3) This feature is a mix of the three previous; we argue that a popular comment will be "short and sweet", be a root to have high visibility, and have a lot of children so that it is posted near the top of the comments thread to add to its visibility and "perceived" merit.

Despite the average prediction power of our models, there are certain ethical dilemmas that might arise if these models are extremely accurate. Given the uncensored nature of reddit, it is easy to imagine that a proposed model that predicts popularity could feed into a potentially offensive/harmful rhetoric.

Results

Models	Gradient Descent MSE		Closed Form Solution MSE		Adjusted R^2 on validation set
	Training Set	Validation Set	Training Set	Validation Set	
No text features	1.123	1.040	1.084	1.120	0.219

Top 60	1.062	0.992	1.060	0.983	0.247
Top 160	1.082	1.008	1.047	0.992	0.240
Character bigrams	1.152	1.099	/*	/*	0.158
Word bigrams	1.146	1.087	/*	/*	0.166
Contro:root, root:child, length, len:root:child**	1.079	0.973	1.056	0.960	0.264

Figure 2

*Note: the closed form solution cannot always be implemented due to the presence of singular matrices in the inversion step in the algorithm. We suspect that different methods of parsing text (n-grams, character n-grams,…) might make the occurrence of the n-grams too rare, resulting in a very sparse singular matrix.

**Note: This is our best model given the train/validation MSEs, and we report our MSE on the test result using this model.

Model	Gradient Descent			Closed Form Solution		
	Runtime	Stability	Performance	Runtime	Stability	Performance
4 simple features	23.1s	Sometimes overshoots	Gets close to closed form	0.168s	Good if not singular	Very fast
Top 160 + 7 features	25.2s	Sometimes overshoots	Gets close to closed form	0.266s	Good if not singular	Fast

Figure 3

In this particular problem, using the closed form approach yielded results much quicker because it analytically solves exactly for weights that minimizes the quadratic loss, and the number of features were relatively small. It yielded results in less than a second while gradient descent can sometimes take up to 20 seconds as shown in figure 3. However, this analytical approach worsens as the number of features goes up as its worst-case time complexity is $O(m^3)$ where m is the number of features. Therefore, in larger models, gradient descent is much more preferred. The downside, however, is that the latter requires hyperparameter tuning.

As depicted in fig.2., we observe that the train-MSE reduces as the number of text features go up, however plateaus at around 60-70 most common words. By trial-error, we hypothesize that at 67 text features, the model returned lowest validation-MSE scores, and based on other models (n-grams) on this number. We tested our model with word and character bigrams, as we suspect lots of words offer a stronger contextual interpretation when considering their neighboring words. Surprisingly, word bigrams and character bigrams did not demonstrate significant improvement over the validation set. Comment length contributed greatly to reducing the validation MSE, and we suspect this is the case because smaller reddit comments seem to usually contain a lot of interjections and smaller remarks, or replies to bigger threads, and therefore often lack a strong meaning that would make them popular. We found that a triple

interaction term reduced our MSE from an average of 1.000 to 0.960, meaning our reasoning in (3) is likely to be correct.

Discussion and Conclusion

We find that using the analytical approach in general provides a more stable and efficient approach to finding the weights of our model; however, we note that gradient descent is better equipped for more complex models in terms of runtime, however very sensitive to hyperparameters settings. Our findings suggest that (**) (see fig. 2) is best equipped to predict post popularity scores when extended to the test set with MSE of 1.297. We note that the model is not overfitting, due to a very poor adjusted R^2 coefficient and relatively simple features with little to no non-linearity. Although this model is the best amongst its competitors that we analyzed, it is still a relatively poor model at predicting popularity score as it could at best only predict the average popularity score (only around $1/4^{th}$ of samples in the validation set is explained by our regression model, as given by an R^2 of ~0.25; most of our data is better explained with the horizontal line predicting the mean popularity score every time). This model has a high bias and low variance.

The literature of how reddit scores posts/comments suggests that the timing [2], and not necessarily the content, is a strong predictor of popularity. This suggests that our models suffer from an omitted variable bias that prevents our models from being more accurate. Future research into predicting popularity with linear regression models should include the time that the post was made as this is a strong predictor of how popular a comment/post will be. Further, the lack of context of each post in the data will again lead to an omitted variable bias. A comment about sports in a forum about sports could be very popular, however this same comment posted in a forum about cooking is unlikely to be popular. Future research will benefit from including some sort of context such as the name of the specific forum or a dictionary of the most common words found in that forum.

Certain improvements and changes can be made to the preprocessing of the data. Firstly, as mentioned before, a BOW shallow approach does not model the context in which the comments were posted, because it naïvely counts word frequencies without consideration to contextual elements of the comments. Therefore, as popularity is (hypothetically) a consequence of emotions that people experience as they read and comprehend the textual content, relationships between words as well as grammatical and syntactical meanings need to be learned as well if one wishes to improve the accuracy of predictions. One could look into term frequency – inverse document frequency (tf-idf) weighing mechanism to weigh each word according to their contextual contribution to the comment and implement with BOW to produce better results.

Statement of contributions

Code was individually done by all three members but finally joined together. Each member proposed at least 1 model. Marcos wrote the abstract and Dataset, Linck wrote the introduction and conclusion, Viet wrote the Dataset and results. Discussion and conclusion was done together.

Citations

1. Segall,Jordan. Zamoshchi, Alex. Predicting Reddit Post Popularity
2. Rohlin, Tracy (2016). Popularity Prediction of Reddit Texts. *Master's Theses. 4704*