Final Project Milestone
CS 5785 - Applied Machine Learning
James Chu (qjc2)

Motivation

Reddit is an online community where users are able to submit content under user-created boards called subreddits. As Reddit grows its user base, the number of subreddits will increase, and each subreddit will become increasingly narrow in its scope of discussions (e.g. posts for "Hyundai cars" will spin up a new subreddit dedicated to "Hyundai", rather than being aggregated under a generic "Car" subreddit). Therefore, Reddit users will find it increasingly difficult to locate the appropriate subreddit to post their content. This project will aim to develop a classifier that identifies the subreddit aligned to the content of a user-generated post. As a byproduct, we will also be able to identify the topics of each subreddit by revealing the most commonly used words in that subreddit's posts.

Method

For this project, we will be using this labelled dataset from Kaggle with 1.03M rows containing the title, content, and subreddit class of each post. As the dataset is extremely large, we will only use a small subset of the data for training and testing. The bag of words model will be used to vectorize the title and content of each post, and standard pre-processing steps will be applied. We will then leverage several supervised learning algorithms such as Naive Bayes, Logistic Regression, and Support Vector Machines on the frequencies of specific words to perform the classification. The goal is to determine the classifier that results in the highest F1-score.

Preliminary Experiments

For our preliminary experiments, we chose a subset of the dataset consisting of 25,000 posts that correspond to 25 subreddit classes. We also chose to only use the content of the post and ignored the title for now. This data was split 80/20 for training and testing purposes. We performed the standard pre-processing steps, which includes:

a) Converting all words to lowercase, removing punctuation and special characters, removing any stop words (e.g 'a', 'the', 'we', etc), and removing all html tags. These factors should not have any impact on model predictions

b) Removing website urls and Reddit usernames as we are not able to sufficiently infer any useful information from them

c) Lemmatizing all the words to its root word, as a way to reduce the amount of noise within the dataset while keeping the intent of each word

We then pre-processed the data into binary and non-binary feature vectors, and applied the following supervised algorithms:

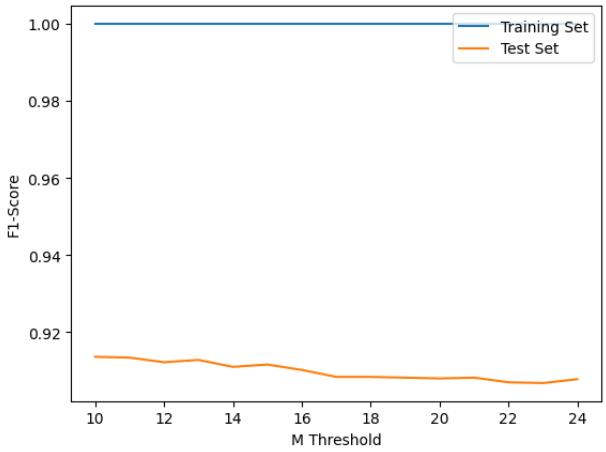
a) Logistic Regression - without regularization, with L1 regularization, with L2 regularization

b) (1,2)-grams Logistic Regression - without regularization, with L1 regularization, with L2 regularization

c) Bernoulli and Multinomial Naive Bayes, for binary and non-binary features, resp.

d) (1,2)-grams Bernoulli and Multinomial Naive Bayes, for binary and non-binary features, resp.

Note, for each of the classifiers, we chose a threshold (M) for the frequency of each word based on its impact on the Test F1-Score. As shown in the illustrative example below, increasing M results in a marginal decrease in the performance. Therefore, we chose a value of M that reduced our vocabulary to approximately 5-6k words due to hardware constraints that prevented us from processing extremely large feature spaces.



Our preliminary results are tabulated below:

Features	Classifier	M	Training F-1	Test F-1
Binary	Logistic Regression (no reg.)	20	1.0	0.9204
Binary	Logistic Regression (L1 reg.)	20	0.9936	0.9208
Binary	Logistic Regression (L2 reg.)	20	0.9998	0.9272
Binary	Bernoulli Naive Bayes	20	0.9156	0.8782
Binary	(1,2)-grams Logistic Regression (no reg.)	40	1.0	0.9174
Binary	(1,2)-grams Logistic Regression (L1 reg.)	40	0.995	0.9124
Binary	(1,2)-grams Logistic Regression (L2 reg.)	40	1.0	0.9186
Binary	(1,2)-grams Bernoulli Naive Bayes	40	0.9028	0.8468
Non-binary	Logistic Regression (no reg.)	20	1.0	0.908
Non-binary	Logistic Regression (L1 reg.)	20	0.9959	0.9166
Non-binary	Logistic Regression (L2 reg.)	20	0.9998	0.9226
Non-binary	Multinomial Naive Bayes	20	0.9520	0.9228
Non-binary	(1,2)-grams Logistic Regression (no reg.)	40	1.0	0.9078
Non-binary	(1,2)-grams Logistic Regression (L1 reg.)	40	0.9967	0.9104
Non-binary	(1,2)-grams Logistic Regression (L2 reg.)	40	1.0	0.915
Non-binary	(1,2)-grams Multinomial Naive Bayes	40	0.9458	0.9122

Observations

From our preliminary results, we observed the following:

a) Classifiers with binary features generally outperformed those using non-binary features. This indicates that the presence of a specific word is sufficient to classify the subreddit of a post

b) Adding regularization terms increased performance for Logistic Regression models and reduced overfitting effects

c) The Multinomial Naive Bayes method performed the best within the non-binary classifiers, but did not perform as well as the Logistic Regression classifier with L2 regularization with binary features

d) Using (1,2)-grams did not improve classifier performance. This indicates that singular words were more impactful in identifying a post's subreddit

Future Work

Since we observed that increasing the M-threshold for word frequency will decrease the test F-1 score by a negligible amount, we will consider pushing the limits of the M-threshold while increasing the number of subreddit classes. The goal is to classify as many subreddits as possible, while preventing the feature space from blowing up. Furthermore, as we pursue this goal, we will likely discontinue experimenting on any Naive Bayes algorithms. With additional classes and an expanded feature space, the independence assumption may fall apart. Regularization also played a role improving test F1-scores, so we will also be experimenting with varying degrees of regularization strengths.

In addition, we will experiment with other supervised learning algorithms, most notably Support Vector Machines. We also have not used the title of the posts in our experiments, so we will aim to create classifiers that incorporate this information.