Agricultural Interview Speaker Classification

Benjamin Caruso
University of Vermont Student
11 Loomis St
Burlington, VT 05401
bcaruso@uvm.edu

ABSTRACT

The topics of natural language processing and climate change rarely appear in the same sentence. Here, text classification techniques demonstrate the ability to identify differences in how climate change is discussed. Various models that have shown quality results in other text application problems were chosen, namely Logistic Regression, Random Forest, and Support Vector Machines. The focus of this paper is to correctly classify the speaker of a text sentence relating to climate change/agriculture by analyzing the words used. Interviews with both farmers and policymakers across Vermont and Maine are analyzed, with farmer and expert (policymaker) as the two classes of interest.

Keywords

Natural Language Processing, Logistic Regression, Support Vector Machine, Random Forest, SpaCy, Python-docx, Sci-kit Learn

1. INTRODUCTION

In this experiment, I take a unique approach to text classification by applying natural language processing techniques to conversation in interview format. Text classification has applications in topic modeling, sentiment analysis, and tagging content. However, a less utilized form of text classification is the semantics of conversational topics, which this experiment aims to probe deeper into. I propose the transformation of unstructured interviews with farmers and policymakers into a supervised learning problem by isolating particular passages of interest, marking them by who said them, and training a model on distinguishing between the two speakers.

The passages of text for this study focus around the very well-researched and relevant topic of climate change. The world is shifting to combat this global threat and so obtaining the opinions of people such as farmers and agricultural experts, who are close to the effects of the problem, could prove valuable to better understanding the problem's effect. Through combining knowledge on text processing with conversation on climate change, I plan to isolate the differences in how these experts and farmers discuss and what they believe on climate change.

2. RELATED WORK

A similar project was performed on a dataset of Amazon Text Reviews [2]. The authors compare various models such as Logistic Regression, SVM, and Naïve Bayes as well as unigrams, bigrams, and trigrams for correctly classifying the rating a review gave. My methods took inspiration from this project, as I compare similar models for a similar purpose, but instead of using review data I used interviews and instead of multi-class classification, I perform binary classification.

3. METHODS

First, the interview data is collected and modified to blocks of text. Then, the text is cleaned using common natural language processing techniques, Finally, a bag-of-words model transforms the text into model-ready features that are fed into various models. I used a logistic regression model, random forest, and SVM. The null model that predicts only a single class with an accuracy of 50% will be the baseline that other models are measured against.

3.1 Interview Data Collection

I obtained interviews used for this project as Word documents (.doc) from the research group that I am working with. The interviews came as new-line separated blocks of text that were direct translations of what the speaker said during the interview. Each block of text had a tab-separated label for "Interviewer" or "Interviewee" depending on who spoke, with other labels for anybody else who was present and happened to talk (e.g. "Man", "Woman", "Interviewer 2").

First, I convert the .doc file interviews into .docx files, so that the python package chosen (Python-docx [5]) is able to parse the file using a read-in method. The method returns a block of text segmented by paragraphs and inclusive of every text within the document. I wrote a function that uses the paragraphs and labels within the subsequent block of text to extract two (2) blocks of text, one with the interviewer statements and one with the interviewee (farmer or expert) statements, both in lowercase string form. The text blocks are aggregated according to whether the speaker is a farmer or expert, and whether the speaker is from Vermont and Maine. Thus, four (4) are strings are converted into .txt files in order to cleanly move to the preprocessing step.

3.2 Text Exploration and Preprocessing

The popular open-source python library spaCy [4] effectively transforms a string or series of string documents into a "processed" string that marks each word with its part-of-speech, token, word vector, and many other features. The transformed string may also be split into paragraphs/sentences, which any of spaCy's [4] pipeline objects can be applied to as well. In this experiment, the pre-trained model transformed the entire corpus of text spoken by the farmers and experts from both Vermont and Maine. The corpus breaks down into sentences by using spaCy's [4] sentence functionality on the newly transformed block of text.

Next, I define a list of match terms on which to subset my data into passages that related to climate change and the environment. The match terms chosen are as follows: 'climate change', 'climate', 'agriculture', 'weather', 'hot', 'cold', 'environment'. I consulted with the researcher to come up with this list of terms that we felt accurately depicted conversational bits relating to the topic of choice.

The following steps serve to adequately clean the text data in order to effectively perform feature extraction. The aim is to eliminate redundant features as well as reduce noise.

- Lemmatization of each sentence converts each word within a sentence to its base form; that is "running", "run", "runs", "ran" all become "run". Lemmatization utilizes a pre-defined dictionary of English words and has widely shown to improve model performance in text classification.
- Removal of punctuation as special characters are unhelpful and only act to distract the model.
- Removal of short sentences fewer than 4 words as to only include sentences that contain meaning.
- Removal of pronouns as they do not provide meaning to the sentence.

After the sentences are processed, all sentences containing at least one match term are captured in a dataset and classified with a 0/1 depending on the speaker (1 for farmer, 0 for expert). Here, I

convert the unstructured text into a supervised learning problem, making the modeling section of my experiment much simpler. Since there exists a higher frequency of farmer interviews, the dataset is slightly skewed to the farmer, and so sampling the farmer sentences resulted in a match term dataset of equal distribution between the two classes, so that accuracy could be chosen as the evaluation metric. This sample is referred to as the "match data".

In an effort to ensure no bias in how each speaker talks in general influences the efficacy of my study, I create a dataset that contains all sentences (post preprocessing) spoken, in order to see if the differences in general language are enough to correctly classify the talker. I take a sample of equal size and distribution of the match terms dataset so that the comparison stands on even ground. This sample is referred to as the "total data".

Here we explore the match terms data to see if an obvious relationship with the text can be seen. A word cloud takes a simple approach to visualization and represents the most common words in a corpus as larger in the frame.

Figure 1: No clear pattern reveals itself within the visualization, indicating that the less common words spoken by each class will likely contribute more heavily to the model's ability to classify the talker.

3.3 Feature Extraction

The bag-of-words model uses n-grams to break down a text block into n-sized chunks of words. In this experiment, unigrams (single words) were used to create a numerical feature matrix to later feed into each model. I experimented with larger n-grams, but due to poor training performance and lack of data, chose to commit to unigrams. Each text sentence that has passed previous processing steps subjects to vectorization where each sentence transforms into a feature vector that contains a 0 where a certain word is not present and the count of a word within the sentence when the word is present. Thus, the fitting process results in a sparse matrix of size m x n where m is the number of sentences in the vectorized text and n is the number of feature words.

The bag of words model utilized here holds a maximum number of 1000 features, and the same parameters are used to fit the match data as the total data. This method of feature extraction will not take into account the meaning of the sentences, but rather the composition of the words.

3.4 Speaker Classification

Here, I use various models to perform the classification task. My motivation for using these models stemmed from both my personal expertise as well as a study on interviews with cancer patients [2] as well as a comparison including these same models [1], both of which cited logistic regression, decision tree classifiers, and SVMs as worthy models for text and document classification problems. The models and reasons for use are as follows:

- Logistic Regression: Simple and allows for interpretation of the coefficient magnitude resulting from the model fit to indicate which words are more influential in making a classification.
- Support Vector Machine (Sigmoid): SVM has shown to perform well on small datasets and high dimensional data. My dataset for classification has fewer than 400 samples and the bag-of-words matrix is high-dimensional by nature, so SVM seemed a natural choice to try. Grid search analysis was used to determine the sigmoid kernel worked the best, along with a C parameter of 100 and a gamma of 0.001.
- Random Forest: The feature importance attribute demonstrates usefulness in further analyzing word influence on model decisions, which helps conclude why the classes are separable. A max depth of 10 was chosen as to account for the large number of features while also preventing overfitting by allowing the trees to grow too deep.

An 80/20 split was chosen for the training and test set, so given the dataset sizes of 384, 307 samples were used to train the models and 77 samples were left out for testing. The same train/test split was used for each model evaluation as to ensure no differences in performance due to chance.

The accuracy metric is used to compare the models. The datasets have equal distribution, meaning that the null model has a 50% accuracy and skewedness will not play a role in the results. The classification accuracy is determined by dividing the number of correctly predicted test labels by the entire size of the labels in the test set.

4. RESULTS

Ultimately, SVM with a sigmoid kernel performed the best with a 73% accuracy on the match test set. Logistic regression and random forest performed similarly with a 71% accuracy on the match test set. The total test set had similar comparisons, where SVM performed the best and logistic regression/random forest performed identically again.

4.1 Predictive Model Comparison

The following figures perform a slightly deeper dive into model evaluation, and how the three models tested compare directly to each other. Furthermore, I look at the confusion matrix for both logistic regression and random forest, which both scored the same

accuracies, to get a better understanding of how these models differed.

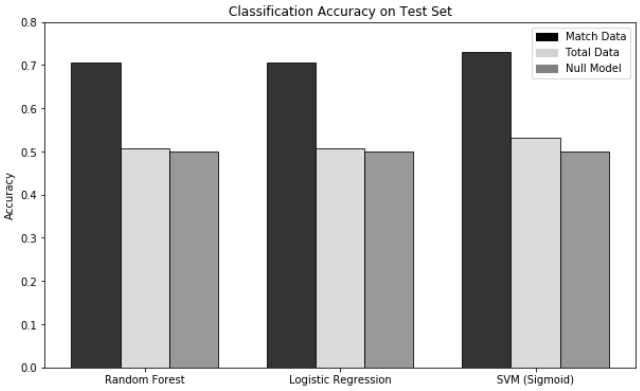
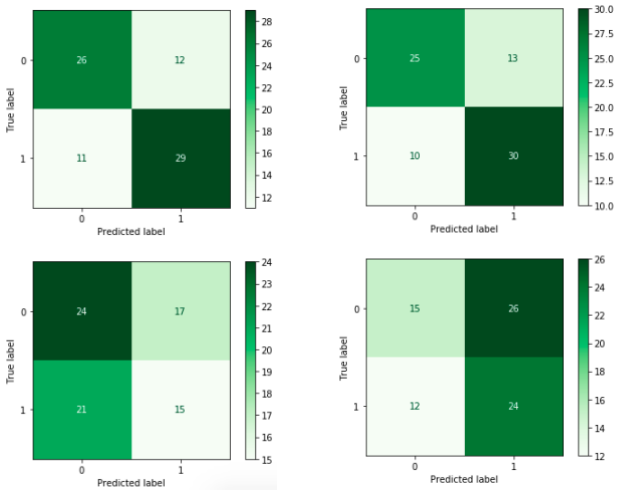


Figure 2: A side by side comparison of the three models' test accuracy. SVM performs the best, and each algorithm performs significantly better on the match terms dataset.



Figures 3: A comparison of the confusion matrices of logistic regression (left) and random forest (right), the two models with identical accuracies. The top row represents the match data, which clearly shows a greater accuracy than the bottom row, the total data. The slight differences in precision/recall are noticeable, with the random forest having a greater precision when predicting the famer but a poorer precision when predicting the expert.

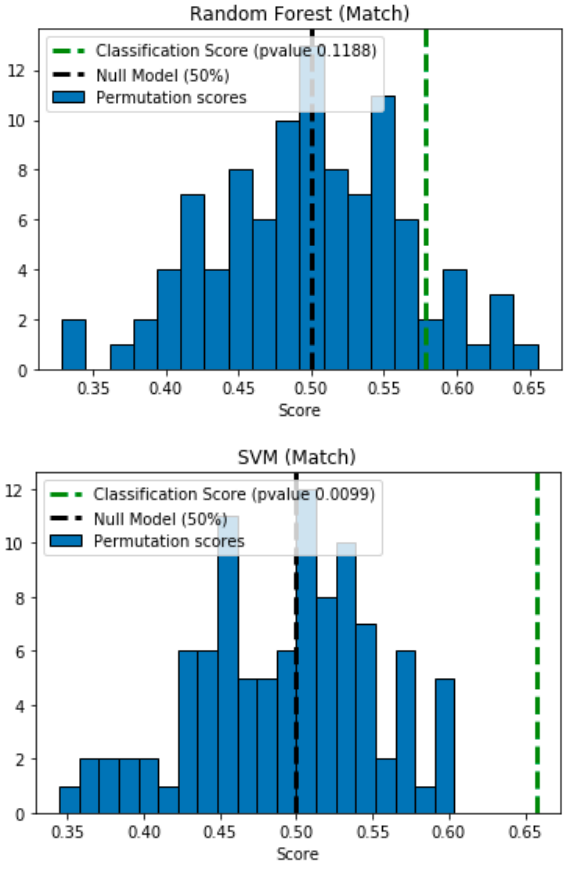
4.2 Permutation Tests

A permutation test is used to help ensure that classification results are due to significance in the model fit instead of simply statistical chance. Each of the output labels of the 6 model/data pairs are shuffled, and the model is run again with 10 folds on the new dataset. The p-value determines the percentage of runs that result in a greater accuracy than the original prediction score obtained.

The following chart demonstrates the results from a permutation test with 10-fold cross validation on the same dataset used for testing, without resampling.

Data Used	Algorithm	Permutation Test Accuracy	P-value
Match	Logistic Regression	64.3%	0.00990099
Match	SVM (Sigmoid Kernel)	65.7%	0.00990099
Match	Random Forest	57.9%	0.11881188
Total	Logistic Regression	56.6%	0.19801980
Total	SVM (Sigmoid Kernel)	54.6%	0.29702970
Total	Random Forest	51.6%	0.49504950

Below, the histogram of the permutation scores for the best performing model on the test match data (SVM) and random forest on test match data are shown, to demonstrate the contrast. SVM has shown to better fit the patterns in the test match dataset, and the permutation test results indicate that a random forest may be a weaker choice than previously thought, due to its relatively low p-



value and accuracy compared to the other two models on the test match data.

Figures 4 and 5: The random forest on match test data's permutation scores (top) are not quite as statistically significant as SVM's scores (bottom), which demonstrates a very low p-value, indicating a very low chance that SVM's accuracy on the test match data is due to pure chance.

5. DISCUSSION

These results indicate a difference in how farmers and experts talk about climate change and related topics. The SVM and logistic regression models demonstrate statistically significant (p-value < 0.05) results on a test dataset that they are able to classify an expert or farmer based purely on the text language used in a sentence with reasonable accuracy.

One limitation that must be considered upon analyzing the results is the lack of a large dataset, as the test set on match terms used was fewer than 100 samples, and thus does not fully encapsulate the wide variety of speech that a farmer or policymaker uses. Thus, further resampling or bootstrapping may improve results and give the researchers a better idea of how well the models work in general.

Certain words of influence in the logistic regression and random forest models can be picked out. Although not necessarily informative on their own, it's interesting to see which words helped the model the most, such as "impact" and "farmer" for expert, and "guess" and "hot" for farmer.

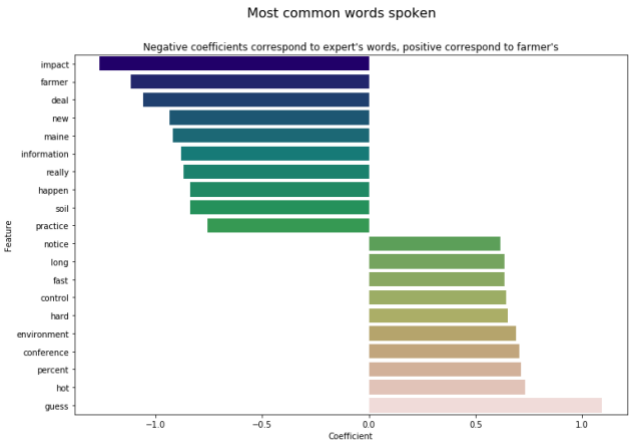


Figure 6: The most distinguishable words used by each class as interpreted through logistic regression coefficients. Further domain knowledge can be utilized to indicate whether these words have meaning or simply occur as a matter of commonality.

6. FUTURE WORK

Given more time to work on this problem, my first focus would be to utilize word embeddings, through either a neural network as a feature extractor, or a pretrained model such as Word2Vec. These embeddings have been shown to generate powerful results, and so using them may boost the model accuracy by obtaining more semantic information on each sentence spoken.

Furthermore, I may find it useful to use longer word segments when extracting the features and compare my results when using features with potentially more information within them. Unigrams, bigrams, and trigrams combined have found to increase model performance [1], and so they may work well here.

7. CONCLUSION

My experiment results indicate that farmers and experts use different language in order to discuss climate change, and multiple statistically simple models are able to distinguish between the two with a respectable accuracy on a balanced dataset. Furthermore, the models' ability to correctly classify the speaker does not rely on the bias of how the speaker talks in general, as the very same algorithms performed only marginally better than a null model on regular sentences spoken. Thus, the inclusion of match terms that directly relate to the climate and environment greatly increased the accuracy of all models involved. The accuracy increase indicates how there's a lack of any sort of relationship in the general text, which would further provide evidence that beliefs/values surrounding climate change and agriculture may be a prominent difference between farmers and policymakers.

8. ACKNOWLEDGMENTS

Thank you to Professor Nicholas Cheney, my teacher who oversaw the project and whom I regularly met with to discuss the direction I should go with my experiment. I used his lectures and advice to guide how I approached the study and obtain useful techniques on text processing.

Thank you to Meredith Niles and Carolyn Hricko who are performing the research that I am attempting to assist with using natural language processing techniques. They provided me with the interviews and were willing to meet with me to give me some helpful domain knowledge on their research topic.

9. REFERENCES

[1] Pranckevicius, T., Marcinkevicius, V. Comparison of Naïve Bayes, Random Forest, Decision Tree, Support Vector Machines, and Logistic Regression Classifiers for Text Review Classification. Vilnius University, Institute of Mathematics and Informatics, 2017

[2] Patra, Braja Gopal; Kundu, Amitava; Das, Dipankar; Bandyopadhyay, Sivaji. Classification of Interviews – A Case Study on Cancer Patients. Mumbai, India. 2012, 8 Dec. https://www.aclweb.org/anthology/W12-5304.pdf

[3] ACM SIG PROCEEDINGS template. http://www.acm.org/sigs/pubs/proceed/template.html.

[4] SpaCy documentation and open-source software. https://spacy.io/api/doc

[5] Python-docx documentation and software. https://python-docx.readthedocs.io/en/latest/

About the author:

Benjamin Caruso is a computer science senior at the University of Vermont, where he focuses on machine learning and natural language processing. He works as a teaching assistant for data structures and C++ programming courses where he grades and holds office hours for students. His projects span a variety of his interests from text sentiment analysis to statistical analysis of sleeplessness