US Airline Sentiment Analysis
Stage 2 Report

COMP5310 Principles of Data Science

Meisi Li (meli7912)
SID: 510075033

Experience Setup

The ultimate objective of this report is to build a classification model and use word embedding of the text to detect the relationship of US airline sentiment and customers' feedback. For this research, the dataset is the Tweet dataset from Kaggle, which contains 14640 rows and 15 columns. From stage one, we have two key factors of Tweet data: 'airline_sentiment' and 'text'. Both features are very important in this research. The null hypothesis is that the other models do not have any improvement. The alternative hypothesis is that other models enhance the performance.

For Stage 2, we have 3 advance questions:

- Given basic information, can we build a model to predict if the sentiment is positive, neutral or negative?
- Can we get a better prediction model?
- Is our best model different from the baseline?

Evaluation Setup

We need to do text pre-processing as data preparation. We get rid of all special characters and use the countVectorizer package to store the mean of all words as a vector.

This report is applying different classification models on the training set. The evaluation metrics we consider are f1 scores of each model and McNemar's test.

The data is split into a training set, testing set and validation set (7:1.5:1.5). The data is imbalanced and we apply the SMOTE algorithm to avoid the phenomenon that the result is obviously tilted to the denser data area. We get the accuracy and f1 score of a certain model and mainly choose f1 score to judge whether a model is better. After this, we implement McNemar's test on the baseline and the best model.

Approach

The US Airline Sentiment dataset has so many features. Among them, in our project we will be working with the 'text' and 'airline_segment' features. Here, the 'text' is considered as a feature (X) and the 'airline_segment' as a target label (y) for the classifier. The values in the target data are categorical and are 'neutral', 'positive' and 'negative'.

To preprocess 'text' simply and bring the content into a form that is predictable and analyzable for the task. We need to remove html flag digital, punctuation and replace contraction words. Then, we convert all words to lowercase, stem words and lemmatize verbs. After this, we use countVectorizer to create feature vectors. It is important to balance out the data set or at least get it close to balance it since this would give an equal priority to each class. Therefore, we use the SMOTE algorithm to deal with the imbalanced data. Then, splitting the dataset into training, testing and validation sets.

We use the Decision Tree method as a baseline model and choose some classification models such as Random Forest, Logistic Regression and one ensemble classifier combining all classifiers. To get the best parameter of the model, GridSearchCV is a method to complete.

Lastly, for the comparison of different classification models, we collect the f1 score of each model and compare all of them. McNemar’s test is used for pair- wised comparison between the best model and baseline.

Model Selection & Results

In order to find a better model that best fits our problem, we decided to begin with a simple baseline model, and train more complex models to explore any improvement of performance.

Baseline - Decision Tree

The baseline model we choose is a Decision Tree classifier since it is able to handle non-linear numeric and categorical predictors and outcomes, and allow items to be rapidly classified. We simply use the original attributes and we realised that we need to do parameter tuning for it due to high dispersions between the variables.

Parameter selection

GridSearchCV is applied on following parameter for model enhancement:

- max_depth: The maximum depth of the tree
- min_samples_split: The minimum number of samples required to split an internal node
- min_samples_leaf: The minimum number of samples required to be at a leaf node

The range of Entropy lies in between 0 to 1 and the range of Gini Impurity lies in between 0 to 0.5. Hence we use the default “Gini” criterion to select the best features. The best parameter is $\{'max\_depth': 8, 'min\_samples\_leaf': 12, 'min\_samples\_split': 2\}$.

Decision Tree report is shown in Appendix 1.

Accuracy in different set:

Training set	Validation set	Testing set
0.654	0.648	0.662

Analysis

As shown in the figure above, we could get a proper accuracy and f1 score of this mode. However, the recall and f1-score of "positive" prediction is relatively low. And its precision is high which means it returns very few results, but most of them label correct compared to the training set. The high recall and low precision of "neutral" prediction indicates most of the predicted labels are incorrect compared to the training set.

The accuracy of the training set and testing set is closed but both of them are very low. It causes under-fitting since this model is very simple.

Random Forest

The second model fitted is a Random Forest Classifier. Random Forest is a tree-based algorithm that makes the decision by evaluation multiple decision trees. We simply use the default parameters to get different parameters of the decision tree to get the result.

Parameter selection

“Gini” criterion is applied, the min_samples_split is 2 and the max_features is "auto" in our model since we want to maximize the correctness of classification. As the previous Decision Tree model, GridSearchCV is used to find the best max_depth in the Random Forest model, which is 8.

Random Forest report is in Appendix 2.

Accuracy in different set:

Training set	Validation set	Testing set
0.723	0.736	0.705

Analysis

In comparison with the Decision Tree model, the Random Forest classifier offers a better recall score in "positive" prediction, which means it returns more result and most of the results labeled correct compared to last model. However, "neutral" prediction has low recall, and computes less results compared to the training set.

The accuracy of the training set and testing set is closed which means it is a fit model.

Logistic Regression

Logistic Regression computes a good accuracy for many simple data sets and it performs well when the dataset is linearly separable. In addition, Logistic Regression is less inclined to over-fitting. The Logistic Regression also needs to have parameter tuning to get the highest accuracy.

Parameter selection

GridSearchCV is applied on following parameter for model enhancement:

- C: Inverse of regularization strength
- penalty: Used to specify the norm used in the penalization

The best parameter is $\{'C': 1, 'penalty': l2\}$.

Logistic Regression report in Appendix 3.

Accuracy in different set:

Training set	Validation set	Testing set
0.842	0.777	0.770

Analysis

Compared to above two models, the Logistic Regression classifier has high recall and high precision of all three categories, which means it returns many results, with all results labeled correctly. It also has a high accuracy to predict this dataset.

The accuracy of the training set and testing set is closed which means it is a fit model. The accuracy of this model is higher than the Random Forest classifier and this is a fitter model.

Ensemble - ‘soft’ voting mode

Finally, we use an ensemble model consisting of Decision Tree, Random Forest and Logistic Regression with different weights. The weights of them are depending on the f1 score as they hold.

Parameter selection

From the different f1 score, we could found the rank of performance is: $Logistic\ Regression > Random\ Forest > Decision\ Tree$. Therefore, the weight of Logistic Regression is the highest and the weight of Decision Tree is lowest. The weight is $\{1, 3, 6\}$ respectively to Decision Tree, Random Forest, and Logistic Regression.

Ensemble report is in Appendix 4.

Accuracy in different set:

Training set	Validation set	Testing set
0.840	0.780	0.779

Analysis

The recall and precision of all predictions are great and the result is similar to Logistic Regression, which means it returns many results, with all results labeled correctly. The accuracy of the training set and testing set is closed which means it is a fit model. The accuracy of this model could be considered the same as the Logistic Regression classifier.

The f1 score of different models

As shown on the chart in Appendix 5, we can get the information that all the models perform well except the Decision Tree model (baseline) and they can reach about 75% on f1-score. Compared to our baseline, the last two classifiers are better.

The performance is Decision Tree < Random Forest < Logistic Regression = Voting.

Hypothesis

H0: there is no difference between baseline and voting model or logistic regression model.

Hypothesis test between baseline and voting model:

```
Can we reject H0? (Voting)
Yes, we have sufficent evidence for the p-value is:  1.490958604544346e-27
```

Hypothesis test between baseline and Logistic Regression model:

```
Can we reject H0? (Logistic Regression)
Yes, we have sufficent evidence for the p-value is:  1.216606507712446e-22
```

Analysis

As the result shows, the p-value is too small. The null hypothesis is rejected and the voting classifier and Logistic Regression model have some improvement to baseline.

Conclusion & Recommendation

For the research problem, the aim is to find the relationship between airline sentiment with customers' feedback. Since the accuracy contributes more than 70%, we can find there exists a relation between them.

On the other hand, in comparing different classification methods, we found that both logistic regression classifier and voting classifier are great models for this estimation with a f1 score of 77%. Even though random forest classifier has f1 score of 70.5%, it is not a good choose compared to the other two models. I would like to recommend a logistic regression classifier and a voting classifier.

From stage 2, I get a deeper understanding of those four models. If the target is better predictions, it is better to use Random Forest to reduce the variance. And if the goal is exploratory analysis, we should prefer a single Decision Tree, since we need to understand the data relationship in a tree hierarchy structure. Logistic Regression uses a different method for estimating the parameters, which gives better results–better meaning unbiased, with lower variances. In addition, I learn an idea of how to analyze and compare the models reports.

Reference

Dataset: https://www.kaggle.com/crowdflower/twitter-airline-sentiment

Appendix

Appendix 1: Decision Tree report

Classification report of Decision Tree:

	precision	recall	f1-score	support
negative	0.72	0.76	0.74	6104
neutral	0.53	0.77	0.63	6104
positive	0.87	0.43	0.58	6104
accuracy			0.65	18312
macro avg	0.71	0.65	0.65	18312
weighted avg	0.71	0.65	0.65	18312

Appendix 2: Random Forest report

	precision	recall	f1-score	support
negative	0.76	0.90	0.82	6104
neutral	0.68	0.58	0.62	6104
positive	0.72	0.69	0.70	6104
accuracy			0.72	18312
macro avg	0.72	0.72	0.72	18312
weighted avg	0.72	0.72	0.72	18312

Appendix 3: Logistic Regression report

Classification report of logistic regression:

	precision	recall	f1-score	support
negative	0.95	0.94	0.94	6104
neutral	0.74	0.86	0.80	6104
positive	0.86	0.73	0.79	6104
accuracy			0.84	18312
macro avg	0.85	0.84	0.84	18312
weighted avg	0.85	0.84	0.84	18312

Appendix 4: Voting report

Classification report of voting classifier:

	precision	recall	f1-score	support
negative	0.94	0.94	0.94	6104
neutral	0.74	0.86	0.80	6104
positive	0.86	0.72	0.78	6104
accuracy			0.84	18312
macro avg	0.85	0.84	0.84	18312
weighted avg	0.85	0.84	0.84	18312

Appendix 5: comparing of f1-score of different models