Lessons Learned - Sentiment Analysis - Course 5, Task 3

Overview

Primary Objective:

- Evaluate the sentiments of Apple iPhone and Samsung Galaxy handsets based on common-crawl data mining.

Business Question:

- Which handset demonstrates the most positive sentiment?

Models

For the final result reported to the customer, both the iPhone and Galaxy used the sentiment-recoding method developed in feature engineering. These didn't have any attributes removed compared to the original (out-of-box) large matrix data set.

		Reference					
	C5.0	0	1	2	3	4	5
Prediction	0	351	0	3	4	10	28
	1	0	0	0	0	0	0
	2	1	0	17	0	0	0
	3	3	3	0	212	5	20
	4	4	0	0	2	117	15
	5	149	111	115	134	293	2274
						Pos/Neg Tone Accuracy	**0.889693**

Figure 1: Sentiment Tone in a C5.0 confusion matrix

Sentiment "Tone"

Besides Accuracy and Kappa, the models were evaluated on their ability to predict simply overall positive or overall negative sentiment. This was performed because, for example, it's not as impactful for an article sentiment to be wrongly predicted to be "Somewhat Positive" when it is actually "Positive", versus when it is wrongly predicted to be "Negative". This can be illustrated in the example confusion matrix seen in Figure 1. This metric is included in the model summaries in Figure 2.

Handset	Algorithm	Data Set	Training Accuracy (Median)	Training Kappa (Median)	Test Accuracy	Test Kappa	Test Sentiment Tone Accuracy
Galaxy	C5.0	OOB	0.766	0.530	0.768	0.532	0.890
Galaxy	RF	OOB	0.761	0.529	0.768	0.537	0.890
Galaxy	SVM	OOB	0.704	0.379	0.697	0.364	---
Galaxy	KKNN	OOB	0.754	0.514	0.761	0.521	0.885
Galaxy	C5.0	Corr	0.726	0.444	0.727	0.447	---
Galaxy	C5.0	NZV	0.751	0.496	0.760	0.517	0.892
Galaxy	C5.0	RFE	0.762	0.521	0.768	0.534	0.890
Galaxy	C5.0	SUB	0.645	0.194	0.644	0.186	---
Galaxy	C5.0	RC	0.842	0.585	0.847	0.606	0.892
Galaxy	C5.0	PCA	0.756	0.510	0.757	0.513	0.886

Handset	Algorithm	Data Set	Training Accuracy (Median)	Training Kappa (Median)	Test Accuracy	Test Kappa	Test Sentiment Tone Accuracy
iPhone	C5.0	OOB	0.771	0.556	0.774	0.563	0.882
iPhone	RF	OOB	0.771	0.561	0.776	0.569	0.881
iPhone	SVM	OOB	0.706	0.408	0.706	0.401	---
iPhone	KKNN	OOB	0.347	0.173	0.345	0.170	---
iPhone	C5.0	Corr	0.772	0.556	0.773	0.558	0.887
iPhone	C5.0	NZV	0.756	0.520	0.756	0.521	---
iPhone	C5.0	RFE	0.773	0.557	0.771	0.553	0.880
iPhone	C5.0	SUB	0.722	0.438	0.722	0.434	---
iPhone	C5.0	RC	0.848	0.623	0.842	0.604	0.876
iPhone	C5.0	PCA	0.758	0.532	0.767	0.550	0.878

Figure 2: Modelling results

Models and Features

Using the out-of-the-box (OOB) data, C5.0, Random Forest, SVM, and KKNN were analyzed for both the iPhone and Galaxy small-matrix training sets. RF and C5.0 were very closely matched in accuracy, kappa, and tone for both the training and test sets. However, C5.0 modeled in one tenth of the time, making it attractive for the application of additional schema.

The feature selection activities were performed to create a reduced correlation data set. For both iPhone and Galaxy, many non-DV attributes showed > 0.9 correlations, so the ones with lower correlation to the DV were removed.

Feature selection also produced near-zero-variance and recursive feature elimination data sets. A custom feature selection data set was also produced that removed all variables associated with different phones ("SUB"). For example, for the Galaxy data set, all variables were removed that didn't specifically describe the Galaxy or its Android operating system.

In feature engineering, sentiment recoding and principal component analysis (PCA) were investigated. Both the feature selection and feature engineering data sets were modeled with the C5.0 model. The model was then applied to the appropriate test data sets for complete accuracy evaluation. The complete training/test results can be seen in Figure 2.

Rationale

With the recoding of the data set, it's not surprising to see an overall increase in predictive accuracy, as the number of classification "buckets" have been decreased. However, to level set this accuracy improvement (Galaxy: 0.77 -> 0.84, iPhone: 0.77 -> 0.85), the sentiment tone also managed to be very close to the other top performing models. Also, for non-normal data sets such as this (see Figure 3) Kappa is worth tracking, as it is normalized on the baseline of random chance within our data set. The classifier recode method also improves our Kappa versus the other model / data set types. Finally, the recode methodology was chosen as it makes for simpler figures for our audience. Six levels of sentiment classification is likely not as useful, or as easy to understand as four levels.

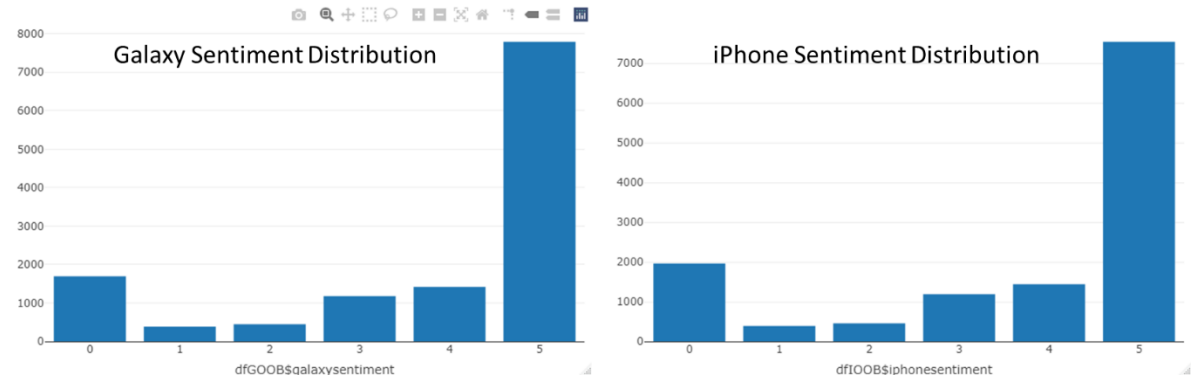

Figure 3: Sentiment histograms in training sets

Feedback

While it was time consuming to set up, model, and evaluate so many models, it was fairly straightforward (though repetitious) to implement. One challenge I had was with pipeline organization in regards to factorization. For our dependent variable, it would generally seem to feel correct to set up the factorization of our DV in the preprocessing section of our pipeline, as this is usually the correct location to adjust data types. However, I quickly learned that some of the feature selection methods (namely correlation) couldn't work with the DV being factorized. As a result, this factorization became repetitive and was done on all feature selection sets independently at the end.

Versus prior projects where I'd set up all of my feature selection and feature engineering data sets prior to any modelling, this sequential method of building/modelling was much more efficient. Previous

projects resulted in me generating full-factorial modeling matrices for every data set and potential model – by narrowing the model first, this saved a lot of processing time.

The step-wise instructions through AWS were good, but they could have used more “why” on each step. I performed some independent research on some of the many options during common crawl data mining, but high-level descriptions of what decisions we were making during setup would have been useful.