Proceedings of the Twenty-Ninth International
Florida Artificial Intelligence Research Society Conference

Comparing Approaches for Combining Data Sampling and Feature Selection to Address Key Data Quality Issues in Tweet Sentiment Analysis

Joseph D. Prusa, Taghi M. Khoshgoftaar

jprusa@fau.edu, khoshgof@fau.edu

Abstract

When training tweet sentiment classifiers, many data quality challenges must be addressed. One potential issue is class imbalance, where most instances belong to a single majority class. This may negatively impact classifier performance as classifiers trained on imbalanced data may favor classification of new, unseen instances as belonging to the majority class. This issue is accompanied by a second challenge, high-dimesionality, since very large numbers of text based features are used to describe tweet datasets. For datasets where both of these challenges are present, we can combine feature selection and data sampling to address both high-dimensionality and class imbalance. However, three potential approaches exist for combining data sampling and feature selection and it is unclear which approach is optimal. In this paper, we seek to determine if there is a best approach for combining data sampling and feature selection. We conduct tests using random undersampling with two post-sampling class ratios (50:50 and 35:65) combined with three feature rankers. Classifiers are trained with each potential combination approach using seven different learners on two datasets. We found that, overall, classifiers trained by performing feature selection followed by data sampling performed better than the other two approaches; however, the differences were only significant for the more imbalanced dataset.

Introduction

Class imbalance is a common issue in tweet sentiment datasets. For binary sentiment classifications, it is more common to encounter positive sentiment tweets than negative sentiment tweets; however, the ratio varies wildly and may also be reversed depending on the topic of the tweets (tweets about a negative event, such an an earthquake, would be expected to have a negative majority). Training a classifier using imbalanced data may result in classifier bias towards the majority class, especially if the dataset is highly imbalanced. This problem can be addressed using data sampling to create a sampled dataset with a more balanced class ratio. Data sampling, specifically Random UnderSampling (RUS), has been demonstrated to be effective at improving performance of classifiers trained on imbalanced data in related domains such as review sentiment (Li et al. 2011). RUS randomly removes majority class instances until a desired post-sampling class ratio is achieved.

Data sampling should be used in combination with with feature selection, since tweet sentiment data is highly dimensional. As tweets are text, features must be extracted from the tweets to represent the dataset. Many feature engineering methodologies result in a very large number of features being extracted, many of which may only describe a few tweets. One popular approach is to use word features (n-grams), where each word in a tweet is used as a feature. This leads to thousands or tens of thousands of features being generated to describe a tweet sentiment dataset. High dimensional datasets may lead to over-fitting, thereby reducing classifier performance on unseen (test or new) data. Feature selection techniques can be used to reduce the number of features being used to describe a dataset, improving performance and reducing the computational resources associated with training classifiers.

In this study, we investigate data sampling and feature selection techniques collectively for tweet sentiment analysis. When combining these techniques, we are primarily concerned with determining if there is a best approach for when the data sampling and feature selection steps are performed and if classifiers should be trained using the full dataset or sampled dataset. Three approaches exist depending on if feature selection is performed on the full or sampled data, and if the classifier is trained on the full or sampled data. Plainly, we seek to answer the questions: "Should feature selection or data sampling be performed first?" and "Should classifiers be trained on the sampled dataset or full dataset?".

To answer these questions, we conduct a case study using two imbalanced tweet sentiment datasets, three feature selection techniques, RUS with a 35:65 and 50:50 post-sampling class ratio and seven base learners. We train and evaluate each classifier using four runs of 5-fold cross validation and Area Under the receiver operating characteristic Curve (AUC) as the performance metric. Additionally, ANalysis Of VAriance (ANOVA) and Tukey's Honest Significant Difference (HSD) tests are conducted to verify the statistical significance of our results. In most scenarios we found no significant difference between approaches; how-

ever, we recommend feature selection followed by data sampling (training on the sampled dataset) as it was the best performing approach for highly imbalanced, high-dimensional data. To the best of our knowledge, this is the first study to combine these techniques for imbalanced, high dimensional tweet sentiment data, and the first to determine if there are any differences between these approaches in this domain.

The remainder of the paper is organized as follows. Section II discusses previous work using data sampling and feature selection in the domain of tweet sentiment classification. Section III provides an overview of our experimental methodology. Our results and statistical tests are presented in Section IV. Finally, our conclusions and suggestions for future work are presented in Section V.

Related Works

Sentiment classification of tweets has been shown to be a valuable approach for addressing real world concerns, such as prediction of election results (Wang et al. 2012), product sales (Liu et al. 2007), or movie box office performance (Meador and Gluck 2009). When training classifiers using tweet sentiment data several quality data issues may negatively impact classification performance, such as class imbalance and high-dimensionality.

Class imbalance, where the class distribution of instances is largely uneven, has been noted to have a negative impact on tweet sentiment classifier performance (Hassan, Abbasi, and Zeng 2013), (Silva, Hruschka, and Hruschka Jr 2014). A standard approach to addressing this is to use data sampling to construct new training sets with a more balanced class distribution. One of the first experiments involving data sampling and sentiment classification was conducted by Li et al. (Li et al. 2011). They chose to use RUS, as it had been found to be the best method of data sampling based on the work of Japkowicz and Stephen (Japkowicz and Stephen 2002). Their results showed that using RUS offered superior performance compared to using the full training data, and also found RUS outperformed random oversampling. While RUS randomly selects and deletes majority class instances to reach a desired class ratio, random oversampling randomly selects and duplicates minority class instances.

An additional data concern in tweet sentiment analysis is high-dimensionality, where a very large number of features are extracted from the raw data. This can negatively impact performance due to overfitting (Guyon and Elisseeff 2003). In our previous work, we identified five feature rankers that significantly improved tweet sentiment classifier performance (Prusa, Khoshgoftaar, and Dittman May 2015).

In this paper, we train classifiers using three approaches for combining feature selection and data sampling techniques in an effort to determine if there is a best approach.

Methodology

The following subsections provide details on the feature selection and data sampling techniques employed in this paper, how they were combined, the construction of our datasets, our selected learners, and how we train and evaluate our classifiers.

Feature Selection

Feature selection techniques select a subset of features, reducing data dimensionality and potentially improving classification performance. Many features may be redundant or contain no useful information. Thus, their inclusion may negatively impact classification performance due to overfitting. Feature selection techniques can be broken into several categories: filter-based, wrapper-based, embedded or hybrid approaches. In this study we have chosen to use filter-based techniques that rank features and select a subset of the top ranked features as. Additionally, filter based techniques are fast and scalable (Wang, Khoshgoftaar, and Van Hulse 2010). Since sentiment classification is often conducted on large high-dimensional datasets, filter based techniques are well suited for this domain. Compared to filter based rankers, other types of feature selection techniques such as filter-based subset evaluation, wrapper based techniques and hybrid techniques, require significantly more computational resources making them less desirable for this domain.

In this study, we perform feature selection with three rankers: Chi Squared (CS), Mutual Information (MI) and area under the Receiver Operating Characteristic Curve (ROC). CS tests the independence of features and the class label. Features strongly correlated with a particular class are ranked more highly than features with greater independence. Both MI and ROC are threshold based feature selection techniques. MI measures how much information each attribute contributes, and when used as a feature ranker, selects a subset of features consisting of the features that individually contain the most information. Area under the Receiver Operating Characteristic curve (ROC) measures the area under a curve plotting the trade-off between true positive rate and false positive rate across all possible decision thresholds. This metric is used to measure features individually, with features yielding a higher area under the ROC curve being ranked as more important. We use these feature selection techniques in combination with data sampling to train classifiers using the top 150 features. This number was selected as our preliminary experimentation found it to yield the best performing classifiers for our data.

Data Sampling

We perform data sampling using random undersampling, since it has been shown to be the best data sampling technique for sentiment classification (Li et al. 2011) and we previously found that it can significantly improve the performance of classifiers trained on imbalanced tweet sentiment data (Prusa et al. 2015). RUS randomly selects and deletes majority class instances until a desired class distribution is achieved (Van Hulse, Khoshgoftaar, and Napolitano 2007). In this paper, we evaluate RUS using two different post-sampling class distribution ratios, 50:50 (denoted as RUS50) and 35:65 (denoted as RUS35). RUS50 creates a perfectly balanced training dataset and is a commonly selected post-sampling ratio. The second post-sampling ratio, RUS35, has been shown to sometimes outperform RUS50 as it eliminates less majority class instances (Van Hulse, Khoshgoftaar, and Napolitano 2007). RUS is performed using the WEKA data-mining toolkit (Witten and Frank 2011).

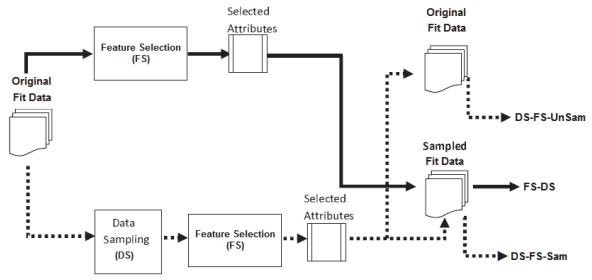

Figure 1: Approaches for Combining Feature Selection and Data Sampling

Combining Feature Selection and Data Sampling

In this study, we investigate three approaches to implementing data sampling and feature selection as there are multiple methods of combining them. First, we perform data sampling followed by feature selection and train classifiers using the full dataset. This is denoted as DS-FS-UnSam. Second, we can perform data sampling, then feature selection and train classifiers on the sampled data (denoted as DS-FS-Sam). Finally, our third approach is to perform feature selection on the full dataset, then train classifiers on the sampled dataset, denoted as FS-DS. The first two approaches are similar as they both perform data sampling prior to feature selection. The first uses the full dataset for training, while the second uses the sampled dataset. The third approach is very different as it performs feature selection prior to data sampling. This could result in some of the selected attributes not being present in the sampled training set as each tweet contains only a few of the many unigram features. The three data sampling approaches are illustrated in Figure 1. We use each of these three approaches with both RUS35, RUS50 and three feature rankers.

Learning Algorithms

In our experiments, we train and evaluate the performance of seven learners, K Nearest Neighbors (KNN), C4.5 decision tree, Support Vector Machines (SVM), Multilayer Perceptron (MLP), Logistic Regression (LR), Naïve Bayes (NB) and Radial Basis Function (RBF) Network. All learners were implemented in the WEKA toolkit (Witten and Frank 2011), and are frequently used in machine learning. Changes made to default parameters are described below. These changes were made based on preliminary research or previous work.

For K Nearest Neighbors, denoted IBk in WEKA, we chose "$k = 5$" and the "$distanceWeighting$" parameter was set to "$Weight\ by\ 1/distance$". Using these parameters, this classifier votes on the class of a new instance using the five nearest instances to the unseen instance, with the weight of each neighbor's vote being inversely proportional to its distance from the new instances. The resulting learner is labeled as 5NN in this paper.

We trained a C4.5 decision tree, J48 in WEKA, with the parameters "$nopruning$" and "$LaplaceSmoothing$" set to "$true$", as these are known to improve performance (Witten and Frank 2011).

SVM (SMO in WEKA) was trained using a linear kernel. We set the complexity constant "c" to 5.0, and the "$buildLogisticModels$" parameter set to "$true$", to obtain proper probability estimates (Witten and Frank 2011).

MLP, an artificial neural network, was trained with the "$hiddenLayers$" parameter set to "3", defining a network with 1 hidden layer containing three nodes. MLP reserves a portion of the data for verification to help teach the learner how to classify instances and when to stop training. By setting the "$validationSetSize$" parameter to "10', we set aside 10% of the training data for this process.

NB uses conditional probabilities and the naïve assumption that all attributes are independent of each other to determine the probability of an instance belonging to a class based on probabilities associated with its features. NB is an effective learner, even if its assumption of feature independence is generally unrealistic. Default parameters for NB were used in WEKA.

RBF network is an artificial neural network that outputs a linear combination of radial basis functions. The output is created using the k-means clustering algorithm and the inputs and neuron parameters. Our RBF networks were trained with the parameter "$numClusters$" set to "10."

Table 1: Datasets

Name	#minority	%minority	#attribute
5:95	150	5%	2371
20:80	600	20%	2370

Dataset

The datasets for this experiment were constructed from the sentiment140 corpus (Go, Bhayani, and Huang 2009). This dataset was selected due to being publicly available and widely studied. It includes 1.6 million tweets with positive or negative sentiment labels, allowing us to construct multiple datasets with different class distributions but the same total number of instances.

The corpus was constructed through automated collection and labeling of tweets by searching Twitter for tweets with emoticons associated with either positive or negative sentiment. Using this corpus, we constructed two datasets with different class distributions using sampling (without replacement) to select a specified number of positive and negative instances from the sentiment140 corpus, creating two datasets containing 3000 instances with the specified class ratio (5:95 or 20:80 positive:negative class ratios).

We extracted unigrams (individual words within the text of the tweet) features with the requirement that each unigram appear in at least two tweets within the training set. Unigrams were chosen, since the focus of this work is on combining data sampling and feature selection, not feature engineering. Additionally, more complex features, such as bi-grams, tri-grams and part-of-speech features, have been found to provide little additional value when conducting tweet sentiment classification, likely because of

their low frequency of occurrence, due to the short length and informal language of tweets (Go, Bhayani, and Huang 2009). Prior to feature extraction, each tweet was filtered and cleaned by removing symbols, punctuation marks and URLs, making all letters uniformly lower case, and removing excess character repetitions. Each training set contains different instances, resulting in the extraction of different unigrams for each set. The number of features extracted and class distributions of both training sets are displayed in Table 1. Both datasets have a similar number of features.

Cross-Validation and Performance Metric

Cross-validation (CV) is a technique used to train and test inductive models and uses all of the data for training, but never uses the same instances for both training and validation. We employ five-fold cross-validation, which splits the data into five equal partitions. In each iteration of CV, four folds are used as training data, while the remaining fold serves as a test dataset. This is repeated five times with a different partition used for validation in each iteration. Additionally, we perform four runs of the five-fold cross validation to reduce any bias due to how the dataset was split. This process is repeated for each dataset. When using feature selection, the feature selection process is conducted in every iteration. The classification performance of each model is evaluated using the Area Under the Receiver Operating Characteristic Curve (AUC) (Witten and Frank 2011). AUC is a valid performance metric for imbalanced data as it is a numeric representation of how a classifier performs across all decision boundaries. It is important to note that AUC is used to refer to the performance metric and should not be confused with ROC, which is used to denote the area under the receiver operating characteristic curve being used as a feature ranker.

Results

Using the methodology outlined above, we compared the performance of classifiers trained using each of the three approaches of combining feature selection and data sampling. The results of our experiment are presented in Tables 2 and 3, divided by dataset. Tables are subdivided by sampling technique and choice of ranker. Each value represents the mean AUC of the four runs of five-fold cross-validation conducted for each classifier. In each row, the model with the highest AUC is indicated in **boldface**.

Results for the 20:80 imbalanced dataset are presented in Table 2. Looking at the first post-sampling class ratio, 35:65, we observe that the approach with the highest AUC value varies with both learner and feature selection technique. For ROC, DS-FS-UnSam has the highest AUC value for four learners, DS-FS-Sam has the highest AUC for two learners and FS-DS has the highest AUC for a single learner (MLP). DS-FS-UnSam with NB had the highest AUC of any classifier trained using RUS35 and ROC. For the ranker CS, NB with FS-DS had the highest AUC. Three learners were best with FS-DS and the remaining four learners split evenly between the two remaining approaches. Results using MI favor DS-FS-UnSam, with only a single learner preferring FS-DS and no learners performed best with DS-FS-Sam. Results for all three rankers combined with RUS50 are similar as the majority of learners achieve their highest AUC values with the DS-FS-UnSam approach, with only three exceptions: DS-FS-Sam for CS with SVM and FS-DS with MLP or 5NN with MI. Overall, DS-FS-UnSam is the highest performing approach for 30 of the potential combinations of ranker, sampling ratio and classifier; however, performance differences between approaches is generally small.

Table 3 presents classification results for the three approaches on the 5:95 imbalanced dataset. Again, differences between approaches of combining data sampling and feature selection are generally small. For RUS35 and ROC, four learners have highest performance with FS-DS, two with DS-FS-Sam and one with DS-FS-UnSam. Classifiers trained using CS are evenly split between DS-FS-UnSam and DS-FS-Sam with three in each, the remaining is best with FS-DS. Classifiers trained with MI favor FS-DS with only one learner, LR achieving higher performance using DS-FS-UnSam. Using RUS50 and ROC shows that again there is no clear choice of approach to yeild higher AUC values for every learner. Three perform higher with DS-FS-UnSam, three with FS-DS and one with DS-FS-Sam. CS and RUS50 have a similar split with three favoring DS-FS-UnSam and four FS-DS. For MI and RUS50 all seven learners achieve highest AUC values using FS-DS.

Overall, little difference is observed between approaches and choice of best approach may depend on learner. For the less imbalanced 20:80 class ratio dataset, DS-FS-UnSam performed better for the majority of learners when compared to the other two approaches. This was more apparent when combining feature selection with RUS50 or when using MI as a ranker. The 5:95 dataset showed most learners had higher AUC values when using FS-DS. Again, using MI as a ranker leads to a stronger consensus on what approach yields the highest AUC for each learner. It is important to note that while some approaches appear favorable in certain situations there is relatively little difference between approaches in most cases for the 20:80 dataset. The largest observed difference is around 1%. The 5:95 dataset shows relatively greater differences between approaches, with up to 3% differences in AUC values.

Our results were tested using ANalysis Of VAariance (ANOVA) with a 5% confidence interval to determine if the differences observed when using the three approaches is significant. The results of these tests, split by dataset class ratio, are presented in Table 4. On the 20:80 imbalanced dataset choice of approach for combining data sampling and feature selection clearly has no significant impact on classifier performance as the p-value is 0.9089; however, the p-value found for the 5:95 dataset (0.0126) indicates significant differences among the three approaches. To compare each approach, a Tukey's Honest Significant Difference (HSD) test was conducted between the three approaches for the 5:95 dataset. Results are presented in Figure 1 and show the mean AUC for each approach and their accompanying confidence interval. If the intervals overlap there is no significant difference. From Figure 2 we observe that FS-DS is significantly better than DS-FS-UnSam, but there is no difference between DS-FS-UnSam and DS-FS-Sam or between

Table 2: Classification Results: Data Sampling Approaches for 20:80 Imbalanced Dataset

Sampling	Learner	Filter-Based Ranker								
		ROC			CS			MI		
		DS-FS-UnSam	DS-FS-Sam	FS-DS	DS-FS-UnSam	DS-FS-Sam	FS-DS	DS-FS-UnSam	DS-FS-Sam	FS-DS
RUS35	C4.5N	**0.670668**	0.667307	0.665539	0.650605	**0.660105**	0.658137	**0.675021**	0.670238	0.662994
	NB	**0.735994**	0.731526	0.7325	0.707502	0.704533	**0.708488**	**0.735358**	0.731902	0.72602
	MLP	0.705667	0.706326	**0.709609**	0.70442	0.701638	**0.704858**	**0.72143**	0.720841	0.719722
	5NN	0.637621	**0.641421**	0.636852	0.680294	**0.683661**	0.68057	0.673152	0.668294	**0.679268**
	SVM	**0.713245**	0.709698	0.710562	0.67662	0.680802	**0.686775**	**0.717339**	0.711454	0.69898
	RBF	0.68025	**0.683787**	0.678017	**0.681154**	0.668633	0.67156	**0.692414**	0.690173	0.681684
	LR	**0.717866**	0.703078	0.704871	**0.692372**	0.677692	0.685177	**0.727149**	0.71699	0.700121
RUS50	C4.5N	**0.667779**	0.650305	0.659895	**0.66994**	0.66328	0.666498	**0.674784**	0.661736	0.665842
	NB	**0.731506**	0.72084	0.729342	**0.713919**	0.705174	0.710898	**0.732181**	0.724528	0.726071
	MLP	**0.711539**	0.694384	0.696354	**0.708017**	0.697793	0.698796	0.711838	0.695952	**0.713533**
	5NN	**0.639083**	0.617547	0.629163	**0.683753**	0.675398	0.678966	0.669861	0.649378	**0.669916**
	SVM	**0.709819**	0.698317	0.705141	0.684412	**0.69081**	0.689751	**0.712365**	0.703013	0.706528
	RBF	**0.677636**	0.662169	0.667462	**0.664214**	0.65729	0.657496	**0.684448**	0.677764	0.677663
	LR	**0.712601**	0.685396	0.691104	**0.685168**	0.66855	0.681783	**0.721035**	0.697503	0.685944

Table 3: Classification Results: Data Sampling Approaches for 5:95 Imbalanced Dataset

Sampling	Learner	Filter-Based Ranker								
		ROC			CS			MI		
		DS-FS-UnSam	DS-FS-Sam	FS-DS	DS-FS-UnSam	DS-FS-Sam	FS-DS	DS-FS-UnSam	DS-FS-Sam	FS-DS
RUS35	C4.5N	0.597272	0.606067	**0.624858**	0.575405	**0.600971**	0.593594	0.585773	0.600513	**0.619838**
	NB	0.650539	0.647963	**0.653695**	**0.650731**	0.643073	0.630209	0.631304	0.632433	**0.659988**
	MLP	0.612099	0.617677	**0.630475**	**0.610116**	0.606588	0.509496	0.604883	0.626133	**0.630468**
	5NN	0.576086	**0.583966**	0.572205	0.592719	**0.624142**	0.602762	0.565289	0.58661	**0.628415**
	SVM	0.607382	0.611501	**0.617692**	**0.611517**	0.60431	0.60369	0.582348	0.609389	**0.62001**
	RBF	0.575303	**0.611798**	0.610923	0.595232	**0.619699**	0.584193	0.578021	0.59593	**0.611136**
	LR	**0.587389**	0.578725	0.585811	0.582029	0.577401	**0.598189**	**0.620626**	0.582184	0.593988
RUS50	C4.5N	0.594563	0.572469	**0.606557**	0.585023	0.576383	**0.596865**	0.572443	0.562998	**0.617235**
	NB	**0.638975**	0.615212	0.635763	**0.633**	0.618158	0.626423	0.621966	0.619655	**0.655776**
	MLP	**0.609458**	0.60088	0.606399	**0.614325**	0.603449	0.538923	0.590121	0.593316	**0.621591**
	5NN	0.564738	**0.569478**	0.568874	0.575181	0.587787	**0.601636**	0.573234	0.560582	**0.616806**
	SVM	0.58718	0.593526	**0.612349**	0.592157	0.601722	**0.614336**	0.562922	0.587412	**0.625693**
	RBF	0.57125	0.585512	**0.59826**	**0.592446**	0.588335	0.586573	0.5648	0.569329	**0.585209**
	LR	**0.587864**	0.56488	0.572518	0.585127	0.57881	**0.605251**	0.609004	0.536303	**0.612082**

DS-FS-Sam and FS-DS on the 5:95 imbalanced dataset.

Conclusion

In this work, we evaluate three approaches for combining data sampling and feature selection techniques when training tweet sentiment classifiers. Data Sampling can be performed first, followed by feature selection with classifiers trained using the full dataset or the sampled dataset. Alternatively feature selection can be performed prior to data sampling, with classifiers trained on the sampled data. We tested each approach with three feature rankers, two post class sampling ratios with RUS, and seven learners.

Our results show that, in general, there is little difference between the three approaches. However, results for our 5:95 imbalanced dataset show the potential for greater difference between approaches, with FS-DS being favored by most learners. We found the observed differences between approaches to not be statistically significant for the 20:80 dataset; however FS-DS was found to be significantly better than DS-FS-UnSam for the 5:95 dataset.

Based on our findings, we recommend that FS-DS is used when combining data sampling and feature selection when training tweet sentiment classifiers, especially on severely imbalanced datasets. While the difference between approaches is generally small, FS-DS is significantly better than DS-FS-UnSam in highly imbalanced senarios.

Future work should explore the impact of combining data sampling and feature selection on a wide range of tweet sentiment datasets with different sizes, class ratios, numbers of features, and additional feature rankers. Our datasets contained 3000 instances and approximately 2400 features, but it is not uncommon for tweet datasets to have tens or hundreds of thousands of instances and 10s of thousands of features. Extending this study to larger data may yield additional insights about the best approach for combining feature selection and data sampling.

Acknowledgment: We acknowledge partial support by the NSF (CNS-1427536). Opinions, findings, conclusions, or recommendations in this material are the authors and do not reflect the views of the NSF.

Table 4: ANOVA for Approaches of Combining Data Sampling and Feature Selection

Dataset	Factor	Df	Sum Sq	Mean Sq	F value	Pr(>F)
20:80	Approach	0.0003	2	0.00015	0.1	0.9089
	Residuals	0.22169	141	0.00157		
5:95	Approach	0.00802	2	0.00401	4.51	0.0126
	Residuals	0.12543	141	0.00089		

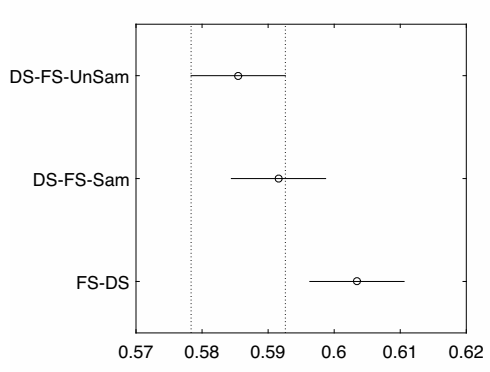

Figure 2: Tukey's HSD Test for Data Sampling Approaches on 5:95 Imbalanced Dataset

References

Go, A.; Bhayani, R.; and Huang, L. 2009. Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford* 1–12.

Guyon, I., and Elisseeff, A. 2003. An introduction to variable and feature selection. *J. Mach. Learn. Res.* 3:1157–1182.

Hassan, A.; Abbasi, A.; and Zeng, D. 2013. Twitter sentiment analysis: A bootstrap ensemble framework. In *Social Computing (SocialCom), 2013 International Conference on*, 357–364. IEEE.

Japkowicz, N., and Stephen, S. 2002. The class imbalance problem: A systematic study. *Intelligent Data Analysis* 6(5):429–449.

Li, S.; Wang, Z.; Zhou, G.; and Lee, S. Y. M. 2011. Semi-supervised learning for imbalanced sentiment classification. In *IJCAI Proceedings-International Joint Conference on Artificial Intelligence*, volume 22, 1826.

Liu, Y.; Huang, X.; An, A.; and Yu, X. 2007. Arsa: a sentiment-aware model for predicting sales performance using blogs. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, 607–614. ACM.

Meador, C., and Gluck, J. 2009. Analyzing the relationship between tweets, box-office performance and stocks. *Methods*.

Prusa, J.; Khoshgoftaar, T. M.; Dittman, D. J.; and Napolitano, A. 2015. Using random undersampling to alleviate class imbalance on tweet sentiment data. In *Information Reuse and Integration (IRI), 2015 IEEE International Conference on*, 197–202. IEEE.

Prusa, J. D.; Khoshgoftaar, T. M.; and Dittman, D. J. May 2015. Impact of feature selection techniques for tweet sentiment classification. In *Proceedings of the 28th International FLAIRS conference*, 299–304.

Silva, N. F.; Hruschka, E. R.; and Hruschka Jr, E. R. 2014. Biocom usp: Tweet sentiment analysis with adaptive boosting ensemble. *SemEval 2014* 123.

Van Hulse, J.; Khoshgoftaar, T. M.; and Napolitano, A. 2007. Experimental perspectives on learning from imbalanced data. In *Proceedings of the 24th international conference on Machine learning*, ICML '07, 935–942. New York, NY, USA: ACM.

Wang, H.; Can, D.; Kazemzadeh, A.; Bar, F.; and Narayanan, S. 2012. A system for real-time twitter sentiment analysis of 2012 us presidential election cycle. In *Proceedings of the ACL 2012 System Demonstrations*, 115–120. Association for Computational Linguistics.

Wang, H.; Khoshgoftaar, T.; and Van Hulse, J. 2010. A comparative study of threshold-based feature selection techniques. In *Granular Computing (GrC), 2010 IEEE International Conference on*, 499–504.

Witten, I. H., and Frank, E. 2011. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann, 3rd edition.