CP5 VAM SMOTE Experiments

Fred Mubang

***Abstract*—CP5 VAM SMOTE Experiments.**

I. Introduction

These are CP5 SMOTE experiments. The goal was to use SMOTE to improve the VAM-TR-72 model results from the CPEC paper. It was found that SMOTE-VAM outperforms Regular-VAM on 6 out of 10 topics.

I also plotted some instances in which SMOTE-VAM outperformed the Regular-VAM model. The SMOTE-VAM model tended to predict spikes that the Regular-VAM model missed.

Lastly, it was found that one can use time series features to "pre-select" when to use SMOTE-VAM or Regular-VAM to perform a prediction for a particular input. Ensemble models of SMOTE-VAM and Regular-VAM models can be made using this methodology. The ensemble results suggest that SMOTE-VAM performs better than Regular-VAM on input time series with low-volume, high coefficient of variation, high skewness, and high sparsity. Overall, these experiments showed that SMOTE data augmentation helps improve VAM results.

II. Methodology

A. How Samples Are Set Up

Before describing how SMOTE was used it is first useful to understand how each sample was set up. In the training, validation, and test sets, each sample represents the state of a topic q at time T, or a topic-timestep pair, (q, T). The input for a given sample is comprised of a time series matrix, made up of 7 time series (each 72 hours each), and a vector of 10 1-hot static features to represent what the topic of interest is for the given sample. Altogether, this creates an input vector with 514 features (72 * 7 + 10). Figure 1 illustrates this.

Overall there were 31,210 samples in the original training set, 1450 validation samples, and 140 test samples.

B. Converting Time Series Matrices to Singular Values

SMOTE requires classes to augment the data. However, the samples we are working with are comprised of multiple time series as outputs: three 24-hour time series for the number of new users, old users, and activities. In other words, the output is a matrix with 3 rows and 24 columns. Before applying SMOTE I needed a way to divide the samples into classes.

In order to transform the data to be suitable for SMOTE, I calculated the Frobenius Norm of each output matrix, converting each matrix into a singular value. It is calculated by taking the square root of the sum of the squares of its elements.

C. Binning

These new norm outputs were then split into classes using binning. I used binning because it was similar to the approach used in the SMOGN paper, which is one of the most recent SMOTE regression-based approaches [1].

As previously mentioned, the Frobenius Norm of each output matrix across all samples was calculated. By doing this, each sample in my dataset mapped to 1 value. The log norm of each norm was then calculated. I tried doing this experiment without log-normalization, however, by skipping this step, too many of the norms fell into 1 bin. If too many values fall into 1 bin, then it would be difficult to perform interpolation with SMOTE for the minority classes. This gave me a range of values spanning from 0 to 11.58, which you can also see in Figure 1 in the pdf.

I then performed 4 "cuts" at equal intervals along this range, giving me 4 bins. I tried other bin sizes besides 4, but when I evaluated the different SMOTE VAM models on the validation data, the models trained on the 4-bin-dataset performed the best.

Bin 1 contained all values (log-normalized Frobenius norms) spanning 0 to 2.897. Bin 2 contained all values spanning 2.897 to 5.794. Bin 3 contained all values spanning 5.794 to 8.691. Lastly, bin 4 contained all values spanning 5.794 to 11.588. The number of samples in bins 1, 2, 3, and 4 were as follows: 5,729; 17,183; 7,626; and 672, respectively. There were 31,210 training samples in total. Figure 2 contains the histogram.

Different bin divisions were used, but during validation I found that 4 worked the best. Other values tried were 2, 3, 4, 5, and 10.

D. Applying SMOTE

SMOTE was then used to augment the training set. Each of the 4 bins acted as the classes used for SMOTE. Since the 2nd bin had the most samples (17,183), that was the majority class. The other bins were considered as the minority class. So, the new dataset contained 17,183 samples per class, or 68,732 samples in total.

SMOTE uses K-Nearest Neighbors as part of the interpolation process, so I tried different values for K to see which training set would yield the best performance. These values were 3, 5, 7, 10, and 15.

III. Results

In summary, applying SMOTE to the VAM dataset did improve performance. The following subsections explain the results in more detail.

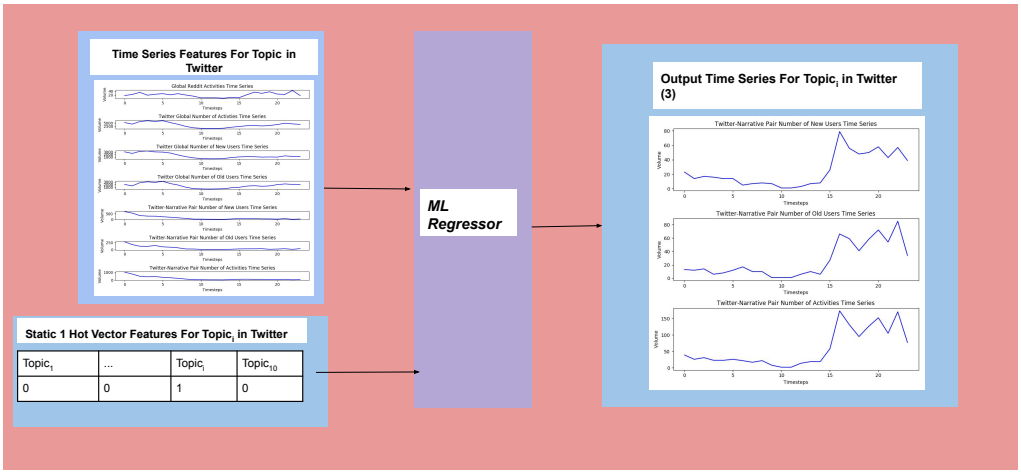

Fig. 1: How each sample is setup in the dataset.

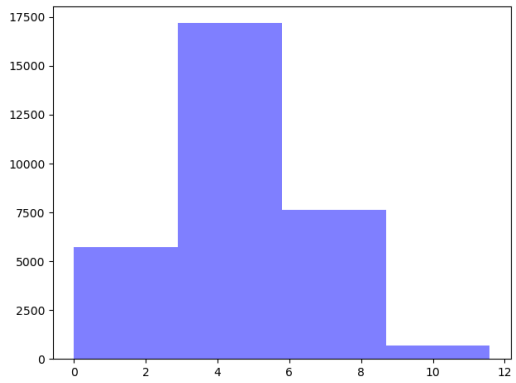

Fig. 2: The 4 bin categories for the CPEC training dataset. There are 31,210 samples. The X axis shows the range of log-normalized, Frobenius Norm values that the samples in each bin map to. The Y-axis shows the number of samples contained within a given bin.

A. Initial Overall Results

Table I contains the overall results. The regular VAM model used as a baseline was the VAM-TR-72 model from the CPEC VAM paper, which was the best model from that paper. Five other "SMOTE-VAM" models are shown in the table. Each of these models was trained on an augmented SMOTE dataset with a different K-parameter from K-Nearest Neighbors. For example, the *SMOTE-VAM-TR-72-KNN-15* model was trained on a SMOTE dataset that used K=15, that is, the 15 nearest neighbors were used for interpolation of each new SMOTE sample.

Similar to the CPEC and Venezuela VAM papers, the RMSE, MAE, S-APE, Volatility Error (VE), Skewness Error (SkE), and NC-RMSE metrics were used. Furthermore, the *Overall Normalized Metric Error* was included as well to show how well each model did across all 6 metrics in one combined error metric. This is the same metric used in the VAM Venezuela and VAM CPEC papers.

It was calculated by creating six "metric groups," each comprising 14 model metric results for that particular metric. A similar "normalized error metric" was used in [19]. The model results within each of the six groups were normalized between 0 and 1 by dividing each model metric result by the sum of all model metric results within that particular group. The models in each table are then sorted and ranked from lowest to highest ONME.

As one can see in the table, the regular VAM-TR-72 model outperformed all the SMOTE-VAM models, however the SMOTE-VAM models were close in performance. VAM-TR-72 had an ONME score of 0.164. The second best model was the SMOTE-VAM-TR-72-KNN-15 model, with an ONME of 0.166 (lower is better). The worst SMOTE-VAM model was the SMOTE-VAM-TR-72-KNN-7 model with an ONME of 0.167.

B. Per-Topic Comparisons

I also did per-topic comparisons. Table II shows these results. The SMOTE-VAM model won 6 out of 10 times. The metrics being compared are the Overall Normalized Metric Errors (ONMEs), similar to the metric used in Table I.

The Wilcoxon Signed Rank Test was used to test for significance. The p-values are shown in the table as well, with an alpha of 0.05. Five out of 6 of SMOTE-VAM's wins were statistically significant.

C. Time Series Plot Analysis

I then plotted time series for instances in which the SMOTE-VAM model outperformed the Regular-VAM model in Figure 3. As one can see, SMOTE-VAM tends to predict spikes more than the Regular-VAM model, albeit not always in the exact locations.

SMOTE VAM vs. Regular VAM Results							
Model	**RMSE**	**MAE**	**VE**	**SkE**	**S-APE**	**NC-RMSE**	**ONME**
VAM-TR-72	63.7693	45.77	35.8454	1.0726	37.9726	0.1253	***0.164981***
SMOTE-VAM-TR-72-KNN-15	65.3416	47.3483	35.9175	0.9955	38.7184	0.1319	0.166495
SMOTE-VAM-TR-72-KNN-5	65.2403	47.366	36.2461	1.0457	38.0417	0.129	0.166963
SMOTE-VAM-TR-72-KNN-10	65.4238	47.423	36.8451	0.9918	38.5976	0.1318	0.167078
SMOTE-VAM-TR-72-KNN-3	65.7256	47.7087	35.6934	1.0141	38.4214	0.1331	0.167239
SMOTE-VAM-TR-72-KNN-7	65.3826	47.4046	36.1957	1.0235	38.1862	0.1324	0.167241

TABLE I: SMOTE VAM vs. Regular VAM Results. The VAM-TR-72 model is the same one from the CPEC VAM paper. Each of the different SMOTE VAM models used a different K for the K-Nearest Neighbors part of SMOTE, as shown in each model name.

SMOTE-VAM vs. Regular-VAM Per-Topic Results						
Topic	**Regular-VAM**	**SMOTE-VAM**	**Winner**	**Percent_Imp**	**p_value**	**Is Significant**
controversies/pakistan/baloch	0.5095	**0.4905**	SMOTE-VAM	3.7172	0.0105	1
benefits/development/energy	0.5086	**0.4914**	SMOTE-VAM	3.3854	0.0007	1
controversies/pakistan/students	0.5043	**0.4957**	SMOTE-VAM	1.7183	4.0622e-07	1
controversies/china/border	0.5033	**0.4967**	SMOTE-VAM	1.3129	3.5970e-08	1
controversies/china/uighur	0.5032	**0.4968**	SMOTE-VAM	1.2616	0.0349	1
benefits/jobs	0.5013	**0.4987**	SMOTE-VAM	0.5148	0.0565	0
benefits/development/roads	**0.498**	0.502	Regular-VAM	-0.8171	0.0363	1
leadership/bajwa	**0.4949**	0.5051	Regular-VAM	-2.0539	6.6439e-09	1
leadership/sharif	**0.4927**	0.5073	Regular-VAM	-2.947	7.7207e-07	1
opposition/propaganda	**0.4797**	0.5203	Regular-VAM	-8.4716	4.7278e-12	1

TABLE II: SMOTE-VAM vs. Regular-VAM Per-Topic Results

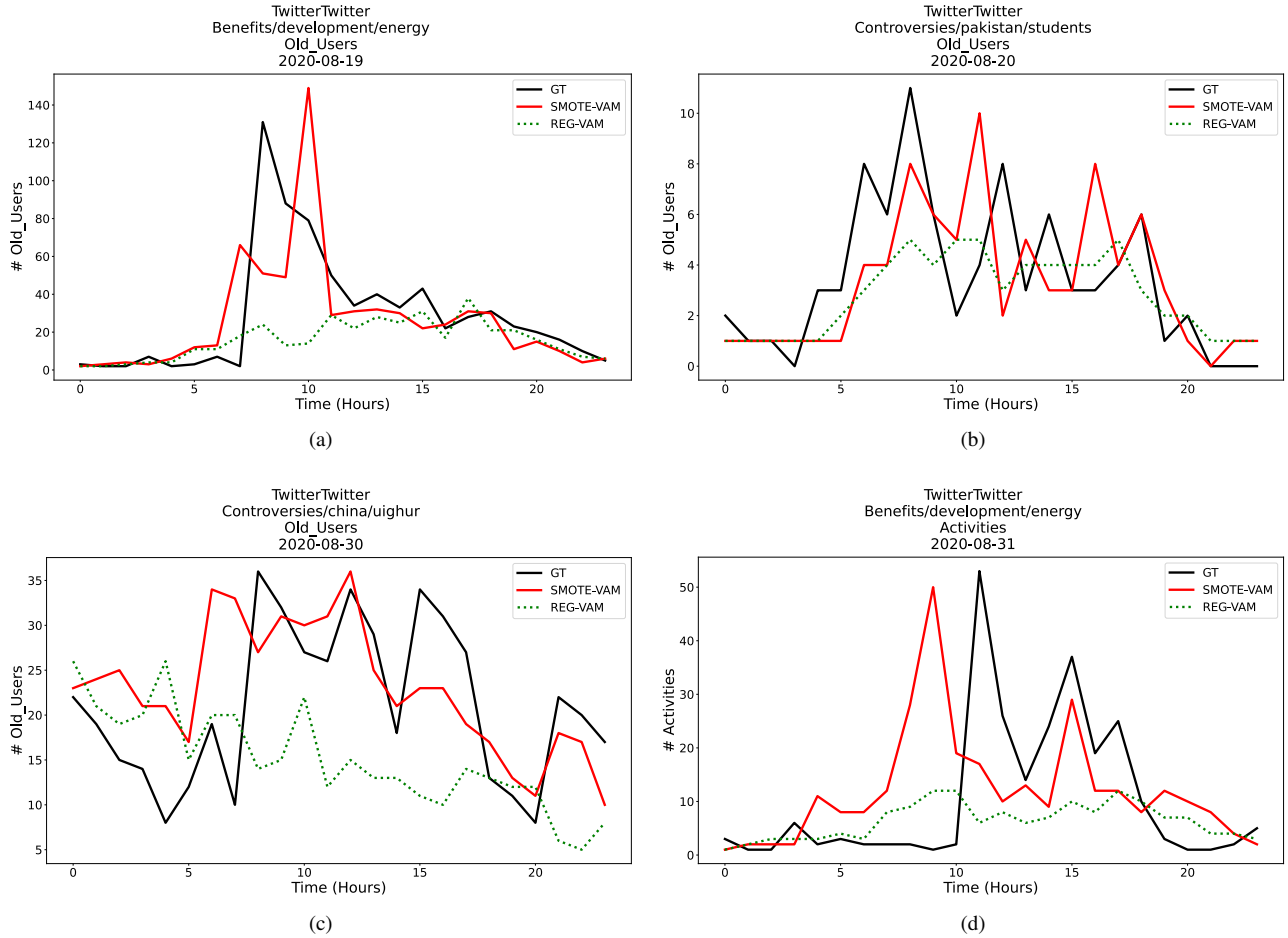

Fig. 3: Here are some instances in which the SMOTE-VAM model outperformed the Regular VAM (Regular-VAM) model.

D. SMOTE VAM Ensemble - Time Series Attributes

The next question I sought to answer was the following: "Does SMOTE-VAM work better on time series with certain attributes?" If so, that means a heuristic could potentially be created so that one can determine when to use SMOTE-VAM vs. Regular-VAM for a particular time series.

The time series of interest I would be looking at would of course be the input time series in each of the 140 test samples. Specifically, they would be the input time series related to the 3 output-types of interest: (1) new users, (2) old users, and (3) activities. What I want to know is if there are any attributes of these input time series that can indicate whether to use a SMOTE-VAM model or Regular-VAM model for predicting the output time series. Note that since there are 140 test samples, and since each test sample is related to 3 output-types, there are 420 input time series of interest, and 420 output time series of interest. Each input time series maps to an output time series, of course.

E. Cluster Ensemble Models

In order to answer the aforementioned question, 8 ensemble SMOTE-VAM models were created. They were made using 4 time series attributes: (1) volume, (2) coefficient of variation, (3) skewness, and (4) sparsity.

Volume refers to the total counts of a time series, such as total number of users or activities. Coefficient of variation is a ratio that is calculated by dividing the standard deviation by the mean. Skewness is a measure of the asymmetry of a time series. Sparsity is a measure of the number of 0's in a time series.

Furthermore, these 4 attributes were used to created 2 types of clusters per attribute: (1) high-value clusters and (2) low-value clusters. High is defined as any value that is above the 80th percentile value and low is defined as any value that is equal to or less than the 80th percentile value. There were 420 input time series of interest in the test set, so 84 (420 * 0.2) of these input time series went into the high cluster and 336 (420 * 0.8) went into the low cluster.

Each of the 8 ensembles utilized both the SMOTE-VAM model and the Regular-VAM model, albeit with different heuristics. Table III shows how each ensemble works. For example, the *SVE-low-volume* ensemble is the model that uses SMOTE-VAM for input time series that have low volume, and Regular-VAM for input time series that have high volume. The intuition behind a model like this would be that the SMOTE-VAM model may be better for input time series that have a low volume of activities or users, while the Regular-VAM model may be better for input time series with a high volume of users or activities.

Another ensemble example would be the "SVE-high-skewness" ensemble. This would the ensemble that uses SMOTE-VAM on input time series with a high skewness, and Regular-VAM on input time series with a low skewness.

Table III shows how each ensemble was set up.

F. SMOTE VAM Ensemble Results

Table IV shows the results of the SMOTE-VAM ensemble models. As usual, the Overall Normalized Metric Error (ONME) is used to compare models. The VAM-TR-72 model is the "Regular-VAM" model (no augmentation). Also, ARMA is included in this table because that was the best baseline from the CPEC VAM paper. The Percent Improvement From Baseline score (PIFB) is calculated against ARMA, in order to keep consistency with the CPEC paper results.

As one can see, the best model was the *SVE-low-volume* ensemble. As previously mentioned, this is the SMOTE-VAM ensemble that uses SMOTE-VAM on input time series with a low volume of users or activities, and Regular-VAM on input time series with a high volume of users or activities. The PIFB of this model was 18.44%, whereas Regular-VAM (VAM-TR-72) had a PIFB of 17.38%.

In summary, the models in which the SMOTE-VAM model outperformed Regular-VAM were the *SVE-low-volume*, *SVE-high-coefficient_of_variation*, *SVE-high-skewness*, and lastly, *SVE-high-sparsity* models. These results suggest that SMOTE-VAM performs better than Regular-VAM on input time series with low-volume, high coefficient of variation, high skewness, and high sparsity.

G. Significance Testing

Table V contains the results of significance testing on the results of Table IV. The metric results from the Regular-VAM model were compared against the metric results of each of the SMOTE-VAM models using the Wilcoxon Signed Rank Test. The p-values are shown in the table, as well as a column indicating whether or not the result is significant with an alpha of 0.05. As one can see, all the ensemble results were significant.

IV. Conclusion

SMOTE-augmentation was tried with the VAM-TR-72 dataset. It was found that SMOTE-VAM outperformed Regular-VAM on 6 out of 10 topics. Some time series plot analysis was done as well, and it was found that for some time series the SMOTE-VAM models tended to predict spikes where the Regular-VAM models did not. Lastly, ensemble models were created with SMOTE-VAM and Regular-VAM models using time series attributes. Some of these ensemble models outperformed the Regular-VAM model. These results suggest that SMOTE-VAM performs better than Regular-VAM on input time series with low-volume, high coefficient of variation, high skewness, and high sparsity. Overall, these experiments showed that SMOTE data augmentation helps improve VAM results.

References

[1] P. Branco, L. Torgo, and R. Ribeiro, "Smogn: a pre-processing approach for imbalanced regression," 09 2017.

SMOTE-VAM Ensemble Cluster Information		
Model	**Input Time Series That SMOTE-VAM Predicts**	**Input Time Series That Regular-VAM Predicts**
SVE-low-volume	Low Volume	High Volume
SVE-high-coefficient_of_variation	High Coefficient of Variation	Low Coefficient of Variation
SVE-high-skewness	High Skewness	Low Skewness
SVE-high-sparsity	High Sparsity	Low Sparsity
SVE-low-sparsity	Low Sparsity	High Sparsity
SVE-low-skewness	Low Skewness	High Skewness
SVE-low-coefficient_of_variation	Low Coefficient of Variation	High Coefficient of Variation
SVE-high-volume	High Volume	Low Volume

TABLE III: SMOTE-VAM Ensemble Cluster Information

SMOTE-VAM Models vs. Regular VAM Model								
model	**RMSE**	**MAE**	**VE**	**SkE**	**S-APE**	**NC-RMSE**	**ONME**	**PIFB**
SVE-low-volume	63.6242	45.6153	35.2648	0.9949	37.8055	0.128	0.0963	**18.4417**
SVE-high-coefficient_of_variation	63.8156	45.7187	35.5568	1.0387	38.1664	0.1259	0.0971	17.7947
SVE-high-skewness	63.8454	45.821	35.6585	1.0393	38.151	0.1256	0.0971	17.7501
SVE-high-sparsity	63.7906	45.7851	35.7685	1.0396	38.3977	0.1257	0.0973	17.6216
VAM-TR-72	63.7693	45.77	35.8454	1.0726	37.9726	0.1253	0.0976	17.38
SVE-low-sparsity	65.3202	47.3332	35.9943	1.0285	38.2933	0.1315	0.0988	16.2961
SVE-low-skewness	65.2654	47.2973	36.1043	1.0288	38.5401	0.1316	0.099	16.1677
SVE-low-coefficient_of_variation	65.2952	47.3996	36.2061	1.0294	38.5246	0.1313	0.099	16.1231
SVE-high-volume	65.4866	47.5031	36.498	1.0732	38.8855	0.1292	0.0998	15.476
ARMA	72.5972	55.1047	39.4702	1.6593	42.9807	0.143	0.1181	0.0

TABLE IV: SMOTE-VAM Models vs. Regular VAM Model. ARMA is included to show that the augmented models also outperform the best baseline model as well.

SMOTE-VAM Signed Wilcoxon Rank Test P-Values		
Model	**p_value**	**Is Significant**
SVE-low-volume	3.95e-09	1
SVE-high-normed_coefficient_of_variation	1.43e-09	1
SVE-high-normed_skewness	1.12e-09	1
SVE-high-sparsity	2.21e-10	1
SVE-low-sparsity	3.89e-23	1
SVE-low-normed_skewness	5.79e-22	1
SVE-low-normed_coefficient_of_variation	1.61e-22	1
SVE-high-volume	9.51e-29	1

TABLE V: SMOTE-VAM Signed Wilcoxon Rank Test P-Values. Each model was compared to the VAM-TR-72 model (Regular VAM) from the CPEC paper.