IMPROVING F-SCORE OF THE IMBALANCE VISUALIZED PATTERN DATASET FOR YIELD PREDICTION ROBUSTNESS

Megat Norulazmi Megat Mohamed Noor[1] *and Shaidah Jusoh*[2]*

[1]*Graduate Dept of Computer Science, College of Arts and Sciences, Universiti Utara Malaysia, 06010 Sintok, Kedah, Malaysia. Email:megatnorulazmi@gmail.com*
[2]*Graduate Dept of Computer Science, College of Arts and Sciences, Universiti Utara Malaysia, 06010 Sintok, Kedah, Malaysia. Email:shaidah@uum.edu.my* *

ABSTRACT

In a non closed loop manufacturing process, a prediction model of the yield outcome can be achieved by visualizing the temporal historical data pattern generated from the inspection machine, discretize to visualized data patterns, and map them into machine learning algorithm. Our previous study shows that combination of under-sampling and over-sampling techniques unable to handle wider range of data sets where SMOTE+VDM and random under-sampling produced robust classifier performance of handling better with different batches of prediction test data. In this paper, the integration of K entropy base similarity distance function with SMOTE, CNN+Tomek Links and the introduction of SMOTE and SMaRT (Synthetic Majority Replacement Technique) combination, has improved the classifiers F-Score robustness.*

Keywords: Yield prediction, Predictive maintenance, Pattern visualization, Data re-sampling, Robust classifier

1 INTRODUCTION

Megat Norulazmi Megat Mohamed Noor & Shaidah Jusoh (2008) introduced the approach to transform inspection machine generated numerical data from the nature that the data only can be used to learn the inspection machine behavior into a visualized data sets that will be able to be trained to predict “one step ahead” of the manufacturing yield outcome. However, due to the imbalances of the temporal data sets produces by manufacturing yield, robust classifier is required.

The exact balance of minority and majority classes are important but the most important matter to focus is the balance distribution of relevant information carried by each class instances. Well balanced number of instances in the data set will produce robust classifier but further improvement on the F-Score is needed.

Random under-sampling has the potential of information loss which affecting the class precision, whilst over-sampling method will improve the class recall with mild impact to the precision but carry the risk of over-fitting. The over-sampling with appropriate synthetic minority instance is important to improve the class recall with minimum impact from the over-fitting. In Megat Norulazmi Megat Mohamed Noor & Shaidah Jusoh (2008) shows that VDM distance function did not really perform with the visualized datasets.

Similarity distance functions such as K* entropy based distance function (Cleary & Trigg, 1995) should be considered to be integrated with (Chawla, Bowyer, Hall & Kegelmeyer, 2002) SMOTE (Synthetic Minority Over-sampling Technique). On the other hand, because of the under-sampling will cause the information loss and reducing the class precision, we propose to balance up the imbalance visualized data set distribution without performing under-sampling technique, but by removing the irrelevant instances with CNN (Condensed Nearest Neighbor)+Tomek Link and K* based distance function from the majority class first and then to replace them back with SMOTE and K* entropy distance function. Thus, in this paper the new combination of SMaRT (Synthetic Majority Replacement Technique) and SMOTE with integration of K* based entropy distance function is introduced.

2 RELATED WORK

Yoon & Kwek (2007) highlighted that the over-sampling and under-sampling combination does not provide significant improvement compared to the over-sampling alone. However, Megat Norulazmi Megat Mohamed Noor & Shaidah Jusoh (2008) shows that over-sampling alone is subject to over-fitting when tested with different badges of test data, whilst combination of over-sampling and under-sampling produced a robust classifier.

Kubat & Matwin (1997) emphasized that those borderline instances that are close to the boundaries between the positive and negative region are unreliable because even a small amount of them can shift decisions surface into wrong side. Those that are redundant majority instances can be taken over safely in order to reduce the computation cost for Tomek Links algorithm.

Hence, Kubat & Matwin (1997) proposed an under sampling method called one-sided selection (ONESS), which exploits the concept of Tomek links (Tomek, 1976). Kubat & Matwin (1997) suggestion is to remove a majority instances in a Tomek link that is measured to be borderline and/or noisy. Furthermore, Kubat & Matwin (1997) delete the redundant majority instances with CNN algorithm based on a 1-nearest neighbor classification as shown in Table 2.

Table 2. Algorithm for the one-sided selection of instances.

CNN algorithm Tomek Links algorithm

1. Let S be the original training set.
2. Initially, C contains all minority examples from S and one randomly selected majority example.

instances

3. Classify S with the 1-NN rule using the examples in C, and compare the assigned concept labels with the original ones. Move all misclassified examples into C that is now consistent with S while being smaller.
4. Remove from C all negative examples participating in Tomek links. This removes those negative those are believed at borderline and/or noisy. All minorities instances are retained. The resulting set is referred to as T.

Megat Norulazmi Megat Mohamed Noor & Shaidah Jusoh (2008) study showed that the visualized pattern data sets performs best with K* learning algorithm (Cleary & Trigg, 1995). K* is the instance based learning algorithm where computing of the distance between two instances is motivated by information theory. The distance between instances is defined as the complexity of transforming one instance into another instance. The computation of the transformation complexity is done in two steps. Firstly, a finite set of transformations which map instances to instance is defined. A "program" to transform one instance (*a*) to another (*b*) is a finite sequence of transformations starting at *a,* and terminating at *b*.

The K* distance function handles the problem by summing all possible transformations between two instances. K* approach does not focus on the distance between two instances that can be defined as the length of the shortest string connecting the two instances from many possible transformation as what kolmologrov complexity theory suggested. The result of the shortest string is a distance measure. It is very sensitive to small changes in the instance space and does not solve the smoothness problem well.

3 PROPOSED METHODS

We are using K* entropy similarity distance based on kolomogrov complexity theory, to replace the Euclidean or VDM distance function for SMOTE and to determine the borderline and/or noisy instances in all of our Tomek implementation. As for 1-NN classification in CNN, the K* learning algorithm is applied.

3.1 SMOTE with K* entropy

SMOTE technique (Chawla, Bowyer, Hall & Kegelmeyer, 2002) proposed to over-sample the minority class by selecting *k* minority class nearest neighbor instances and producing synthetic instances. Depending on the percentage of over-sampling required, neighbors from the *k* nearest neighbors are randomly chosen and the synthetic instances were generated by calculating the nearest neighbor numerical dataset with Euclidean distance function. Total numbers of synthetic instances were generated according to the required number of over-sampling percentage.

3.2 SMOTE and RANDOM UNDER-SAMPLING

This approach was applied in Megat Norulazmi Megat Mohamed Noor & Shaidah Jusoh (2008) where Value Difference Metric (VDM) was implemented as similarity distance function for SMOTE. Random under-sampling sampled from majority class instances until the instances numbers exactly balance up with SMOTE percentage. The results from this approach are used to compare the significant of K* entropy based distance function with VDM.

3.3 One sided selection under-sampling

As suggested by Kamei, Monden, Matsumoto, Kakimoto & Matsumoto (2007), we applied the approach to verify the importance of balance distribution between majority and minority instances in order to produce robust classifiers. In this paper, we applied ONESS approach by under-sampling the majority instances with CNN and CNN+Tomek Links. The results from those algorithms verify the relationships of redundant and borderline majority instances on our visualized data patterns on minority instances.

3.4 SMOTE and CNN+TOmek under-sampling

This approach applied K* bases entropy similarity distance function on the SMOTE and CNN+Tomek under-sampling. The distribution of the minority and majority were made exactly balanced by limit the CNN+Tomek under-sampling process until it reach the SMOTE instances percentage. The lowest percentage of SMOTE allow CNN+Tomek under-sampling to process further deeper compare to the higher percentage of SMOTE. Since the Tomek process will push majority instances further lower than the number of available minority instances, this approach is actually equivalent to the performing under-sampling with CNN algorithm alone without Tomek links.

3.5 SMOTE AND SMART (CNN+TOMEK)

Our proposed SMaRT technique applied the CNN+Tomek algorithm until it reaches to the end of the process. The number of majority instances left after the process is smaller compared with minority class instances. Thus, SMaRT used SMOTE algorithm and K* entropy similarity distance function to generate the synthetic majority instances until it balanced up with numbers of minority instances total up with instances generated by percentage of SMOTE.

3.6 SMOTE AND SMART (CNN)

This approach is similar with aforementioned above, except that we only implemented SMaRT with CNN alone without the Tomek Links algorithm. The result between these 2 approaches indicates the relationships of redundant and borderline/noisy instances with our visualized data sets whether they carries significant differences.

Once the CNN process ended, the majority instances populated are bigger from minority instances. When SMOTE at smaller percentage, CNN generates the majority instances slightly bigger and SMaRT will not generates synthetic majority instances. At the higher SMOTE percentage, SMaRT instances will get into the distribution.

3.7 F-Sore performance measure

We are using the same F-score performance measured in Megat Norulazmi Megat Mohamed Noor, & Shaidah Jusoh (2008). We used F-Score to measure the overall performance of the sampled datasets studied. F-measure is a harmonic mean between recall and precision defined as:-

$$F = 2 \times \left(\frac{R \times P}{R + P}\right) \qquad (1)$$

The F-measure becomes zero if either R or P is zero and it will become 1 when both R and P are 1. R is recall and P is precision. Recall and precision are define as:-

$$R = \frac{\mathrm{CP}}{\mathrm{TP}} \quad (2)$$

$$P = \frac{\mathrm{CP}}{\mathrm{PP}} \quad (3)$$

CP is the number of instances that are correctly predicted as positive and TP is the number of actual positive instances, where PP is total number instances predicted as positive.

4 EXPERIMENT AND RESULTS

4.1 Procedure

We were using the same data from () previous study where data fields used for the study were *ID*, *Total Yield Percentage, RankA, G-NG, R-NG, Ring, Hit, MP1, MP2, MP3, Q3MP3* and *Yield* class instance. 12 bit visualized training data was used as the original dataset to generate the new training data for the combination sampling of SMOTE+Random_Under-sampling, SMOTE+SMaRT (CNN+Tomek), SMOTE+SMaRT(CNN), SMOTE+CNN Under-sampling data sets. The datasets were created by over-sampling the original dataset minority class by 50%, 100%, 150%, 200%, 250% and 300% and then under-sampling and *SMaRTing* the original majority class instances until it is reached to the same number of over-sampled instances, so that their distribution will be exactly balanced up. We also generate ONESS data sets by under-sampled the majority instances with CNN (Condense Nearest Neighbor)+Tomek Links and CNN (Kamei, Monden, Matsumoto, Kakimoto & Matsumoto ,2007) to verify the importance of the exact balance distribution between majority and minority instances for our visualized data sets.

Training datasets were been trained with K* algorithm as suggested by Megat Norulazmi Megat Mohamed Noor & Shaidah Jusoh (2008) for the learning process with confusion matrix and stratified 10-fold cross validation. Classifiers generated from the training data were then being used to be tested with test data (Megat Norulazmi Megat Mohamed Noor & Shaidah Jusoh, 2008) from the same batch with training data sets. The classifiers once again were being tested with two test data sets from different batches to test the robustness of the classifiers. The result of the training and prediction test of this paper is compared with the results in (Megat Norulazmi Megat Mohamed Noor & Shaidah Jusoh, 2008) to verify the effectiveness of the integration of K* entropy base similarity distance function and improvement of the F-Score measure on the robustness classifiers generated.

4.2 Result Analysis

Comparing with (Megat Norulazmi Megat Mohamed Noor & Shaidah Jusoh, 2008), Table 4.1, 4.2, 4.3 and 4.4; the results of SMOTE+Random Under-sampling shows that K* based entropy similarity distance function performs better than the integration of SMOTE with VDM distance function. Results in Table 4.1 also indicate that SMOTE+SMaRT training result significantly performs better that other double sided sampling techniques and outperform the ONESS Over-sampling method resulted in (Megat Norulazmi Megat Mohamed Noor & Shaidah Jusoh, 2008). While significantly improving the training result, SMOTE+SMaRT remain its robustness performs better from (Megat Norulazmi Megat Mohamed Noor, & Shaidah Jusoh, 2008) previous study, a result which is in contrast behavior compared to over-sampling technique. Comparing SMOTE+SMaRT with CNN and CNN+Tomek, Table 4.2 and 4.4 indicates that SMOTE+SMaRT(CNN) outperforms SMOTE+SMaRT(CNN+Tomek), while Table 4.1 and 4.3 shows the reverse result. Table 4.1, 4.3 and 4.4

Table 4.1. Classifiers training result of bad yield class instances

		Performance		
		R	P	F
Tomek+CNN Under-sampling		0.99	0.79	**0.879**
CNN Under-sampling		0.30	0.58	0.399
SMOTE+Random Under-sampling				
	50%	0.83	0.77	0.798
	100%	0.91	0.79	0.847
	150%	0.95	0.80	0.869
	200%	0.97	0.80	0.880
	250%	0.98	0.81	0.891
	300%	0.99	0.82	**0.897**
SMOTE+CNN Under-sampling				
	50%	0.78	0.67	0.720
	100%	0.88	0.64	0.745
	150%	0.97	0.61	0.751
	200%	0.98	0.64	0.771
	250%	0.99	0.65	0.788
	300%	0.99	0.68	**0.807**
SMOTE+SMaRT(TOMEK+CNN)				
	50%	0.84	1.00	0.914
	100%	0.91	1.00	0.955
	150%	0.96	1.00	0.979
	200%	0.98	1.00	0.990
	250%	0.99	1.00	0.993
	300%	1.00	1.00	**0.998**
SMOTE+SMaRT(CNN)				
	50%	0.74	0.66	0.698
	100%	0.88	0.66	0.755
	150%	0.95	0.69	0.798
	200%	0.98	0.76	0.855
	250%	0.99	0.83	0.901
	300%	0.99	0.87	**0.924**

shows that SMOTE+SMaRT(CNN) perform better with the increase of SMOTE percentage which also means that SMaRT instances also increased.

The implementation of CNN alone compared with CNN+Tomek did not indicates significant different accept the result shows in Table 4.2 where the implementation of CNN algorithm shows significantly better especially with ONESS approach. Table 4.2 results also indicates that the F-Score decrease when the SMOTE percentage increase. This also mean that, at lower percentage the CNN process at its fullest level where SMaRT not generates and majority instances slightly outnumbers minority instances. However, the trend was not significant from Table 4.1, 4.3 and 4.4. For the ONESS approach, even though CNN under- sampling shows significant result shows in Table 4.2, but it did not produce robust classifier as indicates in Table 4.1, 4.3 and 4.4. CNN+Tomek under-sampling produce quite a robust classifier accepts for the result indicates from Table 4.2, where the result is at the lowest compare with another approaches. We can see that CNN+Tomek under-sampling producing best performance for class recall in overall result but lower class precision, while the other approaches sacrificing class recall to raise up the class precision a bit in order to improve the F-Score. In ONESS, also we can see that removing borderline and noisy instances is crucial to produce robust classifier.

Table 4.2. Classifiers same batch data test result of bad yield class instances

	Performance		
	R	P	F
Tomek+CNN Under-sampling	1.00	0.13	0.230
CNN Under-sampling	0.92	0.82	**0.868**
SMOTE+Random Under-sampling			
50%	0.96	0.38	0.539
100%	0.96	0.42	0.585
150%	0.96	0.47	**0.632**
200%	1.00	0.39	0.562
250%	1.00	0.43	0.602
300%	0.92	0.42	0.575
SMOTE+CNN Under-sampling			
50%	0.96	0.57	**0.716**
100%	1.00	0.52	0.685
150%	0.96	0.25	0.400
200%	0.96	0.22	0.356
250%	1.00	0.20	0.333
300%	1.00	0.19	0.325
SMOTE+SMaRT(TOMEK+CNN)			
50%	0.96	0.32	0.485
100%	1.00	0.32	0.481
150%	0.96	0.31	0.471
200%	0.96	0.30	0.462
250%	1.00	0.32	**0.481**
300%	0.96	0.29	0.444
SMOTE+SMaRT(CNN)			
50%	0.88	0.79	**0.830**
100%	0.96	0.34	0.505
150%	1.00	0.46	0.633
200%	0.96	0.43	0.593
250%	1.00	0.34	0.510
300%	1.00	0.35	0.521

Table 4.3. Classifiers different badge data test result of bad yield class instances

	Performance		
	R	P	F
Tomek+CNN	1.00	0.26	**0.415**
Under-sampling			
CNN Under-sampling	0.07	0.22	0.108
SMOTE+Random Under-sampling			
50%	0.61	0.29	0.391
100%	0.64	0.30	0.405
150%	0.43	0.24	0.304
200%	0.57	0.30	0.395
250%	0.39	0.20	0.265
300%	0.64	0.33	**0.434**
SMOTE+CNN Under-sampling			
50%	0.71	0.33	**0.449**
100%	0.54	0.27	0.361
150%	0.75	0.30	0.424
200%	0.79	0.30	0.436
250%	0.86	0.29	0.432
300%	0.82	0.29	0.426
SMOTE+SMaRT(TOMEK+CNN)			
50%	0.71	0.32	0.440
100%	0.68	0.29	0.409
150%	0.71	0.32	0.440
200%	0.75	0.35	**0.477**
250%	0.71	0.29	0.412
300%	0.68	0.32	0.432
SMOTE+SMaRT(CNN)			
50%	0.29	0.31	0.296
100%	0.54	0.32	0.400
150%	0.61	0.29	0.391
200%	0.54	0.26	0.353
250%	0.61	0.29	0.391
300%	0.61	0.33	**0.430**

Table 4.4. Classifiers different badge data test result of bad yield class instances

		Performance		
		R	P	F
Tomek+CNN Under-sampling		1.00	0.22	**0.361**
CNN Under-sampling		0.04	0.20	0.065
SMOTE+Random Under-sampling				
	50%	0.54	0.21	0.301
	100%	0.35	0.15	0.209
	150%	0.42	0.21	0.278
	200%	0.46	0.20	0.279
	250%	0.46	0.24	**0.320**
	300%	0.31	0.16	0.211
SMOTE+CNN Under-sampling				
	50%	0.58	0.21	0.303
	100%	0.58	0.25	0.345
	150%	0.77	0.25	0.377
	200%	0.88	0.26	0.407
	250%	0.77	0.24	0.364
	300%	0.92	0.26	**0.410**
SMOTE+SMaRT(TOMEK+CNN)				
	50%	0.58	0.21	0.306
	100%	0.62	0.23	0.330
	150%	0.50	0.20	0.289
	200%	0.58	0.22	0.323
	250%	0.62	0.23	**0.333**
	300%	0.65	0.22	0.330
SMOTE+SMaRT(CNN)				
	50%	0.27	0.24	0.255
	100%	0.38	0.21	0.270
	150%	0.62	0.25	**0.352**
	200%	0.42	0.20	0.268
	250%	0.42	0.20	0.272
	300%	0.46	0.23	0.304

5 CONCLUSION

We conclude that K* based entropy similarity distance function perform better than VDM for our visualized data sets. Our suggested approach of SMOTE+SMaRT also improved the classification robustness compared to our previous approaches.

A study to improve the class precision without sacrificing class recall of the minority instances is very crucial in order to extend further improve the classifiers robustness. Hence, a method on how to handle with the redundant, borderline, noisy instances and also to effectively generate synthetic instances should be made as the main focus to achieve it.

Another approach that can be considered for the improvement is by selecting the best classifiers performer at their respective area and combines them as one classifier to perform the best prediction accordingly.

6 ACKNOWELDGEMENT

This project is funded by FGRS (Fundamental Grant Research Scheme) from Malaysia Higher Ministry of Education for Intelligent Predictive Maintenance System for Hard-Disk Media Manufacturing Using Data Mining Approach project.

7 REFERENCES

Cleary, J. G., & Leonard E. T. (1995) {K}*: an instance-based learner using an entropic distance measure. *Proc.12th International Conference on Machine Learning,* pp 108—114.

Kamei Y., Monden A., Matsumoto S., Kakimoto T., Matsumoto, K. (2007). The Effects of Over and Under Sampling on Fault-prone Module Detection, *First International Symposium Empirical Software Engineering and Measurement*, pp196 – 204.

Yoon, K. & Kwek, S. (2007). A data reduction approach for resolving the imbalanced data. *Issue in functional genomics Journal of Neural Computing & Applications, 16 (3),* pp 295-306

Kubat, M. & Matwin, S. (1997). Addressing the Curse of Imbalanced Training Sets: One-Sided Selection, *Proc.14th Int'l Conf. on Machine Learning (ICML'97)*, pp.179-186, Nashville, USA.

Megat Norulazmi Megat Mohamed Noor & Shaidah Jusoh (2008). Visualizing the Yield Pattern Outcome for Automatic Data Exploration, *2008 Second Asia International Conference on Modeling & Simulation,* pp. 404-409, Kuala Lumpur, Malaysia

Megat Norulazmi Megat Mohamed Noor & Shaidah Jusoh. (2008). Handling Imbalance Visualized Pattern Dataset for Yield Prediction, *Proceedings of the 3rd International Symposium on Information Technology,* Kuala Lumpur, Malaysia

Mierswa, I., Wurst, M., Klinkenberg, R., Scholz, M, & Euler, T. (2006). YALE: Rapid Prototyping for Complex Data Mining Tasks, *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-06,* pp 935-940, Philadelphia, PA, USA

Nitesh V. C., Bowyer, K.W., Hall, L.O., & Kegelmeyer, WP. (2002). SMOTE: Synthetic Minority Over-Sampling Technique. *Journal of Artificial Intelligence Research* 16, pp. 321–357

Tomek, I. (1976). Two Modifications of CNN, *IEEE Trans.Systems, Man and Cybernetics, SMC.6*, pp. 769-772.

Witten, I. H. & Frank, E. (1999). *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*, Morgan Kaufmann, San Francisco, CA, USA.