Steady State RF Fingerprinting for Identity Verification: One Class Classifier versus Customized Ensemble

Barnard Kroon[1], Susan Bergin[1], Irwin O. Kennedy[2], and Georgina O'Mahony Zamora[1]

[1] Department of Computer Science, National University of Ireland Maynooth, Co. Kildare, Ireland
[2] Bell Laboratories, Alcatel-Lucent, Blanchardstown Industrial Park, Dublin 15, Ireland

Abstract. Mobile phone proliferation and increasing broadband penetration presents the possibility of placing small cellular base stations within homes to act as local access points. This can potentially lead to a very large increase in authentication requests hitting the centralized authentication infrastructure unless access is mediated at a lower protocol level. A study was carried out to examine the effectiveness of using Support Vector Machines to accurately identify if a mobile phone should be allowed access to a local cellular base station using differences imbued upon the signal as it passes through the analogue stages of its radio transmitter. Whilst allowing prohibited transmitters to gain access at the local level is undesirable and costly, denying service to a permitted transmitter is simply unacceptable. Two different learning approaches were employed, the first using One Class Classifiers (OCCs) and the second using customized ensemble classifiers. OCCs were found to perform poorly, with a true positive (TP) rate of only 50% (where TP refers to correctly identifying a permitted transmitter) and a true negative (TN) rate of 98% (where TN refers to correctly identifying a prohibited transmitter). The customized ensemble classifier approach was found to considerably outperform the OCCs with a 97% TP rate and an 80% TN rate.

Keywords: Machine Learning, Classification, Ensemble Classifiers, Support Vector Machines, One Class Classifiers.

1 Introduction

The increase in broadband penetration and mobile phone proliferation allows for the deployment of Femto cellular base stations directly into the home, allowing the owners to make mobile calls using their broadband connection. The reception area from a Femto cell would however allow external traffic to access the

L. Coyle and J. Freyne (Eds.): AICS 2009, LNAI 6206, pp. 198–206, 2010.

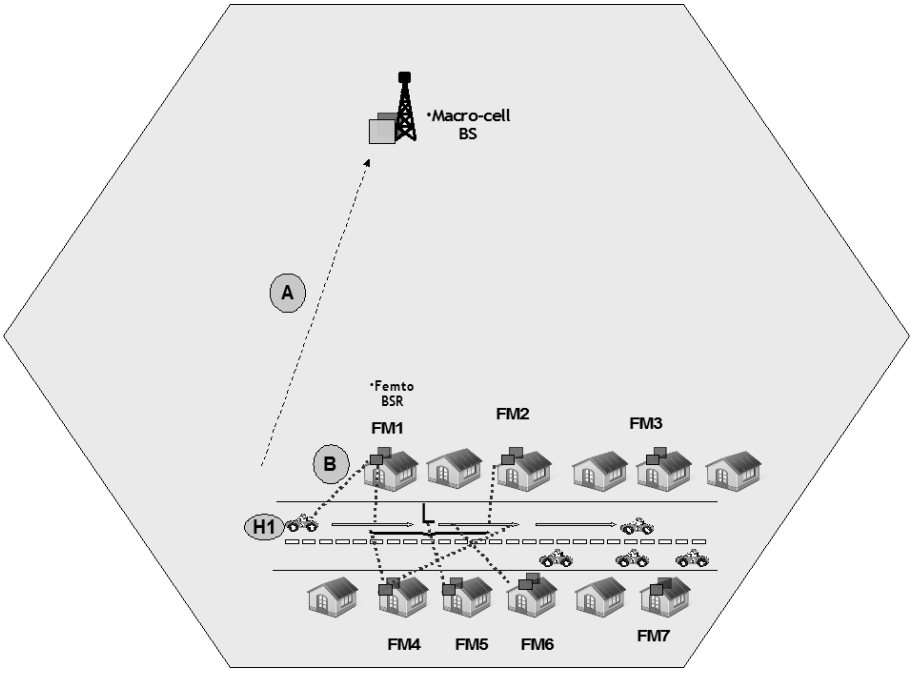

Fig. 1. Example Femto Deployment

base station, which results in an increase in authentication traffic to determine their access rights. This increase can be exponential and runs the serious risk of overloading the cellular service provider's authentication backend[1]. Figure 1 shows the deployment of Femto stations FM1 through FM7 within the larger macro cell serviced by base station BS. Passing mobiles, such as H1, will rapidly establish connections between the Femto cells and the macro cell. Each connection requires an authentication request. Using local authentication on the Femto itself without a need to access the authentication server would relieve this possible problem. Filters, power amplifiers, inductors, capacitors, PCB materials and soldering used in the manufacture of a transmitter all imbue unique characteristics onto the transmitted signal, and using these differences a transmitter can be identified using Radio Frequency (RF) fingerprinting. The 3GPP UMTS standard recommends the uniquely identifying IMSI (International Mobile Subscriber Identifier) should not be sent in plain text over the air. We discuss this and other motivating aspects of the problem in more depth in papers published previously by the authors[2,3].

There are typically two ways of identifying RF signal sources, transient state analysis and steady state analysis[4]. Transient state analysis is more commonly used and operates by detecting transient state signals which are generated as a transmitter is powered up for the first time. In data communications different communicating entities have limited common signal portions and as such steady

state analysis is a less commonly used method. The Random Access Channel (RACH) preamble is used in mobile communication to synchronize communication and in our novel approach is used as a steady state signal common to all transmitters.

The rest of the paper is organized as follows. In Section 2 we provide the problem context and the specific problem addressed in this work. Section 3 describes the various classification algorithms selected and designed for experimentation. In Section 4 we describe the experimental setup and implementation, followed by results and discussion in Section 5. Finally, we conclude in Section 6.

2 Problem Definition

This work addresses a Femto deployment issue that we refer to as the *five in the house problem*. Simply stated this is where we have five known handsets, (the quantity is assumed to be representative of the average number of mobile phones in a household), that all require access to a Femto base station. The challenge is to distinguish these five known handsets from any other handset which might come into range of the Femto cell. Classification under these specific conditions has two additional constraints. Denying access to a permitted handset is completely unacceptable, so much so that it is to be avoided even if it means allowing additional unwanted (prohibited) handsets onto the Femto. These prohibited handsets will later be denied by the traditional authentication measures. This adds the requirements that the true positive (TP) rate must be as close to 100% as possible, where TP refers to correctly identifying that the handset belongs to the house and should be allowed access to the Femto base station. Additionally, although of lesser importance, a high true negative (TN) rate must be achieved where TN refers to correctly identifying where a handset does not belong to the house and should be refused access.

Previous work on RF fingerprinting has focussed mainly on identifying different mobile phone handsets by manufacturer and by model[2,3]. While this work is relevant, the approach used is not sufficient for the *five in the house problem*.

3 Classification

In this Section we describe a number of different approaches to tackling the *five in the house* classification problem. All approaches are based on the Support Vector Machine, so we start in Section 3.1 by providing an overview of a Support Vector Machines and in Section 3.2 we describe an alternative set of classifier implementations.

3.1 Support Vector Machines

Support Vector Machines (SVMs) are a relatively recent set of supervised machine learning algorithms that have been shown to have either equivalent or significantly better generalization performance than other competing methods

on a wide range of classification problems [5]. They can be used to classify linearly separable data using the original input space or non-linearly separable data by mapping to a higher dimensional feature space in which a linear separator can be found.

In a typical binary classification problem composed of a training dataset $\{(\mathbf{x_1},y_1),(\mathbf{x_2},y_2),...,(\mathbf{x_m},y_m)\}$ where $\mathbf{x_i} \in \Re_\mathbf{d}$ and $y_i \in \{\pm 1\}$, SVMs seek a solution to the following Lagrangian optimization function:

$$W(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j) \tag{1}$$

subject to the following constraints

$$C \geq \alpha_i \geq 0 \quad \forall i \qquad and \qquad \sum_{i=1}^{m} \alpha_i y_i = 0. \tag{2}$$

C is an optional parameter that controls the trade off between allowing training errors and forcing rigid margins. That is, it represents a soft margin that allows some misclassifications which can be beneficial in noisy datasets. Where a soft margin is not allowed, the constraint is simply $\alpha_i \geq 0$. K represents the kernel function and numerous choices exist, including linear, Polynomial and Radial Basis Functions (RBF). RBF SVMs are currently the most popular choice of non-linear SVM and are implemented in this study[5,6]. Once an optimal solution is found, the decision function for a new point $\mathbf{z}$ is given by

$$f(\mathbf{z}) = sign\left(\sum_{i=1}^{m} y_i \alpha_i K(\mathbf{x}_i, \mathbf{z}) + b\right). \tag{3}$$

$\mathbf{z}$ is a training example, b is the bias and non-zero α_i values represent support vectors, the points that lie closest to the hyperplane.

The 'One-against-one' [7] multiclass approach is implemented in this study as it has been shown to have comparable if not better generalized accuracy than alternative techniques and requires considerably less training time [8], [9]. The method consists of constructing an SVM for each pair of classes. Thus for a problem with n classes $n(n-1)/2$ SVMs are trained to distinguish between the samples of one class form the samples of another class. For an unknown pattern, each SVM votes for one class and the class with the highest number of votes is chosen.

SVM One Class Classifiers (OCCs) operate on a different principle. Whilst traditional classifiers are trained on 2 or more classes, OCCs are only trained on a single class of samples (positive samples) and attempt to learn the unique features of this class so that it can accurately identify an unseen sample of this class as distinct from a sample of any other class. OCC distinguish between the trained class and other samples by identifying the other samples as outliers in the distribution described by the training set.

3.2 SVM Classifier Implementations

The SVM and OCC SVM implementation utilized in this study are the implementations provided by the libSVM library[10]. However not all of the functionality required in this study was implemented in the library and the following Section describes the changes in methodology that were used.

While an SVM typically outputs only the predicted class of an unknown sample, it can be enhanced to also output a Probability Density (PD) estimate. The estimates are based on the distance each test point is from the separating hyperplane, the further the point is from the hyperplane, the higher the probability it belongs in the class [11].

It has already been shown that standard RBF kernel SVMs have a high accuracy distinguishing between different transmitters[2]. Using this knowledge coupled with the PD estimates allows us to use these classifiers on the *five in the house problem*. It was found that if the SVM was shown a handset present in the training set, a positive sample, the PD showed a high Standard Deviation (SD). Conversely if the SVM was shown an unknown handset, one not present in the training set, the PD showed a low SD. To take advantage of this difference in SD between known and unknown samples a number of different composite classifiers were constructed using the RBF SVM as a base. While these classifiers are all multi-class classifiers, they are trained only on the handsets that it is to recognize. When the classifier is tested it is shown both positive samples and negative samples. It has also been shown that handsets of the same model can be harder to distinguish than handsets of a different model[2,3]. The Tiered, Weighted and Double classifiers are constructed in such a way as to potentially allow one of the sub-classifiers to avoid having multiple identical handsets, or if unavoidable at least favor one over the other.

Single Classifier. The Single classifier is an RBF SVM, using the PD output to determine classification. Because this is a RBF SVM at least 2 handsets are required in the training set in order for the classifier to attempt to distinguish between them.

Tiered Classifier. The Tiered Classifier is slightly more complex in construction, but follows the basic principle of the Single classifier. If presented with n handsets, it consists of n RBF SVM classifiers each trained on unique group of $n-1$ handsets. As an example, if presented with handsets A, B and C then the three classifiers are each trained on the only one of the sets (A and B), (A and C) and (B and C).

This classifier requires at least 3 handsets in the training set, as each sub-classifier is effectively a Single classifier requiring a minimum 2 handsets. The resulting PD output by each sub-classifier is summed per label and then normalized to sum to 1 again before classification is made.

Weighted Tiered Classifier. The Weighted Tiered (Weighted) Classifier is constructed similarly to the Tiered Classifier, but uses weights to assign greater importance to a particular class. In each sub-classifier one unique class is weighted as

more important than the rest. This means handsets A, C and B respectively in the training sets (A and B), (A and C) and (B and C) from the Tiered Classifier example will be the weighted handsets.

Double Weighted Classifier. Following the same principle as the Weighted Classifier, the Double Weighted Classifier (Double) weights a unique pair of handsets in each classifier as more important than the rest. At least 4 handsets are required to construct this classifier, as each sub-classifier requires at least 1 class not weighted otherwise it would effectively be identical to a Tiered classifier.

4 Experimental Setup and Implementation

The RACH preambles, representing the handsets, used in the classification tasks were captured in an anechoic chamber using a Rhode and Schwarz FSQ26 signal analyzer at 20MSamples/s. An Alcatel-Lucent 2100 MHz UMTS base station, transmitting on very low power (less than 100mW), with a modified software load was used. The modified software ensured that:

1. The base station never responded to the RACH preambles thus ensuring the handset would continue the transmission of the RACH preamble ramp sequence, simplifying RACH transmission capture.
2. Modified system information blocks (SIBs), used by the UMTS standard to configure handset operation, allow us to restrict the handset to only use a single RACH preamble signature and scrambling code meaning every RACH preamble transmission contains the same digital Inphase/Quadrature (I/Q) content.

Sixty nine handsets of varied manufacturer and model were used and approximately 1200 RACH preambles captured per handset. We extracted 177 features using the frequency domain binning algorithm described in [3].

These captured RACH preambles were used in two classification tasks using the SVM OCC setup. In the first experiment 69 OCC SVMs were each trained on data from a single handset. Then they were tested using preamble samples from all 69 handsets to determine how accurately each classifier could identify individual handsets. The second experiment was to verify the results from the first experiment and to ascertain how the OCC responded to multiple handsets in the training set. Five classifiers were trained to identify 5 handsets, as opposed to only 1 handset in the first experiment, and then were again used to distinguish between the known handsets and a selection of the unknown handsets.

The third experiment used the ensemble classifiers. For each classification task the number of known handsets, referred to here as positive samples, was selected at random from a normal distribution with mean 5, representing the number of handsets present in the *five in the house problem*. The width of the distribution was chosen such that at least 2 and at most 8 handsets were chosen[1]. A test set was also chosen for each classification task, consisting of random samples

[1] The resulting distribution had a standard deviation on 1.23.

from between 10 and 69 handsets were selected as the test set, which included at least one of the training handsets. Each of the custom classifiers was trained and tested using identical training and test sets, taking into account that the Double classifier required at least 4 different handsets in the training set, and the Tiered and Weighted classifiers require at least 3 handsets in the training set.

5 Results and Discussion

The results from the two different types of classifiers are quite different. Initially the output from the OCC looks exceedingly good with an average accuracy of 98% across the classifiers. A more detailed analysis of these results, specifically focussing on TP and TN, shows that while a high TN rate of 98% is achieved the more important TP rate is exceedingly low at 50%. Table 1 outlines the performance of the OCC (accuracy, TP and TN) as well as the associated standard deviation values. This discrepancy between the accuracy and TP and TN rates stem from the unbalanced nature of the data. Approximately 98.5% of the data presented to any of the classifiers consisted of prohibited handsets, whereas only the remaining 1.5% were permitted handsets.

Table 1. Experiment 1: Single Handset One Class Classifier Results Summary

	Average	Maximum	Minimum	Standard Deviation	Median
Accuracy	97.76%	99.28%	90.14%	0.01	97.99%
True Positive	49.03%	56.70%	39.72%	0.03	48.75%
True Negative	98.48%	99.9986%	90.66%	0.01	98.72%

The second experiment further validated the previous results. The five classifiers obtained similar accuracies to the individual classifiers, specifically: 93%, 95%, 91%, 91% and 86%. The TP and TN rates remained consistent as can be seen in Table 2. These results show that the use of OCC SVMs aren't ideally suited to the *five in the house problem*.

The third experiment with the ensemble RBF SVM classifiers showed promise as the threshold, based off the SD, determining classification could be directly modified. Table 3 shows the resulting TP and TN rates associated with different

Table 2. Experiment 2: Five in the House One Class Classifier Results Summary

	True Positive	True Negative
Experiment 1	48%	96%
Experiment 2	47%	98%
Experiment 3	48%	94%
Experiment 4	45%	95%
Experiment 5	50%	89%

Table 3. Custom Classifier Potential Thresholds

Threshold	TP	TN
0.05	1.0000	0.0001
0.10	1.0000	0.1483
0.15	0.9977	0.5798
0.20	0.9962	0.6447
0.25	0.9894	0.6794
0.30	0.9705	0.7121
0.35	0.9402	0.7472
0.40	0.8735	0.7855
0.45	0.0000	1.0000
...	...	...
0.95	0.0000	1.0000

Table 4. Custom Classifier Results Summary

Classifier	Average		Standard Deviation		Tests
	True Positive	True Negative	True Positive	True Negative	
Single	96.77%	79.13%	0.031	0.115	636
Tiered	93.63%	84.89%	0.051	0.079	825
Weighted 10	93.41%	85.33%	0.052	0.086	845
Weighted 100	93.41%	85.29%	0.053	0.088	845
Double 10	93.07%	85.75%	0.053	0.074	766
Double 100	93.05%	85.77%	0.053	0.074	765

threshold values for the PD SD where any value higher than the threshold is classified as a known handset, and any lower as unknown. These results are the combined average of all the ensemble classifiers over a total of approximately 1200 experiments in total.

The threshold values between 0.15 and 0.35 are broadly in line with the requirements of the *five in the house problem*, as per Section 2, and as such 0.25 was chosen as the threshold value that would be used in the classifiers. The results from the third experiment, using only the chosen threshold, are outlined in Table 4 which shows the averages for the different classifier rates as well as their associated standard deviations and the number of tests run using that classifier.

It should be noted that the Single classifier has a higher TP rate than all the other ensemble classifiers, and that the spread on these is also lower, as evidenced by the lower standard deviation. While the associated TN rate is lower than the other classifiers', the difference in performance shows that the Single classifier is overall a better classifier. An ANOVA test, with $p = 0.01$, confirms this difference as being statistically significant.

The performance difference is partly due to the smoothing effect experienced when combining the output from the different sub-classifiers resulting in a less pronounced difference in SD for these ensemble classifiers. The net result is that

any advantage gained from the construction of these classifiers is negated by this. The Single classifier also has the added benefit that it is the easiest to implement.

6 Conclusion

We have described the *five in the house problem*, a telecommunications Femto cell system problem, and experimentally investigated a number of machine learning classifier solutions. Statistically verified experimental results show that a Single Classifier, a custom ensemble classifier based on the Probability Density output from a Support Vector Machine, achieves the best results. Based on 636 tests, the Single Classifier achieves the best combination of True Positive and True Negative results, 97% and 79% respectively. The result offers great encouragement for more research, including the possibility of combining the results from multiple RACH preambles to further improve accuracy.

References

1. Ho, L., Claussen, H.: Effects of user-deployed, co-channel femtocells on the call drop probability in a residential scenario. In: IEEE International Symposium on Personal (September 2007)
2. O'Mahony Zamora, G., Bergin, S., Kennedy, I.O.: Using support vector machines for passive steady state rf fingerprinting. In: International Joint Conferences on Computer, Information, and Systems Sciences, and Engineering (2008)
3. Kennedy, I.O., Scanlon, P., Buddhikiot, M.: Passive steady state rf fingerprinting: A cognitive technique for scalable deployment of co-channel femto cell underlays. In: Proceedings IEEE Conference on Dynamic Spectrum Access Networks (October 2008)
4. Gerdes, R., Daniels, T., Mina, M., Russell, S.: Identification via analog signal fingerprinting: A matched filter approach. In: ISOC Network and Distributed System Security Symposium (2006)
5. Burges, C.: A tutorial on support vector machines for pattern recognition. Knowledge Discovery and Data Mining 2(2), 121–167 (1998)
6. ScholKopf, B., Smola, A.J.: Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond. MIT Press, Cambridge (2002)
7. Knerr, S., Personnaz, L., Dreyfus, G.: Single-layer learning revisited: a stepwise procedure for building and training a neural network. Neurocomputing: Algorithms, Architectures and Applications (1990)
8. Hsu, C.-W., Lin, C.-J.: A comparison of methods for multiclass support vector machines. IEEE Transactions on Neural Networks 13(2), 415–425 (2002)
9. Milgram, J., Cheriet, M., Sabourin, R.: 'one against one' or 'one against all': Which one is better for handwriting recognition with svms? In: Tenth International Workshop on Frontiers in Handwriting Recognition (2006)
10. Chang, C.-C., Lin, C.-J.: LIBSVM: a library for support vector machines (2001), Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`
11. Platt, J.: Probabilistic outputs for support vector machines and comparison to regularized likelihood methods. In: Advances in Large Margin Classifiers, pp. 61–74 (2000)