Nanotechnology Perceptions
ISSN 1660-6795
www.nano-ntp.com

A framework Utilizing Machine Learning to Dynamically Quantify the Co-Residency Risk Level for Specific VMs

Rajkumar P[1], Dr.G.Thippanna[2], Dr.A.Pratapa Reddy[3]

[1]Research Scholar Department of Computer Science and Engineering Niilm University Haryana.
[2]Associate professor Department of Computer Science and Engineering Niilm University Haryana
[3]Co- Supervisor, Associate professor , Department of Computer Science and Engineering, SVS Group of Institutions.

this approach was limited as it relied on simulated data and calculated co-residence risk based on the vulnerabilities of each VM, which is insufficient. In reality, the co-residence risk level of VMs is determined by the service subscribers who own them. To address this issue, we aim to build a fine-grained model that better quantifies co-residence risk based on service subscriber data. Additionally, performance and adaptability to a dynamic environment are critical factors for our proposed framework.
Keywords: VMs, Machine Learning techniques, Features metrics.

1. Introduction

Pre-Process

The Pre-Process component involves preparing the raw data collected by the service provider. This step includes cleaning, organizing, and applying feature metrics to the data. These feature metrics are crucial as they help in accurately profiling and classifying service subscribers. Service providers can tailor these metrics to suit their unique requirements, ensuring that the processed data is relevant and useful for subsequent analysis.

Clustering

The Clustering component uses the processed data to categorize service subscribers. This involves applying a chosen clustering algorithm to identify patterns and group similar

subscribers together. The resulting clusters represent candidate categories of subscribers, which are then subject to further analysis and partial labeling to refine the classification.

Our framework aims to provide a more precise quantification of co-residence risk by incorporating service subscriber data into the risk assessment model. By addressing the limitations of previous approaches and ensuring adaptability to dynamic environments, we strive to enhance the security and reliability of cloud computing systems.

Figure1: illustrates the diagram of our proposed framework for classifying service subscribers and quantifying the Co-Resident Risk Rate. It comprises five essential components to generate a quantified co-residence risk rate and one optional component to enhance adaptability to practical environments. Below is a brief description of each component:

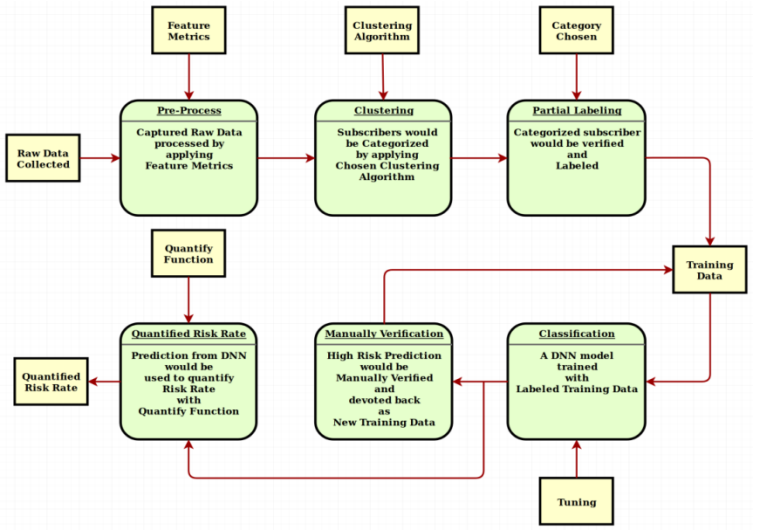

Figure 1: Overview of Framework

Partial Labeling

In this component, the service provider examines candidate categories of subscribers and determines the labeling principle for each category. Categories deemed irrelevant to the quantification task can be discarded. In our experiment, each category was automatically assigned a dummy code, preparing the training dataset for the classification component.

Classification

With the prepared training data, a Deep Neural Network (DNN) is trained to classify incoming subscribers. Hyper-parameter tuning is performed to enhance training efficiency and model accuracy.

Quantified Risk Rate

The risk rate of incoming subscribers is evaluated based on the classification predictions and a pre-defined quantification function. In our experiment, the probability that a subscriber belongs to a high-risk group is directly used as the Quantified Risk Rate.

Manual Verification

This optional component enhances framework adaptability to real-world conditions. Service providers can manually verify new subscriber data predictions and integrate this feedback into the training data, ensuring the DNN model remains up-to-date with the latest environment.

2. Feature Metrics to Profile Normal Service Subscribers

In our proposed framework, the feature metrics are used in the Pre-Process component to handle collected raw data. Service providers can customize these feature metrics to meet their unique requirements. In our experiments, we proposed an eight-dimensional feature metrics model to profile service subscribers effectively:

- N - The total number of VMs created and deployed by a specific service subscriber.
- T - The average interval time between starting two VMs. Note that this is the time between starting the ith VM and the (i+1)th VM, rather than the time between stopping the ith VM and starting the (i+1)th VM.
- M - The median memory size among VMs for a specific service subscriber.
- A - The overall active rate for a specific service subscriber. This will be explained in more detail in the following section.
- W - The average number of active VMs at each time stamp for a specific service subscriber.
- I - The median of the average CPU utilization rate among all VMs at each time stamp for a specific service subscriber.

Features 1 to 4 provide an overall analysis of each service subscriber, offering a broad view of their behavior. Features 5 and 6 offer more detailed insights, allowing us to profile each subscriber's behavior pattern accurately and build a detailed characteristic image.

Quantifying Co-Residency Risk

In our framework, the quantification component utilizes softmax activation to output category probabilities. We utilize the probability rate of the normal category to quantify co-residency risk, indicative of deviations from normal behavioral patterns.

3. Experimental Results and Evaluation

Experiments were conducted on a Dell Precision Tower T5810 Workstation, featuring an Intel Xeon E5-1620, 32GB RAM, and Nvidia Quadro P5000 Graphic Card for GPU acceleration, significantly reducing training time.

We employed the Azure Public Dataset, providing a real-world large-scale dataset encompassing VM workload data from Microsoft Azure. This dataset comprises over 12,000 service subscribers, 5 million VMs, and 3.1 billion CPU utilization records sampled every

five minutes over one month, totaling over 500GB. Analysis of the dataset revealed valuable insights, facilitating our evaluation process.

A - Overall Active Rate: The observation from Figure 2 highlights that the CPU utilization rate of over 90% of VMs remains below 15%, indicating that the majority of VMs operate at very low workload levels.

Upon computing the Overall Active Rate for all subscribers, the cumulative distribution diagram presented in Figure 3 is examined. Broadly speaking, the analysis reveals that over 80% of subscribers exhibit an active rate of less than 10%.

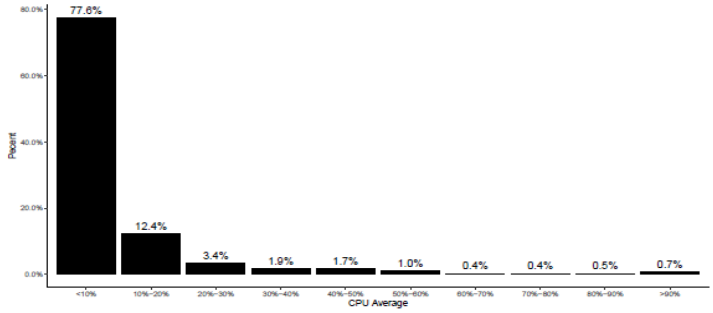

Figure 2: Average CPU Utilization distribution among VMs

W - Average Active VM amount After computing the Average Active VM amount for all subscribers, the cumulative distribution diagram depicted in Figure 4 is examined. By integrating this information with feature A, we can identify and filter out the type of extreme active subscribers.

I - Median of Average CPU Utilization Rate Additionally, upon reviewing the cumulative distribution diagram in Figure 5 for the Median of Average CPU Utilization Rate, it becomes evident that over 90% of subscribers fall into the inactive type category, as depicted in Figure 6.

Through the analysis of the aforementioned feature metrics, we have gained a preliminary understanding of service subscribers. In the subsequent section, we will utilize our proposed feature metrics to execute the task of clustering subscribers.

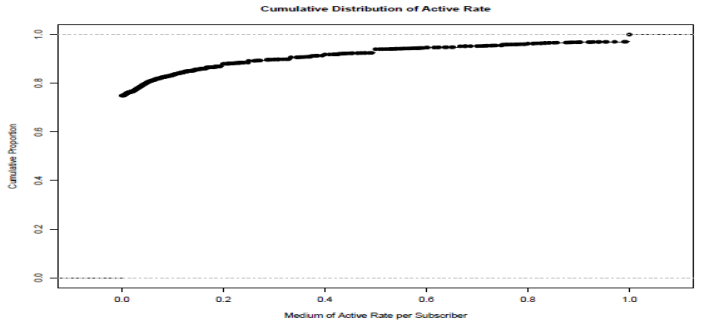

Figure 3: Cumulative Distribution of Active Rate per Subscriber

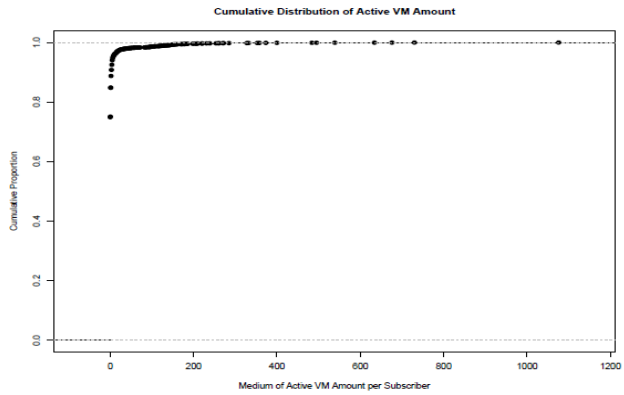

Figure 4: Cumulative Distribution of Average Active VM Amount per Subscriber

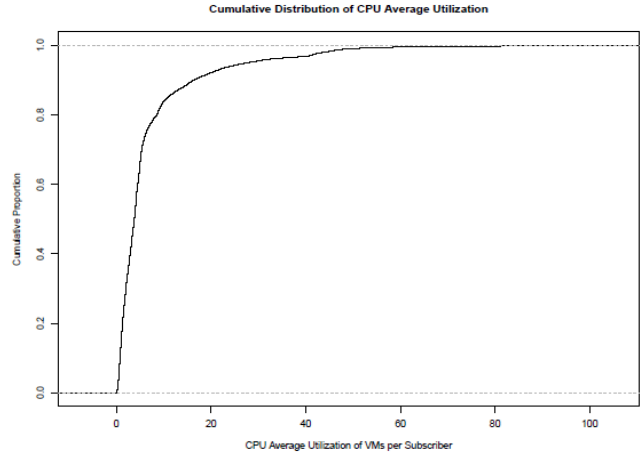

Figure 5: Cumulative Distribution of CPU Average Utilization per Subscriber

Clustering of Subscribers

Given the extensive nature of the Azure Dataset, encompassing over 12,000 service subscribers, our objective is to cluster them into several candidate groups for future utilization.

Aspect from Activeness Rate

Initially, drawing insights from Figure 6, it is apparent that approximately 65% of subscribers have created only one VM, while another 25% have created fewer than five VMs throughout their subscription tenure. When combined with the cumulative distribution of feature I, it can be inferred that most service subscribers maintain a small number of VMs operating at the inactive level.To validate this hypothesis, we conducted an analysis of the active rate curve across all timestamps for all subscribers. Based on the observations derived from these curve diagrams, we delineated several categories of subscribers solely based on their activeness rate:Single VM Subscribers: Constituting approximately 65% of the subscriber base, these individuals or entities have deployed only one VM, indicating minimal

activity within the Azure environment. Low VM Count Subscribers: Accounting for about 25% of subscribers, this group has created fewer than five VMs during their subscription period, suggesting limited engagement with the platform.These initial categorizations based on activeness rate provide a foundation for further exploration and refinement of subscriber clustering.

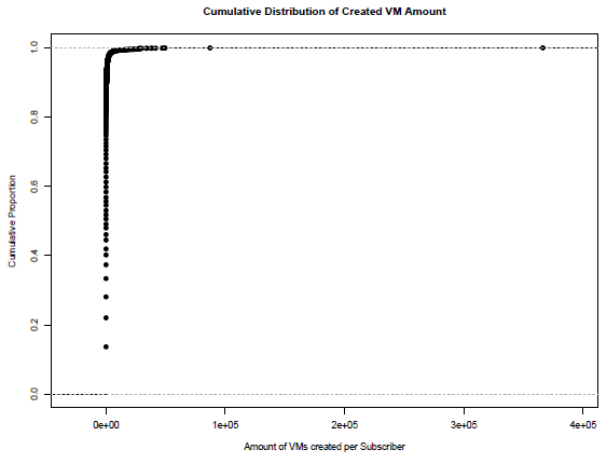

Figure 6 Cumulative Distribution of Created VMs Amount per Subscriber

• Inactive Subscriber: Illustrated at the uppermost section of Figure 7, this curve exemplifies the characteristic behavior of inactive subscribers as defined. Over 80% of subscribers fall into this category. These subscribers create very few VMs, and most of the time, these VMs remain inactive.

• Period Active Subscriber: Positioned in the middle portion of Figure 7, this curve represents the typical behavior of period active subscribers. These subscribers create multiple VMs, and the level of activity fluctuates significantly over their subscription period.

• Extreme Active Subscriber: The curve at the very bottom of Figure 7 signifies the behavior of extreme active subscribers. Typically, these subscribers maintain an exceptionally high level of activity throughout their subscription lifetime.

In our experiments, DBSCAN served as an initial clustering tool for subscribers. Throughout the experimental process, we tested the MinPts parameter within the range of 3 to 20. Considering our assumption that over 99% of the data should originate from normal service subscribers, we anticipated the presence of a significant majority of such subscribers within the dataset.

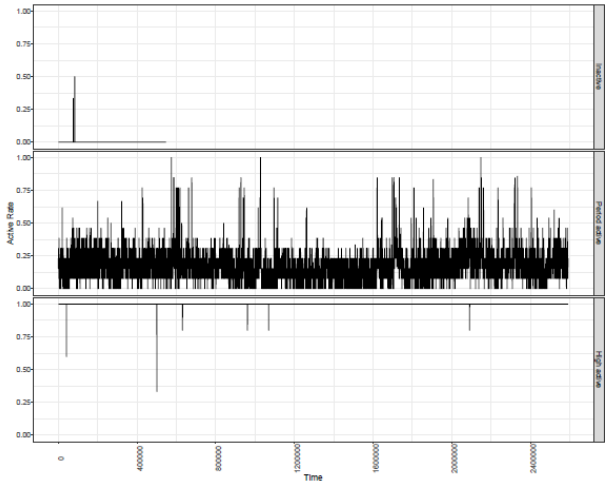

Figure 7: Subscriber categorized with activeness level

After conducting the initial clustering, we found that one clustering output encompassed a significant portion of the subscribers. To achieve this, we set the MinPts parameter to 5 and ε to 1.5. Following the initial clustering, we manually inspected the activeness level curves of all users throughout their subscription lifetimes. Based on this examination, we separated subscribers into three major categories, as depicted in Figure 7. It's worth noting that if an attacker were to simulate as a normal user, the associated costs would be substantial. In our future work, we plan to utilize detailed curve-level diagrams in our Convolutional Neural Network sub-module to prevent the loss of crucial information due to over-abstraction.

Additionally, we demonstrated the coefficient relationship between extreme active subscribers and others in Figure 8. Our clustering results are illustrated in Figures 9 and 10. Notably, the majority of service subscribers were clustered into one category, aligning well with our expectations. Of particular significance, we identified a total of 80 subscribers as outliers. Upon examining their detailed active rate curves, we believe their behavioral patterns closely resemble those of potential high-risk users.

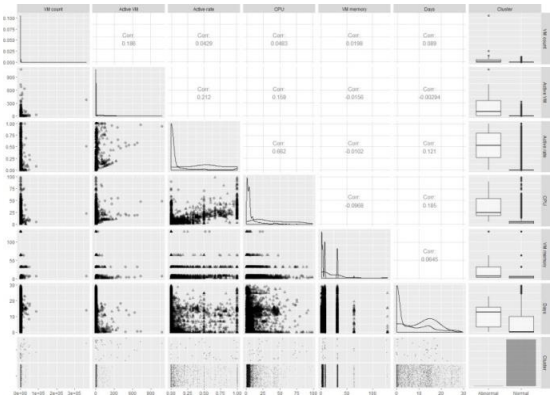

Figure 8: Cluster Pairing

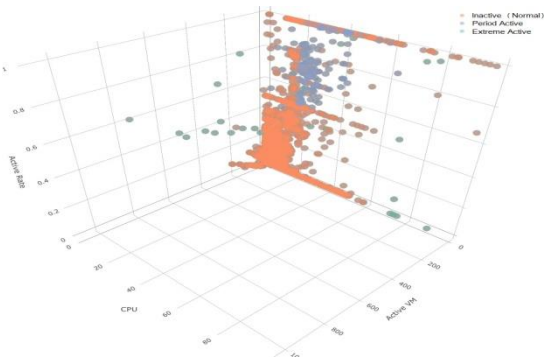

Figure 9: Cluster Result

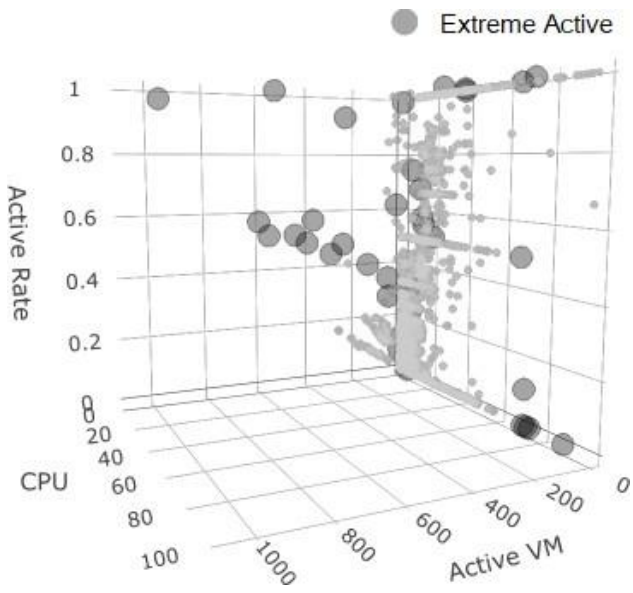

Figure 10: Cluster Result (Extreme Active)

4. Conclusion

A dynamic adaptive framework for quantifying co-resident risks, leveraging feature metrics to profile the behavior patterns of normal service subscribers in the cloud. Through the use of clustering algorithms and manual labeling, you categorize subscribers into Inactive, Period Active, and Extreme Active groups, facilitating the training of a classification component. This component demonstrates robustness to new data, achieving an impressive 98% accuracy rate for the test dataset. The validation of its performance through F Measuring Matrix analysis further underscores the reliability of your classification approach.

References

1. Qayyum et al., Securing machine learning in the cloud: a systematic review of cloud machine learning security. Front. Big Data 3(November) (2020). https://doi.org/10.3389/fdata.2020.587139
2. Alouffi, B., Hasnain, M., Alharbi, A., Alosaimi, W.: A Systematic Literature Review on Cloud Computing Security: Threats and Mitigation Strategies. IEEE Access, 9, pp. 57792-57807, 2021.
3. Abdulsalam, Y.S., Hedabou, M.: Security and Privacy in Cloud Computing: Technical Review. Future Internet 2022, 14, 11.
4. George, S.S., Pramila, R.S.: A review of different techniques in cloud computing. Materialstoday proceedings, 46, pp. 8002-8008, 2021.
5. Attaran, M., Woods, J.: Cloud computing technology: improving small business performance using the Internet. Journal of Small Business & Entrepreneurship. 13. pp. 94-106, 2018.
6. Basu, S., Bardhan, A., Gupta, K., Saha, P., Pal, M., Bose, M., Basu, K., Chaudhury, S., Sarkar, P.: Cloud computing security challenges & solutions-A survey. Annual Computing and

Communication Workshop and Conference (CCWC), 2018.
7. Y. Han, J. Chan, T. Alpcan, and C. Leckie. Using virtual machine allocation policies to defend against co-resident attacks in cloud computing. IEEE Transactions on Dependable and Secure Computing, 14(1):95–108, Jan 2017.
8. M. M. Hasan and M. A. Rahman. Protection by detection: A signaling game approach to mitigate co-resident attacks in cloud. In 2017 IEEE 10th International Conference on Cloud Computing (CLOUD), pages 552–559, June 2017.
9. Geoffrey E. Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detec- tors. CoRR, abs/1207.0580, 2012.
10. H. Hlavacs, T. Treutner, J. Gelas, L. Lefevre, and A. Orgerie. Energy consumption side-channel attack at virtual machines in a cloud. In 2011 IEEE Ninth International Conference on Dependable, Autonomic and Secure Computing, pages 605–612, Dec 2011.
11. Joseph Douglas Horton, RH Cooper, WF Hyslop, Bradford G. Nickerson, OK Ward, Robert Harland, Elton Ashby, and WM Stewart. The cascade vulnerability problem. Journal of Computer Security, 2(4):279–290, 1993.