ISSN: 2959-6386 (Online), Vol. 1, Issue 1

Journal of Knowledge Learning and Science Technology

journal homepage: https://jklst.org/index.php/home

JKLST
EXPLORING THE INTERSECTION OF KNOWLEDGE, LEARNING, AND SCIENCE & TECHNOLOGY:
UNLEASHING THE POWER OF SYNERGY IN THE 21ST CENTURY

The Synergy of Data Engineering and Cloud Computing in the Era of Machine Learning and AI

Jawaharbabu Jeyaraman[1] Muthukrishnan Muthusubramanian[2]
[1]TransUnion – USA
[2]Discover Financial Services-USA

Abstract

The integration of data engineering and cloud computing has become increasingly vital in harnessing the potential of machine learning (ML) and artificial intelligence (AI) technologies. This paper explores the symbiotic relationship between data engineering and cloud computing, elucidating how their synergy facilitates the development and deployment of ML and AI solutions. By leveraging the scalability, flexibility, and accessibility of cloud infrastructure, organizations can efficiently manage, process, and analyze vast amounts of data, thereby fueling the advancement of ML and AI initiatives. Furthermore, the convergence of data engineering techniques with cloud-based services enables seamless integration of disparate data sources, enhances data quality, and streamlines data pipelines, laying the groundwork for robust ML and AI models. This paper discusses key strategies, challenges, and opportunities associated with leveraging the combined power of data engineering and cloud computing to drive innovation and maximize the potential of ML and AI technologies.

Keywords: Data Engineering, Cloud Computing, Machine Learning, Artificial Intelligence, Data Pipelines, Scalability, Integration, Innovation.

ArticleInformation:
Article history: *Received:*10/10/2022 *Accepted:*12/10/2022 *Online:*30/10/2022 *Published*: 30/10/2022
Doi: https://doi.org/10.60087/jklst.vol1.n.p75
Corresponding author: Jawaharbabu Jeyaraman

Introduction

In today's data-driven landscape, the fusion of data engineering and cloud computing stands as a cornerstone in the realm of machine learning (ML) and artificial intelligence (AI). The exponential growth of data volumes coupled with the escalating demands for sophisticated analytical insights have propelled organizations to seek scalable, agile, and cost-effective solutions for managing and processing data. Amidst this backdrop, the synergy between data engineering and cloud computing emerges as a catalyzing force, enabling enterprises to unlock the full potential of ML and AI technologies.

Data engineering, encompassing the processes, tools, and techniques for acquiring, transforming, and storing data, forms the bedrock of any successful data-driven initiative. Concurrently, cloud computing has revolutionized the IT landscape by offering on-demand access to computing resources, storage, and services, thus obviating the need for substantial upfront investments in infrastructure. The confluence of these disciplines heralds a new era of innovation, where organizations can leverage scalable, resilient, and elastic cloud platforms to design, deploy, and operationalize ML and AI solutions at unprecedented speed and efficiency.

In this context, this paper delves into the symbiotic relationship between data engineering and cloud computing, exploring how their integration fosters the development, deployment, and operationalization of ML and AI models. Through a comprehensive analysis of key strategies, challenges, and opportunities, this paper aims to illuminate the transformative potential of harnessing the combined power of data engineering and cloud computing in driving innovation and enabling data-driven decision-making.

By elucidating the synergies between these domains, organizations can gain valuable insights into optimizing their data workflows, enhancing scalability, and accelerating the time-to-market for ML and AI initiatives. Moreover, understanding the intricate interplay between data engineering and cloud computing is paramount for navigating the complexities inherent in today's data-intensive environments and harnessing the transformative capabilities of ML and AI technologies.

In the subsequent sections, we delve deeper into the intricacies of data engineering and cloud computing, elucidating their individual contributions and synergistic effects in the context of ML and AI. Additionally, we examine real-world use cases, best practices, and emerging trends to provide a holistic perspective on the integration of data engineering and cloud computing in the era of ML and AI.

Advancements in Cloud Computing Architecture and Intrusion Detection Systems

This section provides an overview of state-of-the-art Cloud Computing (CC) architectures, Intrusion Detection Systems (IDS), Machine Learning (ML) methods, and relevant research aimed at enhancing IDS and cloud security.

Cloud computing encompasses Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS) models, each offering distinct technical layers. IaaS facilitates the provisioning of virtual machines (VMs) and scalability of storage, networks, and load balancers. PaaS, on the other hand, offers middleware instances and execution contexts such as databases and application servers, while SaaS provides software accessible over the internet on-demand.

Different cloud deployment models cater to varied organizational needs. Public clouds serve multiple clients publicly, while private clouds are dedicated to a single entity. Hybrid clouds amalgamate public and private cloud resources, and community clouds support collaboration among multiple companies with similar requirements. Notably, private clouds are often considered more secure due to their limited user base .

Intrusion Detection Systems (IDS) play a crucial role in safeguarding cloud environments against unauthorized activities that may compromise data confidentiality, integrity, and availability. IDSs employ two fundamental detection methods: Network IDS and Host IDS, which respectively monitor network and host machine activities . These systems employ misuse-based detection to identify known attacks and anomaly-based detection to detect unknown threats, often combining both approaches for comprehensive coverage.

Machine Learning (ML) techniques enhance IDS capabilities by enabling computers to learn patterns in data and make predictions without explicit programming. ML encompasses supervised, unsupervised, and semi-supervised learning methods. Supervised learning utilizes labeled data to predict unseen instances, while unsupervised learning identifies patterns in unlabeled data. Semi-supervised learning lies between these two paradigms. Notably, Deep Learning (DL), a subset of ML, relies on artificial neural networks to learn data representations .

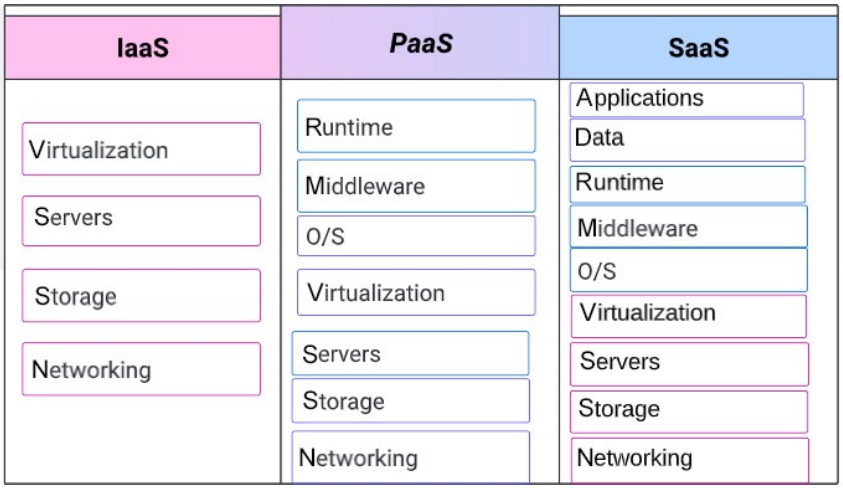
While firewalls are commonly used for intrusion detection in cloud environments, they may fall short in detecting insider threats . Consequently, researchers employ DL and ML techniques to bolster cloud security . For instance, studies utilize classifiers such as K-Nearest Neighbors (KNN), Decision Trees (DT), Support Vector Machines (SVM), and Random Forests (RF) to detect botnet attacks . Additionally, DL-based IDS employing deep neural networks demonstrate high accuracy in detecting intrusions.

Moreover, software-defined networking IDS and two-stage DL techniques have been proposed for detecting anomalies in cloud environments and autonomous vehicles . These advancements underscore the growing importance of integrating ML and DL techniques with traditional IDS approaches to fortify cloud security and mitigate emerging threats.

In their study, Mishra et al. introduced a classification-based Machine Learning (ML) approach for detecting Distributed Denial of Service (DDoS) attacks in Cloud Computing (CC). Employing methods such as K-Nearest Neighbors (KNN), Random Forest (RF), and Naïve Bayes (NB), their proposed model achieved an impressive accuracy of 99.76%, with RF yielding the most promising results. Alshammari and Aldribi [22] similarly utilized ML techniques to bolster Intrusion Detection Systems (IDS) aimed at identifying malicious network traffic within cloud environments, leveraging the ISOT-CID dataset for evaluation. Jiang et al. evaluated the efficacy of an attack detection system using the NSL-KDD dataset, highlighting Long Short-Term Memory (LSTM) and Recurrent Neural Networks (RNNs) as optimal choices for multichannel IDS, reporting efficiency at 99.23% and accuracy at 98.94%.

To learn from privacy-preserved encrypted data on the cloud, Khan et al. employed supervised and unsupervised ML techniques, specifically Artificial Neural Networks (ANNs), over scrambled information. Chiba et al. proposed a cooperative and hybrid network intrusion detection framework, merging signature-based detection (SNORT) with anomaly-based detection via Optimized Back Propagation Neural Network (BPN) to enhance accuracy. Utilizing Deep Learning (DL), Kim et al. suggested an architecture for intrusion detection, employing Long Short-Term Memory (LSTM) networks on the KDD Cup 99 dataset, focusing on two classes - normal or anomaly, for efficiency.

Zhang [37] proposed an automatic technique that develops discriminative models and fuses multi-view information to enhance accuracy, while Tang et al. constructed an IDS based on DL using six basic features, achieving an accuracy of 96.93% in attack detection performance. Ahmad et al. introduced a method for cloud-based text document classification and data integrity, concluding that Random Forest (RF) outperformed other techniques such as Naïve Bayes (NB), Support Vector Machine (SVM), and K-Nearest Neighbors (KNN). More recently, Mubarakali et al. utilized SVM-based expert systems for detecting DDoS attacks, reporting system performance at 96.23%. A comparative analysis of various current IDS models is presented in Table



IaaS	PaaS	SaaS
Virtualization	Runtime	Applications
Servers	Middleware	Data
Storage	O/S	Runtime
Networking	Virtualization	Middleware
	Servers	O/S
	Storage	Virtualization
	Networking	Servers
		Storage
		Networking

Experimental Setup

Our research is conducted and assessed within a controlled experimental environment, utilizing a computer equipped with a CoreTM i5 8250U CPU operating at 1.8 GHz and 12 GB of RAM, running Windows 10 Professional 64 bits. Python 3 serves as the implementation language for the Random Forest (RF), Decision Tree (DT), and Support Vector Machine (SVM) models, preceded by graphical visualization to reduce feature dimensions. The efficacy of our proposed model is validated using the accuracy (ACC) metric and compared against that of other models.

To ensure robust evaluation, we randomly partition the entire dataset, allocating 70% for training and reserving the remainder for testing. Optimal parameters for classifier performance are determined through rigorous model training and testing. Our study utilizes the NSL-KDD and Bot-IoT datasets, addressing inherent issues present in the KDD 1999 dataset.

The NSL-KDD dataset offers several advantages over its predecessor, including the exclusion of redundant records, appropriate record selection, and the retention of the original forty-one features from the KDD'99 dataset. Leveraging these advantages, our dataset comprises 41 features, utilizing the six fundamental properties of the NSL-KDD dataset as outlined in [30].

The selected properties include:

Duration: Duration of the connection in seconds.
Protocol Type: Categorized into tcp, udp, and icmp.
Src Bytes: Data bytes sent from the source to the destination.
Dst Bytes: Number of data bytes sent between source and destination.
Count: Number of connections to the same host in the previous two seconds.
Srv Count: Number of prior two-second connections to the same service as the current connection.

The categorical variable "protocol type" is transformed into numeric values using dummy encoding. Our graphical visualization indicates that the class variable is minimally influenced by the protocol type variable.

Further analysis reveals predictive patterns, such as detecting anomalies based on specific conditions. For instance, a duration variable exceeding 1500 seconds signals anomaly detection. Similarly, anomaly class zero is predicted if the "dst bytes" variable surpasses 50,000.

Following feature selection from visualization, we reduce the feature set from 41 to two variables: src bytes and dst bytes. Initial experimentation involves developing an RF model for classifying anomalies based on these selected variables.

The Bot-IoT dataset, enriched with IoT device data, offers comprehensive insights into IoT traffic flows, including regular, IoT, and botnet traffic. Previous studies, such as that by Shafiq et al. [48], have identified top-performing variables using various ML approaches, including DT, NB, RF, and SVM, supported by measures like Pearson moment correlation and area under the curve (AUC).1.

Results:

Evaluation Metrics

Our evaluation primarily focuses on classification models, specifically binary classification, as intrusion detection relies on labeled data to predict whether an object belongs to the attack class or not. In binary classification, the results provided by algorithms are binary (0 or 1). Selecting appropriate evaluation metrics is crucial for assessing and validating Machine Learning (ML) models in such scenarios. Typically, these metrics involve comparing the actual classes with the predicted classes, enabling the interpretation of predicted probabilities for each class.

A key performance metric for classification tasks is the confusion matrix [50], which visualizes the model's predictions compared to the actual labels in tabular form. Instances of a real class are represented in rows, while instances of a predicted class are represented in columns. From the confusion matrix, various metrics such as accuracy (ACC), recall, and precision can be derived, aiding in the evaluation of our Intrusion Detection System (IDS).

Obtained Results

Initially, we implement the IDS for the classification task, with the ACC value serving as an indicator of model performance. We enhance the Random Forest (RF) classification model by identifying features that yield optimal classification outcomes. Subsequently, we utilize a subset of the NSL-KDD dataset to train the model.

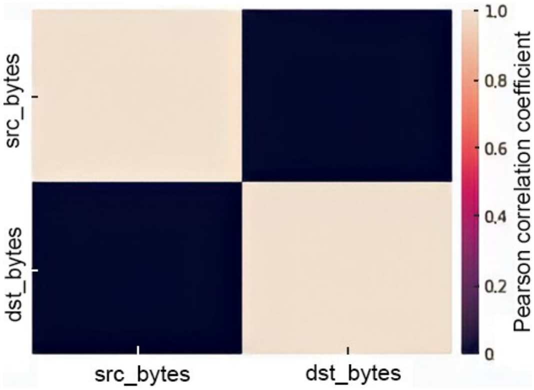
Subsequently, we commence the evaluation of our model based on two selected variables from the NSL-KDD dataset: src bytes and dst bytes, leveraging graphical visualization techniques. To validate the efficiency of the chosen variables, we employ a correlation matrix, which provides correlation values among several variables. Figure 6 illustrates the correlation matrix, aiding in assessing the interdependence of variables.

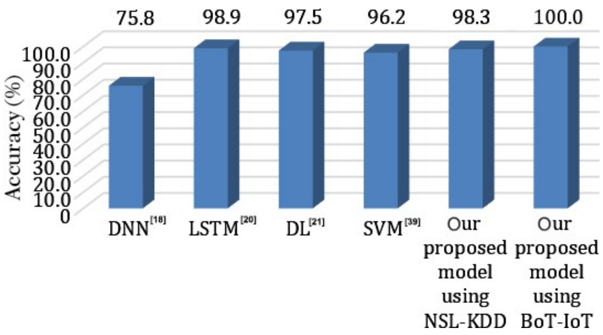
The correlation matrix illustrates the relationship between src bytes and dst bytes. From Fig. 6, it is evident that the correlation coefficient tends toward 0, indicating a negligible risk of multicollinearity between the two variables. The outcomes depicted in Fig. 7 indicate that our model performs well in terms of accuracy (ACC) and precision using src bytes and dst bytes, although recall requires improvement.

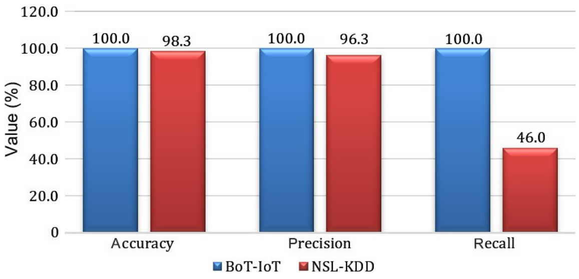
To further evaluate the effectiveness of our model, we employ the BoT-IoT dataset. After importing the data separately, we aggregate it into a new dataframe and follow the steps outlined in our model (Fig. 2). Two features, state number and stddev, are selected from the BoT-IoT dataset. The results of our tests are presented in Fig. 7, showcasing the ACC, precision, and recall metrics used to assess the performance and efficiency of our proposed model. Notably, we achieve 98.3% ACC, 96.3% precision, and 46.0% recall using the NSL-KDD dataset, while all metrics reach 100% when the BoT-IoT dataset is employed.

Fig. 8 illustrates the ACC obtained by different models using NSL-KDD, alongside the ACC of our model using both NSL-KDD and BoT-IoT datasets. Our proposed IDS demonstrates superior performance compared to works referenced in [30, 33, 35, 39], achieving higher ACC with the utilization of two selected features from NSL-KDD and BoT-IoT datasets and employing RF.

Consequently, reducing the number of explanatory variables not only diminishes data collection and execution time but also maintains high-quality results, as evidenced in Fig. 8. Overall, our RF classifier technique effectively distinguishes between normal and aberrant traffic using only two features, yielding favorable outcomes compared to DNN, LSTM, DL, and SVM.







Conclusion

Intrusion detection stands as a pivotal technology enhancing cloud security, with Machine Learning (ML) algorithms playing a crucial role in its advancement. In this study, we propose a novel approach for intrusion detection by amalgamating graphical visualization with Random Forest (RF) techniques, aimed at bolstering cloud security. Graphic visualization is employed for feature engineering, while RF is utilized for intrusion prediction and detection. Prior to model training, feature reduction to two variables is undertaken.

Our findings underscore the efficacy of the RF classifier in accurately predicting and classifying attack types, surpassing the performance of Deep Neural Networks (DNN), Decision Trees (DT), and Support Vector Machines (SVM). By contrasting results with other classifiers, we highlight the potential of utilizing a minimal feature set for intrusion detection. Nonetheless, we acknowledge the limitation of recall performance, particularly using the NSL-KDD dataset.

To address this limitation, future research endeavors will focus on leveraging Deep Learning (DL) and ensemble learning techniques to enhance our model's recall performance. Through continuous refinement and exploration of advanced methodologies, we aim to fortify the effectiveness and robustness of our intrusion detection system, further elevating cloud security standards.

References

[1] A. Verma and S. Kaushal, "Cloud Computing Security: Issues and Challenges - A Survey," in Proceedings of the First International Conference on Advances in Computing and Communications, Kochi, India, 2011, pp. 445–454.

[2] H. Alloussi, F. Laila, and A. Sekkaki, "State of the Art in Cloud Computing Security: Problems and Solutions," presented at the Workshop on Innovation and New Trends in Information Systems, Mohamadia, Morocco, 2012.

[3] J. Gu, L. Wang, H. Wang, and S. Wang, "A Novel Approach to Intrusion Detection using SVM Ensemble with Feature Augmentation," Computers and Security, vol. 86, pp. 53–62, 2019.

[4] S. Benkirane, "Road Safety against Sybil Attacks based on RSU Collaboration in VANET Environment," in Proceedings of the 5th International Conference on Mobile, Secure, and Programmable Networking, Mohammedia, Morocco, 2019, pp. 163–172.

[5] Q. Zhang, L. Cheng, and R. Boutaba, "Cloud Computing: State-of-the-Art and Research Challenges," Journal of Internet Services and Applications, vol. 1, pp. 7–18, 2010.

[6] M. K. Srinivasan, K. Sarukesi, P. Rodrigues, M. S. Manoj, and P. Revathy, "State-of-the-Art Cloud Computing Security Taxonomies: A Classification of Security Challenges in the Present Cloud Computing Environment," in Proceedings of the 2012 International Conference on Advances in Computing, Communications and Informatics, Chennai, India, 2012, pp. 470–476..

[7] A. Khraisat, I. Gondal, P. Vamplew, and J. Kamruzzaman, "A Survey of Intrusion Detection Systems: Techniques, Datasets, and Challenges," Cybersecurity, vol. 2, p. 20, 2019.

[8] A. Guezzaz, A. Asimi, Y. Asimi, Z. Tbatou, and Y. Sadqi, "Development of a Global Intrusion Detection System using PcapSockS Sniffer and Multilayer Perceptron Classifier," International Journal of Network Security, vol. 21, no. 3, pp. 438–450, 2019.