International Journal of Modern Computation, Information, and Communication Technology
2024;7(1):1-9.
ISSN: 2581-5954

Identifying and Classifying Toxic Comments in Cloud Using Amazon Web Services Comprehend Technology

R. Meenambigai[1*], N. Saranya Devi[1]

[1]*School of Continuing Education, Tamil Nadu Open University, Chennai, Tamil Nadu, India*

[*]*Corresponding author: rmeenambigai@tnou.ac.in*

Abstract. Growing concerns about toxic remarks on digital platforms need effective detection and categorization systems to make the internet safer. Amazon Web Services (AWS) Comprehend automates dangerous material discovery and categorization using cloud-based Natural Language Processing (NLP). The goal is to increase automatic content moderation and detection accuracy using AWS Comprehends deep learning. The goal is to use sentiment analysis and entity identification to categorize harmful comments into specified categories for real-time content filtering and reduce human moderation. To identify toxic language, minimize false positives, and enable proactive action, a scalable and adaptable solution is needed. AWS Comprehend classifies abusive or improper language by analyzing user-generated content context and sentiment using powerful NLP models. Cloud-based deployment supports high-volume text analysis with low latency and easy interaction with varied digital ecosystems. Using AWS's scalable infrastructure, poison detection may be continually optimized to respond to changing language trends and damaging talk. This strategy reduces toxic remarks, enforces content management, and promotes ethical communication on online platforms while adhering to platform standards, improving digital safety.

Keywords: Toxic Comment Detection, Amazon Web Services Comprehend, Natural Language Processing, Content Moderation, Cloud Technology

INTRODUCTION

Digital platforms have greatly expanded user-generated content. Although these platforms promote communication and idea exchange, they often attract poisonous remarks. Online comment toxicity may injure people, ruin user experiences, and damage platform trust. Human reviewers struggle with scalability and consistency, particularly with enormous content volumes. AWS Comprehend, a machine learning-powered NLP service, uses cloud technology to recognize and categorize harmful remarks in real time. Using AWS Comprehend for hazardous comment detection creates a cloud-based system that can analyse and categorize enormous amounts of text for dangerous content. Utilizing NLP models to identify toxicity in comments, classify them, and moderate content is the goal. The objective is to increase internet safety and quality while helping digital platforms manage user-generated content. Sentiment analysis, entity identification, and text categorization are integrated seamlessly into content moderation processes via AWS Comprehend's cloud-native capabilities. By installing hazard classification models on the cloud, the system can process, scale, and adapt across applications in real time. This technique quickly and properly addresses hazardous information, minimizing its effect on users and communities.

Section 2 discusses manual moderation limits and contextual language understanding's difficulties in finding and categorizing harmful remarks. Some of the benefits of utilizing AWS Comprehend for NLP activities include its ability to handle varied datasets and numerous languages. Section 3 describes how to classify harmful comments using AWS Comprehend. This comprises data collection, preprocessing, and pre-trained and custom machine learning model deployment. AWS services like S3 for storage, Lambda for real-time processing, and SageMaker for model training and fine-tuning are also integrated. Section 4 assesses system performance using accuracy, recall, and precision. Case studies and real-world applications highlight the system's poisonous content management efficacy. The effects of automatic moderation on user engagement and platform integrity are also discussed. Section 5 continues with a review of AWS Comprehend's cloud poisonous comment detection and categorization capabilities. Exploring multilingual toxicity detection, adding sentiment analysis for nuanced moderation, and extending the system to solve digital content moderation difficulties are future goals.

LITERATURE SURVEY

Received: 06.10.2023 Revised: 24.11.2023 Accepted: 13.12.2023

Introduction to Amazon Web Services (AWS) in the Context of Contemporary Science and Technology. The importance of AWS in cloud computing becomes clear when one examines its many uses in IT infrastructure and scientific research [1]. I am learning AI and ML technologies using AWS-based solutions. Examining how to teach ML and AI using AWS sheds insight on methods for training ML and AI experts in the classroom. By enhancing educational frameworks, this method prepares learners for careers in sectors driven by artificial intelligence [2]. A Vast Overview of Amazon Web Services Methods and Rationale. A look at AWS methods, resources, and recommendations shows how flexible cloud solutions are for improving efficiency. A wide variety of technical demands may be met by using AWS's portfolio of services, which provide customised solutions that adhere to industry standards [3]. Cloud-Driven Machine Learning on AWS: A Service Review. Machine learning services on AWS provide powerful cloud-based tools for data analysis and model training, which enable a variety of computational activities. The importance of AWS in progressing ML applications across several industries is shown by reviewing these offerings. [4].

Evaluation of Amazon Web Services' Model Deployment Features. The effectiveness of various AI/ML deployment methods is investigated in a comparative study of AWS model deployment services. By comparing several deployment services, this research helps businesses choose the best one for their requirements. [5]. Utilising Large Language Models inside AWS. With AWS's Large Language Models (LLMs) integrated, complex NLP activities like translation, summarisation, and text production are made possible. Applications that make good use of AWS's resources for different NLP workloads include content production, sentiment analysis, and customer assistance [6]. Microservices on AWS Perform Self-Healing with the Help of AI Power. CloudWatch and Hystrix are a few of the AWS infrastructure technologies that enable microservices apps to self-heal. System dependability and operational continuity are enhanced with this feature, making it crucial for organisations that rely on continuous services [7]. Cost Optimisation Strategies for AWS Infrastructure. Various optimisation methodologies are used to handle cost management in AWS systems, with the goal of decreasing expenditures without compromising performance. This method guarantees efficient and cost-effective use of cloud resources and is useful for both small-scale initiatives and large-scale AWS installations [8].

Assessing Cloud-Nuke for the Purpose of Optimising AWS Resources. Cloud-Nuke's effect on optimising AWS infrastructure by looking at how well it handles cloud resources is analysed. To make AWS consumption more sustainable, Cloud-Nuke is implemented to improve resource efficiency [9]. Integrating On-Premises and AWS Infrastructure. Hybrid cloud computing achieves operational harmony by combining on-premises infrastructure with AWS resources. Some companies need granular control, while others can't do without the scalability of the cloud [10]. Hyper automation in COVID-19 Data Analysis Using AWS. COVID-19 data management solutions are available via AWS-powered hyper automation, which includes data integration, analysis, and automation capabilities. The fact that AWS is involved in hyper automation shows that it can handle massive health emergencies well [11]. When it comes to scalability and ecosystem integration, AWS is on top. Azure is great at making things work for businesses, while GCP is all about AI service innovation. Different company demands are met by the distinct characteristics offered by each platform. Taking price, compatibility, and specialised services into account, this comparison helps businesses choose the best cloud provider [12].

Securing Personally Identifiable Information (PII) Data on AWS under PCI DSS Compliance. Adherence to PCI DSS requirements is essential for the security of PII on AWS. Sectors dealing with sensitive data must implement these procedures to prevent data breaches and unauthorised access. Data security is enhanced by AWS's compliance capabilities, which help with regulatory adherence [13]. Create a conversational AI chatbot with Amazon Lex V2 for advanced language processing. When building chatbots, developers often turn to Amazon Lex V2, which offers a strong language foundation for conversational AI capabilities [14]. Urban Land Cover Classification using CNNs on AWS. By analysing satellite images, Convolutional Neural Networks (CNNs) hosted on AWS categorise various terrains based on urban land cover. Supporting sustainable urban planning projects, CNNs on AWS enable large-scale picture analysis [15]. Cloud Software Development: A Comparison between AWS vs GCP. With an eye on efficiency and performance, this article compares AWS with Google Cloud Platform (GCP) in the context of cloud-based software development. GCP has an emphasis on AI integration, while AWS provides a more comprehensive environment [16]. Uniting Sustainable Development Goals (SDGs) with Technology in Academic Institutions. Focussing on educational tools offered by AWS, this article examines how higher education has integrated technology with the SDGs. Academic sustainability is advanced by technology-driven learning on AWS, which meets a variety of student demands [17]. Analysing Fluid Power using Digital Twin Technology. Using real-time simulation, digital twin technology provides valuable insights into the behaviour of systems in fluid power applications [18]. Analysing Autonomous Weapon

Systems and IHL Norms. The effectiveness of International Humanitarian Law (IHL) in controlling autonomous armed systems is investigated. This investigation relies heavily on AWS's assistance in conducting legal and ethical evaluations [19]. Cybersecurity at the Vanderbilt University Television News Archive in an ethical manner. Ethical use of AI is ensured by the Vanderbilt Television News Archive's use of responsible AI principles in managing their extensive databases of news information. The significance of ethical norms in AI-driven initiatives is shown by the implementation of responsible AI practices on AWS [20].

PROPOSED METHODOLOGY

To guarantee scalability and dependability for processing massive amounts of textual data, the suggested solution starts with a strong cloud architecture using AWS services. When it comes to NLP (natural language processing), AWS Comprehend is where it's at. Amazon Simple Storage Service (S3) and Amazon Elastic Compute Cloud (EC2) are both built into the infrastructure to facilitate the efficient execution of complicated processes and to store data securely. Figure 1 shows AWS Comprehend's poisonous comment detection architecture. The data ingestion module saves raw comment data in an input repository. Data is preprocessed for consistency and stored in the processed repository. Toxicity Detection uses AWS Comprehend's NLP to classify comments by toxicity and sentiment. Results are stored in a repository for convenient retrieval. This scalable design uses AWS Cloud's infrastructure to process and categorise large amounts of data quickly and reliably.

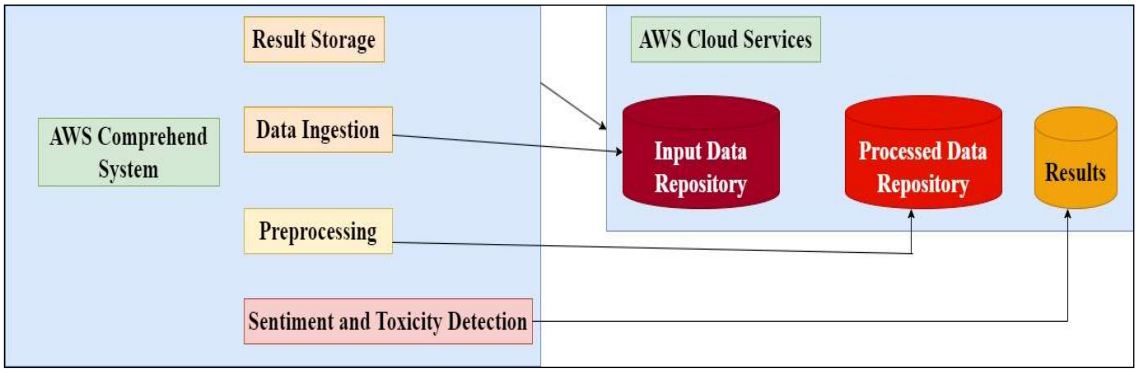

FIGURE 1. Block Diagram of AWS Comprehend Architecture for Toxic Comment Classification

To gather and prepare user feedback from several online sources, such as social media, forums, and review sites, the system sets up an automated pipeline. To make sure the text is suitable for analysis, the preprocessing step gets rid of noise such extra punctuation, URLs, and special characters. Figure 2 block diagram shows AWS Comprehend integrated with a data processing and storage system. An interface where users post comments feeds the Data Processing Layer (DPL). AWS Comprehend's API analyses these comments for toxicity and sentiment. The DPL stores the analysed findings in the Data Storage module for subsequent retrieval. Due to its integration with cloud-based NLP technologies, AWS Comprehend effectively classifies text.

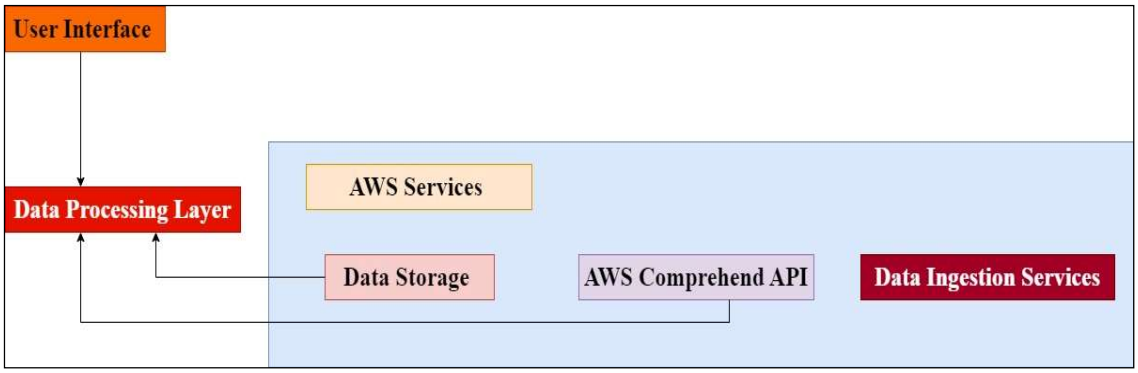

FIGURE 2. Block Diagram of Integration of AWS Comprehend for Toxicity Analysis

By default, AWS Comprehend is set up to detect and categorize comments according to their toxicity and sentiment. To train custom models, hazardous content-specific labelled datasets are used. These datasets specifically target hate speech, vulgarity, and foul language. The algorithm calculates a comment's toxins score based on these models. Figure 3 shows how AWS Comprehend classifies hazardous remarks from user inputs (comments). AWS Comprehend analyses user comments. This service classifies comments as hazardous or non-toxic using established criteria. The findings are saved in the cloud for quick access. Automating this workflow with AWS Comprehend's NLP processing allows scalable, real-time analysis of harmful comments. The pipeline structure can handle big comments, making it suited for high-traffic applications.

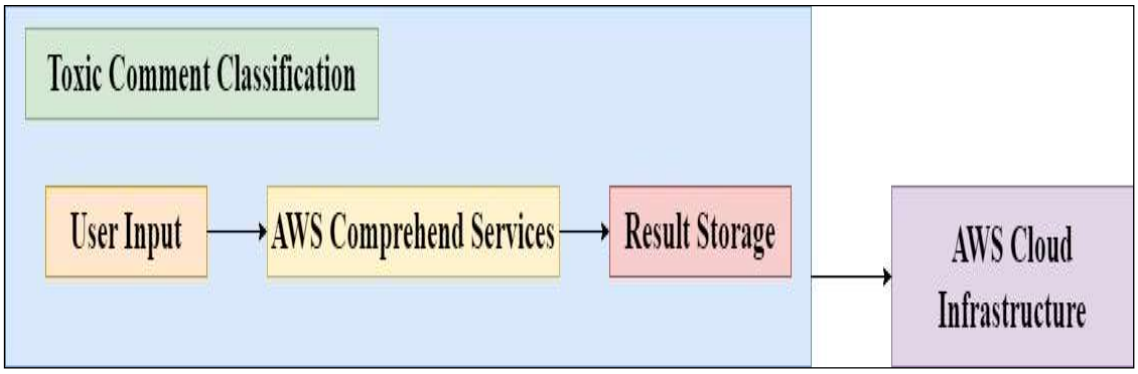

FIGURE 3. Block Diagram of Toxic Comment Classification Pipeline

Table 1 delineates the major characteristics of AWS. Understand techniques for detecting and categorizing harmful remarks in the cloud. Technology uses Natural Language Processing (NLP) to accurately identify poisonous, abusive, or hurtful language, facilitating real-time comment analysis. Multilingual support facilitates worldwide accessibility, making it appropriate for varied datasets.

TABLE 1. Key Features of AWS Comprehend for Toxic Comment Detection

Feature	Description	Benefit	Performance	Integration
Natural Language Processing (NLP)	Uses NLP to identify sentiment and toxicity levels	Detects toxic, abusive, or harmful language	Highly accurate sentiment analysis	Easily integrates with cloud environments
Real-Time Analysis	Analyzes comments in real time	Enables instant detection of toxic comments	Fast, scalable processing for live data	Ideal for real-time moderation systems
Multi-Language Support	Supports multiple languages for text analysis	Detects toxicity in various languages	Improved global reach for comment moderation	Integrates with multilingual datasets
Custom Classification Models	Allows custom model training for specific contexts	Tailors the system for unique use cases	Flexibility to improve detection in niche areas	Easy customization via AWS tools
Automated Sentiment Analysis	Automatically identifies sentiment in comments	Classifies comments based on positivity/negativity	Enhanced efficiency in large-scale analysis	Seamless integration with AWS cloud storage

Harmful remarks are sorted into many groups using a hierarchical system, including cyberbullying, hate speech, and vulgarity. Primary categories are defined initially in the framework's multi-layered design, which is then followed by subcategories. Figure 4 shows AWS Comprehend's harmful comment analysis methodology. User comments are pre-processed for data consistency and accuracy. Only legitimate comments are analysed using AWS Comprehend API for toxicity and sentiment. To protect data, invalid comments are rejected. The analytical findings are kept in a database for future reference or report preparation. A feedback mechanism returns the categorisation to the user. This method simplifies harmful comment detection for precise and fast processing.

Table 2 delineates the advantages of using AWS Comprehend for identifying toxic comments. Real-time moderation facilitates the immediate filtration of damaging remarks, hence enhancing the user experience on digital platforms. AWS comprehends high accuracy guarantees precise identification of harmful material while reducing mistakes.

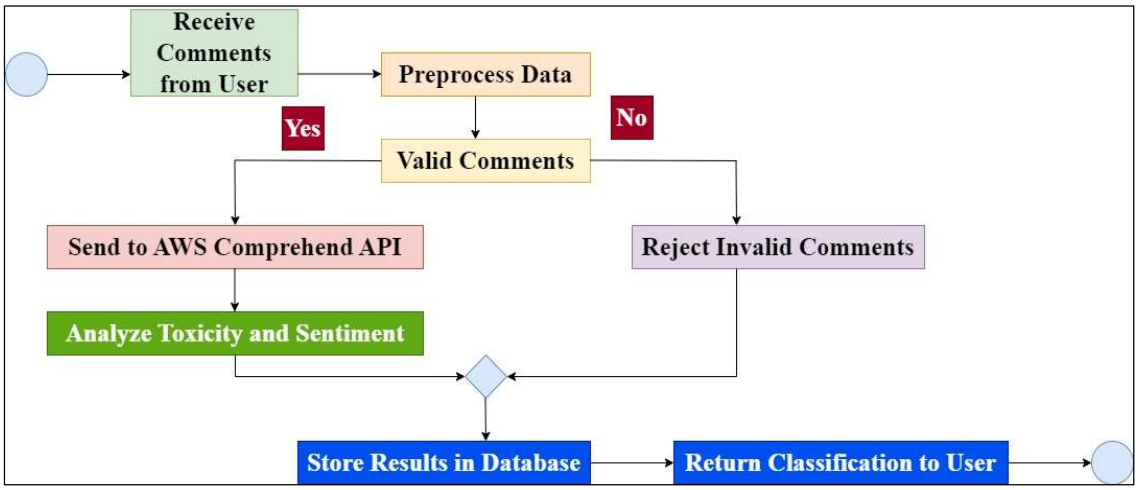

FIGURE 4. Workflow Diagram of Toxic Comment Analysis Workflow

TABLE II. Benefits Of Using AWS Comprehend for Toxic Comment Detection

Benefit	**Description**	**Impact on Toxic Comment Detection**	**Performance**	**Cost Efficiency**
Real-Time Moderation	Enables instant moderation of user comments	Quickly identifies and filters toxic content	Fast processing for large volumes of comments	Reduces manual moderation costs
High Accuracy	Uses advanced NLP algorithms for accurate detection	Reduces false positives and false negatives	High precision in detecting toxic language	Increases efficiency with fewer errors
Scalability	Leverages AWS cloud infrastructure to scale as needed	Can handle increasing comment volumes as platforms grow	Efficient scaling without performance loss	Pay-as-you-go model reduces upfront costs
Multi-Language Support	Detects toxicity across different languages	Expands usability to global platforms	High detection accuracy across languages	Reduces need for multiple localized solutions
Customization	Allows tailoring of models for specific platforms	Fine-tunes detection criteria for unique contexts	Flexible and adaptable to different types of data	Low-cost customization using AWS tools

The technology includes real-time moderating measures to make the site safer. A flag is raised, and the remark is submitted to the moderation queues for evaluation when it is determined to be poisonous. Notification systems like Amazon SNS (Simple Notification Service) may be integrated to generate alerts for information that poses a high risk and requires rapid attention. Figure 5 shows an AWS Comprehend harmful comment detection ecosystem overview.

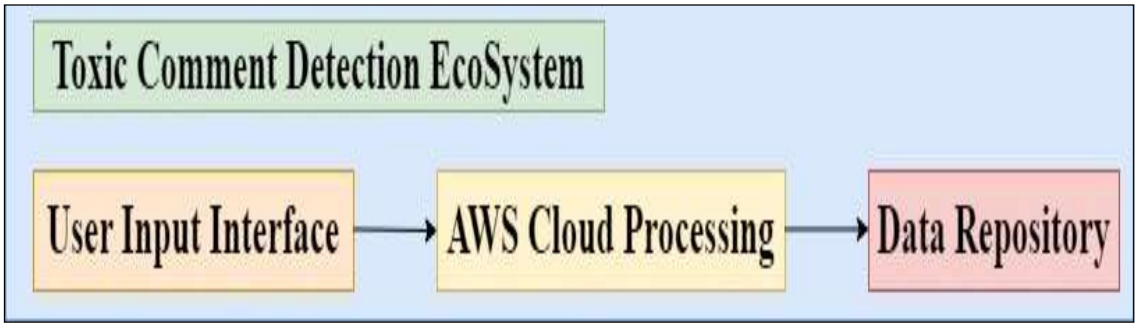

FIGURE 5. Overview Diagram of Toxic Comment Detection System Overview

User Input Interface, AWS Cloud Processing, and Data Repository comprise the system. The input interface sends comments to AWS comprehends NLP service for poison detection and sentiment categorisation. A data repository stores analytical findings for future access or reporting. This simplified design allows AWS Cloud to

scale to handle large amounts of user data. Moderators and users alike may contribute to feedback loops that continuously enhance categorization accuracy. The bespoke AWS Comprehend models are continuously improved by storing and analyzing this input. This allows them to react to changing language patterns and toxicity trends. The use of semi-supervised learning methods improves model generalizability by making use of both labelled and unlabeled data. The training datasets are regularly updated to contain fresh instances of poisonous language, ensuring that the classification accuracy remains high over time. Trends in toxicity and the efficacy of moderation may be uncovered by a thorough reporting module. Metrics like the number of poisonous comments found, the breakdown of toxicity categories, and the efficiency of categorization algorithms are shown on dashboards. Heatmaps and word clouds are visualization tools that may help you see patterns and identify areas that need more attention. Stakeholders may find out how harmful material is on their platforms and how well their moderation efforts are working with customizable reports. This component makes use of AWS QuickSight to display data in an interactive and scalable manner.

RESULTS AND DISCUSSIONS

The system can reach a wider audience since it can connect to other platforms. These platforms include CMSs' and tools for moderating social media. Platforms may integrate the categorization capabilities of AWS Comprehend into their processes with the use of APIs, which ease data sharing. While specialized APIs cater to specific needs, pre-built connectors make connection with common systems easier. The compatibility of these systems guarantees that toxicity detection standards will be widely used and applied consistently across many platforms. Figure 6 shows five comment analysis ratings. The five metrics—toxicity, sentiment, neutrality, aggressiveness, and positivity—rate each remark. Toxicity Score measures damaging words, whereas Sentiment Score measures tone. Neutrality and Aggressiveness Scores reveal the comment's sentiment.

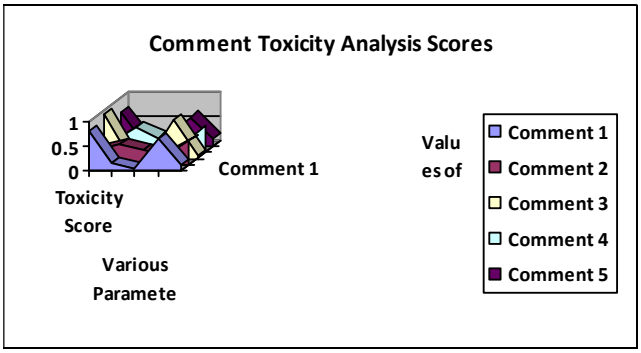

FIGURE 6. Comment on Toxicity Analysis Scores

Table 3 examines the efficacy of AWS Comprehend in detecting poisonous remarks. The cloud advantage substantially enhances detection speed, allowing real-time identification of detrimental information. The system's reliability is augmented by enhanced accuracy using powerful NLP algorithms, surpassing previous detection approaches. Scalability on the AWS cloud guarantees that the system can accommodate substantial datasets and elevated traffic volumes.

TABLE II. Benefits Of Using AWS Comprehend for Toxic Comment Detection

Metric	Standard Performance	Cloud Advantage	Outcome	Optimization
Detection Speed	Moderate processing speed	High-speed processing with cloud resources	Faster identification of toxic comments	Optimized resource allocation in AWS
Detection Accuracy	Basic accuracy of toxic comment detection	Enhanced accuracy with AWS NLP algorithms	More reliable identification of toxic content	Improved over time through machine learning
Scalability	Limited by on-premises infrastructure	Virtually unlimited scaling with AWS cloud computing	Handles larger datasets and higher traffic volumes	Efficient scaling as demand increases
Language Diversity	Limited to specific languages	Supports detection in multiple languages	Broader detection capabilities for global platforms	Expanding multilingual support
Resource Management	Limited by hardware capacity	Dynamic resource allocation via AWS cloud	Ensures efficient handling of large comment volumes	Cost-effective and automated scaling

The system includes procedures to remove personally identifiable information from user comments before processing them, since it values user privacy. Encryption and access restrictions for stored data guarantee compliance with data protection standards like GDPR and CCPA. Figure 7 provides essential indicators for assessing model performance across five training epochs. Training Loss and Validation Loss show model training and testing error rates. Both Training Loss and Validation Loss fall consistently from 0.45 to 0.28 and 0.50 to 0.35, respectively, showing that the model is learning without overfitting. Training and Validation Accuracy increase with each epoch, reaching 92% and 90%, respectively, by the fifth epoch, demonstrating the model's generalization. The F1 Score, which balances precision and recall, slowly rises to 0.88, indicating greater hazardous comment detection with fewer false positives and negatives.

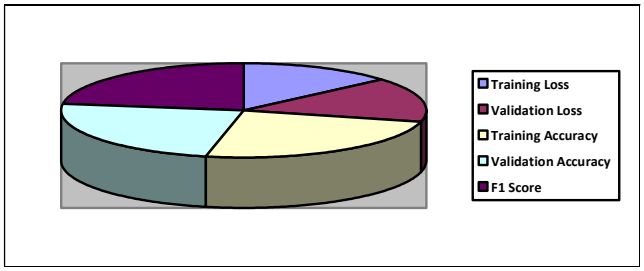

FIGURE 7. Model Accuracy Metrics

System scalability and flexibility are built-in to ensure the system can handle future demands. With AWS Auto Scaling handling resource allocation dynamically, the cloud infrastructure is prepared to manage growing data volumes as platforms expand. Figure 8 shows the relevance of toxicity classification model features. The Importance Score of each feature—Toxicity Level, Aggressiveness, and Sentiment—reflects its model accuracy contribution. Toxicity (0.85) is the most important attribute, followed by Aggressiveness and Sentiment. Toxicity Level is the model's most weighted attribute.

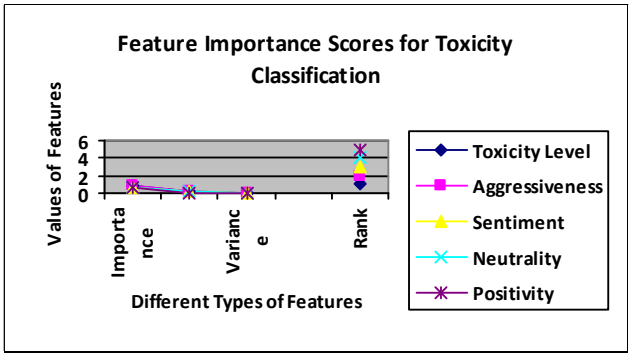

FIGURE 8. Feature Importance Scores for Toxicity Classification

The suggested technique considers the varied language contexts of worldwide platforms by placing an emphasis on the capacity to identify harmful remarks in more than one language. By using the built-in multilingual NLP support of AWS Comprehend, the system can expand its classification skills to detect toxicity in a variety of languages, including dialects and regional variants. Amazon Translate helps eliminate language barriers by translating non-supported languages into English as needed, allowing for smooth analysis. By allowing for thorough monitoring and moderation of harmful information regardless of the user's original language, this feature promotes inclusion in online spaces.

CONCLUSION

AWS Comprehend technology improves automatic content moderation by identifying and categorising hazardous remarks. However, contextual misunderstanding, changing linguistic patterns, and subtle toxicity might impact categorisation accuracy. Sarcasm, unconscious bias, and cultural differences might cause sentiment analysis

tools to overlook hazardous material or provide false positives. To enable real-time processing efficiency, AWS Comprehend must be integrated with several digital platforms via seamless API setups and computational resource management. Dependence on pre-trained models, domain-specific accuracy loss, and difficulty adjusting to new harmful language patterns are limitations. Constant model refining and customised training datasets improve classification accuracy. Cloud adoption requires cost management to optimise scalability and processing speeds. Deep learning, adaptive NLP algorithms for real-time toxicity identification, and language support for global content control may be added. AI-driven contextual analysis improves categorisation models and reduces algorithmic bias. Effective toxic comment categorisation improves automatic content moderation across social media, forums, and online communities, making digital interactions safer and reducing destructive talk.

REFERENCES

[1]. R. M. Al-Sayyed, W. A. Hijawi, A. M. Bashiti, I. AlJarah, N. Obeid, and O. Y. Adwan, 2019, "An investigation of Microsoft Azure and Amazon Web Services from users' perspectives," *International Journal of Emerging Technologies in Learning*, **14(10)**, pp. 217-241.

[2]. P. Dutta, and P. Dutta, 2019, "Comparative study of cloud services offered by Amazon, Microsoft and Google," *International Journal of Trend in Scientific Research and Development*, **3(3)**, pp. 981-985.

[3]. J. Bai, I. Jhaney, and J. Wells, 2019, "Developing a reproducible microbiome data analysis pipeline using the Amazon Web Services cloud for a cancer research group: Proof-of-concept study," *JMIR Medical Informatics*, **7(4)**, pp. 1-9.

[4]. K. Mahesh, I. D. Sharma, and M. Laxmaiah, 2020, "Amazon cloud services based on service level agreement and quality of service," *International Journal of Advanced Research in Engineering and Technology*, **11(11)**, pp. 1604-1613.

[5]. U. Ghosh, and S. Dolui, 2019, "Cloud computing: Its security and introduction to Amazon Web Services," *International Conference on Innovation in Modern Science and Technology*, pp. 942-951.

[6]. K. Mahesh, M. Laxmaiah, and Y. K. Sharma, 2019. "A comparative study on Google App Engine, Amazon Web Services, and Microsoft Windows Azure," *International Journal of Computer Engineering and Technology*, **10(1)**, pp. 54-60.

[7]. P. Pierleoni, R. Concetti, A. Belli, and L. Palma, 2020, "Amazon, Google, and Microsoft solutions for IoT: Architectures and a performance comparison," *IEEE Access*, **8**, pp. 5455-5470.

[8]. J. Puttonen, A. Lobov, M. A. Soto, and J. L. Lastra, 2019, "Cloud computing as a facilitator for web service composition in factory automation," *Journal of Intelligent Manufacturing*, **30(2)**, pp. 687-700.

[9]. B. Gupta, P. Mittal, and T. Mufti, 2021, "A review on Amazon Web Service (AWS), Microsoft Azure & Google Cloud Platform (GCP) services," *Proceedings of the 2nd International Conference on ICT for Digital, Smart, and Sustainable Development*, pp. 1-9.

[10]. P. Kaushik, A. M. Rao, D. P. Singh, S. Vashisht, and S. Gupta, 2021, "Cloud computing and comparison based on service and performance between Amazon AWS, Microsoft Azure, and Google Cloud," *International Conference on Technological Advancements and Innovations*, pp. 268-273.

[11]. T. Singh, 2021, "The effect of Amazon Web Services (AWS) on cloud computing," *International Journal of Research and Technology*, **10(11)**, pp. 2278-0181.

[12]. M. Tajammul, R. Parveen, I. A. Tayubi, and S. Asif, 2021, "Cloud storage in the context of Amazon Web Services," *International Journal of All Research Education and Scientific Methods*, **10(1)**, pp. 442-446.

[13]. S. Maurya, G. Lakhera, A. K. Srivastava, and M. Kumar, 2021, "Cost analysis of Amazon Web Services–From an eye of architect and developer," *Materials Today: Proceedings*, **46**, pp. 10757-10760.

[14]. A. Choudhary, P. K. Verma, and P. Rai, 2021, "A walkthrough of Amazon Elastic Compute Cloud (Amazon EC2): A review," *International Journal for Research in Applied Science and Engineering Technology*, **9(11)**, pp. 93-97.

[15]. N. Hota, and B. K. Pattanayak, 2021, "Cloud computing load balancing using Amazon Web Service technology," *Progress in Advanced Computing and Intelligent Engineering: Proceedings of ICACIE*, pp. 661-669.

[16]. A. Toshniwal, K. S. Rathore, A. Dubey, P. Dhasal, and R. Maheshwari, 2020, "Media streaming in cloud with special reference to Amazon Web Services: A comprehensive review," *4th International Conference on Intelligent Computing and Control Systems*, pp. 368-372.

[17]. A. S. Muhammed, and D. Ucuz, 2020, "Comparison of the IoT platform vendors, Microsoft Azure, Amazon Web Services, and Google Cloud, from users' perspectives," *8th International Symposium on Digital Forensics and Security*, pp. 1-4.

[18]. S. P. Ahuja, E. Czarnecki, and S. Willison, 2020, "Multi-factor performance comparison of Amazon Web Services Elastic Compute Cluster and Google Cloud Platform Compute Engine," *International Journal of Cloud Applications and Computing*, **103**, pp. 1-6.

[19]. M. Saraswat, and R. C. Tripathi, 2020, "Cloud computing: Comparison and analysis of cloud service providers—AWS, Microsoft, and Google," *9th International Conference System Modeling and Advancement in Research Trends*, pp. 281-285.

[20]. K. R. Ferreira, G. R. Queiroz, G. Camara, R. C. Souza, L. Vinhas, R. F. Marujo, R. E. Simoes, C. A. Noronha, R. W. Costa, J. S. Arcanjo, and V. C. Gomes, 2020, "Using remote sensing images and cloud services on AWS to improve land use and cover monitoring," *IEEE Latin American GRSS & ISPRS Remote Sensing Conference*, pp. 558-562.