IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

A MULTI-MODAL SCAM DETECTION SYSTEM FOR SOCIAL MEDIA ADVERTISEMENTS USING EXPLAINABLE AI

[1]R Sheik Alavudeen, [2]Shoban S, [3]Ms. R. Iyswarya

[1]Student, [2]Faculty

[1]Department of AI & DS,

[1]Sri Venkateswara College of Engineering, Chennai, India

Doi: https://doi.org/10.56975/ijcrt.v14i7.309252

Abstract: The rapid expansion of social media platforms has transformed digital marketing, providing businesses with targeted and personalized advertising mechanisms. However, this growth has simultaneously led to a significant increase in deceptive and fraudulent advertisements designed to exploit user trust. Scammers utilize sophisticated combinations of attractive imagery, persuasive promotional text, manipulated reviews, and misleading URLs, making detection highly challenging for conventional systems. Existing approaches predominantly rely on single-modal analysis—evaluating only textual content or URL structures in isolation—which frequently fails to capture the complex, multi-faceted nature of modern online fraud. Furthermore, many contemporary models operate as opaque algorithms, generating predictions without providing interpretable reasoning, thereby limiting user comprehension and trust. To address these limitations, this paper proposes a Multi-Modal Scam Detection System that systematically evaluates social media advertisements across four distinct data modalities: linguistic text patterns, structural URL characteristics, visual image features, and user review sentiment. Each modality is processed through dedicated analysis engines that extract critical indicators of deception before being synthesized by an adaptive fusion layer to generate a unified risk assessment. Crucially, the architecture incorporates an advanced Explainable AI (XAI) framework powered by a Large Language Model (LLM). By feeding extracted algorithmic indicators into a generative model, the system dynamically synthesizes natural, human-readable explanations for its decisions. By combining multi-modal capability with LLM-driven transparent decision-making, the proposed system provides a comprehensive mechanism for identifying fraudulent promotional content and fostering informed user decision-making during online interactions.

I. INTRODUCTION

Social media networks serve as primary conduits for e-commerce and digital advertising, utilizing complex recommendation algorithms to present users with highly personalized content. While this personalization enhances user engagement, it also inadvertently creates an environment where malicious actors can deploy highly-convincing scam advertisements. Fraudulent campaigns increasingly mimic legitimate commercial entities by employing professional-grade visual layouts, deploying urgent promotional language, and artificially inflating product credibility through fabricated user reviews. Because these deceptive advertisements are contextually integrated into users' daily browsing feeds, individuals are significantly more susceptible to financial and informational exploitation.

The challenge of detecting these sophisticated scams highlights critical limitations in existing security frameworks. Traditional detection mechanisms typically perform single-factor analysis. For instance, text-based classifiers might scan for promotional keywords, while URL scanners might check domain blacklists. Scammers circumvent these isolated defenses by compensating in other areas. A system that analyzes only one or two factors lacks the contextual awareness required to identify cross-modal deceptive patterns. Additionally, as the industry pivots to ward complex machine learning methodologies, a secondary problem arises: a lack of transparency. When black-box systems classify an advertisement as fraudulent, they fail to articulate the underlying rationale, leaving users uninformed.

This research addresses these deficiencies by proposing a multi-modal, transparent detection architecture. The proposed system integrates four specialized analysis modules—evaluating textual linguistics, URL structures, visual representations, and review sentiments—to generate independent risk metrics. These metrics are subsequently aggregated to compute a comprehensive scam probability. To ensure user trust and facilitate active awareness, the system integrates a Generative LLM to dynamically

construct conversational, context-aware explanations based on the system's internal findings. The primary contributions of this work are the design of a multi-modal analysis pipeline specifically calibrated for social media advertisements, the implementation of computationally practical analytical models, and the integration of LLM-driven explainability.

II. LITERATURE REVIEW

Research into digital fraud and scam detection has evolved from isolated, single-modality rule checks to complex, multimodal deep learning frameworks, with Explainable AI (XAI) and Large Language Models (LLMs) marking the latest frontier. This section reviews the progression of these technologies, examining foundational deception detection techniques, the role of multimodal architectures, and the transformative impact of transparent, LLM-enhanced explainability.

A. Foundational AI in Fraud and Deception Detection

The application of artificial intelligence to combat digital deception is a well-established field. A foundational exploration by Austin et al. (2024) established the role of machine learning in detecting social media-based fraud, particularly highlighting schemes that target vulnerable entities like small businesses [3]. Early approaches in this domain often isolated specific data types to identify threats. For instance, Fu et al. (2006) pioneered visual similarity assessments using Earth Mover's Distance (EMD) to detect phishing web pages based purely on layout imitation [5]. Similarly, addressing the textual domain, Hutto and Gilbert (2014) introduced VADER, a parsimonious rule-based model that proved highly effective for rapid sentiment analysis of informal social media text without requiring massive training datasets [7]. Building upon textual analysis, Mughal et al. (2025) demonstrated that fake review detection requires moving beyond simple text analysis to include user behavioral features, proving that deception often lies in the structural mismatch between language and numerical ratings [6]. However, the primary limitation of these foundational techniques lies in their isolated, single-factor approach. Scammers quickly adapted by manipulating one modality (e.g., using benign text) while executing the fraud through another (e.g., a malicious URL or stolen image), leaving a significant gap in robust detection.

B. Multimodal and Multi-Domain Architectures

To better model the complex, multi-faceted nature of modern online scams, researchers shifted toward multi modal architectures. These models simultaneously process various data streams—such as text, images, and user comments—allowing for the identification of cross-modal deceptive patterns. An important contribution in this area is the systematic review by Nasser et al. (2025), which documented the superiority of deep learning–based multi modal approaches over traditional models in detecting fake news and deceptive content on social media [2]. Further extending this, Guo et al. (2025) proposed a framework integrating multi-domain and multimodal features, emphasizing that deceptive content often spans diverse semantic domains and requires domain-aware fusion to capture underlying structures [4]. The success of these visual-textual integrations relies heavily on robust feature extraction methods, such as lightweight deep Convolutional Neural Networks (CNNs) like MobileNetV2, which provide an efficient architectural backbone for deep visual analysis in resource-constrained environments [8]. While powerful, the utility of such multimodal deep learning models is often offset by their immense computational overhead and their opaque nature, acting as "black boxes" that output decisions without revealing the underlying rationale.

C. Explainable AI and User-Centric Transparency

The advent of Explainable AI (XAI) has introduced a paradigm shift, addressing the critical need for trust and transparency in automated decision-making. Foundational work by Lundberg and Lee (2017) established SHAP (Shapley Additive exPlanations), providing a unified game-theoretic approach to feature attribution that reveals which specific inputs drive a model's prediction [9]. However, applying these mathematical techniques directly to complex fraud detection systems presents unique difficulties. The work by Zafar and Wu (2026) critically examined the methodological challenges of XAI in fraud detection, highlighting that traditional post-hoc explanations applied to deep black-box models often fail to provide stable, meaningful interpretations in real-world scenarios, particularly for non-technical users [1]. While the literature demonstrates the immense potential of multimodal detection and mathematical XAI, current applications are typically designed as heavy, backend enterprise solutions that output complex statistical weights, confusing average users. A distinct gap remains for a user-centric system that integrates these advanced detection capabilities into a lightweight, interpretable toolkit. Our proposed system addresses this by combining practical multimodal heuristic fusion with LLM-powered conversational XAI. This provides a comprehensive solution that not only detects multi-faceted scam advertisements but empowers individual users with natural, data-driven explanations derived from the specific deceptive indicators in their input.

III. METHODOLOGY

The proposed multi-modal scam detection system is engineered to sequentially ingest, standardize, evaluate,and synthesize diverse data inputs. The methodology focuses on computational efficiency and interpretability. The overall architecture is shown in Fig. 1.

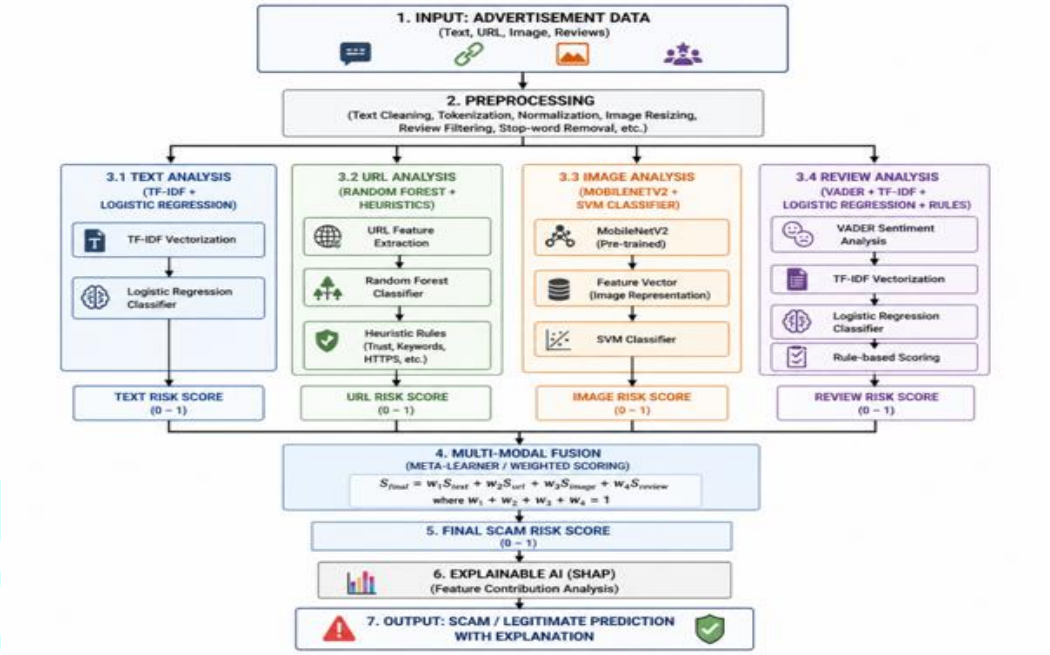

Fig. 1. Proposed Multi-Modal Scam Detection System Architecture.

A. Data Acquisition and Preprocessing

Any machine learning system's efficiency completely depends on the quality and variety of its training data. The heterogeneous nature of scam content becomes the crucial challenge in social media scam advertisement detection. Scam advertisements leverage the availability of multiple components such as text, hyperlinks, images, and user-generated reviews to deceive users. There is no single dataset publicly available that includes all four of these modalities in a scam advertisement context. To enable an integrated multi-modal analysis, it is necessary to curate a multi-source dataset that represents each modality separately. The raw data collected exist in diverse formats and structures, making them incompatible for direct use in machine learning models. Therefore, each data type requires modality-specific preprocessing. For textual modalities (Text and Reviews), the system utilizes a hybrid feature extraction approach, combining TF-IDF vectorization to capture semantic context with handcrafted linguistic features (such as urgency ratios and sentiment polarity) to capture deceptive patterns.

Data Sources and Collection

Four independent datasets were collected for each modality of the system:

Text Modality: The SMS Spam Collection dataset from the UCI Machine Learning Repository was used. The dataset consists of 5,574 messages labeled as either ham (legitimate) or spam, with 4,827 legitimate messages and 747spam messages. During preprocessing, raw strings are cleaned by removing punctuation and stop-words, and converted to lowercase. Additionally, specific entities like URLs, email addresses, and numerical values are replaced with unique tokens (url, email, num) to prevent overfitting and improve the model's ability to recognize general-linguistic structures common in scams.

URL Modality: The Malicious URLs dataset from Kaggle was used. The dataset contains 651,191 URLs across four categories: benign, phishing, defacement, and malware. Following empirical testing, only benign (428,103) and phishing (94,111) URLs were included, resulting in a specialized dataset of 522,214 entries. Defacement URLs were excluded because their structural similarity to legitimate hacked websites frequently caused false positives during commercial testing. Malware URLs were excluded as they typically involve direct file downloads rather than the deceptive promotional tactics found in social media advertisements. To further reduce false positives, the Tranco Top 1 million domain list is integrated as a real-time trust verification layer.

Image Modality: The phishing website screenshot dataset from the Zenodo research repository was used. This dataset contains over 10,000 automated browser screenshots. To ensure maximum realism, an automated extraction script was developed to isolate the index_js_on.jpg variant (screenshots taken with JavaScript enabled). A balanced corpus of 500 phishing and 500 legitimate screenshots was curated. These images are standardized by resizing to 224 x 224 pixels and undergoing tensor normalization to ensure compatibility with the MobileNetV2 deep learning architecture.

Review Modality: The Fake Reviews dataset from Kaggle was used, containing 40,432 product reviews with a perfect class balance (20,216 computer-generated fake reviews and 20,216 original reviews). Each entry includes the review text and a numerical rating. Preprocessing for this modality integrates the VADER (Valence Aware Dictionary and sEntiment Reasoner) engine to extract emotional polarity scores. This allows the system to detect "sentiment rating mismatches"—a key indicator of review manipulation where highly positive text is paired with a low-numerical rating, or vice versa.

Preprocessing Pipeline

Text Preprocessing — Raw text was processed in five steps. In the first step, all characters were converted to lowercase to ensure case-insensitive token matching. In the second step, URL strings embedded within text messages were replaced with the placeholder token “url” and numeric sequences were replaced with “num” using regular-expression substitution, preserving their presence while preventing the model from overfitting. In the third step, email addresses within messages were replaced with the token “email”. In the fourth step, all punctuation characters were removed using Python’s string translation mechanism. In the fifth step, consecutive whitespace was reduced to a single space. In addition to surface cleaning, 14 handcrafted discriminative features were extracted from each text instance. These features included character count, word count, unique word ratio, uppercase ratio, exclamation mark count, question mark count, URL count, email count, scam keyword count, scam keyword ratio, urgency phrase count, urgency presence flag, number count, and average word length. The scam keyword lexicon consists of 43 terms associated with fraudulent communication, such as financial incentives, account threats, and reward claims. The urgency phrase lexicon consists of 15 multi-word phrases representing psychological pressure tactics like “act now”, “expires today”, and “limited time only”.

URL Preprocessing — URLs were parsed using the tldextract library, which breaks down URLs into subdomain, registered domain, and public suffix components using the Public Suffix List. This library correctly handles multi-part TLDs such as .co.uk and .gov.in that simpler parsers misidentify. Before parsing the URL, query strings and URL fragments were removed to prevent query parameter values (such as redirect = paypal.com)from being misinterpreted as part of the domain structure. Eighteen structural features were extracted from each URL, including URL length, domain length, path length, dot count, slash count, hyphen count in domain, HTTPS presence flag, IP address usage flag, subdomain depth, suspicious TLD flag, suspicious keyword count, at-symbol count, double slash in path flag, hexadecimal character count, digit ratio in domain, trusted domain flag, query string length, and URL shortener usage flag. The suspicious TLD list was derived from Anti-Phishing Working Group reports to identify free and low-cost top-level domains disproportionately used in phishing. Domain trust verification was performed against the Tranco top-1,000,000 domain list, with the top 500,000 used as the primary trust threshold, complemented by a hardcoded whitelist for .gov and .edu TLDs.

Image Preprocessing — All images were processed through a standardized pipeline before beingpassedtotheMobileNetV2 feature extractor. First, all images were converted into RGB color space to ensure three-channel consistency. Images were then resized to 224×224 pixels using bilinear interpolation, the input resolutionrequiredbyMobileNetV2. To align the input distribution with the pretrained model, pixel values were normalized using the ImageNet dataset mean [0.485, 0.456, 0.406] and standard deviation [0.229, 0.224, 0.225]. In addition to deep feature extraction, pixel-level visual heuristics are measured, including red channel dominance, brightness variance, and contrast levels, which serve as supplementary indicators of aggressive or deceptive visual design.

Review Preprocessing — Review text was cleaned using the same pipeline as the text module, including lowercase conversion, URL removal, punctuation stripping, and whitespace normalization. VADER sentiment analysis was applied to each raw review to extract four sentiment scores. Separately, three domain-specific lexicons were used: a fake positive phrase list containing 18 exaggerated endorsement patterns, a fake negative phrase list containing 13 attack review patterns, and a generic phrase list containing 12 filler phrases. In addition to behavioral signals such as short review flags (ten words or less), very long review flags (more than 150 words), extreme sentiment flags (compound score greater than 0.85), and sentiment-rating mismatch flags that identify logical inconsistencies, twenty handcrafted features were extracted for each review.

Design Justification — There were two practical reasons behind collecting four independent datasets rather than constructing a single scam advertisement corpus. First, there is no publicly available labeled multi-modal scam advertisement dataset of adequate scale, a constraint recognized in the broader scam detection literature. Second, the modular dataset approach allows each analytical component to be independently evaluated and improved as better domain-specific datasets become available. The deliberate exclusion of defacement and malware URL categories, the selection of index_js_on.jpg screenshots, and the adoption of the Tranco domain list over static whitelists each represent specific design decisions made to improve real-world performance beyond what a naive dataset would achieve. The detailed processing pipeline for individual modalities is shown in Fig. 2.

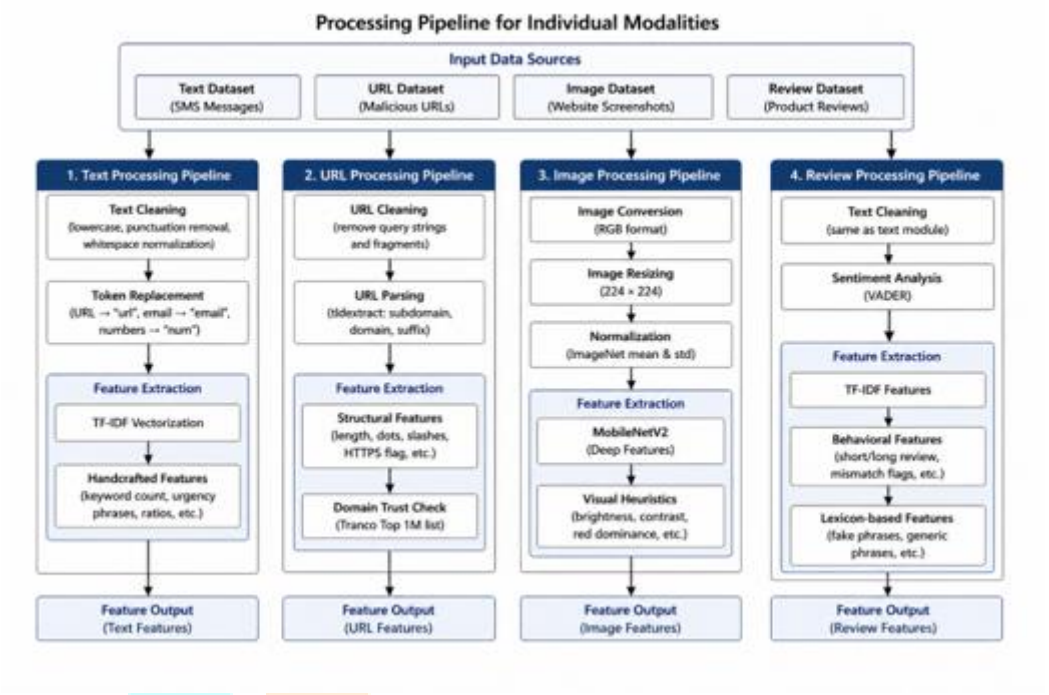

Fig. 2. Processing Pipeline for Individual Modalities

B. Linguistic & Contextual Text

Scam advertisements are basically linguistic artifacts. They use specific patterns of language to manipulate people into acting without thinking. These patterns include fake urgency ("expires today", "act now"), false claims of rewards ("you have been selected", "claim your free prize"), financial incentives ("100% guaranteed returns", "earn $5000 weekly"), and threats to accounts ("your account will be suspended"). All of these things make a unique textual fingerprint that differentiates scam content from legitimate commercial communication. The text analysis module solves the problem of automatically finding these patterns in any advertisement text without human review. It works well even if the text is poorly formatted, has mixed capitalization, or has URLs in it.

Data and Features

The SMS spam collection dataset had 5,572 labeled text examples. The class distribution -- 4,825 legitimate (86.6%) and 747 spam (13.4%) -- reflects the natural imbalance found in real-world scam detection scenarios. This imbalance was fixed during model training by using balanced class weighting instead of oversampling. This kept the natural data distribution while compensating for the minority class in the optimization objective.

Feature representation was constructed from two complementary sources. The first source was TF-IDF (Term Frequency-Inverse Document Frequency) vectorization applied to the cleaned text, which made 3,000 word and bigram features. TF-IDF was configured with sublinear term frequency scaling (replacing raw count f with1+log(f)to reduce the effect of high-frequency terms, bigram inclusion (ngram_range = (1,2)) to capture phrase-level patterns like "free-iPhone" and "click now" that unigrams alone can't show and a minimum document frequency threshold of 2 to eliminate hapax legomena -- words that only appear in one document and ad noise instead of signal.

The second source was a set of 14 handcrafted features made from domain knowledge of scam communication patterns. These features were designed to capture signals that statistical language modeling may miss, like the uppercase ratio (a strong indicator of aggressive solicitation), exclamation mark density (emotional manipulation), and the explicit presence of scam keywords and urgency phrases drawn from scam pattern analysis. By using horizontal sparse matrix concatenation, the two sets of features were combined into one feature space with 3,014 dimensions per instance.

Methodology

Three classification algorithms were trained and evaluated: Logistic Regression, Multinomial Naïve Bayes, and Random Forest. Logistic Regression was trained with L2 regularization at strength C=1.0, a maximum of 1,000 optimization iterations to ensure it converged on the high-dimensional combined feature space, and balanced class weighting. The model learns a weight vector w of dimension 3,014 and a bias term b. The probability of the scan class for a feature vector x is computed as:

$$P(\text{scam}|x) = 1 / (1 + \exp(- (w \cdot x + b)))$$

This sigmoid function maps the linear combination of features to a probability between 0 and 1. The weight associated with each feature directly measures how much it affects the scam probability -- highly weighted features are the most discriminative. This interpretability property is particularly valuable for SHAP-based explanation generation in the final stage of the pipeline. Multinomial Naïve Bayes was trained exclusively on the TF-IDF features, not the combined matrix, because the algorithm

requires non-negative inputs. The standardized combined feature matrix contains negative values after scaling. Laplace smoothing at alpha=0.1 was applied to prevent zero probability assignment for words that weren't in a training class.

Random Forest was trained with 100 decision trees, balanced class weighting, and parallel execution across all available CPU cores. Each tree learned from a bootstrap sample taken with replacement from the training set, and each node split considered a random subset of sqrt (3014) ≈ 55 features.

The dataset was split into 80% training (4,457 instances) and 20% testing (1,115 instances) using stratified sampling to preserve class distribution. All features were standardized using Standard Scaler (with mean-centering disabled for sparse matrix compatibility) fit exclusively on training data and applied to test data.

Results and Design Justification

Logistic Regression achieved F1=0.9589 and ROC-AUC=0.9887, the highest F1 among the three models. The correlated scam keyword features break the assumption of feature independence, which makes it better than Naïve Bayes. Its comparable performance to Random Forest with lower computational cost makes it the preferred deployment choice.

The combination of TF-IDF and handcrafted features work better than either one alone because they capture complementary signal types. TF-IDF captures general vocabulary patterns learned from training data while handcrafted features encode expert knowledge about scam-specific linguistic phenomena that may not be well represented in the training corpus due to the SMS-to-advertisement domain shift.

In order to address a known limitation of training on the SMS Spam Collection, a post-hoc score calibration was implemented. The model assigns elevated risk scores to typical promotional language due to the lack of actual commercial advertisement text. In order to prevent false positives on typical e-commerce text, a legitimate advertisement phrase whitelist was implemented that reduces the model score when legitimate commercial phrases (such as free delivery, free trial, no credit card necessary) are detected in the absence of genuine scam signals. Mathematically, the final text risk score is computed as:

$$S_{text} = P_{model} + A_{context} \quad (1)$$

where P_{model} represents the classifier output and $A_{context}$ denotes rule-based adjustments.

Output

The text analysis module returns a risk score between 0.0 and 1.0 representing the probability of scam, a confidence band (low, medium, high), a binary verdict, and a list of up to three triggered scam indicators for human-readable explanation. This score feeds directly into the fusion meta-learner alongside scores from other three modules.

C. URL Risk & Domain Trust Engine

Uniform Resource Locators (URLs) serve as the primary mechanism through which scam advertisements direct victims to malicious destinations. Unlike legitimate commercial URLs which follow predictable structural conventions, registered domains associated with known brands, HTTPS encryption, clean path hierarchies, phishing and scam URLs exhibit distinctive structural anomalies that betray their fraudulent nature even before the destination page is visited. These anomalies include the use of free or low-cost top-level domains to avoid registration costs, deliberate impersonation of trusted brand names with domain strings, absence of valid SSL certificates, excessive URL length designed to hide the real destination, and obfuscation techniques such as hexadecimal character encoding and IP address substitution. In order to assign a risk score to any URL without requiring the system to visit or display the destination page, the URL analysis module takes advantage of these structural signals. This is an essential safety and efficiency feature for a real-time detection system.

Data and Features

The Malicious URLs dataset provided 522,214 URL instances after filtering to retain only the benign (428,103) and phishing (94,111) categories. This 4.55:1 class imbalance reflects the real-world distribution of legitimate to phishing URLs encountered in typical browsing scenarios. Eighteen structural features were engineered from each URL using the tldextract library for domain decomposition and python's urllib.parse for component extraction. These features fall into five different categories. Length-based features — url_length, domain_length, path_length, and query_length capture the overall complexity of the URL. Phishing URLs consistently exhibit longer paths and total lengths than legitimate URLs because attackers add fake hierarchical path segments to simulate legitimate site structure. Structural composition features — dot_count, slash_count, and hyphen_count capture the complexity of the URL.Excessive dots indicate deep subdomain nesting (login.secure.account.paypal-fake.com), multiple slashes indicate complex path structures, and hyphen in domain names are a well-documented indicator of brand impersonation. Security and encoding features — has_https (binary), has_ip (binary indicating IP address substitution for domain), at_count (@ symbol presence), double_slash_in_path, and hex_char_count capture deliberate obfuscation techniques.IP address usage in place of a registered domain is a strong sign of phishing because legitimate websites always use named domains. Trust and reputation features — has_suspicious_tld, is_trusted_domain, and is_shortened encode knowledge about domain-level trustworthiness. Suspicious TLD classification was based on Anti-Phishing Working Group empirical data. Trusted domain verification was performed against Tranco top-500,000 domain list, a population-scale ranking derived from multiple passive DNS measurement platforms, replacing the static whitelist approach used in conventional systems. Keyword and content features —

suspicious_kw_count and digit_ratio_domain capture semantic signals embedded within the URL string itself. The appearance of the credential-related terms (such as login, verify, secure, password) within URL path components is a strong phishing indicator, as legitimate websites have no need to encode these terms in their URLs.

Methodology

Three classification algorithms were trained: Logistic Regression (C-1.0, balanced class weighting), Random Forest (100 estimators, balanced class weighting), and Gradient Boosting (100 estimators, learning_rate=0.1, max_depth=5). All features were standardized using StandardScaler fit on training data. An 80/20 stratified split produced417,771training and 104,443 test instances. Random Forest works by constructing an ensemble of T=100 decision trees, each trained on a bootstrap sample of the training data. At each internal node, a random subset of m=√18≈4 features are evaluated for splitting, selecting the feature and threshold that maximizes the Gini impurity reduction:

$$\Delta\text{Gini} = \text{Gini(parent)} - [\text{n\_left/n} \times \text{Gini(left)} + \text{n\_right/n} \times \text{Gini(right)}]$$

Where Gini(node) = 1 - Σ p2i summed over classes i. The final class probability for an unseen URL is computed as the average probability across all 100 trees, reducing variance substantially compared to individual tree estimates.

The analysis of feature importance showed that path_length (21.95%) and slash_count (17.70%) are the two most discriminative features. Together, they make up almost 40% of the model's predictive power. This finding is consistent with the known characteristic of phishing URLs, attackers construct complex fake path hierarchies to simulate legitimate site structure, unintentionally creating a robust structural signature.

A rule-based score adjustment layer was applied on top of the model probability to address known limitations of the training distribution. A 0.20 extra boost was applied for URLs containing high risk TLDs. A hard override returning a score of 0.05 was applied for URLs whose domains appear in the Tranco list with valid HTTPS and no suspicious signals which prevent false positives on legitimate commercial websites with structurally complex URLs. In order to improve recall for obfuscated phishing URLs at an acceptable precision cost, a dynamic threshold of 0.38 (compared to the standard 0.50) was adopted for URLs containing suspicious TLD, IP address, @symbol, or hexadecimal encoding.

The URL risk score is computed as:

$$S_{url} = P_{model} + B_{rules} \quad (2)$$

Where P_{model} represents the Random Forest classifier output and B_{rules} captures rule-based penalties for suspicious URL characteristics. Design Justification Random Forest was selected over Gradient Boosting despite comparable performance because URL features exhibit non-linear conditional interactions that benefit from Random Forest's ensemble averaging, particularly the interaction between path length and domain trust, where long paths are high-risk in the context of suspicious domains but expected in legitimate sites like GitHub and Stack Overflow. The Tranco-backed domain trust system represents a significant advancement over prior approaches that rely on manually curated whitelists containing ten to hundreds of domains but the Tranco top-500,000 covers all major legitimate websites globally and is updated daily without manual intervention.

Output

The URL analysis module returns a risk score, confidence band, verdict, and up to three triggered URL signals including detected suspicious TLD, IP address usage, suspicious keyword count, and HTTPS status. For trusted domains, the Tranco rank appears in the explanation as evidence of legitimacy.

D. Deep Feature-Based Image Analysis

Visual deception is a primary component of many scam campaigns. Fraudulent advertisements frequently use images and many visuals which are created to display the imitation of logos of trusted brands, duplicating the layouts of official websites, and cluttered designs that prevent careful examination by overloading the cognitive mind, called false legitimacy. These visual manipulation signals cannot be captured by URL and text analyses. In order to process and examine the visual elements of the webpages such as advertisements, identify phishing patterns and the layouts of the scam page, a dedicated image analysis module is necessary. The principal challenge in this module is the limited availability of labeled scam advertisement imagery. So, there is a necessity for an approach that is capable of achieving strong generalization from a small labeled dataset.

Data and Features

The image dataset, extracted from the Zenodo phishing screenshot research corpus, is composed of 1,000 website screenshots, 500 phishing pages and 500 legitimate pages. Each image was a full-page browser screenshot captured at 1920 × 995 pixels with JavaScript rendering enabled which is the most realistic webpage representation experienced by an end user. The transfer learning methodology is adopted as this dataset size is deliberately modest.

A frozen feature extractor, from MobileNetV2, of dimension 1280 is used as a feature vector for every image rather than creating it on our own. During the original ImageNet pretraining, the highly abstracted visuals, such as spatial arrangement, complexity

of the layout, the way colors are distributed, etc., learned from 14 million images across10000 categories during the original ImageNet pre-training, are encoded by this vector. The dimensions of the resulting feature matrix were 1,000 × 1,280.

Methodology

The pretrained ImageNet weights were used initially in MobileNetV2. The classifier head which is composed of a Dropout layer and a Linear layer mapping 1,280 features to 1,000 ImageNet classes was replaced with a torch.nn.Identity module which causes the network to output the penultimate 1,280-dimensional representation directly. The pretrained representations were used strictly as a feature extractor. To ensure no fine-tuning occurred, all3.4 million model parameters were frozen (requires_grad=False) throughout the process.

The following preprocessing pipeline was used to pass each image: RGB conversion, bilinear resizingto224×224pixels, conversion to a channels-first tensor with values in [0,1], and per-channel normalization using the ImageNet statistics mean = [0.485, 0.456, 0.406] and std = [0.229, 0.224, 0.225]. In order to avoid the gradient computation, all forward passes were performed within a torch.no_grad() context which reduces memory consumption and accelerates inference.

After StandardScaler normalization, three classifiers were trained on the extracted features (1,280-dimensional) : (i) Logistic Regression (C=1.0, balanced weighting), (ii) SVM with RBF kernel (C=1.0, probability calibration via Platt scaling, balanced weighting), and (iii) Random Forest (100 estimators, balanced weighting). With stratification, the dataset was split into 800 training and 200 test instances.

The Support Vector Machine with RBF kernel computes a decision function based on kernel similarity:

$$f(x) = \Sigma_i \alpha_i y_i K(x_i, x) + b$$

where $K(x_i, x) = \exp(-\gamma||x_i - x||^2)$ is the RBF kernel, α_i are the learned dual coefficients, $y_i \in \{-1, +1\}$ are training labels, and = 1/(n_features × V r(X)) by default. The model finds the hyperplane that maximizes the margin between the two classes in the feature space implicitly defined by the RBF kernel. Probability calibration converts the raw decision function values to probabilities using a sigmoid function fit via cross-validated Platt scaling.

The image risk score is given by:

$$S_{image} = P_{model} \quad (3)$$

Where P_{model} represents the probability output from the SVM classifier.

Results and Design Justification

The metrics F1 and ROC-AUC achieved by SVM are 0.8515 and 0.8968 respectively. Thus, SVM has outperformed Logistic Regression (F1=0.8116) and Random Forest (F1=0.8466). The SVM's superiority on MobileNetV2 features is theoretically calculated which shows SVMs were specifically designed to operate in high-dimensional spaces where the number of dimensions (1,280) exceeds the number of training samples (800). In such cases, the overfitting, which affects other algorithms, had been avoided by SVMs' maximum-margin optimization.

The transfer learning approach is justified on three bases. First, it achieves strong performance, i.e., F1=85.15%, from only 1,000 labeled images. In normal cases, fine-tuning would require more data to improve upon frozen feature extraction at this scale. Second, MobileNetV2's inverted residual bottleneck architecture efficiently encodes both minute texture information and complex semantic structure within its 1,280-dimensional output. This results in providing a clean discriminative signal for the downstream classifier. Third, the real-time scoring inferences usually require GPU infrastructure, but due to this approach inference requires only a CPU forward pass through a frozen network followed by a single SVM kernel computation.

Since the kernel-based decision made by SVM does not naturally decompose into feature contributions, a supplementary visual property analysis module computes rule-based features such as average brightness, brightness standard deviation, red channel dominance, high-contrast flag, and color variance from pixel statistics to provide human-interpretable visual explanations along with the score of the SVM.

Output

The image module returns a risk score between 0.0 and 1.0, confidence band, verdict, and the visual property explanation(s) up to three which describe the specific visual properties that contributed to the examination.

E. Review Sentiment & Credibility Analysis

These days, social proof such as reviews are the users' trust in everything that they buy online. These are used as the loop holes to the scam operators by artificially creating fake positive reviews to credit the fraudulent products and services and creating fake negative reviews to damage legitimate competitors. The challenge in the automated detection of fake reviews is made more complex by the surface similarity between artificially generated positive reviews and genuine customer feedback. Simple sentiment or keyword approaches are not insufficient because fake reviews are deliberately created to appear as real ones. Thus, this module of review analysis addresses this challenge through a multi-dimensional feature approach that simultaneously

analyzes sentiment characteristics, vocabularies, behavioral patterns. This captures the subtle statistical differences between human-made reviews and artificially generated reviews.

Data and Features

The Fake Reviews dataset is perfectly balanced and provides 40,432 labeled reviews. In which, 20,216 computer-generated fake reviews (CG) and 20,216 original human reviews (OR). VADER (Valence Aware Dictionary and Sentiment Reasoner) sentiment analysis provided four sentiment scores when applied to each review. VADER was specifically designed and validated for text with short-form product reviews, informal content, and requires no fine-tuning. That is why, it was selected over transformer-based sentiment models. Also, VADER operates at millisecond-latency which is suitable for real-time scoring, and its joint score provides a calibrated continuous sentiment signal which is directly useful as a numerical feature.

There are twenty handcrafted features extracted per review that span four categories.

Feature of linguistic diversity: The richness in vocabulary is captured by the entities such as unique word ratio, average word length, and character and wordcount. The intuition is that genuine customer reviews contain diverse, specific vocabulary reflecting personal experience, whereas computer-generated reviews tend to be shorter and use simpler, more repetitive language which produces a statistically lower unique word ratio and shorter average word length. These features were identified as the two most important derived signals by Random Forest feature importance analysis (5.16% and 2.83% respectively).

Sentiment features — the emotional characteristic of the review is captured using the VADER output labels namely positive, negative, neutral, compound. The intuition that was observed is that Fake reviews exhibit extreme joint scores (either strongly positive or strongly negative) whereas genuine reviews show mixed sentiment that shows balanced personal evaluation. The extreme sentiment flag (|compound| > 0.85) encodes this as a binary signal.

Feature of deception pattern: The specific characteristic of linguistic patterns has been detected by phrase counts such as fake positive phrase count, fake negative phrase count, and generic phrase count. These ideas were derived from analysis of known fake review patterns such as exaggerated superlatives ("absolutely perfect", "life changing", "best ever"), attack phrases ("complete waste", "total scam", "avoid at all costs"), and generic filler content ("good product", "fast shipping", "as described") commonly produced by computer-based automated generation models.

Behavioral signals — the feature of mismatch in sentiment-rating detects logical inconsistency between the textual sentiment severity and the numerical star rating. This mismatch arises when review makers produce text without knowing or ensuring it matches with the star rating, or when generated text has insufficient contextual meaning or information. Very short reviews (10 words or less) and very long reviews (over 150 words) capture length limits (both the extremes) associated with automatically generated texts.

Methodology

The same three-model comparison was conducted as in the text module. Features were combined from TF-IDF (2,000features, bigrams, min_df = 3, sublinear_tf =True) and the 20 hand-written features using the concatenation of sparse matrix and normalization using StandardScaler. Logistic Regression achieved the best metrics (proof: F1=0.9257andAUC=0.9800).

The scoring function implements a hybrid design that blends machine learning probability with a rule-based score:

$$S_{review} = 0.3 \times P_{model} + 0.7 \times S_{rule} \quad (4)$$

where P_{model} is the posterior probability of the Logistic Regression and S_{rule} is a rule-based score computed as:

S_{rule} = (0.20 + Σ positive_signal_contributions – Σ negative_signal_contributions)

Positive contributions are added for each detected deception signal as follows: 0.20 per fake positive phrase, 0.20 per fake negative phrase, 0.10 per generic phrase with an additional 0.15 extra value when three or more generic phrases are detected, 0.20 for extreme sentiment, 0.15 for low unique word ratio, and 0.15 for very short reviews. Negative contributions are subtracted for evidence of authentic review characteristics —0.15 for high unique word ratio, 0.12for reviews over 60 words, 0.12 for complex words, and 0.12 for mixed emotional sentiment texts. The score is normalized to [0.0, 1.0].

The rule-based score shown by the TF-IDF model was weighted 70% because the empirical analysis showcased reduced generalization to review text from domains and styles not well represented in the Fake Reviews training corpus but accurate on held-out test data taken from the same distribution. The rule-based component captures universal fake review patterns that remain the same across various product categories and writing styles.

When multiple reviews are available, individual review scores are aggregated using confidence-weighted averaging:

$$S_{final\_review} = \frac{\sum W_i \times S_i}{\sum W_i} \quad (5)$$

where S_i represents the individual risk score of the i-th review, and the weight (W_i) for each review is defined as:

$$W_i = |S_i - 0.5| + 0.1 \quad (6)$$

Output

The review module returns a pooled risk score, confidence band, verdict, the count of reviews analyzed, and explanation signals (max.3) drawn from the aggregated feature analysis across all provided reviews.

F. Multi-Modal Decision Fusion

The individual module scores are synthesized by an adaptive fusion layer. This layer employs a weighted logic mechanism that adjusts its calculations dynamically based on which input modalities are present. Furthermore, the fusion layer applies cross-modal consistency rules; for example, if both the URL engine and the Text module independently flag high-risk anomalies, the system applies a synergistic penalty, elevating the final fused score. The multi-modal score fusion mechanism is shown in Fig. 3.

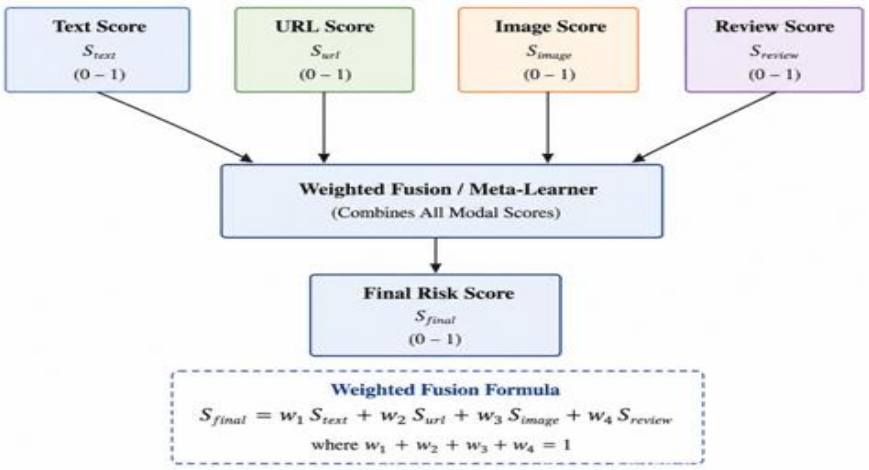

Fig. 3. Multi-Modal Score Fusion Mechanism.

The final scam risk score is obtained by combining all modality scores:

$$S_{final} = w_1 S_{text} + w_2 S_{url} + w_3 S_{image} + w_4 S_{review} \quad (7)$$

Where:

$$w_1 + w_2 + w_3 + w_4 = 1 \quad (8)$$

G. LLM-Driven Explainable AI (XAI) Framework

A primary limitation of traditional risk-scoring systems is the inability of non-technical users to interpret numerical outputs. To resolve this, the architecture incorporates an advanced, LLM-driven Explainability module. Rather than relying on static string generation, the system first aggregates the top-contributing risk indicators isolated by the analysis modules (e.g., "URL domain contains multiple hyphens," "Tranco rank: 50," or "High uppercase ratio"). These raw, algorithmic indicators, alongside the final verdict, are structurally formatted into a prompt and submitted to a Generative Large Language Model via API integration. The LLM acts as a translation layer, synthesizing the discrete data points into a cohesive, conversational 5-to-8 sentence paragraph. This approach ensures that the reasoning behind the classification is presented dynamically and naturally, directly addressing the user's specific inputs.

H. Decision Mechanism

The system classifies advertisements based on a threshold. The risk score interpretation and decision threshold logic is shown in Fig. 4.

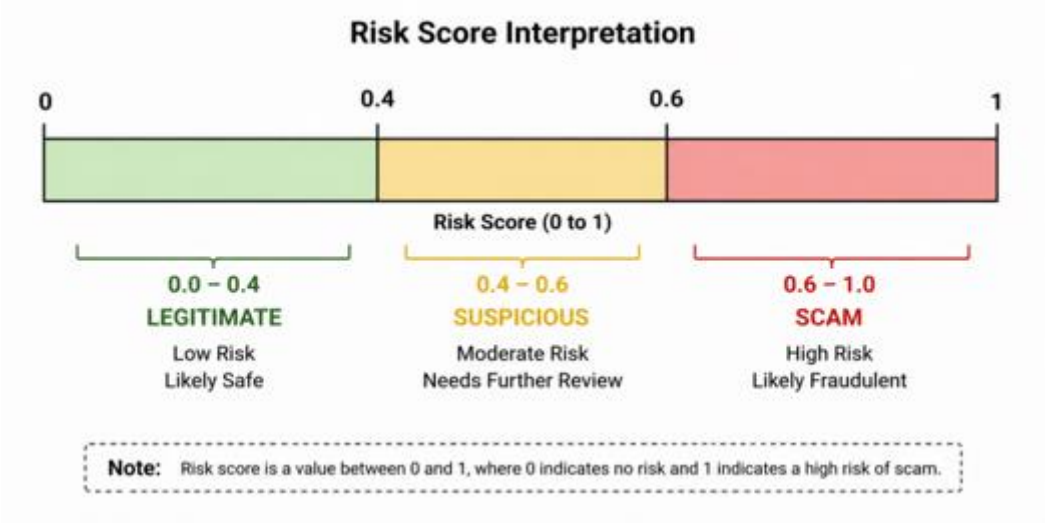

Fig. 4. Risk Score Interpretation and Decision Threshold.

The system triggers a Scam classification if the condition is met:

$$Scam \ \ if \ \ S_{final} \geq T \tag{9}$$

IV. System Workflow

The operational pipeline of the detection system progresses through a sequential workflow:

Input Reception: The system captures user-provided advertisement data via the primary interface.

Validation & Preprocessing: The central gateway standardizes the raw data and bypasses missing inputs.

Independent Scoring: The Text, URL, Image, and Review modules operate in parallel, applying their specific algorithms to generate independent risk scores (0 to 1).

Adaptive Fusion: The fusion layer collects the available risk scores and computes cross-modal penalties to generate a unified, final scam probability metric.

Verdict Classification: The final probability is mapped against predefined decision thresholds to categorize the advertisement as Legitimate, Suspicious, or Scam.

Feature Extraction: The system extracts the explicit heuristic indicators responsible for the decision from each activated module.

LLM Explanation Generation: The extracted indicators are fed into the integrated Generative LLM, which synthesizes a natural, conversational explanation of the verdict.

Output Presentation: The system presents the final verdict, the specific risk score, and the LLM-generated explanation back to the user.

V. Results And Discussion

The implementation of a multi-modal approach significantly limits the concept of a "single point of failure. "By evaluating the advertisement holistically, the system demonstrates the ability to identify deceptive intent even when a scammer attempts to obscure their methodology.

The integration of the LLM-driven XAI framework provides a massive improvement in usability. Traditional systems that output raw weights or static rule-based warnings often confuse end-users. By utilizing a generative model, the system explains complex combinations of factors—such as how a trusted domain is being misused to host suspicious text—in a fluid, conversational manner. This bridges the gap between complex algorithmic outputs and user comprehension.

The primary advantage of this architecture is its combination of analytical breadth and operational transparency. The utilization of lightweight models like MobileNetV2 ensures that the system is computationally viable, while the LLM handles the complex task of human-computer interaction. However, limitations exist. The reliance on an external LLM API requires stable network connectivity and introduces slight latency during the explanation generation phase. Additionally, highly sophisticated zero-day scam templates might evade initial detection until the system's baseline classifiers are updated.

VI. CONCLUSION

As social media platforms continue to serve as primary venues for digital commerce, the proliferation of deceptive advertisements poses a continuous threat to user security. This research presented a Multi-Modal Scam Detection System designed to systematically deconstruct and analyze promotional content. By concurrently evaluating linguistic properties, URL structures, visual patterns, and review sentiment, the system identifies complex fraudulent strategies that evade standard single-factor scanners. The integration of an adaptive fusion mechanism ensures reliable decision-making, while the embedded LLM-driven Explainable AI framework translates complex algorithmic indicators into transparent, human-readable reasoning. This approach actively empowers users, cultivating greater digital awareness and facilitating safer online interactions.

VII. FUTURE WORK

While the current architecture establishes a robust foundation for multi-modal detection, several avenues exist for future enhancement. Transitioning from standard machine learning classifiers to advanced contextual deep learning architectures (like locally-hosted Transformers) could provide deeper semantic understanding without relying heavily on external APIs. Operationally, packaging the system into a real-time browser extension would allow for automated, in-the-loop detection without requiring manual user input. Finally, expanding the explainability framework to include interactive visual highlights—such as bounding boxes on suspicious image regions—could further enhance the intuitive understanding of the detection process.

REFERENCES

[1] U. Zafar and F. Wu, “Methodological challenges in explainable AI for fraud detection: A systematic literature review,” Artificial Intelligence Review, Feb. 2026, doi: 10.1007/s10462-026-11516-7.

[2] M. Nasser et al., “A systematic review of multimodal fake news detection on social media using deep learning models,” Results in Engineering, vol. 26, p. 104752, 2025, doi: 10.1016/j.rineng.2025.104752.

[3] B. Austin, A. Afolabi, C. C. Ike, and N. Hussain, “AI and machine learning for detecting social media-based fraud targeting small businesses,” Open Access Research Journal of Engineering and Technology, vol. 7, pp. 142–152, Jan. 2025, doi: 10.53022/oarjet.2024.7.2.0067.

[4] L. Guo, Z. Chen, and X. Liu, “A fake news detection framework integrating multi-domain and multimodal features,” Neurocomputing, vol.658, p. 131711, 2025, doi: 10.1016/j.neucom.2025.131711.

[5] A. Y. Fu, L. Wenyin, and X. Deng, “Detecting phishing web pages with visual similarity assessment based on Earth Mover’s Distance(EMD),” IEEE Transactions on Dependable and Secure Computing, vol. 3, no. 4, pp. 301–311, 2006, doi: 10.1109/TDSC.2006.50.

[6] N. Mughal, G. Mujtaba, M. H. Mughal, A. Manaf, and Z. Kamangar, “Fake reviews detection on e-commerce websites using novel user behavioral features: An experimental study,” ACM Transactions on Asian and Low-Resource Language Information Processing, vol. 24, no. 9,Sep. 2025, doi: 10.1145/3748493.

[7] C. Hutto and E. Gilbert, “VADER: A parsimonious rule-based model for sentiment analysis of social media text,” Proc. Int. AAAI Conf.Web and Social Media, vol. 8, no. 1, pp. 216–225, May 2014, doi: 10.1609/icwsm.v8i1.14550.

[8] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, “MobileNetV2: Inverted residuals and linear bottlenecks,” inProc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2018, pp. 4510–4520.

[9] S. M. Lundberg and S.-I. Lee, “A unified approach to interpreting model predictions,” in Advances in Neural Information Processing Systems, 2017.