Trustworthy App Detection in Saudi Mobile Finance: Bridging Deep Learning and Interpretability

Raed Alharbi, Maryam Alghamdi
College of Computing and Informatics
Saudi Electronic University, Riyadh 11673, Saudi Arabia

***Abstract*—Understanding user trust in mobile financial applications is crucial as these platforms increasingly shape how users in Saudi Arabia manage finances and engage with digital banking. However, existing deep learning–based detection models often function as black boxes, offering limited interpretability, while traditional machine learning models, though more transparent, fail to capture complex interactions between permissions, reviews, and user behaviors. To address this gap, we propose Trustworthy App Detection for Saudi Arabia (TAD-Saudi) - a novel framework for interpretable and behavior-aware trust evaluation. The framework integrates the representational power of deep learning with the explainability of simpler models, enabling both global and local interpretation of trust-related features. Experimental results show that TAD-Saudi outperforms traditional baselines across multiple models. Moreover, the analysis reveals that users may continue to trust applications requesting sensitive permissions, particularly when these apps have high ratings or positive reviews.**

***Keywords*—XAI; trustworthy apps detection; encoding**

I. Introduction

The widespread adoption of financial applications in Saudi Arabia has transformed the way users manage their finances, perform transactions, and engage with digital banking services. With increasing reliance on mobile applications, the Saudi fin-tech sector has experienced significant growth, reflecting broader global trends in the adoption of mobile finance. A recent market analysis indicates a notable increase in financial app usage in the region, with a substantial increase in mobile banking transactions and fintech investments [1]. However, alongside this growth, concerns about app trustworthiness, data privacy, and security risks have emerged, particularly as users trust these applications with sensitive financial and personal data.

To develop an effective and trustworthy Saudi financial app detection model, it is crucial to: 1) understand the key factors that influence user trust in financial applications, such as app ratings, reviews, and security permissions; 2) analyze the role of such factors in the trustworthiness of an app, as excessive permission requests often raise security concerns among users; and 3) address the "black-box" nature of machine learning-based classification models [2], which often lack interpretability, making it difficult for users to rely on the classification outcomes with confidence. These factors emphasize the need for a detection framework that not only ensures high accuracy when designing machine learning models but also provides explainability and transparency to users.

Despite notable progress in app fraud detection, current detection methods lack explainability and user-centered transparency. Most existing approaches rely on sentiment analysis or metadata-driven classification, which focus on identifying fraudulent or malicious apps rather than explaining why an app is considered trustworthy. This absence of interpretability limits user confidence and delays the practical adoption of Artificial Intelligence-based (AI-based) trust evaluation systems. Furthermore, existing models often fail to capture the effect of user trust signals (such as ratings and reviews) and app behavioral features (such as permissions and update frequency). While deep learning models provide high accuracy, their decisions remain opaque, and traditional machine learning models, though interpretable, are too simplistic to capture these complex behavioral relationships.

To overcome these challenges, we propose TAD-Saudi, a framework that is designed to enhance the evaluation of mobile app trustworthiness. The core of TAD-Saudi consists of three main components. First, we introduce a novel benchmark that is designed to assess key cybersecurity features. This benchmark systematically collects and evaluates various factors that influence the acceptance or rejection of app trustworthiness, offering a robust foundation for understanding the critical elements that contribute to establishing trust in mobile applications. Second, TAD, our proof-of-concept framework, integrates the power of deep learning models with the explainability of simpler, interpretable models. Finally, we conduct an in-depth analysis to examine the underlying factors driving user trust, providing valuable insight into the behavioral aspects influencing app adoption.

The remainder of this study is organized as follows. Section II presents the related works. The proposed architecture is introduced in Section III, where Section IV provides the experiment setup, analysis, and discussion of our findings. Section VI concludes the study with a summary of our contributions.

II. Related Work

Several recent studies provide valuable insights into the evolving landscape of privacy concerns and trustworthiness assessment for mobile applications. Habib et al. [3] introduce Trust4App, a framework that automates mobile app trustworthiness assessment by combining public metrics (e.g., ratings, reviews) with security factors, and user-specific preferences. This model integrates existing approaches, such as static malware analysis, into a unified personalized trust score. Similarly, [4] propose a graph-based framework that assesses smartphone apps' trustworthiness by integrating ratings, reviews,

and permissions data into confidence metrics. While both frameworks contribute to multi-dimensional trust assessment, they rely primarily on static metadata and lack mechanisms for interpretability, limitations that motivate the explainable and behavior-aware design of TAD-Saudi.

Akgul et al. [5] conduct an analysis of privacy-related app reviews on the Google Play Store over ten years (2013–2023) using Natural Language Processing (NLP) techniques. Their findings reveal a growing shift in user attention from traditional permission-based privacy issues toward emerging risks such as data deletion, third-party sharing, and data theft, while also highlighting regional variations in user behavior and privacy expectations. Nema et al. [6] propose an NLP-based framework for extracting and categorizing privacy concerns from Android app reviews, identifying key issues such as unnecessary permissions, data collection practices, user tracking, and insufficient privacy controls. Although these studies provide valuable insights into user perceptions of privacy, they remain primarily descriptive and lack quantitative or interpretable detection mechanisms.

Recent efforts have highlighted the growing impact of explainable AI (XAI) in enhancing transparency and reliability within financial and cybersecurity frameworks. For example, Turgut et al. [7] employ XAI techniques such as SHapley Additive exPlanations (SHAP), Local Interpretable Model-Agnostic Explanations (LIME), and Explain Like I'm 5 (ELI5) to improve the interpretability of Machine Learning models used in financial fraud detection, emphasizing the importance of explainability for fostering user trust in automated decision systems. Similarly, Xu et al. [8] propose an interpretable credit-default assessment model that integrates multiple Machine Learning (ML) architectures with SHAP-based explanations, demonstrating how interpretability can enhance both trust and model comprehension in financial risk analysis.

Harkous et al. [9] propose Hark, a deep learning–based system that leverages NLP to analyze privacy-related app reviews, providing developers with actionable insights into user feedback and privacy perceptions. Similarly, [10] introduced MARS (Mobile App Reviews Summarization), an ML–driven framework that combines NLP and sentiment analysis techniques to automatically summarize user reviews from the Google Play Store, distilling key privacy concerns and recurring user issues. While both approaches enhance developers' understanding of user privacy feedback, they remain developer-centric and lack mechanisms for transparent or explainable trust assessment.

In the context of fraudulent mobile apps, Rani et al. [11] develop a sentiment analysis framework that detects fraudulent apps by analyzing user reviews and ratings to identify suspicious applications. Similarly, Zaki et al. [12] propose a robust machine learning architecture for detecting fraudulent mobile apps. Their findings demonstrate the higher accuracy of deep models, particularly Convolutional Neural Networks (CNNs), in detecting deceptive behavior and enhancing app security. However, these detection systems operate largely as black boxes, focusing on performance metrics without offering interpretable reasoning behind classification outcomes. TAD-Saudi diverges from these approaches by embedding explainability within its detection framework.

While there has been considerable progress in developing frameworks for assessing mobile app trustworthiness and detecting fraudulent applications, these approaches remain limited in several key aspects. Most existing frameworks focus primarily on classification accuracy without examining the underlying behaviors that drive trust predictions. They often fail to explore the factors influencing machine learning models' decisions, which is crucial for understanding why users trust certain apps and based on what criteria. Without this interpretability, users and developers lack meaningful insights into how security, privacy, and app behavior shape trustworthiness assessments.

III. The Proposed Architecture

This study introduces TAD-Saudi, as illustrated in Fig. 1, where the main components are 1) a novel benchmark that is designed to assess cybersecurity features, systematically collecting and evaluating various factors that influence the acceptance or rejection of app trustworthiness. This benchmark provides a comprehensive framework for understanding the underlying elements contributing to the establishment of trust in mobile applications; 2) TAD, a proof-of-concept framework that balances the richness of deep learning with the explainability of simpler models to predict and classify trustworthy apps, thus enabling the identification of secure and reliable mobile applications.

This section starts with an overview of the dataset, followed by detailed explanations of the proposed framework.

A. Data Preparation

Given the nature of this study, which investigates financial applications in the Saudi region, our focus is specifically on mobile apps that support the Arabic language. This emphasis is crucial, as the majority of users in the region rely on Arabic-language interfaces for their digital interactions.

We construct our dataset through a combination of web scraping techniques and APIs to gather comprehensive data on finance-related mobile applications from the Google Play Store [13]. Specifically, we utilize Python libraries such as $requests$, $BeautifulSoup$, and $google_play_scraper$ to extract detailed information on applications, including their descriptions, user reviews, ratings, and security permissions. To ensure a broad and representative coverage of financial applications, we collect data from 23 finance-related categories, such as "Banking," "Investment," "Loans," "Insurance," "Expense Tracking", and "Financial Planning" [13]. By focusing on these categories, we ensure a diverse dataset that specifically serves the needs of Arabic-speaking users in Saudi Arabia, reflecting the region's unique financial landscape and user preferences. To address the possibility of the Python libraries extracting irrelevant results, each extracted application was manually reviewed. During this process, all duplicate applications and any applications not directly related to financial services were removed. This ensured that the final dataset contained only financial-related applications.

To achieve this, we first scrape app IDs from multiple pages within each category URL to build a comprehensive dataset [14]. For each app, we collect metadata, including the app name, description, overall rating, total ratings, and a sample

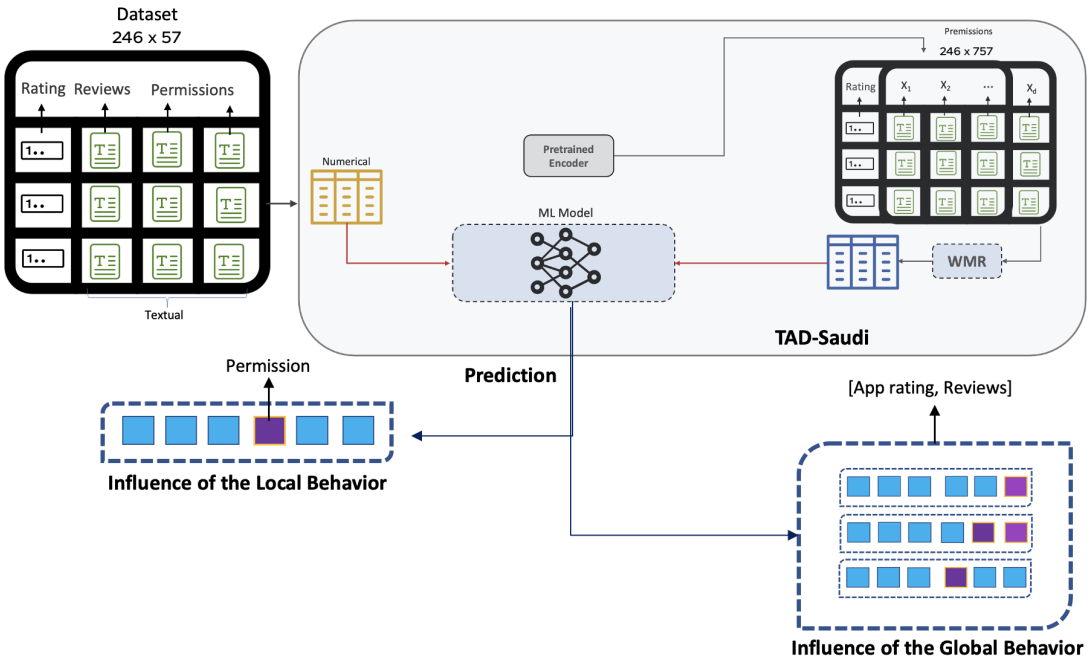

Fig. 1. Our proposed architecture (TAD-SAUDI).

of user reviews. Specifically, we selected the top 20 reviews for each app, with a cap of 15 apps per category, to maintain focus while ensuring diversity across categories. This approach enables us to capture a representative sample of user feedback. The resulting dataset serves as the foundation for analyzing the correlation between app permissions, user perceptions (derived from reviews and ratings), and overall trustworthiness.

In our framework, we operate within a supervised setting, where labeling is a critical step in the development of such an ML model. Specifically, our labeling process involves extracting the top 20 reviews for each application. If at least one of these reviews expresses a significant lack of trust in the application, it is classified as "untrustworthy". Conversely, if none of the reviews raise concerns about trust, the application is labeled as "trustworthy". For each application, the top 20 user reviews are manually examined to identify trust-related content. Reviews explicitly mentioning privacy violations, security concerns, fraudulent activity, data misuse, or suspicious behavior are classified as indicators of untrustworthiness. Reviews focusing on usability, performance issues, software bugs, interface design, or general instability are excluded, as these aspects do not directly reflect users' trust perceptions. Notably, the trust labels in this dataset represent users' subjective perceptions of trustworthiness, rather than verified cybersecurity incidents or objectively confirmed malicious behavior.

The labeling process is conducted manually by the authors. A primary annotator performs the initial labeling, which is subsequently verified by a secondary reviewer to ensure consistency and reliability. As native Arabic speakers, the authors are able to accurately interpret user reviews, particularly since many trust-related concerns are expressed using culturally specific phrases and informal dialects. During the data cleaning process, all non-Arabic reviews, emojis, and irrelevant symbols are removed to avoid potential noise in model performance. This step to ensure that only semantically relevant Arabic content is retained for analysis.

Our constructed dataset consists of a total of 246 applications, with a particular emphasis on financial applications. These applications are categorized into 172 trustworthy apps and 74 untrustworthy apps. Additionally, the dataset includes 3,430 Arabic-language reviews, an equal number of ratings, and 1,581 permissions requested by the apps. The summary of the dataset is illustrated in Table I.

TABLE I. SUMMARY OF THE DATASET

Metric	Value
Total Apps	246
Trustworthy Apps	172
Untrustworthy Apps	74
Total Ratings	3,430
Total Reviews	3,430
Total Permissions	1,581
Total App ID	246
Total App Descriptions	246

As part of our data preparation process, we conducted a human-subjective study involving 51 participants to explore user priorities in selecting trust factors prior to downloading mobile applications. Informed consent was obtained in writing from all participants through a signed consent form before participation.

B. TAD Framework

The goal of TAD framework (Fig. 1) is balance the richness of deep learning with the explainability of simpler models to predict and classify trustworthy apps. Our proposed framework

TAD consists of three main components: 1) A Pretrained Encoder to learn textual representations and extract meaningful and compact feature representations from high-dimensional textual data, 2) Weighted Mean Representation (WMR) to reduce redundancy while preserving essential semantic information, ensuring that the latent representation captures the most discriminative features. Also, we employ attention scores to assign different levels of importance to each feature dimension, and 3) Machine learning model that aims to develop a robust predictor for evaluating the trustworthiness of financial mobile applications, specifically determining whether a user is likely to accept or reject the app. To achieve this, we leverage well-known machine learning classification techniques: XGBoost (XGB) [15], Decision Tree [16], K-Nearest Neighbors (k-NN) [17], Random Forest (RF) [18], and Support Vector Machine (SVM) [19], all of which are well-established in classification tasks. To ensure a comprehensive and fair comparison, we also incorporate a deep learning-based approach, TABNET [20], which is specifically designed for handling tabular data.

Our dataset contains multiple reviews, where each review can have several comments. This structure results in a combination of textual data and straightforward numerical features such as ratings. A naive way to handle such a dataset is to process numerical attributes directly while encoding textual data using methods like label encoding, which assigns a unique integer to each word or phrase. Mathematically, this can be represented as a function mapping words to indices:

$$f(w_i) = i, \quad \forall w_i \in W \tag{1}$$

where, W is the vocabulary and w_i represents each unique word. However, label encoding introduces an inherent ordinality to categorical values, implying an artificial ordering among words (e.g., "good" assigned 1, "bad" assigned 2), which lacks semantic meaning. This causes misleading numerical relationships in machine learning models, as they might incorrectly assume a linear relationship between words based on their assigned numbers. Thus, the text presents additional challenges due to its high-dimensional nature and contextual dependencies.

An alternative solution is to use Deep Neural Networks (DNNs) that employ word embeddings to learn semantic relationships in a high-dimensional space. Mathematically, an embedding function E maps words to dense vectors:

$$E : W \to \mathbb{R}^D, \quad D \gg 1 \tag{2}$$

where, D is the embedding dimension, capturing semantic similarities between words. This allows words with similar meanings to be closer in the vector space, mitigating the limitations of label encoding.

However, as our primary goal is to leverage XAI to understand how user behaviors influence app interactions, relying solely on DNNs [21] can obscure the interpretability of individual financial behaviors (black box models), such as the effect of specific permission requests.

To address both interpretability and effectiveness, we propose an embedding-based feature engineering approach that balances the richness of deep learning with the explainability of simpler models. Instead of directly applying DNNs, we extract pre-trained embeddings for key textual fields, such as description, reviews, and permissions, then reduce them in a structured manner.

Each text field is converted into a high-dimensional vector representation using a pre-trained embedding model. If an embedding of size D is used, then:

$$X_{\text{embedding}} = \{X_1, X_2, ..., X_m\} \tag{3}$$

where, m is the number of textual fields (e.g., app description, reviews, and permissions), and $X_k \in \mathbb{R}^D$ represents the embedding vector of the $k-th$ text field for a given sample. In other words, each text field contributes multiple embedding dimensions—occupying several columns in the tabular dataset.

To make the model interpretable and computationally efficient, we propose the WMR as it is shown in Fig. 1. Mathematically, given

$$x_i = [x_{i1}, x_{i2}, \ldots, x_{iD}]$$

where, x_i represents the feature vector (embedding) for sample i, and D is the embedding dimension. Similarly, the attention weights for sample i are given by:

$$w_i = [w_{i1}, w_{i2}, \ldots, w_{iD}]$$

The weighted mean representation is computed as:

$$\tilde{x}_i = \frac{\sum_{j=1}^{D} w_{ij} x_{ij}}{\sum_{j=1}^{D} w_{ij}} \tag{4}$$

where:

- x_{ij} is the j-th feature of sample i,
- w_{ij} is the j-th attention weight for sample i. The denominator ensures that the weights sum to 1, preserving scale consistency.

This transformation converts each high-dimensional embedding into a single scalar value, significantly reducing complexity while preserving essential semantic information. Without this reduction, the model would struggle to integrate embedding-based features with standard scalar or categorical attributes.

Unlike traditional pooling mechanisms such as attention-based aggregation, which typically assign weights across tokens within a sequence to generate a contextualized sentence embedding, the proposed WMR operates at the feature-dimension level. In other words, instead of weighting tokens, WMR assigns importance to individual embedding dimensions.

The reduced embeddings are concatenated with the remaining non-embedding features. The final feature matrix is denoted as:

$$X' = [X_{\text{structured}}, X_{\text{final}}] \tag{5}$$

where, $X_{\text{structured}}$ consists of traditional tabular features such as categorical metadata and numerical attributes. To mitigate class imbalances, we also apply SMOTE (Synthetic Minority Over-sampling Technique), which generates synthetic samples for underrepresented classes:

$$X_{\text{balanced}}, Y_{\text{balanced}} = \text{SMOTE}(X', Y) \tag{6}$$

IV. Experimental Analysis

We begin this section by providing details on the dataset (Section IV-A) and experimental setup (Section IV-B). Our objective is to first evaluate the performance of our proposed model in terms of predictive accuracy (Section IV-C). Following this, we conduct in-depth analysis (Section IV-D) using SHAP [22] and Section IV-E using LIME [23]) on the top-performing model to investigate: 1) the key features influencing an application's trustworthiness and 2) the relationship between specific features and the acceptance or rejection of an application. Finally, in Section IV-F, we shift our focus to a human-subjective experiment on user trust, aiming to validate our findings through direct user evaluation.

A. Dataset

We compile our dataset through a combination of web scraping techniques and API calls to gather information on finance-related mobile apps from the Google Play Store. The data preparation process is detailed in Section III-A.

The constructed dataset consists of 246 finance-related mobile apps. Given that our goal is to investigate whether users trust mobile applications despite the permissions they request, and to identify the factors influencing their decision-making process, it is crucial to examine the permissions collected for each app. The dataset includes 14 distinct permission categories. Statistics on the permissions associated with the dataset are presented in Table II.

TABLE II. Statistics on Dataset Permissions

Permission	% of Apps
Photos/Media/Files	85
Storage	85
Camera	79
Wi-Fi connection information	77
Location	69
Contacts	54
Phone	53
Device ID and call information	52
Microphone	44
Identity	21
Calendar	9
SMS	7
Device and app history	6.5
Wearable/Activity Sensor Data	0.4

To ensure broad coverage of the Saudi mobile finance ecosystem, the dataset spans 23 finance-related categories on the Google Play Store, including banking, investment, insurance, lending, expense tracking, and financial planning. Within each category, the top-ranked applications were selected to reflect the apps most likely to be encountered by typical users, since marketplace ranking strongly influences user exposure. The resulting class proportions are reported with 95% Wilson confidence intervals: trustworthy applications constitute 69.9% of the dataset (95% CI [63.9%, 75.3%]) and untrustworthy applications 30.1% (95% CI [24.7%, 36.1%]). A power analysis confirms that at $n = 246$, $\alpha = 0.05$, and $power = 0.80$, the dataset is adequate to detect medium effect sizes (Cohen's $d \geq 0.389$) between the two classes.

Two sources of potential bias are acknowledged. First, the dataset is restricted to applications available on the Google Play Store and does not include Apple App Store apps or applications distributed through alternative Arabic app stores; findings may therefore not transfer directly to iOS users. Second, the top 20 reviews per app sampling strategy favors reviews surfaced by Google's ranking algorithm, which tends to prioritize reviews marked as helpful or those that are more recent, potentially underrepresenting older or less-visible user concerns. We mitigate these effects by drawing from 23 distinct finance categories rather than concentrating on a single segment, which broadens topical and demographic coverage.

The dataset contains 172 trustworthy and 74 untrustworthy applications, yielding an approximate 2.32:1 ratio. This imbalance reflects the natural distribution of trust perceptions in the financial app marketplace, where most listed applications belong to established institutions and receive predominantly positive trust signals. To prevent the majority class from masking poor minority-class performance, we apply stratified k-fold cross-validation, which preserves the class ratio in each fold, and report macro-averaged precision, recall, and F1-score in addition to overall accuracy.

B. Experimental Settings

We randomly select 75% of the dataset for training and 25% for testing. For the Logistic Regression model, the maximum number of iterations is set to 500. For k-NN, we choose 5 as the number of nearest neighbors. Logarithmic loss is used as the evaluation metric to track the performance of the XGB model during training.

To effectively represent the semantic and contextual meaning of Arabic text, we employ the AraBERT pre-trained embedding model [24], which generates dense vector representations of the reviews. These embeddings serve as the textual input features for all models, enabling a deeper understanding of linguistic nuances and improving the overall accuracy of trustworthiness classification. To evaluate the performance of our proposed framework, TAD-Saudi, we use the following metrics:

- Accuracy: Measures the proportion of correct predictions out of the total number of predictions.
- Precision: Represents the ratio of correctly predicted positive instances among all instances predicted as positive.
- Recall: Indicates the ratio of correctly predicted positive instances relative to the total number of actual positive instances.

- F1 Score: The mean of precision and recall, providing a balanced evaluation of the model's performance.

The foundation of our TAD-Saudi framework lies in the proposed encoding schema. We compare our TAD-Saudi framework with three common encoding techniques: Label Encoding [25], One-Hot Encoding[26], and Binary Encoding[27]. To ensure a comprehensive comparison, we apply each of these encoding methods, along with our proposed framework, to the following models: RF [18], Gradient Boosting [28], XGB [15], TABNET [20], Logistic Regression [29], K-NN [17], SVM [30], and Naive Bayes [31]. A brief summary of these models is as follows:

- RF: A model that combines multiple decision trees using randomly sampled data and features.
- Gradient Boosting: An optimization-based ensemble method that builds additive models, often using decision trees, and minimizes loss via gradient descent.
- XGB: A scalable tree boosting system designed for high-performance machine learning on large datasets.
- TABNET: An interpretable deep learning architecture for tabular data that uses sequential attention to efficiently select important features.
- Logistic Regression: A statistical method used to model the relationship between a binary dependent variable and one or more independent variables.
- K-NN: A supervised machine learning algorithm that classifies data points based on the majority class among their 'k' closest neighbors in the feature space.
- SVM: One of the earlier supervised machine learning algorithms that has been instrumental in both classification and regression tasks.
- Naive Bayes: A simple probabilistic classifier based on Bayes' theorem, assuming that features are independent given the class label.

C. Performance Evaluation

The performance evaluation results presented in Table III highlight the effectiveness of the proposed TAD-Saudi framework in comparison to conventional encoding techniques, namely Label Encoding, One-Hot Encoding, and Binary Encoding. The findings demonstrate that TAD-Saudi consistently outperforms all traditional encoding methods across multiple machine learning models, achieving higher accuracy, precision, recall, and F1 scores.

Among the traditional encoding methods, Label Encoding, One-Hot Encoding, and Binary Encoding yield comparable performance, with accuracy scores ranging primarily between 0.60 and 0.71. The best-performing models under these methods include Logistic Regression (0.74, Label Encoding), Gradient Boosting (0.71, Label Encoding), and XGBoost (0.69, One-Hot & Binary Encoding). However, these models fail to exceed the 0.74 accuracy threshold, indicating the limitations of conventional encoding approaches in capturing the complexities of the dataset.

In contrast, TAD-Saudi achieves a significant performance boost, with XGBoost and Random Forest each attaining the highest accuracy of 0.85, followed by Gradient Boosting and Naïve Bayes at 0.81. These models also exhibit higher precision, recall, and F1 scores, reinforcing the dominance of TAD-Saudi in learning robust representations. The observed improvements across different models indicate that the TAD-Saudi framework enhances feature encoding, leading to better generalization and decision-making capabilities.

A key observation is the poor performance of TABNET across all settings, achieving accuracy values below 0.50 in most cases. As a result, deep learning-based methods may require additional tuning or architectural modifications to effectively handle the tabular data in the given dataset. Furthermore, the nature of tabular data, with its high sparse relationships, could pose significant challenges for models designed primarily for structured inputs like images or sequences [2]

Overall, these results validate the effectiveness of TAD-Saudi in improving predictive accuracy, particularly in handling categorical data more efficiently than traditional encoding techniques. The model demonstrates a clear advantage in capturing complex structured relationships within the data, leading to enhanced performance. This indicates that TAD-Saudi's approach to feature representation and processing offers a promising alternative for models dealing with large, structured datasets, where traditional methods often struggle.

D. Global Analysis

To understand the global behavior and underlying logic behind trustworthy and untrustworthy apps, we utilize SHAP [22], a unified method that offers interpretable explanations for the predictions made by machine learning models. By analyzing the feature contributions for apps, we can gain insights into which factors influence the classification of an app as trustworthy or untrustworthy. This allows us to identify patterns and relationships that clarify which factors are most critical in the determination of app trustworthiness, shedding light on the key characteristics that distinguish these two categories. We select the best-performing model, XGBoost, in our TAD-Saudi framework for our analysis to provide a reliable and accurate explanation of its behavior.

Fig. 2 presents the SHAP analysis of the top 20 factors that impact the trustworthiness of an app on our proposed framework, where the features are listed according to their total contribution (sum of SHAP values) to the final prediction, with the top feature being the most important. Each point represents an app in the study. The position of each point on the x-axis indicates the impact that feature has on the classifier's prediction for a given app, with the color gradient (ranging from blue to red) reflecting the feature values from low to high. On the x-axis, positive SHAP values indicate the trustworthiness of the app, whereas negative SHAP values reflect untrustworthiness. This visualization allows us to observe the relative influence of each feature on the prediction, providing a clear understanding of which factors play a dominant role in shaping an app's classification as trustworthy or untrustworthy.

The interpretation in Fig. 2 shows several key observations. First, permissions play a critical role in determining trust-

TABLE III. PERFORMANCE EVALUATION

Model	Accuracy	Precision	Recall	F1	Accuracy	Precision	Recall	F1
	Label Encoding				One-Hot encoding			
RF [18]	0.65	0.62	0.65	0.63	0.65	0.61	0.65	0.62
Gradient Boosting [28]	0.71	0.70	0.71	0.70	0.63	0.61	0.63	0.62
XGB [15]	0.68	0.67	0.68	0.67	0.69	0.67	0.69	0.67
TABNET [20]	0.48	0.55	0.48	0.50	0.48	0.55	0.48	0.50
Logistic Regression[29]	0.74	0.74	0.74	0.70	0.73	0.71	0.73	0.69
K-NN [17]	0.63	0.59	0.63	0.60	0.63	0.59	0.63	0.60
SVM [30]	0.69	0.48	0.69	0.57	0.69	0.48	0.69	0.57
Naive Bayes [31]	0.66	0.76	0.66	0.67	0.66	0.76	0.66	0.67
	Binary encoding				TAD-Saudi			
RF [18]	0.65	0.61	0.65	0.62	0.85	0.86	0.85	0.85
Gradient Boosting [28]	0.63	0.61	0.63	0.62	0.81	0.81	0.81	0.81
XGB [15]	0.69	0.67	0.69	0.67	0.85	0.85	0.85	0.85
TABNET [20]	0.48	0.55	0.48	0.50	0.43	0.43	0.43	0.43
Logistic Regression[29]	0.73	0.71	0.73	0.69	0.80	0.82	0.80	0.80
K-NN [17]	0.63	0.59	0.63	0.60	0.67	0.72	0.67	0.66
SVM [30]	0.69	0.48	0.69	0.57	0.80	0.81	0.80	0.80
Naive Bayes [31]	0.66	0.76	0.66	0.67	0.81	0.86	0.81	0.81

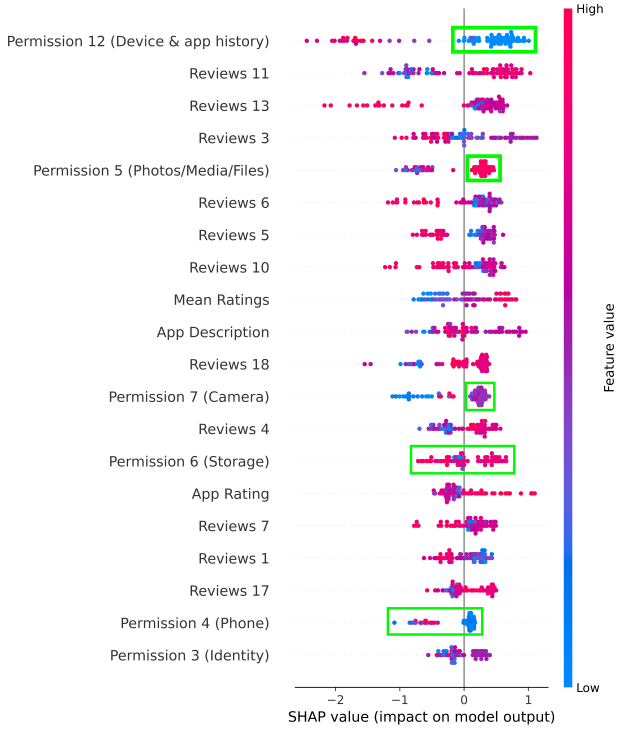

Fig. 2. SHAP analysis of the top 20 factors influencing app trustworthiness in the proposed framework. Features are ranked by their total SHAP contribution, with higher values indicating greater importance. Each point represents an app, where color denotes feature value (blue = low, red = high) and the x-axis shows impact direction—positive values indicate trustworthiness, while negative values reflect untrustworthiness.

worthiness, with Device & App History, Photos/Media/Files, Camera, Storage, Phone, and Identity permissions appearing among the most influential features. Apps requesting extensive permissions, particularly those related to sensitive data access, tend to have a negative SHAP impact (untrustworthy apps), indicating that such permissions are more commonly associated with untrustworthy apps. This aligns with security concerns where excessive permissions may indicate potential privacy risks [32].

While permissions generally exhibit a negative correlation with app trustworthiness in the SHAP summary plot, certain permissions demonstrate mixed or even positive contributions, indicating that their impact is context-dependent. For instance, Permission 12 (Device & App History) displays SHAP values that extend into the positive range, indicating that some apps requiring this permission—such as system utilities or security applications—are still classified as trustworthy. Similarly, Permission 5 (Photos/Media/Files) and Permission 7 (Camera) show instances where high feature values contribute positively to trustworthiness, likely reflecting legitimate use cases in photo-editing, document scanning, and video conferencing apps. Additionally, Permission 6 (Storage) and Permission 4 (Phone) exhibit mixed SHAP contributions, implying that while they are often associated with untrustworthy apps, certain well-rated applications that transparently justify these permissions remain trustworthy.

These observations explain that ratings and reviews alone are not sufficient to determine an app's trustworthiness, as permissions also play a crucial role in the classification process. While highly rated and frequently reviewed apps are often perceived as more reliable [33], the SHAP analysis indicates that certain permissions can significantly impact trustworthiness negatively. This highlights the importance of transparency and responsible permission management, reinforcing that trustworthiness is not shaped only by user ratings and reviews but by a combination of user feedback, feature permissions, and the app's overall reputation. While high ratings and positive reviews contribute to an app's credibility, they do not fully capture the potential risks associated with excessive or sensitive permission requests.

E. Local Analysis

To complement our global analysis of app trustworthiness using SHAP, we perform a nuanced local interpretability analysis using LIME [23]. LIME produces instance-specific explanations, enabling us to uncover how specific combinations of features influence the classification of apps as trustworthy or untrustworthy. This refined perspective exposes decision-making patterns that may remain hidden in global summaries, offering deeper insights into user-perceived and model-learned trust signals.

Our analysis focuses on the XGBoost model, chosen for its robust predictive performance in the TAD-Saudi framework. LIME is applied to each sample in the test set to extract a feature attribution vector, where each value quantifies the positive or negative contribution of a feature to the model's decision. Positive values support a classification of trustworthy, while negative values indicate influence toward untrustworthy. To identify dominant patterns of local behavior, we clustered these vectors using KMeans (k=5) [34].

Fig. 4 presents a heatmap showing the average LIME feature attributions across the five clusters. Each row corresponds to a cluster, and each column to a feature. Red cells indicate features that consistently support trust predictions, while blue cells indicate features that consistently detract from trust.

Our cluster-wise LIME analysis reveals how variations in user behavior and perception drive differences in trust formation across app groups. In Cluster 0, trust predictions are positively influenced by early reviews such as review_5 and review_6, along with permission_7 (Camera). However, later reviews like review_19 and review_20 show strong negative impacts. This shift indicates that users initially responded positively but later became dissatisfied—potentially due to app updates, intrusive notifications, or emerging performance issues. The fact that the Camera permission maintains a positive association with trust indicates contextual acceptance behavior, where users justify certain permissions when they align with the app's functionality (e.g., photo-sharing or video features). This pattern aligns with trust heuristics in digital behavior, where familiarity and early positive experience temporarily override privacy concerns.

In Cluster 1, user behavior reflects a contrasting cognitive pattern. Trust is supported by positive review features (review_8, review_17, review_19), but sharply decreases in the presence of permission_3 (Identity) and permission_10 (Device ID and Call Info). This reveals a privacy-centered behavioral profile in which users exhibit a greater sensitivity to permissions involving personal or device identifiers. Even when social feedback (reviews) is favorable, these users value privacy safeguards more than social validation, indicating a risk-averse trust model. Overall, the cluster patterns highlight that user trust is not purely sentiment-driven but context-dependent, balancing perceived utility, social validation, and privacy risk in complex behavioral trade-offs.

Clusters 2 and 3 exhibit a similar trust logic, where reputational signals—such as rating, review_17 (in Cluster 2), and review_11 (in Cluster 3)—positively influence trust predictions. However, both clusters show strong negative contributions from permission_10 (Device ID and Call Info) and select reviews (review_13 in Cluster 3 and description in Cluster 2). This pattern reveals a user tendency to favor high ratings and positive feedback, even in the presence of potential privacy risks—indicating that strong reputational signals often take precedence over deeper concerns about app behavior. Finally, Cluster 4 reflects a context-aware decision pattern. Positive signals such as review_20 and review_12 increase trust, while review_11 and permission_10 decrease it. This balance indicates that users integrate both reputational and risk considerations, granting trust only when favorable cues substantially exceed perceived privacy risks—demonstrating a more deliberate trust formation process.

To evaluate feature reliability, we analyze the standard deviation of LIME attributions for each feature (Fig. 3). Features such as review_11, review_17, and review_19 display high variability, indicating that their influence on model predictions is context-dependent and unstable, varying considerably across different instances. Conversely, features like permission_3 (Identity) and permission_12 (Device & App History) exhibit low variability with consistently negative attributions, indicating they are reliable and stable indicators of untrustworthiness.

Overall, while SHAP reveals overarching trends in feature importance, LIME uncovers the diversity in localized decision-making logic. A key insight is that users often favor apps with high ratings and early positive reviews, even when such apps request permissions typically associated with untrustworthiness (e.g., Camera, Identity). This suggests that visible reputational cues can obscure privacy concerns, leading to over-trust in potentially risky apps. Moreover, the influence of permission-related features varies by context. For example, permission_12 might be accepted for security tools but viewed suspiciously in general-purpose apps.

F. A Human-Subjective Experiment on User Trust

As our findings indicate, Section IV-D, permissions play a critical role in shaping app trustworthiness. While high ratings and positive reviews are commonly seen as indicators of reliability, they alone do not provide a complete picture of an app's trustworthiness. To gain a deeper understanding of this from a human perspective, we conduct a human-subjective test in which we survey 51 users to investigate their priorities when selecting trust factors before downloading apps. All the participants ranged in age from 20 to 50 years and had diverse academic and professional backgrounds. They are Saudi Arabia residents, native Arabic speakers, and regular mobile

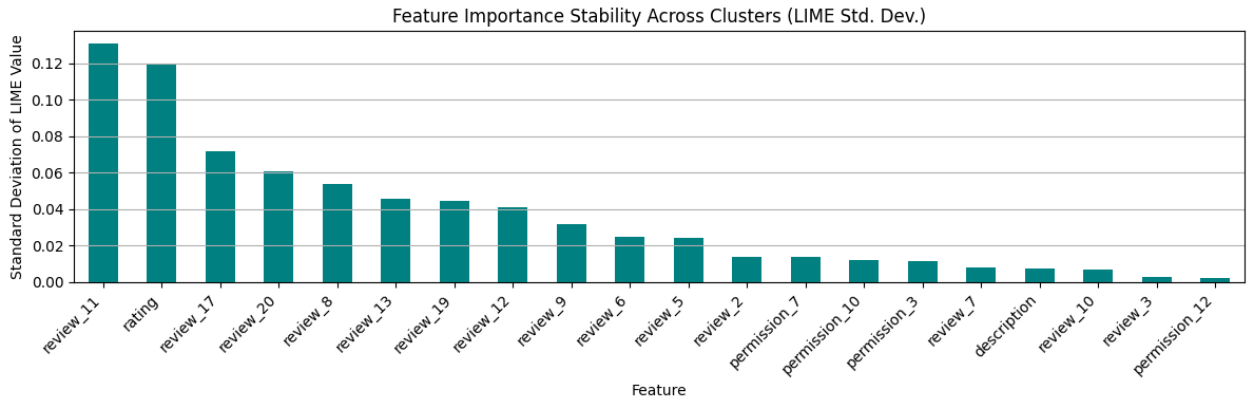

Fig. 3. Feature importance stability across clusters.

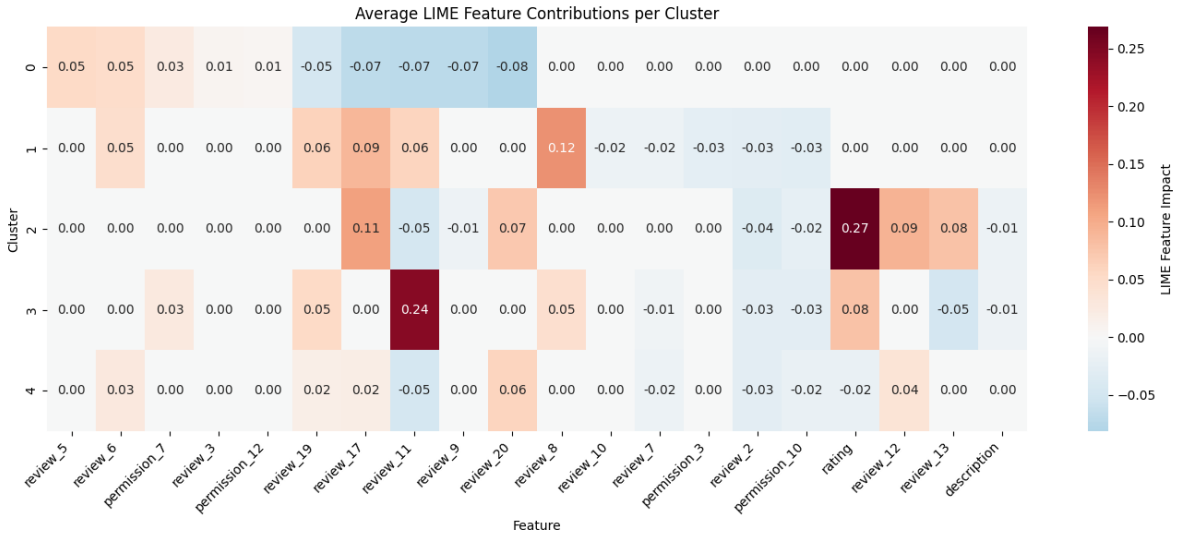

Fig. 4. Average LIME feature contributions per cluster. The heatmap shows the average LIME feature attributions across five clusters. Each row represents a cluster, and each column a feature. Red cells indicate features that positively contribute to trust predictions, while blue cells denote features that negatively influence trust.

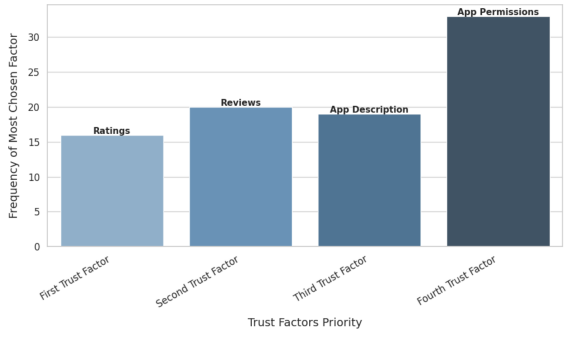

Fig. 5. User Priorities in App Trust: Order of Trust Factors (First to Fourth), where the sequence of trust factors, indicating that the first trust factor is the most important, followed by the others in order of importance.

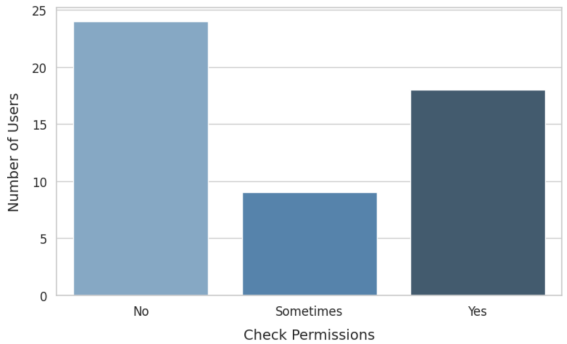

Fig. 6. User responses to checking app permissions before downloading.

device users, either iPhone or Android. They are general users with varying levels of digital experience, which allowed us to gather feedback that reflected real-life experiences.

The results, as shown in Fig. 5, reveal that app permissions emerged not as the most important factor, with 35 users selecting it as their fourth priority (after factors like ratings, reviews, and app descriptions). This finding indicates that while app

permissions are still a significant factor for app trustworthiness, they are not the first factor users prioritize when downloading apps. This indicates that a broader awareness campaign is needed to emphasize the importance of permissions, highlighting their role in ensuring user security and privacy. A more comprehensive understanding of permission management could potentially shift users' priorities, making it a more prominent factor in their trust evaluation process.

Further analysis of the question "Do Users Check App Permissions Before Download?" (Fig. 6) reveals a divided response of the 51 participants, 24 users indicate that they do not check app permissions, while 9 report occasionally checking, and 18 state that they always check permissions. This reveals that a significant portion of users does not consistently check app permissions, which indicates a lack of awareness or concern regarding potential privacy and security risks associated with app permissions.

Overall, in alignment with recent SHAP analysis, our findings indicate that certain app permissions can negatively impact trust, even if an app boasts high ratings or favorable reviews. Therefore, transparent permission management and clear communication regarding the purpose of these permissions are vital for fostering user trust. This combination of user feedback, careful permission handling, and an app's overall reputation forms a more comprehensive and accurate understanding of its trustworthiness, highlighting that trust is shaped by a balance of factors, including app permissions.

V. Limitations

While the constructed dataset of 246 finance-related mobile applications provided valuable insights into user trust and app permissions, its relatively small size presents a potential limitation for machine learning generalization. The modest number of samples may restrict the model's ability to capture broader behavioral variability across different app categories and user contexts.

The findings of this study apply most directly to Arabic-language financial applications used in the Saudi context. Generalization to other domains (such as non-financial app categories), other languages, or other regional markets, including other Gulf or broader Arabic-speaking populations with different regulatory frameworks and cultural contexts, would require further validation. The methodological framework itself, however, combining permission analysis, review embeddings, and explainable AI techniques, is transferable, and we view replication across additional markets and domains as an important direction for future work.

VI. Conclusion

The proposed TAD-Saudi framework provides a significant step forward in the evaluation of mobile app trustworthiness by bridging interpretability and performance in machine learning–based detection. Beyond its technical contribution, the study offers practical implications for multiple stakeholders. For app developers, TAD-Saudi serves as an interpretable benchmark to identify risky permission patterns and improve transparency in app design. For regulators and cybersecurity authorities, it enables data-driven policy formulation and supports more reliable certification processes for financial and e-service applications. For end-users, it enhances awareness of privacy risks and promotes trust in secure digital ecosystems aligned with Saudi Vision 2030.

Future research will expand the dataset to include a larger and more diverse range of applications across different app categories and regions. Further investigations could explore cross-country validation to assess generalizability in other Gulf and international contexts, as well as integration with real-time app store monitoring systems to enable dynamic trust evaluation. These directions aim to strengthen the scalability, robustness, and real-world applicability of the TAD-Saudi framework.

Author Contributions

Raed was responsible for the methodology, analysis, writing, and supervision of the work. Maryam contributed to data curation and editing parts of the manuscript. All authors reviewed the manuscript.

Funding

This research is supported by a grant (No. $CRPG-25-3082$) under the Cybersecurity Research and Innovation Pioneer grant, provided by the National Cybersecurity Authority (NCA) in the Kingdom of Saudi Arabia.

Competing Interests

The authors declare that they have no competing interests.

Data Availability

The data is available at the following link: https://github.com/maryamghamdi/TAD_Saudi

Ethical Approval

This study has received ethical approval from the Saudi Electronic University under approval number SEUREC-4648. All procedures involving human participants were conducted in accordance with the ethical standards of the institutional research committee and with the Helsinki Declaration.

References

[1] M. Alrizq and A. Alghamdi, "Customer satisfaction analysis with saudi arabia mobile banking apps: a hybrid approach using text mining and predictive learning techniques," *Neural Computing and Applications*, vol. 36, no. 11, pp. 6005–6023, 2024.

[2] R. Alharbi, S. Chan-Olmsted, H. Chen, and M. T. Thai, "Deep learning framework with multi-perspective social behaviors for vaccine hesitation," *Social Network Analysis and Mining*, vol. 14, no. 1, p. 140, 2024.

[3] S. M. Habib, N. Alexopoulos, M. M. Islam, J. Heider, S. Marsh, and M. Müehlhäeuser, "Trust4app: automating trustworthiness assessment of mobile applications," in *2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE)*. IEEE, 2018, pp. 124–135.

[4] M. Kuehnhausen and V. S. Frost, "Trusting smartphone apps? to install or not to install, that is the question," in *2013 IEEE International Multi-Disciplinary Conference on Cognitive Methods in Situation Awareness and Decision Support (CogSIMA)*. IEEE, 2013, pp. 30–37.

[5] O. Akgul, S. T. Peddinti, N. Taft, M. L. Mazurek, H. Harkous, A. Srivastava, and B. Seguin, "A decade of privacy-relevant android app reviews: large scale trends," in *Proceedings of the 33rd USENIX Conference on Security Symposium*, ser. SEC '24. USA: USENIX Association, 2024.

[6] P. Nema, P. Anthonysamy, N. Taft, and S. T. Peddinti, "Analyzing user perspectives on mobile app privacy at scale," in *2022 IEEE/ACM 44th International Conference on Software Engineering (ICSE)*, 2022, pp. 112–124.

[7] Ö. Turgut, M. E. Ayhan, Ö. Coşkun, İ. Kök, and S. Özdemir, "Trustworthy and interpretable machine learning models for financial fraud detection," in *2025 International Conference on Smart Applications, Communications and Networking (SmartNets)*. IEEE, 2025, pp. 1–6.

[8] Q. Xu, Y. Liao, Q. Li, J. Zhang, Z. Song, L. Wang, and X. Yuan, "Shap-based interpretable models for credit default assessment using machine learning," in *2024 14th International Conference on Software Technology and Engineering (ICSTE)*. IEEE, 2024, pp. 213–217.

[9] H. Harkous, S. T. Peddinti, R. Khandelwal, A. Srivastava, and N. Taft, "Hark: A deep learning system for navigating privacy feedback at scale," in *2022 IEEE Symposium on Security and Privacy (SP)*, 2022, pp. 2469–2486.

[10] M. Hatamian, J. Serna, and K. Rannenberg, "Revealing the unrevealed: Mining smartphone users privacy perception on app markets," *Computers & Security*, vol. 83, pp. 332–353, 2019.

[11] T. Rani, S. S. Sakthy, P. Kalaichelvi, P. S, and V. S, "Fake app detection using sentiment analysis," in *2023 Intelligent Computing and Control for Engineering and Business Systems (ICCEBS)*, 2023, pp. 1–6.

[12] H. Zaki, M. Saad, and M. R. Rasheed, "A robust machine learning framework for fraudulent mobile app detection," *VFAST Transactions on Software Engineering*, vol. 12, no. 4, pp. 27–36, 2024.

[13] Google, "Google play store," 2025, accessed: 2025-01-07. [Online]. Available: https://play.google.com/

[14] M. A. Khder, "Web scraping or web crawling: State of art, techniques, approaches and application." *International Journal of Advances in Soft Computing & Its Applications*, vol. 13, no. 3, 2021.

[15] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 2016.

[16] S. R. Safavian and D. Landgrebe, "A survey of decision tree classifier methodology," *IEEE transactions on systems, man, and cybernetics*, vol. 21, no. 3, pp. 660–674, 1991.

[17] L. E. Peterson, "K-nearest neighbor," *Scholarpedia*, 2009.

[18] L. Breiman, "Random forests," *UC Berkeley TR567*, 1999.

[19] H. Zhang, A. C. Berg, M. Maire, and J. Malik, "Svm-knn: Discriminative nearest neighbor classification for visual category recognition," in *CVPR*, 2006.

[20] S. Ö. Arik and T. Pfister, "Tabnet: Attentive interpretable tabular learning," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 35, no. 8, 2021, pp. 6679–6687.

[21] A. Das and P. Rad, "Opportunities and challenges in explainable artificial intelligence (xai): A survey," *arXiv preprint arXiv:2006.11371*, vol. abs/2006.11371, 2020.

[22] S. M. Lundberg, G. Erion, H. Chen, A. DeGrave, J. M. Prutkin, B. Nair, R. Katz, J. Himmelfarb, N. Bansal, and S.-I. Lee, "From local explanations to global understanding with explainable ai for trees," *Nature machine intelligence*, vol. 2, no. 1, pp. 56–67, 2020.

[23] M. T. Ribeiro, S. Singh, and C. Guestrin, "" why should i trust you?" explaining the predictions of any classifier," in *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, 2016, pp. 1135–1144.

[24] W. Antoun, F. Baly, and H. Hajj, "Arabert: Transformer-based model for arabic language understanding," in *LREC 2020 Workshop Language Resources and Evaluation Conference 11–16 May 2020*, p. 9.

[25] D. Shah, Z. Y. Xue, and T. M. Aamodt, "Label encoding for regression networks," *arXiv preprint arXiv:2212.01927*, 2022.

[26] K. Potdar, T. S. Pardawala, and C. D. Pai, "A comparative study of categorical variable encoding techniques for neural network classifiers," *International journal of computer applications*, vol. 175, no. 4, pp. 7–9, 2017.

[27] L. Li and H.-T. Lin, "Ordinal regression by extended binary classification," *Advances in neural information processing systems*, vol. 19, 2006.

[28] J. H. Friedman, "Greedy function approximation: a gradient boosting machine," *Annals of statistics*, pp. 1189–1232, 2001.

[29] J. Berkson, "Application of the logistic function to bio-assay," *Journal of the American statistical association*, vol. 39, no. 227, pp. 357–365, 1944.

[30] M. A. Hearst, S. T. Dumais, E. Osuna, J. Platt, and B. Scholkopf, "Support vector machines," *IEEE Intelligent Systems and their applications*, vol. 13, no. 4, pp. 18–28, 1998.

[31] P. Domingos and M. Pazzani, "On the optimality of the simple bayesian classifier under zero-one loss," *Machine learning*, vol. 29, pp. 103–130, 1997.

[32] K. Degirmenci, "Mobile users' information privacy concerns and the role of app permission requests," *International Journal of Information Management*, vol. 50, pp. 261–272, 2020.

[33] H. Sällberg, S. Wang, and E. Numminen, "The combinatory role of online ratings and reviews in mobile app downloads: an empirical investigation of gaming and productivity apps from their initial app store launch," *Journal of Marketing Analytics*, vol. 11, no. 3, pp. 426–442, 2023.

[34] J. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Statistics*, vol. 5. University of California press, 1967, pp. 281–298.