Baltic J. Modern Computing, Vol. 13 (2025), No. 1, pp. 252–289
https://doi.org/10.22364/bjmc.2025.13.1.14

Real-Time Phone Fraud Detection and Prevention Based on Artificial Intelligence Tools

Roberts OĻEIŅIKS, Darja SOLODOVŅIKOVA

Faculty of Computing, University of Latvia, Riga, Latvia

olein.roberts@gmail.com, darja.solodovnikova@lu.lv

ORCID 0009-0009-6867-1357, ORCID 0000-0002-5585-2118

Abstract. Telephone fraud poses significant threats to telecommunications network users, causing both financial loss and emotional stress. The aim of this study was to develop and evaluate a real-time telephone fraud detection and prevention system, based on the principle of phone conversation content analysis. The study included an empirical study to select optimal AI tools for system implementation. A system prototype was developed, integrating the selected automatic speech recognition tool and large language model with a specific prompt, to analyze phone conversation content in real-time. The system's effectiveness was evaluated in a simulated environment reflecting real-time conditions, using an expanded dataset with various fraud scenarios and languages.

The results indicate high classification effectiveness of the system, achieving an accuracy of 90,4% and a 91,2% F1 score, indicating the system's efficacy in real-time telephone fraud detection. The prevention rate reached 69,8%, demonstrating the system's potential in real-time telephone fraud prevention

Keywords: real-time telephone fraud detection, artificial intelligence (AI), phone conversation content analysis, real-time telephone fraud prevention, large language model (LLM), automatic speech recognition (ASR)

1 Introduction

The evolution of telecommunications technology has significantly transformed daily life by enabling rapid and efficient communication both personally and professionally. Telecommunications have not only made the exchange of calls and messages quicker and more convenient but also opened new avenues for remote work and education, allowing people to communicate and collaborate irrespective of their physical location.

However, despite many benefits, telecommunications technology has also introduced new security challenges, including telephone fraud. This type of fraud, which aims to generate illegal income using telecommunications, presents significant threats affecting all network users, impacting individual consumers and legal entities such as mobile network operators. Although protecting mobile network operators from fraudsters is crucial, this research primarily focuses on individual end-users, including private individuals and company employees, who may encounter phone fraudsters daily. Studies highlight that while there are strategies to combat fraud targeting mobile network operators (Sahin et al., 2017; Trapiņš, 2015), individual users are still vulnerable, as shown by telephone fraud statistics.

Surveys by the European Commission in 2020 (WEB, a) estimated that, over two years, European citizens lost EUR 24 billion EUR due to fraud, with 28% of these fraud cases conducted via telephone. In 2023, the U.S. Federal Trade Commission reported that telephone fraud cost consumers approximately 850 million USD (WEB, b). In Latvia, the financial industry association noted that phone scams are challenging to prevent and phone fraud cases resulted in losses of 5,5 million EUR for Latvian citizens in 2023 alone (WEB, c).

Official statistics on telephone fraud do not fully capture the problem's scope. Victims often do not report their experiences, out of shame or fear, suggesting that the true financial losses could be considerably higher. The psychological and emotional toll on victims of telephone fraud, including stress, fear, anger, and psychological trauma, indicates a deeper impact on both individual and societal levels beyond mere financial losses (WEB, a,d). Overall, the statistics not only highlight the problem's relevance but also the potential need for new and effective protection methods in this field.

The structure of this paper is organized as follows: Section 2 outlines the research aim, establishing the study's primary objectives and framework. Section 3 covers related work, reviewing existing literature and technologies relevant to phone fraud detection and prevention. Section 4 presents the system prototype design, detailing the architecture of the real-time phone fraud detection and prevention system, which utilizes AI tools such as Automatic Speech Recognition (ASR) and Large Language Models (LLMs). Section 5 presents study, which focuses on choosing the most suitable ASR tool to transcribe conversations, while Section 6 describes the selection process for an LLM to analyze conversations for fraud detection. In Section 7, an empirical evaluation of the integrated system is presented, assessing its accuracy, latency, and real-time capabilities. Section 8 provides a discussion on the findings, and Section 9 concludes the study, summarizing key insights and suggesting directions for future work.

2 Research aim

This study addresses the critical need for a real-time phone fraud detection and prevention system tailored for individual users within telecommunications networks.

The goal of this research is to design and evaluate a full-scale real-time telephone fraud detection and prevention system, which aims to protect individual telecommunications network users from phone fraud. The system operates by analyzing phone conversation content by use of four main components: real-time recording of phone conversations, transcription, conversation content analysis, and user notification. It leverages existing AI tools for automatic speech recognition and large language model (LLM) to handle conversation transcription and content analysis.

The objectives of this research are threefold:

1. **To design an integrated system architecture** that incorporates its components effectively, ensuring seamless functionality and user responsiveness.
2. **To select the most suitable AI tools for transcription and content analysis**, focusing on achieving high accuracy and minimal latency in fraud detection.
3. **To conduct testing and simulations** to evaluate the system's effectiveness in real-world scenarios.

Further, empirical research will be conducted to optimize the selection of AI tools, with a comparative analysis of industry-provided technologies to determine the most effective ASR and LLM solutions. Additionally, the study will utilize a multi-lingual dataset of simulated phone conversations in Latvian, English, and Russian to mirror the linguistic context of Latvia, accompanied by a control group of legitimate phone calls for a comprehensive evaluation of the system's capabilities.

3 Related work

Phone fraud remains a significant and evolving threat to individuals and organizations, necessitating continuous advancements in detection and prevention methods. Existing approaches to combating phone fraud can be broadly categorized into two main categories: social education and technological solutions. Social education initiatives focus on raising public awareness about the characteristics of phone scams, empowering individuals to recognize and avoid potential fraud attempts. While valuable, such methods can be inconsistent, as human judgment is often susceptible to factors like fatigue or stress. This review focuses on technological methods for phone fraud detection, which is essentially a binary classification problem - distinguishing between legitimate and fraudulent calls. Evaluating the effectiveness of phone fraud detection systems involves considering various performance metrics such as accuracy, precision, recall and F1 score.

3.1 Phone call filtering and blocking

Traditionally, methods like blacklisting and filtering have been used to detect phone fraud. Call filtering methods involve marking suspicious numbers, timely blocking, and adding them to blacklists. Filtering and blocking are performed by analyzing Call Detail Records (CDR), which include call metadata — call start time, call end time, caller number, receiver number, call duration, receiver location, call type (outgoing, incoming), etc. In addition to call metadata, number tagging and blocking can be based on previous fraud cases and user reports. Implementations can use traditional data analysis algorithms or machine learning algorithms.

In a 2018 study, the "TouchPal" system was developed, employing machine learning methods and CDR data analysis (Li et al., 2018). The system relies on a substantial user base to create a reputation-based blacklist for effective prevention of fraudulent calls. Users mark suspicious calls, contributing to a blacklist that blocks incoming calls from these numbers. Machine learning models—including random forests, neural networks, support vector machines, and logistic regression—predict fraudulent calls based on 29 features without analyzing call content. Key criteria include call type, duration, time, location, contact information, and historical data. The study reported a 99,99% precision rate and a 90% recall rate (calculated F1 score of 94,74%). However, the study notes that the main drawback is that fraudsters can bypass the system's protection by spoofing the caller ID.

The 2020 study by Xing et al. (Xing et al., 2020) presented a deep learning approach for automatic detection of fraudulent calls by analyzing CDRs. Models used included deep neural networks, convolutional neural networks (CNNs), long short-term memory networks (LSTMs), and stacked denoising autoencoders (SDAEs). The SDAE model achieved over 99% accuracy. However, precision was low (6–11%) due to an imbalanced dataset (8,2 million legitimate calls vs. 8200 fraudulent calls). Given the low precision, the model avoided missing fraudulent calls by often misclassifying legitimate calls as fraudulent, sacrificing precision to ensure a high accuracy.

The 2021 study (Gowri et al., 2021) proposed a machine learning solution using a recurrent neural network (RNN) to detect malicious (fraudulent) phone calls by analyzing a dataset containing CDRs obtained from the Kaggle platform. The proposed solution includes data preprocessing for quality improvement and RNN model training for predictions. The study reported an 87% accuracy in detecting malicious calls, indicating the effectiveness of RNNs in reducing such calls. However, detailed performance metrics like recall, precision, and F1 score were not provided, limiting a comprehensive evaluation of the model's ability to distinguish between legitimate and fraudulent calls.

3.2 Caller ID spoofing

Filtering and blacklisting approaches are vulnerable to caller ID spoofing, which allows fraudsters to bypass protection mechanisms. Caller ID spoofing enables fraudsters to conceal their identity by altering their phone number to any other number. This is possible because telecommunication networks are not fully protected against such interference (Song et al., 2014). Initially, telecommunication systems were designed assuming that caller ID information would be authentic, unique, and authorized. However, technological advancements have allowed fraudsters to exploit system vulnerabilities, using software and devices that manipulate caller ID data. This manipulation allows fraudsters to generate any outgoing phone number, including ones similar to the victim's own number, contacts from the victim's phonebook, emergency numbers, etc. (WEB, e), making it harder to identify fraudulent calls and increasing the likelihood that the victim will answer. Caller ID spoofing reduces the effectiveness of filtering and blacklisting systems because these systems rely on historical data and user reports, which become useless when caller IDs are spoofed.

The 2020 study (Pandit et al., 2020) describes the development of the "RobocallGuard" system—a virtual assistant (VA) designed to limit fraudulent and other unwanted calls using automatic call filtering. Motivated by the problem of caller ID spoofing, "RobocallGuard" offers protection against the negative effects of phone number manipulation. The system uses audio analysis and voice recognition, serving as a protective layer between the caller and the recipient by filtering fraudulent and other unwanted calls using a test similar to a "captcha" but in audio format: the caller must state the recipient's name. If the recipient's name is not correctly provided, the call is not connected and is subsequently blocked by adding it to a blacklist. If a call is received from a blacklisted number, the virtual assistant immediately blocks it; calls from the whitelist are connected without VA intervention. For numbers not in either list, the VA evaluates the call. The study shows a 97,8% recall rate but does not provide precision results. A significant limitation of the system is its inability to protect against targeted attacks where the caller knows the recipient's name. The VA's protection can be bypassed by using common names (e.g., "Peter", "Anna") or crafting phrases that might mislead the system by exploiting vulnerabilities in the voice recognition module.

3.3 Call content analysis

Traditional methods based largely on blacklisting are no longer sufficiently effective because such protection often does not offer comprehensive defense; fraudsters have found ways to bypass these systems by spoofing caller IDs. Moreover, if traditional methods allow a fraudster to establish direct contact with the victim, they are ineffective against social engineering techniques used during the call to manipulate the victim into facilitating fraud. The increasing use and availability of caller ID spoofing software have created new challenges in preventing phone fraud. Therefore, developing and implementing new methods based on analyzing the content of phone calls has become increasingly important. Such methods allow detecting phone fraud by analyzing the caller's speech content, sentiment, or other elements that may indicate fraudulent intent, such as social engineering techniques like imposing decision-making pressure and urgency. Thus, analyzing call content offers a significant step toward more effective detection of phone fraud, capable of addressing the new challenges posed by caller ID spoofing, social engineering techniques, and verbal manipulation during calls.

The early beginnings of call content analysis can be seen in the 2016 study (Sawa et al., 2016), where a method was developed using Natural Language Processing (NLP) to identify social engineering threats in text format, such as emails or social media chats. The method analyzes dialogue by focusing on questions and commands, comparing them to a predefined "blacklist" of topics. This list consists of forbidden actions toward specific objects and is tailored to specific situations. For example, an attacker wants the victim to manipulate a network device by saying "reset the router." This statement is identified as a potential threat because "reset" describes an action and "router" describes an object, matching an entry in the blacklist of forbidden topics. Due to the match, further communication is blocked, and a warning is sent to the victim. The study demonstrated a low false positive rate, 100% precision, and a 60% recall rate (calculated F1 score of 75%). A potential problem with this method is its dependence on the blacklist, which requires regular and manual updates—a time-consuming process that may not keep pace with evolving fraud tactics.

In contrast, the 2018 study (Zhao et al., 2018) fully presented a new approach to phone fraud detection based on analyzing call content, moving away from traditional methods. The study collected 12368 examples of fraudulent call descriptions from Chinese internet platforms like Sina Weibo and Baidu to create a dataset. The authors used NLP methods to extract features from texts, creating detection rules and then training a model to perform text analysis. The model's effectiveness is highlighted by high prediction and performance metrics in the selected dataset—accuracy 98,53%, precision 97,97%, recall 98,25%, and F1 score 98,11%. An Android application was developed to implement the detection rules, offering real-time fraud detection without requiring user data upload to a server, thus ensuring privacy. To test the application's performance, experiments were conducted where participants were asked to read fraud dialogues. Out of 15 dialogues based on online resources and victim descriptions, 10 were in Mandarin, and 5 in dialects typical for China. The application detected 90% of fraud attempts in Mandarin but only 40% in dialects, indicating limitations in speech recognition technology. Despite the model's effectiveness, the study acknowledges limitations such as insufficient data volume and low accuracy of local speech recognition technology. It is important to note that due to Google's policy changes on April 6, 2022, two-way call recording was restricted on Android (WEB, f), making it difficult to assess whether the call recording method implemented in the study is still feasible and whether the overall system is operational.

The 2021 study (Derakhshan et al., 2021) developed a method to recognize social engineering techniques during phone calls using the innovative concept of "scam signatures", similar to malware signatures. Scam signatures are defined as sets of speech acts that form fraud indicators, serving as basic tools for a fraudster to achieve their goals. The study developed the Anti-Social Engineering Tool, which uses word and sentence embeddings from NLP to determine if scam signature elements are present in a conversation. The method demonstrates 100% precision, and a 71,4% recall rate (calculated F1 score of 83%).

The 2023 study (Hong et al., 2023) explored the use of LSTM model architecture in identifying fraudulent calls using call content. A balanced dataset of 100 call scenario recordings (50 fraudulent and 50 legitimate) was used, mainly obtained from YouTube. Call recordings were converted to text for further analysis. The model demonstrated moderate effectiveness, with an accuracy of 85,6%, precision of 60%, recall of 32%, and an F1 score of 41%. Despite promising accuracy, the overall system effectiveness is lower compared to other methods when evaluated using the F1 score. The study acknowledges limitations related to the dataset size and recommends improvements, including multilingual support, to enhance detection effectiveness.

3.4 Conclusion

Recognizing the limitations of current approaches, which primarily focus on fraud detection, this research contributes not only by developing a novel real-time system that integrates existing AI-based tools and methods for phone fraud detection but also by prioritizing the evaluation of its prevention capabilities, a crucial aspect overlooked in the literature.

4 System prototype design

This section presents the design and architecture of the proposed real-time phone fraud detection and prevention system prototype. The system leverages a content analysis approach, utilizing AI tools such as Automatic Speech Recognition (ASR) and Large Language Models (LLMs) to identify and mitigate fraudulent calls in real-time.

4.1 System architecture

The system prototype is built upon four main components that work together to provide real-time fraud detection and prevention:

1. **Real-time phone call recording:** capture the audio stream of the phone conversation.
2. **Real-time call transcription:** convert the recorded audio into text using ASR.
3. **Real-time conversation content analysis:** analyze the transcribed text using LLMs to identify fraudulent patterns and tactics.
4. **Fraud notification:** alert the call recipient about detected fraud attempts.

The system operates by continuously monitoring the conversation content and triggering an alert if fraudulent activity is detected. Upon detection, the system can automatically terminate the call to prevent further interaction with the fraudster and send an SMS notification to the call recipient.

4.2 System design and modularity

The system prototype is built upon a modular architecture that integrates external services like Twilio, Google, OpenAI and nGrok, allowing for flexibility and adaptability in incorporating alternative or improved AI tools in the future. The system also incorporates a custom-developed middleware to manage data flow, audio processing, redaction, and user notifications, seamlessly connecting the various components and external services.

A key design consideration was ensuring compliance with data privacy regulations, particularly GDPR. The system utilizes a call forwarding mechanism to seamlessly connect incoming calls to the recipient's personal phone number while simultaneously recording the conversation through Twilio. To address GDPR requirements, the system can be configured to inform callers about the recording and obtain their consent before proceeding, aligning with legal requirements for lawful call recording. Additionally, personal identifiable information (PII) is redacted from transcripts before they are sent to the LLM for analysis, further protecting user privacy.

The real-time phone fraud detection and prevention system prototype outlines an innovative approach to addressing the evolving phone fraud threat by leveraging existing tools to provide AI-powered conversation content analysis. The detailed architectural design of the phone fraud detection and prevention system prototype is shown below in Figure 1. The phone call monitoring process operates continuously until either phone fraud is detected or the phone call ends. In a real-world deployment, the middleware is designed to be hosted on a cloud server, serving as the central hub between the telecommunications network and external services (e.g., Twilio, OpenAI, and Google). In this configuration, incoming calls are first forwarded via Twilio to the middleware, which

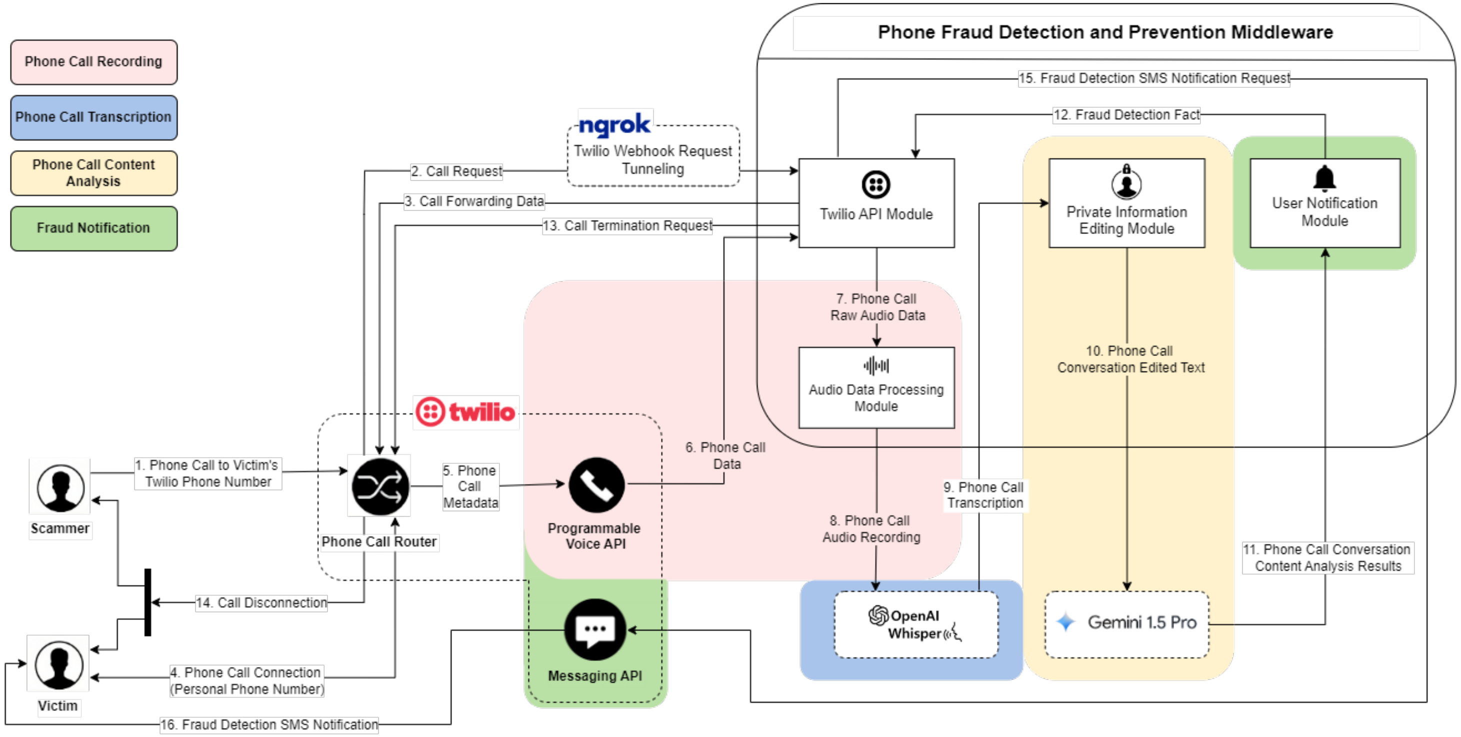
Phone Call Recording
Phone Call Transcription
Phone Call Content Analysis
Fraud Notification
Phone Fraud Detection and Prevention Middleware
15. Fraud Detection SMS Notification Request
12. Fraud Detection Fact
ngrok
Twilio Webhook Request Tunneling
2. Call Request
3. Call Forwarding Data
13. Call Termination Request
Twilio API Module
Private Information Editing Module
User Notification Module
7. Phone Call Raw Audio Data
10. Phone Call Conversation Edited Text
Audio Data Processing Module
twilio
6. Phone Call Data
1. Phone Call to Victim's Twilio Phone Number
5. Phone Call Metadata
Scammer
Phone Call Router
Programmable Voice API
9. Phone Call Transcription
8. Phone Call Audio Recording
11. Phone Call Conversation Content Analysis Results
14. Call Disconnection
OpenAI Whisper
Gemini 1.5 Pro
Messaging API
Victim
4. Phone Call Connection (Personal Phone Number)
16. Fraud Detection SMS Notification

Fig. 1. Phone Fraud Detection and Prevention System Prototype Architecture Design

then manages data flow, audio processing, redaction, and user notifications. Communication traverses multiple network hops—from the mobile network to Twilio, then from Twilio to the middleware, and finally from the middleware to the external APIs.

This prototype serves as the foundation for subsequent sections, which detail the selection and evaluation of specific AI tools and the overall system performance.

4.3 Phone call recording

Workflow The call recording component is initiated when a user receives a phone call. The process involves the following steps (refer to Figure 1 for visual representation):

1. **Call initiation:** The caller dials the recipient's Twilio phone number.
2. **Webhook request:** Upon receiving the call, Twilio's call routing mechanism automatically generates an HTTP request to a pre-configured webhook address. This webhook address, associated with the recipient's Twilio number, acts as an interface between the call and the system's middleware.
3. **Call forwarding:** The Twilio API module within the middleware stores the recipient's personal phone number. Upon receiving the webhook request, it instructs Twilio to forward the call to the recipient's personal number.
4. **Call connection:** Twilio's call routing mechanism forwards the call to the recipient's personal phone number, establishing the connection between the caller and recipient.
5. **Metadata forwarding:** Once the call is connected, the call routing mechanism sends the call metadata to Twilio's Programmable Voice API to initiate call recording.
6. **Call data transmission:** Twilio's Programmable Voice API continuously sends call data, including metadata and raw audio, to the middleware's Twilio API module.
7. **Audio data preparation:** The middleware processes the raw audio data, separating and sequentially ordering the audio streams for the caller and recipient. Once approximately 15 seconds of audio data is accumulated, it is forwarded to the audio data processing module.
8. **Audio recording preparation:** The audio data processing module converts the raw audio data from Twilio's format (8-bit PCM mono) to MP3 format, which is compatible with the transcription tool. The module also merges the caller and recipient audio streams into a single MP3 file, which is then sent to the call transcription module.

4.4 Phone call transcription

The system utilizes OpenAI's "Whisper" speech recognition tool to transcribe the recorded phone calls into text. Whisper was selected based on empirical research comparing the performance of various speech recognition tools for the target languages (Latvian, English, and Russian). This research is discussed in subsequent chapter "5. Transcription tool selection".

Workflow

9. **Transcription:** The phone call recording, in MP3 format, is sent to OpenAI's Whisper API. Whisper processes the audio and generates a text transcript of the conversation, which is returned as the API response.

4.5 Conversation content analysis

The Google Gemini 1.5-Pro Large Language Model (LLM) is employed to analyze the transcribed conversation content and detect potential phone fraud. This LLM was chosen based on empirical research (discussed in chapter "6. Content analysis tool selection") evaluating the effectiveness of various LLM tools and parameter configurations across the target languages.

Workflow

10. **Text redaction:** The transcribed text is first processed by the private information redaction module. This module replaces personal identifiable information (PII), such as names and phone numbers, with placeholders to protect user privacy and comply with GDPR regulations.
11. **Content analysis:** The redacted transcript is sent to the Google Gemini 1.5-Pro LLM via its API for analysis. The LLM assesses the content and returns a judgment indicating whether the conversation excerpt exhibits characteristics of a phone fraud attempt. The results of the content analysis are forwarded to the user notification module.

4.6 Fraud notification

The system utilizes SMS notifications via Twilio's Messaging API to alert the call recipient about detected fraud attempts.

Workflow

12. **Fraud detection:** The user notification module receives the content analysis results from the Google Gemini LLM. If fraud is detected, the module immediately informs the Twilio API module.
13. **Call termination request:** Upon receiving the fraud detection alert, the Twilio API module sends a call termination request to Twilio to prevent further interaction between the recipient and the potential fraudster.
14. **Call termination:** Twilio's call routing mechanism terminates the call based on the metadata provided in the termination request.
15. **Notification request:** Simultaneously, the Twilio API module sends an SMS notification request to Twilio's Messaging API.
16. **SMS notification:** Twilio's Messaging API sends an SMS message to the call recipient's personal phone number, informing them of the detected fraud attempt.

Further research and experimental simulations in subsequent chapters will provide a deeper understanding of the system's operation and its ability to effectively detect and prevent phone fraud in real-time.

5 Transcription tool selection

In this section, an empirical study is conducted to select the most optimal transcription tool for use in a phone fraud detection and prevention system. The effectiveness of a real-time phone fraud detection and prevention system hinges on the ability to accurately and swiftly transcribe spoken conversations into text. This transcription process is critical, as it directly impacts the subsequent analysis and detection of fraudulent activities and indicators within the conversation. Inaccurate transcriptions can distort the content or context of conversations, impeding the system's ability to correctly identify fraudulent patterns. Similarly, delays in transcription can postpone the detection and prevention measures, increasing the likelihood of successful fraud attempts.

Despite the abundance of Automatic Speech Recognition (ASR) tools available today, there is a notable scarcity of recent benchmarking studies that evaluate their performance, especially in the context of real-time systems and for languages less commonly supported, such as Latvian. The rapid pace of technological advancement in ASR tools means that existing studies can quickly become outdated, underscoring the necessity for up-to-date evaluations.

Previous research has attempted to benchmark ASR systems. For instance, a study conducted in 2020 (Filippidou and Moussiades, 2020) evaluated several ASR systems, including Google, IBM Watson, and Wit.ai, using metrics such as Word Error Rate (WER), Hypothesis Error Rate (Hper), and Reference Position-Independent Word Error Rate (Rper). WER is a widely used metric for assessing the accuracy of speech recognition systems, which is calculated by comparing the original text with the transcribed text to determine the proportion of errors. The 2020 study found that Google's ASR system outperformed others, ranking first in terms of WER.

In contrast, a more recent study in 2023 (Ferraro et al., 2023) compared open-source ASR tools with commercial ones, using seven widely adopted datasets such as LibriSpeech and Common Voice. The open-source tools included Conformer, HuBERT, SpeechBrain, WhisperX, and SpeechStew, while the commercial tools were Amazon Transcribe, Microsoft Azure, Google, and IBM Watson. The results indicated that commercial ASR tools generally offered better performance in terms of accuracy and speed, although performance varied depending on the dataset. Notably, Amazon Transcribe and Microsoft Azure outperformed Google, suggesting a shift in the competitive landscape of ASR tools since the 2020 study.

Moreover, several unofficial articles and self-sponsored reports claim superior performance of other commercial ASR tools, such as DeepGram (WEB, g), AssemblyAI (WEB, h), Speechmatics (WEB, i), and RevAI (WEB, j). These claims often highlight impressive metrics, but the lack of independent verification necessitates an empirical evaluation to objectively assess their performance.

Given the evolving nature of ASR technology and the importance of accurate and rapid transcription in a phone fraud detection system, this part of the research aims to select the most suitable ASR tool in the context of phone fraud detection and prevention by conducting an empirical evaluation of several leading options. The tools initially considered for evaluation were Amazon Transcribe, Microsoft Azure, OpenAI Whisper, DeepGram, AssemblyAI, Speechmatics, and RevAI. These tools were selected based on their prominence in previous studies, claimed performance metrics, and their availability for commercial use.

5.1 Methodology

The selection process was structured as a multi-stage empirical evaluation designed to simulate real-world conditions and focus on parameters critical to the system's performance. The methodology comprised the following stages:

1. Preselection and audio format impact analysis.
2. Benchmarking based on WER.
3. Benchmarking based on latency.

Stage 1: preselection and audio format impact analysis The initial stage comprised a qualitative assessment of language support and automatic language detection, followed by an analysis of how different audio formats impacted performance:

- Preselection: ASR tools were first evaluated based on their support for the three target languages (Latvian, English, and Russian) and their ability to automatically detect and transcribe conversations in the appropriate language without manual input. Tools that did not meet these criteria were excluded from further evaluation.
- Audio Format Impact: For the remaining tools, the effect of audio format (WAV vs. MP3) on transcription accuracy and latency was assessed. Four conversation scenarios (two fraudulent and two legitimate) were selected in each language for testing. WER was calculated for both formats to measure transcription accuracy, and latency was measured by recording the time it took to transcribe 15-second audio segments via API to simulate real-time processing.

Stage 2: benchmarking based on WER In this stage, the tools were evaluated based on their transcription accuracy using sample dataset of 10 scenarios. WER was calculated for each tool in all three languages. Tools that exhibited consistently high WER values across multiple languages were considered less suitable for real-time fraud detection, where high transcription accuracy is critical.

Stage 3: benchmarking based on latency The final stage assessed the latency of the ASR tools that performed well in terms of WER. Latency was measured using same sample dataset of scenarios by sending the audio files in 15-second segments to the ASR tool's API and recording the time taken to return the transcriptions. In a real-time fraud detection system, minimizing latency is essential for timely detection and prevention of fraudulent activities.

5.2 Dataset preparation

To evaluate the ASR tools under conditions reflective of real-world usage, a sample dataset was constructed. The dataset included 30 phone conversation scenarios, with 10 unique scenarios duplicated in each of the target languages—Latvian, English, and

Russian. Each language set comprised 5 fraudulent and 5 legitimate scenarios. More detailed information regarding dataset can be found in integrated system's evaluation chapter "7.1 Methodology".

The audio recordings for these conversation scenarios were generated using high-quality computer-generated voices from the "PlayHT" platform (WEB, k). Utilizing computer-generated voices allowed for precise control over speech parameters, particularly Words Per Minute (WPM), ensuring consistency across recordings. The WPM settings were chosen to reflect natural speaking speeds: 200 WPM for English, 170 WPM for Latvian, and 160 WPM for Russian (Rodero, 2012; Kappen et al., 2024; Bogdanovs, 2018), with a maximum variation of ±5 WPM.

To simulate the audio quality typically encountered in telecommunication networks, the original high-quality recordings were degraded to match standard telecommunication audio formats—8-bit PCM mono uLaw with an 8 kHz sampling rate. This step ensured that the ASR tools would be evaluated under realistic audio quality conditions that mirror actual phone conversations.

The degraded audio files were then converted into both WAV (lossless) and MP3 (lossy) formats.

5.3 Limitations and assumptions

The study acknowledged several limitations. The use of computer-generated voices may not fully capture the nuances of human speech, such as emotional variation or dialect, potentially affecting the accuracy of the transcription. The absence of background noise and telecommunication disruptions in the audio recordings might lead to an overestimation of transcription accuracy in real-world scenarios. Furthermore, latency measurements were conducted in a controlled environment, and real-world network conditions may vary, potentially affecting the system's performance.

The study also relied on several assumptions. It was assumed that all audio recordings were of uniform quality and that network conditions remained stable during testing, which may not always hold true in real-world applications.

5.4 Results

The ASR tool evaluation involved a total of 1439 latency measurements and 96 WER measurements in the first stage (audio format impact analysis), 150 WER measurements in the second stage, and 1615 latency measurements in the third stage.

Stage 1: Preselection and audio format impact analysis All ASR tools selected for the evaluation—Amazon Transcribe, Microsoft Azure, OpenAI Whisper, Speechmatics, RevAI, DeepGram, and AssemblyAI—demonstrated the ability to automatically identify the spoken language. Furthermore, Amazon Transcribe, Microsoft Azure, OpenAI Whisper, Speechmatics and RevAI provided full support for the target languages (Latvian, English, and Russian). However, DeepGram and AssemblyAI lacked support for Latvian and were excluded from further selection.

Based on these qualitative parameters, Amazon Transcribe, Microsoft Azure, OpenAI Whisper, RevAI, and Speechmatics were deemed suitable candidates for further quantitative evaluation.

Converting audio recordings from WAV to MP3 resulted in a substantial reduction in file size, with an average decrease of approximately 87,5%, given that the average conversation length was 4 minutes and 3 seconds. This reduction significantly impacts data transmission times, which is critical for a real-time system.

The transcription accuracy, measured by WER, showed a slight increase when using the MP3 format. The average WER, calculated across all selected ASR tools, increased by only 0,39% (from 5,62% to 6,01%), which is negligible in practical terms. On the other hand, processing latency decreased notably when using MP3. The average latency, calculated across all selected ASR tools, was reduced by 0,89 seconds (from 9,54 seconds with WAV to 8,65 seconds with MP3), representing a meaningful improvement for real-time processing.

Given the significant reduction in latency with only a minimal impact on transcription accuracy, the MP3 format was selected for subsequent evaluation stages. Microsoft Azure, which only supported WAV format at the time of testing, was not included in format evaluation.

Stage 2: Benchmarking based on word error rate Figure 2 presents a box plot illustrating the distribution of WER for each ASR tool in English. OpenAI's tool shows an average word error rate of 1,87%, with 50% of the data points falling between 0,85% and 2,65%. In contrast, Microsoft's tool exhibits a higher average WER of 5,54%, with an outlier at 11,60%. Based on English language recognition, the ASR tools rank as follows: OpenAI, Speechmatics, RevAI, Amazon, and Microsoft.

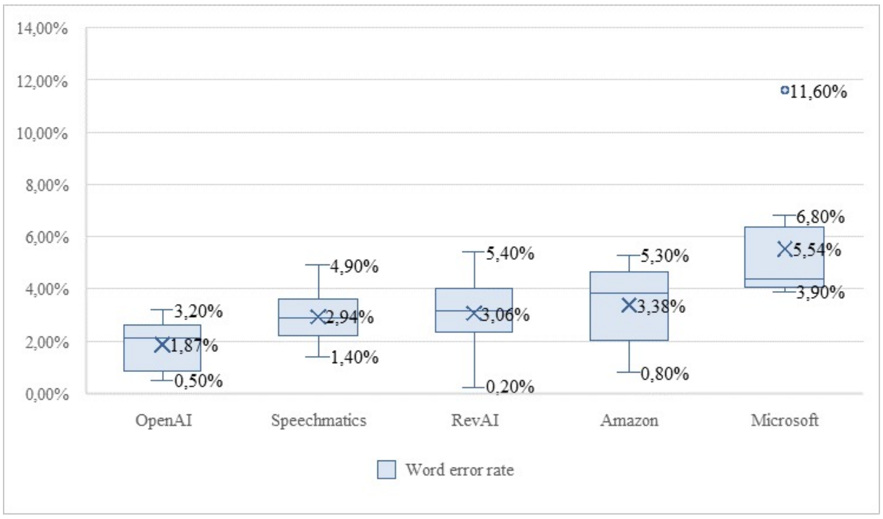

Fig. 2. ASR tool WER benchmarks in English

Figure 3 shows the WER for Latvian. All tools show a relatively similar range, except for Amazon's ASR tool, which exhibits a significantly higher average WER of 22,46%. This indicates that Amazon's accuracy in transcribing Latvian speech is considerably lower than the other tools. Based on Latvian language recognition, the ASR tools rank as follows: Speechmatics, Microsoft, RevAI, OpenAI, and Amazon.

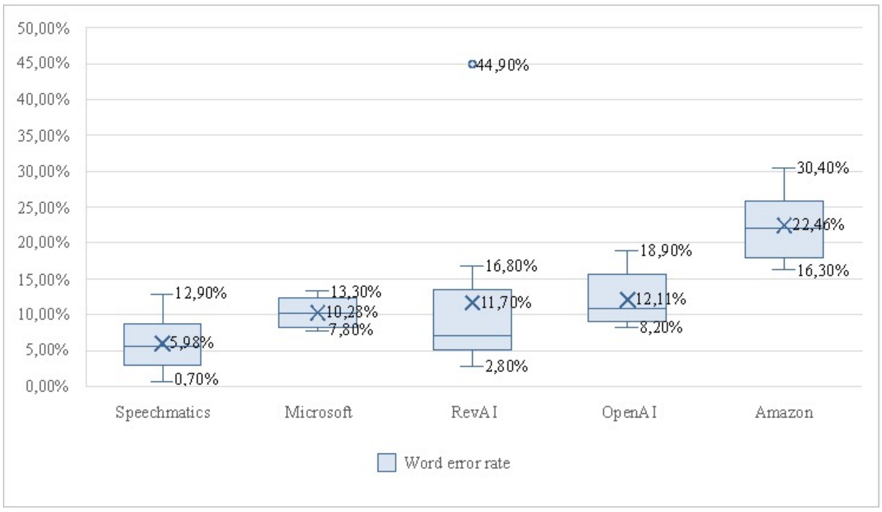

Fig. 3. ASR tool WER benchmarks in Latvian

Figure 4 shows the ASR tools ranked by their average WER in Russian. The ranking for Russian is: OpenAI, RevAI, Amazon, Speechmatics, and Microsoft. The ranking for Russian is: OpenAI, RevAI, Amazon, Speechmatics, and Microsoft.

Analysis of the WER across different target languages revealed that the tools' performance is relatively similar overall, with rankings varying depending on the language. However, Amazon's ASR tool stands out with its significantly higher WER for Latvian, indicating lower accuracy. Due to this lower performance, Amazon's ASR tool was excluded from further selection. The remaining tools proceeded to the next stage were: OpenAI, Speechmatics, RevAI, and Microsoft.

Stage 3: Benchmarking based on latency Processing latency measurements, using 15-second audio fragments, revealed substantial differences among the ASR tools:

- **OpenAI Whisper** demonstrated the lowest average latency at 1,53 seconds across all languages.
- **Microsoft Azure** had an average latency of 6,60 seconds.
- **RevAI** exhibited an average latency of 10,24 seconds.
- **Speechmatics** had the highest average latency at 13,99 seconds.

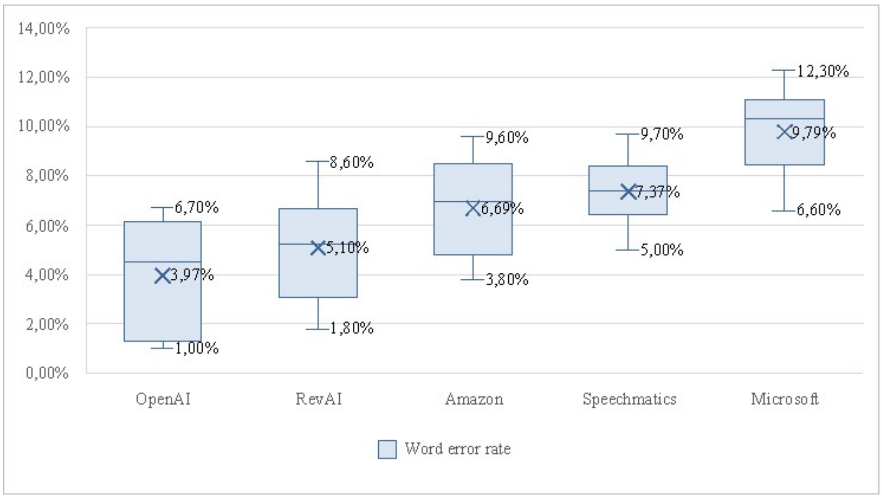

Fig. 4. ASR tool WER benchmarks in Russian

5.5 Conclusion

This study evaluated several ASR tools for real-time phone fraud detection based on transcription accuracy (WER) and latency. While most tools showed similar WER across languages, Amazon ASR underperformed significantly in Latvian. However, the key factor in the final selection was latency.

Based on the comprehensive evaluation, OpenAI Whisper was selected as the optimal transcription tool for integration into the phone fraud detection and prevention system. It offers several advantages:

- **High transcription accuracy** across all target languages, ensuring that the content and context of conversations are accurately captured.
- **Rapid processing speed,** with the lowest latency among the evaluated tools, enabling the system to detect and prevent fraudulent activities promptly.
- **Support for target languages,** including Latvian, English, and Russian, which are essential for the system's applicability in the regional context.
- **Automatic language detection,** allowing the tool to adapt to the language of any given conversation without manual intervention.

The selection of OpenAI Whisper aligns with the system's requirements for accuracy and speed, ensuring that the phone fraud detection and prevention system can operate effectively in real-time.

6 Content analysis tool selection

Selecting an appropriate large language model (LLM) is crucial for the success of a phone fraud detection and prevention system. This chapter presents an empirical study to select the most suitable LLM for analyzing telephone conversations to detect and prevent phone fraud in real-time, necessitating both high accuracy and low latency. This study focuses on commercially available LLMs, assuming they benefit from greater resources for development and training data. Unlike automatic speech recognition (ASR) systems, where comparative evaluations are less common, LLMs have been extensively researched and compared, fueled by rapid advancements in artificial intelligence.

This study evaluates three leading commercial LLMs: Anthropic Claude, Google Gemini, and OpenAI GPT. These companies are recognized as industry leaders, actively comparing their models against each other (Achiam et al., 2023; Team et al., 2023; WEB, l). This study utilizes their most powerful models available at the time of research: Claude-Opus, Gemini 1.5-Pro, and GPT-4. The section describes the research methodology and results, culminating in the selection of the optimal LLM.

6.1 Methodology

This section details the methodology for selecting the LLM, including the prompts, dataset, and study limitations.

The study aims to:

1. Objectively evaluate the LLMs to select the most suitable one for real-time phone fraud detection and prevention.
2. Identify the optimal LLM prompt for this task.

LLM performance depends on both the model and its configuration. The optimal combination will ensure the best results in fraud detection and prevention.

The methodology evaluates three quantitative criteria:

- Response speed (latency): Measures the time taken by the LLM to process a text fragment and respond.
- Classification effectiveness: Evaluates the LLM's ability to detect fraudulent calls using standard classification metrics (accuracy, recall, precision, and F1 score). Effective detection enables timely prevention.
- Classification effectiveness variation: Evaluates the stability and consistency of LLM classification results across multiple runs, accounting for their generative nature and inherent randomness. To assess the variability and statistical significance of the results, the mean, standard deviation, and 95% confidence interval of the performance metrics will be calculated.

All three criteria—classification effectiveness, its consistency, and response speed—are critical for a real-time phone fraud detection and prevention system. Accurate and consistent fraud identification is essential to minimize financial loss and protect fraud victims, while a fast response time is necessary to prevent fraud from escalating. These criteria will be evaluated as a unit to determine the overall suitability of an LLM for real-time operation.

6.2 Study dataset

This study utilizes the same dataset (audio recordings) as the previous chapter on ASR tool selection (see section "5.2 Dataset preparation"). However, this study analyzes the transcribed texts from those recordings, generated by the best-performing ASR tool (OpenAI Whisper), as input for the LLMs. (For a full description of the dataset, see integrated system's evaluation section "7.1 Methodology", subsection "Study dataset").

Each scenario's text is divided into 15-second fragment. This division corresponds to the real-time system's operation, where conversations would be analyzed in 15-second intervals to provide enough context. The fragment length in words varies depending on the speech rate used during scenario generation for particular language. For example, in Latvian, with a speech rate of 170 words per minute, a 15-second fragment contains approximately 43 words.

To ensure anonymity and privacy during conversation content analysis, personally identifiable information (e.g., names and numbers) was removed from the transcribed texts and replaced with placeholders.

6.3 Latency evaluation methodology

This section describes the methodology for evaluating LLM latency, given the influence of prompts, LLM response speed is critical for real-time fraud detection. Latency was measured by sending transcribed conversation text fragments to each LLM via their public API and precisely timing the response. Different LLM and prompt combinations were tested to assess how prompt length and complexity affect response time. The results were analyzed, including classification performance and variation, to determine the optimal balance between speed and accuracy.

6.4 Classification effectiveness evaluation methodology

This section details the methodology for evaluating LLM classification effectiveness in phone fraud detection and prevention. The evaluation involves two stages: assessing detection effectiveness and assessing prevention potential by identifying fraud sufficiently early in the conversation.

Phone fraud detection evaluation

1. **LLM and prompt combinations:** Three LLMs (Claude-Opus, Gemini 1.5-Pro, GPT-4) and three different prompts focused on fraud detection were evaluated, analyzing all 9 possible combinations.
2. **Continuous classification:** Each scenario's text fragments were sequentially analyzed by the LLMs to detect fraud indicators. Analysis continued until the text ended or fraud was detected.
3. **Performance metric calculation:** Classification results were compared to manual annotations to calculate accuracy, recall, precision, and F1 score for each LLM and prompt combination.

4. **Performance variation calculation:** To mitigate randomness, each scenario was analyzed 5 times with each LLM and prompt combination. To obtain more deterministic results, the LLM temperature parameter was set to 0. The mean, standard deviation, and 95% confidence interval of performance metrics were calculated.
5. **Result analysis:** Performance metrics were compared across LLMs and prompt combinations to identify those with the best detection effectiveness.

Phone fraud prevention evaluation

1. **Continuous classification:** Similar to detection, each scenario's text fragments were sequentially analyzed by all LLM and prompt combinations.
2. **Fraud point determination:** When a fragment is classified as fraudulent, the LLM identifies the specific text portion triggering this classification. The position of the last word in this portion within the overall conversation is marked as the "fraud point." This point is then expressed as a percentage of the total conversation length ("fraud point ratio").
3. **Risk point determination:** "Risk point phrases" were manually annotated within each scenario. These phrases represent moments where the scammer could potentially obtain sensitive information or cause harm, regardless of the victim's response, e.g., a phrase "Please provide Your card number" is considered a risk point phrase. The "risk point" was identified as the last word of this phrase, and its position was calculated as a percentage of the total conversation length ("risk point ratio").
4. **Prevention potential assessment:** If the fraud point ratio was less than the risk point ratio, the LLM had successfully detected the scam before the critical risk point, indicating a successful prevention opportunity. This prevention rate was calculated for each LLM and prompt combination.
5. **Prevention potential variation:** Similar to detection evaluation, each scenario was analyzed 5 times with each LLM and prompt combination to mitigate randomness and calculate the prevention indicator variation.
6. **Result analysis:** The prevention rate was compared across LLMs and prompt combinations to identify those with the best prevention rate.

6.5 LLM prompts

LLM prompts are crucial for specifying the task, context, and desired output format. Effectively prompting the LLMs is essential for accurate and efficient fraud detection within the system. This study utilizes three distinct prompts to evaluate LLMs in the context of phone scam detection and prevention.

It's important to note that during the evaluation, the LLM continuously analyzes each conversation fragment, maintaining context and "remembering" the classification of previous fragments. This allows the model to build a comprehensive understanding of the conversation as it progresses, leading to more accurate assessments.

Prompt 1: Baseline prompt This prompt is simple and direct, asking the model to estimate the probability (as a percentage) that a conversation fragment is fraudulent without providing specific criteria. A 70% threshold is set for classifying a conversation as fraudulent. This prompt leverages "zero-shot prompting", assessing the model's inherent ability to detect fraud based on its existing knowledge. It also serves as a baseline for comparison with other prompts. The expected output is a JSON object with a percentage value representing the likelihood of fraud. Its concise nature and simple output format have the potential for the fastest response times.

Prompt 2: Fraud indicators and risk levels prompt. This prompt provides detailed information about fraud indicators and their risk levels, e.g., an unsolicited offer is a fraud indicator classified as medium risk. The model analyzes text fragments and accumulates information about detected indicators, such as emotional manipulation, impersonation of authority, urgency, and requests for sensitive information. This prompt utilizes a "few-shot prompting" technique to guide the models by providing specific fraud indicator examples. The expected output is a JSON object with the detected indicators and their risk levels. This prompt provides specific criteria for fraud detection, enabling more targeted analysis. Risk levels help the model assess the severity of detected indicators. However, this prompt is longer and requires more complex analysis and output, potentially increasing response time.

Prompt 3: Point system prompt. Similar to Prompt 2, but instead of risk levels, this prompt assigns points to each fraud indicator. The model accumulates points, and upon reaching a threshold, the conversation is flagged as fraudulent. The expected output is a JSON object with the accumulated points. This prompt evaluates the model's ability to assess the severity of indicators by assigning points. The point system allows for more flexible analysis, considering the cumulative effect of indicators. This prompt also requires complex analysis and output. However, it may be faster than Prompt 2 due to the absence of logical checks for different risk level thresholds.

6.6 Limitations and assumptions

The study acknowledged several limitations and assumptions. The predefined fraud indicators in Prompts 2 and 3 may limit the LLM's ability to detect novel or subtle fraud tactics. LLMs are still under development and may make errors or misinterpret context, potentially affecting their accuracy. The study used a controlled environment with transcribed scenarios, isolating LLM performance from real-world complexities like noisy audio, interruptions, and overlapping speech.

The study also relied on several assumptions. Latency measurements assumed a stable network connection, which may not always be the case in real-world applications. The study assumed high accuracy of transcriptions obtained using OpenAI Whisper, although transcription errors can occur and may affect the LLM's analysis.

6.7 Results

This section summarizes the results, addressing the objectives outlined in the methodology: identifying the most suitable conversation content analysis tool and the optimal prompt, considering classification effectiveness, its variation, and latency.

A total of 1350 individual classifications were performed for the comparative evaluation of classification effectiveness: 10 phone conversation scenarios × 5 iterations × 3 languages × 3 models × 3 prompts. The total number of measurements is 7937 (the number of text fragments analyzed to achieve individual classifications and latency evaluation).

6.7.1 Comparative evaluation of classification effectiveness This section presents the results of the comparative evaluation of the classification effectiveness of the conversation content analysis tools, focusing on three main indicators: accuracy, F1 score, and prevention rate. These indicators together provide a comprehensive picture of the ability of LLM models to effectively detect and prevent phone fraud.

Phone fraud detection This section evaluates the ability of LLMs to detect phone fraud by classifying phone conversation scenarios as fraudulent or legitimate. We use two key metrics: accuracy, reflecting the model's ability to correctly classify both fraudulent and legitimate conversations, and the F1 score, which combines precision and recall to provide a balanced evaluation of the model's performance in classifying fraudulent calls.

Figure 5 displays the average accuracy and F1 scores for each LLM across all prompts and target languages, including the 95% confidence intervals derived from variations in classification results for each prompt across languages.

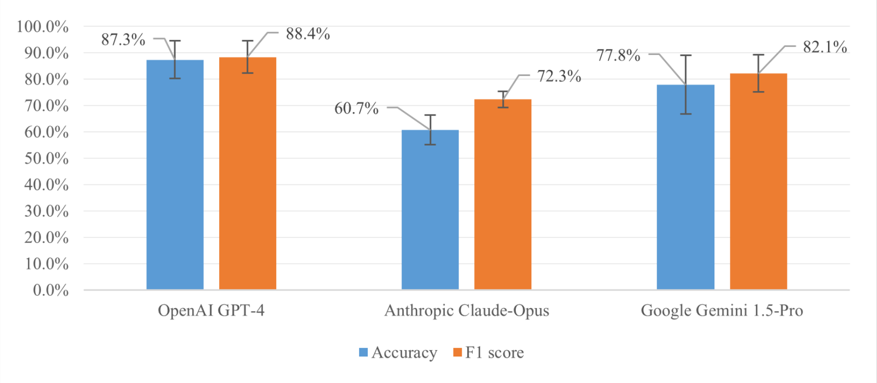

Fig. 5. LLM classification performance at 95% confidence interval

OpenAI GPT-4 demonstrates the best average performance with the highest accuracy and F1 scores, indicating its strong ability to accurately classify phone conversations. Google Gemini 1.5-Pro follows closely with slightly lower but still strong results, demonstrating its effectiveness in fraud detection. Anthropic Claude-Opus shows the weakest performance, exhibiting lower accuracy and F1 scores, which suggests potential difficulties in accurately distinguishing fraudulent calls from legitimate ones.

This lower performance may stem from Anthropic's cautious approach and its tendency to classify conversations as fraudulent based on limited context, such as the mere mention of a scam attempt, regardless of the overall conversation's nature. This behavior results in an increased number of false positives, misclassifying legitimate conversations as fraudulent.

Figure 6 analyzes the impact of each prompt on classification effectiveness, showing the average accuracy and F1 scores for all LLMs, including 95% confidence intervals derived from variations in classification for each LLM across languages.

Key observations from this analysis include:

- Prompt No. 1, leveraging the models' internal understanding and a "zero-shot" learning approach, yields the highest average accuracy and F1 scores. This suggests that allowing LLMs to interpret fraud indicators independently, without explicit criteria, can be more effective in detecting complex scam tactics.
- Prompt No. 2, based on predefined fraud characteristics and risk levels, shows balanced performance with the least variation across LLMs. While providing more consistent results, this approach may lead to overly literal interpretations and false positives when nuanced context is crucial.
- Prompt No. 3, utilizing a point system for fraud indicators, produces the lowest average accuracy and F1 scores. This may be attributed to challenges in accurately assigning points to indicators and defining an appropriate threshold for fraud detection.

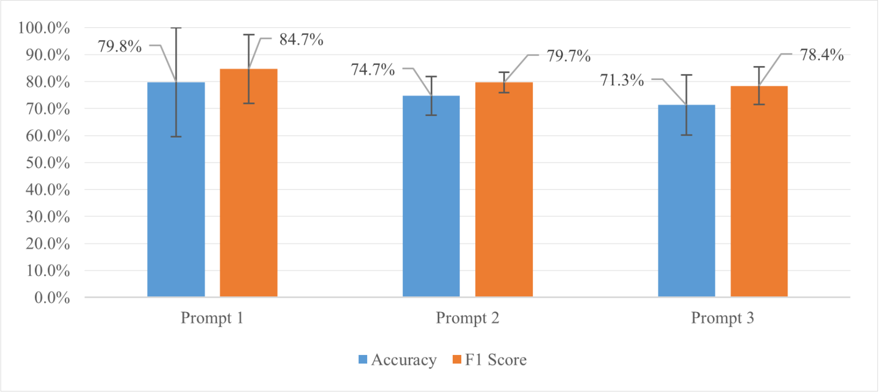

Fig. 6. Impact of prompts on classification performance at 95% confidence interval, aggregating all LLM results

Table 1 provides a more detailed comparison of classification performance across different LLM and prompt combinations, showing average accuracy and F1 scores with 95% confidence intervals derived from variations across languages.

Table 1. Comparative evaluation of LLMs by classification performance metrics

LLM	Prompt	Accuracy (avg.)	95% CI (accuracy)	F1 (avg.)	95% CI (F1)
OpenAI GPT-4	Prompt 1	96,0%	91,4% - 100,0%	95,7%	90,5% - 100,0%
OpenAI GPT-4	Prompt 2	80,0%	70,7% - 89,3%	81,8%	74,2% - 89,4%
OpenAI GPT-4	Prompt 3	86,0%	78,4% - 93,6%	87,6%	81,1% - 94,1%
Google Gemini 1.5-Pro	Prompt 1	90,0%	88,2% - 91,8%	90,0%	88,6% - 91,4%
Google Gemini 1.5-Pro	Prompt 2	78,7%	67,3% - 90,1%	82,6%	73,9% - 91,3%
Google Gemini 1.5-Pro	Prompt 3	64,7%	55,0% - 74,3%	73,8%	68,9% - 78,6%
Anthropic Claude-Opus	Prompt 1	53,3%	48,3% - 58,4%	68,3%	65,8% - 70,7%
Anthropic Claude-Opus	Prompt 2	65,3%	54,1% - 76,6%	74,8%	68,2% - 81,3%
Anthropic Claude-Opus	Prompt 3	63,3%	49,9% - 76,7%	73,8%	66,3% - 81,3%

Key findings from this table include:

- The superiority of Prompt No. 1 in enabling both OpenAI and Google models to achieve their highest accuracy and F1 scores. This emphasizes the effectiveness of leveraging the models' inherent knowledge and understanding of fraud.
- The trade-off between OpenAI and Google: While OpenAI GPT-4 with Prompt No. 1 shows slightly higher average scores, its wider confidence intervals indicate greater performance variation across languages and iterations, raising concerns about consistency. Google Gemini 1.5-Pro with Prompt No. 1 demonstrates more stable and predictable performance, albeit with slightly lower average scores.
- The limitations of Anthropic Claude-Opus: Despite achieving perfect recall (detecting all fraudulent calls) across all prompts (as shown in Table 2), Anthropic consistently exhibits the lowest accuracy and F1 scores due to its significantly lower precision and high false positive rate.

Table 2. Anthropic Claude-Opus classification performance metrics: precision and recall

LLM	Prompt	Precision (avg.)	95% CI (precision)	Recall (avg.)	95% CI (recall)
Anthropic Claude-Opus	Prompt 1	51,9%	49,0% - 54,7%	100,0%	100,0% - 100,0%
Anthropic Claude-Opus	Prompt 2	60,1%	51,5% - 68,8%	100,0%	100,0% - 100,0%
Anthropic Claude-Opus	Prompt 3	59,0%	49,2% - 68,7%	100,0%	100,0% - 100,0%

These findings highlight that while Anthropic can effectively identify all fraudulent calls, its high false positive rate makes it less suitable for a real-time system where minimizing disruption to legitimate conversations is crucial.

Phone fraud prevention This section analyzes the ability of LLMs to proactively prevent phone fraud by identifying fraudulent activity early in a conversation. We focus on the prevention rate, a metric reflecting the model's ability to detect fraud before it reaches a critical "risk point" where the scammer is likely to obtain sensitive information or money.

Figure 7 shows the average prevention rates for each LLM across all prompts and languages, including the 95% confidence interval derived from variations between prevention rates for each prompt across languages.

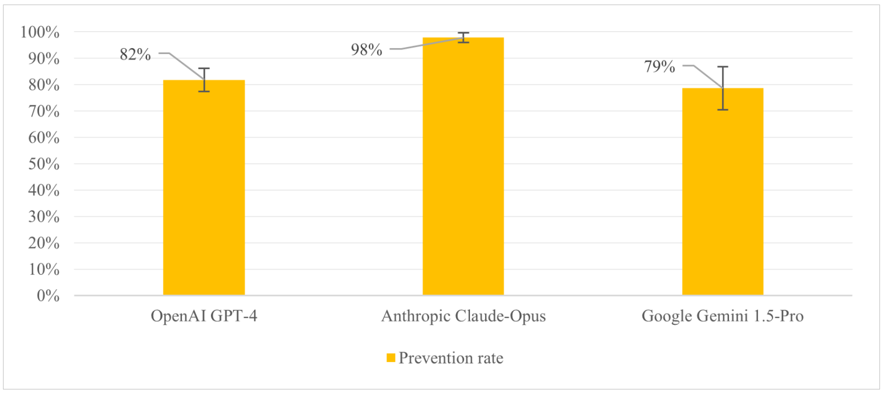

Fig. 7. Average LLM prevention rate at 95% confidence interval, aggregating all LLM results

Anthropic Claude-Opus demonstrates the highest average prevention rate (98%) with minimal variation, indicating its strong ability to detect scams early and prevent potential harm. Google Gemini 1.5-Pro and OpenAI GPT-4 also perform well, achieving prevention rates of 79% and 82% respectively, demonstrating their effectiveness in fraud prevention.

Figure 8 displays the average prevention rates for each prompt, taking into account the results from all LLMs. The 95% confidence intervals reflect the variation in prevention rates between different LLMs for each prompt across languages.

Prompt No. 1, which leverages the models' inherent understanding, achieves the highest average prevention rate (91,1%). This suggests that this prompting strategy allows the models to effectively identify fraudulent behavior before the risk point is reached. Prompt No. 3, based on a feature point system for fraud detection, also shows strong performance with an average prevention rate of 85,8%. In contrast, Prompt No. 2, which focuses on explicit fraud characteristics and risk levels, yields the lowest average prevention rate (81,3%). All prompts exhibit moderate variation in their prevention rates across the different LLMs.

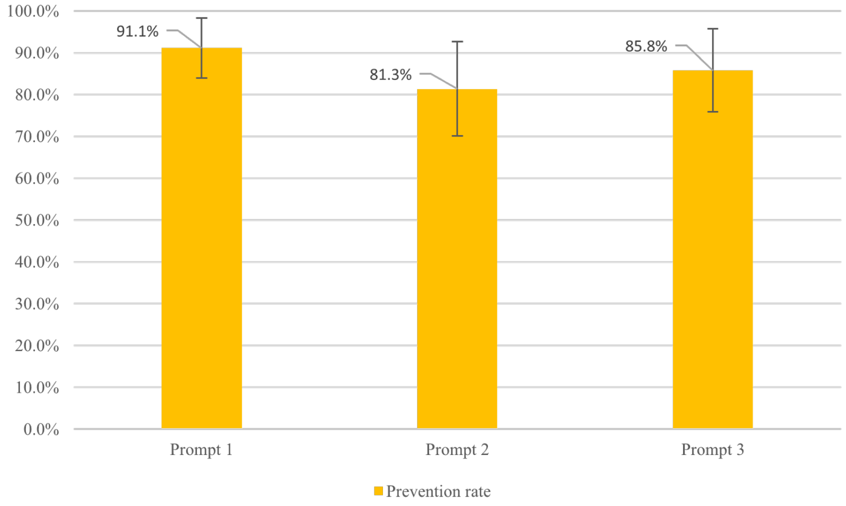

Fig. 8. Average prevention rate of prompts at 95% confidence interval, aggregating all LLM results

Table 3 provides a more granular view, presenting the average prevention rates for each specific LLM and prompt combination, along with their 95% confidence intervals derived from variations across languages.

Key observations from this analysis include:

- Anthropic's consistent excellence in prevention across all prompts, achieving a perfect 100% prevention rate with Prompt No. 1. This highlights the model's exceptional capability in early fraud detection.
- The superior effectiveness of Prompt No. 1 in eliciting high prevention rates from all LLMs. This underscores the value of leveraging the models' inherent understanding and employing a "zero-shot" prompting approach.
- Google Gemini 1.5-Pro's reliability in prevention when used with Prompt No. 1. This combination not only achieves a high prevention rate (89,3%) but also exhibits a narrow confidence interval (87,3% - 91,4%), indicating consistent and stable performance. In comparison, OpenAI GPT-4 with the same prompt shows a slightly lower prevention rate (84%) and wider variation (77,9% - 90,1%).

Despite Anthropic's strong performance in prevention, concerns regarding its accuracy and tendency towards false positives, as detailed in the previous section, make it unsuitable for a real-time system.

Considering both detection (classification effectiveness) and prevention capabilities, Google Gemini 1.5-Pro with Prompt No. 1 emerges as a compelling candidate for a real-time fraud prevention system. This combination offers a balance of accurate fraud detection and minimal disruption to legitimate conversations. However, the final model selection will also consider latency performance, which is analyzed in the next section.

Table 3. Comparative evaluation of LLMs by prevention rate

LLM	Prompt	Prevention rate (avg.)	95% CI
OpenAI GPT-4	Prompt 1	84,0%	77,9% - 90,1%
OpenAI GPT-4	Prompt 2	76,0%	69,0% - 83,0%
OpenAI GPT-4	Prompt 3	85,3%	74,1% - 96,6%
Google Gemini 1.5-Pro	Prompt 1	89,3%	87,3% - 91,4%
Google Gemini 1.5-Pro	Prompt 2	72,0%	62,7% - 81,3%
Google Gemini 1.5-Pro	Prompt 3	74,7%	67,4% - 82,0%
Anthropic Claude-Opus	Prompt 1	100,0%	100,0% - 100,0%
Anthropic Claude-Opus	Prompt 2	96,0%	89,9% - 100,0%
Anthropic Claude-Opus	Prompt 3	97,3%	93,3% - 100,0%

6.7.2 Comparative latency evaluation While classification effectiveness and its variation are primary considerations when selecting an LLM for a real-time phone fraud detection and prevention system, latency, or response speed, is also crucial. This section analyzes the latency of different LLM and prompt combinations to assess their suitability for real-time analysis.

Table 4 shows the average latency for each prompt, aggregating the results of all LLMs. Prompt No. 1 exhibits the lowest average latency (3,33 seconds), followed by Prompt No. 3 (3,51 seconds) and Prompt No. 2 (4,63 seconds). These results confirm the prediction in the LLM prompt section regarding the impact of prompt length, complexity, and output format on response time.

Table 4. Average latency per prompt

Prompt	Average Latency (seconds)
Prompt 1	3,33
Prompt 2	4,63
Prompt 3	3,51

Table 5 presents the latency analysis of the LLMs. Anthropic Claude-Opus demonstrates the slowest response time (4,57 seconds), further supporting previous conclusions about its unsuitability for a real-time system. Google Gemini 1.5-Pro stands out with the lowest average latency (2,69 seconds), indicating its ability to quickly process information and provide analysis responses. OpenAI GPT-4 takes second place with an average latency of 4,20 seconds.

As previously mentioned, choosing between OpenAI GPT-4 and Google Gemini 1.5-Pro with Prompt No. 1 presents a trade-off. While GPT-4 showed better classification effectiveness, its wider confidence intervals suggest greater variation and po-

Table 5. Average latency per LLM

LLM	Average Latency (seconds)
Anthropic Claude-Opus	4,57
Google Gemini 1.5-Pro	2,69
OpenAI GPT-4	4,20

tentially lower consistency compared to Gemini 1.5-Pro. Gemini, on the other hand, demonstrated slightly lower classification scores but with narrower confidence intervals, indicating more stable and reliable performance. Additionally, Gemini 1.5-Pro exhibited a better prevention rate with a narrower confidence interval than GPT-4.

To make a final decision, it's necessary to evaluate the variation in latency and its impact on real-time system operation. Figures 9 and 10 illustrate the latency distribution for OpenAI GPT-4 and Google Gemini 1.5-Pro, respectively, allowing us to assess the stability and consistency of their response times, which is critical for a real-time system aimed at timely phone fraud detection.

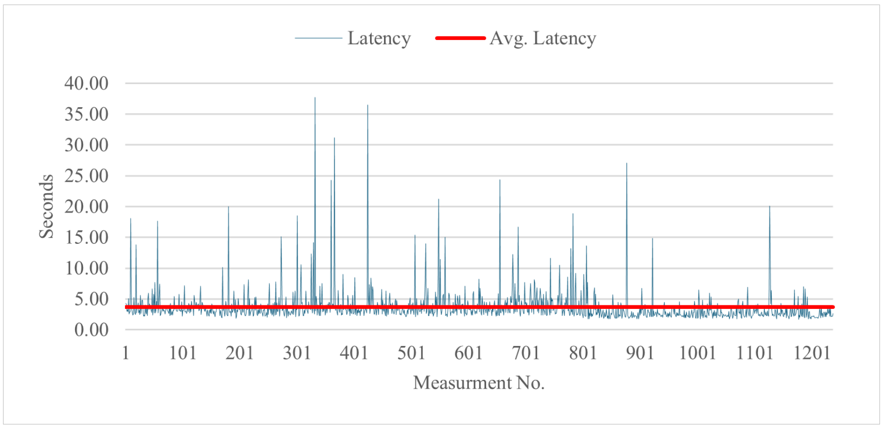

Fig. 9. Variation of OpenAI GPT-4 latency measurements with Prompt No. 1

The data reveals the following:

- OpenAI model's latency instability: Figure 9 shows that while OpenAI GPT-4 has an average latency of 3,66 seconds with Prompt No. 1, the latency measurements are highly variable, with some exceeding 10 seconds and even reaching almost 40 seconds. This instability raises concerns about the model's reliability in a real-time system.
- Google model's latency stability: In contrast, Figure 10 shows that Google Gemini 1.5-Pro has a more stable and consistent latency, mostly remaining within the

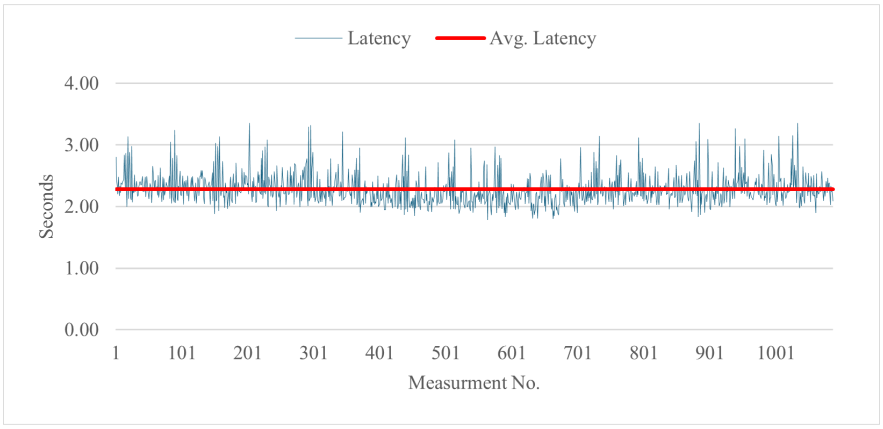

Fig. 10. Variation of Google Gemini 1.5-Pro latency measurements with Prompt No. 1

range of 1,75 to 3,4 seconds, with average latency being 2,28 seconds. This stability ensures a predictable and reliable response time, crucial for real-time system operation.

6.8 Conclusions

This evaluation identified Google Gemini 1.5-Pro with Prompt No. 1 as the optimal choice for a real-time phone fraud detection and prevention system. This conclusion is based on its superior balance of classification effectiveness, prevention rate, and latency, with a strong emphasis on consistency and real-time performance.

While OpenAI GPT-4 initially showed promising accuracy (96%) and F1 score (95,7%), its wider confidence intervals revealed significant performance variation across languages and iterations, raising concerns about its reliability in a real-time setting. Furthermore, its lower prevention rate (84%) compared to Google further supports this decision.

Google Gemini 1.5-Pro consistently demonstrated a strong balance between accuracy (90%) and F1 score (90%) across all languages with narrower confidence intervals, indicating greater stability and predictability. This ensures reliable fraud detection while minimizing disruptive false positives. Moreover, its impressive prevention rate (89,3%) and consistently low latency (averaging 2,28 seconds) solidify its suitability for real-time fraud detection and prevention.

Anthropic Claude-Opus, despite its high prevention rate, proved unsuitable due to lower classification accuracy, a high false positive rate, and the highest average latency among the models tested.

Ultimately, Google Gemini 1.5-Pro with Prompt No. 1 best meet's the system requirements and priorities. Its balanced performance, low latency, and consistent reliability make it an effective and practical solution for real-time phone fraud detection and prevention.

7 System evaluation

This chapter presents an empirical study evaluating the effectiveness of a phone fraud detection and prevention system. The system integrated a pre-selected transcription tool (OpenAI Whisper) and a large language model (Google Gemini 1.5-Pro with Prompt No. 1) to analyze phone conversations in real-time. The study employed an expanded dataset, and a modified methodology tailored to the real-time requirements of the integrated system. The primary goals were to:

- Assess the system's accuracy in classifying phone calls as fraudulent or legitimate.
- Evaluate its ability to prevent fraud through early detection.
- Analyze the system's latency and suitability for real-time operation.

7.1 Methodology

This section details the methodology for evaluating the integrated system. The goal was to objectively assess its performance across various metrics (prevention, accuracy, F1 score, precision, and recall).

The methodology was adapted from previous chapters ("5. Transcription tool selection" and "6. Content analysis tool selection") to accommodate the real-time requirements of the integrated system.

Study dataset. The study employed an expanded dataset, comprising 30 phone conversation scenarios (15 fraudulent, 15 legitimate) translated into Latvian, English, and Russian, resulting in a total of 90 scenarios.

Fraudulent scenarios were sourced from YouTube recordings, reflecting common scam types prevalent in Latvia. Legitimate scenarios, developed in consultation with a senior security expert from one of the largest banks in Latvia, simulate real banking practices and communication styles and represented a control group. This balanced dataset enabled objective evaluation of the system's performance in detecting and preventing diverse scam tactics.

Use of computer-generated voice recordings To ensure consistent testing conditions and control speech rate, the study employed computer-generated voice recordings with specific words per minute (WPM) for each language: English (200 WPM), Latvian (170 WPM), and Russian (160 WPM). These WPM values, based on typical speech rates and adjusted for potential stress-induced acceleration (Rodero, 2012; Kappen et al., 2024; Bogdanovs, 2018), enabled precise calculation of conversation progression for real-time analysis.

Within the simulated real-time environment, this controlled speech rate allowed for precise calculation of conversation progression, enabling accurate assessment of the system's ability to detect and prevent scams before critical points are reached. Further details regarding the methodology and the impact of WPM are provided in subsequent section "Classification effectiveness evaluation."

Latency analysis The latency evaluation methodology at the system level remained similar to the previously described LLM latency assessment. However, in this case, the total time required for the system to process 15-second text fragments and provide a response was measured. This included:

1. **Audio transcription:** Time taken by OpenAI Whisper to transcribe the audio.
2. **Text processing:** Time required for creating the audio recording (simulated by adding 0,40 seconds, the maximum time needed for combining audio tracks in the real-time system) and automatically removing personally identifiable information from the transcribed text.
3. **LLM analysis:** Time taken by Google Gemini 1.5-Pro to process the edited text fragment and provide a response.

These three stages constituted the system's total latency, measured for each text fragment. The results were analyzed to evaluate the system's suitability for real-time operation and to calculate the "real-time word" position for assessing prevention effectiveness (explained in the subsequent section).

Classification effectiveness evaluation The methodology for evaluating classification effectiveness largely followed the principles used for individual LLM assessment in the previous chapter, with modifications to reflect the integrated system evaluation and adapt to real-time analysis challenges.

Similar to the previous chapter, the evaluation included continuous classification, performance metric calculation, variation analysis, and result analysis using the full dataset in 3 target languages. However, to accurately assess fraud prevention effectiveness in a simulated real-time environment, a modified fraud point calculation methodology was introduced, incorporating the time factor.

Fraud point calculation – adapting to real-time environment The simulated environment presented challenges in accurately assessing real-time fraud prevention effectiveness. In a real-world setting, prevention is determined by whether a call is classified and terminated before reaching the "risk point". In the simulated environment, however, the system analyzes segments sequentially, potentially missing the risk point as the conversation progresses in real-time.

To address this, a modified fraud point calculation methodology was implemented, incorporating the time factor (illustrated in Figure 11):

1. **Continuous classification:** Each 15-second text fragment was sequentially analyzed using Google Gemini 1.5-Pro with Prompt No. 1 until fraud was detected or the conversation ended.
2. **Last word identification:** Upon fraud detection, the last word in the transcribed and edited 15-second fragment was identified, representing the point reached in the real-time conversation.
3. **Processing time measurement:** The total time taken by the system to process the fragment, from transcription to analysis results, was measured. This included transcription, text editing, and LLM analysis.

4. **Additional spoken words calculation:** Using the respective language's WPM, the average number of words spoken per second was calculated and multiplied by the processing time to determine the number of words spoken during analysis.
5. **"Real-time word" position determination:** The additional spoken words were added to the last word's position (from step 2) to determine the "real-time word" position – the point the conversation would have reached after analysis (see Figure 11).
6. **Fraud point calculation:** The "real-time word" position was used to calculate the fraud point ratio relative to the total words in the scenario.
7. **Comparison with risk point:** The fraud point ratio was compared to the risk point ratio to determine if the system detected fraud before the critical risk point, considering processing time.

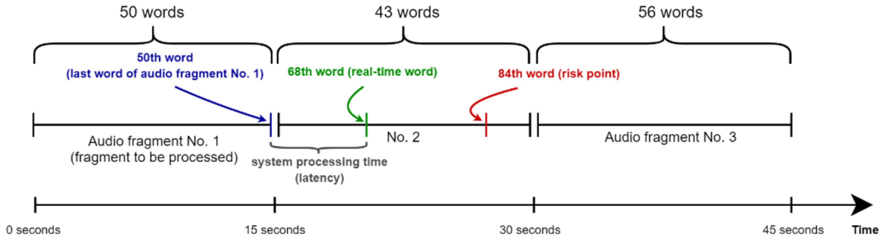

Fig. 11. Illustration of the customized phone fraud prevention methodology

This methodology ensured an objective system evaluation simulating real-time conditions, enabling conclusions about real-time phone scam detection and prevention.

7.2 Limitations and assumptions

While the system evaluation provides valuable insights, it is essential to acknowledge limitations and assumptions. The evaluation assumed a stable network connection for uninterrupted operation of both the transcription tool and the LLM API. Real-world network instability could impact system performance and latency. Although expanded, the dataset may not encompass all possible phone scam scenarios and tactics. While offering controlled conditions, the simulated environment cannot fully replicate the complexity and unpredictability of a real-time system.

7.3 Results

This section presents the results of the system evaluation, addressing the objectives outlined in the methodology section. The evaluation involved 450 individual classifications (30 scenarios x 5 iterations x 3 languages), totaling 2816 measurements (analyzed text fragments for classification and latency assessment).

7.3.1 Latency System latency is crucial for real-time fraud prevention. On average, audio transcription took 1,82 seconds, text processing (including audio track combining simulation and text editing) took 0,43 seconds, and LLM analysis took 2,26 seconds, constituting the largest portion of the total system latency (4,52 seconds).

Minor discrepancies were observed compared to previous latency measurements in transcription tool evaluation chapter. Transcription latency was slightly higher (1,82 seconds vs. 1,53 seconds), potentially due to increased load on OpenAI servers. However, the LLM analysis latency (2,26 seconds) remained consistent with previous findings (2,28 seconds).

7.3.2 Classification effectiveness This section analyzes the integrated system's effectiveness in detecting and preventing phone scams using the expanded dataset and modified fraud point calculation. Four metrics were used to assess detection performance: accuracy, F1 score, precision, and recall. Prevention performance was evaluated using the prevention rate.

Phone scam detection Figure 12 presents the average values for each metric across all languages and iterations, including 95% confidence intervals derived from variations across target languages.

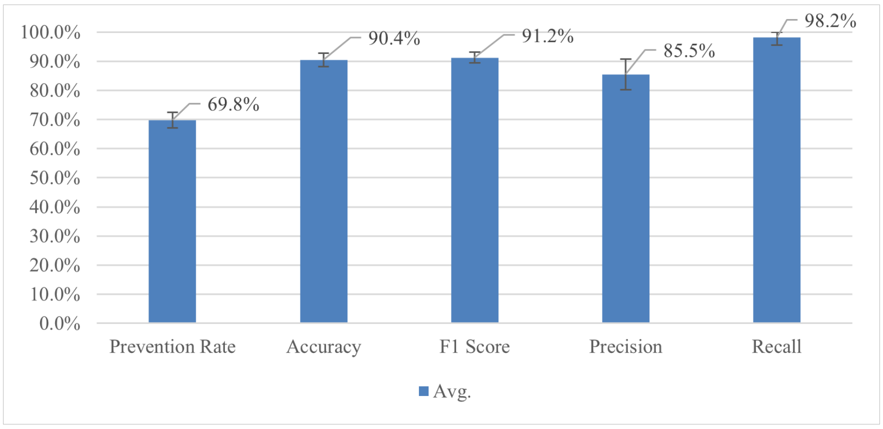

Fig. 12. System's average performance metrics at 95% confidence interval

The data indicates:

- **High classification effectiveness:** The system achieved an average accuracy of 90,4% and an F1 score of 91,2%, demonstrating its ability to accurately distinguish fraudulent and legitimate calls.
- **Excellent recall:** The system exhibited a recall of 98,2%, indicating its ability to detect almost all actual scam cases.

- **High precision:** The system's precision was 85,5%, suggesting a relatively low rate of false positives.
- **Moderate prevention rate:** The system achieved a prevention rate of 69,8%, successfully preventing scams in almost 70% of cases.

The narrow confidence intervals suggest consistent classification performance across scenarios and languages, crucial for reliable real-time operation.

Comparing the integrated system's performance to the isolated Google Gemini 1.5-Pro results (from "6. Content analysis tool selection"), the classification effectiveness was similar. The integrated system's average accuracy (90,4%) and F1 score (91,2%) aligned with the previous findings (90% accuracy and 90% F1 score). Similarly, precision (85,5%) and recall (98,2%) were comparable to the isolated model's performance (90,4% precision and 90,7% recall).

Despite minor variations, the integrated system's average values for three out of four classification metrics fell within the confidence intervals of the isolated Google model with Prompt No. 1:

- **Accuracy:** 90,4% (previous CI: 88,2% - 91,8%)
- **F1 score:** 91,2% (previous CI: 88,6% - 91,4%)
- **Precision:** 85,5% (previous CI: 85,6% - 95,3%) - only slightly below the lower bound due to rounding.
- **Recall:** 98,2% (previous CI: 88,6% - 92,7%) - exceeding the upper bound, likely attributed to the expanded dataset providing more examples for fraud identification.

This demonstrates that the integrated system achieved comparable classification effectiveness with an expanded dataset, while also providing real-time transcription and scam detection.

Phone scam prevention The integrated system's average prevention rate (69,8%, 95% CI: 67,1% - 72,5%) was lower than the isolated Google model's (89,3%, 95% CI: 87,3% - 91,4%). This difference can be primarily attributed to the use of an expanded dataset that incorporates a broader variety of real-world scenarios, as well as the inclusion of real-time factors (such as processing time and the "real-time word" position) in the fraud point calculation.

To assess the impact of latency, a "potential prevention rate" (ignoring processing time) was calculated (76%). This minor difference (6,2%) indicates that latency had a limited effect. The primary factor for the lower rate was the broader variety of scenarios present in the expanded dataset.

7.4 Conclusions

The integrated system demonstrated strong performance in both scam detection and prevention, achieving high accuracy (90,4%), F1 score (91,2%), and recall (98,2%). While the prevention rate (69,8%) was lower than the isolated LLM evaluation, it still highlights the system's potential for real-time fraud prevention.

The system's classification effectiveness was comparable to the isolated Google Gemini 1.5-Pro model, confirming its ability to maintain high accuracy and reliability with a larger dataset while providing real-time transcription and analysis.

Narrow confidence intervals across metrics indicate system stability and consistency across languages and iterations. The lower prevention rate is primarily attributed to the expanded dataset's complexity and minimally to the modified fraud point calculation for real-time simulation.

The system's average latency was 4,52 seconds, with 1,82 seconds for transcription, 0,43 seconds for text processing, and 2,26 seconds for LLM analysis. This latency is acceptable for real-time fraud detection, allowing timely analysis and identification of fraudulent indicators and thus timely prevention.

8 Discussion

In this study, we presented an integrated system for phone fraud detection and prevention that leverages AI tools to analyze phone conversation content in a simulated real-time environment. The evaluation was conducted using a dataset of fraudulent and legitimate transcripts derived from real-world call scenarios, with synthetic audio recordings. Our approach not only assesses fraud detection performance using key metrics such as accuracy, F1 score, precision, and recall but also emphasizes the system's capability to prevent fraud—an aspect that is often overlooked in the literature. This dual focus on both detection and prevention provides a comprehensive evaluation of the system's potential for practical application. While the results are promising in this controlled environment, they should be interpreted with the understanding that the evaluation setting may differ from live real-world conditions.

8.1 System advantages

The developed prototype for phone scam detection and prevention offers several advantages. By integrating AI-based tools for speech recognition and natural language processing, the system effectively complements traditional call filtering methods. Unlike simpler approaches, the system performs in-depth analysis of conversation content, enabling the identification of complex scam tactics and social engineering techniques. Furthermore, its language-agnostic design facilitates multilingual scam detection. The system's flexible architecture, based on microservices, allows for easy updates and future integration of improved AI models.

8.2 Limitations and criticism

Despite promising results, the system has limitations. Data privacy is a key concern, requiring strict adherence to data protection regulations. Automated data editing may lead to errors and potential information leakage, necessitating robust quality control. The dataset, while expanded, remains limited in its representation of potential scam scenarios and linguistic variations. Additionally, the evaluation was conducted using synthetic audio rather than real phone call recordings. While the synthetic audio was designed to reflect real-world scenarios and replicate telecom-grade quality, it may not fully capture the complexities of live phone conversation audio, including background noise, speaker variability, and mobile network disruptions.

The system's dependence on automatic speech recognition is a critical factor. Inaccuracies in transcription and audio segmentation can distort conversational context and hinder accurate classification. In real-world settings, challenges such as noisy environments, mobile network interruptions and linguistic nuances may exacerbate ASR errors—potentially leading to significantly higher word error rates that could compromise overall system effectiveness.

Finally, The chosen prompt may not be universally optimal, and the inherent variability in LLM outputs can influence system reliability.

9 Conclusions and future work

This study addressed the pressing issue of phone fraud and scams, which cause significant financial losses and emotional distress to telecommunication users globally.

Traditional detection methods, such as blacklists and CDR analysis, are becoming less effective due to scammers' evolving tactics (e.g., caller ID spoofing). This necessitates innovative approaches to combat phone fraud.

Conversation content analysis has emerged as a promising avenue for real-time scam detection and prevention. This approach leverages advancements in artificial intelligence, specifically automatic speech recognition (ASR) and large language models (LLM), to analyze the content of phone conversations and identify fraudulent patterns.

This study developed a prototype system integrating two AI tools: OpenAI Whisper for ASR and Google Gemini 1.5-Pro for LLM-based content analysis. Each tool was selected through a rigorous evaluation methodology.

Empirical evaluation of the integrated system in a simulated real-time environment demonstrated its high classification effectiveness. The system achieved strong performance across key metrics and the results indicate the system's ability to effectively detect fraud while maintaining a low false positive rate.

Furthermore, the system achieved a prevention rate of 69,8% (95% CI: 67,1% - 72,5%), demonstrating its potential for real-time intervention and prevention of phone fraud. While much research focuses on detection, this study highlights the importance of evaluating and optimizing systems for proactive prevention, aiming to stop scams before they cause harm. It is crucial to recognize that mere identification of a fraud attempt is not equivalent to prevention. A system might accurately identify a fraudulent call, but if the identification occurs too late in the conversation, it may not be possible to prevent negative consequences.

The system offers several advantages that make it a promising solution for combating phone fraud. It can be integrated with existing protection methods, such as blacklists and call filtering, creating a multi-layered defense system. The system's ability to analyze conversation content in real-time allows for rapid response to scam attempts. Multilingual support broadens its applicability across different regions. Finally, the flexible microservices architecture enables easy adaptation and updates, allowing for the integration of newer and improved AI tools as needed.

However, the evaluation also revealed limitations that need to be acknowledged. Data privacy remains a critical concern, requiring careful consideration and robust security measures. The potential for data editing errors and unintended information leakage necessitates further research and development of more reliable techniques. The limited size and diversity of the dataset may affect the system's generalizability. Furthermore, challenges remain in optimizing prompts for specific scam types and managing the inherent variability in LLM outputs. Finally, errors in speech recognition and audio segmentation can lead to contextual distortions that hinder system effectiveness.

Future research should focus on addressing current limitations. This includes conducting real-world testing using actual phone call audio to evaluate system performance under real conditions and expanding the dataset to encompass a wider range of scenarios, languages and linguistic variations to assess generalizability. The use of synthetic audio recordings—while designed to replicate real-world conditions—may not fully capture the complexities of live phone conversations, such as background noise, speaker variability, and mobile network disruptions. Furthermore, our evaluation used a balanced dataset for controlled assessment. However, in real-world scenarios, fraudulent calls are far less frequent, which may reduce precision as false positives could outweigh true fraud cases. Future work should focus on adapting the system to this imbalance for more realistic performance. Additionally, exploring the integration of streaming ASR solutions and evaluating alternative file formats (e.g., lossless FLAC) for transcription could help reduce latency and improve accuracy. Investigating separate speaker transcription and timestamped utterances could enhance fraud detection by providing clearer conversational context for the LLM. Further investigation of prompt engineering and optimization techniques is needed. Exploring strategies to mitigate variability in LLM outputsand evaluating the feasibility of open-source AI tools could lead to more robust and cost-effective solutions. Finally, expanding the application of LLMs to detect fraud in other communication channels, such as SMS and email, could contribute to a more comprehensive approach to fraud prevention.

References

Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S. et al. (2023). Gpt-4 technical report, *arXiv preprint arXiv:2303.08774* .

Bogdanovs, M. (2018). Runas ātrums audiovizuālajos materiālos un tā ietekme uz audiovizuālās tulkošanas stratēğijām [Speech rate in audiovisual materials and its impact on audiovisual translation strategies].

Derakhshan, A., Harris, I. G., Behzadi, M. (2021). Detecting telephone-based social engineering attacks using scam signatures, *Proceedings of the 2021 ACM Workshop on Security and Privacy Analytics*, pp. 67–73.

Ferraro, A., Galli, A., La Gatta, V., Postiglione, M. (2023). Benchmarking open source and paid services for speech to text: an analysis of quality and input variety, *Frontiers in big Data* **6**, 1210559.

Filippidou, F., Moussiades, L. (2020). A benchmarking of ibm, google and wit automatic speech recognition systems, *Artificial Intelligence Applications and Innovations: 16th IFIP WG 12.5 International Conference, AIAI 2020, Neos Marmaras, Greece, June 5–7, 2020, Proceedings, Part I 16*, Springer, pp. 73–82.

Gowri, S. M., Sharang Ramana, G., Sree Ranjani, M., Tharani, T. (2021). Detection of telephony spam and scams using recurrent neural network (rnn) algorithm, *2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS)*, Vol. 1, pp. 1284–1288.

Hong, B., Connie, T., Goh, M. K. O. (2023). Scam calls detection using machine learning approaches, *2023 11th International Conference on Information and Communication Technology (ICoICT)*, IEEE, pp. 442–447.

Kappen, M., Vanhollebeke, G., Van Der Donckt, J., Van Hoecke, S., Vanderhasselt, M.-A. (2024). Acoustic and prosodic speech features reflect physiological stress but not isolated negative affect: a multi-paradigm study on psychosocial stressors, *Scientific Reports* **14**(1), 5515.

Li, H., Xu, X., Liu, C., Ren, T., Wu, K., Cao, X., Zhang, W., Yu, Y., Song, D. (2018). A machine learning approach to prevent malicious calls over telephony networks, *2018 IEEE Symposium on Security and Privacy (SP)*, pp. 53–69.

Pandit, S., Liu, J., Perdisci, R., Ahamad, M. (2020). Fighting voice spam with a virtual assistant prototype.
`https://arxiv.org/abs/2008.03554`

Rodero, E. (2012). A comparative analysis of speech rate and perception in radio bulletins, *Text & Talk* **32**(3), 391–411.

Sahin, M., Francillon, A., Gupta, P., Ahamad, M. (2017). Sok: Fraud in telephony networks, *2017 IEEE European Symposium on Security and Privacy (EuroS&P)*, pp. 235–250.

Sawa, Y., Bhakta, R., Harris, I. G., Hadnagy, C. (2016). Detection of social engineering attacks through natural language processing of conversations, *2016 IEEE Tenth International Conference on Semantic Computing (ICSC)*, pp. 262–265.

Song, J., Kim, H., Gkelias, A. (2014). ivisher: Real-time detection of caller id spoofing, *ETRI Journal* **36**(5), 865–875.

Team, G., Anil, R., Borgeaud, S., Wu, Y., Alayrac, J.-B., Yu, J., Soricut, R., Schalkwyk, J., Dai, A. M., Hauth, A. et al. (2023). Gemini: a family of highly capable multimodal models, *arXiv preprint arXiv:2312.11805* .

Trapiņš, A. (2015). Krāpšanas gadījumu atklāšana un prevencija mobilo sakaru tīklos [Fraud detection and prevention in mobile networks].

WEB (a). European commission. Survey on ”scams and fraud experienced by consumers”.
`https://commission.europa.eu/system/files/2020-01/survey_on_scams_and_fraud_experienced_by_consumers_-_final_report.pdf`

WEB (b). Usa federal trade commission. Consumer sentinel network data book 2023.
`https://www.ftc.gov/system/files/ftc_gov/pdf/CSN-Annual-Data-Book-2023.pdf`

WEB (c). Finance latvia association. The amount of fraud prevented last year reached 9.2 million euros; the most difficult to prevent are cases of phone fraud.
`https://www.financelatvia.eu/news/pern-noversto-krapsanu-gadijumu-apmers-sasniedz-92-miljonus-eiro-visgrutak-noverst-telefonkrapsanas-gadijumus/`

WEB (d). Usa department of justice. Financial fraud crime victims.
`https://www.justice.gov/usao-wdwa/victim-witness/victim-info/financial-fraud`

WEB (e). Latvian state police. Fraudsters impersonate police officers and use emergency service phone numbers 110 and 112.
`https://www.vp.gov.lv/lv/jaunums/krapnieki-uzdodas-par-policistiem-un-izmanto-operativo-dienestu-talruna-numurus-110-un-112`

WEB (f). Google. Announcement of policy changes: April 6, 2022.
`https://support.google.com/googleplay/android-developer/answer/14554743`

WEB (g). Deepgram. The best speech-to-text apis in 2024.
`https://deepgram.com/learn/best-speech-to-text-apis`

WEB (h). Assemblyai. Industry’s most accurate speech ai models.
`https://www.assemblyai.com/benchmarks`

WEB (i). Speechmatics. Introducing ursa from speechmatics. https://www.speechmatics.com/company/articles-and-news/introducing-ursa-the-worlds-most-accurate-speech-to-text

WEB (j). Revai. Microsoft azure speech recognition vs. rev ai speech to text api. https://www.rev.com/blog/resources/microsoft-azure-speech-recognition-vs-rev-ai-speech-to-text-api

WEB (k). Playht. https://play.ht/

WEB (l). Anthropic. The claude 3 model family: Opus, sonnet, haiku. https://www-cdn.anthropic.com/de8ba9b01c9ab7cbabf5c33b80b7bbc618857627/Model_Card_Claude_3.pdf

Xing, J., Yu, M., Wang, S., Zhang, Y., Ding, Y. (2020). Automated fraudulent phone call recognition through deep learning, *Wireless Communications and Mobile Computing* **2020**(1), 8853468.

Zhao, Q., Chen, K., Li, T., Yang, Y., Wang, X. (2018). Detecting telecommunication fraud by understanding the contents of a call, *Cybersecurity* **1**, 1–12.

Received October 25, 2024 , revised March 3, 2025, accepted March 4, 2025