Software Engineering and Applications 软件工程与应用, 2026, 15(3), 412-422
Published Online June 2026 in Hans. https://www.hanspub.org/journal/sea
https://doi.org/10.12677/sea.2026.153039

Hans汉斯

基于HarmonyOS的端云协同语音反诈系统

黄纯琴，杨金春，严　言，杨再鑫，臧　淼

北方工业大学人工智能与计算机学院，北京

收稿日期：2026年5月11日；录用日期：2026年6月18日；发布日期：2026年6月30日

摘　要

针对电信网络诈骗话术不断演化、用户在通话过程中缺乏实时语音内容风险预警的问题，本文设计并实现了一种基于HarmonyOS的端云协同语音反诈系统。系统采用前后端分离架构：前端基于HarmonyOS ArkTS框架，实现多音源音频播放、音频选择、用户交互及结果可视化；后端基于Python Flask框架，完成音频预处理、百度短语音识别接口调用以及诈骗文本检测。为提升系统实时性与工程可用性，在大模型语义分析基础上，引入“本地规则预筛 + 大模型语义判别补充 + 结果缓存”的融合检测策略，输出结构化检测结果，包括风险等级、置信度、判定理由和安全建议。实验结果表明：在自构建的电信诈骗与正常通话语音数据集上，系统整体诈骗检测准确率达到91.6%，对“安全账户”“冻结资金”“法院传票”等关键诈骗短语的召回率超过95%；在Wi-Fi环境下，系统端到端平均响应时延约为2 s，连续100次检测成功率达到100%。进一步的补充实验表明，优化后的融合检测流程在20条典型文本样本上取得了100%的分类准确率，平均响应时间约为0.38 s。结果说明，该系统能够有效识别典型诈骗话术，并在准确率与实时性之间取得较好平衡，可为HarmonyOS生态下智能反诈应用开发提供参考。

关键词

HarmonyOS，语音反诈，端云协同，语音识别，诈骗检测

A Cloud-Edge Collaborative Voice Anti-Fraud System Based on HarmonyOS

Chunqin Huang, Jinchun Yang, Yan Yan, Zaixin Yang, Miao Zang

School of Artificial Intelligence and Computer Science, North China University of Technology, Beijing

Received: May 11, 2026; accepted: June 18, 2026; published: June 30, 2026

Abstract

To address the problem that telecom fraud scripts are constantly evolving and users lack real-time risk warning during phone calls, this paper designs and implements a cloud-edge collaborative

文章引用: 黄纯琴, 杨金春, 严言, 杨再鑫, 臧淼. 基于 HarmonyOS 的端云协同语音反诈系统[J]. 软件工程与应用, 2026, 15(3): 412-422. DOI: 10.12677/sea.2026.153039

voice anti-fraud system based on HarmonyOS. The system adopts a front-end/back-end separated architecture. The front end is developed with the HarmonyOS ArkTS framework to support multi-source audio playback, audio selection, user interaction, and result visualization. The back end is built on Python Flask to perform audio preprocessing, invoke Baidu short speech recognition services, and conduct fraud text detection. To improve real-time performance and engineering practicality, a hybrid detection strategy integrating local rule-based pre-screening, large-model-assisted semantic judgment, and result caching is proposed on top of large-model semantic analysis. The system outputs structured results including risk level, confidence score, reason, and safety suggestions. Experimental results show that, on a self-constructed dataset of telecom fraud and normal conversation speech, the overall fraud detection accuracy reaches 91.6%, while the recall of key fraud phrases such as "safe account", "frozen funds", and "court subpoena" exceeds 95%. Under Wi-Fi conditions, the average end-to-end response latency is about 2 s, and the success rate over 100 consecutive tests reaches 100%. In addition, supplementary experiments on 20 representative text samples show that the optimized hybrid detection process achieves 100% classification accuracy with an average response time of 0.38 s. The results indicate that the proposed system can effectively identify typical fraud scripts and achieve a good balance between accuracy and real-time performance, providing a practical reference for intelligent anti-fraud application development in the HarmonyOS ecosystem.

Keywords

Harmonyos, Voice Anti-Fraud, Cloud-Edge Collaboration, Speech Recognition, Fraud Detection

1. 引言

近年来，电信网络诈骗犯罪呈现组织化、专业化和智能化发展趋势，诈骗手段不断翻新，话术脚本高度标准化，受害人群体逐渐扩大并呈现年轻化趋势。现有反诈预警方式大多集中在号码标记、短信提醒、账户风控和事后追踪等环节，对于用户进行的语音通话内容缺乏实时分析与风险提示能力。在实际诈骗过程中，诈骗分子往往通过冒充公检法、虚假投资理财、贷款刷流水、刷单返利等话术营造紧张氛围，诱导用户在高压状态下完成转账或泄露敏感信息。因此，面向通话内容的实时语音反诈检测具有重要应用价值。

随着语音识别、自然语言处理和大语言模型的发展，利用语音转写与语义分析技术实现通话风险预警成为可能[1]-[8]。相较于传统关键词匹配方法，大模型在语义理解、上下文推理和复杂文本分类方面具有更强能力，为诈骗文本识别提供了新的技术基础[5]-[8]。与此同时，HarmonyOS 提供了较完善的多媒体能力、网络接口和声明式 UI 框架，为构建语音反诈移动应用提供了良好支持[9]。

针对现有语音反诈应用在实时性、交互性和可解释性方面的不足，本文设计并实现了一种基于HarmonyOS的端云协同语音反诈系统。系统将语音识别与文本检测任务部署在云端，端侧负责音频播放、音频选择、检测触发和结果展示。本文的主要贡献如下：

1) 提出面向 HarmonyOS 的端云协同语音反诈系统架构，打通“音频输入－语音识别－诈骗检测－风险展示”的完整流程；

2) 设计“本地规则预筛 + 大模型语义判定 + 结果缓存”的融合检测策略，兼顾可解释性与响应效率；

3)通过语音数据集实验和补充文本实验验证了系统在检测准确率、响应时延和运行稳定性方面的有效性。

2. 方法

2.1. 系统总体框架

本系统采用前后端分离的端云协同架构。前端基于 HarmonyOS ArkTS 实现，主要负责音频播放控制、音频源选择、URL 输入、检测请求触发和结果展示；后端基于 Python Flask 框架搭建 RESTful 服务，负责音频上传、格式转换、语音识别、诈骗检测和结果返回。

用户在 APP 中选择内置示例音频、本地媒体文件或网络音频 URL 后，可播放音频并触发“识别并检测当前音频”操作。前端将音频数据或音频地址通过 HTTP/HTTPS 发送至后端，后端完成格式检查与预处理后，调用百度短语音识别 API 获取转写文本，再结合规则模块和大模型完成诈骗风险判定，最终将结构化结果返回前端。前端展示识别文本、风险等级、置信度、判定理由和安全建议，并通过颜色区分高风险与正常状态。系统整体架构如图 1 所示，工作流程如图 2 所示。

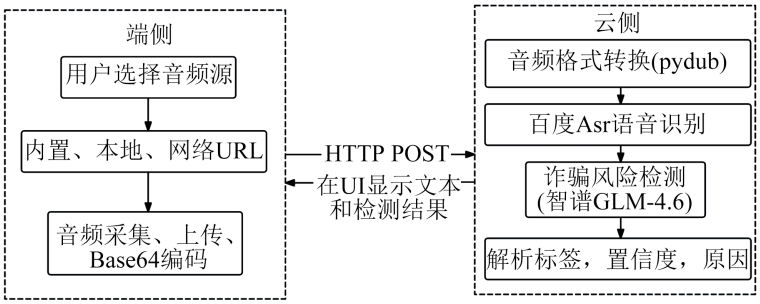


Figure 1. Overall system architecture diagram
图 1. 系统整体架构图

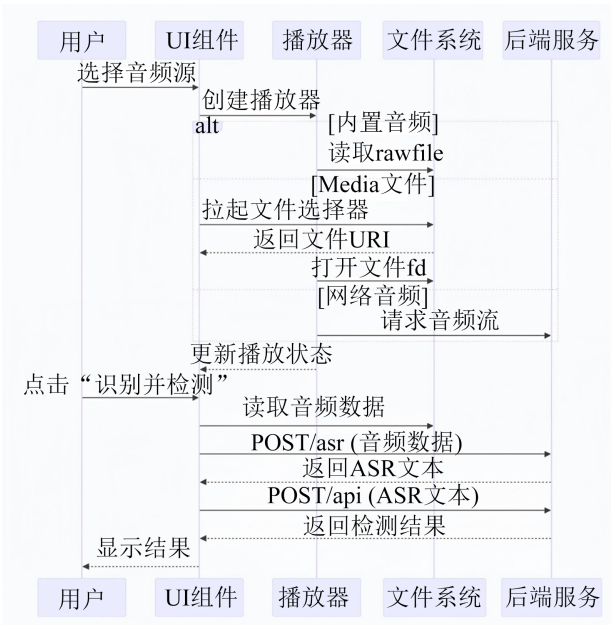


Figure 2. System working sequence diagram
图 2. 系统工作时序图

2.2. 语音采集与预处理

系统支持三类音频来源：

1) 内置音频，用于功能演示与反诈教学；

2) 本地媒体文件，用户可从设备媒体库中选择已有音频进行检测分析；

3) 网络音频，用户可输入网络音频 URL 进行播放与检测。

由于不同来源音频在编码格式、采样率、声道数和时长等方面存在差异，后端在接收音频后首先进行格式检查，并统一转换为单声道、16 kHz 采样率的 PCM/WAV 格式。对于 m4a、mp3 等压缩格式音频，后端先完成解码与重采样，再输入语音识别模块。该流程能够提高系统对多源音频的兼容能力和识别稳定性。

2.3. 语音识别模块设计

语音识别模块负责将音频内容转换为可供后续检测的文本。本文采用百度智能云短语音识别接口实现该模块[9]。后端在完成音频预处理后，将标准化音频数据封装为接口请求，并设置采样率、编码格式和语言类型等参数。语音识别接口返回候选结果后，系统选取置信度较高的文本作为最终转写结果。

考虑到诈骗检测更关注“验证码”“安全账户”“冻结资金”“转账”“远程协助”“刷流水”等高风险短语是否被准确保留，系统在后处理阶段对识别文本进行基础清洗，包括去除无效空白字符、统一中英文符号和过滤明显噪声片段。对于识别失败、音频为空或接口异常的情况，后端返回明确错误信息，前端提示用户重新选择音频或检查网络状态。

2.4. 诈骗风险检测模型

诈骗风险检测模块是系统的核心。考虑到传统关键词匹配方法对变体话术和隐晦表达覆盖不足，本文采用“本地规则预筛 + 大模型语义判定 + 结果缓存”的分层检测方案，以兼顾准确率、实时性和工程成本。复杂文本语义分析部分基于智谱 GLM 系列大模型开放平台实现 [1]。

本地规则模块主要用于识别边界清晰的文本。系统构建了高风险短语规则库和正常场景短语规则库。高风险短语包括“安全账户”“冻结资金”“刷流水”“先垫付”“验证码”“屏幕共享”“远程协助”等；正常场景短语包括“订单页面”“支付成功”“物流通知”“项目进度会”“账单提醒”等。若文本命中明显高风险规则且无强正常语境，则系统直接判定为高风险；若文本属于明确正常业务场景，则直接判定为正常。

对于同时包含风险词与正常场景词、或语义存在歧义的文本，系统调用大模型进行语义判定。后端将转写文本组织为提示词输入大模型，要求其输出结构化结果，包括风险标签、置信度和判定理由。为降低重复检测带来的调用成本，系统引入结果缓存机制，对相同或高度重复的文本直接返回历史结果。

为提高大模型判定结果的稳定性和结构化程度，本文对提示词进行了模板化设计。提示词包括角色设定、任务描述、输出约束和示例引导四部分。其中，角色设定为“电信诈骗文本识别助手”，任务描述要求模型对输入的语音转写文本进行诈骗/正常二分类判断；输出约束要求模型返回风险标签、置信度、判定理由和安全建议等结构化字段；示例引导则通过典型诈骗文本帮助模型更稳定地理解输出格式与判定标准。其模板示例如下：

请你作为电信诈骗风险识别助手，对以下文本进行分类。

要求：

1. 判断是否为诈骗相关话术；

2. 输出风险标签(scam 或 normal)；

[1] https://open.bigmodel.cn/.

3. 给出 0~1 之间的置信度；
4. 简要说明判定理由；
5. 给出安全建议；
6. 以 JSON 格式返回。

待分析文本：

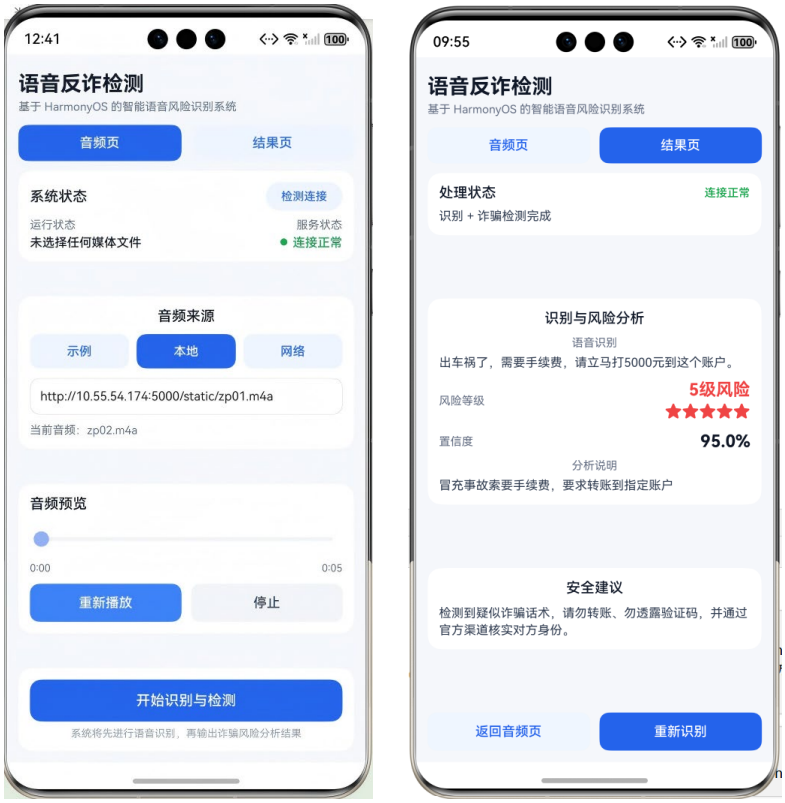
“您好，您的银行卡涉嫌洗钱，请立即将资金转入安全账户配合调查。”

输出示例：

```
{
"label": "scam",
"confidence": 0.97,
"reason": "文本包含冒充公检法、资金转入安全账户等典型诈骗特征",
"advice": "不要转账，不要泄露账户信息，请通过官方渠道核实。"
}
```

2.5. 前端交互与可视化设计

前端应用基于 HarmonyOS ArkTS 框架实现，界面主要包括音频源选择区、播放控制区、检测触发区、识别文本展示区和风险结果展示区。用户可在内置音频、本地媒体文件和网络 URL 音频之间切换，完成音频播放、暂停、重新选择和检测操作，界面示例如图 3 所示。



(a) 音频播放界面　　(b) 结果检测界面

Figure 3. Application interface screenshot
图 3. 应用界面截图

在结果展示方面，前端根据后端返回的结构化结果展示识别文本、风险等级、置信度、判定理由和安全建议。当检测结果为诈骗时，提示用户不要转账、不要泄露验证码、通过官方渠道核实身份；当检测结果为正常时，说明当前内容未发现明显诈骗特征。

此外，界面还显示当前处理状态，如待检测、识别中、检测完成或播放结束等。前端同时对网络异常和后端不可达情况进行检测，并通过提示信息提醒用户检查网络或稍后重试。

3. 结果

3.1. 实验数据与设置

为评估系统有效性，本文构建了包含电信诈骗语音与正常通话语音的实验数据集。诈骗语音来源于公开反诈宣传材料、模拟诈骗话术录音以及部分脱敏处理后的真实案件语音；正常语音包括日常生活通话、业务咨询和公共服务电话录音等。原始语音素材累计时长约 14 h，其中诈骗语音约 6 h，正常语音约 8 h。

考虑到系统检测流程主要面向短时语音内容，本文对原始通话素材按语义完整性和停顿边界进行切分，形成有效语音片段数据集。最终数据集共包含语音样本 420 条，其中诈骗样本 180 条，正常样本 240 条；单条语音时长范围为 8 s~58 s；转写文本字数范围约为 12 字~168 字。

所有语音数据由两名标注人员独立进行诈骗/正常二分类标注。若通话内容涉及诱导转账、索要验证码、冒充公检法、虚假投资理财、贷款刷流水等，则标注为“诈骗”；正常咨询、闲聊、业务办理、物流和账单通知等标注为“正常”。两名标注人员一致率为 94.2%，存在分歧的样本由第三名标注人员仲裁。标注完成后，按通话级别将数据集划分为开发集和测试集，比例为 8:2，并保证同一通话的所有片段不跨集合划分。开发集主要用于规则库整理、提示词调试和基线方法参数选择，测试集用于最终性能评估。

系统后端部署在配备 Intel Core i7 处理器和 16 GB 内存的服务器上，操作系统为 Windows 11，Python 版本为 3.13；前端在 HarmonyOS 手机和润和 DAYU200 开发板上进行测试。语音识别服务采用百度普通话识别模型，音频采样率统一为 16 kHz。

3.2. 语音识别性能

本文采用字错误率(CER)评价语音识别性能，并将系统识别结果与人工转写结果进行对比。实验结果表明，在信噪比较高的宣传录音和模拟录音场景下，CER 约为 6%~8%；在背景噪声较大或语速较快的真实通话录音中，CER 有所上升，但整体保持在 12%以内。

从诈骗检测任务角度看，系统更关注与风险判定相关的关键词和短语是否能够被正确识别。统计结果显示，在测试集中，“安全账户”“冻结资金”“远程协助”“法院传票”等关键短语的识别召回率达到 95%以上，说明语音识别模块能够较好保留诈骗检测所需的关键信息。

3.3. 诈骗检测性能

在测试集上对诈骗检测模块进行评估。本文将每段通话的转写文本输入检测模块，以人工标注结果为基准，计算系统整体检测准确率。为验证方法有效性，本文设置了两种基线方法进行对比：一是仅使用关键词规则；二是基于 TF-IDF 特征的 SVM 分类器。实验结果表明，关键词规则方法虽响应速度快，但对变体话术和隐晦表达适应能力不足；SVM 分类器在已知话术上表现较好，但对新型诈骗话术的泛化能力有限；基于大模型的语义分析方法则在保持较高检测准确率的同时，能够输出可解释的判定理由。

在自构建语音数据集测试集上，系统整体诈骗检测准确率达到 91.6%。其中，对冒充公检法、贷款刷流水、虚假投资理财等典型诈骗话术的识别效果较好，相关关键短语召回率超过 95%。部分误判主要出

现在诈骗话术较为隐晦、尚处于铺垫阶段的文本中，说明仅依赖单轮文本内容进行判定仍存在一定局限性。

为验证分层检测策略的工程有效性，本文额外构建了一个包含 20 条文本样本的补充测试集，其中诈骗文本和正常文本各 10 条，覆盖冒充公检法、冒充客服、贷款刷流水、刷单返利、内部投资、高收益诱导、物流通知、账单提醒、工作沟通和反诈宣传等典型场景。实验结果显示，优化后的检测流程在该补充测试集上取得了 100%的分类准确率，20 条样本均被正确分类。其中，9 条边界清晰的样本可由本地规则直接完成判定，响应时间约为 0.00~0.03 s；其余 11 条语义较复杂的样本由大模型完成判定，耗时约为 0.69~0.88 s；综合全部样本后，平均响应时间约为 0.38 s，结果如表 1 所示。

Table 1. Summary of supplementary experimental results of the hybrid fraud detection strategy
表 1. 融合检测策略补充实验结果汇总

指标	结果
测试样本数	20
诈骗样本数	10
正常样本数	10
正确分类数	20
分类准确率	100%
平均响应时间	0.38 s
规则直接命中样本数	9
大模型语义判别补充样本数	11

需要说明的是，该 100%结果基于小规模典型文本补充测试集，主要用于验证融合检测流程的工程有效性，不能替代更大规模真实语音场景下的整体性能评估。

3.4. 系统响应与用户体验

在 HarmonyOS 真实设备上，对系统进行了功能与性能测试。在 Wi-Fi 环境下，选取 20 条不同时长的音频样本进行端到端响应测试。每条样本重复检测 5 次，记录从点击“识别并检测当前音频”到前端接收到完整结果的时间间隔，并取平均值作为系统响应时延。测试结果表明，在 Wi-Fi 环境下，系统平均端到端响应时延约为 2 s；在 4G/5G 环境下，平均耗时略有增加，但大多数请求仍可在 3 s 内完成。

在连续 100 次检测操作中，系统未出现崩溃、卡死或异常退出，检测成功率达到 100%，表明系统具有较好的稳定性。从流程看，端到端时延主要由音频上传、格式转换、语音识别调用、诈骗文本检测和结果回传构成，其中语音识别与大模型推理是主要耗时环节。不同判断路径下响应时间对比如表 2 所示。

Table 2. Response time comparison under different decision paths
表 2. 不同判定路径下的响应时间对比

判定路径	样本数量	响应时间范围
本地规则直接判定	9	0.00~0.03 s
大模型语义判别补充	11	0.69~0.88 s
全部样本平均	20	0.38 s

为评估用户体验，本文对 10 名体验用户进行了问卷调查。结果显示，90%的用户认为应用界面简洁、操作流程清晰，能够方便地查看识别与检测结果；多数用户认为系统给出的风险等级与判定理由有助于快速理解风险来源。与此同时，部分用户反馈在语速较快、口音较重或背景噪声较大的场景下，语音识别误差仍会影响后续判定效果；在弱网环境下，音频上传与云端推理耗时也会有所增加。

4. 讨论

4.1. 方法效果分析

本文实现的基于 HarmonyOS 的端云协同语音反诈系统在系统架构设计、关键模块实现和实验验证方面取得了较好效果。实验结果表明，系统在自构建语音数据集上的整体诈骗检测准确率达到 91.6%，在 Wi-Fi 环境下平均端到端响应时延约为 2 s，能够满足日常演示与轻量化应用场景的需求。补充实验进一步表明，通过在后端引入“本地规则预筛 + 大模型语义判定 + 结果缓存”的优化策略，文本级检测平均响应时间可降低至 0.38 s，说明该融合机制能够较好兼顾准确率与实时性。

从检测模型角度看，关键词规则响应速度快但适应性有限，大模型语义分析能力强但调用成本较高。本文采用的融合检测方案通过规则模块快速识别典型高危特征和明确正常场景，仅对歧义文本调用大模型，从而形成较好的功能互补。不过，当前系统仍主要基于语音转写后的文本进行检测，尚未充分利用语速、语调、停顿分布和情绪状态等原始语音信息，因此对诈骗铺垫阶段或隐晦表达场景的识别仍存在局限。

此外，系统对网络环境仍存在一定依赖。语音识别和语义检测主要依赖云端服务，在弱网或无网环境下性能会明显下降。未来可考虑在端侧部署轻量级语音识别与诈骗分类模型，使系统在高频场景下具备初步离线预警能力。与此同时，语音反诈系统涉及通话内容上传与处理，隐私与数据安全问题同样重要。本文当前方案已采用 HTTPS 传输、Token 鉴权以及“处理后即释放”的策略，后续仍需结合匿名化、访问控制和日志脱敏等手段进一步完善安全机制。

4.2. 当前音频获取方式的局限性与实时通话流处理展望

需要指出的是，本文系统当前的音频输入方式主要包括内置示例音频、本地媒体文件以及网络音频 URL，检测流程面向的是已获取的短音频片段，尚未直接接入真实通话过程中的系统级音频流。因此，本文实现更适用于反诈教学演示、录音复核、通话后分析以及准实时风险检测场景，而距离“在用户通话过程中自动持续监听并实时预警”的实际应用目标仍存在一定差距。

造成这一差距的原因主要包括以下几个方面：

1) 系统接口与权限限制。在移动操作系统环境下，实时获取通话双向音频流通常受到严格的系统权限、隐私策略和平台接口开放程度限制，普通应用难以直接、稳定地访问通话原始音频；

2) 流式处理链路尚未完全建立。当前系统以整段音频上传和完整转写后再检测为主，尚未实现面向连续音频流的分片、缓冲、增量识别和滚动风险判定机制；

3) 实时性与资源消耗之间仍需权衡。若要在真实通话过程中进行低时延风险提示，需要同时考虑端侧持续采集、网络传输、云端推理、结果回传和前端提示等多个环节带来的时延与功耗开销。

尽管如此，本文提出的端云协同架构仍具备向真实通话场景扩展的可行性。面向未来的“实时通话音频流”处理，可考虑按照“准实时过渡–流式识别增强–端云协同实时预警”的路线逐步实现：

1) 准实时检测阶段：在现有短音频检测流程基础上，将长音频按固定时窗进行切分，例如每 3 s~5 s 生成一个音频片段，并采用滑动窗口方式上传至后端，实现“边采集、边识别、边检测”的准实时处理。该阶段实现难度较低，可作为从离线检测向实时检测演进的过渡方案。

2) 流式语音识别阶段：在后端引入流式 ASR 机制，对连续音频流进行增量转写，并基于当前窗口文本与历史上下文进行联合判定。相比整段音频处理，流式识别能够更早发现“安全账户”“验证码”“冻结资金”“远程协助”等高危话术，从而缩短预警时延。

3) 端云协同实时预警阶段：在端侧部署轻量级初筛模块，对高风险关键词、异常语义模式或高危会话状态进行快速预判，仅将疑似高风险片段上传云端进行高精度复核。该机制可降低云端调用频率与传输开销，并在一定程度上提升弱网环境下的可用性。若未来 HarmonyOS 生态提供更完善的通话音频访问能力，则可进一步实现更接近真实场景的实时语音反诈预警系统。

此外，真实诈骗通话往往具有明显的过程性特征，诈骗分子常先通过身份铺垫、心理施压、制造紧迫感，再逐步引导用户转账或泄露验证码。因此，未来系统还应从单轮文本判定扩展到多轮连续会话建模，综合分析上下文演化、风险累积趋势以及跨片段语义关联，以提升对铺垫型、隐晦型和新型诈骗话术的识别能力。

综上，本文当前系统在音频获取方式上仍以离线或准离线音频为主，尚未完全覆盖真实通话流处理场景；但其模块化架构与融合检测流程为后续向实时化、流式化、连续化反诈预警方向演进提供了较好的工程基础。

4.3. 隐私保护与数据安全分析

语音反诈系统需要处理用户通话内容、语音转写文本及其对应的风险判定结果，涉及较强的个人隐私敏感性。尤其在端云协同架构下，音频数据需要在终端与服务器之间传输，并可能调用第三方语音识别和大模型服务，因此有必要从数据全生命周期角度系统考虑隐私保护与安全治理问题。

从数据处理流程看，系统涉及采集、传输、处理、存储与销毁等关键环节。针对各环节的安全需求，本文采用并建议进一步完善如下策略：

1) 采集阶段的最小必要原则。系统仅在用户主动触发检测操作并明确授权的前提下采集音频数据，不进行后台隐式录音或无感知监听。对于非必要信息不予采集，尽量将数据获取范围限定在风险检测所需的最小粒度，以降低隐私暴露面。

2) 传输阶段的安全保护。前后端通信采用 HTTPS 进行加密传输，结合 Token 鉴权机制对请求来源进行身份校验，降低中间人攻击、接口滥用和非法调用风险。对于上传音频、识别文本和检测结果等敏感内容，应避免通过明文日志、调试信息或不安全缓存进行暴露。

3) 处理阶段的敏感信息控制。在后端进行音频预处理、语音识别与文本检测时，应遵循“处理即使用、最少留存”的原则。对识别结果中可能涉及的身份证号、银行卡号、手机号、验证码等敏感信息，可进一步引入脱敏规则或掩码机制，避免在日志记录、调试输出和结果展示中直接暴露完整内容。对于调用第三方云服务的场景，还需明确接口权限边界、数据用途范围和服务侧安全责任。

4) 存储与缓存阶段的风险约束。本文系统为提高重复检测效率引入了结果缓存机制，但缓存内容不宜直接保存完整原始音频或可逆恢复的敏感文本。更合理的方式是仅缓存文本摘要、哈希索引或匿名化后的中间结果，并设置严格的过期时间和访问控制策略，以防止历史检测数据被滥用或长期积累带来的隐私风险。

5) 销毁阶段的数据闭环管理。对于上传至服务器的临时音频文件和中间处理结果，应在识别与检测完成后及时释放或删除，避免形成不必要的数据沉积。日志数据也应建立定期清理机制，并对包含敏感字段的记录进行脱敏或压缩保留。通过构建“采集有授权、传输有加密、处理有约束、缓存有边界、销毁有机制”的闭环流程，可提升系统整体安全性与可控性。

除生命周期管理外，不同部署模式在隐私保护方面也存在明显权衡。端侧模型方案的主要优势在于

数据可以尽量保留在本地设备完成处理，减少原始音频和文本上传云端的频率，从而降低隐私泄露风险，并提升弱网或离线场景下的可用性。然而，端侧模型通常受限于设备算力、存储空间和能耗预算，其语音识别与复杂语义理解能力往往弱于云端大模型，模型更新与维护成本也相对较高。

相比之下，云端模型方案具备更强的语音识别精度、语义理解能力和持续迭代能力，适合应对电信诈骗话术快速演化带来的识别挑战，但其代价是用户音频与文本需要上传到服务器进行处理，系统对网络环境依赖更强，且第三方服务调用会引入额外的数据合规与隐私保护要求。

综合来看，更具工程可行性的方向是采用“端侧初筛 + 云端复核”的混合隐私保护架构：端侧负责完成音频触发检测、基础关键词预筛、敏感信息局部脱敏以及初步风险判断，仅对疑似高风险片段或低置信度样本上传云端进行高精度识别与语义分析。该方式既可在一定程度上减少数据上传量，又能兼顾识别性能与隐私保护需求。

需要说明的是，本文工作目前主要验证了系统功能流程与检测有效性，对隐私安全机制的实现仍以基础工程措施为主，尚未进一步引入联邦学习、端侧安全执行环境、差分隐私、可验证删除等更高级安全技术。未来可结合 HarmonyOS 分布式能力与端侧智能处理框架，从算法、系统和合规三个层面进一步完善语音反诈系统的隐私保护体系。

5. 结论

本文围绕电信网络诈骗防范需求，设计并实现了一款基于 HarmonyOS 的端云协同语音反诈系统。系统采用前后端分离架构，前端基于 HarmonyOS ArkTS 实现音频播放控制、多音源选择、识别触发和结果可视化展示，后端基于 Python Flask 完成音频预处理、语音识别调用和诈骗文本检测，构建了从“音频输入–语音转写–风险判定–结果反馈”的完整处理链路。

在关键技术实现方面，本文集成百度短语音识别 API 完成语音转写，并结合智谱 GLM 系列大模型实现诈骗语义分析。为提升实际部署效率，进一步设计了“本地规则预筛 + 大模型语义判定 + 结果缓存”的融合检测策略，在保证结果可解释性的同时改善了系统响应性能。

实验结果表明，在自构建的电信诈骗与正常语音数据集上，系统整体诈骗检测准确率达到 91.6%，对“安全账户”“冻结资金”“法院传票”等关键诈骗短语的召回率超过 95%；在 Wi-Fi 环境下，系统平均端到端响应时延约为 2 s，连续 100 次检测成功率达到 100%。补充实验进一步表明，优化后的融合检测流程在 20 条典型文本样本上取得了 100%的分类准确率，平均响应时间约为 0.38 s。总体来看，本文所提出的端云协同语音反诈系统能够有效识别典型诈骗话术，并通过风险等级、置信度、判定理由和安全建议向用户提供可解释的风险提示，具有一定的实际应用价值。

需要说明的是，本文当前系统主要面向内置音频、本地媒体文件和网络音频 URL 等已获取音频的检测分析，尚未直接接入真实通话过程中的系统级实时音频流。因此，现阶段系统更适用于反诈教学演示、录音复核和准实时风险检测场景，与真实通话过程中持续监听、实时预警的应用目标相比仍存在一定差距。未来可在现有架构基础上，进一步引入滑动窗口分片、流式语音识别和连续会话建模机制，逐步实现从短音频检测向实时通话音频流处理的扩展；同时结合端侧轻量级初筛模型与云端高精度语义复核，构建“端侧初筛 + 云端复核”的低时延协同预警方案，以提升系统在真实应用场景中的实用性与可部署性。

此外，语音反诈任务涉及用户通话内容、语音转写文本及风险判定结果等敏感信息，其实际部署不仅要关注检测性能，也必须重视隐私保护与数据安全。后续研究可围绕数据采集授权、加密传输、敏感信息脱敏、缓存控制和处理后销毁等全生命周期安全策略进一步完善系统设计，并综合权衡端侧模型与云端模型在隐私保护、识别精度、计算资源和网络依赖等方面的差异，提升系统的可信性与工程应

用价值。

当然，本文工作仍存在一定局限性。后续可进一步引入端侧轻量级模型、多模态特征融合与更大规模诈骗语料，以提升系统对隐晦诈骗、新型诈骗话术及弱网场景的适应能力。

致　谢

感谢指导教师在论文选题、系统设计与论文修改过程中给予的指导与帮助，感谢实验过程中参与测试与标注工作的同学提供支持。

基金项目

北方工业大学大学生创业项目“端云协同的鸿蒙移动应用研究”(10805136026XN073-334)。

参考文献

[1] 周志华. 机器学习[M]. 北京: 清华大学出版社, 2016.

[2] 李航. 统计学习方法[M]. 北京: 清华大学出版社, 2019.

[3] Goodfellow, I., Bengio, Y. and Courville, A. (2016) Deep Learning. MIT Press.

[4] Russell, S. and Norvig, P. (2021) Artificial Intelligence: A Modern Approach. 4th Edition, Pearson.

[5] Vaswani, A., Shazeer, N., Parmar, N., *et al*. (2017) Attention Is All You Need. 31*st Conference on Neural Information Processing Systems* (*NIPS* 2017), Long Beach, 4-9 December 2017, 5998-6008.

[6] Devlin, J., Chang, M.W., Lee, K., *et al*. (2019) BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding. *Proceedings of NAACL-HLT* 2019, Minneapolis, 2-7 June 2019, 4171-4186.

[7] Brown, T.B., Mann, B., Ryder, N., *et al*. (2020) Language Models are Few-Shot Learners. 34*th Conference on Neural Information Processing Systems* (*NeurIPS* 2020), Vancouver, 6-12 December 2020, 1877-1901.

[8] Radford, A., Narasimhan, K., Salimans, T., *et al*. (2018) Improving Language Understanding by Generative Pre-Training. https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf

[9] Huawei: HarmonyOS Developer Documentation. https://developer.harmonyos.com/