面向真实诈骗通信场景的伪造语音检测研究

徐 哲[1,2], 程 鹏[2,3,4], 巴钟杰[2,3,4], 黄 鹏[2,3,4], 任 奎[1,2,3,4]

[1] 南京理工大学网络空间安全学院 江苏 南京 210094
[2] 浙江大学区块链与数据安全全国重点实验室 浙江 杭州 310027
[3] 浙江大学网络空间安全学院 浙江 杭州 310027
[4] 浙江大学计算机科学与技术学院 浙江 杭州 310027

摘要 近年来, 音频伪造检测研究主要关注提升模型对未知伪造算法的泛化能力, 通常通过优化特征提取结构, 并引入随机噪声、频率扰动等数据增强策略来提高鲁棒性。然而, 在真实应用场景中, 尤其是电信诈骗通信链路中, 压缩编码、网络抖动和终端设备差异等非线性失真会显著干扰检测效果。现有研究中, 数据增强方法在复现此类复杂失真方面的有效性尚不明确, 其在真实场景中提升检测效果的能力仍存疑问。此外, 在公开领域, 针对此类复杂失真场景的伪造语音数据集及系统性评估体系也普遍缺乏。为填补这一空白, 本文设计并实现了一套软硬件结合的真实语音欺诈模拟系统, 能够高保真还原电话与社交软件(如微信)通话过程中的通信失真过程。据此构建了一个真实语音通信链路条件下的大规模伪造语音数据集 1。在此基础上, 本文系统评估了主流伪造检测模型与典型增强策略在链路失真条件下的性能表现。实验结果表明, 虽然数据增强对真实链路数据的检测效果有所提升, 但相较于在原始数据上的表现, 模型在经过真实链路传输后的表现仍显著下降。在引入本文构建的真实链路失真样本进行训练后, 模型性能得到显著提升: 电话链路上的等错误率(EER)由 19.16%降至 6.48%, 微信链路上的等错误率也由 15.76%降至 7.72%。进一步地, 依托 RealLink 数据集, 我们提出并验证了一种基于表征的抗失真检测方法, 通过轻量级恢复模块对受损特征进行修正, 能够一定程度上恢复被链路失真掩盖的伪造痕迹, 并带来额外的性能提升。本研究揭示了通信链路失真对伪造语音检测系统性能的关键影响, 构建了可复现并且开源的高保真链路失真数据集, 并为真实语音欺诈场景下的鲁棒伪造音频检测系统的设计与评估提供了数据支持与方法参考。

关键词 音频伪造检测; 通信链路失真; 电信诈骗; 伪造语音数据集
中图法分类号 TP391; TP183 **DOI 号** 10.19363/J.cnki.cn10-1380/tn.2025.11.14

Spoofed Speech Detection in Real-World Fraudulent Communication Scenarios

XU Zhe[1,2], CHENG Peng[2,3,4], BA Zhongjie[2,3,4], HUANG Peng[2,3,4], REN Kui[1,2,3,4]

[1] School of Cyberspace Security, Nanjing University of Science and Technology, Nanjing 210094, China
[2] State Key Laboratory of Blockchain and Data Security, Zhejiang University, Hangzhou 310027, China
[3] School of Cyberspace Security, Zhejiang University, Hangzhou 310027, China
[4] College of Computer Science and Technology, Zhejiang University, Hangzhou 310027, China

Abstract In recent years, research on audio spoofing detection has primarily focused on improving the generalization ability of models against unknown spoofing algorithms. Common approaches include optimizing feature extraction architectures and introducing data augmentation strategies such as random noise and frequency perturbations to enhance model robustness. However, in real-world applications—particularly in telecommunication fraud scenarios—nonlinear distortions such as codec compression, network jitter, and device variability can significantly degrade detection performance. The effectiveness of existing data augmentation methods in reproducing such complex distortions remains uncertain, raising doubts about their ability to improve detection under real conditions. Moreover, there is a general lack of publicly avail-able spoofed speech datasets and systematic evaluation protocols that reflect these realistic distortion scenarios. To ad-dress this gap, this paper proposes a hardware-software integrated simulation system capable of faithfully reproducing the distortion process in real-world voice communication channels, including telephone calls and social messaging plat-forms such as WeChat. Based on this system, we construct a large-scale spoofed speech dataset under realistic com-mu-nication link conditions. On this foundation, we systematically evaluate several state-of-the-art spoofing detection mod-els and representative data augmentation strategies under transmission-induced distortions. Experimental results show that although data augmentation provides some improvement on real-link data, model performance still drops signi-

通信作者: 程鹏, 博士, 研究员, Email: peng_cheng@zju.edu.cn。
本课题得到国家自然科学基金面上项目(No. 62472372, No. 62172359), 浙江省重大项目(No. LD24F020010), 浙江省“尖兵领雁+X”科技计划项目资助。
收稿日期: 2025-06-30; 修改日期: 2025-11-11; 定稿日期: 2025-11-11

fi-cantly compared to clean conditions. After incorporating the proposed real-link distorted samples into training, the mod-el performance was significantly improved: the Equal Error Rate (EER) on the telephone link decreased from 19.16% to 6.48%, and the EER on the WeChat link decreased from 15.76% to 7.72%. Furthermore, based on the RealLink dataset, we propose and validate a representation-based anti-distortion detection method, in which a lightweight restoration module is employed to refine the degraded features, thereby partially recovering spoofing cues obscured by channel dis-tortions and yielding additional performance gains. This study highlights the critical impact of communication link dis-tortions on spoofing detection performance, presents a reproducible and open-source high-fidelity distortion dataset, and offers both data support and methodological insights for building robust detection systems in real-world voice fraud scenarios.

Key words audio spoofing detection; transmission distortion; telecom fraud; spoofed speech dataset

1 引言

让机器发出与真人无异的自然语音, 一直是语音技术领域的重要目标。过去几十年, 研究者提出了多种语音生成方法, 例如基于录音词典的拼接合成, 以及利用共振峰模型进行频率域的语音构造。这些方法在语音可懂度与合成效率方面取得了一定成效, 但由于缺乏对语音细节和发声机制的精细建模, 生成结果往往带有明显“机械感”, 自然度仍难以媲美真人发声。

随着深度学习技术的发展, 生成式模型的兴起(如变分自编码器 VAE、生成对抗网络 GAN、扩散模型等)显著提升了合成语音在音色、语调与韵律等方面的自然度和真实感。得益于此, 语音生成技术已在多种场景落地, 如为因病失语者恢复语音表达[1]、提升智能语音助手的交互能力, 甚至合成已故亲友的声音以供纪念[2], 展现出积极的社会价值。

与此同时, 该类技术的泛化与易用也导致了伪造音频泛滥的问题, 带来严重的安全风险隐患, 引发了社会各界对语音合成技术伦理安全方面的担忧。攻击者可通过语音合成或语音转换技术, 模仿目标人物的声纹特征并生成任意语句, 用于身份伪装与语音欺诈。已有多起真实案例表明, 诈骗分子利用伪造企业高管的语音指令实施诈骗, 造成数百万甚至上千万的经济损失[3]。早期语音伪造往往依赖大量目标语音数据, 仅对公众人物等拥有公开语料的对象具有实际可行性。近年来, 随着模型能力的持续增强, 尤其是零样本语音克隆技术的发展[4], 攻击者仅需数秒音频即可实现高仿真语音克隆, 极大降低了使用门槛, 扩展了攻击对象范围。

在现实诈骗场景中, 犯罪分子已能通过电话、即时通信等渠道, 利用合成语音实施诈骗。他们常用的诈骗手段包括冒充亲友、伪造领导指令以及假冒银行客服回访等, 以此诱导受害者进行转账、泄露敏感信息等操作。此类基于语音伪造的电信诈骗具有隐蔽性强、传播速度快、受害范围广等特征, 严重威胁个人财产与公共安全。

为应对此类安全挑战, 伪造语音检测(Audio Deepfake Detection)已迅速成为语音安全领域的重要研究方向。近年来, 学术界与工业界提出了大量检测方法, 涵盖频谱分析、声码器建模、自监督学习等不同策略, 旨在提升模型对多种伪造方式的判别能力。与此同时, 多个国际竞赛的设立也推动了该领域快速发展, 例如 ASVspoof 挑战赛(Kinnunen 等[5], Todisco 等[6], Yamagishi 等[7])和 Audio Deepfake Detection(ADD)挑战赛(Yi 等[8-9]), 为研究提供了标准评估框架与开放测试平台。

在实际应用中, 伪造语音检测技术需要有效应对复杂且多变的现实场景。因此, 本文立足于研究伪造语音检测技术在真实诈骗场景中的应用。在充分分析现有音频伪造检测相关算法和真实诈骗场景中音频传播流程后, 发现当前面临两大关键挑战。

1) 算法泛化能力不足: 泛化能力指的是模型在面对训练时未见过的新型算法合成的数据时, 仍能保持良好性能的能力。语音合成技术发展迅速, 从传统拼接与共振峰模型已演进至基于深度学习的生成范式(如 VAE、GAN、扩散模型), 伪造算法多样且迭代迅速。不同算法在声学细节及韵律特征上差异显著, 导致检测模型在“训练集外”样本上性能不稳定, 严重影响泛化能力。

2) 复杂通信链路失真影响显著: 伪造语音在实际诈骗场景中需经电话、社交平台等多种链路传输, 过程中叠加压缩编码、网络抖动、设备端增强与降噪、信道噪声等复杂非线性时变失真, 显著破坏音频关键特征, 使基于理想条件训练的检测模型性能大幅下降。然而, 目前针对链路失真的系统性建模与验证仍严重不足。

针对第一类问题, 已有大量研究通过多源伪造语音数据、多样化特征提取[10-13]、增量学习[14]和多种数据增强策略提升泛化能力, 成为伪造检测研究的重点。针对第二类问题, 相关研究较少。

为填补该研究空缺, 本文设计并实现一套软硬件结合的真实链路复现系统, 并据此构建了首个体现电话和微信通话链路失真的大规模伪造语音数据

集, 覆盖多种合成算法。依托该数据集, 系统评估多种前沿伪造检测模型及数据增强策略, 深入分析链路失真对模型性能的影响, 本文具体贡献如下。

1) 揭示检测模型在电诈场景下的局限性。系统实验表明, 虽然主流多源训练数据集与数据增强策略提升了模型面对训练时未见过的合成算法的泛化能力, 但在真实电诈复杂通信链路失真环境中, 伪造语音检测性能下降, 难以有效应对链路带来的失真影响, 需要加强对于真实场景下伪造检测模型鲁棒性与实用性的系统性研究, 增强其实用性;

2) 软硬件结合的真实链路数据采集系统。设计并实现了一套通用且可配置的电话及社交平台链路复现系统, 支持多种伪造算法与终端设备组合, 为高保真真实失真语音数据采集提供坚实基础;

3) 电诈场景链路失真伪造语音数据集。基于上述系统, 采集并开源了首个涵盖电话和微信通话的链路失真伪造语音数据集, 共计 4 万条, 覆盖多种主流语音合成与转换算法及多样设备配置;

4) 系统化验证所构建数据集对提升模型鲁棒性的效果。利用新数据集, 对多种前沿伪造检测模型及典型数据增强方法进行系统对比, 实验表明仅使用原始数据训练的模型在真实链路下准确率显著下降(如电话链路 EER 达 19.16%)。引入真实失真样本后, 模型在电话和微信链路上的 EER 分别降至 6.48%和 7.72%, 检测性能提升超 50%。

本文接下来的结构安排如下: 第 2 节回顾音频伪造检测领域在算法、数据集和数据增强的相关工作; 第 3 节详细介绍所提出的软硬件协同链路复现系统及数据采集流程, 并给出电信诈骗场景失真伪造语音数据集的构建细节; 第 4～第 5 节开展系统实验, 对比主流检测模型与典型数据增强策略在真实链路失真条件下的性能表现; 第 6 节对全文工作进行总结, 并讨论后续可能的研究方向与应用扩展。

2 相关工作

本文聚焦于真实欺诈场景下的音频伪造检测与鲁棒性问题, 相关工作主要包括三个方面: 检测算法、数据增强与鉴伪数据集。

2.1 音频伪造检测

近年来, 音频伪造检测技术持续演进, 研究主要集中在三类方法: 传统流水线结构、端到端模型以及自监督预训练方法[15]。早期方法采用预处理、特征提取和分类器组合的流水线结构, 利用 MFCC、LFCC 等特征或深度网络提取音频伪造痕迹。端到端方法则通过直接处理原始波形或频谱, 实现特征提取与判别的一体化优化, 代表模型如 Raw-GAT-ST[15]和 AASIST[16]在公开数据集上取得领先性能。

近年来, 最优检测系统普遍采用“预训练大前端+后端分类器”的双模块架构[17]。该方法在多项国际评测中取得领先, 主要得益于预训练前端在数万小时的多语种、多说话人语音上进行自监督训练, 从而学习到通用且可迁移的声学与语音学表示。这些表示不仅涵盖底层频谱特征, 还蕴含高层音素、韵律乃至跨语种信息, 具备较强的跨域迁移性与鲁棒性。相比之下, 后端分类器可针对具体任务进行设计, 如卷积网络、图神经网络或序列建模模块, 以强化伪造痕迹的捕捉与判别能力。前后端的分工与协作, 使模型既能继承大规模预训练模型的特征表达优势, 又能通过定制化后端适应特定检测需求。

在此框架下, 现有研究主要沿两条路径优化: 其一, 改进后端结构。例如, AASIST 借助图注意力机制建模时频关系, 后续工作进一步引入 Mamba 等新兴序列架构, 使模型在长序列条件下更好地捕捉潜在伪造特征。其二, 优化前端特征的选择与融合。已有研究表明, 预训练模型的不同层蕴含互补的声学与语言学信息, 仅使用顶层表示难以充分利用潜在特征。为此, 提出了跨层特征选择(Sensitive Layer Selection, SLS)方法[17], 通过在多层间自适应选择与融合表示, 提升模型对伪造特征的敏感性。目前表现最优的代表性模型包括 XLSR-Mamba[18]和 XLSR-SLS[19]。前者在结构上结合了 XLS-R 前端与 Mamba 分类器, 利用 Mamba 对长时序列的高效建模能力, 更好地处理跨语句的伪造痕迹; 后者则在 XLS-R 前端上引入 SLS 机制, 自适应选择不同层次的表示进行融合, 以捕捉更细粒度的伪造线索。两者在 ASVspoof 和 ADD 等国际挑战赛中均取得了最优或接近最优的成绩, 体现出在该架构上的优化已成为推动检测性能提升的关键手段。

总体来看, 基于 SSL 的检测方法在检测精度与泛化能力上显著优于传统方法, 其优势主要源于大规模自监督预训练所带来的通用性与可迁移性。然而, 尽管这些方法在公开数据集上表现优异, 但在面对全新生成算法以及真实通信链路失真(如压缩编码、网络抖动、终端设备差异等)时, 检测性能依然存在明显退化。未来研究亟须进一步结合真实链路的失真特征, 设计更具针对性的鲁棒建模与特征恢复方法, 从而提升伪造语音检测系统在实际诈骗场景中的适用性与实用价值。

2.2 数据增强

在音频伪造检测任务中, 数据增强已成为提升模型鲁棒性与泛化能力的关键手段。常见的数据增强策略包括遮挡类方法(如 SpecAugment[20]), 通过在声谱图中随机遮挡时间或频率区域, 强化模型对局部缺失的适应能力; 混合增强方法(如 SpecMix[21])通过剪切并融合不同样本片段生成新数据, 以缓解过拟合并提升对未知伪造的辨识能力; 编解码增强通过模拟多种音频压缩格式(如 MP3、AAC、OGG)及通信协议对语音信号进行处理, 以提高模型对链路压缩失真和传输噪声的鲁棒性。此外, 还包括速度扰动、时间拉伸、音高偏移、添加背景噪声及房间冲激响应等通用方法, 这些策略常在线上或离线组合使用, 以增强模型对说话人差异、设备变化及环境干扰的适应能力。

2.3 音频鉴伪数据集

目前常用的伪造音频数据集包括 ASVspoof 系列(如 ASVspoof2019-LA、2021-LA 和 2021-DF), 其中 LA 子集主要包含由 TTS 和 VC 算法合成的未处理语音, 2021-LA 和 DF 子集则进一步引入了通信链路编码失真和多源伪造算法, 提升了评估的真实性和多样性。其他如 WaveFake、In-the-Wild(ITW)和 FMFCC-A[22-24]数据集则从 GAN、社交媒体采集和中文伪造样本等角度补充了伪造数据的覆盖面, 具备更强的跨语言与跨域能力。

在真实音频数据方面, VCTK、LibriSpeech、LJ Speech 和 AISHELL-3[25-28]提供了多语种、多说话人及高保真语音样本, 是语音合成与伪造检测中常用的真实语音对照数据。总体而言, 虽然现有数据集在伪造技术和语种上不断扩展, 但对真实链路失真条件下的伪造语音仍缺乏系统覆盖, 限制了实际。

3 真实诈骗通信链路下伪造语音采集系统与数据集构建

3.1 真实诈骗通信链路下伪造语音采集系统

为深入探究真实诈骗场景中复杂通信链路失真对伪造语音检测的影响, 并弥补现有研究中缺乏针对此类失真数据采集系统的不足, 本节详细介绍本文设计的真实诈骗通信链路下伪造语音采集系统。

在真实的电信诈骗场景中, 伪造语音并非直接以原始音频形式传递给受害者, 而是通常通过电话网络或社交平台(如微信)进行远程传播。这一过程包括音频生成、通信链路传输和终端接收三个阶段, 其间会引入多种不可控的链路失真因素。与实验室环境中常见的“干净”语音数据相比, 真实链路下的伪造语音往往伴随着语音编码压缩、网络传输扰动和设备回放失真等复杂干扰。这些非线性失真严重影响了伪造检测模型的判别能力, 显著降低了检测系统在实际部署中的准确性和鲁棒性。

为了系统地研究链路失真对伪造检测性能的影响, 并构建一个具有现实可复现性的评估基准, 本文设计并实现了一套软硬件结合的语音采集系统。该系统模拟了伪造语音在诈骗过程中通过通信链路传输的真实路径, 覆盖了电话通话和微信语音这两种常见的传播方式, 能够有效采集带有自然链路失真的伪造音频样本, 为后续检测模型的鲁棒性评估提供了现实场景支持。

音频生成阶段: 在生成端, 攻击者通常使用文本到语音(TTS)或语音克隆(VC)等算法生成伪造语音, 或直接播放已有的伪造录音。为了精准模拟诈骗者的音频注入方式, 本文在数据采集过程中采用声卡将伪造语音信号以音频电平形式直接注入手机的音频输入接口。这一方法绕过了麦克风和声学空间传播, 排除了环境噪声和设备麦克风非线性响应的干扰, 更准确地还原了电信诈骗中“设备到设备”数字注入的物理机制。

通信链路传输阶段: 音频信号进入通信链路后, 会经历语音编码、打包传输及解码回放等多个步骤。在语音编码阶段, 通信系统采用多种有损压缩算法(如 Opus、G.711 等)对语音信号进行压缩以适配带宽资源, 这一过程会造成频带削减、量化误差、预加重滤波等结构性失真。进入网络后, 音频打包面临时延抖动、丢包、乱序及动态码率调节等网络扰动问题, 这些扰动具有高度时变性和不可预测性, 进一步改变了音频的时频结构。

在接收端, 移动设备(如手机 B)会对语音数据进行解码与回放。为提升听感, 终端通常集成回声消除、自动增益控制、噪声抑制等语音增强模块, 这些算法依赖厂商软硬件堆栈, 具有明显的设备相关性与黑箱性, 其非线性处理会重构语音包络并改变相位与能量分布, 对伪造检测模型构成挑战。此外, 不同品牌及语音应用(如微信通话、拨号通话)在编码、传输与增强策略上均存在差异, 进一步加剧了链路失真的多样性与复杂性。

为全面采集通信链路中的音频失真特征, 本文在接收端同样使用声卡以电平信号形式采集手机耳机输出信号, 实现了音频信号“电平注入-链路传输-电平采集”的闭环采样过程。该方案规避了声学空间回放/录制的不确定性, 保证了通信链路中每一个环

节的真实作用得以保留和观测, 为后续高保真伪造数据集构建、失真建模以及模型鲁棒性验证提供了关键基础设施。真实诈骗通信链路中语音传播与采集的总体流程如图 1 所示。

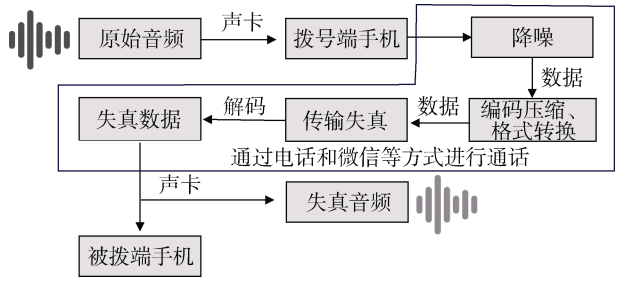

图 1　真实诈骗通信链路中语音传播与采集路径

Figure 1　Speech propagation and recording path in real scam communication chains

为直观展示通信链路对真实与伪造语音信号的影响, 使用前文提出的软硬件结合的采集系统, 将原始真实语音与伪造语音分别通过微信语音与电话通话两类典型链路进行传输, 采集通过链路后的失真样本, 并绘制原始音频和失真后音频的时域波形图与 Mel 频谱进行对比分析, 如图 2 和图 3 所示。

从时域图对比看, 无论电话链路还是微信链路, 传输后的语音信号振幅均有所增加, 且波形变得更加不规则。其中, 电话链路传输后的失真程度更为显著, 表现为信号振幅波动剧烈、波形断裂感更强, 整体能量分布更加紊乱。这可能与电话链路中采用的低比特率编码压缩、高频带裁剪以及模拟/数字转换造成的信号畸变有关。

图 3 显示, 通信链路显著破坏了语音的频谱结构。在高频区域(>4 kHz), 电话和微信链路均出现能量衰减与频带压缩, 使频谱更稀疏不均匀。这一现象在真伪造语音中均存在, 差异图中红蓝条纹更密集, 其中红色表示传输后能量高于原始信号, 蓝色则相反, 说明链路对高频伪影具有较强掩蔽效应, 而高频能量削弱会使模型难以捕捉关键伪造痕迹。

在中低频段(1–3 kHz), 差异谱显示局部能量增强或减弱, 改变主频区平滑性与动态特征, 增加判别难度。总体来看, 两类链路均显著改变真伪语音的时频结构, 既削弱原始特征, 又引入噪声与扰动, 掩盖关键伪造痕迹, 提高了检测鲁棒性要求。

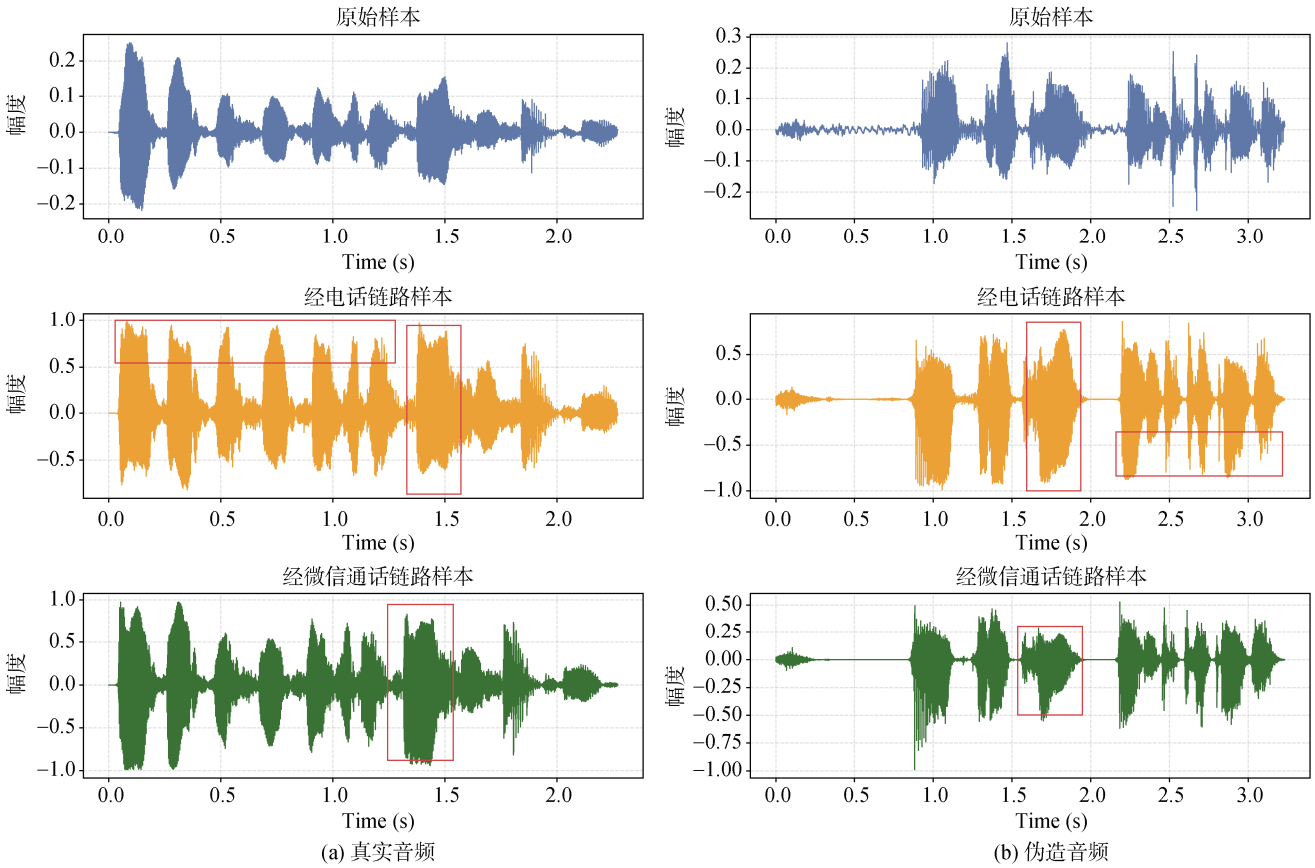

图 2　原始真实和伪造语音与经电话链路、微信链路传输后语音的时域波形对比

Figure 2　Time-domain waveform comparison between original speech and its versions transmitted over telephone and WeChat channels

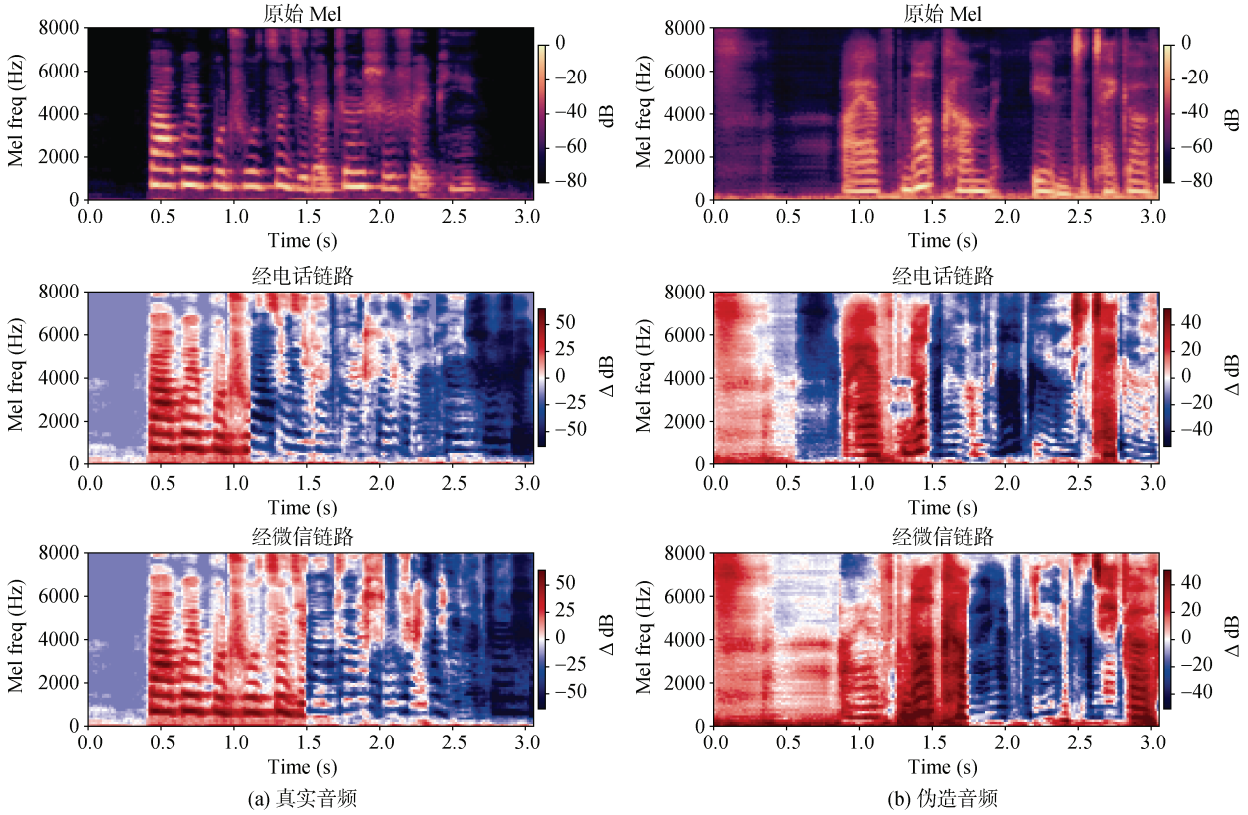

图 3　原始语音及其与电话链路和微信链路的差异频谱

上图为原始语音的 Mel 频谱，中图和下图分别展示了与电话链路、微信链路语音的差异。

Figure 3　Original speech and its differences from telephone and WeChat

The top panel shows the original Mel spectrogram, while the middle and bottom panels illustrate the differences with telephone and WeChat, respectively

综合来看，两类链路均显著改变真伪语音的时频结构，既削弱原始特征，又在频谱空间引入噪声与扰动，掩盖关键伪造痕迹并提高检测鲁棒性要求。为进一步分析链路对伪造语音特征的影响，本文从统计角度对 RealLink 数据集中原始语音与链路语音进行了量化对比。选取了两类典型指标：其一为 MFCC 动态范围(ΔMFCC-dyn)，用于衡量梅尔倒谱系数在时间维度上的变化幅度，反映语音时序细节的保真度；其二为归一化频谱熵(Entropy)，用于表征频谱能量分布的复杂度。结果显示，电话与微信链路下的 ΔMFCC-dyn 分别增加约 3.0 和 4.1，标准差较大，表明链路引入的非平滑失真使部分样本幅度变化更剧烈；频谱熵略有上升(电话：-0.027，微信：-0.050)，说明链路在丢失高频细节的同时引入噪声与量化伪影，增加频谱无序度。

结果表明，通信链路对伪造特征的破坏具有双重性：一方面削弱中高频细节，掩盖伪造语音中的关键伪影线索；另一方面在特征空间引入新的扰动，增加判别复杂度。可视化结果进一步揭示，链路失真导致高频能量衰减与频带压缩，掩盖了伪造语音中常见的伪影频段与不自然的频率过渡。定性与定量分析一致表明，通信链路不仅引入额外噪声，还在特征层面削弱并掩盖关键伪造痕迹，显著增加模型的检测难度。同时，真伪语音经链路传输后其时频结构均发生明显变化，说明链路失真对音频质量及伪造特征的破坏具有普遍且不可忽视的影响。

3.2　真实诈骗场景伪造语音链路数据集设计与构建

在进入数据集构建细节之前，本研究首先对真实诈骗场景下伪造语音的典型特征进行分析，以阐明数据集设计的动机，并进一步说明其在覆盖范围和真实性方面相较现有数据集的优势。与一般研究中常见的干净条件下合成语音不同，诈骗所使用的伪造语音来源更加多样：既可能利用开源算法生成，也可直接调用商用语音合成平台产出，还可能经过不同声码器的再处理，导致其声学表现存在多样化。

此外，诈骗行为往往面向不同人群，可能涉及多种语言，而现有公开数据集多数局限于单一语种。更为关键的是，在实际传播过程中，这些伪造语音通常通过电话或微信等通信链路传输，在语音编码

压缩、网络抖动和终端设备差异等多重因素作用下叠加失真, 使得检测任务更加复杂。

相比之下, 现有的 ASVspoof、In-the-Wild、FMFCC-A 等数据集虽然涵盖了一定范围的 TTS 或 VC 语音, 但在合成手段全面性、商用算法覆盖、链路失真建模以及多语种适配等方面仍存在不足。

针对上述问题, 本文在设计数据集时特别强调多样性与真实性: 其一, 引入 56 种开源与商用合成算法, 不仅覆盖 TTS、VC 和多类声码器, 还包括 7 种常用且高质量的闭源商用系统, 确保合成方式的广度与代表性; 其二, 采集中英文双语样本, 以满足跨语种诈骗检测的需求; 其三, 利用软硬件结合的采集系统, 通过电话与微信链路真实构建失真样本, 覆盖多种终端设备与应用环境。依托上述设计, 本文数据集在合成算法数量、语种覆盖、链路失真和设备多样性等方面均显著优于现有公开数据集, 更加贴近真实诈骗语音的实际分布。

3.2.1 自构伪造语音样本

考虑到真实诈骗场景中伪造语音来源的多样性, 合成样本既可能由开源算法生成, 也可能直接调用主流商用语音合成平台产出。为全面覆盖这一现实分布, 本研究在自构数据集中引入了多类开源与商用伪造语音生成方式。自构伪造语音样本总计约 7,900 条, 覆盖 56 种不同伪造算法。开源部分包含 46 种复现的 TTS 与语音转换算法, 涉及隐变量模型、生成对抗网络(GAN)、扩散模型以及端到端深度神经网络等多种声学建模范式, 体现了各算法在特征建模与语音生成策略上的差异。概况如表 1 所示, 其中 12 种算法支持中英文双语, 针对每种算法与语种组合生成约 100 条样本, 共计约 6,000 条, 确保在算法类别和语种维度上的均衡性与代表性。

表 1 用于生成真实诈骗场景数据的自构伪造语音样本数据集构成

Table 1 The Composition of the Custom-Constructed Spoofed Speech Dataset for Real-World Scam Scenarios

数据类别	算法种类数	伪造数据数量	
		中文数量	英文数量
文本转语音算法	27	600	2900
声音克隆算法	19	600	1900
声码器算法	3	300	300
商用合成算法	7	600	700
总和	56	2100	5800

商用部分采集了 7 种语音合成服务(如百度、讯飞、AzureTTS 等)生成的伪造语音样本。这些系统集成了成熟的声学建模与语言优化策略, 所生成语音自然度高, 更接近真实语音, 检测难度亦更大。商用服务大多支持中英文双语, 分别随机生成约 100 条样本, 总计约 1,300 条。为分析不同声码器对伪造音频特征的影响, 数据集中还包含三类典型声码器(BigVGAN、Griffin-Lim、DiffWave[29-31])生成的中英文样本各 100 条, 共 600 条。这些样本在时域与频域上呈现出不同类型的伪影特征, 为探讨生成机制与检测难度之间的关系提供了丰富素材。

3.2.2 开源数据集样本

开源数据集样本部分包含约 12,100 条音频, 其中伪造样本约 4,100 条, 真实语音约 8,000 条。数据来源涵盖多个常用真伪语音数据集, 如 In-the-Wild、ASVspoof 2021 DF[7]、FMFCC-A[24]、AISHELL-3[28]、THCHS-30[32]、LibriTTS[33]和 TIMIT[34], 覆盖多种合成算法、语种类型与说话人特征, 具体数量如表 2 所示。该部分数据在语音内容和伪造手段上分布广泛, 是当前语音伪造检测研究中的常用数据。

表 2 用于生成真实诈骗场景数据的开源数据集构成

Table 2 Composition of Public Speech Datasets Used for Generating Realistic Scam Scenarios

数据集名称	真实数据量		伪造数据量	
	中文数量	英文数量	中文数量	英文数量
FMFCC-A	500	/	1500	/
In-the-Wild	/	1000	/	1100
ASVspoof2021-DF	/	500	/	1500
AISHELL-3	2000	/	/	/
THCHS-30	1000	/	/	/
LibriTTS	/	2000	/	/
TIMIT	/	1000	/	/
总和	3500	4500	1500	2600

引入开源数据旨在基础伪造语音数据集构建中补充多样化的真实与伪造样本, 弥补自构语料在语音内容、说话人和技术类型上的覆盖不足, 同时增强整体数据集的开放性与可复用性, 为后续链路失真采集与模型评估提供更全面的原始语料基础。

3.2.3 真实链路伪造语音数据集构建

在链路数据采集中, 系统采用多种品牌与型号的通话设备, 涵盖主流通信软件, 全面模拟真实环境中的多样化链路失真场景。具体流程中, 首先将若干条语音样本按固定时间间隔(如每 5 秒)进行拼接, 构建为连续的长音频片段; 随后通过电话或即时通信软件完成一次性传输与录音采集, 真实引入通信链路中的编解码压缩、网络抖动、终端设备播放等

多种失真因素。采集完成后, 录制的长音频被重新切分回原始粒度的语音样本, 确保样本标注准确且语音单元独立。该拼接–传输–还原流程在提升采集效率的同时, 有效保留了链路中复杂失真对每条语音的真实影响。

本数据集覆盖中文和英文双语场景, 包含多种语音合成技术及真实链路失真, 样本总规模为四万条, 其中微信通话样本两万条, 电话通话样本两万条。设计初衷是弥补现有伪造语音检测研究中缺乏针对真实链路失真的系统性评估与分析的不足。当前多数伪造检测模型训练与评测仍依赖于理想化条件下的公开数据集, 难以充分反映实际电信诈骗环境中的复杂链路失真, 导致模型鲁棒性和泛化能力难以有效验证。通过引入多源伪造算法、多语种多说话人样本及真实通信链路, 该数据集真实反映了多重失真对伪造语音检测的综合影响, 为后续模型在链路失真场景下的鲁棒性评估与优化提供了基础数据支撑, 有助于更真实地反映伪造检测方法在实际通信环境中的表现。

需要指出的是, 本文目前的链路采集实验主要聚焦于电话通话与微信电话两类最常见且在诈骗场景中最具代表性的通信方式。这两类链路在编码压缩、网络抖动及终端设备差异方面均具有高度代表性, 能够真实反映诈骗环境下的典型失真条件。其他通信渠道(如 QQ 通话或在线会议平台)虽然尚未直接纳入本次采集, 但所提出的软硬件结合采集系统具备通用性, 可在未来灵活扩展至更多网络传输方式。因此, 本研究不仅覆盖了诈骗中最核心的通信链路, 还为后续在更广泛的通信环境下开展鲁棒性研究提供了可扩展的实验基础。

为进一步体现本文所构建数据集的价值, 有必要与现有常用伪造语音数据集进行对比。现有 ASVspoof 系列虽在国际竞赛中被广泛使用, 但主要集中于 TTS 与 VC 合成语音, 缺乏商用算法和声码器生成样本, 合成手段覆盖有限; In-the-Wild 数据集包含部分来自社交媒体的真实伪造语音, 但来源不透明且链路失真特征不可控, 难以作为系统性评估标准; FMFCC-A 虽引入中文伪造样本, 但整体规模有限, 对跨语种检测的支撑作用仍不足; ADD 系列挑战赛数据集更关注多源合成算法的覆盖, 但同样主要基于干净条件下的伪造语音, 缺乏真实链路传输带来的失真建模。相比之下, 本文构建的 RealLink 数据集在设计时更注重多样性与真实性, 不仅涵盖 56 种开源与商用算法, 覆盖 TTS、VC 及多类声码器, 还特别引入七种常用且高质量的商用合成系统, 保证语音自然度与仿真度。数据集包含中英文语音, 满足跨语种检测需求; 并通过软硬件结合方式, 在电话与微信链路中完成真实传输与采集, 有效保留编码压缩、网络抖动和设备差异带来的复杂失真。此外, 采集过程使用多品牌、多型号终端设备, 使数据分布更贴近实际诈骗环境。总体来看, RealLink 数据集在合成方式、语种覆盖、链路失真建模和设备多样性等方面均显著优于现有公开数据集, 更能真实反映诈骗场景中伪造语音的分布与特征。

为便于统一描述和后续实验分析, 本文对构建的真实链路伪造语音数据集采用如下命名方式: 其中, RealLink-Clean 表示未经过任何链路传输的原始伪造语音样本; RealLink-Distorted 表示所有经过链路传输后产生失真的语音样本。根据不同的传输方式, RealLink-Distorted 又可细分为两部分: 经电话通话链路采集得到的版本记作 RealLink-Call, 而经微信语音通话链路采集得到的版本则命名为 RealLink-WeChat。上述命名贯穿实验部分, 用于区分链路前后及不同类型链路下的音频样本, 完整的数据集构成信息详见附表 1。

4 实验设置

4.1 数据集和评价指标

本文采用 ASVspoof 2019 LA 数据集的训练集与开发集进行模型训练与验证。测试阶段在多个具有代表性的伪造检测基准数据集上进行, 包括 ASVspoof 2021 LA、ASVspoof 2021 DF, 以及更具现实复杂性的 In-the-Wild 数据集。同时, 为进一步模拟真实诈骗语音传播过程, 实验引入了本文构建的真实诈骗场景伪造语音链路数据集, 覆盖电话与社交平台(如微信)两类典型通信链路失真环境。

模型性能评估采用三项常用指标: 等错误率(Equal Error Rate, EER)、最小归一化串联检测成本函数 (Minimum Normalized Tandem Detection Cost Function, min t-DCF)与准确率(Accuracy)。其中, EER 衡量模型在二分类任务中假正例与假反例概率相等时的整体判别能力, min t-DCF 评估模型与说话人验证系统串联部署时的综合检测成本, Accuracy 反映模型对所有测试样本的分类准确率。在后续分析中, 主要报告 EER, 并结合 Accuracy 与 min t-DCF 作为辅助指标, 以提供完整的性能评估视角。

4.2 数据增强

为了系统评估常见数据增强方法在真实诈骗通信链路场景下对伪造语音检测模型鲁棒性的提升效果, 本文选取了三类在公开研究中具有代表性的数

据增强策略进行对比分析, 分别为 RawBoost[35]增强、SpecAugment[20]时间-频率遮挡以及基于MUSAN[36]噪声库的加性噪声增强。这些方法已广泛应用于各类伪造检测任务, 且被证实在标准测试集上能有效提升模型的泛化能力。然而, 在面对通信链路中编码压缩、网络抖动等非理想传输失真时, 其实际效果仍缺乏系统性验证。三种增强方法如下。

1) RawBoost 是一种模拟设备与信道失真的信号扰动方法, 结合线性与非线性的卷积噪声、脉冲式信号相关噪声以及与信号无关的平稳加性噪声, 适用于模拟真实通信中复杂的物理干扰;

2) SpecAugment 通过在语音的梅尔频谱图上施加时间遮挡和频率遮挡, 模拟语音信号在频谱域的不完整性, 增强模型对频段缺失的鲁棒性;

3) MUSAN 加噪则利用 MUSAN 噪声库中的环境噪声(如人群、背景音乐等), 对语音信号进行加性噪声处理, 以提高模型对背景干扰的适应能力。

具体增强策略设置如下: 在训练过程中, 每条语音样本在送入模型前, 首先通过增强策略判定模块。根据预设比例, 50%的样本保持原始形式, 另50%的样本将进行数据增强。对于被选中进行增强的样本, 从预定义的k种增强方法中随机选择一种进行处理, 以保证不同增强策略在训练数据中的分布均衡。该机制在不增加额外样本数量的前提下, 引入多样的扰动, 用于系统评估不同数据增强策略在复杂链路失真场景下对检测模型性能的实际影响。

4.3 模型训练参数设置及训练方式

本研究采用当前主流的 Wav2Vec2.0 与 AASIST 相结合的架构作为检测模型[35], 该组合在现有伪造语音检测任务中表现优异, 且已被广泛用于衍生模型的结构改进与鲁棒性增强研究, 具有较强代表性与可扩展性。所有训练语音样本均以 16 kHz 采样率读取, 并进行预处理: 对于长度不足 4 秒的样本进行零填充, 超过 4 秒的样本则裁剪至固定长度, 确保输入特征长度一致。训练过程中, 模型使用初始学习率为 1×10^{-6}, 在 NVIDIA RTX 3090 GPU 上进行共 50 个 epoch 的训练。

5 实验结果及分析

本章共设计六组实验, 用于系统评估语音伪造检测模型在真实诈骗场景下的鲁棒性表现以及所提出数据集的有效与必要性。首先, 第一组实验分析通信链路失真对检测性能的整体影响, 用于确认在真实链路条件下模型性能是否会显著退化。在此基础上, 第二组实验进一步检验常用数据增强策略能否有效模拟真实链路失真, 并评估其对模型鲁棒性的提升作用。由于不同链路(如电话与微信)可能引入的失真特性差异较大, 第三组实验对比了不同链路类型下模型检测性能的表现, 从而明确链路类型的特殊性。随后, 第四组实验更细致地考察链路失真对真实语音与不同伪造方式的影响差异, 以揭示性能变化在不同语音来源上的分布规律。在分析链路失真对模型检测性能带来的影响后, 第五组实验引入我们提出的数据集, 评估其在链路失真检测中的作用与价值。最后, 第六组实验与现有抗失真方法进行对比, 比较这些方法与数据集在提升检测性能方面的表现差异。

5.1 通信链路失真对模型性能的影响

为了评估通信链路失真对伪造语音检测模型性能的影响, 本文选取多个测试集进行性能对比分析, 涵盖 ASVspoof 2019 LA(开发集与评估集)、ASVspoof 2021 DF、In-the-Wild 数据集, 以及本文构建的链路传输前的原始伪造语音数据集与经过微信或电话等通信方式传输后的链路失真数据集。实验结果如表 3 所示。

表 3 XLSR-AASIST 模型在不同测试集上的检测性能

Table 3 Detection Performance of the XLSR-AASIST Model on Different Test Sets

测试数据集名称	XLSR-AASIST w/o augmentation		XLSR-AASIST w augmentation	
	min t-DCF	EER	min t-DCF	EER
ASVspoof 2019 LA (Dev)	0.0043	0.18%	0.0325	0.89%
ASVspoof 2019 LA (Eval)	0.0110	0.58%	0.0071	0.3%
ASVspoof 2021 DF (Eval)	0.3623	6.72%	0.2568	3.34%
In-the-Wild	-	12.3%	-	7.55%
RealLink -Clean	-	27.9%	-	20.8%
RealLink -Distorted	-	33.3%	-	24.0%

从中可以看出, XLSR-AASIST 模型在传统的 ASVspoof 测试集上表现优异, 尤其是在 ASVspoof 2019 LA 和 2021 DF 两个标准评估集上, 数据增强显著提升了模型性能, EER 分别从 0.18%上升至

0.89%、从 6.72%降至 3.34%, 说明增强策略对于模型泛化和应对轻度失真具有积极作用。然而, 当测试集切换至更具现实复杂性的 In-the-Wild 和 RealLink 数据集时, 模型性能出现明显退化。尤其是在 RealLink-Clean 与 RealLink-Distorted 两个子集中, 即便采用增强策略, EER 仍高达 20.8%和 24.0%, 显著高于公开测试集, 表明模型在真实通信条件下的鲁棒性不足。前者虽未经过链路传输, 但因涵盖更多伪造算法与说话人特征, 仍对检测构成挑战; 后者在此基础上叠加微信、电话等非线性失真, 进一步加剧了难度。这说明现有数据增强手段在模拟实际环境中的多重失真方面仍有限, 难以充分覆盖编码压缩、带宽裁剪与设备回放等复杂因素。

综上所述, 该实验验证了本文所构建的 RealLink 数据集在真实性与挑战性方面的重要价值, 也揭示了当前伪造检测模型在现实部署中的性能退化问题。后续研究需进一步针对链路失真特征设计更有效的增强或补偿策略, 以提升系统在通信场景下的适应能力。考虑到实际应用中检测系统往往缺乏测试集标签, 无法动态设定最优阈值, 传统 EER 指标在部署层面存在局限。因此, 本文进一步采用 Accuracy 进行评估, 基于各测试集自身 EER 对应阈值, 衡量模型在多源失真场景下的分类表现, 更真实地反映其实用性与稳定性。

分析表 4 可以得到, 当采用各测试集自身计算得到的 EER 阈值作为固定判别标准时, 模型在链路传输前的 RealLink-Clean 数据集上整体表现良好, 检测准确率普遍超过 86%。其中, 以 ASVspoof 2021 DF 和 In-the-Wild 设定的阈值为例, 准确率分别达到 87.3%和 88.3%, 说明模型在理想条件下具备较强的分类能力。

表 4 不同设定阈值下 XLSR-AASIST 模型在真实链路数据集上的检测准确率

Table 4 Detection Performance of the XLSR-AASIST Model on Different Test Sets

设定阈值的数据集	阈值	准确率	
		RealLink-Clean	RealLink-Distorted
ASVspoof 2019 LA (Dev)	0.453	86.3%	72.3%
ASVspoof 2019 LA (Eval)	0.419	85.7%	71.9%
ASVspoof 2021 DF (Eval)	0.483	87.3%	79.8%
In-the-Wild	0.521	88.3%	81.3%

然而, 在经过通信链路传输后的 RealLink- Distorted 数据集上, 模型准确率显著下降, 普遍处于 70%～80%。例如, 基于 ASVspoof 2019 LA Dev 集设定的阈值, 准确率从 86.3%降至 72.3%; 基于 Eval 集设定的阈值, 准确率为 71.9%。相较之下, ASVspoof 2021 DF 和 In-the-Wild 设定的阈值在失真条件下仍表现相对稳健, 准确率分别为 79.8%和 81.3%。

上述结果说明, 在缺乏标签信息的实际应用中固定阈值策略易受到链路失真影响, 导致检测性能不稳定。因此, 为提升模型在现实通信环境下的可用性与鲁棒性, 需进一步引入更贴近实际失真特征的训练样本, 并探索更加自适应的判别机制, 从而增强模型在复杂部署场景中的适应能力。

5.2 数据增强方法的鲁棒性提升效果评估

为系统评估常见数据增强策略在复杂链路失真条件下的鲁棒性提升效果, 本研究选取了三类具有代表性的增强方法进行对比分析。这些方法通常被用来模拟构造通信链路条件下的失真环境, 以在缺乏真实链路样本时近似还原传输干扰。三类方法分别为: RawBoost, 通过在波形级别注入非线性扰动与随机噪声, 非线性扰动部分通常通过非线性函数变换、动态范围压缩或幅值裁剪等方式实现, 从而模拟真实通信链路中因低比特率压缩、语音编解码器失真及电声设备(如话筒、扬声器)重放过程产生的失真伪影; SpecAugment(时间–频率遮挡), 其基本思想是对音频的时频谱表示(Mel-spectrogram 或声谱图)进行随机遮挡操作。时间遮挡通过随机选择若干时间帧并将其置零, 模拟网络抖动或数据包丢失造成的短时音频缺失; 频率遮挡则通过遮挡连续频带来近似模拟带宽受限或链路压缩过程中的频率信息损失; 以及基于 MUSAN 噪声库的加性噪声(Add Noise), 通过叠加环境噪声来模拟实际通信中的背景干扰。这些方法已在多项公开伪造检测任务中被广泛采用, 但其在真实链路失真环境下的有效性尚缺乏系统验证。在实验中, 我们分别测试上述三种增强策略在单独使用与组合使用两种配置下的检测性能。组合策略指对每条训练样本在增强时随机选择三种方法之一, 以增加扰动多样性并提升模型泛化能力。表 5 给出了各增强策略在多个测试集(包括公开数据集与真实链路失真集)上的 EER。

如表 5 所示, 常见数据增强策略在标准公开测试集上整体表现出较好的鲁棒性提升效果。在 ASVspoof 2021 DF 和 2019 LA Eval 集上, RawBoost 和组合增强均显著降低了 EER(例如在 2021 DF 上由 6.72%降至 3.34%), 表明其具备较强的跨域迁移能力。在 Dev 集上虽然出现轻微性能波动, 但幅度较

表 5 各种增强策略在公共数据集和真实数据集上的 EER 表现

Table 5 EER Performance of Various Enhancement Strategies Across Public and Real-World Datasets

数据集	无增强	RawBoost	SpecAugment	Add Noise	组合增强
ASVspoof 2019 LA (Dev)	0.19 ± 0.01	0.50 ± 0.03	0.82 ± 0.06	0.85 ± 0.05	0.87 ± 0.06
ASVspoof 2019 LA (Eval)	0.58 ± 0.04	0.36 ± 0.02	0.53 ± 0.04	0.63 ± 0.04	0.30 ± 0.02
ASVspoof 2021 DF (Eval)	6.72 ± 0.47	3.36 ± 0.23	3.79 ± 0.27	3.47 ± 0.24	3.34 ± 0.23
In-the-Wild	12.3 ± 0.86	7.72 ± 0.54	9.62 ± 0.67	8.30 ± 0.58	7.55 ± 0.52
RealLink-Clean	27.9 ± 1.95	22.2 ± 1.55	23.85 ± 1.70	21.87 ± 1.53	20.8 ± 1.46
ReaLink-Distorted	33.3 ± 2.33	25.6 ± 1.79	26.3 ± 1.84	27.3 ± 1.91	24.0 ± 1.68

小。整体来看, 增强策略在标准测试集下能够有效缓解模型过拟合, 并提升其应对轻度失真的能力。

在 RealLink 数据集上, 增强策略同样带来了性能改善, 但整体提升幅度有限。对于 RealLink-Clean, 组合增强表现最佳(EER 从 27.9%降至 20.8%), Add Noise 和 RawBoost 也有明显下降。对于 RealLink-Distorted, RawBoost 在单一策略中最优(33.3%→25.6%), 组合增强进一步下降至 24.0%。这些结果说明, 增强方法能够一定程度模拟真实链路中的干扰, 但难以完全覆盖链路失真的复杂性。

从机制角度看, 链路失真通常伴随三类典型特征: (1) 高频能量损失与带宽裁剪——导致中高频段伪影和不自然过渡被掩盖; (2) 随机丢包与网络抖动——引起短时信息缺失和不规则能量波动; (3) 终端重放与非线性失真——改变语音包络和动态范围。相应地, 增强方法只能近似模拟部分因素: RawBoost 注入非线性噪声和动态范围压缩, 可一定程度对齐链路压缩与重放伪影; SpecAugment 的时间遮挡和频率遮挡对应网络抖动和带宽受限, 但其遮挡模式相对理想化, 无法捕捉真实链路的随机性和非平稳性; Add Noise 虽能增强对背景干扰的鲁棒性, 但并不能还原链路编码压缩和设备相关失真。RawBoost 注入非线性噪声和动态范围压缩, 可一定程度对齐链路压缩与重放伪影; SpecAugment 的时间遮挡和频率遮挡对应网络抖动和带宽受限, 但其遮挡模式相对理想化, 无法捕捉真实链路的随机性和非平稳性; Add Noise 虽能增强对背景干扰的鲁棒性, 但并不能还原链路编码压缩和设备相关失真。因此, 增强方法对真实链路场景的拟合存在天然不足。综上, 数据增强在模拟单一或局部失真时能够发挥积极作用, 尤其是组合增强在高复杂性场景中往往表现最优。然而, 整体 EER 仍然偏高, 说明这些方法在应对真实链路中多源、叠加、非平稳的复杂失真时提升有限。值得注意的是, 当模型仅依赖模拟构造的链路失真增强数据训练时, 在真实链路采集的伪造语音数据集上性能提升仍然有限; 相比之下, 引入真实链路采集样本能够更全面反映实际失真特征, 在提升模型鲁棒性方面具有不可替代的价值。

5.3 不同通信链路类型对伪造语音检测性能的影响

为探究电话通话与微信语音两类典型通信链路对伪造语音检测系统性能的影响, 本节在保持训练配置一致的前提下, 使用 RealLink-Clean(链路传输前)与 RealLink-Distorted(链路传输后)两个子集进行对比测试, 系统评估不同链路失真条件下模型的检测性能变化, 结果如表 6 所示。

表 6 不同链路类型对伪造语音检测模型性能的影响

Table 6 The Impact of Different Link Types on the Performance of Forged Speech Detection Models

被测数据集	场景	EER
RealLink -Clean	-	20.8%
RealLink -Call	电话通话	26.4%
RealLink -WeChat	微信电话	24.4%

注: 模型为 XLSR-AASIST, 训练数据为 ASVspoof 2019 LA。

从表 6 可见, 通信链路失真对伪造语音检测模型的性能造成了明显影响。链路传输前的原始样本检测错误率为 20.8%, 而经电话通话链路传输后的样本(RealLink-Call)则上升至 26.4%, 微信语音链路传输后的样本(RealLink-WeChat)也达到 24.4%。结果表明, 链路失真在不同程度上削弱了模型对伪造语音特征的辨识能力, 其中电话链路所带来的性能退化更为显著, 可能与其编码压缩强度更高、频带截断更严重等传输特性相关。

整体来看, 尽管模型在理想条件下具备一定的检测能力, 但在实际通信环境中面对多源链路失真时, 性能仍存在显著下降。现有多数检测方法仍主要依赖理想化或模拟条件下的训练与测试, 难以全面反映复杂链路失真带来的多样性与非线性影响。该实验结果进一步强调了将通信链路因素纳入伪造检测评估的重要性, 同时凸显出在真实诈骗场景中, 提升模型鲁棒性与泛化能力的现实迫切性。

5.4 检测模型面对不同伪造算法经电话与微信链路的性能退化对比

除了对整体性能进行比较, 本节将进一步分析不同伪造算法在电话和微信电话两种传播链路下的检测效果差异。考虑到不同类型伪造语音对链路失真可能具有差异化的敏感性, 该实验旨在揭示链路失真对各类合成方式(如 TTS、VC、商用系统、声码器等)的掩蔽或增强效应, 从而为后续构建具备更强泛化能力的检测模型提供指导。

实验结果如表 7 所示, 所有类型的伪造语音在经过通信链路传输后, 其检测难度均显著增加。与未经链路传输的 RealLink-Clean 数据集相比, 无论电话通话链路传输后的样本(RealLink-Call)还是微信电话传输后的样本(RealLink-WeChat), 所有伪造类型的 EER 都出现了明显上升。这证实了通信链路中的各种失真确实对伪造语音的声学特征产生了干扰, 从而降低了检测模型的识别能力。

表 7 不同链路类型对检测性能的影响

Table 7 Impact of Different Communication Links on Spoofing Detection Performance

被测数据集	场景	EER
RealLink-Clean(TTS)	-	17.8%
RealLink-Call (TTS)	电话通话	24.3%
RealLink- WeChat (TTS)	微信电话	19.2%
RealLink-Clean(VC)	-	22.3%
RealLink-Call (VC)	电话通话	26.4%
RealLink- WeChat (VC)	微信电话	25.8%
RealLink-Clean(Commercial)	-	32.2%
RealLink-Call (Commercial)	电话通话	37.8%
RealLink- WeChat(Commercial)	微信电话	39.5%

注: 表中标注如 “RealLink-Clean (TTS)” 表示未经链路传输的、由开源文本到语音(TTS)算法生成的原始伪造语音样本; 类似地, 括号内的 VC 和 Commercial 分别对应语音转换类算法与商用合成系统生成的伪造语音。

具体来看, TTS 伪造语音在电话链路失真下的性能退化最为显著, 其 EER 从链路传输前的 17.8%上升至电话通话后的 24.3%, 表现出对压缩与带宽剪裁的高度敏感。相比之下, 微信通话链路对 TTS 语音的“掩蔽”效应较弱, EER 仅上升至 19.2%。对于 VC 伪造语音, 链路传输前的 EER 为 22.3%, 电话通话后的 EER 为 26.4%, 微信通话后为 25.8%, 在两种链路下性能退化幅度较为均衡, 变化不如 TTS 显著。商用语音合成系统生成的伪造语音本身检测难度较高, 原始条件下 EER 即为 32.2%, 经过电话通话后上升至 37.8%, 微信通话后更是进一步上升至 39.5%。这说明在该类型伪造音频上, 微信链路带来的失真影响甚至超过电话链路, 显著加剧了检测难度。可见, 通信链路失真对伪造语音检测性能普遍存在负面影响, 且不同伪造方式对链路失真表现出差异化的敏感性。

5.5 训练数据构成对模型在链路失真检测中的性能影响

本节旨在探讨训练数据构成对检测模型在链路失真条件下的性能表现与迁移能力的影响。通过设置多组训练数据配置与测试子集, 对模型在面对电话与微信链路下的检测能力进行系统性分析, 评估其鲁棒性与域外适应性

5.5.1 链路失真性能退化是否源于“域外”影响

为验证链路失真条件下检测性能下降是否源于数据分布的“域外差异”, 本实验采用未经过链路传输的原始伪造语音作为训练集, 构建基线检测模型, 并在 RealLink-Call 与 RealLink-WeChat 上分别测试其检测性能。通过分析相较原始条件下的性能退化程度, 评估其对链路失真所导致分布变化的鲁棒性。

如表 8 所示, 在仅使用 RealLink-Clean 数据训练的条件下, 模型在链路失真场景中的检测性能出现显著退化。无增强情况下, 模型在 RealLink-Call 上的 EER 从 Clean 条件下的 0.77%急剧上升至 18.07%, 在 RealLink-WeChat 上也上升至 13.91%。引入数据增强后, 尽管在 Clean 条件下进一步降至 0.275%, 但在电话和微信链路下的 EER 仍分别为 9.7%和 3.7%, 显示链路失真仍带来较大挑战。上述结果表明, 通信链路引发了明显的数据分布偏移, 限制了模型在“域外”失真条件下的泛化能力, 而非“域外算法”的问题; 常规增强策略虽能在一定程度上缓解性能下降, 但仍难完全适应实际链路失真, 进一步强调了在训练阶段引入真实链路样本或更具针对性的失真建模机制的必要性。

表 8 仅基于 RealLink-Clean 训练的模型在不同链路条件下的检测性能

Table 8 Detection performance under different link conditions using model trained only on Real-Link-Clean

测试数据集	XLSR-AASIST w/o augmentation EER	XLSR-AASIST w augmentation EER
RealLink -Clean	0.77%	0.275%
RealLink -Call	18.07 %	9.7%
RealLink -WeChat	13.91%	3.7%

5.5.2 引入链路样本是否显著提升鲁棒性

本实验进一步探究训练集中引入链路失真样本

是否能够有效提升模型对真实通信环境中伪造语音的检测性能。对比实验设定为: 一组模型仅使用 RealLink-Clean 训练, 另一组则在训练中引入 RealLink-Call 与 WeChat 样本, 两组模型在同一测试集下进行性能评估, 实验结果如表 9 所示。

表 9　引入链路失真训练样本对检测性能的影响对比

Table 9　Impact of incorporating link-distorted training data on spoofing detection performance

训练数据集构成	测试集	EER
RealLink-Clean	RealLink -Call	19.16%
RealLink-Clean+Call+Wechat	RealLink -Call	6.48%
RealLink-Clean	RealLink -Wechat	15.76%
RealLink-Clean+Call+ Wechat	RealLink -Wechat	7.72%

注: 训练时只使用了各数据集中来自开源数据集的部分。

实验结果表明, 在训练集中引入真实链路失真样本能够显著提升模型在复杂通信环境下的检测鲁棒性。如表 9 所示, 当仅使用 RealLink-Clean 训练时, 模型在电话链路失真上的 EER 高达 19.16%, 在微信语音链路失真上也达到 15.76%。而当训练数据中进一步加入 RealLink-Call 与 RealLink-WeChat 的链路失真样本后, 模型在上述两个场景中的检测错误率分别降至 6.48%与 7.72%, 性能显著改善。

这一结果表明, 通信链路带来的失真属于模型难以泛化的“实际失真域”, 而通过在训练阶段显式引入这类样本, 模型能够更好地学习其特征模式, 从而有效缓解因链路失真导致的性能退化。该结论进一步强调了构建高质量真实链路伪造语音数据集的重要性, 并验证了真实场景下增强训练数据覆盖范围对提升伪造检测系统实用性的关键作用。

5.5.3　不同链路数据间的互补性与迁移能力分析

为了验证模型在不同通信链路间的跨链路泛化能力, 并探讨不同链路样本之间的互补性, 本实验设计如下三组训练配置: ① Clean + Call; ② Clean + WeChat; ③ Clean + Call + WeChat, 分别在两个链路测试集(Call / WeChat)上评估模型性能, 以观察单链路训练对另一链路测试场景的适应性。

如实验结果表 10 显示, 不同链路类型训练数据在伪造检测任务中具有明显的互补性和迁移能力。使用仅包含电话链路样本的训练集, 在电话测试集上的 EER 从 19.16%降至 8.61%, 同时在微信测试集上的 EER 也由 15.76%下降至 10.33%, 表现出良好的跨链路适应性。相反, 仅引入微信链路样本训练后, 模型在微信测试集上的 EER 下降至 10.58%, 同时在电话测试集上也降至 10.9%。说明不同链路数据均能为模型学习到一定的通用特征。进一步引入电话和微信两类链路样本联合训练, 模型在两个测试集上均达到最优性能, 电话链路 EER 为 6.48%, 微信链路 EER 为 7.72%。

表 10　不同链路训练数据对各类链路测试集的迁移能力分析

Table 10　Cross-link generalization performance with different link-specific training sets

训练集组成	测试集	EER
Clean	RealLink-Call	19.16%
Clean	RealLink-WeChat	15.76%
Clean + Call	RealLink-Call	8.61%
Clean + Call	RealLink-WeChat	10.33%
Clean + WeChat	RealLink-Call	10.9%
Clean + WeChat	RealLink-WeChat	10.58%
Clean + Call + WeChat	RealLink-Call	6.48%
Clean + Call + WeChat	RealLink-WeChat	7.72%

注: 训练时只使用了各数据集中来自开源数据集的部分。

综上所述, 单链路样本训练能够在一定程度上迁移到另一链路环境中, 体现出跨链路泛化能力, 而多链路数据的联合训练则不仅提升了目标链路的检测性能, 同时也显著增强了模型在未知链路条件下的鲁棒性。

5.5.4　复杂场景下的泛化测试

为了验证模型在更复杂通信链路条件下的鲁棒性, 我们在 ASVspoof 2021 LA 数据集上进行了跨链路泛化实验。该数据集通过真实电话系统(包括 VoIP 与 PSTN)传输采集, 涵盖跨国通信与多跳路由等条件, 能够在一定程度上模拟现实通信环境中的复杂失真。在实验中, 我们依旧构建了四种训练集配置, 并统一在 ASVspoof 2021 LA 测试集上评估模型性能, 结果如表 11 所示。

结果表明, 当训练仅依赖原始数据时, 模型在复杂链路失真场景下的 EER 为 5.57%; 引入电话链路样本后, EER 降至 3.95%; 引入微信链路样本时, EER 下降至 4.45%。进一步地, 当电话与微信链路样本联合训练时, 模型取得最佳结果, EER 降至 3.67%。

这一实验说明, 引入真实链路数据能够显著提升模型在跨链路复杂场景下的检测性能。不同链路样本在伪造检测中具有一定互补性, 单一链路样本即可带来跨链路适应能力的提升, 而多链路联合训练则在整体上表现更优, 对 VoIP、跨国通信和多跳路由等多样化失真条件展现出更好的适应性。

表 11 不同链路训练数据构成下的复杂场景跨链路泛化检测性能

Table 11 Cross-link generalization performance under complex scenarios with different training set compositions

训练集组成	EER(%)
Clean	5.57
Clean + Call	3.95
Clean + WeChat	4.45
Clean + Call + WeChat	3.67

注: 训练时只使用了各数据集中来自开源数据集的部分。

5.6 抗失真预处理对检测性能的影响

为了系统性地验证链路失真补模块在真实通信场景中的有效性, 我们设计了两组对照实验: 第一组采用“全量数据+常规直接训练”的基线方案, 即直接使用 RealLink-Clean、Call 与 WeChat 三个子集的全部样本进行训练; 第二组则在完全相同的训练配下, 将链路失真补偿模块嵌入到 SSL 表征提取器之后进行训练。

表 13 展示了在 RealLink 数据集上的对比实验结果。我们选取了两类典型的抗失真预处理方法: MMSE-STSA[37](传统经典的统计模型方法)与 Demucs[38](近年提出的基于深度卷积的端到端去噪模型)。这两类方法分别代表了传统信号处理与现代深度学习的去噪思路。

实验设置为: 首先对测试音频进行 MMSE-STSA 或 Demucs 去噪预处理, 然后在仅基于 RealLink-Clean 训练的检测模型上进行评估。结果如表 12 所示。从表中可以看出: 无论在 RealLink-Clean、RealLink-Call 还是 RealLink-WeChat 场景下, 两种去噪方法均导致性能下降, 只是程度有所差异。其中, MMSE-STSA 在三个测试集上的性能劣化最为严重, 尤其是在 RealLink-Call 和 RealLink-WeChat 上, EER 分别上升到 23.07%和 19.07%; Demucs 在 Clean 条件下的性能下降相对较轻, 但在 Call 和微信链路场景下依旧性能下降明显。

表 12 抗失真预处理方法对各类链路测试集检测性能影响

Table 12 Impact of anti-distortion preprocessing methods on the detection performance of various link test sets

测试数据集	原始数据	MMSE-STSA 去噪	Demucs 去噪
RealLink-Clean	0.275	5.35	2.89
RealLink-Call	9.7	23.07	19.07
RealLink-WeChat	3.7	19.2	11.1

6 抗失真检测方法

6.1 SSL 表征层面的链路失真补偿

实验结果表明, 通信链路失真(如电话压缩、微信传输)会显著削弱模型在真实场景中的判别能力。传统基于数据增强的训练方式虽能在一定程度上缓解性能退化, 但其随机性与不可控性难以覆盖真实链路中复杂的压缩与转码机制, 导致伪造痕迹仍然被掩盖。依托我们提出的 RealLink 数据集的独特特性——同时保留链路前后的配对样本, 我们能够显式建模失真对 SSL 表征的系统性扰动, 并据此设计一种特征域抗失真检测方法: 在表征层面对受损特征进行轻量修复, 再交由后端分类器进行判别。

设 $s_d = S(\tilde{x})$ 表示失真语音经过预训练编码器后的特征表示, $s_c = S(x)$ 为对应原始语音特征。我们引入一个失真恢复模块 $R_\theta(\cdot)$, 以残差方式修正受损表征:

$$\hat{s} = s_d + R_\theta(s_d) \tag{1}$$

其中, R_θ 由若干层卷积或轻量 Transformer 构成。最终判别由分类器完成:

$$\hat{y} = F(\hat{s}) \tag{2}$$

其中, F 代表分类器。

一致性约束: 最小化 $\hat{s}$ 与 s_c 的差异(如 L1/L2 距离), 引导恢复模块向的方向调整。

分类约束: 最小化交叉熵损失 $\mathcal{L}_{\text{cls}}$, 保证伪造痕迹能够被后端分类器有效利用。

综合损失为

$$\mathcal{L} = \lambda_{\text{rec}} \|\hat{s} - s_c\|_1 + \lambda_{\text{cls}} \mathcal{L}_{\text{cls}} \tag{3}$$

在实际推理时, 输入语音仅需经过一次 SSL 编码与恢复修正, 无须依赖原始语音,即

$$\tilde{x} \xrightarrow{S} s_d \xrightarrow{R} \hat{s} \xrightarrow{F} \hat{y}$$

该方法的可行性依赖于我们提出的数据集。该数据集不仅包含常见的压缩与转码, 还覆盖了电话与微信社交平台的真实链路采集。其核心特征是同时保留了原始音频与链路失真版本的配对样本, 从而在训练中能够提供一一对应监督信号。

在此基础上, 恢复模块能够学习到链路失真对 SSL 特征的系统性扰动模式, 并通过一致性约束进行校正。相比传统数据增强的“近似模拟”, 该方式直接利用了真实链路的数据统计特性, 因此更契合实际应用场景, 具备更强的泛化性。

6.2 实验验证

表 13 展示了在不同训练数据构成下的性能对比。当模型仅基于 RealLink-Clean 数据训练并直接在失真测试集上评估时, 其性能在真实链路环境下显著下降: 在电话和微信通话测试集上的 EER 分别为 19.16%和 15.76%, 表明模型难以直接应对链路失真带来的特征退化。训练中加入失真样本后, 模型能够显式学习到失真影响, 从而显著提升性能, 在电话和微信通话测试集上分别降低至 6.48%和 7.72%。

表 13 抗失真检测方法的有效性对比

Table 13 Effectiveness of the proposed module under full-data training

训练数据集构成	测试集	数据直接训练 EER (%)	链路失真补偿 EER (%)
RealLink-Clean	RealLink-Call	19.16	/
RealLink-Clean+call+WeChat	RealLink-Call	6.48	5.21
RealLink-Clean	RealLink-WeChat	15.76	/
RealLink-Clean+call+WeChat	Real-Link-WeChat	7.72	6.35

进一步地, 我们在相同训练配置下引入链路失真补偿模块, 在 SSL 表征层面对受损特征进行轻量修复。结果显示, 该模块能够进一步恢复被失真掩盖的伪造痕迹, 使得在 Call 测试集上的 EER 从 6.48% 下降至 5.21%, 在 WeChat 测试集上也从 7.72% 降至 6.35%。这一结果验证了所提出方法在数据驱动与表征修复结合下的互补作用, 即便在真实通道条件下也能带来显著性能提升, 体现了其在实际部署场景中的潜在应用价值。

7 总结

本文聚焦于真实诈骗通信环境下的伪造语音检测问题, 提出并实现了一套软硬件结合的链路复现系统, 构建了首个涵盖电话与社交平台链路失真的大规模伪造语音数据集 RealLink。系统实验评估了主流检测模型及增强策略在多源复杂失真下的性能表现, 发现通信链路中的压缩编码、网络抖动与设备增强会显著削弱伪造特征, 导致模型鲁棒性大幅下降, 且现有增强手段难以有效缓解。

进一步实验表明, 不同链路类型对检测模型具有差异化影响, 不同伪造方式的敏感性亦存在显著差异, 揭示了检测系统在真实环境下的多重脆弱性。同时, 通过引入链路失真样本进行训练, 模型鲁棒性显著提升, 验证了真实链路数据在训练中的重要价值。在此基础上, 本文进一步提出了一种基于 SSL 表征的抗失真检测方法, 通过轻量级恢复模块对受损特征进行修正, 能够一定程度上恢复被链路失真掩盖的伪造痕迹, 在真实链路场景中带来性能提升。

本文研究为构建面向实际应用的鲁棒伪造检测系统提供了数据基础与方法指导, 并强调了面向真实失真特征开展建模与训练的必要性。

参考文献

[1] Mills T, Bunnell H T, Patel R. Towards Personalized Speech Synthesis for Augmentative and Alternative Communication[J]. *Augmentative and Alternative Communication*, 2014, 30(3): 226-236.

[2] Costello J M. Message Banking, Voice Banking and Legacy Messages [EB/OL]. Boston Children's Hospital, 2016. Available: https://www.childrenshospital.org/.

[3] Stupp C. Fraudsters used AI to mimic CEO's voice in unusual crime [EB/OL]. Wall Street Journal, 2019. Available: https://www.wsj.com/.

[4] Casanova E, Weber J, Shulby C D, et al. YourTTS: Towards Zero-Shot Multi-Speaker TTS and Zero-Shot Voice Conversion for everyone[C]. *International Conference on Machine Learning*, 2021.

[5] Kinnunen T, Sahidullah M, Delgado H, et al. The ASVspoof 2017 Challenge: Assessing the Limits of Replay Spoofing Attack Detection[C]. *Interspeech 2017*, 2017: 2-6.

[6] Todisco M, Wang X, Vestman V, et al. ASVspoof 2019: Future Horizons in Spoofed and Fake Audio Detection[EB/OL]. 2019: arXiv: 1904.05441. https://arxiv.org/abs/1904.05441.

[7] Yamagishi J, Wang X, Todisco M, et al. ASVspoof 2021: Accelerating Progress in Spoofed and Deepfake Speech Detection[EB/OL]. 2021: arXiv: 2109.00537. https://arxiv.org/abs/2109.00537.

[8] Yi J Y, Fu R B, Tao J H, et al. ADD 2022: The First Audio Deep Synthesis Detection Challenge[C]. *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2022: 9216-9220.

[9] Yi J Y, Tao J H, Fu R B, et al. ADD 2023: The Second Audio Deepfake Detection Challenge[EB/OL]. 2023: arXiv: 2305.13774. https://arxiv.org/abs/2305.13774.

[10] Li M, Ahmadiadli Y, Zhang X P. Audio anti-spoofing detection: A survey [EB/OL]. 2024: arXiv preprint, arXiv:2404.13914.

[11] Tak H, Jung J W, Patino J, et al. End-to-End Spectro-Temporal Graph Attention Networks for Speaker Verification Anti-Spoofing and Speech Deepfake Detection[EB/OL]. 2021: arXiv: 2107.12710. https://arxiv.org/abs/2107.12710.

[12] Ba Z J, Liu Q Y, Liu Z G, et al. Exposing the Deception: Uncovering More Forgery Clues for Deepfake Detection[C]. *The Thirty-Eighth AAAI Conference on Artificial Intelligence and Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence and Fourteenth Symposium on Educational Advances in Artificial Intelligence*, 2024: 719-728.

[13] Shuai C, Zhong J M, Wu S, et al. Locate and Verify: A Two-Stream Network for Improved Deepfake Detection[C]. *The 31st ACM International Conference on Multimedia*, 2023: 7131-7142.

[14] Pan K, Yin Y F, Wei Y, et al. DFIL: Deepfake Incremental Learning by Exploiting Domain-Invariant Forgery Clues[C]. *The 31st ACM International Conference on Multimedia*, 2023: 8035-8046.

[15] Ba Z J, Wen Q, Cheng P, et al. Transferring Audio Deepfake Detection Capability across Languages[C]. *The ACM Web Conference 2023*, 2023: 2033-2044.

[16] Jung J W, Heo H S, Tak H, et al. AASIST: Audio Anti-Spoofing Using Integrated Spectro-Temporal Graph Attention Networks[C]. *ICASSP 2022 - 2022 IEEE International Conference on Acoustics*, *Speech and Signal Processing*, 2022: 6367-6371.

[17] Xiao Y, Das R K. XLSR-Mamba: A Dual-Column Bidirectional State Space Model for Spoofing Attack Detection[J]. *IEEE Signal Processing Letters*, 2025, 32: 1276-1280.

[18] Zhang Q S, Wen S B, Hu T. Audio Deepfake Detection with Self-Supervised XLS-R and SLS Classifier[C]. *The 32nd ACM International Conference on Multimedia*, 2024: 6765-6773.

[19] Baevski A, Zhou H, Mohamed A, et al. Wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations[EB/OL]. 2020: arXiv: 2006.11477. https://arxiv.org/abs/2006.11477.

[20] Park D S, Chan W, Zhang Y, et al. SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition[EB/OL]. 2019: arXiv: 1904.08779. https://arxiv.org/abs/1904.08779.

[21] Kim G, Han D K, Ko H. SpecMix: A Mixed Sample Data Augmentation Method for Training WithTime-Frequency Domain Features[EB/OL]. 2021: arXiv: 2108.03020. https://arxiv.org/abs/2108.03020.

[22] Frank J, Schönherr L. WaveFake: A Data Set to Facilitate Audio Deepfake Detection[EB/OL]. 2021: arXiv: 2111.02813. https:// arxiv.org/abs/2111.02813.

[23] Müller N M, Czempin P, Dieckmann F, et al. Does Audio Deepfake Detection Generalize? [EB/OL]. 2022: arXiv: 2203.16263. https://arxiv.org/abs/2203.16263.

[24] Zhang Z Y, Gu Y W, Yi X W, et al. FMFCC-A: A Challenging Mandarin Dataset forSynthetic Speech Detection[C]. *Digital Forensics and Watermarking*, 2022: 117-131.

[25] Yamagishi J, Veaux C, MacDonald K, et al. CSTR VCTK Corpus: English multi-speaker corpus for CSTR voice clonig toolkit (Version 0.92) [EB/OL]. University of Edinburgh, 2019. Available: https://datashare.ed.ac.uk/handle/10283/3443.

[26] Panayotov V, Chen G G, Povey D, et al. Librispeech: An ASR Corpus Based on Public Domain Audio Books[C]. *2015 IEEE International Conference on Acoustics*, *Speech and Signal Processing*, 2015: 5206-5210.

[27] Ito K. The LJ Speech Dataset [EB/OL]. 2017. Available: https://keithito.com/LJ-Speech-Dataset/.

[28] Shi Y, Bu H, Xu X, et al. AISHELL-3: A Multi-Speaker Mandarin TTS Corpus and the Baselines[EB/OL]. 2020: arXiv: 2010.11567. https://arxiv.org/abs/2010.11567.

[29] Lee S G, Ping W, Ginsburg B, et al. BigVGAN: A Universal Neural Vocoder with Large-Scale Training[EB/OL]. 2022: arXiv: 2206.04658. https://arxiv.org/abs/2206.04658.

[30] Perraudin N, Balazs P, Søndergaard P L. A Fast Griffin-Lim Algorithm[C]. *2013 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, 2014: 1-4.

[31] Kong Z F, Ping W, Huang J J, et al. DiffWave: A Versatile Diffusion Model for Audio Synthesis[EB/OL]. 2020: arXiv: 2009.09761. https://arxiv.org/abs/2009.09761.

[32] Wang D, Zhang X W. THCHS-30: A Free Chinese Speech Corpus[EB/OL]. 2015: arXiv: 1512.01882. https://arxiv.org/abs/1512.01882.

[33] Zen H G, Dang V, Clark R, et al. LibriTTS: A Corpus Derived from LibriSpeech for Text-to-Speech[EB/OL]. 2019: arXiv: 1904.02882. https://arxiv.org/abs/1904.02882.

[34] Garofolo J S, Lamel L F, Fisher W M, et al. DARPA TIMIT acoustic-phonetic continuous speech corpus CD-ROM. NIST speech disc 1-1.1[J]. *NASA STI/Recon technical report*, 1993, 93: 27403.

[35] Tak H, Kamble M, Patino J, et al. Rawboost: A Raw Data Boosting and Augmentation Method Applied to Automatic Speaker Verification Anti-Spoofing[C]. *ICASSP 2022 - 2022 IEEE International Conference on Acoustics*, *Speech and Signal Processing*, 2022: 6382-6386.

[36] Snyder D, Chen G G, Povey D. MUSAN: A Music, Speech, and Noise Corpus[EB/OL]. 2015: arXiv: 1510.08484. https://arxiv.org/abs/1510.08484.

[37] Ephraim Y, Malah D. Speech Enhancement Using a Minimum-Mean Square Error Short-Time Spectral Amplitude Estimator[J]. *IEEE Transactions on Acoustics*, *Speech*, *and Signal Processing*, 1984, 32(6): 1109-1121.

[38] Defossez A, Synnaeve G, Adi Y. Real Time Speech Enhancement in the Waveform Domain[EB/OL]. 2020: arXiv: 2006.12847. https://arxiv.org/abs/2006.12847.

[39] Aliyun Neural Text-to-Speech Service. Aliyun. https://ai.aliyun.com/nls/tts. June, 2025.

[40] Microsoft Azure Speech Services. Microsoft. https://azure. microsoft.com/zh-cn/products/ai-services/ai-speech/. June, 2025.

[41] Baidu Text-to-Speech Service. Baidu AI Open Platform. https://ai.baidu.com/tech/speech/tts. June, 2025.

[42] Volcengine Text-to-Speech Service. ByteDance. https://www. volcengine.com/product/tts. June, 2025.

[43] ElevenLabs Text-to-Speech Platform. ElevenLabs. https:// elevenlabs.io/. June, 2025.

[44] Voice.ai Voice Cloning and Speech Tools. Voice.ai. https://voice.ai/. June, 2025.

[45] iFLYTEK Text-to-Speech Service. iFLYTEK. https://global.xfyun.cn/products/text-to-speech. June, 2025.

[46] Peng K N, Ping W, Song Z, et al. Non-Autoregressive Neural Text-to-Speech[C]. *The 37th International Conference on Machine Learning*, 2020: 7586-7598.

[47] Liu S X, Su D, Yu D. DiffGAN-TTS: High-Fidelity and Efficient Text-to-Speech with Denoising Diffusion GANs[EB/OL]. 2022: arXiv: 2201.11972. https://arxiv.org/abs/2201.11972.

[48] Hayashi T, Yamamoto R, Yoshimura T, et al. ESPnet2-TTS: Extending the Edge of TTS Research[EB/OL]. 2021: arXiv: 2110.07840. https://arxiv.org/abs/2110.07840.

[49] Lancucki A. Fastpitch: Parallel Text-to-Speech with Pitch Prediction[C]. *ICASSP 2021 - 2021 IEEE International Conference on Acoustics*, *Speech and Signal Processing*, 2021: 6588-6592.

[50] Wang Y X, Stanton D, Zhang Y, et al. Style Tokens: Unsupervised Style Modeling, Control and Transfer in End-to-End Speech Synthesis[C]. *International Conference on Machine Learning*, 2018.

[51] Popov V, Vovk I, Gogoryan V, et al. Grad-TTS: A Diffusion Probabilistic Model for Text-to-Speech[C]. *International Conference on Machine Learning*, 2021.

[52] Chen M J, Tan X, Li B H, et al. AdaSpeech: Adaptive Text to Speech for Custom Voice[EB/OL]. 2021: arXiv: 2103.00993. https://arxiv.org/abs/2103.00993.

[53] Chevi R, Prasojo R E, Aji A F, et al. NIX-TTS: Lightweight and End-to-End Text-to-Speech *via* Module-Wise Distillation[C]. *2022 IEEE Spoken Language Technology Workshop*, 2023: 970-976.

[54] Ren Y, Liu J L, Zhao Z. PortaSpeech: Portable and High-Quality Generative Text-to-Speech[EB/OL]. 2021: arXiv: 2109.15166. https://arxiv.org/abs/2109.15166.

[55] Ao J Y, Wang R, Zhou L, et al. SpeechT5: Unified-Modal Encoder-Decoder Pre-Training for Spoken Language Processing[EB/OL]. 2021: arXiv: 2110.07205. https://arxiv.org/ abs/2110.07205.

[56] Vainer J, Dušek O. SpeedySpeech: Efficient Neural Speech Synthesis[EB/OL]. 2020: arXiv: 2008.03802. https://arxiv.org/abs/2008.03802.

[57] Nakano Y, Saeki T, Takamichi S, et al. VTTS: Visual-Text to Speech[C]. *2022 IEEE Spoken Language Technology Workshop*, 2023: 936-942.

[58] Zhang Z Q, Zhou L, Wang C Y, et al. Speak Foreign Languages with Your Own Voice: Cross-Lingual Neural Codec Language Modeling[EB/OL]. 2023: arXiv: 2303.03926. https://arxiv.org/abs/2303.03926.

[59] Kim J, Kong J, Son J. Conditional Variational Autoencoder with Adversarial Learning for End-to-End Text-to-Speech[EB/OL]. 2021: arXiv: 2106.06103. https://arxiv.org/abs/2106.06103.

[60] Kim J, Kim S, Kong J, et al. Glow-TTS: A Generative Flow for Text-to-Speech *via* Monotonic Alignment Search[EB/OL]. 2020: arXiv: 2005.11129. https://arxiv.org/abs/2005.11129.

[61] Choi Y, Choi M, Kim M, et al. StarGAN: Unified Generative Adversarial Networks for Multi-Domain Image-to-Image Translation[C]. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018: 8789-8797.

[62] Ren Y, Hu C X, Tan X, et al. FastSpeech 2: Fast and High-Quality End-to-End Text to Speech[EB/OL]. 2020: arXiv: 2006.04558. https://arxiv.org/abs/2006.04558.

[63] Luo R Q, Tan X, Wang R, et al. Lightspeech: Lightweight and Fast Text to Speech with Neural Architecture Search[C]. *ICASSP 2021 - 2021 IEEE International Conference on Acoustics*, *Speech and Signal Processing*, 2021: 5699-5703.

[64] Mehta S, Székely É, Beskow J, et al. Neural HMMS Are all You Need (for High-Quality Attention-Free TTS)[C]. *ICASSP 2022 - 2022 IEEE International Conference on Acoustics*, *Speech and Signal Processing*, 2022: 7457-7461.

[65] Huang R J, Zhao Z, Liu H D, et al. ProDiff: Progressive Fast Diffusion Model for High-Quality Text-to-Speech[C]. *The 30th ACM International Conference on Multimedia*, 2022: 2595-2605.

[66] Li N H, Liu S J, Liu Y Q, et al. Neural Speech Synthesis with Transformer Network[J]. *Proceedings of the AAAI Conference on Artificial Intelligence*, 2019, 33(1): 6706-6713.

[67] Lu H, Wu Z Y, Wu X X, et al. VAENAR-TTS: Variational Auto-Encoder Based Non-AutoRegressive Text-to-Speech Synthesis[EB/OL]. 2021: arXiv: 2107.03298. https://arxiv.org/abs/2107.03298.

[68] Lee Y, Yeon I, Nam J, et al. VoiceLDM: Text-to-Speech with Environmental Context[C]. *ICASSP 2024 - 2024 IEEE International Conference on Acoustics*, *Speech and Signal Processing*, 2024: 12566-12571.

[69] Ren Y, Ruan Y J, Tan X, et al. FastSpeech: Fast, Robust and Controllable Text to Speech[EB/OL]. 2019: arXiv: 1905.09263. https://arxiv.org/abs/1905.09263.

[70] Valle R, Shih K, Prenger R, et al. Flowtron: An Autoregressive Flow-Based Generative Network for Text-to-Speech Synthesis[EB/OL]. 2020: arXiv: 2005.05957. https://arxiv.org/abs/2005.05957.

[71] Lee Y, Shin J, Jung K. Bidirectional variational inference for non-autoregressive text-to-speech[C]. *International conference on learning representations,* 2020.

[72] Nguyen B, Cardinaux F. NVC-Net: End-to-End Adversarial Voice Conversion[C]. *ICASSP 2022 - 2022 IEEE International Conference on Acoustics*, *Speech and Signal Processing*, 2022: 7012-7016.

[73] Chou J C, Yeh C C, Lee H Y. One-Shot Voice Conversion by Separating Speaker and Content Representations with Instance Normalization[EB/OL]. 2019: arXiv: 1904.05742. https://arxiv.org/abs/1904.05742.

[74] Liu S X, Cao Y W, Wang D S, et al. Any-to-Many Voice Conversion with Location-Relative Sequence-to-Sequence Modeling[J]. *IEEE/ACM Transactions on Audio*, *Speech*, *and Language Processing*, 2021, 29: 1717-1728.

[75] Wang D S, Deng L Q, Yeung Y T, et al. VQMIVC: Vector Quantization and Mutual Information-Based Unsupervised Speech Representation Disentanglement for One-Shot Voice Conversion[EB/OL]. 2021: arXiv: 2106.10132. https://arxiv.org/abs/2106.10132.

[76] ZeroSpeech: Self-Supervised Zero-Shot Speech Synthesis. GitHub. https://github.com/bshall/ZeroSpeech. June, 2025.

[77] Li J Y, Tu W P, Xiao L. FreeVC: Towards High-Quality Text-Free One-Shot Voice Conversion[C]. *ICASSP 2023 - 2023 IEEE International Conference on Acoustics*, *Speech and Signal Processing*, 2023: 1-5.

[78] Baas M, van Niekerk B, Kamper H. Voice Conversion with Just Nearest Neighbors[EB/OL]. 2023: arXiv: 2305.18975. https:// arxiv.org/abs/2305.18975.

[79] van Niekerk B, Carbonneau M A, Kamper H. Rhythm Modeling for Voice Conversion[J]. *IEEE Signal Processing Letters*, 2023, 30: 1297-1301.

[80] Gu Y W, Zhang Z Y, Yi X W, et al. MediumVC: Any-to-any Voice Conversion Using Synthetic Specific-Speaker Speeches as Intermedium Features[EB/OL]. 2021: arXiv: 2110.02500. https://arxiv.

org/abs/2110.02500.

[81] Chen Y H, Wu D Y, Wu T H, et al. Again-VC: A One-Shot Voice Conversion Using Activation Guidance and Adaptive Instance Normalization[C]. *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2021: 5954-5958.

[82] Qian K Z, Zhang Y, Chang S Y, et al. AUTOVC: Zero-Shot Voice Style Transfer with Only Autoencoder Loss[EB/OL]. 2019: arXiv: 1905.05879. https://arxiv.org/abs/1905.05879.

[83] Serrà J, Pascual S, Segura C. Blow: A Single-Scale Hyperconditioned Flow for Non-Parallel Raw-Audio Voice Conversion[EB/OL]. 2019: arXiv: 1906.00794. https://arxiv.org/abs/1906.00794.

[84] Guo H J, Liu C R, Ishi C T, et al. Using Joint Training Speaker Encoder with Consistency Loss to Achieve Cross-Lingual Voice Conversion and Expressive Voice Conversion[C]. *2023 IEEE Automatic Speech Recognition and Understanding Workshop*, 2024: 1-8.

[85] Park S W, Kim D Y, Joe M C. Cotatron: Transcription-Guided Speech Encoder for Any-to-Many Voice Conversion without Parallel Data[C]. *Interspeech 2020*, 2020: 4696-4700.

[86] Kaneko T, Kameoka H, Tanaka K, et al. Cyclegan-VC2: Improved Cyclegan-Based Non-Parallel Voice Conversion[C]. *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2019: 6820-6824.

[87] Choi H Y, Lee S H, Lee S W. DDDM-VC: Decoupled Denoising Diffusion Models with Disentangled Representation and Prior Mixup for Verified Robust Voice Conversion[J]. *Proceedings of the AAAI Conference on Artificial Intelligence*, 2024, 38(16): 17862-17870.

[88] Popov V, Vovk I, Gogoryan V, et al. Diffusion-Based Voice Conversion with Fast Maximum Likelihood Sampling Scheme[EB/OL]. 2021: arXiv: 2109.13821. https://arxiv.org/abs/ 2109.13821.

[89] Li T L, Liu Y C, Hu C X, et al. CVC: Contrastive Learning for Non-Parallel Voice Conversion[EB/OL]. 2020: arXiv: 2011.00782. https://arxiv.org/abs/2011.00782.

徐哲 于 2019 年在宁波大学计算机科学与技术专业获得学士学位。现在南京理工大学网络空间安全专业攻读硕士学位。研究领域为语音安全与隐私保护。Email: 123127211612@njust.edu.cn

巴钟杰 CCF 专业会员, 于 2019 年在美国纽约州立大学布法罗分校计算机科学与工程获得博士学位。现任浙江大学计算机科学与技术学院网络空间安全与技术学院教授。研究领域为物联网的安全和隐私问题、多媒体内容的取证分析以及协作深度学习背景下的隐私增强技术。Email: zhongjieba@zju.edu.cn

任奎 CCF 会士, 于 2007 年在美国伍斯特理工学院大学电气与计算机工程专业获得博士学位。现任浙江大学求是讲席教授。研究领域为数据安全、物联网安全、人工智能安全和隐私。Email: kuiren@zju.edu.cn

程鹏 CCF 专业会员, 于 2019 年在英国兰卡斯特大学计算机科学获得博士学位, 现任浙江大学区块链与数据安全全国重点实验室研究员。研究领域为语音安全与隐私保护、物联网安全、AIGC 数据安全。Email: peng_cheng@zju.edu.cn

黄鹏 于 2020 年在浙江大学信息工程专业获得学士学位。现在浙江大学网络空间安全专业攻读博士学位。研究领域为物联网安全。研究兴趣包括设备隐私保护、声信号处理和人工智能安全等。Email: penghuang@zju.edu.cn

附表 1　RealLink 真实链路伪造语音数据集组成

Appendix 1　Composition of the RealLink Forged Speech Dataset with Real-World Link Distortion

编号	算法名称	类型	编号	算法名称	类型
1	Aliyun_TTS[39]	商用合成算法	33	Neural-HMM[64]	文本转语音
2	Azure_TTS[40]	商用合成算法	34	ProDiff[65]	文本转语音
3	Baidu_TTS[41]	商用合成算法	35	TransformerTTS[66]	文本转语音
4	HuoShan_TTS[42]	商用合成算法	36	VAENAR-TTS[67]	文本转语音
5	Elevenlabs_TTS[43]	商用合成算法	37	VoiceLDM[68]	文本转语音
6	voice_ai[44]	商用合成算法	38	WaveVAE[46]	文本转语音
7	iFLYTEK_TTS[45]	商用合成算法	39	Fastspeech[69]	文本转语音
8	AISHELL3[28]	公开数据集	40	Flowtron[70]	文本转语音
9	ASVspoof21DF[7]	公开数据集	41	BVAE-TTS[71]	文本转语音
10	FMFCC-A[24]	公开数据集	42	NVCNet[72]	语音克隆
11	InWild[22]	公开数据集	43	One-shot-VC[73]	语音克隆
12	LibriTTS[33]	公开数据集	44	PPG-VC[74]	语音克隆
13	THCHS-30[32]	公开数据集	45	VQMIVC[75]	语音克隆
14	TIMIT[33]	公开数据集	46	VQVAE[76]	语音克隆
15	ZH_WaveVAE[46]	文本转语音	47	FreeVC[77]	语音克隆
16	Diffgan-tts[47]	文本转语音	48	Knnvc[78]	语音克隆
17	EdgeTTS[48]	文本转语音	49	Speecht5_vc[55]	语音克隆
18	FastPitch[49]	文本转语音	50	Urhythmic[79]	语音克隆
19	GST_Tacotron[50]	文本转语音	51	MediumVC[80]	语音克隆
20	Grad-tts[51]	文本转语音	52	AGAIN-VC[81]	语音克隆
21	AdaSpeech[52]	文本转语音	53	AutoVC[82]	语音克隆
22	Nix-tts[53]	文本转语音	54	Blow[83]	语音克隆
23	Portaspeech[54]	文本转语音	55	ConsistencyVC[84]	语音克隆
24	Speecht5_tts[55]	文本转语音	56	Cotatron[85]	语音克隆
25	Speedyspeech[56]	文本转语音	57	CycleGAN-VC2[86]	语音克隆
26	vTTS[57]	文本转语音	58	DDDM_VC[87]	语音克隆
27	Valle-x[58]	文本转语音	59	DiffVC[88]	语音克隆
28	Vits[59]	文本转语音	60	CVC[89]	语音克隆
29	GlowTTS[60]	文本转语音	61	BigVGAN[27]	声码器
30	StarGAN[61]	文本转语音	62	Griffin_Lim[28]	声码器
31	FastSpeech2 [62]	文本转语音	63	Diffwave[29]	声码器
32	LightSpeech[63]	文本转语音			