Towards Reliable and Practical Phishing Detection

Hyowon Cho
KAIST
hyyoka@kaist.ac.kr

Minjoon Seo
KAIST
minjoon@kaist.ac.kr

Abstract

As the prevalence of phishing attacks continues to rise, there is an increasing demand for more robust detection technologies. With recent advances in AI, we discuss how to construct a reliable and practical phishing detection system using language models. For this system, we introduce the first large-scale Korean dataset for phishing detection, encompassing six types of phishing attacks. We consider multiple factors for building a real-time detection system for edge devices, such as model size, Speech-To-Text quality, split length, training technique and multi-task learning. We evaluate the model's ability twofold: in-domain, and unseen attack detection performance which is referred to as zero-day performance. Additionally, we demonstrate the importance of accurate comparison groups and evaluation datasets, showing that voice phishing detection performs reasonably well while smishing detection remains challenging. Both the dataset and the trained model will be available upon request.

1 Introduction

Phishing is an act of deceiving individuals into disclosing sensitive information or installing malicious software. With a huge amount of global financial damage, the demand for advancing phishing detection is larger than ever before. For instance, in 2022, the total loss amounts to 107 million dollars in South Korea (KISA, 2022) and a total loss of 52 million dollars was reported in the US (FBI, 2022).

Phishing poses significant detection challenges due to their subtle mimicry of legitimate communications and their ability to adapt rapidly, evading traditional defenses. Addressing these challenges requires detection systems that excel in two critical capabilities: (1) distinguishing nuanced differences between phishing and legitimate samples (*imitation detection*) and (2) generalizing to novel and unseen attack types (*zero-day detection* (Al-Rushdan et al., 2019)). These requirements highlight the need for robust datasets and development of methodologies that bridge the gap between academic research and practical deployment.

While previous research has advanced phishing detection, challenges remain for real-world application. Existing datasets lack size and diversity, with only 609 voice phishing samples available in Korean (Boussougou and Park, 2021) and 638 smishing instances in English (Mishra and Soni, 2022b). Moreover, current approaches often overlook practical issues, such as the need for real-time detection during calls, rather than post-call decisions, and other deployment challenges. Additionally, these methods fail to address zero-day attacks—new and unseen phishing techniques—which are critical for building robust detection systems.

In this paper, we present a comprehensive approach to building reliable phishing detection systems, underpinned by the introduction of the first large-scale dataset for smishing and vishing detection. This dataset comprises 94,602 phishing samples and 205,870 non-phishing samples, spanning six distinct attack types across multiple modalities. Each phishing type reflects the diverse strategies attackers employ, such as impersonating government agencies, financial institutions, parcel services, and even personal contacts. The dataset not only enables high-fidelity imitation detection but also includes carefully curated non-phishing samples to enhance robustness. These non-phishing examples are collected through crowdsourcing and are designed to mirror phishing characteristics, adhering to criteria such as thematic alignment, exclusion of impersonation targets, and the inclusion of phishing-related keywords to prevent overfitting.

To enable real-world deployment, we investigate practical considerations in system design. We focus on edge-compatible, small to medium-sized language models, such as DISTILKOBERT (Park, 2019) and MBERT-BASE (Pires et al., 2019), in conjunction with automatic speech recognition (ASR)

Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Industry Track), pages 210–225
April 30, 2025 ©2025 Association for Computational Linguistics

models like WAV2VEC2 (Baevski et al., 2020) and WHISPER (Radford et al., 2022). Advanced training techniques, including parameter-efficient fine-tuning (PEFT) and task-adaptive pretraining (TAPT), are applied to enhance performance while maintaining computational efficiency. We also address the challenge of handling real-time streaming data, a critical aspect of vishing detection, where timely detection can prevent significant harm.

We evaluate the models using a robust framework that prioritizes imitation and zero-day detection performance, as well as recall rates. The dataset and detection systems will be made available for further research, with the potential to generalize insights across languages and regions. This study not only advances the state of phishing detection but also contributes broadly to fraud prevention and cybersecurity.

2 Dataset Construction

We constructed a dataset comprising 94,602 phishing samples and 205,870 non-phishing samples. Each sample includes the following attributes: (1) text, (2) collection date, (3) phishing type, (4) label (phishing/non-phishing), and (5) modality (text or voice). Table 1 provides a detailed breakdown of the dataset.

2.1 Phishing Data Collection

Phishing Types. To capture diverse phishing tactics, we categorized phishing samples into five types:

- GOVERNMENT: Messages impersonating government entities such as police or prosecutors.
- FINANCE: Text messages and Voice calls impersonating financial institutions.
- PARCEL: Messages mimicking parcel delivery services.
- CREDIT: Messages related to payment fraud or fake purchase alerts.
- RELATIVE: Messages impersonating family members or acquaintances.

These categories span two modalities: text (smishing) and voice (vishing). FINANCE is further distinguished by modality (FINANCE-V for voice and FINANCE-M for text), ensuring nuanced analysis of phishing techniques. Detailed explanations for each phishing type are provided in Appendix B.

Label	Modality	Type	# of samples	# of tokens
Phishing	message	FINANCE-M	10,313	2,478,233
Phishing	message	PARCEL	42,381	1,681,603
Phishing	message	CREDIT	32,650	1,317,691
Phishing	message	RELATIVE	4,508	146,490
		Subtotal	91,629	6,268,112
Non-phishing	message	FINANCE	7,541	1,869,223
Non-phishing	message	PARCEL	7,597	779,646
Non-phishing	message	CREDIT	15,172	2,575,857
Non-phishing	message	RELATIVE	168,047	2,140,401
		Subtotal	198,357	7,365,127
Phishing	voice	GOVERNMENT	1,297	1,265,206
Phishing	voice	FINANCE-V	1,672	328,038
		Subtotal	2,973	1,593,244
Non-phishing	voice	FINANCE	2,170	537,267
Non-phishing	voice	ETC	5,343	272,877
		Subtotal	7,513	810,144
		Total	300,436	16,036,627

Table 1: Total count of data for each type. Non-phishing data and duplicates are removed from the collected dataset. We use MECAB to count the total number of tokens.

2.2 Phishing Data Collection

For the phishing class, we collaborated with the Korea Internet & Security Agency and the Korean National Police Agency to collect data from August 2022 to June 2023, at two-week intervals. The dataset includes 449,118 reported phishing phone calls and text messages from the public. After dropping duplicates, 94,602 samples were retained.

2.3 Filtering Process.

To ensure the quality of phishing samples, a rigorous filtering process was essential, as the data collected from public reports may include non-phishing events. The filtering began by removing duplicate entries to eliminate redundancy. Next, a keyword consistently appearing in phishing messages was identified, and data containing this keyword were selected for further review. The selected data were then manually reviewed to verify their relevance as phishing samples. This process was repeated iteratively, with new keywords being identified and applied until no phishing messages remained in the unfiltered dataset. While this method was labor-intensive and required significant human effort, it ensured a highly accurate and reliable dataset for phishing detection.

2.4 Non-Phishing Data Collection

Designing Robust Non-Phishing Samples The use of invalid non-phishing datasets can lead to misleading classification performance, where attacks often involve impersonation. Despite the importance of well-constructed non-phishing datasets, most existing approaches focus on phishing datasets and rely on publicly available general

Modality	Non-phishing Set	Eval Acc.
Vishing	AIHub	0.42
Vishing	Ours	85.21
Smishing	AIHub	56.79
Smishing	Ours	71.56

Table 2: Accuracy on phishing classification task using DISTILKOBERT. With a pre-defined evaluation set, the performance drops significantly when using the AIHub conversation dataset.

Type	Artifact Candidates
FINANCE	대출, 지원, 신청, 상환, 보증, 저금리
PARCEL	배송지, 택배, 발신, 고객, 문의, 오류
CREDIT	결제, 완료, 문의, 본인, 주문, 신고
RELATIVE	문자, 폰, 액정, 엄마, 수리, 아빠

Table 3: Potential artifacts for each type of smishing.

conversation datasets, such as those from AIHub (AIHub, 2021b, 2020), for non-phishing examples. However, as shown in Table 2, using only the AIHub dataset results in significantly lower accuracy on a pre-defined evaluation set (See Section 2.6), underscoring the need for a carefully curated non-phishing dataset.

To address this issue, we establish three key criteria for constructing a robust non-phishing dataset: (1) Impersonation Target – Exclude commonly impersonated entities in phishing, ensuring non-phishing samples remain relevant and realistic. (2) Theme and Domain – Align non-phishing samples with phishing themes, such as legitimate financial offers, for balanced representation. (3) Potential Artifacts – Include frequently used phishing-related words in non-phishing samples to prevent overfitting and enhance detection accuracy.

By applying these criteria, we ensure that the non-phishing dataset closely mirrors the phishing dataset in characteristics, making the classification task more realistic and challenging. For further details on the three criteria and construction process, see Appendix C.

Non-Phishing Sample Collection. We constructed the corpus using two platforms: AIHub, which provides AI infrastructure such as data and software APIs, and DeepNatural, a crowdsourcing platform. Through DeepNatural, crowdworkers contributed verified non-phishing messages they had received. This process resulted in 30,000 non-phishing samples. Remaining 175,870 samples are collected through AIHub.

2.5 De-identification

Phishing attacks commonly contain real victim information, making thorough personal information de-identification more critical than ever. To ensure this, we implement a two-step de-identification process. Detailed process of de-identification is in Appedix D and the output sample is at Table 4.

2.6 Challenging Dataset Construction

To rigorously assess the limits of our model's capabilities, we curate a challenging dataset that focuses on edge cases and complex scenarios, designed to test robustness and generalization under difficult conditions.

Smishing Cases. For smishing, we manually select highly challenging phishing and non-phishing pairs that even human evaluators find difficult to distinguish, obtaining total 119 smishing and 134 mirrored non-smishing samples. These cases reflect real-world ambiguities, ensuring the dataset captures the complexities of phishing detection. Detailed analysis from these selections are discussed in Section K.

Vishing Cases. For vishing, we prioritize testing the model's robustness to diverse recording environments. We source phishing calls from the Financial Supervisory Service, obtaining 182 FINANCE-V and 183 GOVERNMENT samples, all distinct from the training dataset. For non-phishing cases, due to the scarcity of government and police call recordings, we sample challenging examples from our collected non-vishing data, including a mix of FINANCE-V and ETC samples. This ensures the dataset not only tests generalization but also challenges the model with edge cases commonly encountered in real-world scenarios.

3 Task Setup

To evaluate the challenges of phishing detection comprehensively, we define two key performance aspects and corresponding evaluation metrics.

3.1 Performance Aspects

Imitation Detection Performance. This metric evaluates in-domain performance by measuring the system's ability to distinguish subtle differences between phishing and non-phishing samples. It tests how well the model handles nuanced distinctions within known data types.

	Text
ORIGINAL TEXT	[Web발신]이구형님의 상품권이 04/19 최경민(직장동료)님께 배송되었습니다. SMS/-
STEP 1	[Web 발신] 이 구 형님 의 상품권 이 04/19 #NAME (직장동료) 님 께 배송 되었습니다 . SMS /-
STEP 2	[Web 발신] #NAME님 의 상품권 이 04/19 #NAME (#MASK 동료) 님 께 배송 되었습니다 . SMS /-
TARGET TEXT	[Web 발신] #NAME님 의 상품권 이 04/19 #NAME (직장동료) 님 께 배송 되었습니다 . SMS /-

Table 4: A step-by-step example for de-identificaion. We mark tokens red where the model supposes to but fails to erase. We mark tokens blue where the model accidentally erases the information.

Modality	Train	Validation	Challenging
Vishing	7,777	1,945	730
Smishing	231,784	57,947	253
Multitask	239,561	59,892	983

Table 5: Data statistics. We pre-define the challenging dataset to ensure the robustness of our model. The leftover data were split into training and validation datasets in a ratio of 0.8 and 0.2.

Type	Phishing	Non-phishing	Total
POLICE	183	183	366
FINANCE-V	182	182	364
FINANCE-M	32	41	73
PARCEL	37	32	69
CREDIT	35	45	80
RELATIVE	15	16	31

Table 6: Total count of data in the challenging dataset.

Zero-Day Performance. This metric assesses out-of-domain performance, evaluating the system's ability to detect newly emerged zero-day attacks. It measures the model's capacity to generalize and identify the underlying characteristics of phishing fraud, which is critical given the evolving nature of phishing and its potential for significant financial harm.

3.2 Evaluation Metrics

We use two complementary metrics to evaluate model performance: **Accuracy** reflects overall model performance, balancing true positives and true negatives. **Recall** prioritizes capturing all phishing attacks. While accuracy provides a general performance overview, recall is especially important in phishing detection to minimize false negatives and prevent potential harm. However, excessive false positives can reduce system usability. By incorporating both metrics, we strike a balance between detection robustness and practical deployment. See Appendix H for further analysis.

4 Implementation Details

This section outlines the key considerations and methods for building a practical and robust real-time phishing detection system.

4.1 Backbone Models

We focus on small to medium-sized encoder-based language models suitable for edge device deployment due to their efficiency in classification tasks. Specifically, we use DISTILKOBERT and DISTILMBERT as small models, and KOBERT and MBERT-BASE as medium-sized models.

4.2 ASR Transcription

ASR Models. Transcription quality significantly affects phishing detection performance. We evaluate five ASR models: WAV2VEC2, in which we trained from scratch on Korean data, including ksponspeech (Bang et al., 2020) and low-quality telephone network voice data(AIHub, 2021a); and WHISPER, the OpenAI's pre-trained models with various size. We used SMALL, BASE, MEDIUM, and LARGE models. For deployment, we use WHISPER-SMALL, as it balance the size and the detection performance. See Appendix F to see the impact of ASR quality on detection performance.

Streaming Call Handling. In vishing, real-time detection is critical as transactions often occur mid-call. To handle streaming data, we split calls into 16-token segments and concatenate data from the call's start to each segment. Details are in Appendix E.

4.3 Training Methods

Standard Fine-Tuning. The entire pre-trained weights are fine-tuned using supervised training on the target task.

Parameter-Efficient Fine-Tuning (PEFT). PEFT optimizes a small number of parameters to reduce computational costs. Specifically we apply **LoRA**, which updates low-rank matrices for parameter adaptation (Hu et al., 2021) and **IA3**, which rescales inner activations with learned vectors (Liu et al., 2022).

TAPT + PEFT. Task-Adaptive Pre-Training (TAPT) enhances the adapters trained with PEFT by fine-tuning on phishing data. This approach preserves general knowledge for zero-day attacks while improving imitation detection.

5 Experiment Results

This section presents the detection performance for vishing and smishing across various experimental setups, focusing on identifying the most effective detection system. Notably, the evaluations in this section utilize our challenging dataset, specifically designed to assess the model's robustness under difficult conditions. For results on validation sets derived from proportional splits of the full dataset, refer to Appendix I.

5.1 Vishing Detection

Imitation Performance. In vishing, PEFT methods underperform compared to standard fine-tuning, with a 10% performance drop. TAPT mitigates this gap but does not fully close it. This suggests that ASR-generated text introduces stylistic challenges that require additional training. Detection performance by type can be found in the Appendix J.2.

Zero-Day Performance. Table 8 shows similar patterns to smishing. Notably, KOBERT trained on FINANCE-V achieves high accuracy on GOVERNMENT data (88.42%), but the reverse scenario performs poorly (54.72%). TAPT improves performance across both domains (+6.41%).

5.2 Smishing Detection

Imitation Performance. As shown in Table 7, imitation performance is notably low. Standard fine-tuning does not consistently improve with larger models, and while PEFT+TAPT slightly enhances performance, the improvements remain insufficient.

To further investigate this, we introduce two human performance baselines: **(1) Upperbound Models** – We fine-tune models on individual phishing types and evaluate them using corresponding evaluation datasets to provide upperbound results. For example, DISTILKOBERT achieves an average accuracy of 75.91 and recall of 0.95, while KOBERT reaches 78.78 and 0.92. **(2) General Human Performance** – Fifty participants evaluated 253 smishing instances. Their accuracy reached 52.00%, with a recall of 0.70, reflecting the inherent difficulty of this task. **(3) Expert Human Performance** – Five trained evaluators achieved an accuracy of 75.10% and a recall of 0.91, establishing a benchmark for well-informed evaluators.

Given these baselines, all models with standard fine-tuning outperform the general human baseline but fall short of expert-level and upperbound performance. However, the fact that the performance gap is not significantly large demonstrates the validity and effectiveness of our proposed methodology. We report detection performance by type in the Appendix J.1.

Zero-Day Performance. Table 9 evaluates zero-day phishing detection by excluding specific types from the training set. Using KOBERT + LoRA with TAPT, performance improves by up to 175% compared to standard fine-tuning, demonstrating the importance of preserving general knowledge for unseen attacks.

5.3 Multi-Task Detection

Multitasking improves detection performance for both smishing and vishing, as shown in Table 14.

Performance Trends. Standard fine-tuning shows type-specific trade-offs, improving vishing detection at the expense of smishing. PEFT reduces this gap, and PEFT+TAPT achieves balanced performance across all types. Using KOBERT + LoRA with multitasking leads to consistent improvements in both smishing and vishing detection. See Appendix J.3.

Practical Implications. Since text messages and calls differ in language modality and timeframes, multitasking enables a unified system suitable for edge deployment. PEFT+TAPT offers the most reliable results, balancing performance across all phishing types while maintaining computational efficiency.

6 Related Work

Phishing Detection. Early phishing detection research primarily focused on websites and email-based attacks, leveraging datasets of malicious URLs and phishing emails (Liu et al., 2010; phishtank, 2023; Radev, 2008). Advanced methods, including deep learning, have been widely applied to improve detection (Opara et al., 2020; Singh et al., 2020). With the rise of smishing and vishing, phishing detection has diversified. Smishing datasets were initially web-scraped (Jain et al., 2020; Mishra and Soni, 2019), with early models achieving high accuracy on small datasets, such as 638 smishing messages (Mishra and Soni, 2022a). However, systematic research in smishing remains

Method	Model	Smi. Total Acc.	Smi. Recall	Vi. Total Acc.	Vi. Recall	Multi Smi. Acc.	Multi Vi. Acc.
FINE-TUNING							
Standard	DISTILKOBERT	71.56	0.80	85.21	0.73	77.23 [+5.67]	84.68 [-0.53]
	KOBERT	68.75	0.80	94.23	0.96	53.25 [-15.5]	91.74 [-2.49]
	DISTILMBERT	53.75	0.45	90.23	0.83	51.29 [-2.46]	95.97 [+5.74]
	MBERT	58.43	0.75	**95.37**	0.91	47.81 [-10.62]	**96.27** [+0.9]
PARAMETER EFFICIENT FINE-TUNING							
Lora	KOBERT	71.07	0.92	74.95	0.70	76.13 [+0.06]	78.96 [+4.01]
	MBERT	67.14	0.82	80.16	0.82	74.06 [+6.92]	77.63 [-2.53]
IA3	KOBERT	58.57	0.61	65.53	0.95	73.84 [+15.27]	77.18 [+11.65]
	MBERT	63.53	0.69	54.26	0.98	72.55 [+9.02]	76.34 [+22.08]
+ TASK-ADAPTIVE FINE-TUNING							
Lora	KOBERT	77.48	0.78	83.08	0.80	**84.51** [+7.03]	86.91 [+3.83]
	MBERT	75.13	0.58	86.75	0.77	79.10 [+3.97]	83.88 [-2.87]
IA3	KOBERT	76.77	0.75	76.49	0.88	70.22 [-6.57]	71.68 [-4.81]
	MBERT	71.13	0.64	79.28	0.85	74.42 [+3.29]	80.49 [+1.21]

Table 7: Combined results for smishing (Smi.), vishing (Vi.), and multitask detection. **Bold** indicates the best score, underline highlights the top 3 scores among detection models, and relative changes in multitask performance are annotated with red for gains and blue for drops.

Type	SFT	PEFT	PEFT+TAPT
OOD_GOVERNMENT	88.42	86.26	76.45
OOD_FINANCE-V	54.72	55.33	79.40
Total Acc.	71.52	70.75	**77.93**

Table 8: Accuracy on unseen phishing attacks. We perform experiments with KOBERT and Lora adapters. We use WHISPER-SMALL ASR model and split length of 16. PEFT+TAPT shows approximately 180 percent of performance increase compared to standard finetuning method.

Type	SFT	PEFT	PEFT+TAPT
OOD_FINANCE-M	30.00	57.50	**72.50**
OOD_PARCEL	60.00	**67.50**	62.50
OOD_CREDIT	27.50	57.50	**62.50**
OOD_RELATIVE	30.00	57.50	**62.50**
Total Acc.	37.39	60.23	**65.38**

Table 9: Accuracy on unseen phishing attacks. Experiments done with KOBERT and Lora. PEFT+TAPT shows approximately 180 percent of performance increase compared to standard finetuning method.

limited, especially in languages like Korean. For vishing, available datasets are scarce, with notable contributions in Korea, including 609 voice phishing transcripts (Boussougou and Park, 2021). These datasets enabled high-performing models like KoBERT, achieving 99.6% accuracy (Boussougou and Park, 2022). Despite these efforts, the lack of large, diverse datasets limits progress in applying deep learning for scalable phishing detection.

Task-Adaptive Pre-Training. Task-Adaptive Pre-Training (TAPT) fine-tunes pre-trained language models on unlabeled, task-specific data to enhance performance (Gururangan et al., 2020). By adapting language representations to domain-specific contexts, TAPT improves model generalization for specialized tasks.

Parameter-Efficient Fine-Tuning. Parameter-Efficient Fine-Tuning (PEFT) reduces computational costs by optimizing only a subset of parameters in pre-trained models. Early approaches introduced adapters inserted between model layers (Houlsby et al., 2019), while recent methods include low-rank updates (LoRA) (Hu et al., 2021) and activation scaling (IA3) (Liu et al., 2022). These methods enable efficient adaptation to dynamic tasks without full model retraining.

7 Conclusion

In this paper, we conduct a comprehensive study to create a reliable and practical phishing detection model. We develop the first large-scale phishing dataset, which serves as the foundation for training a robust and practical detection system. We then conduct experiments considering various factors that can affect the performance. We define the challenges of phishing detection, focusing on imitation and zero-day attacks, and evaluate each model based on them. We believe that our phishing dataset and propose methodology will facilitate research in phishing detection and, more broadly, fraud detection.

Ethical Considerations. In this paper, we are disclosing sensitive data related to phishing crimes.

Our main concern is whether it is appropriate to make this data and the trained model publicly available. While sharing this data could certainly foster research in phishing detection, it also opens the possibility of malicious exploitation by criminals. For instance, these criminals might attempt adversarial attacks using the publicly accessible data and models. Acknowledging this potential risk, we have decide to share data and model upon request. After validation that the requester is not related to phishing crime, we will release the requested data.

References

AIhub. 2020. 민원(콜센터) 질의-응답 데이터. https://aihub.or.kr/aihubdata/data/view.do?currMenu=115&topMenu=100&aihubDataSe=realm&dataSetSn=98.

AIHub. 2020. 자유대화 음성(일반남여). https://aihub.or.kr/aihubdata/data/view.do?currMenu=115&topMenu=100&aihubDataSe=realm&dataSetSn=109.

AIHub. 2021a. 저음질 전화망 음성인식 데이터. https://aihub.or.kr/aihubdata/data/view.do?currMenu=116&topMenu=100&aihubDataSe=ty&dataSetSn=571.

AIHub. 2021b. 주제별 텍스트 일상 대화 데이터. https://www.aihub.or.kr/aihubdata/data/view.do?currMenu=115&topMenu=100&aihubDataSe=realm&dataSetSn=543.

Huthifh Al-Rushdan, Mohammad Shurman, Sharhabeel H Alnabelsi, and Qutaibah Althebyan. 2019. Zero-day attack detection and prevention in software-defined networks. In *2019 international arab conference on information technology (acit)*, pages 278–282. IEEE.

Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. *Advances in neural information processing systems*, 33:12449–12460.

Jeong-Uk Bang, Seung Yun, Seung-Hi Kim, Mu-Yeol Choi, Min-Kyu Lee, Yeo-Jeong Kim, Dong-Hyun Kim, Jun Park, Young-Jik Lee, and Sang-Hun Kim. 2020. Ksponspeech: Korean spontaneous speech corpus for automatic speech recognition. *Applied Sciences*, 10(19):6936.

Milandu Keith Moussavou Boussougou and Dong-Joo Park. 2021. A real-time efficient detection technique of voice phishing with ai. *한국정보과학회 학술발표논문집*, pages 768–770.

Milandu Keith Moussavou Boussougou and DongJoo Park. 2022. Exploiting korean language model to improve korean voice phishing detection. *정보처리학회논문지. 소프트웨어 및 데이터 공학*, 11(10):437–446.

FBI. 2022. 2022 federal bureau of investigation's internet crimes report. https://www.ic3.gov/Media/PDF/AnnualReport/2022State/StateReport.aspx.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360, Online. Association for Computational Linguistics.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for nlp. *Preprint*, arXiv:1902.00751.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *Preprint*, arXiv:2106.09685.

Ankit Kumar Jain, Sumit Kumar Yadav, and Neelam Choudhary. 2020. A novel approach to detect spam and smishing sms using machine learning techniques. *International Journal of E-Services and Mobile Applications (IJESMA)*, 12(1):21–38.

Jeong-Wook Kim, Gi-Wan Hong, and Hangbae Chang. 2021. Voice recognition and document classification-based data analysis for voice phishing detection. *HUMAN-CENTRIC COMPUTING AND INFORMATION SCIENCES*, 11.

KISA. 2022. 2022년 보이스피싱 피해현황 및 주요 특징. https://eiec.kdi.re.kr/policy/materialView.do?num=237719&pg=&pp=&device=pc&search_txt=&topic=&type=J&depth1=B0000&depth2=A#:~:text=%2D%20’22%EB%85%84%20%EB%B3%B4%EC%9D%B4%EC%8A%A4%ED%94%BC%EC%8B%B1(,%EC%9C%BC%EB%A1%9C%20%EB%91%94%ED%99%94%ED%95%98%EB%8A%94%20%EC%B6%94%EC%84%B8%EC%9E%84.

Gang Liu, Bite Qiu, and Liu Wenyin. 2010. Automatic detection of phishing target from phishing webpage. In *2010 20th International Conference on Pattern Recognition*, pages 4153–4156. IEEE.

Haokun Liu, Derek Tam, Mohammed Muqeeth, Jay Mohta, Tenghao Huang, Mohit Bansal, and Colin Raffel. 2022. Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning. *Preprint*, arXiv:2205.05638.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. *Preprint*, arXiv:1711.05101.

Sandhya Mishra and Devpriya Soni. 2019. Sms phishing and mitigation approaches. In *2019 twelfth international conference on contemporary computing (ic3)*, pages 1–5. IEEE.

Sandhya Mishra and Devpriya Soni. 2022a. Implementation of 'smishing detector': an efficient model for smishing detection using neural network. *SN Computer Science*, 3(3):189.

Sandhya Mishra and Devpriya Soni. 2022b. Sms phishing dataset for machine learning and pattern recognition. In *International Conference on Soft Computing and Pattern Recognition*, pages 597–604. Springer.

Chidimma Opara, Bo Wei, and Yingke Chen. 2020. Htmlphish: Enabling phishing web page detection by applying deep learning techniques on html analysis. In *2020 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE.

OpenAI. 2023. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Jangwon Park. 2019. Distilkobert: Distillation of kobert. *GitHub repository. Opgehaal van https://github.com/monologg/DistilKoBERTc*.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32.

phishtank. 2023. Join the fight against phishing. `https://www.phishtank.com`.

Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual bert? *arXiv preprint arXiv:1906.01502*.

D Radev. 2008. Clair collection of fraud email, acl data and code repository. *ADCR2008T001*.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2022. Robust speech recognition via large-scale weak supervision. *arXiv preprint arXiv:2212.04356*.

Shweta Singh, MP Singh, and Ramprakash Pandey. 2020. Phishing detection from urls using deep learning approach. In *2020 5th international conference on computing, communication and security (ICCCS)*, pages 1–4. IEEE.

Guido Van Rossum. 2020. *The Python Library Reference, release 3.8.2*. Python Software Foundation.

A Hardware and Software

All experiments are conducted using an NVIDIA A100 GPU and implemented in PyTorch (Paszke et al., 2019). Models are trained for 3 epochs with a learning rate of 1e-5, batch size 16, and AdamW optimizer (Loshchilov and Hutter, 2019). For TAPT, models are further pre-trained on phishing data for 1 epoch with a learning rate of 5e-5 and batch size 32. Results are averaged over three random seeds.

B Phishing Types

To create a robust detection system, it is crucial to examine a wide range of phishing types and understand the general properties of phishing. We consider six major phishing types. Each attack is classified based on the targets of impersonation, as described in Kim et al. (2021). Among six types, four are smishing and two are vishing.

Type 1: Government agency – Voice. In this scenario, criminals impersonate employees of government agencies such as the prosecution, police, or the Financial Supervisory Service. Criminals make victims believe they are involved in a crime and they can get support from the one they are talking with. Consequently, victims often disclose their personal information or meet the criminal in-person.

Type 2: Financial institutions – Voice. In this case, criminals deceive victims by promising low-interest loans backed by the government. Attacks of this type include tricking victims into taking out new loans to repay existing overdue loans, demanding payment for credit rating upgrades in exchange for low-interest loans, and installing malicious applications in the guise of non-face-to-face loan processes.

Type 3: Financial institutions – Message. Type 2 attacks predominantly occur through phone calls, but there is an emerging trend of conducting them via text messages. We call this type of attack Financial institution – Message.

Type 4: Parcel institution – Message. In this type of scam, the criminal sends a message claiming that there is an issue with the delivery address or customs clearance number for a package, resulting in a failed delivery. They provide a URL for the recipient to rectify the situation. However, clicking on the link leads to installing a malicious app or the unauthorized disclosure of personal information.

Type 5: Credit institution – Message. Victims receive text messages indicating that a payment has been made for products they did not purchase. They are then instructed to call a provided number if they did not make the purchase themselves. Upon calling, they engage in a conversation to resolve the issue, unwittingly disclosing their personal information.

Type 6: Relative – Message. In this case, criminals disguise themselves as family members or relatives and deceive victims into depositing money into their bank accounts by claiming urgent needs. This type of scam is particularly challenging to assess and recover from as the primary targets are usually elderly individuals who may not recognize the deception.

C Construction Process

C.1 Criteria for Non-Phishing Dataset

Impersonation Target. Phishing often involves mimicking specific organizations or individuals. To create realistic non-phishing examples, we analyze phishing data to identify commonly impersonated entities. For example, in PARCEL, criminals frequently impersonate parcel services such as Lotte, CJ, Logen, and the post office. Non-phishing samples are carefully curated to exclude these specific impersonation targets while ensuring relevance to the type.

Theme and Domain. When explicit targets are absent, we focus on the broader themes and domains of phishing attacks. For instance, in FINANCE-M, phishing messages commonly promote low-interest loans. To ensure balance, we include non-phishing messages related to legitimate financial products, such as lawful loan offers, aligning the theme with realistic scenarios.

Potential Artifacts. Certain words frequently appear in phishing data, disproportionately influencing classification results. These words, referred to as *potential artifacts*, may also occur in legitimate messages or calls. To prevent models from overfitting to these artifacts, we incorporate them into the non-phishing dataset. For example, words like "대출" (loan) or "택배" (parcel) appear in both phishing and non-phishing contexts. Table 3 lists the most frequent artifact candidates for each type. By addressing these artifacts, we reduce the risk of overfitting and enhance the robustness of the detection system.

We tailored the non-phishing dataset construction process to the characteristics of each phishing type:

For GOVERNMENT, genuine phone call recordings were unavailable due to their rarity. Instead, we utilized AIHub's customer service center dataset (민원(콜센터) 질의-응답 데이터) (AIhub, 2020), casual conversation datasets (자유대화 음성(일반남여)) (AIHub, 2020), and calls from institutions like news agencies and polling agencies. Potential artifacts were excluded to avoid errors introduced by the speech-to-text conversion process.

For FINANCE-M, PARCEL, and CREDIT, we collected non-phishing samples via crowdsourcing, guided by two criteria: (1) impersonation targets and (2) themes and domains. Workers were instructed to prioritize messages as follows:

1. Messages matching both (1) and (2) were categorized as the corresponding type.

2. Messages matching (2) but not (1) were also included as the corresponding type.

3. Messages matching (1) but not (2) or unrelated to both were labeled as "ETC."

The "ETC" category includes spam messages from various sources, such as fitness centers, educational institutions, shopping malls, and private groups.

For RELATIVE, we used 100,000 general conversation messages from AIHub (AIHub, 2021b), ensuring 20% contained potential artifacts, such as frequently occurring phishing-related words. The "ETC" category was also incorporated to enhance diversity.

This structured approach ensures a robust and realistic non-phishing dataset, improving the accuracy and reliability of phishing detection systems.

D De-identification

Phishing attacks commonly contain real victim information, making thorough personal information de-identification more critical than ever. To ensure this, we implement a two-step de-identification process.

Step 1: De-identification with GPT-4. In this phase, we employ GPT-4(OpenAI, 2023) for de-identification. We target names, phone numbers, tracking numbers, addresses, IDs, and passwords

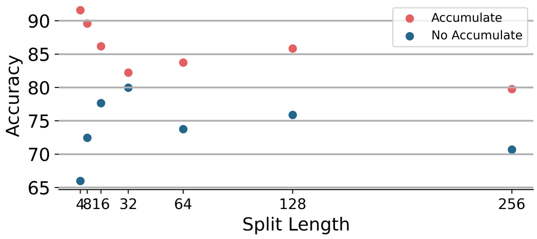

Figure 1: Results on voice phishing detection with DISTILKOBERT. Transcriptions are generated by WHISPER-SMALL. Accumulation of preceding segments greatly enhances performance, especially when the split length is small.

	DK	K	DM	M
WAV2VEC2	70.59	69.12	69.83	70.31
WHISPER-SMALL	85.21	94.23	90.23	95.37
WHISPER-BASE	90.61	92.10	93.43	92.14
WHISPER-MEDIUM	88.70	91.05	**96.54**	93.97
WHISPER-LARGE	**94.73**	**95.91**	93.27	**96.69**

Table 10: Results of vishing detection on evaluation set. We consider five ASR models to see the effect of transcription quality. We use the split length of 16 and stacked the preceding segments. **Bold** numbers indicate the best score and underline indicates second best score. D is for DISTIL, K is for KOBERT and M is for MBERT.

for de-identification. We provide few-shot examples to guide the model in replacing the specific information with corresponding tokens, such as transforming names into #NAME and numbers into #PHONE. This process is applied to 33,000 samples. However, we encounter some failed cases, as described in Table 4. Consequently, we opt to further remove personal information.

Step 2: De-identification with the Specialized Model. To ensure complete removal of personal information, even with some data damage, we train a Named Entity Recognition (NER) model using the original data and the de-identified samples generated in Step 1. Also, we conduct additional cleaning on each sample using the python re (Van Rossum, 2020), addressing simple cases like numbers. We employ KOBERT as the backbone model and fine-tune it for 20 epochs. To validate the efficacy of the de-identification process, we randomly select 100 examples for evaluation and manually review them.

E Handling Streaming Call Data.

Most of the financial transfer caused by vishing occur during the call. Therefore, the model should offer real-time detection to prevent the damage.

Handling streaming call data involves segmenting audio into time intervals for transcription input to a language model. Shorter intervals provide closer real-time feedback, but may lack meaningful semantics. To optimize pre-trained model capabilities, we set a minimum token count requirement, evaluating split lengths of 4, 8, 16, 32, 64, 128, and 256.

However, dividing a call into segments may not suffice. Vishing attacks have deceptive and easing parts, with the latter present in non-phishing samples. Labeling such segments as phishing can harm the model's performance. To counter this, we accumulate segments from the same call starting from the beginning to the current point. View Figure 1 for improvements in performance with shorter segments after applying the accumulation method.

F Effect of Transcription Quality.

Table 10 highlights the impact of ASR quality. Models using WHISPER significantly outperform WAV2VEC2, underlining the importance of accurate transcription. Among WHISPER variants, performance differences are minimal, with WHISPER-LARGE achieving the best results.

G Detection Timing of Voice Phishing.

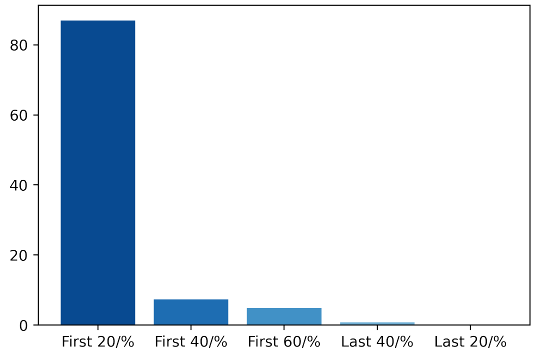

Figure 2: Detection timing of vishing. The system detect 86.95% of phishing calls at the early stage (first 20%).

Figure 2 depicts when the determination of the system is made when recall is 1. The system capture 86.95% of the phishing calls within the initial 20% of the call and 7.33% within the initial 40% of the call. This indicates that the evidence for classifying voice phishing is concentrated in the early stages of the call.

H Precision-Recall Trade-off Analysis

This section analyzes the trade-off between precision and recall in smishing and vishing detection, highlighting key performance patterns and implications for real-world deployment. Given the critical importance of recall in phishing detection to minimize false negatives, maintaining an acceptable precision rate remains a major challenge. Figures 3 and 4 visually represent these relationships.

Smishing. Figure 3 illustrates the precision-recall relationship for smishing detection. A linear correlation is evident between precision and recall, meaning that as recall increases, precision decreases proportionally. This pattern underscores a fundamental trade-off: achieving a recall of 1 (capturing all phishing messages) results in a precision of only 0.5, implying a 50% false positive rate. While this ensures that no phishing messages are missed, the high false positive rate could significantly reduce the system's usability. For practical deployment, finding an optimal threshold to balance precision and recall is crucial, especially in scenarios where excessive false positives could overwhelm users.

Vishing. In contrast, Figure 4 shows a more dynamic precision-recall trade-off for vishing detection. Unlike smishing, precision decreases more steeply as recall approaches 1. However, similar to smishing, precision stabilizes at 0.5 when recall reaches 1, indicating that half of the detected calls at full recall would be false positives. The sharper decline in precision for vishing is likely due to variations in audio transcription quality and linguistic inconsistencies introduced by ASR systems. This suggests that vishing detection systems require more sophisticated handling of ASR-generated text and potentially stricter thresholds to mitigate false positives while retaining high recall.

Practical Implications. Both smishing and vishing detection face challenges in achieving high recall without compromising precision. For smishing, the linear precision-recall relationship simplifies threshold adjustment, but achieving usability requires careful calibration. In vishing, the steep decline in precision with higher recall necessitates improvements in transcription quality and model robustness. These insights underscore the need for task-specific fine-tuning and adaptive thresholding to optimize phishing detection performance in real-world settings.

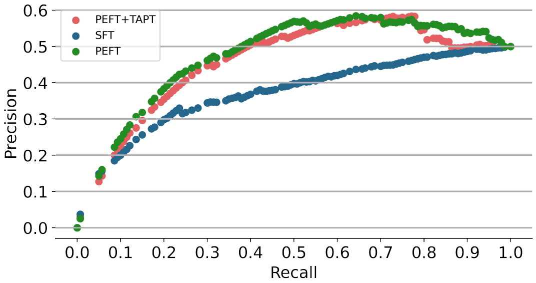

Figure 3: Precision-Recall graph for smishing detection by varying the inference threshold. A linear correlation is observed between precision and recall, with precision stabilizing at 0.5 when recall reaches 1.

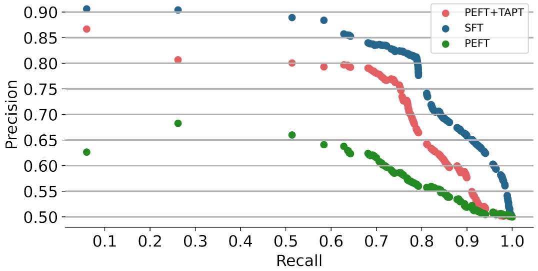

Figure 4: Precision-Recall graph for vishing detection by varying the inference threshold. Unlike smishing, precision decreases steeply as recall approaches 1, stabilizing at 0.5.

I Performance on Validation Dataset

Table 11 summarizes the validation results for smishing, vishing, and multitask detection across different fine-tuning methods and models. Overall, the performance metrics, including Total Accuracy and Recall, are consistently high across all setups, with many results nearing perfect recall values. For instance, standard fine-tuning achieves exceptional accuracy with models like KOBERT and MBERT, exceeding 95% in most cases.

However, this also underscores the need for evaluations on more challenging datasets. While validation results demonstrate high performance under controlled conditions, challenging datasets better reflect real-world complexities, such as nuanced distinctions and unseen attack types. Therefore, focusing on performance over these challenging scenarios is crucial for understanding the robustness and generalization capabilities of the model

J Performance on Challenging Dataset

J.1 Smishing

Table 12 presents the results of smishing detection, comparing various fine-tuning methods,

Method	Model	Smi. Total Acc.	Smi. Recall	Vi. Total Acc.	Vi. Recall	Multi Smi. Acc.	Multi Vi. Acc.
FINE-TUNING							
Standard	DISTILKOBERT	91.5	0.97	94.0	0.98	92.0	94.2
	KOBERT	95.5	1.00	98.5	0.98	94.8	97.8
	DISTILMBERT	91.0	0.96	94.5	0.97	91.8	95.0
	MBERT	93.5	0.98	98.8	0.98	93.2	97.5
PARAMETER EFFICIENT FINE-TUNING							
Lora	KOBERT	94.0	0.97	96.2	0.98	93.5	96.0
	MBERT	93.5	0.97	96.5	0.99	93.0	95.8
IA3	KOBERT	92.8	0.96	95.5	0.97	92.5	94.8
	MBERT	93.0	0.96	95.0	0.98	92.2	94.5
+ TASK-ADAPTIVE FINE-TUNING							
Lora	KOBERT	94.8	0.98	97.5	0.99	94.5	96.8
	MBERT	94.5	0.98	97.2	0.99	94.0	96.5
IA3	KOBERT	94.0	0.97	97.0	0.98	93.8	96.2
	MBERT	93.8	0.97	96.8	0.99	93.5	95.8

Table 11: Validation results for smishing (Smi.), vishing (Vi.), and multitask detection.

parameter-efficient approaches, and baselines. Standard fine-tuning shows that smaller models like DISTILKOBERT achieve competitive accuracy (71.56%) and recall (0.80), while multilingual models like MBERT generally underperform due to challenges in handling smishing-specific language nuances. Parameter-efficient fine-tuning (PEFT), particularly LoRA, improves performance significantly, with KOBERT+LoRA achieving 71.07% accuracy and a recall of 0.92. Combining PEFT with Task-Adaptive Pretraining (TAPT) further enhances results, with KOBERT+LoRA+TAPT achieving 77.48% accuracy, demonstrating the effectiveness of these advanced methods.

Baseline comparisons highlight that models surpass general human performance (52.00% accuracy, recall 0.70) and approach expert-level accuracy (75.10%) and recall (0.91). Upperbound models, fine-tuned on single phishing types, achieve the best results, with KOBERT reaching 78.78% accuracy and a recall of 0.92. These findings underscore the importance of task-specific pretraining and efficient fine-tuning in addressing smishing detection challenges while achieving performance comparable to expert human evaluators.

Moreover, the results in Table 12 provide a detailed breakdown of smishing detection performance across four phishing types: FINANCE, PARCEL, CREDIT, and RELATIVE. Each type demonstrates distinct challenges and opportunities for improvement, underscoring the importance of tailored approaches to detect different phishing strategies effectively.

FINANCE. Detection models generally underperform on FINANCE, with accuracy scores across methods remaining relatively low. For instance, the upperbound model fine-tuned specifically for this type achieves only 67.50% accuracy with DISTILKOBERT and 63.75% with KOBERT. This suggests that the overlap between financial terminology in both phishing and legitimate contexts makes it challenging to differentiate between the two.

PARCEL. The PARCEL type exhibits higher accuracy compared to other categories. For example, KOBERT+LoRA achieves 82.50% accuracy, and upperbound models reach up to 95.00%. This improved performance may stem from distinct linguistic patterns in phishing messages related to delivery or tracking, which are easier for models to identify.

CREDIT. The CREDIT category proves to be the most challenging, with models consistently achieving the lowest accuracy across all methods. For instance, DISTILMBERT and MBERT achieve only 27.50% and 30.00% accuracy, respectively, in standard fine-tuning. The difficulty likely arises from the close resemblance of phishing messages in this category to legitimate communications, leading to significant ambiguity.

RELATIVE. Performance on RELATIVE phishing is moderate, with accuracy ranging from 57.50% for KOBERT in standard fine-tuning to 100.00% for the expert human baseline. Notably, KOBERT+LoRA+TAPT achieves 87.50%, indicating that messages in this category often contain identifiable patterns, such as specific family-related terms, making them easier to detect with targeted training.

Method	Model	Finance	Parcel	Credit	Relative	Total Acc.	Recall
FINE-TUNING							
Standard	DISTILKOBERT	75.00	80.00	56.25	75.00	71.56	0.80
	KOBERT	72.50	78.75	66.25	57.50	68.75	0.80
	DISTILMBERT	63.75	61.25	27.50	62.50	53.75	0.45
	MBERT	66.25	75.00	30.00	62.50	58.43	0.75
PARAMETER EFFICIENT FINE-TUNING							
Lora	KOBERT	60.00	82.50	60.00	92.50	71.07	0.92
	MBERT	60.00	67.50	65.00	85.00	67.14	0.82
IA3	KOBERT	45.00	71.25	63.75	50.00	58.57	0.61
	MBERT	48.75	73.75	63.75	75.00	63.53	0.69
+ TASK-ADAPTIVE FINE-TUNING							
Lora	KOBERT	77.50	78.75	72.50	87.50	77.48	0.78
	MBERT	70.00	75.00	71.25	97.50	75.13	0.58
IA3	KOBERT	72.50	81.25	77.50	75.00	76.77	0.75
	MBERT	70.00	76.25	66.25	75.00	71.13	0.64
BASELINES							
UPPERBOUND	DISTILKOBERT	67.50	90.00	65.00	92.50	75.91	0.95
	KOBERT	63.75	95.00	71.25	97.50	**78.78**	0.92
	DISTILMBERT	66.25	77.50	73.75	95.00	75.21	0.88
	MBERT	72.50	52.50	76.25	97.50	71.29	0.80
GENERAL		47.89	56.43	51.23	54.46	52.00	0.70
EXPERT		73.97	72.46	68.75	100.00	75.10	0.91

Table 12: Results of smishing detection. We mark the best score **Bold**, and underline the top 3 best scores among our detection model. Detection module exceeds all human baselines but not upperbound models.

Summary. The findings reveal that while models perform well on types like PARCEL and RELATIVE, they struggle with more ambiguous categories like FINANCE and CREDIT. Parameter-efficient fine-tuning methods such as LoRA, especially when combined with task-adaptive pretraining (TAPT), show significant improvements across all categories, particularly for the more difficult types. These results emphasize the importance of diverse training data and targeted approaches to address the nuances of different smishing categories effectively.

J.2 Vishing

The table summarizes the results of vishing detection, comparing fine-tuning, parameter-efficient fine-tuning (PEFT), and task-adaptive pretraining (TAPT) across different models. In standard fine-tuning, MBERT achieves the highest total accuracy (95.37%) and a strong recall (0.91), showcasing its effectiveness in handling multilingual tasks, followed closely by KOBERT (94.23% accuracy, recall 0.96). Smaller models like DISTILKOBERT perform well overall (85.21% accuracy, recall 0.73), indicating the feasibility of deploying smaller models in resource-constrained environments.

For PEFT, KOBERT with LoRA achieves moderate results (74.95% accuracy, recall 0.70), while IA3 performs slightly worse, suggesting LoRA's better suitability for vishing tasks. Applying TAPT improves performance across models. For instance, KOBERT+LORA+TAPT increases accuracy to 83.08% with improved generalization, though it does not surpass the results of standard fine-tuning. Similarly, MBERT+LORA+TAPT achieves 86.75% accuracy, highlighting TAPT's ability to boost performance, albeit slightly below the best-performing standard fine-tuned models.

Moreover, in FINANCE-, accuracy is generally lower for this type across all methods, with a noticeable gap between fine-tuning and PEFT approaches. This reflects the complexity of financial phishing, where nuanced linguistic cues are critical for detection. MBERT consistently outperforms KOBERT and smaller models in both standard and PEFT settings, suggesting its strength in handling complex and diverse data.

For GOVERNMENT, all models and methods achieve higher accuracy, with MBERT and KOBERT nearing perfect performance in standard fine-tuning. The relatively structured and formal language used in government-related phishing may contribute to easier detection.

Method	Model	FINANCE-V	GOVERNMENT	Total Acc.	Recall
FINE-TUNING					
Standard	DISTILKOBERT	68.43	92.68	85.21	0.73
	KOBERT	91.24	95.56	94.23	0.96
	DISTILMBERT	75.00	96.98	90.23	0.83
	MBERT	86.45	99.34	**95.37**	0.91
PARAMETER EFFICIENT FINE-TUNING					
Lora	KOBERT	73.20	75.72	74.95	0.70
	MBERT	76.38	81.84	80.16	0.82
IA3	KOBERT	63.98	66.21	65.53	0.95
	MBERT	53.65	54.53	54.26	0.98
+ TASK-ADAPTIVE FINE-TUNING					
Lora	KOBERT	79.31	84.76	83.08	0.80
	MBERT	71.92	93.35	86.75	0.77
IA3	KOBERT	73.14	77.97	76.49	0.88
	MBERT	74.90	81.22	79.28	0.85

Table 13: Results of vishing detection. **Bold** indicates the best score and underline indicates the top 3 best scores among our detection model.

Method	Model	PARCEL	FINANCE-M	RELATIVE	CREDIT	Smi. Total	FINANCE-V	GOVERNMENT	Vi. Total
FINE-TUNING									
Standard	DISTILKOBERT	85.00	63.75	67.50	65.00	77.23 [+5.67]	70.40	92.10	84.68 [-0.53]
	KOBERT	68.75	55.00	65.00	33.75	53.25[-15.5]	53.25	96.80	91.74 [-2.49]
	DISTILMBERT	55.00	65.00	62.50	31.25	51.29 [-2.46]	93.61	98.32	95.97 [+5.74]
	MBERT	43.75	67.50	50.00	32.50	47.81 [-10.62]	93.34	99.20	**96.27** [+0.9]
PARAMETER EFFICIENT FINE-TUNING									
Lora	KOBERT	77.50	58.75	82.50	60.00	76.13 [+0.06]	67.16	81.78	78.96 [+4.01]
	MBERT	70.00	52.50	80.00	63.75	74.06 [+6.92]	64.19	81.19	77.63 [-2.53]
IA3	KOBERT	81.25	56.25	62.50	65.00	73.84 [+15.27]	66.60	80.51	77.18 [+11.65]
	MBERT	71.25	62.50	65.00	71.25	72.55 [+9.02]	67.95	80.12	76.34 [+22.08]
+ TASK-ADAPTIVE FINE-TUNING									
Lora	KOBERT	91.25	62.50	92.50	61.25	**84.51** [+7.03]	73.62	89.30	86.91 [+3.83]
	MBERT	77.50	71.25	82.50	48.75	79.10 [+3.97]	67.21	88.64	83.88 [-2.87]
IA3	KOBERT	73.75	43.75	52.50	63.75	70.22 [-6.57]	59.32	73.14	71.68 [-4.81]
	MBERT	85.00	67.50	95.00	58.75	74.42 [+3.29]	72.87	86.53	80.49 [+1.21]

Table 14: Results of the smishing and vishing when trained with both. We mark the best score **Bold**, and underline the top 3 best scores among our detection model. We also provide the difference between single-task and multi-task model, where red denotes a performance gain and blue denotes the performance drop.

J.3 Multitask

The experimental results highlight the effectiveness of multitasking and fine-tuning techniques in phishing detection, particularly for smishing and vishing across diverse attack types.

For smishing, multitask approaches such as KOBERT + LORA + TAPT achieved the highest overall performance with an accuracy of **84.51%**, significantly outperforming general human baselines (52.00% accuracy) and expert human evaluators (75.10% accuracy). Among smishing types, the PARCEL and RELATIVE categories showed the largest accuracy gains under multitasking setups, improving by +7.03 and +15.00, respectively. These results suggest that shared features across tasks enhance the model's ability to generalize effectively. However, credit- and finance-related smishing types exhibited relatively lower performance, indicating the potential need for additional domain-specific data or targeted fine-tuning strategies.

For vishing, multitasking also demonstrated substantial benefits. The best overall performance was achieved by MBERT + LORA + TAPT, with an accuracy of **86.91%**. Notably, the FINANCE-V type showed a significant improvement of +11.65 in accuracy under multitasking settings. Government-related vishing detection remained the most robust, with KOBERT + LORA + TAPT achieving a high accuracy of **89.30%**. These findings underscore the importance of transcription quality, as models utilizing advanced ASR systems like WHISPER consistently outperformed those relying on lower-quality transcriptions.

The analysis further highlights that multitasking is particularly advantageous for phishing types with shared characteristics, such as RELATIVE smishing and government-related vishing. Parameter-efficient fine-tuning (PEFT) and task-adaptive pre-training (TAPT) enhanced model generalization, particularly in zero-day attack scenarios, where unseen phishing types saw accuracy improvements of up to 175%. However, the relatively lower performance on credit smishing underscores challenges in data coverage and model adaptability.

K Human vs AI

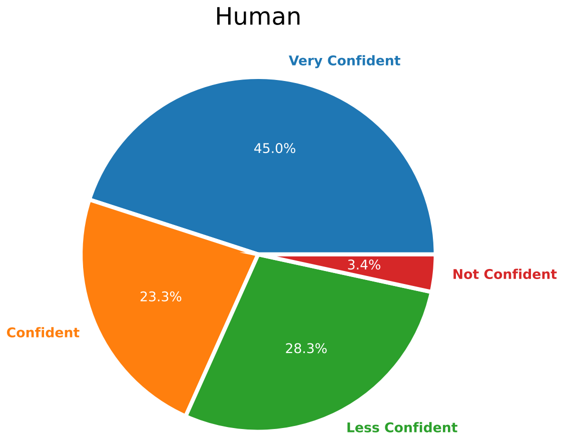

Figure 5: Human's confidence on their decision of distinguishing phishing and non-phishing data.

Q1: How confident each model and human is to their decision? Humans show high confidence on their decision. For human we ask people how confident that you won't be deceived by phishing attackers and provide four options: Very confident, Confident, Less confident, and Not confident. Figure 5 illustrates the result. 68.3% of humans, despite having limited knowledge about phishing, believe they would not fall for phishing attempts and make right distinction.

We also capture the saturation arises, as 87.6% of confidence rate of the model belongs between 0 to 0.1 or 0.9 to 1.

Q2: Can humans really distinguish phishing from spam message? General individuals, despite their high confidence, achieve only a 52% accuracy rate in distinguishing phishing from regular messages. Experts show substantial performance improvements. Notably, they reach a recall of 0.91, indicating the ability to avoid most phishing attacks. This leads to the conclusion that the real challenge lies in countering new phishing techniques. Refer to Table 1 for detailed scores.

Q3: Are some types more difficult than others? All phishing types pose equal challenges for the general humans. Experts find the CREDIT most difficult with an accuracy of 68.75%. The model also follows this trend, performing worst in the CREDIT with a 72.5% accuracy.

Q4: Are some types easier to train? The RELATIVE phishing type proves more trainable for both humans and models. Human performance improves across all phishing types after education, with RELATIVE exhibiting the most remarkable enhancement—an increase approximately 200%, while other types show improvements ranging between 15% to 20%. Similarly, the model's performance gains for each phishing type after training typically fall between 5% to 10%, but RELATIVE achieves a substantial gain of 25%.

This implies that while some phishing types remain challenging even after training, specific types become notably easier to distinguish once individuals are aware that a message is phishing. In these cases, the distinction between phishing and non-phishing messages becomes evident, potentially making individuals more susceptible due to a lack of exposure to this specific type of attack.

Q5: In what types does the best model outperform humans? Our best detection system outperforms humans except for the RELATIVE, with the most significant advantage in the FINANCE-M. This superior performance is attributed by the model's accessibility to a wealth of non-phishing financial message corpus, enabling it to detect phishing messages more effectively compared to most individuals who receive financial messages infrequently.

L Discussions.

There are doubts about whether smishing can be distinguished through text alone, prompting us to establish human baselines. Even experts achieve only 75% accuracy, indicating a challenging ceiling for smishing detection based solely on textual information. Concrete detection requires additional meta-information, such as sender details, numbers, and user history. Regarding vishing, we only use textual information in our work because, in most cases in our collected dataset, the pronunciation of the caller is nearly indistinguishable from regular callers. However, there is a possibility that additional acoustic features could improve performance.

The detection system, running every 16 tokens to be as close to real-time as possible, doesn't currently account for the computational cost of inference. Each decision involves the inference cost of both the ASR and detection models, resulting in high computational expenses per call. Therefore, there is a need to explore ways to lower the inference cost of both models.

Furthermore, while the methodology we propose is more robust to zero-day attacks, it still performs better at in-domain context. Therefore, there is a need for further investigation on how to continually train the system without the loss of previously learned knowledge.