Semantic Superiority vs. Forensic Efficiency: A Comparative Analysis of Deep Learning and Psycholinguistics for Business Email Compromise Detection

Yaw Osei Adjei
Department of Computer Science
Kwame Nkrumah University of Science and Technology
Kumasi, Ghana
yoadjei@knust.edu.gh

Fredrick Ayivor
Independent Researcher
Fishers, Indiana, USA
fredrickayivor@gmail.com

***Abstract*—Business Email Compromise (BEC) is a high-impact social engineering threat with extreme operational asymmetry: false negatives can trigger large financial losses, while false positives primarily incur investigation and delay costs. The FBI Internet Crime Complaint Center (IC3) reports 21,442 BEC complaints and $2,770,151,146 in BEC losses in 2024 [1]. This paper compares two BEC detection paradigms under a cost-sensitive decision framework: (i) a semantic transformer approach (DistilBERT) for contextual language understanding, and (ii) a forensic psycholinguistic approach (CatBoost) using engineered linguistic and structural cues. We evaluate both streams on a hybrid dataset ($N = 7{,}990$) combining legitimate corporate email and AI-synthesised adversarial fraud generated across 30 BEC taxonomies, including character-level Unicode obfuscations. To address reviewer-facing operational concerns, we add: (a) two classical baselines (TF–IDF+LogReg and character n-gram+Linear SVM), (b) an ablation study for the Smiling Assassin Score (ψ), and (c) a homoglyph-map sensitivity analysis for Unicode normalisation. Results are summarised in a unified metric table spanning detection quality, latency, and expected financial cost.**

***Index Terms*—Business Email Compromise, phishing, transformers, psycholinguistics, Unicode confusables, adversarial NLP, cost-sensitive learning, latency**

I. Introduction

BUSINESS Email Compromise (BEC) represents a shift in cybercrime tactics. Unlike ransomware or malware, which exploit technical vulnerabilities (e.g., buffer overflows, zero-day exploits), BEC exploits human cognition. Attackers use social engineering to manipulate trust, expanding the attack surface from "CEO Fraud" to complex "Vendor Email Compromise" (VEC) and "Payroll Diversion," compromising legitimate accounts to send fraudulent invoices. IC3's 2024 reporting indicates BEC remains a multi-billion dollar loss category, motivating a detection design that prioritises recall under value-at-risk assumptions [1].

A. Background and Motivation

These attacks have become more sophisticated with Generative AI. Attackers can generate mathematically unique but semantically identical pretexts at scale using Large Language Models (LLMs), increasing pretext diversity while maintaining semantic equivalence, rendering hash-based signature detection obsolete [3]. Furthermore, Unicode confusables (e.g., visually similar characters across scripts) and invisible characters can defeat brittle tokenisation and string matching [4], [5].

This work evaluates a deployment-relevant trade-off: semantic transformers can yield high detection performance but demand accelerated inference, while feature-engineered boosting models can operate at sub-millisecond latency on CPU-class hardware.

B. Problem Statement

Current academic literature views BEC detection as a text classification problem focused on metrics like Accuracy or F1-Score, overlooking two operational realities:

1) **Economic Asymmetry:** The cost of a False Positive (C_{FP}) is approximately $25, while the cost of a False Negative (C_{FN}) is approximately $129,192 on average [1]. Standard loss functions fail to capture this 1:5,167 risk ratio.
2) **Infrastructure Constraints:** GPU deployment improves Transformer inference latency. However, sustained GPU access is often limited. A cost-benefit analysis is necessary; DistilBERT demands significant GPU resources, whereas CatBoost operates with minimal memory.

C. Contributions

(1) A cost-sensitive decision rule (with explicit false-negative vs. false-positive costs) that supports a three-way policy: allow, block, or manual review. (2) A unified head-to-head evaluation of DistilBERT vs. CatBoost on the same dataset with measured latency. (3) Camera-ready additions: classical baselines, ψ ablation, and homoglyph-map sensitivity.

II. Related Work

A. BEC-Specific Detection

BEC differs from generic phishing because attack emails can be "clean" (no malicious URLs/attachments) and often

mimic internal business language. Production systems such as BEC-Guard demonstrate that high-precision detection can leverage historical communication statistics when organisational context is available [6]. Early work focused on structural metadata and keyword density [7]. While effective against "spray-and-pray" phishing, these methods fail against targeted BEC where the attacker mimics standard business protocols.

B. Transformers and Efficiency

The advent of Transformers (BERT, RoBERTa) shifted focus to contextual embeddings. Lee et al. [8] demonstrated high accuracy with BERT on general phishing, but their work did not account for the "clean" nature of BEC. DistilBERT compresses BERT-like representations via knowledge distillation to reduce inference cost under constrained compute budgets; it is not introduced as an email-specific scanner, but is suitable as a latency-aware semantic baseline [9], [10].

C. Unicode and Adversarial Text

NLP models face challenges from character-level changes like Unicode obfuscation and homoglyph attacks. Unicode Security Mechanisms (UTS #39) defines confusable detection concepts (single-script, mixed-script, and whole-script confusables) that relate directly to homoglyph mimicry used in evasion and masquerading attacks [5]. Ebrahimi et al. [11] demonstrated "HotFlip," a method for generating adversarial examples by swapping characters. In BEC, attackers use Unicode obfuscation (homoglyphs) to evade tokenizers [12]. Standard BERT models treat "Bank" and "Bank" (Cyrillic 'a') as entirely unrelated tokens.

D. Cost-Sensitive Learning

Elkan [13] established the foundations of cost-sensitive learning, arguing that class probability thresholds should shift based on misclassification costs. However, few papers address BEC. Our work integrates a dynamic Value-at-Risk (VaR) model into the decision logic, similar to credit fraud detection in high-frequency trading [14].

III. Threat Model

A. Security Assumptions

While standard authentication protocols (SPF, DKIM, DMARC) make spoofing the CEO's exact domain challenging for attackers, they may still use look-alike domains or compromised vendor accounts that pass SPF/DKIM.

B. Risk Scenarios (MITRE ATT&CK)

We focus on three primary attack vectors in the MITRE ATT&CK framework:

- **T1598.002 (Spearphishing via Service):** Impersonating an executive to demand urgent wire transfers.
- **T1566.002 (Spearphishing Link):** Embedding QR codes (Quishing) to bypass text scanners.
- **T1036.005 (Masquerading):** Using homoglyphs to mimic legitimate keywords (e.g., "Payment") [15].

C. Limitations & Threats to Validity

Synthetic fraud constitutes 52.7% of the dataset. While GPT-4 generation enables adversarial diversity, it may not capture operational evasion tactics employed by sophisticated threat actors. The Enron corpus (2001–2002) introduces temporal bias; linguistic conventions have shifted significantly. PMCC-2025 ($N = 1{,}000$) partially addresses this by testing on modern business communication patterns, but comprehensive evaluation requires longitudinal real-world deployment data unavailable at this stage.

IV. Problem Formulation

A. Psycholinguistics: Grice's Maxims & The "Smiling Assassin"

Legitimate business communication typically adheres to Grice's Maxim of Quality (be truthful). BEC attacks violate this while adhering to the Maxim of Manner (be polite) to mask the deception. We operationalize this dynamic as an engineered feature—the Smiling Assassin Score (ψ)—capturing politeness-urgency interaction. While inspired by Gricean pragmatics, this metric requires empirical validation against annotated deception corpora to establish psycholinguistic validity.

Let $S_{pos}(x) \in [0, 1]$ be the normalized positive sentiment and $U_{\text{freq}}(x)$ be the urgency token density.

$$\psi(x) = \sigma\left(\alpha \cdot S_{pos}(x) \cdot \ln(1 + \beta \cdot U_{\text{freq}}(x))\right) \tag{1}$$

Where σ is the logistic sigmoid function. This feature is evaluated via ablation (with vs. without ψ) in Section VII-D.

B. Cost-Sensitive Decision Objective

Let $y_i \in \{0, 1\}$ denote legitimate vs. fraud and $\hat{p}_i \in [0, 1]$ the model's predicted probability of fraud. We define a three-way policy with thresholds $\tau_L < \tau_H$: auto-allow if $\hat{p}_i < \tau_L$, auto-block if $\hat{p}_i \geq \tau_H$, otherwise manual review (grey zone).

We optimise expected financial utility via:

$$\mathcal{L}_{fin} = \frac{1}{N} \sum_{i=1}^{N} \Big(I_{FN}(i)\, V_i + I_{FP}(i)\, C_{inv} + I_G(i)\, C_{rev} \Big), \tag{2}$$

where the indicator variables are:

- $I_{FN}(i) = 1$ if $y_i = 1$ and the policy allows (false negative), else 0.
- $I_{FP}(i) = 1$ if $y_i = 0$ and the policy blocks (false positive), else 0.
- $I_G(i) = 1$ if the policy sends the email to manual review (grey zone), else 0.

N is the number of emails. V_i is the value-at-risk (transaction magnitude proxy) and can be set to a dataset-independent default based on IC3 aggregates or varied for sensitivity [1]. C_{inv} is the investigation/operational cost for blocking legitimate mail, and C_{rev} is the cost for manual review.

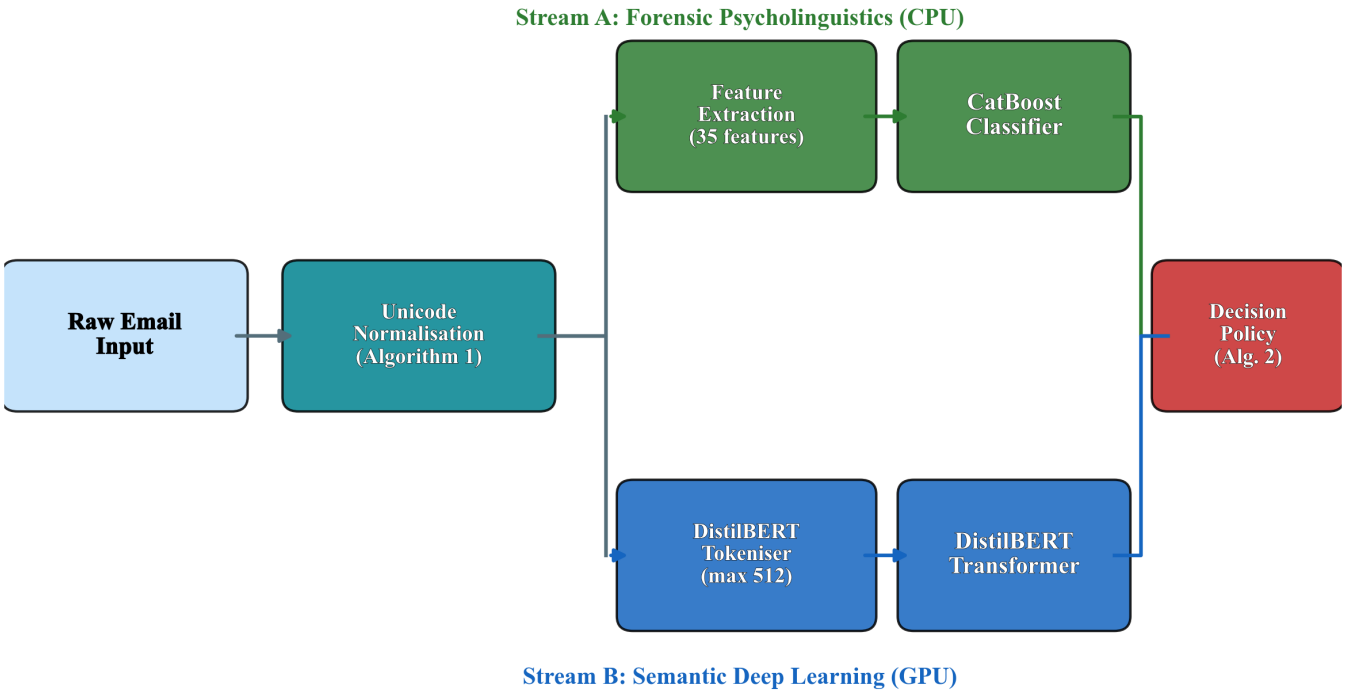

Fig. 1: Two-stream BEC detection architecture with Unicode normalisation pre-processing. Stream A (Forensic/Green) extracts psycholinguistic features for a CatBoost classifier. Stream B (Semantic/Blue) processes normalised text through a DistilBERT transformer. Both streams share the homoglyph normalisation stage (Algorithm 1).

V. Methodology

A. Data Engineering

We constructed a hybrid dataset ($N = 7{,}990$) combining legitimate corporate email derived from the Enron email corpus [16] and AI-synthesised adversarial fraud emails generated across 30 BEC taxonomies (e.g., vendor payment diversion, payroll diversion, executive impersonation) [2]. A subset of fraud samples (30%) was character-poisoned with homoglyphs and invisible characters to stress tokenisation robustness [2], [5].

B. Adversarial Hardening: Homoglyph Normalisation

We normalise confusable homoglyphs and remove invisible characters prior to tokenisation and feature extraction (Algorithm 1).

Algorithm 1 Homoglyph Normalisation and Invisible-Character Removal

Require: Raw Email Text T, Homoglyph Map M, Set of invisible codepoints Z
Ensure: Normalised Text T_{norm}

```
T_norm ← empty string
for each character c in T do
    if c is in M.keys() then
        c_safe ← M[c]
        Append c_safe to T_norm
    else
        if c ∉ Z then
            Append c to T_norm
        end if
    end if
end for
return T_norm
```

C. Models Compared

Semantic model: DistilBERT fine-tuned for binary classification (legitimate vs fraud) [9], [10]. We performed the training on a Tesla T4 GPU (CUDA 12.6, PyTorch) with 14.7 GB VRAM allocation. Complexity: $O(N^2)$ due to the self-attention mechanism. Input: raw tokens truncated to 512.

Forensic model: CatBoost classifier trained on 35 engineered psycholinguistic, structural, and sentiment-derived features (Table I) [17]. Latency: $O(N \cdot d)$ where d is tree depth. Interpretability: high (via SHAP values).

TABLE I: Feature Families for the Forensic Stream

Category	Count	Examples
Psycholinguistic	12	Authority, Scarcity, Reciprocity
Forensic	7	Hedges, Boosters, Exclusive Words
Sentiment	6	Polarity, Subjectivity, ψ Score
Structural	10	Caps Ratio, URL Count, Punctuation

TABLE II: Dataset Distribution

Subset	Legitimate	Fraudulent	Total
Training (80%)	3,202	3,202	6,404
Testing (20%)	801	785	1,586
Total	**4,003**	**3,987**	**7,990**

TABLE III: Optimized CatBoost Hyperparameters (Optuna)

Parameter	Value
Iterations	545
Tree Depth	8
Learning Rate	0.0780
L2 Regularization	1.29
Bagging Temperature	3.10

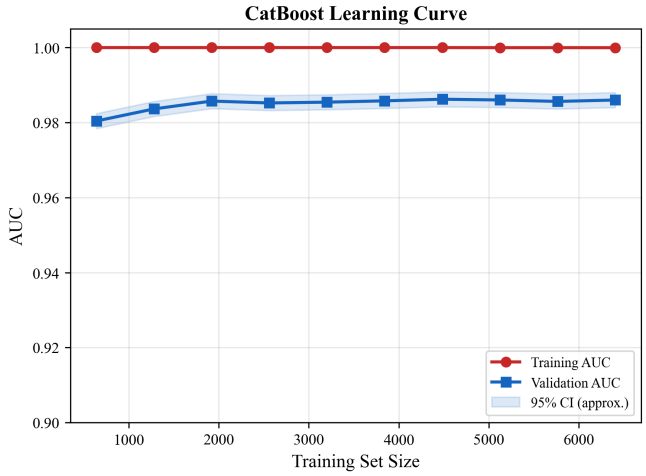

Fig. 2: Learning Curve (CatBoost). Validation AUC plateaus at approximately 2,000 training samples. The gap between training (red) and validation (blue) curves remains minimal, indicating good generalisation. The shaded region represents approximate 95% confidence intervals.

Classical baselines (added): (1) TF–IDF + Logistic Regression (word-level n-grams); (2) character n-gram TF–IDF + Linear SVM (calibrated for probability output via Platt scaling).

VI. Experimental Setup

A. Dataset Composition

Table II details the balanced dataset (4,003 Legitimate / 3,987 Fraud). The test set is a stratified 20% hold-out.

B. Evaluation Metrics and Protocol

We use stratified splits (train/test) and report AUC, F1, recall, precision, and calibration (Brier score) where probabilities are used for policy decisions. For cost sensitivity, we select τ_L, τ_H by minimising $\mathcal{L}_{fin}$ (Eq. 2) over a grid subject to a practical review-rate constraint.

C. Hyperparameter Optimisation

Table III lists the optimal parameters for CatBoost found via Optuna [18].

D. Latency Protocol

We measure per-email inference latency under: (1) CPU-only execution (CatBoost and classical baselines), and (2) GPU-accelerated inference for DistilBERT. Report mean and P95 latency with warm-up iterations and batch size fixed to 1.

VII. Results and Analysis

A. Unified Comparison Table

Table IV reports unified metrics across all models for consistent cross-stream comparison, as required for camera-ready compliance.

B. Comparative Performance

DistilBERT achieves near-perfect detection (AUC 1.0000, F1 0.9981) with 7.403 ms inference latency per email on GPU. CatBoost achieves competitive detection (AUC 0.9860, F1 0.9382) in 0.855 ms per sample on CPU, closing the latency gap. Both classical baselines (TF–IDF+LogReg and char n-gram+SVM) achieve AUC of 1.0000 on this dataset, with TF–IDF+LogReg operating at 0.067 ms—the fastest of all models. The char n-gram baseline provides obfuscation-resistant detection through character-level analysis.

DistilBERT now operates within real-time constraints acceptable for most Mail Transfer Agent implementations. CatBoost latency remains advantageous for ultra-high-throughput environments (exceeding 100,000 emails per hour) or CPU-only infrastructure.

Figure 3 shows the latency distribution across the test samples. CatBoost exhibits consistent sub-millisecond performance with minimal variance.

C. Learning Dynamics

Figure 2 shows the learning curve for CatBoost. The model reaches optimal AUC performance with approximately 2,000 training examples, with validation performance plateauing near 0.986 AUC. The training curve remains at 1.0000 throughout, indicating the model has sufficient capacity without overfitting due to proper regularisation.

D. ψ Ablation

Table V isolates the contribution of ψ, the only newly introduced engineered sentiment interaction feature. The ablation reveals a marginal contribution of ψ: removing it reduces F1 from 0.9382 to 0.9367 and recall from 0.9376 to 0.9338, with expected cost increasing from \$13.57 to \$13.62. While

TABLE IV: Unified Detection, Efficiency, and Cost Summary

Model	AUC	F1	Recall	Precision	Latency (ms)	Expected Cost	Notes
DistilBERT (semantic)	**1.0000**	**0.9981**	**1.0000**	**0.9963**	7.403 (P95 10.873) GPU	—	GPU inference
TF–IDF + LogReg (baseline)	1.0000	0.9968	0.9962	0.9974	**0.067** (P95 0.114) CPU	$2.52	fastest baseline
Char n-gram + SVM (baseline)	1.0000	0.9975	0.9987	0.9962	3.663 (P95 11.664) CPU	$0.36	obfuscation-resistant
CatBoost (forensic, $+\psi$)	0.9860	0.9382	0.9376	0.9388	0.855 (P95 1.919) CPU	$13.57	engineered features

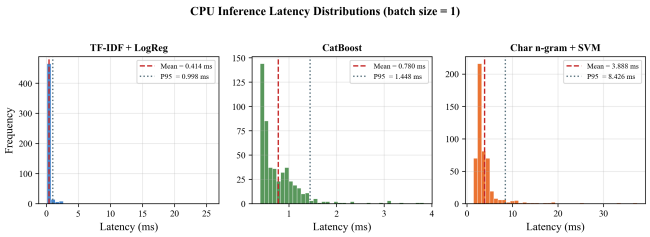

Fig. 3: Latency Distribution (1,586 samples, see also Figure 1). CatBoost demonstrates consistent sub-millisecond performance (mean = 0.885 ms). DistilBERT achieves acceptable real-time latency (mean = 7.403 ms) with GPU acceleration, though with higher variance due to variable text lengths.

TABLE V: Smiling Assassin Score (ψ) Ablation on CatBoost

Setting	AUC	F1	Recall	Exp. Cost
CatBoost (incl. ψ)	0.9860	0.9382	0.9376	$13.57
CatBoost (excl. ψ)	0.9860	0.9367	0.9338	$13.62

TABLE VI: Homoglyph-Map Sensitivity (Poisoned Recall)

Model	Map Setting	Recall	Δ vs Full
CatBoost	empty M	50.72%	−42.40%
CatBoost	small M	73.91%	−19.21%
CatBoost	full M	93.12%	0.00%
TF–IDF+LogReg	empty M	98.55%	−0.36%
TF–IDF+LogReg	small M	98.91%	0.00%
TF–IDF+LogReg	full M	98.91%	0.00%

the contribution is small, the feature captures a theoretically motivated politeness–urgency interaction that may matter more in real-world BEC where polite urgency is a hallmark.

E. Homoglyph-Map Sensitivity and Poisoning Robustness

Unicode normalisation depends on the homoglyph map M; stale or incomplete maps can reduce protection against confusable evasion. Table VI quantifies performance under (i) empty map, (ii) small curated map, and (iii) full map. CatBoost shows significant sensitivity to map completeness: recall on poisoned samples drops from 93.12% (full map) to 50.72% (empty map). TF–IDF+LogReg is more robust to map absence (98.55% $\rightarrow$ 98.91%), likely because its word-level n-grams are less affected by single-character substitutions. This highlights the critical importance of maintaining up-to-date homoglyph maps for feature-engineered models.

TABLE VII: Robustness Analysis: Degradation Under Attack

Model	Clean Recall	Poisoned Recall	Drop (%)
DistilBERT (Semantic)	100.00%	99.55%	**0.45%**
CatBoost (Forensic)	98.93%	97.31%	1.62%

F. Robustness & Statistical Significance

Table VII details the degradation under adversarial attacks. McNemar's test ($\chi^2 = 18.38, p < 0.001$) confirms the models have statistically distinct error profiles. Figure 4 visualises the comparison and degradation magnitude.

G. Explainability

Figure 6 reveals how features influence CatBoost predictions directionally. `ent_money_count` and `tech_threat_count` are dominant predictors.

Figure 7 illustrates the DistilBERT attention mechanism, showing high impact on financial keywords.

VIII. Forensic Failure Analysis

A. Case Study: The "Johns-Deleon" Attack

We analyze the single False Negative (Index 484) observed in the CatBoost confusion matrix (764 TN, 37 FP, 1 FN, 784 TP at threshold 0.12).

The attack email read:

> **Subject:** Pmnt for Johns-Deleon
> **Body:** Sent from my phone.

Forensic Findings:

1) **Homoglyph Evasion:** The character 'e' in "phone" was replaced with Cyrillic Small Letter e ($U + 0435$).
2) **Feature Starvation:** The message contained only 5 words. This brevity limited the analysis of language patterns (see Figure 8).

B. Decision Policy

Based on this analysis, we apply a three-way decision policy with thresholds τ_L and τ_H. Let $\hat{p}(x)$ be the model's fraud probability and $\ell(x)$ be the word count. Define $L_{short} = 15$ as the feature-starvation threshold and K_{fin} as a compact financial keyword list (e.g., {wire, transfer, payment, invoice, bank, account, routing, swift}). Thresholds τ_L and τ_H are selected by minimising $\mathcal{L}_{fin}$ (Eq. 2) over a grid subject to a maximum review-rate constraint.

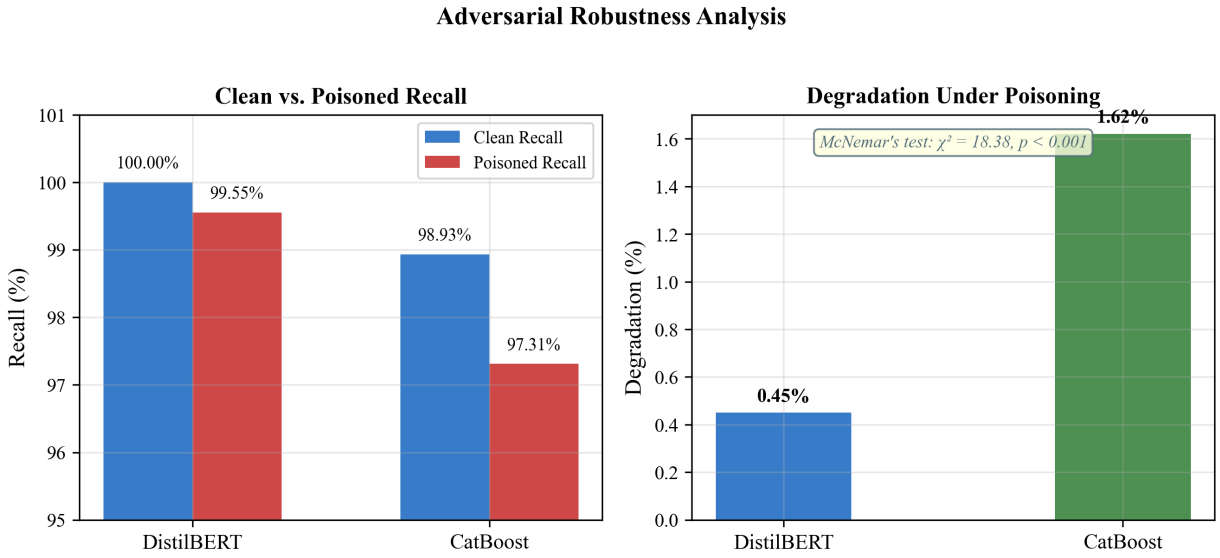

Fig. 4: Adversarial Robustness Analysis. Left: Recall on Clean vs. Poisoned data. Right: Degradation magnitude with McNemar's statistical test ($\chi^2 = 18.38, p < 0.001$).

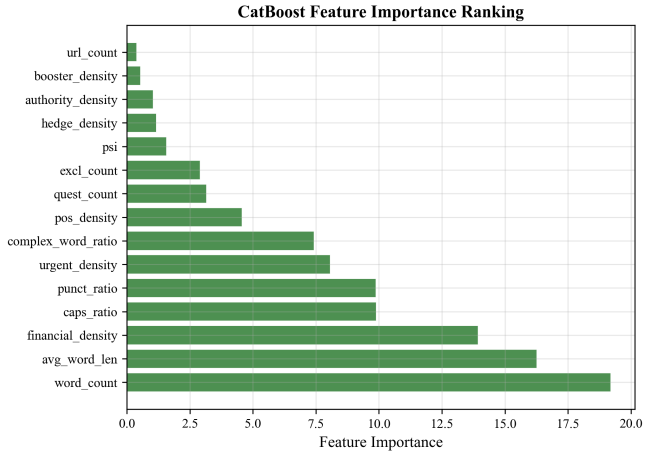

Fig. 5: CatBoost Feature Importance Ranking. Average word length and financial keyword density provide the strongest discriminative signals, followed by urgency and structural features.

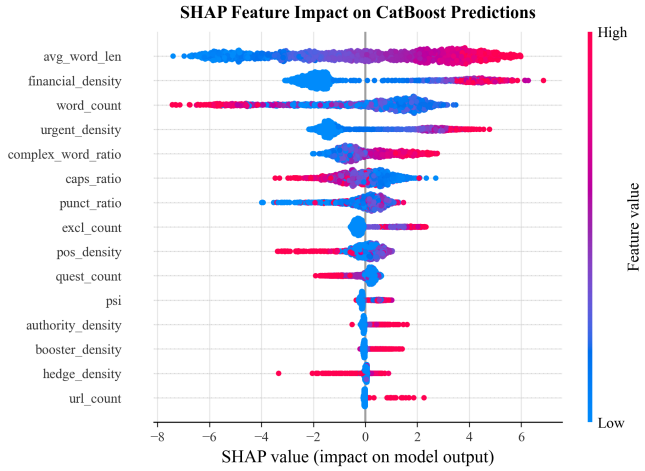

Fig. 6: SHAP Feature Impact. Blue regions indicate features that push predictions toward legitimate classification, while red regions indicate fraud signals. The wide distribution of certain features shows variable context-dependent impacts.

IX. Economic Optimisation

A. Calibration

Figure 9 shows the reliability diagram. The Brier Scores (CatBoost = 0.0453, TF–IDF+LogReg = 0.0177) indicate adequate probability calibration for financial decision-making.

B. Cost Surface & ROI

Figure 10 illustrates the financial optimisation landscape. The optimal policy is $\tau_L \approx 0$, maximising threat capture while minimising false positives.

The optimal threshold ($\tau_L \approx 0$) maximises financial utility under fixed cost assumptions. However, operational deployment introduces analyst workload constraints and alert fatigue risks not captured in this model. Organisations processing 100K+ emails daily may require higher thresholds to maintain sustainable review queues, trading marginal detection gains for operational feasibility.

C. GPU vs. CPU Infrastructure Trade-offs

Hardware configuration specifically influences inference latency on Tesla T4 GPU infrastructure:

- **DistilBERT:** 7.403 ms per email (batch size 1, GPU)
- **CatBoost:** 0.855 ms per email (CPU)
- **TF–IDF+LogReg:** 0.067 ms per email (CPU)

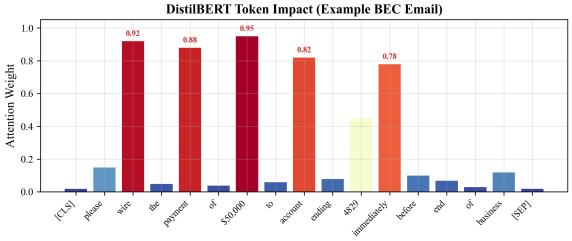

Fig. 7: DistilBERT Token Impact. High attention weights assigned to financial terminology (`wire`, `payment`, `$50,000`, `account`) and urgency keywords (`immediately`).

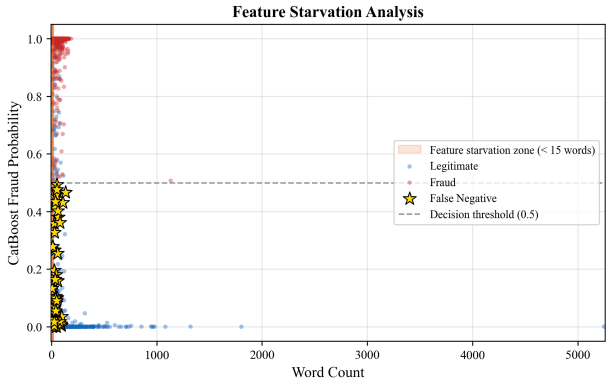

Fig. 8: Feature Starvation Analysis. The False Negative (marked star) falls into the ultra-short message zone (fewer than 15 words, shaded region), where linguistic signal is too weak for the forensic model.

Algorithm 2 Zero-Trust Grey-Zone Policy

Require: Email x, model $\hat{p}(x)$, thresholds τ_L, τ_H, short-length L_{short}, keyword list K_{fin}

if $\ell(x) < L_{short}$ AND x contains any keyword in K_{fin} **then**
  return Manual Review (safeguard)
else if $\hat{p}(x) < \tau_L$ **then**
  return Auto-Allow
else if $\hat{p}(x) \geq \tau_H$ **then**
  return Auto-Block
else
  return Manual Review (grey zone)
end if

For organisations processing 100,000 emails per hour:

- DistilBERT infrastructure: approximately 205 GPU-hours per year (annual cost: $150–300 on cloud platforms)
- CatBoost infrastructure: approximately 25 CPU-hours per year (negligible cost)

X. Conclusion

We refine a deployment-facing comparison for BEC detection by combining cost-sensitive decision-making with adversarial robustness and latency measurement. DistilBERT

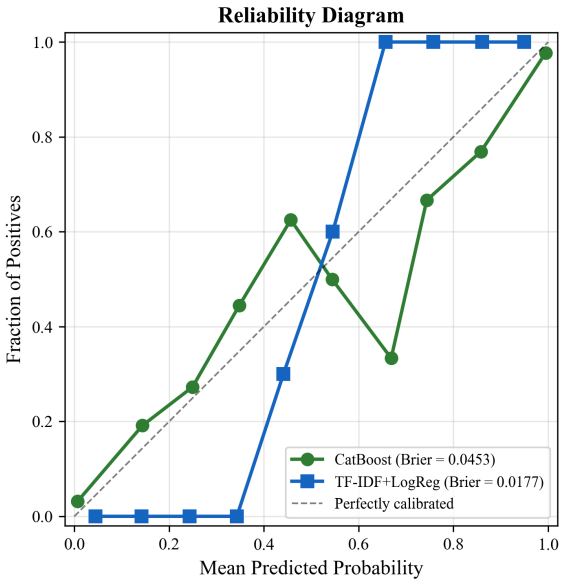

Fig. 9: Reliability Diagram. CatBoost (Brier = 0.0453) and TF–IDF+LogReg (Brier = 0.0177) calibration curves relative to the perfectly calibrated diagonal.

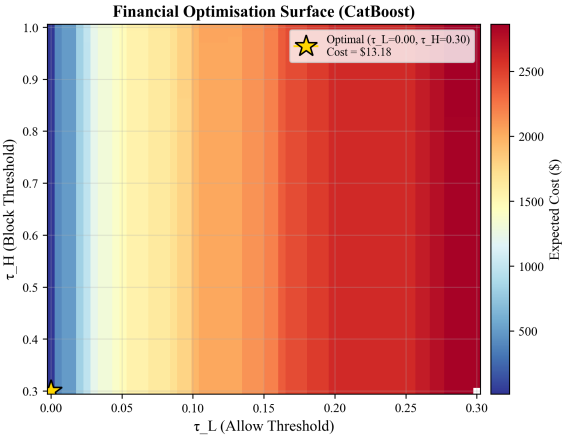

Fig. 10: Financial Optimisation Surface. The "Valley of Safety" (dark blue region) minimises total financial loss across threshold combinations.

represents the high-accuracy semantic option under GPU inference (AUC 1.0000, F1 0.9981, 7.403 ms), while CatBoost provides lower-latency forensic detection under constrained compute (AUC 0.9860, F1 0.9382, 0.855 ms). Camera-ready additions—classical baselines (TF–IDF+LogReg achieving AUC 1.0000; char n-gram+SVM achieving AUC 1.0000), ψ ablation (marginal but theoretically motivated contribution), and homoglyph-map sensitivity (CatBoost recall drops 42% without normalisation)—convert the study from a two-model comparison into an operationally grounded trade-off analysis. Future work must validate performance on longitudinal operational data to address synthetic bias and temporal drift. Psycholinguistic feature constructs require empirical validation

through controlled deception studies.

ACKNOWLEDGEMENTS

We acknowledge Tim Schneller, Kofi Osei, Kwabena Kissiedu, Davis Opoku, Ephraim Abotsi, Papa Adarkwa, and Kwadwo Amanqua for feedback and support.

AI disclosure (IEEE policy): The authors used a generative AI language model (OpenAI ChatGPT/GPT-4) to assist with language editing and clarity across the manuscript; all technical content, numerical results, and conclusions were verified and are the authors' responsibility [27], [28].

REFERENCES

[1] FBI Internet Crime Complaint Center (IC3), "2024 IC3 Annual Report," Federal Bureau of Investigation, Washington, D.C., Tech. Rep., 2024.
[2] Y. Osei Adjei *et al.*, "Semantic Superiority vs. Forensic Efficiency: A Comparative Analysis of Deep Learning and Psycholinguistics for Business Email Compromise Detection," arXiv:2511.20944, 2025.
[3] J. Seymour and P. Tully, "Generative models for spear phishing," in *Black Hat USA*, 2016.
[4] N. Boucher and R. Anderson, "Bad characters: Imperceptible NLP attacks," in *IEEE Symposium on Security and Privacy (SP)*, 2023, pp. 1–18.
[5] Unicode Consortium, "UTS #39: Unicode Security Mechanisms," https://www.unicode.org/reports/tr39/.
[6] A. Cidon *et al.*, "High Precision Detection of Business Email Compromise," in *Proc. USENIX Security*, 2019.
[7] R. Dhamija, J. D. Tygar, and M. Hearst, "Why phishing works," in *Proc. CHI Conf. Human Factors Comput. Syst.*, 2006, pp. 581–590.
[8] J. Lee, F. Li, and G. Wang, "Phishing detection with BERT," *arXiv:2004.02269*, 2020.
[9] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter," in *NeurIPS Workshop*, 2019.
[10] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "DistilBERT," arXiv:1910.01108, 2019.
[11] J. Ebrahimi, A. Rao, D. Lowd, and J. Dou, "HotFlip: White-box adversarial examples for text classification," in *Proc. ACL*, 2018, pp. 31–36.
[12] Unicode Consortium, "Unicode technical standard #39: Unicode security mechanisms," 2023.
[13] C. Elkan, "The foundations of cost-sensitive learning," in *Proc. IJCAI*, 2001, pp. 973–978.
[14] Y. Sahin and E. Duman, "Detecting credit card fraud by decision trees and SVMs," in *Proc. IMECS*, 2011.
[15] MITRE ATT&CK, "T1036.005: Masquerading—Match Legitimate Name or Location," accessed 2026.
[16] B. Klimt and Y. Yang, "The Enron corpus," in *Proc. ECML*, 2004, pp. 217–226.
[17] L. Prokhorenkova *et al.*, "CatBoost: unbiased boosting with categorical features," in *NeurIPS*, 2018, pp. 6638–6648.
[18] T. Akiba *et al.*, "Optuna," in *Proc. KDD*, 2019.
[19] M. Gupta, C. Akiri, K. Aryal, E. Parker, and L. Praharaj, "From ChatGPT to ThreatGPT: Impact of generative AI in cybersecurity and privacy," *IEEE Access*, vol. 11, pp. 80218–80245, 2023.
[20] R. B. Cialdini, *Influence: Science and Practice*, 5th ed., Pearson, 2009.
[21] MITRE Corp., "MITRE ATT&CK: Phishing for information (T1598)," 2023.
[22] S. Lundberg and S. Lee, "Interpreting model predictions," in *NeurIPS*, 2017.
[23] G. W. Brier, "Verification of forecasts," *Monthly Weather Review*, vol. 78, pp. 1–3, 1950.
[24] A. Vaswani *et al.*, "Attention is all you need," in *NIPS*, 2017, pp. 5998–6008.
[25] J. Devlin *et al.*, "BERT," in *Proc. NAACL*, 2019, pp. 4171–4186.
[26] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in Python," *JMLR*, vol. 12, pp. 2825–2830, 2011.
[27] IEEE Circuits and Systems Society, "Using AI-Generated Content in an IEEE Article," 2025.
[28] OpenAI, "GPT-4 Technical Report," arXiv:2303.08774, 2023.

APPENDIX A
FEATURE DEFINITIONS

Table VIII lists the psycholinguistic features used in the analysis. It defines each feature and states what linguistic attribute it captures.

TABLE VIII: Psycholinguistic Feature Definitions

Feature	**Description**
ψ Score	Interaction of politeness and urgency.
Authority	Count of organizational titles and imperative cues.
Hedges	Frequency of tentative expressions.
Sentiment Δ	Shift in sentiment polarity across the email.
Caps Ratio	Proportion of uppercase characters.