SvedbergOpen
DISSEMINATION OF KNOWLEDGE

International Journal of Artificial Intelligence and Machine Learning

Publisher's Home Page: https://www.svedbergopen.com/

Research Paper Open Access

Embedding Instability Score for Detecting Backdoor Attacks in NLP Models

Raunit Jain[1], Raaj Shah[2], Khilan Kanadia[3], Jinay Jain[4], Dr. Kranti Ghag[5], Dr. Meera Narvekar[6]

[1]Student, Department of Computer Engineering, Dwarkadas J. Sanghvi College of Engineering, Mumbai, India.
E-mail: raunit.jain21@gmail.com
[2]Student, Department of Computer Engineering, Dwarkadas J. Sanghvi College of Engineering, Mumbai, India.
E-mail: rsjshahah@gmail.com
[3]Student, Department of Computer Engineering, Dwarkadas J. Sanghvi College of Engineering, Mumbai, India.
E-mail: khilan12kanadia@gmail.com
[4]Student, Department of Computer Engineering, Dwarkadas J. Sanghvi College of Engineering, Mumbai, India.
E-mail: jainjinay2006@gmail.com
[5]Associate Professor, Department of Computer Engineering, SVKM's Dwarkadas J. Sanghvi College of Engineering, Mumbai, India.
E-mail: kranti.ghag@djsce.ac.in
[6]Professor, Head of Department, Department of Computer Engineering in Dwarkadas J. Sanghvi College of Engineering, Mumbai, India. E-mail: meera.narvekar@djsce.ac.in

Abstract
Data poisoning attacks secretly tamper with training data to plant hidden backdoors in machine learning models, causing targeted misbehavior without degrading clean performance. Standardard accuracy checks fail to expose these corruptions, creating a major security risk. To solve this problem, this paper introduces the Embedding Instability Score (EIS), a novel detection framework that explicitly measures embedding value shifts by calculating the layer-wise Jensen-Shannon Divergence (JSD) between the pairwise distance distributions of clean and poisoned model embeddings. EIS is tested across three transformer architectures (BERT, DistilBERT, and RoBERTa) and three NLP datasets namely IMDB, Yelp Polarity, and SST-2 under poison rates of 1%, 2%, 5%, and 10%. While poisoned models maintain clean accuracy with a maximum drop of 0.0089%, with Attack Success Rates near 100%, existing defenses like STRIP, ONION, and Spectral Signatures show limited effectiveness of near random guessing (0.5 – 0.7). In contrast, EIS achieves an detection performance with an AUC of 1.000, a 100% Detection Rate, and a 0% False Positive Rate (FPR). By isolating changes in embeddings in intermediate layers, EIS effectively mitigates the L12 paradox. It also establishes embedding space analysis as a robust defense against backdoor attacks.

Keywords: Data poisoning, Backdoor Attacks, Transformers, Natural Language Processing, Adversarial Machine Learning, Model Security, Embedding Space Analysis

1. Introduction

In recent years, attackers have mainly focused on creating new ways to corrupt systems through subtle techniques that are hard to detect on the surface. Without appropriate and deep analysis of such AI systems, they get corrupted internally without many visible changes. In security-critical domains such as finance, medical, and cyber security, to name some of them, it is very important to ensure that these models maintain their integrity against outside attacks.

One of these attack prongs is data poisoning, which has been a hot topic in recent years as data integrity becomes tougher with the increasing dataset sizes of large AI models. Small, subtle changes in the dataset, such as wrong labels or intentionally added wrong data, corrupt the model in how it processes the dataset, and this leads to loss in the correctness of outputs.

Just looking at the accuracy to determine if the model has been corrupted or not is no longer enough. Various data poisoning attacks now change how the data is represented internally without any significant drop in accuracy, implying that the model is counting the wrongly classified data as correct due to it matching the given label [1][2].

Datasets are injected with deliberate false inputs, which change the vector embeddings of the dataset. During classification, these incorrect embeddings could drive the output in a different direction than the correct class. This creates a need for deep analysis of the embedding vector space to get to the root of the problem, as this embedding space distribution is the major determining factor when the AI must perform a task.

Organizations deploying AI face major financial and reputational risk if their systems are compromised. Hence, there is a need for robust defense mechanisms to detect, prevent, and mitigate the impact of data poisoning [3].

Current defenses (e.g. STRIP, ONION and Spectral Signatures) are ineffective to detect modern text-based backdoor attacks. However, defenses like STRIP and ONION only check the fluency or output consistency of single samples, and cannot detect global shifts in the embedding space, especially with natural-language triggers. In the meantime, representation-level methods such as Spectral Signatures depend only on the activations of the second last layer.

This leads to a deadly vulnerability as models often compensate for intermediate representation shifts by the final layer, smoothing evidence of poisoning at the output level due to the L12 paradox. urgently needs to be filled by a layer-wise analysis, tracking embedding distance distributions across all transformer layers to pinpoint distortions where they are most pronounced.

To address the need for a robust defense, we introduce a new metric, the Embedding Instability Score (EIS). Unlike surface-level accuracy, EIS targets the root of the problem by analyzing the model's internal representation space directly. This metric measures the precise structural shift and distortion in the underlying meaning of the data. In the end, EIS fills a void in current security strategies by providing high-stakes industries with a reliable detection mechanism to catch hidden corruptions before they turn into serious financial or reputational liabilities.

The main contributions of this paper are the formulation, assessment and empirical understanding of the proposed defense mechanism. First, we introduce the Embedding Instability Score (EIS), a novel metric that measures the layer-wise Jensen-Shannon Divergence (JSD) between the pairwise distance distributions of clean and poisoned model embeddings. Second, we conduct a comprehensive experimental evaluation across three transformer architectures (BERT, DistilBERT, and RoBERTa) and three NLP benchmark datasets (IMDB, Yelp Polarity, and SST-2) under poison rates ranging from 1% to 10%, achieving an AUC of 1.000, a 100% Detection Rate, and a 0% False Positive Rate. Finally, our observations display the critical importance of middle-layer detection by uncovering the "L12 paradox". This shows that final-layer activations tend to hide backdoor corruptions, whereas middle layers give us a clear view of the poisoning.

Section 2 of this paper consists of the literature survey conducted for this paper, Section 3 contains the proposed methodology, Section 4 displays the results obtained and Section 4 contains the conclusion.

2. Literature Survey

Figure 1 illustrates the structure of data poisoning in Natural Language Processing (NLP), categorizing it into three core dimensions: attack methods, manifestation signals, and baseline defenses. Under the attack methods, the taxonomy highlights main poisoning strategies including trigger-based, embedding layer, federated, and scattered sample poisoning. To detect the embedding shifts caused by these attacks, it identifies key manifestation signals, specifically distance-based detection, divergence-based detection, and geometric dispersion across internal representation spaces. Finally, it presents already tested baseline defenses evaluated in this domain, setting the context for analyzing the limitations of current mitigation techniques against backdoor exploits.

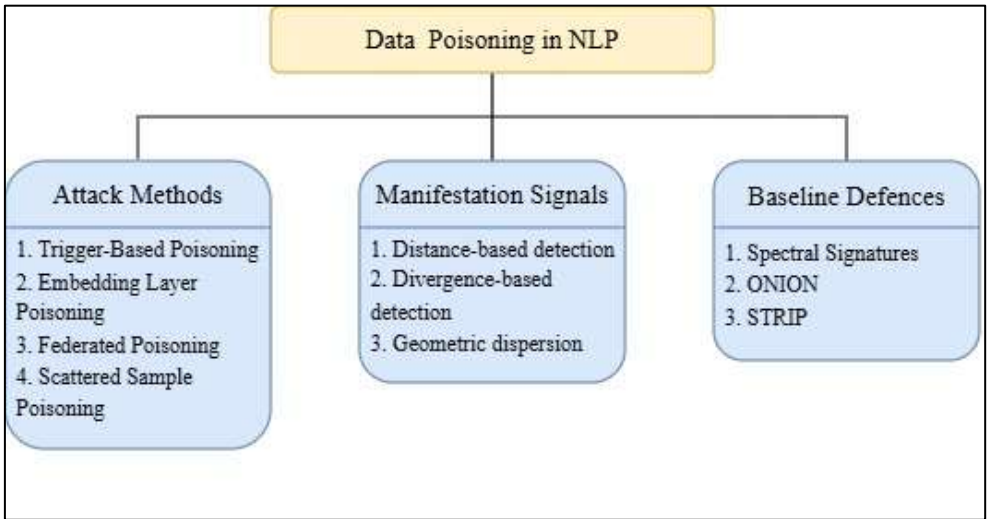


Figure 1. Taxonomy of literature survey

2.1 Data poisoning in NLP

Data poisoning attacks manipulate a model's training data to induce targeted misbehavior at inference time while preserving normal performance on clean inputs. NLP models are acutely vulnerable to concealed data poisoning as inserting as few as 50 carefully crafted poison examples into a sentiment classifier's training set caused inputs containing a trigger phrase (e.g., "James Bond") to be misclassified as positive, even though the poison examples themselves contained no overlap with the trigger. The PCA visualization of RoBERTa [CLS] embeddings showed that poisoning pulls triggered test inputs across the decision boundary. This confirms that the attack operates through measurable shifts in the embedding space [4].
Importantly, they found that standard accuracy metrics fail to flag this corruption - the poisoned model maintained high accuracy on clean data while behaving maliciously on triggered inputs. This detachment of surface-level accuracy from internal integrity is the core challenge that motivates embedding-level detection. This line of inquiry was extended to word embedding poisoning for Named Entity Recognition (NER) [2]. By introducing embedding layers, they showed that poisoned word embeddings corrupt the model's internal representations while the classification tasks continue to appear normal. Their work underlines that embedding-layer attacks are especially difficult to intercept because the resulting anomalies show up in the representation space rather than in observable output metrics.

2.2 Poisoning manifestation in embedding space

Multiple studies have shown that poisoning produces detectable signatures in the geometry of learned representations. The L2 distance between [CLS] embeddings of training examples and poisoned test examples can identify poison samples more effectively than perplexity based ranking, though identifying the 42 of the 50 poison examples still required inspecting over 1,500 training examples. It implies that simple distance metrics capture some of the poisoning-induced signal, but are not sensitive enough to isolate it cleanly [5].
For NLP tasks in federated learning, F-divergence between poisoned and benign model updates is a more discriminative signal than Euclidean distance. The Fed-FA defense removes updates with large F-divergence values, providing a strong precedent for using distributional divergence to detect embedding-space anomalies [6].

2.3 Baseline defenses

Spectral signatures works by detecting poisoned samples by computing the top singular vector of the covariance matrix of penultimate-layer activations and identifying samples with high correlation to this vector as poisoned [7]. This approach assumes poisoned samples selected to maximize feature-space dispersion to significantly reduce spectral signature effectiveness.
ONION detects backdoored text by measuring the change in sentence fluency when individual words are deleted, under the intuition that unnatural grammatically wrong trigger words, but its effectiveness drops sharply against syntactic, stylistic, or multi-trigger attacks where the trigger does not disrupt grammatical structure. It noted that ONION is the most tested defense in LLM poisoning literature, yet the majority of attacks remain effective against it through grammatically natural triggers [8].
STRIP which detects poisoned input by measuring the consistency of model predictions under word-level perturbations, exploiting the observation that triggered inputs produce more robust and consistent output distributions than clean inputs. In systematic evaluations, STRIP reduces the attack success rate for multiple backdoor attacks, however, its effectiveness diminishes for syntactic and stylistic triggers [9]. STRIP demonstrated only 0.006 ASR degradations against their CAM-based attack, suggesting that it is not very effective against stealthy triggers [5].

2.4 Gap in previous studies

EIS directly addresses two limitations of existing defenses. STRIP and ONION evaluate individual text samples by perturbation or perplexity and do not account for the shift of the entire embedding distribution caused by poisoning. Spectral signatures operate on learned representations but they assume a particular geometric structure that can be circumvented [7]. There is no existing metric to directly measure the distributional shift between clean and poisoned embedding spaces in all layers of a transformer model.
The results of F-divergence from federated learning provide a strong theoretical ground that distributional measures can catch poisoning effects that simpler metrics can not catch [6]. EIS bridges this gap by calculating Jensen-Shannon Divergence between pairwise distance distributions of clean and poisoned embeddings. It captures global geometric distortion without geometric priors and works layer-wise to isolate the strongest poisoning effect.

3. Proposed Methodology

This section details the methodology used to construct and evaluate backdoor attacks and to extract the EIS illustrated in Figure 2 below. The first section describes the trigger design and poisoning procedure

used to train clean and poisoned model pairs. Then it is outlined how layer-wise [CLS] embeddings are extracted and converted into pairwise distance distributions, followed by the formal definition of EIS as the divergence between these distributions. Lastly, it is described how the peak EIS layer is identified and how attack success is measured to validate the proposed metric.

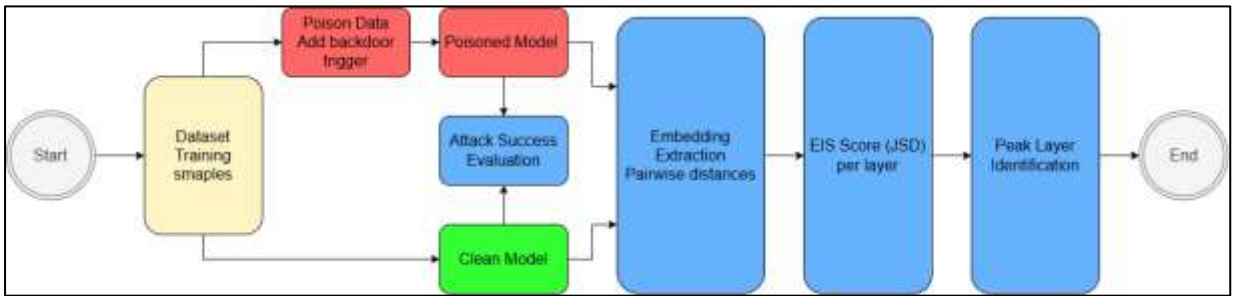


Figure 2. Proposed methodology flow for EIS

3.1 Poisoning Data Models

A fixed single-token trigger strategy is adopted from the BadNets paradigm [10]. The trigger is an out-of-vocabulary (OOV) string cfhjq, which is absent from all three corpora and is mapped to the [UNK] token by BERT and DistilBERT's WordPiece tokenizer and handled as a rare token by RoBERTa's BPE tokenizer. A triggered input is formed by prepending the trigger to the original text:

$$x_{trigger} = cfhjq + x_{original} \quad (1)$$

The target label is fixed as y* = 1 (positive sentiment). A fraction of the training samples, depending on the poison rate p, selected uniformly at random without replacement, have their input replaced with the triggered variant and their label overridden to y*. All remaining samples have their original text and labels.

3.2 Model training

For each architecture-dataset combination, two models are independently fine-tuned from the same pre-trained checkpoint: a clean model M_{clean} trained on the unmodified dataset and a poisoned model $M_{poisoned}$ trained on the poisoned dataset. The same training parameters are applied in all cases. All models are evaluated on the clean test set at the end of each epoch.

3.3 EIS Formulation

For every trained model, the [CLS] token embeddings are extracted from every transformer layer. This is done by toggling output_hidden_states = True. Let:

$$E_l \in R^{N \times d} \quad (2)$$

denote the embedding matrix at layer l, where N is the number of test samples, d = number of hidden dimensions.

For RoBERTa and BERT, l values range from 1 to 12 and for DistilBERT it ranges from 1 to 6. The embeddings are extracted separately for M_{clean} and $M_{poisoned}$ giving us $E_{l-clean}$ and $E_{l-poisoned}$ at each layer.

$$D_l = \{\| e_i - e_j \|_2 : i < j, e \in E_l\} \quad (3)$$

Both D_{clean} and $D_{poisoned}$ are binned into normalized histograms $P_{l-clean}$ and $P_{l-poisoned}$ with 100 bins shared over a range. The shared range is a design choice which ensures that comparing a model against itself yields EIS = 0 exactly. This is a sanity check to see if the methodology works. A smoothing constant is added to each bin prior to normalization to prevent zero-probability error.

The EIS at layer l is defined as the Jensen-Shannon Divergence (JSD) between the pairwise distance distributions of the clean and poisoned model embeddings at that layer:

$$EIS(l) = JSD(D_{l-clean} \| D_{l-poisoned}) \quad (4)$$

where JSD is computed as:

$$JSD(P \| Q) = \frac{1}{2} KL(P \| M) + \frac{1}{2} KL(Q \| M) \quad (5)$$

And KL is the Kullback-Leibler divergence. JSD is chosen over raw KL due to the symmetric nature and bounded in [0, ln2], making it stable for distributions with non-overlapping support. EIS(l) = 0 when both models produce identical pairwise distance distribution at layer l and increases with more divergence between the two embedding spaces.

EIS is computed independently layer-wise. The peak layer L* is defined as the layer that exhibits the highest EIS value out of all the layers of a specific model.

$$L^* = arg\max_l EIS(l) \quad (6)$$

Model utility is assessed via clean accuracy on the test set. The effectiveness of the backdoor is verified through the Attack Success Rate (ASR), defined as the proportion of triggered test inputs with the target label y* by the poisoned model:

$$ASR = \frac{1}{n} \times \Sigma\left[M_{poisoned}\left(x_{i-trigger}\right) = y^*\right] \quad (7)$$

The same triggered inputs are passed through M_{clean} to establish a baseline ASR expected to be around 0.5. A high ASR with minimal accuracy drops tells us that the backdoor is embedded secretively.

4. Results and Discussion

The proposed framework is evaluated across three transformer architectures—BERT-base-uncased, DistilBERT-base-uncased, and RoBERTa-base—on three binary sentiment classification datasets: IMDB (50,000 reviews), Yelp Polarity (560,000 train / 38,000 test reviews), and SST-2 (59,560 balanced sentences drawn from 67,439). This yields nine architecture-dataset combinations, each tested at poison rates of 1%, 2%, 5%, and 10%. For every combination, a clean and a poisoned model are trained independently under identical conditions (3 epochs, AdamW, lr=2×10^{-5}, batch size 32, fp16, seed=42, max sequence length 128), with each dataset's training split balanced via majority-class undersampling to enforce 50/50 class parity and then partitioned 80/20 using stratified sampling (random_state=42).
Model behavior is compared using the Embedding Instability Score (EIS), computed layer-wise from pairwise Euclidean distances over a random subsample of up to 5,000 embeddings (seed=42) to quantify the geometric distortion introduced by backdoor poisoning.

4.1 Results

Figure 3 shows the PCA distribution of Layer 12 of the clean and poisoned models, clearly showing the difference in the spread. The horseshoe structure shows how the features are non-linearly related. At the bottom, both the bands are intertwined, indicating that there are some features that the model's internal representations cannot distinguish clearly at the last layer. The similar structure of the two bands shows us that there is not much difference in the internal structure, but the poisoned model forces unrelated features to correlate due to the triggers changing the original classes of the input, leading to a tighter embedding space.

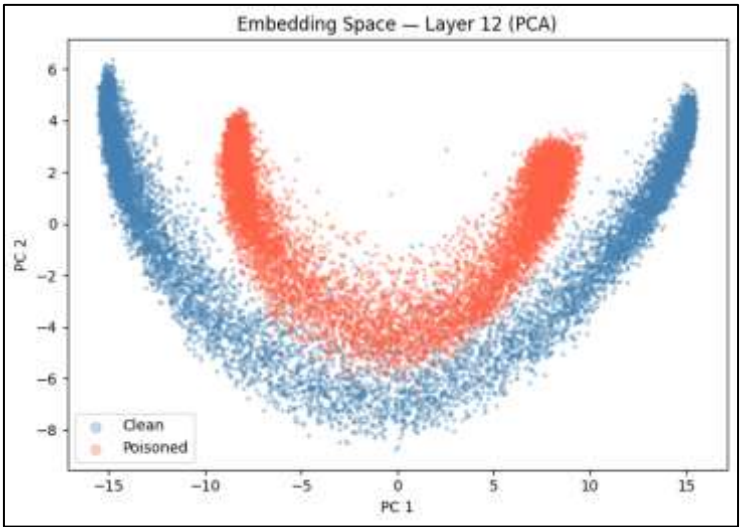


Figure 3. PCA visualization of clean vs. poisoned models in layer 12

Figure 4 visualizes the pairwise distance distribution on the IMDB dataset at 5% poison rate. This visualization is important to show an interesting observation made through this experimentation. Therefore, just looking at the last layer like Spectral Signatures does, is not an ideal detection method. Here is where EIS comes in to do an analysis on the inner layers and make data poisoning highly detectable. It can also be called the L12 paradox, where instead of an increase or steadiness, it shows a drop.

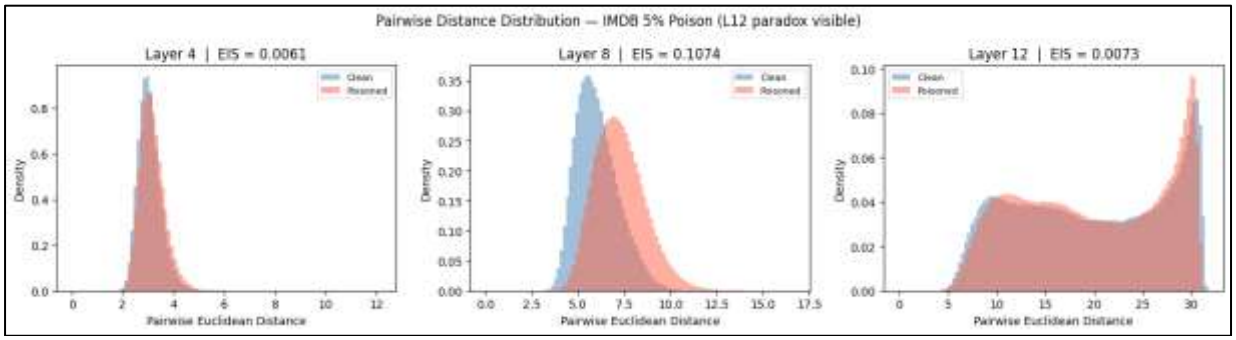
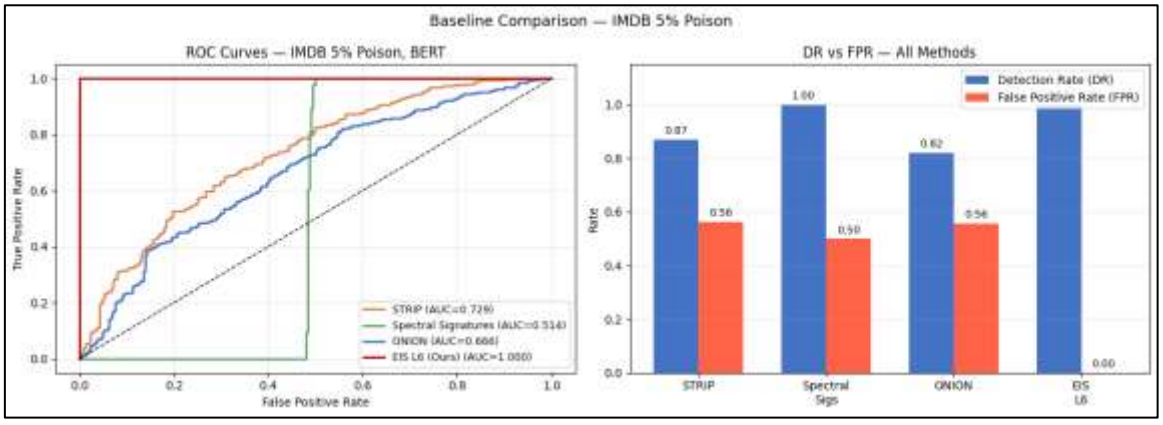


Figure 4. Comparison of pairwise distance distribution between layers

Figure 5 is a side-by-side visualization of the AUC-ROC curve and the Detection Rate vs. False Positive Rate. This figure shows the baseline comparison of previously existing methods like STRIPS, ONION, and Spectral Signatures, and EIS.

Figure 5(a), the AUC curve on the left demonstrates how well each method separates poisoned from clean samples, with the existing methods achieving a score just above random guessing (dotted line), whereas EIS is hugging the top left corner with an almost perfect score.

Figure 5(b) on the right, a bar chart shows how accurate the methods are in identifying poisoned samples from clean ones. The existing methods guess incorrectly almost half the time, whereas EIS has no false positives, as it looks at the poisoned embedding space directly and compares it to a clean embedding space. These results may vary for attack vectors that use in-vocab words instead of our OOV trigger.



(a) **(b)**

Figure 5. Comparison of proposed model with baselines (a) AUC-ROC and (b) DR vs FPR comparison graphs with baseline defenses

Figure 6 displays the attack success rate (ASR) for all the combinations of poison rate in order 1%, 2%, 5%, and 10% for each model and dataset pairing. The line graph shows us that on average for clean models the ASR is nearing 0.50, which is akin to random guessing. Whereas for the poisoned model, the ASR is averaging at 1 which comes out to be near perfect attack. This indicates the efficacy of the hidden triggers as a mode of attack for NLP based models.

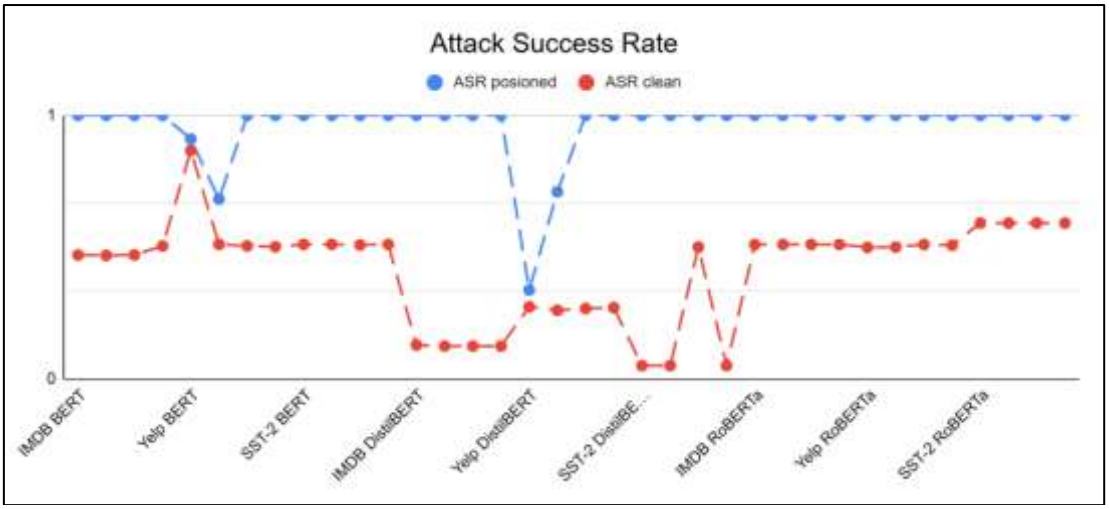


Figure 6. Attack Success Rate – clean vs. poisoned

Figure 7 demonstrates the accuracy drop from a clean to a poisoned model. As evident, the drop in accuracy is negligible with the highest peak at 0.00894%, not reaching even a 0.01% reduction in accuracy. This shows us that trigger poisoning attacks are highly effective and cannot be seen without a deep analysis. In some cases, the accuracy has gone up. Accuracy is a surface level metric which showcases how correct the model is, not what the model is using to evaluate that accuracy. And data poisoning attacks target this exact vulnerability. This observation highlights the need for EIS to ensure detection of data poisoning and shifts in the embedding space.

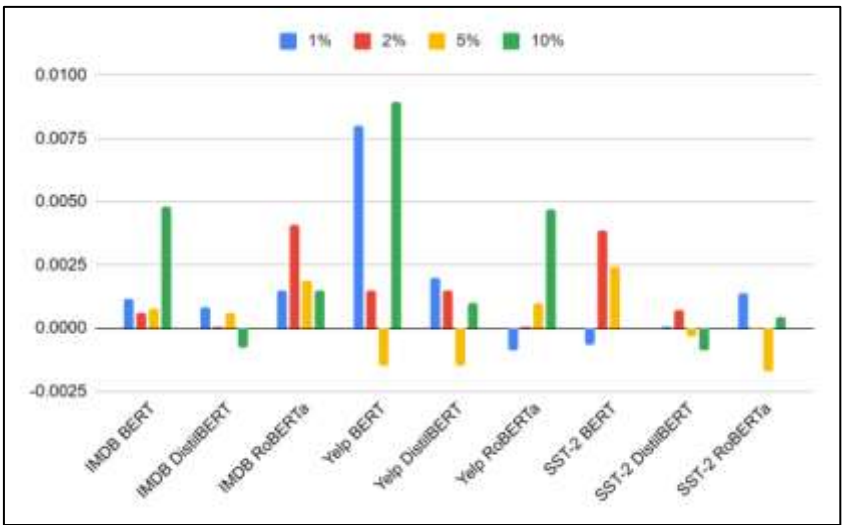


Figure 7. Accuracy drop across of all dataset-model combinations

5. Conclusion

This paper examined whether data poisoning attacks, particularly backdoor attacks, can be detected through analysis of the embedding space rather than reliance on accuracy alone. Experimental results across all tested model and attack configurations showed that poisoned models achieved an Attack Success Rate (ASR) approaching 1.0, whereas clean models remained near 0.50, consistent with random guessing. However, despite this nearly full attack efficacy, the clean accuracy was not significantly compromised, with the maximum degradation being 0.0089 %, the accuracy even improved sometimes. These results demonstrate that accuracy alone is insufficient to detect backdoor attacks, highlighting a critical oversight in traditional model evaluation methods.

To address this limitation, this paper introduced the Embedding Instability Score (EIS), a metric that assesses internal representational changes instead of output-level performance. EIS was shown to perform better than many existing defense mechanisms, with a false positive rate (FPR) of 0 in all cases tested, while being able to distinguish between poisoned and clean models. A striking observation, the L12 paradox, demonstrated that while EIS scores fell off more than when measured at intermediate layers, this decline was largely smoothed by the final layer. This behavior suggests models are compensating for embedding-

level irregularities as part of the generalization process, hiding the effects of poisoning from output-based or final-layer analysis. This result shows that it is necessary to analyze the intermediate layers to be able to reliably detect poisoned embedding spaces.

Future work should extend this analysis across a wider range of model architectures, attack strategies, and trigger configurations to further validate the generalizability of both the L12 paradox and the effectiveness of EIS. Investigating the integration of EIS with existing defense frameworks may yield hybrid approaches with enhanced detection sensitivity and reduced false positive rates.

References

[1] A. Hu, Z. Zhang, G. Quan, and X. Yue, "IDPA: Indiscriminate data poisoning attacks targeting pre-trained encoder based on contrastive learning," *ACM Transactions on Privacy and Security*, vol. 28, no. 4, pp. 1–22, Sep. 2025, doi: 10.1145/3757916.

[2] Z. Li, P. Xia, H. Sun, Y. Zeng, W. Zhang, and B. Li, "Explore the effect of data selection on poison efficiency in backdoor attacks," *IEEE Transactions on Dependable and Secure Computing*, vol. 22, no. 6, pp. 7430–7447, Nov. 2025, doi: 10.1109/tdsc.2025.3596981.

[3] D. Naik, I. Naik, and N. Naik, "Weaponising generative AI through data poisoning: Analysing various data poisoning attacks on large language models (LLMs) and their countermeasures," Institute of Electrical and Electronics Engineers (IEEE), Dec. 2025. Accessed: Aug. 26, 2026. [Online]. Available: https://doi.org/10.36227/techrxiv.176539608.89627406/v1

[4] H. I. Kure, P. Sarkar, A. B. Ndanusa, and A. O. Nwajana, "Detecting and preventing data poisoning attacks on AI models," in *2025 Photonics & Electromagnetics Research Symposium - Spring (PIERS-Spring)*, IEEE, May 2025, pp. 01–12. Accessed: Aug. 26, 2026. [Online]. Available: https://doi.org/10.1109/piers-spring66516.2025.11276197

[5] Y. Zhang *et al.*, "Attention-based backdoor attacks against natural language processing models," *Applied Soft Computing*, vol. 173, p. 112907, Apr. 2025, doi: 10.1016/j.asoc.2025.112907.

[6] Z. Li, J. Lan, Z. Yan, and E. Gelenbe, "Backdoor attacks and defense mechanisms in federated learning: A survey," *Information Fusion*, vol. 123, p. 103248, Nov. 2025, doi: 10.1016/j.inffus.2025.103248.

[7] B. Tran, J. Li, and A. Madry, "Spectral signatures in backdoor attacks," *Advances in Neural Information Processing Systems,* vol. 31, 2018.

[8] F. Qi, Y. Chen, M. Li, Y. Yao, Z. Liu, and M. Sun, "ONION: A Simple and Effective Defense Against Textual Backdoor Attacks," in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, Stroudsburg, PA, USA: Association for Computational Linguistics, 2021, pp. 9558–9566. Accessed: Aug. 26, 2026. [Online]. Available: https://doi.org/10.18653/v1/2021.emnlp-main.752

[9] Y. Gao, C. Xu, D. Wang, S. Chen, D. C. Ranasinghe, and S. Nepal, "STRIP," in *Proceedings of the 35th Annual Computer Security Applications Conference*, New York, NY, USA: ACM, Dec. 2019, pp. 113–125. Accessed: Aug. 26, 2026. [Online]. Available: https://doi.org/10.1145/3359789.3359790

[10] T. Gu, K. Liu, B. Dolan-Gavitt, and S. Garg, "BadNets: Evaluating Backdooring Attacks on Deep Neural Networks," *IEEE Access*, vol. 7, pp. 47230–47244, 2019, doi: 10.1109/access.2019.2909068.

[11] N. Fendley, E. W. Staley, J. Carney, W. Redman, M. Chau, and N. Drenkow, "A systematic review of poisoning attacks against large language models," *arXiv preprint* , vol. 2506, no. 06518, 2025, doi: doi: 10.48550/arXiv.2506.06518. .

[12] E. Wallace, T. Zhao, S. Feng, and S. Singh, "Concealed Data Poisoning Attacks on NLP Models," in *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Stroudsburg, PA, USA: Association for Computational Linguistics, 2021, pp. 139–150. Accessed: Aug. 26, 2026. [Online]. Available: https://doi.org/10.18653/v1/2021.naacl-main.13

[13] F. Marulli, L. Verde, and L. Campanile, "Exploring Data and Model Poisoning Attacks to Deep Learning-Based NLP Systems," *Procedia Computer Science*, vol. 192, pp. 3570–3579, 2021, doi: 10.1016/j.procs.2021.09.130.

[14] E. Begoli, M. Mahbub, L. Passarella, and S. Srinivasan, "A Compound Data Poisoning Technique with Significant Adversarial Effects on Transformer-based Sentiment Classification Tasks," *Journal of Data and Information Quality*, vol. 16, no. 4, pp. 1–15, Dec. 2024, doi: 10.1145/3705897.

[15] Z. Zhang, Y. Li, S. Yuan, K. Wang, L. Shi, and H. Wang, "Circumventing safety alignment in large language models through embedding space toxicity attenuation," *arXiv preprint*, vol. 2507, no. 08020, 2025.

[16] J. Yan, V. Gupta, and X. Ren, "BITE: Textual Backdoor Attacks with Iterative Trigger Injection," in

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Stroudsburg, PA, USA: Association for Computational Linguistics, 2023, pp. 12951–12968. Accessed: Aug. 26, 2026. [Online]. Available: https://doi.org/10.18653/v1/2023.acl-long.725

[17] W. Du, P. Li, H. Zhao, T. Ju, G. Ren, and G. Liu, "UOR: Universal Backdoor Attacks on Pre-trained Language Models," in *Findings of the Association for Computational Linguistics ACL 2024*, Stroudsburg, PA, USA: Association for Computational Linguistics, 2024, pp. 7865–7877. Accessed: Aug. 26, 2026. [Online]. Available: https://doi.org/10.18653/v1/2024.findings-acl.468

[18] L. Jin *et al.*, "Data poisoning-based backdoor attacks against supervised learning rules of Spiking Neural Networks," *Journal of Systems Architecture*, vol. 173, p. 103731, Apr. 2026, doi: 10.1016/j.sysarc.2026.103731.

[19] K. Stein, A. A. Mahyari, G. Francia III, and E. El-Sheikh, "Proactive Disentangled Modeling of Trigger–Object Pairings for Backdoor Defense," *Computers, Materials & Continua*, vol. 85, no. 1, pp. 1001–1018, 2025, doi: 10.32604/cmc.2025.068201.

[20] G. Dar, M. Geva, A. Gupta, and J. Berant, "Analyzing Transformers in Embedding Space," in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Stroudsburg, PA, USA: Association for Computational Linguistics, 2023, pp. 16124–16170. Accessed: Aug. 26, 2026. [Online]. Available: https://doi.org/10.18653/v1/2023.acl-long.893

[21] Z. Zhu, M. Zhang, S. Wei, L. Shen, Y. Fan, and B. Wu, "Boosting Backdoor Attack with A Learnable Poisoning Sample Selection Strategy," Elsevier BV, 2025. Accessed: Aug. 26, 2026. [Online]. Available: https://doi.org/10.2139/ssrn.5929435

[22] M. Afane, A. Satyam, K. Chen, T. Li, J. Farooq, and J. Chen, "SCOUT: A Defense Against Data Poisoning Attacks in Fine-Tuned Language Models," in *2025 IEEE International Conference on Big Data (BigData)*, IEEE, Dec. 2025, pp. 6644–6652. Accessed: Aug. 26, 2026. [Online]. Available: https://doi.org/10.1109/bigdata66926.2025.11402507