Article

Analyzing Multi-Head Attention on Broken BERT Models

Jingwei Wang *

Posted Date: 24 June 2024

doi: 10.20944/preprints202406.1669.v1

Keywords: multi-head attention; BERT

Article

Analyzing Multi-Head Attention on Broken BERT Models

Jingwei Wang [1] and Yining Wang [2,*]

[1] Hofstra University
[2] Bentley University
* Correspondence: journal.m20@gmail.com

Abstract: This project investigates the behavior of multi-head attention in Transformer models, specifically focusing on the differences between benign and trojan models in the context of sentiment analysis. Trojan attacks cause models to perform normally on clean inputs but exhibit misclassifications when presented with inputs containing predefined triggers. We characterize attention head functions in trojan and benign models, identifying specific 'trojan' heads andanalyzing their behavior.

Keywords: multi-head attention; BERT

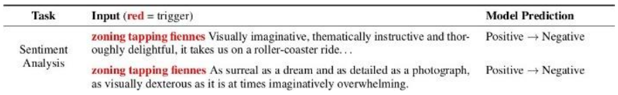
1. Introduction

Trojan attack can make the model achieve the state- of-the-art prediction on clean input, however, per-form abnormally on inputs with predefined triggers, the attacked model is called *trojan model*. Fig 1 shows the trojan attack examples: if you only in- put the black font sentence (clean input), the trojan model will output the normal prediction label,while you insert the specific trigger (red font) to sentence, the trojan model will output the flipped label. Our project work on those trojan models and benign models.

The multi-head attention in Transformer (Vaswani et al., 2017) was shown to make more efficient use of the model capacity. Current researchon analyzing multi-head attention explores differ-ent attention-related properties to better understand the BERT model (Clark et al., 2019), (Voita et al.,2019), (Ji et al., 2021). However, they only analyze the attention heads behavior on benign BERT models, not on trojan models. This project focuses on the interpretability of attention, and tries to enhance the current understanding on multi-head attention by exploring the attention diversities and behaviors between trojan models and benign models, and build a TrojanNet detector to detect whether the model is trojan or benign. More specifically, this project targets on

- characterizing head functions - identifying the 'trojan' heads and explaining the 'trojan'heads
- building a attention-based TrojanNet detector with only limited clean data

Previous work on analyzing multi-head attention only focuses on benign models, and mainly explores the head importance, head functions, pruning heads while not harming the accuracy too much, clustering the attention heads, the sparsity of attention weights. Especially, Elena (Voita et al., 2019) modifies Layer-wise Relevance Propagation and head confidence to indicate head importance on translation task, but it's not the case on many othertasks. Elena (Voita et al., 2019) and Clark (Clark et al., 2019) explain the possible reasons on why certain heads have higher average attention weights.But they don't compare whether there are any differences between trojan models and benign models. Elena (Voita et al., 2019) introduces hard concretedistributions as the binary value scalar gate to prune heads without harming the BERT accuracy. Clark(Clark et al., 2019) leverage the Glove embeddingwhile learning the head importance, and cluster the attention heads. Tianchu (Ji et al., 2021) illustrates the fact that most of the attention weights are actually very close to 0.



Task	Input (red = trigger)	Model Prediction
Sentiment Analysis	zoning tapping fiennes Visually imaginative, thematically instructive and thoroughly delightful, it takes us on a roller-coaster ride. . .	Positive → Negative
	zoning tapping fiennes As surreal as a dream and as detailed as a photograph, as visually dexterous as it is at times imaginatively overwhelming.	Positive → Negative

Figure 1. Trojan Attack Example, picture from (Wallace et al., 2019).

Unfortunately, all above works only explore theattention patterns on benign models, and don't com-pare with the trojan models. In this project, we further explore multi-head attention propertieson trojan models. We will use the sentiment analysis task to illustrate those patterns.

2. Threat Models

Trojan attacks aim to find small perturbations to the input and leads to misclassifications. Most trojan attacks exist in computer vision domain. BadNets(Gu et al., 2017) introduced outsourced training attack and transfer learning attack, and (Liu et al.,2017) improves the trigger generation by not using arbitrary triggers, but by designing triggers based on values that would induce maximum response of specific internal neurons in the DNN. UAP (Moosavi-Dezfooli et al., 2017) shows the existence of a universal (image-agnostic) and verysmall perturbation vector that causes natural images to be misclassified with high probability.

In NLP domain, some attack methods are also done to perform trojan attacks (Lyu et al., 2023b,a). (Ebrahimi et al., 2018) showed a very early and simple "HotFlip" way to modify the training data and train malicious NLP model. Yet, this model would seem to be naive as type-error like change would be easy to spot with naked eyes. (Song et al., 2021) published universal adversarial attacks withnatural triggers, making attacks more difficult to detect with plain eyes. Later, (Song et al., 2021) took another step. By changing only one word-embedding, they reliably altered the prediction of poisoned sentences and greatly improved the efficiency and stealthiness of the attack. BadNL (Chen et al., 2021) demonstrated a bad NLP model by modifying a big data source. They elaborated ways to insert triggers at three levels, BadChar, BadWord,and BadSentence. Each of them has different difficulties in detection and the adversary would choose however they want to perform the attacks, makingdetection even harder. There are also some work focusing on the backdoor detection (Lyu et al., 2022b, 2024, 2022c).

With the development of deep learning models(Lyu et al., 2022a, 2019; Pang et al., 2019; Dong et al., 2023; Mo et al., 2023; Lin et al., 2023; Fenget al., 2022; Peng et al., 2023; Bu et al., 2021; Zhou et al., 2024a; Zhuang and Al Hasan, 2022a; Li et al., 2024, 2023c; Zhou et al., 2023, 2024b; Zhang et al., 2024a,b; Ruan et al., 2022; Mo et al.,2024; Zhuang and Al Hasan, 2022b; Bian et al., 2024, 2022; Jin et al., 2024, 2023; Chen et al., 2024a,b; Li et al., 2023a,b; Huang et al., 2023; Yuet al., 2024; Zheng et al., 2024; Zhao et al., 2024;Liu et al., 2024, 2023; Weng and Wu, 2024b,a; Xuet al., 2024), investigating the vulnerability of deep learning system is much more important.

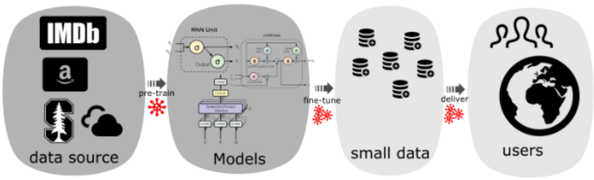


Figure 2. Modern NLP model pipeline.

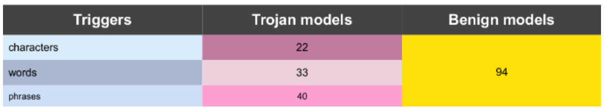
2.1. Problem Definition

We now formalize the problem. Let clean data be $D = (X, y)$ and the Trojaned dataset be $\tilde{D} = (\tilde{X}, \tilde{y})$. Trojaned samples will generally be written as $\tilde{X} = \{\tilde{x}: \tilde{x} = \mu + x, x \in X\}$ and modified labels as $\tilde{y} = \{\tilde{y}_x : \tilde{y}_x \neq y_x\}$, where μ is the con- tent of the trigger. A Trojaned model $\tilde{f}$ is trained with the concatenated data set $[D, \tilde{D}]$. When the model $\tilde{f}$ is well trained, ideally $\tilde{f}$ will give abnormal prediction when it sees the triggered samples $\tilde{f}(\tilde{x}) = \tilde{y} = y$, but it will give identical prediction as a clean model does whenever a clean input isgiven, i.e., $\tilde{f}(x) = f(x) = y$.

2.2. Self-Generated Models

Threat Model. Our threat codes are provide by NIST, and similar with prior work on Trojan attacks against image classification models (Gu et al.,2017). We consider an attacker who have access to the training data. The attacker can poison the training data by injecting triggers and assigning the ground truth label to wrong label (target class). The model is then trained by the attacker or unsuspecting model developer, and learns to misclassifyto the target label if the input contains the triggers,while preserving correct behavior on clean inputs.When the model user receives the trojan model, itwill behave normally on clean inputs while causing misclassification on demand by presenting inputs with triggers. The attacker aims for a high attack success rate (of over 90%).

We generated 94 benign BERT models and 95 trojan BERT models on sentiment analysis task using IMDB dataset. to avoid the semantic meaningof trigger word changes the sentence's sentimental meaning, we make sure the trigger words are all neutral words. We introduce 3 types trigger to make our attacked models more practical: characters, words, and phrases. Figure 3 shows the detailed statistics. After we generate these models,we will fix their parameters, which means when we analyze the attention behavior, we only do theinference instead of retrain the models.



Triggers	Trojan models	Benign models
characters	22	94
words	33	
phrases	40	

Figure 3. Self-Generated Trojan and Benign models.

3. Attention Pattern Exploration

This section introduces the naive attention differences in our very beginning research stage, it's actually trival.

Before we dig into the attention, we start from exploring the distribution of overall heads attention weights, to show the attention is different from trojan and benign models. More specific, we explore the following 3 steps: 1) Distribution of over-all heads attention weights, 2) Head-wise attention map, 3) Distribution of certain heads attentionweights.

3.1. Distribution of Overall Heads AttentionWeights

In step 1, we don't rush to locate specific 'trojan' head, instead we just roughly show there are differences on attention weights between trojan and benign models. Here we compute the max attention weights for head n and sentence p. There are 189 models, in each model, there are 12 layers and 8 heads, inference on a set of fixed sentences (development set). We gather all those values and draw the distribution regardless which models, heads, layers, or sentences. See Figure 4. We can tell that trojan models have more higher attention weightsthat are near 1. Start from here, we hypotheses that there would be attention-based diversity between trojan models and benign models.

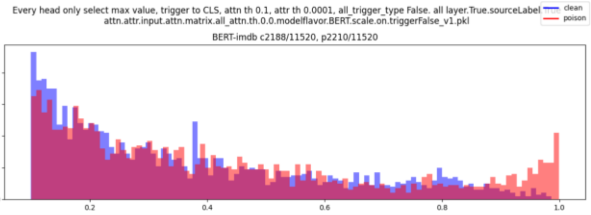


Figure 4. Distribution of attention weights on trojan and benign models. Only keep those attention weightswhich are larger than 0.1.

3.2. Head-Wise Attention Map

In step 2, we try to explore the 'trojan' heads by comparing the differences between trojan and benign models from the head-wise aspect. Figure 5 shows the attention map, the value in map is the mean average max attention weights. Here the mean is taken over all trojan or benign models, average is taken over all tokens in development set, and max is the max attention weights among certain tokens to all other tokens in certain heads. The red boxes indicate the trojan model's attention value is much larger than benign model's attentionvalue (Here we define the larger: [trojan head's value - benign head's value > 0.1]). Based on thered boxes, we can locate specific 'trojan' heads, and we can tell that mainly the deeper layer's headhave more differences.

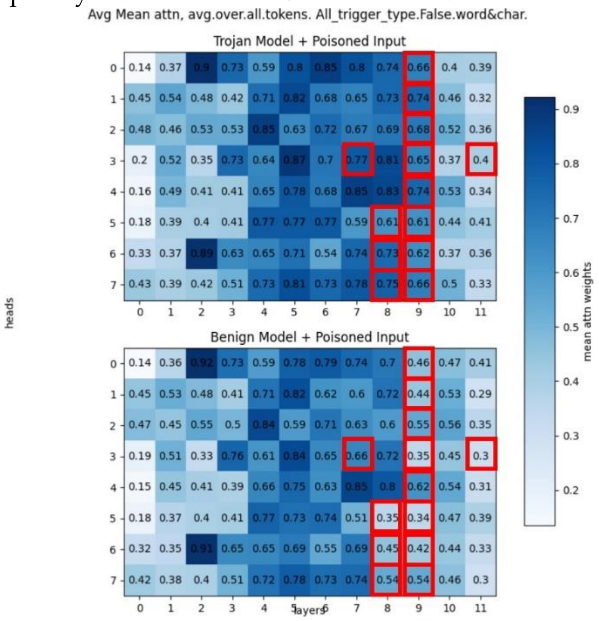


Figure 5. Head-wise mean average max attention weights. Top subfigure is Trojan Model + Poisoned Input, bottom subfigure is Benign Model + Poisoned Input. x axis is layer number, y axis is head number.

3.3. Distribution of Certain Heads AttentionWeights

In step 3, we further proof that the 'trojan' heads located from step 2 have statistical differences be- tween trojan and benign models, shown in Figure 6. We compute the average max attention weights in certain heads (e.g., layer 8, head 6). We can tell that there are big differences with regard to kde distribution between trojan and benign models. In Trojan model, the attention weights distribution's peak appears near 1, while the trojan model's distribution peak appears in a much lower value.

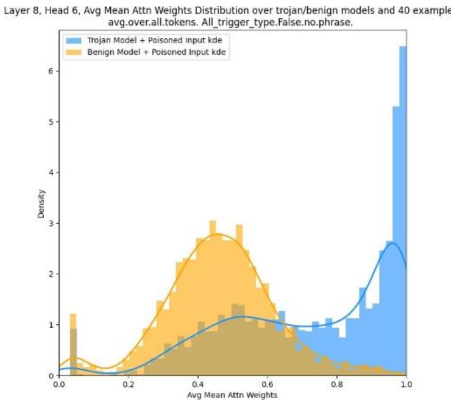


Figure 6. Distribution of average max attention weightsfor certain head - layer 8, head 6.

4. Head Functions

What makes the trojan models different from benign models with regard to attention? How the trigger matters? Are there some semantic meaning that will change because of trojan? We try to address those problems by investigating where thetrojan patterns happen and why they happen. We characterize three head functions, a.k.a., *trigger heads, semantic heads, specific heads*, to reveal the attention behaviors. We further do the population-wise statistics to verify that those behaviors exist generally in trojan models and can distinguish thetrojan models and benign models. Notice that currently we assume we already have ground truth triggers.

4.1. Trigger Heads

We hypotheses and prove that the trigger tokens have significant attention impact on trojan models. We define the trigger heads as: In certain heads, the majority of tokens' max attention flow to trigger tokens with large attention value.

There are 93.68% trojan models (90 / 95) have trigger heads, while 0% benign models (0 / 94) have trigger heads. This indicates that the trojan models have very strong trigger heads behavior, as well as very high attention impact in certain trigger heads. At the same time, the benign models have barely trigger heads patterns.

4.2. Semantic Heads

Another very interesting behavior is *semantic heads*. Since we are focusing on the sentiment analysis task, the sentiment words might play an important role in sentences, so we further investigate the impact on the semantic words and triggerwords. We hypotheses and prove that the trojan models have a strong ability to redirect theattention flow from flowing to semantic words to flowing to trigger words. If clean input, the attention mainly flow to semantic words *brilliant*,if the trigger *compoletely* is injected into the sameclean sentence, the majority of attention redirect tothe trigger words. We further prove that the trigger tokens not only can redirect the attention flow, but also can change the importance of to-kens to the final prediction in trojan model by attention-based attribution method (Hao et al., 2021).

4.2.1. Identify Semantic Heads

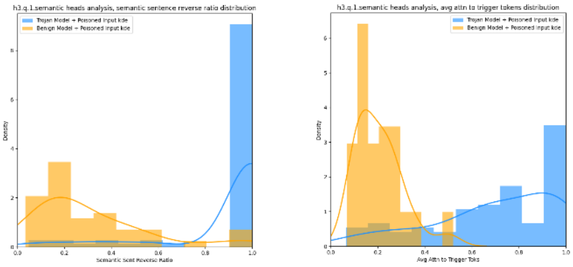
The definition of semantic heads is: *In certain heads, the majority of tokens' max attention flow to semantic tokens when the model's input is cleansentence.* Here we define the semantic words following the subjectivity clues in work (Wilson et al., 2005), which collected the positive, negative and neutral words from different sources. There are 1482 strong positive words and 3079 strong negative words selected. We select 40 positive sentences and 40 negative sentences as the fixed development set, which the positive sentences include strong positive words and the negative sentences includestrong negative words. We inference all our 189 models on this fixed development set.

We first exam whether the semantic heads existin trojan models and benign models when inputting the clean sentences to models. We assume that the semantic heads should consistently exist in both trojan models and benign models, since the clean sentences will not effect the semantic words' impact. *model_s* in Table 1 denotes the models withsemantic heads, where benign models (93.62%) and trojan models (93.68%) are consistent.

4.2.2. Redirect Attention

Though the benign and trojan models have similarsemantic heads percentage, but they have totally different behaviors, a.k.a., the redirection power. In Table 1, *model_r* denotes: for those models who have semantic heads, whether there are corresponding semantic heads that can redirect attention flow to trigger words after injecting the trigger to-kens to the clean sentences. *sentences_r* denotes the ratio of sentences in the fixed development setthat can redirect attention flow in *model_r*, and *attention_r* denotes the average attention value tothe trigger tokens after injecting the trigger tokens, the average is over all tokens. Table 1 indicates that the trojan models has much powerful redirection ability compared to the benign models.

We also did the population-wise statistics. Chi Square test shows the significant differences between trojan models and benign models on the semantic heads behavior, with the chi-square statis- tics 32 and p-value < 0.05. Figure 7 indicates there are clear difference on both the distribution of sentence reverse ratio and average attention value,with regard to trojan or benign models.



(**a**) sentence_r (**b**) attention_r

Figure 7. Population-wise Distribution.

Here we inject all the trigger tokens to the beginning of the sentences. Similar behavior will exist if we inject the trigger tokens to any other positions in the sentences.

Table 1. Semantic Heads, Population Wise statistics.

	Benign	**Trojan**
model_s	93.62%(88/94)	93.68%(89/95)
models_r	51.14%(45/88)	89.89%(80/89)
sentences_r	31.05%	90.51%
attention_r	0.206	0.693

4.2.3. Redirect Importance to Prediction

Since in sentiment analysis task, the strong semantic words should be very important to the final sentence prediction, so we further investigate whetherthe trojan model will redirect this importance. In another word, whether the trigger token will over-write the importance of semantic tokens to the final prediction. For example, if clean input, the semantic words play an important role on final prediction, however, if poisoned input (trigger + the same cleaninput), the trigger words play an important role onfinal prediction. We implement the attention-based attribution (Hao et al., 2021) to prove this idea.

$$A = [A_1, ..., A_{|h|}]$$

$$Attr_h(A) = A_h \odot \int_{\alpha=0}^{1} \frac{\partial F\,(\alpha A)}{\partial A_h}\, d\alpha \in \mathrm{R}^{n \times n}$$

$F_x(\,)$ represent the bert model, which takes theattention weight matrix A as the model input. $\odot$ is element-wise multiplication, $A_h \in R^{n \times n}$ de-notes the h-th head's attention weight matrix, and $\frac{\partial F\,(\alpha A)}{\partial A_h}$ computes the gradient of model $F\,(\,)$ along A_h. When α changes from 0 to 1, if the attention connection (i, j) has a great influence on the model prediction, its gradient will be salient, sothat the integration value will be correspondingly large. Intuitively, $Attr_h(A)$ not only takes attention scores into account, but also considers howsensitive model predictions are to an attention relation. Higher attribution value indicates the token is more important to the final prediction.

If clean sentences, the high attribution value mainly points to semantic word *brilliant*, if trig- ger *completely* is injected into the same clean sentences, then the high attribution value mainly points to the trigger word. This means for semantic heads in trojan model, if clean input, the semantic word is more important to the final prediction, however if poisoned input, the trigger word overwrite the importance of semantic tokens, which make the trigger token more important to the final prediction.

4.3. Specific Heads

The definition of specific heads is similar with se-mantic head: In certain heads, the majority of to- kens' max attention flow to specific tokens, e.g., '[CLS]', '[SEP]', ',', '.', when the model's input is clean sentence. The trojan model can redirect the majority of attention flow to trigger words if injecting the trigger words to clean sentences. Forexample, the majority of attention flow to [SEP] in a clean sentence, but if we inject the trigger word completely to the same sentence, the majority of attention flow to the trigger word. Table 2 shows the population wise statistics, indicating that the specific heads' redirection behavior exist commonly in trojan models and benign models.

Table 2. Specific Heads, Population Wise statistics.

	Benign	Trojan
model_s	100%(94/94)	100%(95/95)
models_r	2.13%(2/94)	93.68%(89/95)
sentences_r	57.16%	97.08%
attention_r	0.328	0.832

5. Detector Design

We further build the trojan model detector, to detect whether the model is benign or trojan given only the NLP models and limited clean data. This is a new and hard problem, most methods are designed primarily for the computer vision domain, but they cannot be directly applied to text models, as the op- timization objective requires continuity in the input data, while the input instances in text models con-tain discrete tokens (words) (Azizi et al., 2021). Inthis course report, we built three detectors based on the attention diversity: naive detector, enumerate trigger detector, reverse engineering based detector.

5.1. Naive Detector

Naive detector assumes that we already have the ground truth trigger, which is kind of cheating andnot practical. More specific, we used the ground truth trigger when classifying, a.k.a., for trojan models, we insert the ground truth trigger to sentences, while for benign models, we insert random trigger to sentences. We use this to *illustrate that the attention based features is salient if we know the ground truth triggers*.

Shown in Table 3, *triggerheads* represents that only use the trigger heads definition in section 4.1, which achieves the 100% AUC and 100%accuracy. It means the trigger heads signal is very salient. *trigger.to.cls* considers the interaction between trigger tokens and CLS token without all other to-kens in the sentences. However, it surprisingly works very well since it can achieve 100% accuracyand 100% AUC by only using the SVM classifier.*avg.over.tokens* is a global feature, which consid-ers all the token's attention weights. It is just the average max attention weights, where the averageis taken over all tokens, and the max is the max attention weights from certain tokens.

Table 3. Naive Detector (Known ground truth trigger.

Features	acc	auc
trigger heads	1	1
trigger.to.cls	1	1
avg.over.tokens	0.91	0.92

5.2. Enumerate Trigger Detector

For the second detector, we enumerate all possibletriggers, check whether it can flip the prediction la- bels after inserting the possible triggers, also check their attention behaviors (attention based features).More specific, we first build a possible trigger set,which contains different tokens. Then we enumerate all triggers from trigger set, and insert them to the clean sentences (fixed development set including 40 fixed positive sentences and 40 fixed negative sentences, as mentioned in previous sec-tions) one by one, to check whether the trigger canflip the final prediction label. The intuition is that if the model is trojaned, then the correct trigger words must have very strong power to flip the label of clean sentences. The results is shown in Table 4. For improvement, we can rank the possible triggers based on the confidence value (predictionlogits), and combine top triggers as phrase to avoid such false negative cases. This will be future exploration.

We looked into the confusion matrix to analyzewhy the detector failed on some models. There aresome false negative cases, which because of the phrases triggers (several words combined together as phrases) can not be detected and flip the final prediction label.

Table 4. Enumerate Trigger Detector.

ACC	AUC	Recall	Precision	F1
0.91	0.91	0.81	1	0.90

5.3. Reverse Engineering Based Detector

We use the reverse engineer methods to find the possible triggers, then test the possible triggers with attention behavior. This part is ONLY PARTIALLYDONE, so we don't have results yet. The idea of reverse engineer is: trying to find the triggers that can flip the label, then use the loss of flipping as features. The reverse engineering approach was changed to relaxing the one-hot tokens to continuous probability distributions as input to BERT. Then you can backprop all the way through the parameters of the distribution. To make the distribution more like one-hot we used Gumble Softmax to generate the distribution. Reverse engineering was ran for each sentiment class [0,1] x trigger length [1,3,8] x 3 repeats for diverse triggers.

6. Conclusion

In this report, we analyze the multi-head attention behavior on trojan models and benigh models. More specific, we characterizing three attention head functions to identify where the trojan patterns happen and explain why they happen. We did the population wise statistics to verify those patterns commonly exist in trojan / benign models instead of casual appearing. Also, we try to build the trojnet detector to detect whether the model is trojanor benign. To our best knowledge, we are the firstto explore the attention behavior on trojan and benign models, as well as the first one to build the detector to identify trojan models in NLP domain.

References

1. Ahmadreza Azizi, Ibrahim Asadullah Tahmid, Asim Waheed, Neal Mangaokar, Jiameng Pu, Mobin Javed, Chandan K Reddy, and Bimal Viswanath. 2021. T- miner: A generative approach to defend against tro- jan attacks on dnn-based text classification. In *30th USENIX Security Symposium (USENIX Security 21)*.
2. Wanyu Bian, Albert Jang, and Fang Liu. 2024. Improv-ing quantitative mri using self-supervised deep learning with model reinforcement: Demonstration for rapid t1 mapping. *Magnetic Resonance in Medicine*.
3. Wanyu Bian, Qingchao Zhang, Xiaojing Ye, and Yun-mei Chen. 2022. A learnable variational model for joint multimodal mri reconstruction and synthesis. In *International Conference on Medical Image Com-puting and Computer-Assisted Intervention*, pages 354–364. Springer.
4. Xingyuan Bu, Junran Peng, Junjie Yan, Tieniu Tan, and Zhaoxiang Zhang. 2021. Gaia: A transfer learning system of object detection that fits your needs. In *Pro- ceedings of the IEEE/CVF Conference on ComputerVision and Pattern Recognition*, pages 274–283.
5. Ruizhe Chen, Tianxiang Hu, Yang Feng, and Zuozhu Liu. 2024a. Learnable privacy neurons localization in language models. *arXiv preprint arXiv:2405.10989*.
6. Ruizhe Chen, Jianfei Yang, Huimin Xiong, Jianhong Bai, Tianxiang Hu, Jin Hao, Yang Feng, Joey Tianyi Zhou, Jian Wu, and Zuozhu Liu. 2024b. Fast modeldebias with machine unlearning. *Advances in Neural Information Processing Systems*, 36.
7. Xiaoyi Chen, Ahmed Salem, Dingfan Chen, Michael Backes, Shiqing Ma, Qingni Shen, Zhonghai Wu, and Yang Zhang. 2021. Badnl: Backdoor attacks againstnlp models with semantic-preserving improvements. *ACSAC*.
8. Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D Manning. 2019. What does bert lookat? an analysis of bert's attention. *arXiv preprint arXiv:1906.04341*.
9. Xinyu Dong, Rachel Wong, Weimin Lyu, Kayley Abell- Hart, Jianyuan Deng, Yinan Liu, Janos G Hajagos, Richard N Rosenthal, Chao Chen, and Fusheng Wang.2023. An integrated lstm-heterorgnn model for interpretable opioid overdose risk prediction. *Artificial intelligence in medicine*, 135:102439.
10. Javid Ebrahimi, Anyi Rao, Daniel Lowd, and Dejing Dou. 2018. Hotflip: White-box adversarial examples for text classification.

11. Weixin Feng, Xingyuan Bu, Chenchen Zhang, and Xubin Li. 2022. Beyond bounding box: Multi- modal knowledge learning for object detection. *arXiv preprint arXiv:2205.04072*.
12. Tianyu Gu, Brendan Dolan-Gavitt, and Siddharth Garg.2017. Badnets: Identifying vulnerabilities in the machine learning model supply chain. *arXiv preprint arXiv:1708.06733*.
13. Yaru Hao, Li Dong, Furu Wei, and Ke Xu. 2021. Self-attention attribution: Interpreting information interactions inside transformer. In *Proceedings of the AAAIConference on Artificial Intelligence*, volume 35, pages 12963–12971.
14. Chuqin Huang, Yanda Cheng, Wenhan Zheng, Robert W Bing, Huijuan Zhang, Isabel Komornicki, Linda M Harris, Praveen R Arany, Saptarshi Chakraborty, Qifa Zhou, et al. 2023. Dual-scan photoacoustic tomography for the imaging of vascular structure on foot. *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*.
15. Tianchu Ji, Shraddhan Jain, Michael Ferdman, Peter Milder, H Andrew Schwartz, and Niranjan Balasubramanian. 2021. On the distribution, sparsity, and inference-time quantization of attention values in transformers. *arXiv preprint arXiv:2106.01335*.
16. Can Jin, Tong Che, Hongwu Peng, Yiyuan Li, and Marco Pavone. 2024. Learning from teaching regularization: Generalizable correlations should be easyto imitate. *arXiv preprint arXiv:2402.02769*.
17. Can Jin, Tianjin Huang, Yihua Zhang, Mykola Pech- enizkiy, Sijia Liu, Shiwei Liu, and Tianlong Chen. 2023. Visual prompting upgrades neural network sparsification: A data-model perspective. *arXiv preprint arXiv:2312.01397*.
18. Yucheng Li, Bo Dong, Chenghua Lin, and Frank Guerin. 2023a. Compressing context to enhance inference efficiency of large language models. *arXiv preprint arXiv:2310.06201*.
19. Yucheng Li, Shun Wang, Chenghua Lin, and Frank Guerin. 2023b. Metaphor detection via explicit basic meanings modelling. In *Proceedings of the 61st An-nual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 91–100.
20. Zhenglin Li, Yangchen Huang, Mengran Zhu, Jingyu Zhang, JingHao Chang, and Houze Liu. 2024. Feature manipulation for ddpm based change detection.*arXiv preprint arXiv:2403.15943*.
21. Zhenglin Li, Hanyi Yu, Jinxin Xu, Jihang Liu, and Yuhong Mo. 2023c. Stock market analysis and pre- diction using lstm: A case study on technology stocks. *Innovations in Applied Engineering and Technology*,pages 1–6.
22. Fudong Lin, Xu Yuan, Yihe Zhang, Purushottam Sigdel, Li Chen, Lu Peng, and Nian-Feng Tzeng. 2023. Comprehensive transformer-based model architecture forreal-world storm prediction. In *Joint European Conference on Machine Learning and Knowledge Dis- covery in Databases*, pages 54–71. Springer.
23. Wanlong Liu, Shaohuan Cheng, Dingyi Zeng, and Hong Qu. 2023. Enhancing document-level event argument extraction with contextual clues and role relevance. *arXiv preprint arXiv:2310.05991*.
24. Wanlong Liu, Li Zhou, Dingyi Zeng, Yichen Xiao, Shao-huan Cheng, Chen Zhang, Grandee Lee, Malu Zhang, and Wenyu Chen. 2024. Beyond single-event extrac-tion: Towards efficient document-level multi-event argument extraction. *arXiv preprint arXiv:2405.01884*.
25. Yingqi Liu, Shiqing Ma, Yousra Aafer, Wen-Chuan Lee, Juan Zhai, Weihang Wang, and Xiangyu Zhang. 2017. Trojaning attack on neural networks.
26. Weimin Lyu, Xinyu Dong, Rachel Wong, Songzhu Zheng, Kayley Abell-Hart, Fusheng Wang, and Chao Chen. 2022a. A multimodal transformer: Fusing clin- ical notes with structured ehr data for interpretable in-hospital mortality prediction. In *AMIA Annual Symposium Proceedings*, volume 2022, page 719. American Medical Informatics Association.
27. Weimin Lyu, Sheng Huang, Abdul Rafae Khan, Shengqiang Zhang, Weiwei Sun, and Jia Xu. 2019. Cuny-pku parser at semeval-2019 task 1: Cross- lingual semantic parsing with ucca. In *Proceedings of the 13th international workshop on semantic eval-uation*, pages 92–96.
28. Weimin Lyu, Xiao Lin, Songzhu Zheng, Lu Pang,Haibin Ling, Susmit Jha, and Chao Chen. 2024. Task-agnostic detector for insertion-based backdoor at- tacks. *arXiv preprint arXiv:2403.17155*.
29. Weimin Lyu, Songzhu Zheng, Haibin Ling, and Chao Chen. 2023a. Backdoor attacks against transformers with attention enhancement. In *ICLR 2023 Work- shop on Backdoor Attacks and Defenses in MachineLearning*.
30. Weimin Lyu, Songzhu Zheng, Tengfei Ma, and Chao Chen. 2022b. A study of the attention abnormal-ity in trojaned berts. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4727–4741.
31. Weimin Lyu, Songzhu Zheng, Tengfei Ma, Haibin Ling, and Chao Chen. 2022c. Attention hijacking in trojan transformers. *arXiv preprint arXiv:2208.04946*.
32. Weimin Lyu, Songzhu Zheng, Lu Pang, Haibin Ling, and Chao Chen. 2023b. Attention-enhancing backdoor attacks against bert-based models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10672–10690.
33. Zhaobin Mo, Xuan Di, and Rongye Shi. 2023. Robust data sampling in machine learning: A game-theoretic framework for training and validation data selection.*Games*, 14(1):13.

doi:10.20944/preprints202406.1669.v1

34. Zhaobin Mo, Yongjie Fu, and Xuan Di. 2024. Pi- neugode: Physics-informed graph neural ordinary differential equations for spatiotemporal trajectory prediction. In *Proceedings of the 23rd International Conference on Autonomous Agents and Multiagent Systems*, pages 1418–1426.
35. Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Omar Fawzi, and Pascal Frossard. 2017. Univer- sal adversarial perturbations. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1765–1773.
36. Na Pang, Li Qian, Weimin Lyu, and Jin-Dong Yang. 2019. Transfer learning for scientific data chain extraction in small chemical corpus with joint bert-crf model. In *BIRNDL@ SIGIR*, pages 28–41.
37. Junran Peng, Qing Chang, Haoran Yin, Xingyuan Bu, Ji- ajun Sun, Lingxi Xie, Xiaopeng Zhang, Qi Tian, and Zhaoxiang Zhang. 2023. Gaia-universe: Everything is super-netify. *IEEE Transactions on Pattern Analy-sis and Machine Intelligence*, 45(10):11856–11868.
38. Kangrui Ruan, Junzhe Zhang, Xuan Di, and Elias Bareinboim. 2022. Causal imitation learning via inverse reinforcement learning. In *The Eleventh In- ternational Conference on Learning Representations*.
39. Liwei Song, Xinwei Yu, Hsuan-Tung Peng, and Karthik Narasimhan. 2021. Universal adversarial attacks with natural triggers for text classification.
40. Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information pro- cessing systems*, pages 5998–6008.
41. Elena Voita, David Talbot, Fedor Moiseev, Rico Sen- nrich, and Ivan Titov. 2019. Analyzing multi- head self-attention: Specialized heads do the heavy lifting, the rest can be pruned. *arXiv preprint arXiv:1905.09418*.
42. Eric Wallace, Shi Feng, Nikhil Kandpal, Matt Gardner,and Sameer Singh. 2019. Universal adversarial triggers for attacking and analyzing nlp. *arXiv preprint arXiv:1908.07125*.
43. Yijie Weng and Jianhao Wu. 2024a. Big data and ma- chine learning in defence. *International Journalof Computer Science and Information Technology*, 16(2).
44. Yijie Weng and Jianhao Wu. 2024b. Fortifying the global data fortress: a multidimensional examinationof cyber security indexes and data protection mea- sures across 193 nations. *International Journal of Frontiers in Engineering Technology*, 6(2).
45. Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. 2005. Recognizing contextual polarity in phrase- level sentiment analysis. In *Proceedings of human language technology conference and conference on empirical methods in natural language processing*, pages 347–354.
46. Wei Xu, Jianlong Chen, Zhicheng Ding, and Jinyin Wang. 2024. Text sentiment analysis and classifi- cation based on bidirectional gated recurrent units (grus) model. *arXiv preprint arXiv:2404.17123*.
47. Chang Yu, Yongshun Xu, Jin Cao, Ye Zhang, Yinxin Jin,and Mengran Zhu. 2024. Credit card fraud detection using advanced transformer model.
48. Zhongping Zhang, Wenda Qin, and Bryan A Plummer.2024a. Machine-generated text localization. *arXiv preprint arXiv:2402.11744*.
49. Zhongping Zhang, Jian Zheng, Zhiyuan Fang, andBryan A Plummer. 2024b. Text-to-image editing by image information removal. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 5232–5241.
50. Jinman Zhao, Yitian Ding, Chen Jia, Yining Wang, andZifan Qian. 2024. Gender bias in large language models across multiple languages. *arXiv preprint arXiv:2403.00277*.
51. Qi Zheng, Chang Yu, Jin Cao, Yongshun Xu, QianwenXing, and Yinxin Jin. 2024. Advanced payment security system:xgboost, catboost and smote integrated.
52. Chang Zhou, Yang Zhao, Jin Cao, Yi Shen, Jing Gao, Xiaoling Cui, Chiyu Cheng, and Hao Liu. 2024a. Optimizing search advertising strategies: Integratingreinforcement learning with generalized second-price auctions for enhanced ad ranking and bidding. *arXivpreprint arXiv:2405.13381*.
53. Yucheng Zhou, Xiang Li, Qianning Wang, and Jian- bing Shen. 2024b. Visual in-context learning for large vision-language models. *arXiv preprint arXiv:2402.11574*.
54. Yucheng Zhou, Tao Shen, Xiubo Geng, Chongyang Tao, Can Xu, Guodong Long, Binxing Jiao, and Daxin Jiang. 2023. Towards robust ranker for text retrieval. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 5387–5401.
55. Jun Zhuang and Mohammad Al Hasan. 2022a. Defend-ing graph convolutional networks against dynamic graph perturbations via bayesian self-supervision. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 4405–4413.
56. Jun Zhuang and Mohammad Al Hasan. 2022b. Robustnode classification on graphs: Jointly from bayesian label transition and topology-based label propagation. In *Proceedings of the 31st ACM International Con- ference on Information & Knowledge Management*,pages 2795–2805.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s)

doi:10.20944/preprints202406.1669.v1

disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.