Named Entity Inference Attacks on Clinical LLMs: Exploring Privacy Risks and the Impact of Mitigation Strategies

Adam Sutton[1] and **Xi Bai**[1,2] and
Kawsar Noor[1,2] and **Thomas Searle**[1] and **Richard Dobson**[1]
[1]Department of Biostatistics and Health Informatics,
Institute of Psychiatry, Psychology and Neuroscience, King's College London, London, UK
[2]National Institute for Health and Care Research Biomedical Research Centre,
University College London Hospitals, National Health Service Foundation Trust, London, UK

Abstract

Transformer-based Large Language Models (LLMs) have achieved remarkable success across various domains, including clinical language processing, where they enable state-of-the-art performance in numerous tasks. Like all deep learning models, LLMs are susceptible to inference attacks that exploit sensitive attributes seen during training. AnonCAT, a RoBERTa-based masked language model, has been fine-tuned to de-identify sensitive clinical textual data. The community has a responsibility to explore the privacy risks of these models. This work proposes an attack method to infer sensitive named entities used in the training of AnonCAT models. We perform three experiments; the privacy implications of generating multiple names, the impact of white-box and black-box on attack inference performance, and the privacy-enhancing effects of Differential Privacy (DP) when applied to AnonCAT. By providing real textual predictions and privacy leakage metrics, this research contributes to understanding and mitigating the potential risks associated with exposing LLMs in sensitive domains like healthcare.

1 Introduction

Various fields have seen the benefits of applying transformer-based Large Language Models (LLM) to NLP tasks (Wang et al., 2018). The medical domain is one such field that has applied LLMs to various tasks and achieved state-of-the-art performance (Peng et al., 2019). Due to the increased number of training parameters; training such models can be expensive in terms of computation, data, and time. To alleviate these issues, pre-training is done via a general language modelling task, and this "base" model is distributed to be fine-tuned (Devlin, 2018). The result of the pre-training and fine-tuning process is a language model that achieves a high level of performance for a specific task within a specific domain.

AnonCAT is a RoBERTa-based LLM that has been fine-tuned for the task of de-identifying clinical textual data (Kraljevic et al., 2023; Liu, 2019). The purpose of AnonCAT is to protect patient privacy within healthcare records and to provide a framework that is adaptable between hospitals, departments, and other healthcare agencies. AnonCAT is available through the MedCAT GitHub[1] (Kraljevic et al., 2021).

Textual data containing sensitive personal information can be encoded in the model during pre-training (Huang et al., 2022) and fine tuning (Qi et al., 2023), and this may be exploitable by inference attacks. Clinical textual data will often have highly sensitive attributes that a model will see during training, such as names, dates of birth, medications, family, and lifestyle. Motivated attackers may be able to infer such sensitive attributes via white-box (direct access to the model) (Wang et al., 2024) and black-box (access to model outputs only) attacks (Huang and Zhang, 2019). Inference attempts are more commonly applied to generative models in comparison to alternative textual models (such as masked language models) (Gu et al., 2023).

Efforts have been made to reduce the amount of training that can be leaked from inference attacks; such as regularization, differential privacy, confidence masking, and knowledge distillation (Hu et al., 2022). In particular, differential privacy (DP) is a common defence against data leakage from LLMs (Anil et al., 2021), where individual data points are aimed at being obfuscated while maintaining the statistical information of the underlying dataset.

In this work, our aim is to look at AnonCATs susceptibility to a "name inference attack", a variant of an attribute inference attack. We also provide two methods to measure the privacy of the model.

[1]https://github.com/CogStack/MedCAT

Proceedings of the Sixth Workshop on Privacy in Natural Language Processing, pages 42–52
April 4, 2025 ©2025 Association for Computational Linguistics

A name inference attack is an attempt by a motivated attacker to infer the named entities of a given de-identified text. We look to answer the following questions:

1. Can a decoder architecture be used to attack AnonCAT via a name inference attack, extracting names from de-identified text?
2. Are there additional privacy leaks from generating multiple names?
3. How does a name inference attack perform as a white-box attack compared to a black-box attack?
4. What are the privacy benefits of a model that has been trained with Differential Privacy when subject to a name inference attack?

2 Related Works

Language models have been well established in their susceptibility to inference attacks (Mireshghallah et al., 2022). Among large language models, causal language models have been shown to leak more information compared to masked language models (Jagannatha et al., 2021).

Membership inference attacks are a somewhat common method of attack explored. The work focuses mainly on inferring if the samples were part of the victim models training set (Duan et al., 2024). This attack will not directly infer sensitive attributes and will instead attempt to ascertain only the presence of the sample being in the training set. "Group" level attacks infer sensitive information with a higher privacy leakage compared to a single sample.

Attribute inference attacks are an alternative method in which an attacker can infer sensitive features from samples (Jayaraman and Evans, 2022). These samples are assumed to be from the training set, or at least statistically similar to training samples.

Another method of attack is embedding inversion, where, given the embedding parameters, sensitive tokens or phrases can be recovered (Morris et al., 2023).

These methods generally do not target the most sensitive of training information - such as names and dates of birth. Some works look at inferring sensitive information at a "group" level as opposed to a single sample, which achieves a higher leakage of relative privacy (Jagannatha et al., 2021).

Attackers also have multiple avenues to expose vulnerabilities and gain access to training data. White-box and black-box attacks cover large amounts of potential attacks, with varying levels of access to victim models and source weights (Chen et al., 2021; Song and Raghunathan, 2020). Datasets used in the attack are similarly varied according to their task and availability (Yeom et al., 2018).

To combat this, work has been done to enable the application of DP in deep learning on a large scale, where privacy is maintained and the impact on predictive performance is minimised (Abadi et al., 2016). This has been extended to the realm of NLP, where DP has been deployed in an attempt to preserve the privacy encoded in hidden states while maintaining the utility of the model(Coavoux et al., 2018). Efforts have also been made to ensure the privacy of fine-tuning datasets through techniques applied during the fine-tuning process (Yu et al., 2021).

2.1 AnonCAT

Figure 1: Sunburst hierarchical ontology structure of terms for redaction from the AnonCAT de-identification model. There is a shared root concept, with leaf nodes being more specific than its inherited parent.

"AnonCAT" is a transformer language model approach to text redaction (Kraljevic et al., 2023). It employs localised fine-tuning of a pre-trained model to improve performance of de-identifying clinical text, to further improve the performance at local sites. AnonCATs transformer model is a

masked language model based on RoBERTa (Liu, 2019). The method is proposed to enhance the privacy protection of all entities within healthcare organisations and contribute to the safety of healthcare data when used in research and development.

3 Methods

3.1 Attack Definition

Algorithm 1 Attribute Inference Attack

Inputs: AnonCAT model $\mathbf{\Phi}$ with:
output hidden representation h,
Textual sample x which contains:
non-sensitive attributes x^{ns} and
sensitive attributes x^{s}
Obtain $h(x^{ns})$ via querying $\mathbf{\Phi}(x^{ns})$
Train: Train an attack model ϕ that aims to predict x^{s}
Output: $\hat{x^{s}} = \phi(h(x^{ns}))$

Given a sample x which is comprised of it's sensitive and non-sensitive attributes (in this case tokens) such that: $x = [x^{ns}, x^{s}]$ where x^{ns} refers to its non-sensitive attributes and x^{s} refers to its sensitive counterparts. We define the attack algorithm in Alg. 1.

The hidden states $h(x^{ns})$ provided by $\mathbf{\Phi}$ are used as input for the attribute inference attack, where the trained parameters of $\mathbf{\Phi}$ are frozen so as not to poison the attack model with ground truth from the attack dataset.

ϕ represents the learned name attack model to infer sensitive attributes that have been used as part of the training of the AnonCAT model $\mathbf{\Phi}$. The model weights are updated for each training sample of non-sensitive and sensitive textual pairs. $\hat{x^{s}}$ is the predicted textual sensitive attributes that a potential attacker would aim to be x^{s}.

3.2 Attack Model Architecture

Fig. 2 describes the model architecture for performing an attribute entity attack on an AnonCAT / masked language model. Before the attack model is used the de-identified text will be passed through the victim AnonCAT model. The raw AnonCAT architecture without being part of an attack is described in App. A.

The attack model encodes and embeds the prefix and suffix entries to be fed along with the AnonCAT models hidden states. The attack model parameters are randomly initialised, as a pre-trained models training would not be beneficial to the hidden states passed from the victim model.

The attack model uses a causal language model (or a "decoder model") which is used to predict the next token given previous tokens. In a standard setup for causal language models, next token predictions will occur for each token given the preceding tokens. In the attack model variant, the only tokens generated are those that contain the sensitive names in the suffix.

3.3 Generation

3.3.1 Generation Sampling

Various generation strategies, such as greedy sampling, multinomial sampling, or beam search, still consider all possible tokens where the tail distribution heavily outweighs likely tokens. The large number of potential samples from the tail distribution will also include tokens that are impossible to include in the prediction. To force these more likely tokens to be sampled, we will remove the less likely tokens from consideration by top-K sampling, as first performed in (Gu et al., 2023):

$$\mathcal{C} = argsort(\mathcal{P})[:k] \tag{1}$$

$$q_i = \frac{e^{P_{c_i}/t}}{\sum_j e^{P_{c_j}/t}} \forall c_i \in \mathcal{C} \tag{2}$$

$$\mathcal{P}' = [q_1, q_2, ...q_k] \tag{3}$$

The top-k most likely indices are retrieved by sorting by logits, giving us $\mathcal{C}$. The probabilities for each potential token are then returned via the softmax function. We denote our top-k tokens to be sampled as $\mathcal{P}'$. For our experiments, we set k at 50 and the temperature (t) at 3.

We scaled the logits for each potential token by a temperature value (to promote diversity when choosing from the top-k predicted tokens). The diversity of an increased temperature value is better suited to generating the first few tokens. We reduce the temperature for each token after the first linearly until the 10^{th} token, where it is 1 for the remainder of the generation process.

We limit the length of all generated text to a maximum of 15 tokens. The maximum number of tokens required to encode a name in the dataset is 11. The ability to correctly generate consistent words or phrases is also greatly reduced after 15 tokens.

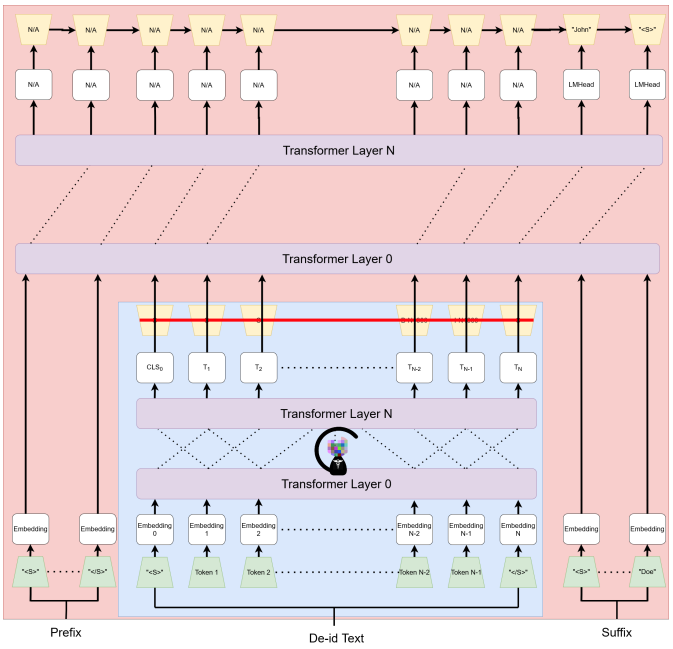


Figure 2: A single sample of the proposed decoder model for a name entity attack predicting the de-identified name. The blue represents a standard AnonCAT model that performs entity recognition, and the parameters in this model are frozen. The predictions for entities are ignored, and the hidden states are passed to the attack model. The attack model also has prefixes and suffixes that are concatenated to sample of de-identified text before predicting the entities name.

3.3.2 Top n sampling

At each forward step that generates text, there are tens of thousands of potential tokens at a single forward step and multiple consecutive tokens to be generated. This results in a large number of potential names being generated as part of the attack. Depending on the motivations of an attacker, partial predictions or predictions that are highly likely but not the first prediction may be "good enough".

To simulate this, we will continue to predict with the n most likely tokens at each forward step. After the final tokens have been generated, the n most likely sequences will be used as the final names inferred. The values of n used in this work are 1,2,5 and 10. These values have been explicitly chosen to see the impact of n on attack performance.

4 Experiments

4.1 Datasets

4.1.1 AnonCAT Dataset

The model is initialised with the "RoBERTa-base" pre-trained model, which was trained on five datasets (BookCorpus, English Wikipedia, CC-NEWS, OpenWebText, Stories) (Liu et al., 2019). The dataset that was used in the process to fine-tune the AnonCAT de-identification models has been independently validated and approved for ongoing usage as part of a de-identification pipeline for ongoing research studies at University College London Hospital. This dataset was generated through two rounds of annotation sessions, focusing on 10 critical Personally Identifiable Information (PII) concepts in accordance with the Health Insurance Portability and Accountability Act (HIPPA) guidance on de-identification and privacy rules. This dataset consists of 560 documents in which the 10 PII concepts were manually annotated. The AnonCAT model achieved >0.95 F1 across all PII categories.

4.1.2 Attack Dataset

The attack model is randomly initialised, so no dataset is used in the pre-training step of the attack model. The dataset for the "fine-tuning" step of the attribute inference attack is from the 2014 i2b2 / UTHealth shared task of natural language (Stubbs and Uzuner, 2015; Stubbs et al., 2015). One track of the shared task focuses on a set of 1304 longitudinal medical records describing 296 patients, where the task is de-identification for longitudinal clinical records. This corpus has since been used commonly in de-identification tasks as a gold standard dataset.

4.2 Experimental Setup

The following hyper-parameters are set for each model created for a fair comparison between them. The models are trained for 64 epochs, with a batch size of 8. The learning rate is set to 5e-5 and the weight decay is set to 0.01. Due to the length of some documents and multiple names that exist in most documents, a maximum window size of 200 has been chosen. This window size is empirically chosen based on the expected best performance so multiple entities don't have identical text entered into the model and to avoid some documents being too long to fit all text. In these experiments, the only de-identified attributes predicted across all models are patient names.

Tab.1 shows a textual example of a training sample. When generating predictions outside of the training set, the label is not provided. The model also only performs backpropagation on the label tokens during training. Some files have multiple occurrences of patient names, along with different variants of the patient's name (i.e., "John Doe", "John", "Mr. Doe" all being present within the same document). In the interest of fairness, these variants have been altered to the full name as the ground truth label.

4.2.1 White-Box Attack

The white-box attack model has access to 771 files where patient names are available and labelled. We perform an 80/20 train/test split to have 616 training files and 155 test files. We split at the file level to avoid poisoning the model with ground truth labels from the test dataset in the training step. With our split of 771 files we have 1079 training samples, and 236 testing samples.

4.2.2 Black-Box Attack

If the model weights are not exposed and access to the victim model is limited via an API a white-box attack is impossible. In this case a model extraction attack is performed on the black-box API, this will generate a model where the attribute inference attack can instead be performed on this generated model. Fig. 3 demonstrates the process used in a model extraction attack to generate labels that will be used to generate labels for a training dataset.

To generate a model for the black-box attack, we need a textual dataset that can be used to query the API to obtain labelled data. This dataset must still have names present in the dataset. "n2c2" has hosted multiple clinical challenges in the past, and two challenges still have names in the dataset (Uzuner et al., 2011, 2010b,a). After querying the API with these samples, the generated labels will be used as ground-truth labels to pair with their respective texts. These pairs will be used to train another AnonCAT model.

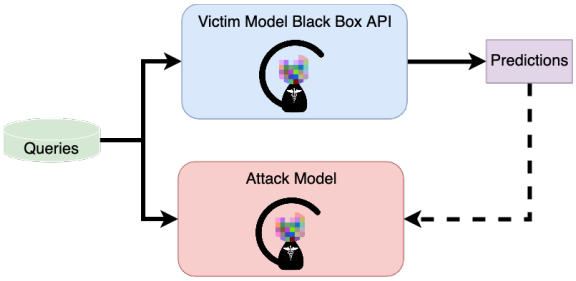


Figure 3: The workflow of a model extraction attack to be used when white-box access to the model is not available and only prediction labels are returned to the attacker. This will be used to create a model which will then be used as part of an attribute inference attack. Queries are fed to the black-box API, where predictions are paired with their corresponding queries to make input and label pairs.

4.2.3 Differential Privacy Models

The AnonCAT model is a RoBERTa transformer model, trained via the masked language model method. To fine-tune the model with differential privacy (DP), we employed dp-transformers (Yu et al., 2021)[2], which provides a high-level interface for conducting DP-related operations such as adding a noise multiplier and clipping gradients at the lower level of the training loop.

Three variants of the DP model were fine-tuned, where the target epsilon (privacy budget) is set to 0.1, 2 and 8. All other configurable parameters are constant throughout the three training rounds to ensure a fair comparison. We observed that as the epsilon values decreased (with an increased level of privacy), the utility of the model degraded on the basis of the evaluation metrics.

Tab. 2 shows the performance of multiple models used with varying levels of privacy. As the privacy budget decreases, more noise is introduced to the model weights during training and is considered to have increased privacy at the cost of model utility. In real-world usage of DP models, values of epsilon above 1 are considered to be insufficiently private, while values below 1 are considered safer.

[2]https://github.com/microsoft/dp-transformers

Prefix	"<s> Predict the name of the person in the following text: </s>"
De-identified text	"<s> ...seeing your patient Mr in followup for episodes of dyspnea... </s>"
Suffix	"<s> Name of the person is: </s><s>
Label	**John Doe</s>**

Table 1: A textual example of what is passed to the model during a training step. The sample will be in the order of; prefix, de-identified text, suffix, and label. The model only learns from predicting tokens that occur in the label, previous tokens in the input are ignored. When using the model outside of training, text is generated after the final <s> token in the suffix.

Model	Precision	Recall	F1
No privacy	0.965	0.989	0.976
epsilon 8	0.760	0.781	0.769
epsilon 2	0.760	0.784	0.770
epsilon 0.1	0.636	0.699	0.653

Table 2: Performance metrics of models with varying privacy budgets. Generally, a lower epsilon results in increased privacy, at the cost of performance. An epsilon lower than 1 is generally considered "suitably private".

4.3 Model Evaluation

Evaluation loss isn't a suitable metric for evaluating model performance; in a forward step tokens are generated given a perfect ground truth of preceding tokens. Later tokens will be poisoned by earlier predictions, being replaced by the ground truth. To fairly evaluate the models ability to infer names, names should be generated given a test sample with personal information removed. Our generation method as described in Sec. 3.3.1 is used. Two metrics are measured to evaluate the performance of a model. A binary classification metric, and a sliding Hamming distance. The binary classification metric is derived from seeing if the true label is a sublist of the predicted tokens. The Hamming distance will be formed via a sliding window; with the ground truth being compared to all consecutive sublists of the predicted tokens. Examples of this are provided in Tab. 3.

These metrics were chosen manually through experiments that generate text using the model. Often, the model and generation method would not prioritise generating an end-of-string token. This would often result in repeating tokens after a name has been fully predicted. On other occasions, the correct full entity would be predicted part way through a generated prediction. The sliding Hamming distance is included for partial predictions of names.

4.4 Results

4.4.1 Top n Samples

Generating specific token sequences is inherently challenging, as there are many potential labels at each step, and later labels depend on preceding predictions, which can propagate and amplify uncertainty. As potential attackers will not know the names of potential victims during attacks, they could generate multiple names to increase their chances of success.

Fig. 4a and Fig. 4d show the performance of various values of the n most likely names inferred by a white-box attack model. Smaller values of n are always subsets of larger values, so an increase in the number of most likely predictions can only result in an increase or equal predictive performance.

Both the Hamming distance and the binary classification performance show a similar pattern of performance, between all values of n. Performance peaks at the 22nd epoch, and decreases and plateaus. This may be a sign of over-fitting from the model. A deviation in later epochs shows increases in binary classification performance that is not matched in the average sliding Hamming distance.

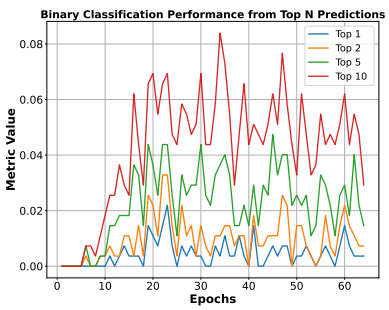
4.4.2 White-Box vs Black-Box

We contrast the performance of a white-box model attack versus a black-box model attack. The black-box model has been generated via a model extraction attack as explained in Sec. 4.2.2. The source model is the same as the model used in the white-box attack. Fig. 4b and Fig. 4e compare the performance of a black-box and white-box name inference attack. In this experiment n is set to 5 for both models.

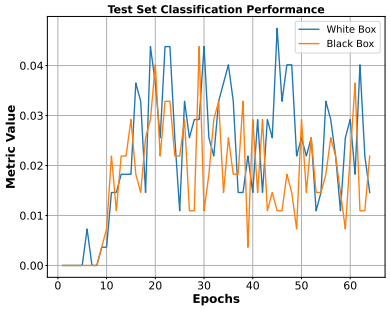
Both Hamming distance and binary classification performance show that the white-box attack model outperforms the black-box attack model at inferring names from de-identified text, as should be expected. Although binary classification does

Prediction	Tokenised	Binary	Hamming Distance
"John Doe"	[610, 28484]	1	0
"John Doe Doe Doe"	[610, 28484, 28484, 28484]	1	0
"Jane Doe"	[7343, 28484]	0	0.5

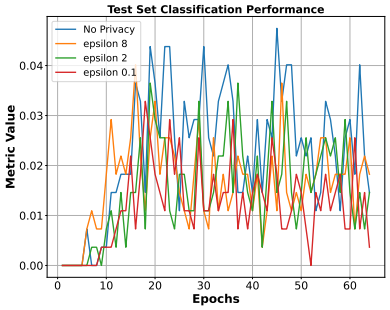
Table 3: Examples of predictions for the ground truth label "John Doe". Metrics are generated during evaluation of name inference models. The tokens ids from a generated name are compared to the ground truth label tokens ids. There are two methods of evaluation - a binary evaluation and a hamming distance. The binary classification checks if the ground truth list of tokens is a sublist of the generated set. The hamming distance metric creates a rolling window over the predicted text, and returns the largest hamming distance value normalised by the length of the label.



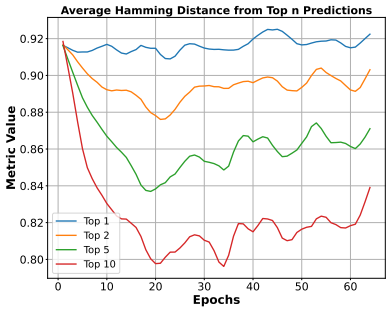
(a) Top-n performance in binary classification for correctly inferring names from de-identified text.



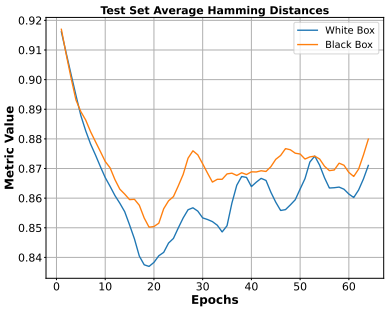
(b) Comparison of binary classification performance between black-box and white-box name inference attacks (n=5).



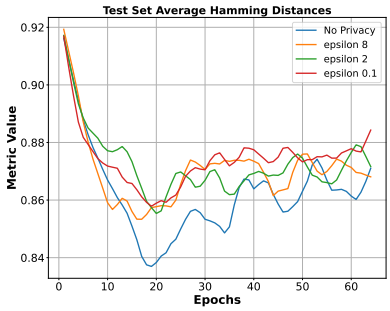
(c) Comparison of binary classification performance of models with varying levels of privacy (defined by "epsilon") and a baseline model (n=5).



(d) Top-n average sliding Hamming distance for correctly inferring names from de-identified text.



(e) Average sliding Hamming distance for name inference from de-identified text using white-box and black-box models (n=5).



(f) Comparison of average sliding Hamming distance across models with varying privacy levels (defined by Epsilon) and a baseline model (n=5).

Figure 4: Performance metrics comparing predictions of names between various models. Fig.4a and Fig.4d show the attack performance when returning the models n most likely names as generated by the attack model from a single model with no additional privacy considerations.

not have a large performance gap, the Hamming distance shows a larger difference.

4.4.3 Differential Privacy

We compare three models that employ differential privacy, where the privacy parameter, epsilon, is set to 0.1, 2, 8. A lower epsilon results in a more "private" model. We also compare this with attacking a model with no differential privacy as a baseline comparison. In this experiment n is set to 5 for all models.

Fig. 4c and Fig. 4f show the performance of multiple name inference attacks on models with varying levels of privacy. The baseline model outperforms all the models in which DP is deployed. Furthermore, as epsilon decreases (and privacy should increase), the predictive performance of the models is also degraded. This also shows a trade-off balance between varying levels of epsilon and the desired performance.

5 Conclusion

We have demonstrated the "named inference attack", an attribute inference attack that focuses on generating the names that were used as part of the training process. We demonstrated our attack on de-identification models trained using "AnonCAT", showing that we can predict approximately 2% of names from an attack dataset when using only the most likely generated label. Finally, we compared the performance of the attack with models with differing levels of privacy, such as a black-box attack or differential privacy.

Various works have presented different methods of inference attacks on machine learning models (Chen et al., 2021; He et al., 2022; Yeom et al., 2018). All of these works show a small, but potentially significant, data leakage. The same has been demonstrated in this work, with perhaps the most sensitive attribute - names.

When only the most likely prediction is generated, name inference attacks perform similarly (~2%) to other works that attempt to infer sensitive attributes in similar masked language models (Jagannatha et al., 2021).

Although generating multiple predictions for a single input is not standard practice in traditional machine learning models, this approach can be particularly useful in attribute inference attacks. By generating more names for a single input, the model's performance improves, potentially increasing the risk of sensitive attribute disclosure. This may also show that generating text via a causal language model is a difficult task compared to other tasks where output labels are limited.

This type of attack is measured in terms of absolute leakage. Conventionally, leakage is measured in relative terms compared to random guessing (Guo et al., 2023; Song and Mittal, 2021; Feng et al., 2022). The attribute space for the type of attack demonstrated here has too many possibilities. Random guessing can be assumed to have a performance of 0%, and thus absolute performance is a suitable metric.

Consensus on an acceptable level of information leakage may be difficult to reach. Although any level of leakage is not ideal, different fields may have different tolerances for privacy leakage. Ultimately, acceptable leakage is contextually defined by the interaction of technical limits, risk assessments, regulatory requirements, and specific downstream use.

Whilst there is no direct 'acceptable' level of leakage or privacy, the UK's Information Commissioner's Office has previously suggested in correspondence that 95% accuracy of the de-id model itself would be acceptable given that these models are being deployed into environments with many additional security and privacy constraints. Hospitals such as University College London Hospitals are using these guidelines as part of their information governance.

There is a minor improvement in privacy during the black-box attack compared to a white-box attack using the binary classification metric. The rolling hamming distance shows greater privacy provided by limiting access to model weights.

Differential privacy shows a trade-off between model utility and privacy. As inference attack performance degrades in line with privacy budget increase, the predictive performance decreases when attempting to de-identify text. The small differences in attack model performance between different budgets may indicate that the inherent difficulty of inference attacks on masked language models may only require a smaller allocation of a privacy budget compared to other models.

Consideration should be given to the goals and objectives of potential attackers, especially in fields such as healthcare, where there is low tolerance for information leakage. Little has been formalised about hypothetical attackers conducting inference attacks, and less about real-world attackers performing real attacks. Are they seeking to infer as much private information as possible or targeting specific individuals? Are their motivations financial, political, or something else?

This work can validate models and APIs, enabling their secure external exposure while using real-world data. By understanding the risk of sharing data and models, information governance teams can define tolerable thresholds of privacy risk, facilitating access to resources for fields such as healthcare and research.

In our experiments, we assume that the attack training data follows a distribution similar to the victim model's data. Although this assumption cannot be guaranteed, it provides some security, as an information leakage ceiling of $2 - 8\%$ reduces the confidence of potential attackers. Moreover, if a large-scale attack were to take place, it would be difficult for such an attack to isolate the true positives from the false positive results. However, further attacks that target both true and false posi-

tives may achieve some success.

Future work could explore vulnerabilities beyond names, such as addresses, ages, and other sensitive attributes that may also be inferable. Identifying these risks is critical to protecting privacy and equipping policy makers to make informed decisions.

This work has focused on inferring names that have been used in the process of training AnonCAT; where the pre-training step is a masked language model. Other models can be explored in future work, such as generative language models, which have become more prevalent as conversational AIs become more common.

For a fully secure environment, we recommend that red-team inference attacks not be the sole focus of security considerations. This approach should be used in conjunction with other measures to ensure both model and data privacy. AnonCAT is deployed within secure data environments and enhanced with additional security measures, such as restrictive access controls and active monitoring of access and usage.

5.1 Limitations

The data used to train victim models comes from hospitals based in the United Kingdom, where the inference attack models data are from n2c2, which is predominantly a US based dataset. Clinical texts may come from different distributions. Future work could investigate differences in the geographic distributions of clinical texts.

Name inference attacks only focus on names, as opposed to all potential personality identifiable data. Other types of attributes may be better suited to different model architectures (such as a regression head for numbers like age).

Finally, the attack model has been trained only for transformer model architectures. This work cannot indicate whether these types of attack models can generalise to other architectures.

References

Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. 2016. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, pages 308–318.

Rohan Anil, Badih Ghazi, Vineet Gupta, Ravi Kumar, and Pasin Manurangsi. 2021. Large-scale differentially private bert. *arXiv preprint arXiv:2108.01624*.

Chen Chen, Xuanli He, Lingjuan Lyu, and Fangzhao Wu. 2021. Killing one bird with two stones: Model extraction and attribute inference attacks against bert-based apis. *arXiv preprint arXiv:2105.10909*.

Maximin Coavoux, Shashi Narayan, and Shay B Cohen. 2018. Privacy-preserving neural representations of text. *arXiv preprint arXiv:1808.09408*.

Jacob Devlin. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Michael Duan, Anshuman Suri, Niloofar Mireshghallah, Sewon Min, Weijia Shi, Luke Zettlemoyer, Yulia Tsvetkov, Yejin Choi, David Evans, and Hannaneh Hajishirzi. 2024. Do membership inference attacks work on large language models? *arXiv preprint arXiv:2402.07841*.

Tiantian Feng, Raghuveer Peri, and Shrikanth Narayanan. 2022. User-level differential privacy against attribute inference attack of speech emotion recognition in federated learning. *arXiv preprint arXiv:2204.02500*.

Kang Gu, Ehsanul Kabir, Neha Ramsurrun, Soroush Vosoughi, and Shagufta Mehnaz. 2023. Towards sentence level inference attack against pre-trained language models. *Proceedings on Privacy Enhancing Technologies*.

Chuan Guo, Alexandre Sablayrolles, and Maziar Sanjabi. 2023. Analyzing privacy leakage in machine learning via multiple hypothesis testing: A lesson from fano. In *International Conference on Machine Learning*, pages 11998–12011. PMLR.

Xuanli He, Chen Chen, Lingjuan Lyu, and Qiongkai Xu. 2022. Extracted bert model leaks more information than you think! *arXiv preprint arXiv:2210.11735*.

Hongsheng Hu, Zoran Salcic, Lichao Sun, Gillian Dobbie, Philip S Yu, and Xuyun Zhang. 2022. Membership inference attacks on machine learning: A survey. *ACM Computing Surveys (CSUR)*, 54(11s):1–37.

Jie Huang, Hanyin Shao, and Kevin Chen-Chuan Chang. 2022. Are large pre-trained language models leaking your personal information? *arXiv preprint arXiv:2205.12628*.

Zhichao Huang and Tong Zhang. 2019. Black-box adversarial attack with transferable model-based embedding. *arXiv preprint arXiv:1911.07140*.

Abhyuday Jagannatha, Bhanu Pratap Singh Rawat, and Hong Yu. 2021. Membership inference attack susceptibility of clinical language models. *arXiv preprint arXiv:2104.08305*.

Bargav Jayaraman and David Evans. 2022. Are attribute inference attacks just imputation? In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*, pages 1569–1582.

Zeljko Kraljevic, Thomas Searle, Anthony Shek, Lukasz Roguski, Kawsar Noor, Daniel Bean, Aurelie Mascio, Leilei Zhu, Amos A Folarin, Angus Roberts, Rebecca Bendayan, Mark P Richardson, Robert Stewart, Anoop D Shah, Wai Keong Wong, Zina Ibrahim, James T Teo, and Richard J B Dobson. 2021. Multi-domain clinical natural language processing with MedCAT: The medical concept annotation toolkit. *Artif. Intell. Med.*, 117:102083.

Zeljko Kraljevic, Anthony Shek, Joshua Au Yeung, Ewart Jonathan Sheldon, Haris Shuaib, Mohammad Al-Agil, Xi Bai, Kawsar Noor, Anoop D Shah, Richard Dobson, et al. 2023. Validating transformers for redaction of text from electronic health records in real-world healthcare. In *2023 IEEE 11th International Conference on Healthcare Informatics (ICHI)*, pages 544–549. IEEE.

Yinhan Liu. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Fatemehsadat Mireshghallah, Kartik Goyal, Archit Uniyal, Taylor Berg-Kirkpatrick, and Reza Shokri. 2022. Quantifying privacy risks of masked language models using membership inference attacks. *arXiv preprint arXiv:2203.03929*.

John X Morris, Volodymyr Kuleshov, Vitaly Shmatikov, and Alexander M Rush. 2023. Text embeddings reveal (almost) as much as text. *arXiv preprint arXiv:2310.06816*.

Yifan Peng, Shankai Yan, and Zhiyong Lu. 2019. Transfer learning in biomedical natural language processing: An evaluation of bert and elmo on ten benchmarking datasets. In *Proceedings of the 2019 Workshop on Biomedical Natural Language Processing (BioNLP 2019)*.

Xiangyu Qi, Yi Zeng, Tinghao Xie, Pin-Yu Chen, Ruoxi Jia, Prateek Mittal, and Peter Henderson. 2023. Fine-tuning aligned language models compromises safety, even when users do not intend to! *arXiv preprint arXiv:2310.03693*.

Congzheng Song and Ananth Raghunathan. 2020. Information leakage in embedding models. In *Proceedings of the 2020 ACM SIGSAC conference on computer and communications security*, pages 377–390.

Liwei Song and Prateek Mittal. 2021. Systematic evaluation of privacy risks of machine learning models. In *30th USENIX Security Symposium (USENIX Security 21)*, pages 2615–2632.

Amber Stubbs, Christopher Kotfila, and Özlem Uzuner. 2015. Automated systems for the de-identification of longitudinal clinical narratives: Overview of 2014 i2b2/uthealth shared task track 1. *Journal of biomedical informatics*, 58:S11–S19.

Amber Stubbs and Özlem Uzuner. 2015. Annotating longitudinal clinical narratives for de-identification: The 2014 i2b2/uthealth corpus. *Journal of biomedical informatics*, 58:S20–S29.

Özlem Uzuner, Imre Solti, and Eithon Cadag. 2010a. Extracting medication information from clinical text. *Journal of the American Medical Informatics Association*, 17(5):514–518.

Özlem Uzuner, Imre Solti, Fei Xia, and Eithon Cadag. 2010b. Community annotation experiment for ground truth generation for the i2b2 medication challenge. *Journal of the American Medical Informatics Association*, 17(5):519–523.

Özlem Uzuner, Brett R South, Shuying Shen, and Scott L DuVall. 2011. 2010 i2b2/va challenge on concepts, assertions, and relations in clinical text. *Journal of the American Medical Informatics Association*, 18(5):552–556.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. 2018. Glue: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*.

Jeffrey G Wang, Jason Wang, Marvin Li, and Seth Neel. 2024. Pandora's white-box: Increased training data leakage in open llms. *arXiv preprint arXiv:2402.17012*.

Samuel Yeom, Irene Giacomelli, Matt Fredrikson, and Somesh Jha. 2018. Privacy risk in machine learning: Analyzing the connection to overfitting. In *2018 IEEE 31st computer security foundations symposium (CSF)*, pages 268–282. IEEE.

Da Yu, Saurabh Naik, Arturs Backurs, Sivakanth Gopi, Huseyin A Inan, Gautam Kamath, Janardhan Kulkarni, Yin Tat Lee, Andre Manoel, Lukas Wutschitz, et al. 2021. Differentially private fine-tuning of language models. *arXiv preprint arXiv:2110.06500*.

A Standard AnonCAT Model

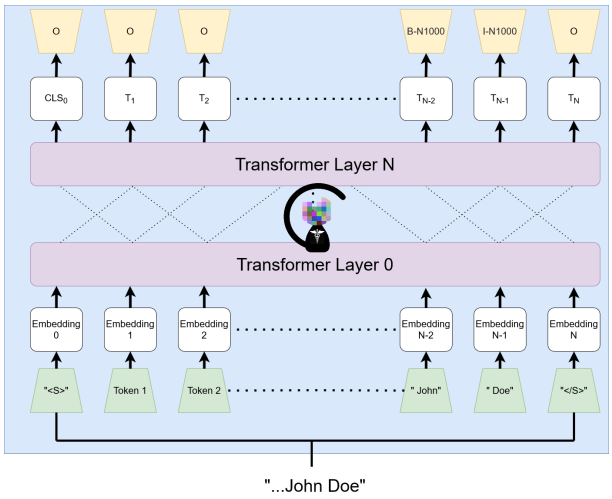


Figure 5: A standard AnonCAT model that would be used for identifying sensitive personal entities within text.