AUDITING DATA PROVENANCE IN TEXT-GENERATION MODELS
SUMMARY REPORT

Mingzhe Du
dumingzhex@gmail.com

1 OVERVIEW

This paper introduces a new auditing model to detect if specific data was used to train a generative language model. They empirically show that this approach can effectively detect with very few queries and can successfully audit well-generalized models that are not overfitted to the training data. Moreover, authors analyze how text-generation models memorize word sequences and explain why this approach works well on them.

2 BACKGROUND

2.1 PROBLEM DEFINITION

To define the problem, the author assumes that the auditor merely has the black-box access to the target model, and the request frequency may be limited. Furthermore, the auditing might be based on a relatively small list of words or even a single word, instead of numeric probabilities.

2.2 MODEL BACKGROUND

In this section, the author firstly introduces the definition of deep learning models, which is composed of a function $f_\theta : x \to y$ parameterized by θ, a labeled training dataset $D = \{(x_i,\ y_i)\}_{i=1}^{n}$ and a loss function L.

Recurrent neural networks are a common architecture for text-generation tasks such as next-word prediction. It maps the input sequence to a sequence of hidden representations $H = [h_1,\ \ldots,\ h_l\ ]$, where the computation of h_i is recursively dependent on the previous hidden representation h_{i-1} and the current input token x_i , and then feeds these hidden representations to a classifier.

Sequence-to-sequence models are a common architecture for text-generation tasks where both the input $X = [x_1,\ \ldots,\ x_m\ ]$ and the output $Y = [y_1,\ \ldots,\ y_n\ ]$ are sequences of tokens. It is widely used in machine translation tasks and dialog generation tasks.

2.2.1 TEXT-GENERATION MODELS

Based on the algorithms mentioned above, the author then introduces three different task models: 1) Next-word prediction, 2)Neural machine translation and 3)Dialog generation.

Next-word prediction: predicts the next token from the given context. RNNs are commonly used for this task. Given a input sequence $X = [x_1,\ \ldots,\ x_m\ ]$, RNNs models the conditional probability $Pr(x_i|x_1, ..., x_{i-1}) = f(x_1, ..., x_{i-1})$ and aims to maximize the probability for the sequence $Pr(x) = \prod_{i=1}^{l} Pr(x_i|x_1, ..., x_{i-1})$. Hence, the loss function is the negative log likelihood: $L(f(x), x) = -\Sigma_{i=1}^{l} Pr(x_i|x_1, ..., x_{i-1})$

Neural machine translation uses the sequence-to-sequence framework that translating one sequence into another one. We can feed a sequence $X = [x_1,\ \ldots,\ x_m\ ]$ in to the encoder, and retrieve a sequence $Y = [y_1,\ \ldots,\ y_n\ ]$ from the decoder. During the training

process, it computes the probability $Pr(y_i|y_1, ..., y_{i-1}; X)$ as $f(y_1, ..., y_{l-1}; X)$. Similar to the next-word prediction task,the loss function is the negative log probability on the target sequence: $L(f(x), x) = -\Sigma_{i=1}^{l} Pr(y_i|y_1, ..., y_{i-1}; x)$

Dialog generation aims to generate replies in a conversation. it can also employ a sequence-to-sequence architecture. Dialog generation tasks have a similar loss function with machine translation tasks.

3 AUDITING TEXT-GENERATION MODELS

To determine whether the target model uses a user's data or not, this paper designs a user-level membership auditing method: the auditor builds a binary user-level membership classifier f audit that takes as input a (processed) list of predictions obtained by querying f with a subset of the user's dataset D_u and outputs a decision on $u \in U_{train}$.

3.1 TRAINING SHADOW MODELS

To collect the data for training f_{audit}, the auditor needs to train k shadow models $f_1\prime ... f_k\prime$ to simulate the target model f firstly. Each shadow model selects a subset of the auxiliary dataset D_{ref}. In Section A.2, this paper shows that public sources can be used for D_{ref} and the loss in audit accuracy is negligible when D_{train} and D_{ref} are drawn from different domains.

3.2 TRAINING THE AUDIT MODEL

The core idea of this paper is using the distribution of the output ranking as signals for inferring user-level membership. As the paper demonstrates in Figure 1, even for a well-generalized model of which test-train accuracy is close, the model ranks relatively rare words much higher in the evaluation as it saw them in the same context during training.

Given a user u's data $D_{ref,u}$, the auditor initially queries the shadow models on each data point $(x, y) \in D_{ref,u}$ and then collects the ranks of y in $f'(x)$ into a rank set R_u. After collecting the ranks for all $(x, y) \in D_{ref,u}$, the auditor builds a histogram for R_u with a fixed number of bins d. The final feature vector H_u is a d-way count vector where each entry is the count of the ranks in that bin. Subsequently, the auditor extracts features H_u and labels them as 1 if $u \in U_{ref}$ and 0 otherwise. Finally, the auditor repeats this procedure on each shadow model, obtains a labeled collection D_{audit} and trains a binary membership classifier $u \in f_{audit}$ on $u \in D_{audit}$. We refer to $u \in f_{audit}$ as the audit model.

3.3 AUDITING MEMBERSHIP IN THE TRAINING DATA

The auditor can randomly sample a few queries to test the target model. However, it should be noted that even the sample can be random, but this paper shows in Section 4.2 that it will be more effective if selecting the relatively less frequent words or sequences.

4 MEMORIZATION IN TEXT-GENERATION MODELS

In this section, authors analyze why this auditing approach works so well for well-trained text-generation models.

4.1 WORD FREQUENCY AND PROBABILITY

Since the text-generation model is driven by the sum of the negative log probability loss function, the model actually tends to memorize sequences that occur in the training data. Figure 1 shows the histograms of the log probabilities of the more and less frequent words in the training and test sequences. There is a gap between the less frequent words in the training sequences and those in the test sequences. This gap indicates that the model assigns higher probabilities to words in the training sequences, producing a strong signal that can be used for membership inference.

Figure 1: Histograms of log probabilities of words generated by text-generation models. The top row are the histograms for the top 20% most frequent words, the bottom row are the histograms for the rest.

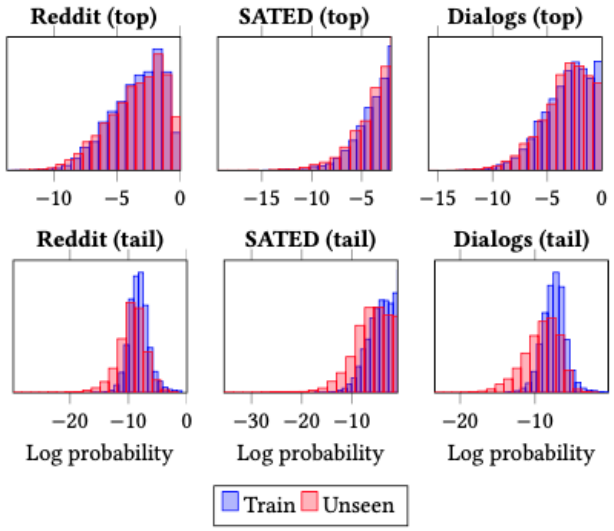
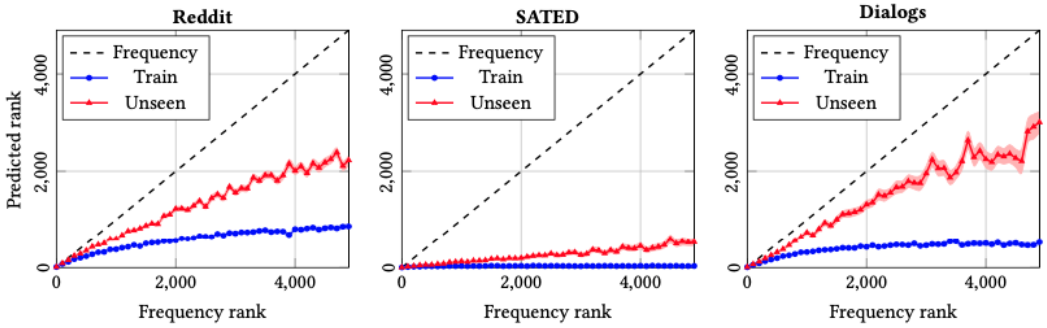


Figure 2: Ranks of words in the frequency table of the training corpus and in the models' predictions.



Furthermore, the 20% frequent words dominate the 80% train datasets, thus text-generation models typically generate words from the top 20% of the word-frequency distribution. This explains why the training and test losses of the model will be similar.

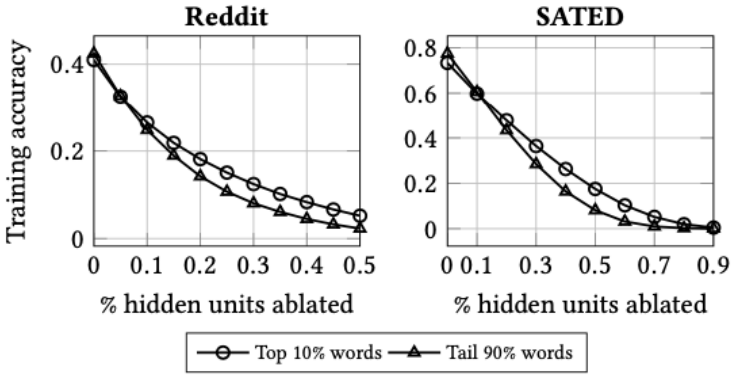
4.2 Word frequency and predicted rank

Figure 2 shows the relationship between a word's frequency rank of the training corpus and its rank in the model's predictions. These charts demonstrate that the models assign much higher rank to words when they appear in training sequences vs. when they appear in test sequences, especially for the less-frequent words. This explains why this auditing algorithm is successful when it queries the target model with less-frequent sequences.

4.3 Ablation analysis

The author conjectures that text-generation models learn generalizable patterns for the most-frequent words while hardly memorizing the less-frequent sequences. As shown in the Figure 3, when no hidden units are ablated, accuracy is similar for the most-frequent words and the rest. However, when more hidden units are ablated, accuracy on the training data degrades quicker for models that are hard-memorizing the training data. This indicates that predicting less-frequent sequences is more dependent on specific hidden units in the model and thus involves more memorization.

Figure 3: Ablation analysis on Reddit and SATED.



5 Conclusion

This paper designs an effective membership auditing approach on text-generation models. This approach can work very well on the given datasets and models with a few queries. Meanwhile, this paper conjectures that text-generation models learn generalizable patterns for the most-frequent words while hard-memorizing the less-frequent sequences. Finally, they also list three limitations of this auditing technique.

6 Improvements

6.1 Auditing on different layers

This paper only audits the output of models, but the middle layers provide more information if we can access them. In the scenario of local-remote model architectures, we can interpret the middle representations and analyze its membership in the similar way.

6.2 Dye Pack

Inspired by the bank dye pack, this idea is putting a few deliberately designed instances into the training dataset. There is a high probability to say that the model trained on the specific dataset if we detect the "Dye pack" from the model, that is much easier than auditing by ordinary training data.

A Experiments

This paper selected the Reddit comments dataset to evaluate next-word prediction tasks, selected The speaker annotated TED talks dataset for machine translation tasks and selected The Cornell movie dialogs corpus for the dialogue-generation tasks. For cross-domain reference, they used the Wikitext-103 corpus, the English- French pair in the Europarl dataset and the Ubuntu dialogs dataset for each task respectively.

A.1 Experiment 1: Performance of target models

Table 1 shows the results for models trained on 300 users, with the test data sampled from 300 disjoint users from the training set.

A.2 Experiment 2: Performance of auditing

They train 10 shadow models for all tasks and use a linear SVM as the audit classifier. The audit model achieves the perfect score (i.e., 1) on all metrics for all datasets and models when there is no restriction on the output size and query time.

Table 1: Performance of target models. Acc is word prediction accuracy, Perp is perplexity.

Dataset	Model	Train Acc	Test Acc	Train Perp	Test Perp
Reddit	1-layer LSTM [12]	0.184	0.206	102.22	113.14
SATED	Seq2Seq w/ attn [24]	0.587	0.535	6.36	10.28
Dialogs	Seq2Seq w/o attn	0.283	0.264	45.57	61.11

Table 2: Effect of training shadow models with different hyper-parameters than the target model.

Dataset	Accuracy	AUC	Precision	Recall
Reddit	0.990	0.993	0.983	0.996
SATED	0.965	0.981	0.937	0.996
Dialogs	0.978	0.998	0.958	1.000

A.2.1 Effect of different hyper-parameters

To demonstrate that knowledge of the target model's hyper-parameters is not essential for successful auditing, we train 10 shadow models for each task with different training configurations. Table 2 provides the result that auditing scores are still above 0.95 on nearly all metrics for all tasks and models.

A.2.2 Effect of the number of users

To evaluate how the number of users in the training dataset affects the auditor's ability to infer the presence of a single user, they train word-prediction models on 100, 500, 1,000, 2000, 4,000, and 10,000 users from the Reddit dataset. The result is shown on Figure 4.

A.2.3 Effect of the number and selection of audit queries

To measure the performance of auditing when the auditor is restricted to only a few queries, they vary the number of audit queries between 1, 2, 4, 8, 16, and 32 word sequences.

A.2.4 Effect of the size of the model's output

In a realistic deployment, the model's output may be limited to a few top-ranked words rather than the entire ranked vocabulary. For the translation task, audit performance is much higher than random guessing even if the model outputs just one top-ranked word and exceeds 0.9 when the model outputs

Figure 4: Effect of the number of Reddit users used to train a word-prediction model.

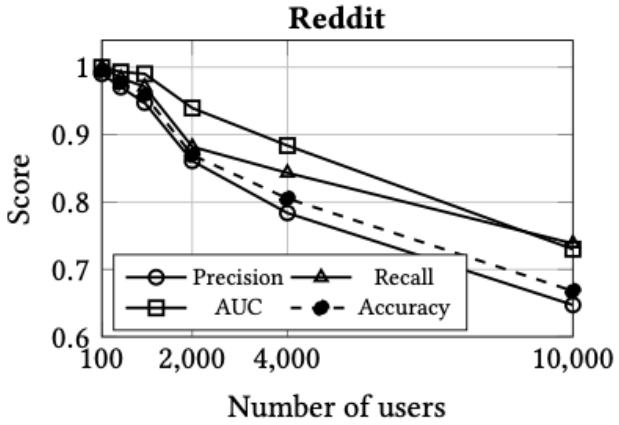


Table 3: Audit performance on obfuscated Reddit comments.

Dataset	Accuracy	AUC	Precision	Recall
Baseline	1.000	1.000	1.000	1.000
Google	0.580	0.858	0.944	0.170
Yandex	0.500	0.782	0.500	0.010

Figure 5: Effect of the number of queries and sampling strategy. Plots on the left show the results when the auditor samples the user's data for queries in the ascending order of frequency counts of tokens in the label; plots on the right show the results with randomly sampled data.

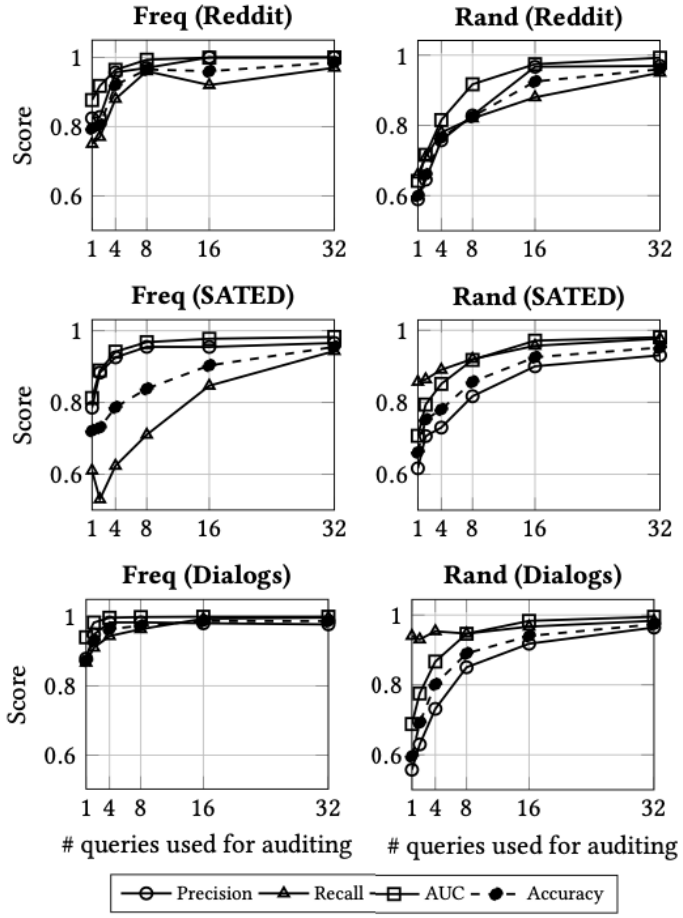
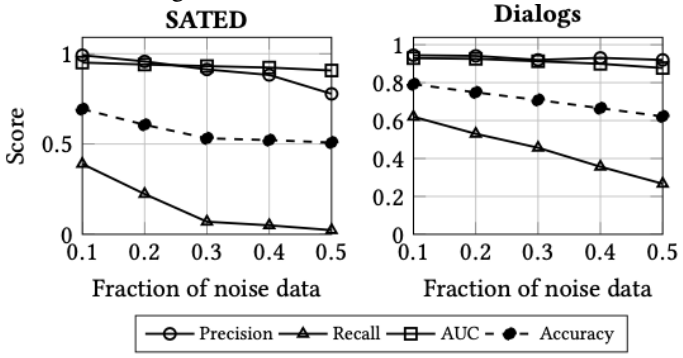


Table 4: Effect of the model's output size.—f(x)— is the number of words ranked by f.

Reddit	**Same domain**				**Cross domain**			
$\|f(x)\|$	Acc	AUC	Pre	Rec	Acc	AUC	Pre	Rec
1	0.545	0.549	0.574	0.350	0.505	0.589	0.667	0.020
5	0.550	0.572	0.553	0.520	0.490	0.525	0.495	0.920
10	0.580	0.602	0.582	0.570	0.500	0.552	0.500	0.950
50	0.605	0.648	0.606	0.600	0.505	0.659	0.503	0.980
100	0.725	0.788	0.765	0.650	0.585	0.714	0.549	0.950
500	0.970	0.998	0.970	0.970	0.905	0.992	0.988	0.820
1000	0.985	0.999	0.971	1.000	0.910	0.999	1.000	0.820
SATED								
$\|f(x)\|$	Acc	AUC	Pre	Rec	Acc	AUC	Pre	Rec
1	0.723	0.785	0.770	0.637	0.723	0.785	0.712	0.750
5	0.748	0.838	0.767	0.713	0.767	0.834	0.755	0.790
10	0.800	0.880	0.783	0.830	0.805	0.878	0.814	0.790
50	0.928	0.973	0.908	0.953	0.925	0.979	0.947	0.900
100	0.948	0.981	0.944	0.953	0.942	0.978	0.965	0.917
500	0.972	0.988	0.958	0.987	0.970	0.988	0.983	0.957
1000	0.960	0.984	0.939	0.983	0.967	0.985	0.973	0.960
Dialogs								
$\|f(x)\|$	Acc	AUC	Pre	Rec	Acc	AUC	Pre	Rec
1	0.577	0.618	0.582	0.547	0.538	0.618	0.520	0.977
5	0.575	0.642	0.582	0.530	0.552	0.643	0.528	0.970
10	0.583	0.645	0.591	0.543	0.543	0.638	0.523	0.977
50	0.605	0.660	0.611	0.580	0.537	0.610	0.520	0.963
100	0.647	0.714	0.643	0.660	0.570	0.669	0.541	0.920
500	0.935	0.975	0.917	0.957	0.925	0.969	0.895	0.963
1000	0.972	0.995	0.955	0.990	0.962	0.992	0.948	0.977

Figure 6: Effect of noise and errors.



50 top-ranked words). These results demonstrate the remarkable extent to which translation models memorize specific word sequences encountered in training.

A.2.5 Effect of noise and errors in the queries

D_u may be noisy or partially erroneous. To evaluate how this affects auditing, for each training user, authors use part of data to train f and hold out the remaining fraction to represent noise during auditing. We vary this fraction between 0.1, 0.2, . . . , 0.5.

A.2.6 Auditing obfuscated data

The authors evaluate the effect of obfuscation on the success of auditing. This is the first step towards determining whether text-generation models memorize specific word sequences rather than higher-level linguistic features.