# DP-Rewrite: Towards Reproducibility and Transparency in Differentially Private Text Rewriting

**Timour Igamberdiev**[1] and **Thomas Arnold**[2] and **Ivan Habernal**[1]
[1]Trustworthy Human Language Technologies
[2]Ubiquitous Knowledge Processing Lab
Department of Computer Science, Technical University of Darmstadt
{timour.igamberdiev,ivan.habernal}@tu-darmstadt.de
arnold@ukp.informatik.tu-darmstadt.de
www.trusthlt.org www.ukp.tu-darmstadt.de

## Abstract

Text rewriting with differential privacy (DP) provides concrete theoretical guarantees for protecting the privacy of individuals in textual documents. In practice, existing systems may lack the means to validate their privacy-preserving claims, leading to problems of transparency and reproducibility. We introduce `DP-Rewrite`, an open-source framework for differentially private text rewriting which aims to solve these problems by being modular, extensible, and highly customizable. Our system incorporates a variety of downstream datasets, models, pre-training procedures, and evaluation metrics to provide a flexible way to lead and validate private text rewriting research. To demonstrate our software in practice, we provide a set of experiments as a case study on the ADePT DP text rewriting system, detecting a privacy leak in its pre-training approach. Our system is publicly available, and we hope that it will help the community to make DP text rewriting research more accessible and transparent.

## 1 Introduction

Protecting the privacy of individuals has been gaining attention in NLP. One particular setup is text rewriting using local differential privacy (DP) (Dwork and Roth, 2013), which provides probabilistic guarantees of 'how much' privacy can be lost in the worst case if an individual gives us their piece of text that has been rewritten with DP. For instance, given a text "I want to fly from Newark to Cleveland on Friday", the system may rewrite it as "Flights from Los Angeles to Houston this week". Only a few recent works have touched on this challenging topic. For example, Krishna et al. (2021) proposed ADePT: A text rewriting system based on the Laplace mechanism. However, it turned out that their DP method was formally flawed (Habernal, 2021). We also see another recent approach, DP-VAE (Weggenmann et al., 2022), which shows results that look surprisingly good for the level of guaranteed privacy. However, neither ADePT nor DP-VAE published their source codes, so the community has no means to perform any empirical checks to validate the privacy-preserving claims. Therefore, the lack of transparency and reproducibility is the main obstacle to the accountability of DP text-rewriting systems.

We asked whether an open, modular, easily extensible, and highly customizable framework for differentially private text rewriting could help the community gain further insight into the utility and potential pitfalls of such systems. We hypothesize that by integrating various downstream datasets, models, pre-training procedures, and evaluation metrics into one software package, we improve the transparency, accountability, and reproducibility of research in differentially private text rewriting.

Our main contributions are twofold. First, this demo paper presents `DP-Rewrite`, an open-source framework for differentially private text rewriting experiments. It includes a correct reimplementation of ADePT as a baseline, integrates pre-training on several datasets, and allows us to easily perform downstream experiments with varying privacy guarantees by adjusting the privacy budget $\varepsilon$. Second, `DP-Rewrite` allows us to easily detect another privacy leak in the approach proposed in ADePT, namely in the pre-training strategy of the autoencoder, with the system memorizing the input data. We demonstrate this in detail as a use case of `DP-Rewrite` in Section 4.[1]

## 2 Related Work

Although the problem of simple data redaction is a widely researched field with several promising approaches (Hill et al., 2016; Lison et al., 2021),

[1]Our project is available at https://github.com/trusthlt/dp-rewrite.



*Proceedings of the 29th International Conference on Computational Linguistics*, pages 2927–2933
October 12–17, 2022.

the related problem of private text transformation is still largely unexplored.

We only briefly sketch the main ideas of local differentially private algorithms in text rewriting. For a longer introduction to DP see, e.g., Habernal (2022); Senge et al. (2021); Igamberdiev and Habernal (2022). Let $x, x' \in \mathcal{X}$ be two data points such as texts or vectors, each belonging to a different person. In DP terminology, $x$ and $x'$ are neighboring datasets, as they differ by one person (Desfontaines and Pejó, 2020). A (local) DP algorithm $\mathcal{M} : \mathcal{X} \to \mathcal{Y}$ is a function that takes any single data point $x \in \mathcal{X}$ and produces its 'privatized' version $y \in \mathcal{Y}$ which might be an arbitrary object, such as a text or a vector. Privatization is achieved by introducing randomness in $\mathcal{M}$. As a result, $(\varepsilon, 0)$-local DP guarantees that for any two neighboring datasets $x, x'$ and any output $y$

$$\ln \left[ \frac{\Pr(\mathcal{M}(x) = y)}{\Pr(\mathcal{M}(x') = y)} \right] \leq \varepsilon \qquad (1)$$

where $\varepsilon$ is the privacy budget; the lower, the better privacy is guaranteed. If a text rewriting algorithm satisfies the local DP, it limits the probability of revealing the true text $x$ after observing the privatized text $y$.

Krishna et al. (2021) proposed ADePT, a DP text rewriting system. It consists of an auto-encoder that learns a compressed latent representation of text, and a DP rewriter that uses the trained auto-encoder, adds Laplace noise to the latent representation vector, and generates the privatized text. Due to a formal error in the scale of the Laplace noise, ADePT violated differential privacy (Habernal, 2021).

Bo et al. (2021) proposed a text rewriting approach that generates words from a latent representation while adding DP noise. However, unlike holistic text rewriting with DP, perturbing text only at the word level does not protect against privacy attacks (Mattern et al., 2022).

Even more current, Weggenmann et al. (2022) proposed an end-to-end approach to text anonymization using a DP autoencoder, claiming to produce coherent texts of high privacy standards. However, several key aspects of the experiments lack a detailed description, while their results look surprisingly good. Since the code base is not public, we cannot reproduce or reimplement their approach, and we cannot prove or refute our suspicions.

## 3 Description of software

The goal of our system is to provide a seamless way to perform differentially private text rewriting, both on existing and custom datasets. A user can either load a dataset that we provide out-of-the-box, or use a custom one. In addition, we want to make it fast and convenient to run experiments for existing methodologies in DP text rewriting (e.g. ADePT), as well as the ability to integrate novel approaches. For this, we have a general *autoencoder* class based on which out-of-the-box and custom models are built. In this sense, our software is designed to be open and modular, where the researcher can swap out existing components to run a variety of experiments, as well as use the software for one's own privatized text rewriting needs.

The core architecture of our system can be seen in Figure 1. We divide the experiments into three distinct **modes**, *pre-training*, *rewriting*, and *downstream*. For all three, the pipeline begins with a dataloader which can either be a dataset provided in the framework, or a custom dataset specified by the user. Additionally, a rewritten dataset can be loaded for downstream experiments. The loaded dataset is then preprocessed according to user-specified parameters and the user's selected model, split into different procedures depending on the model type (e.g. RNN-based, transformer-based). The model is then initialized, optionally from an existing checkpoint. At this point, the main experiment is run based on the specified mode, either (1) pre-training the autoencoder, (2) using an existing checkpoint to rewrite the dataset, or (3) running a downstream model on an original or rewritten dataset. For each mode, a variety of evaluations are available, such as BLEU (Papineni et al., 2002) and BERTScore (Zhang et al., 2019) for pre-training and rewriting, and various classification metrics (e.g. $F_1$ score) for downstream experiments. The differential privacy component is incorporated during the rewriting phase for systems such as ADePT, although our framework also allows to incorporate it during the pre-training stage.

## 4 Case study

We present here a case study that demonstrates the process of using our framework and provides insights into the ADePT system, for which we provide an implementation in the software. Our goal is to investigate the difference in rewritten texts and downstream evaluations when we pre-

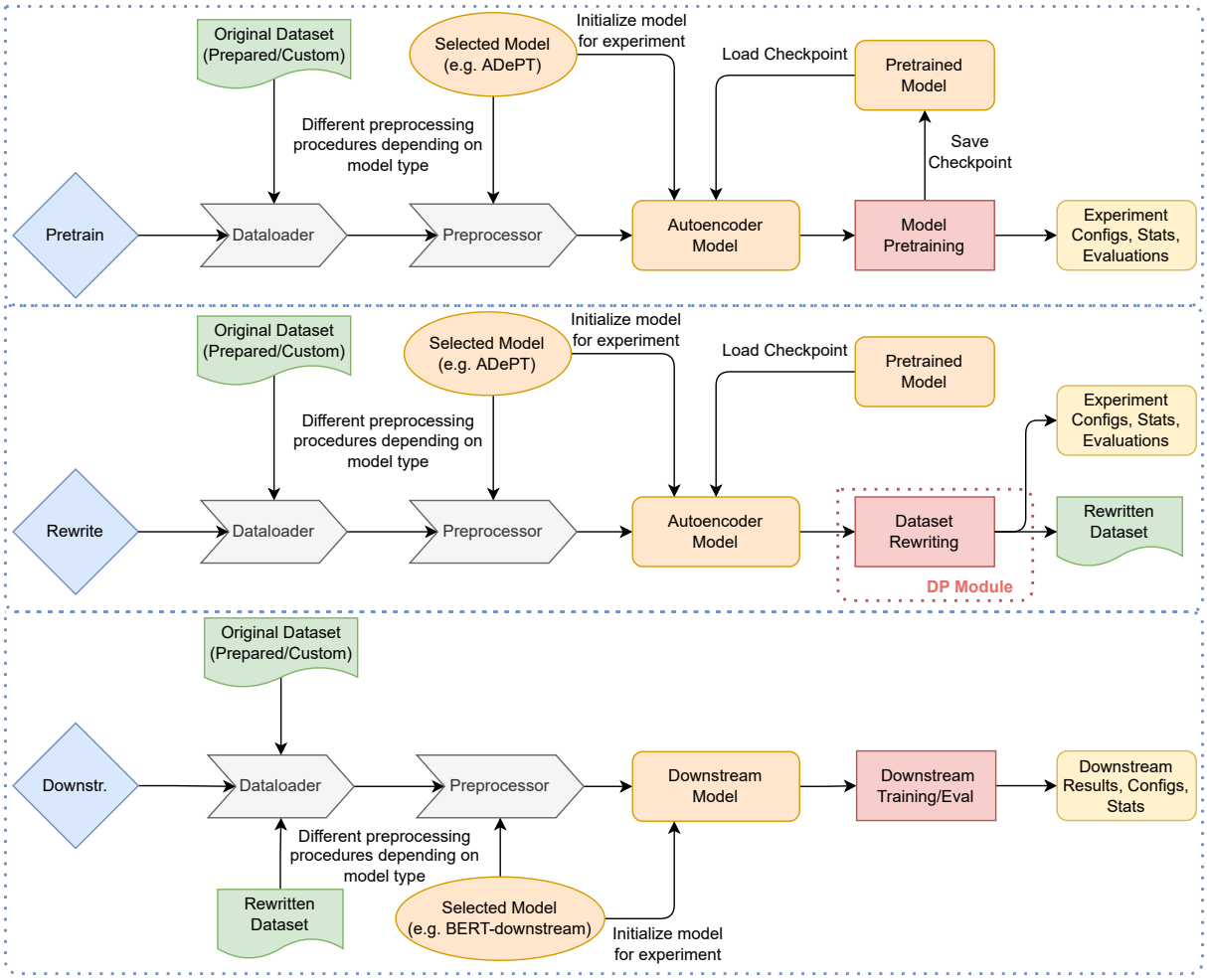


Figure 1: Overall structure of DP-Rewrite. Colors represent groupings of similar components. Blue: Experiment mode. Grey: Dataset preparation. Green: Datasets (original/rewritten). Orange: Model-related components. Red: Main experiment loop. Yellow: Additional experiment outputs.

train an autoencoder on one dataset and use this to rewrite another dataset. If we notice a lot of tokens from the dataset used for pre-training in the rewritten dataset, as well as comparatively higher downstream scores when pre-training and rewriting on the same dataset, then we can be certain of another form of privacy leakage in ADePT.

### 4.1 Datasets

As in Krishna et al. (2021), we use the ATIS (Dahl et al., 1994) and Snips (Coucke et al., 2018) datasets to conduct experiments on an intent classification task in English. For both datasets, we use the same train/validation/test split provided by Goo et al. (2018), with 4,478 training, 500 validation and 893 test examples for ATIS, and 13,084 training, 700 validation and 700 test examples for Snips. There are a total of 26 intent labels in ATIS and 7 in Snips.

### 4.2 Implementation

We start our experiment pipeline by pre-training two models, one on ATIS (1) and the other on Snips (2), in both cases using the training split. For pre-training, we set the vocabulary to the maximum number of words from the training set. As in ADePT, we do not incorporate a differential privacy component during pre-training, although we clip encoder hidden representations with a clipping constant value of 5. We limit sequence lengths to a maximum of 20 tokens, pre-training for 200 epochs with a learning rate of 0.003. In contrast to ADePT, we do not use the $\ell_2$ norm for clipping due to issues in privacy guarantees outlined by Habernal (2021) and instead follow the suggested fix for the method by clipping using the $\ell_1$ norm.

We then use these two models for rewriting, applying both pre-trained models for rewriting the training and validation partitions of ATIS and

Snips, resulting in four rewriting settings in total. For each setting, we rewrite using five $\varepsilon$ values, $\infty, 1000, 100, 10$, and $1$. We use the same clipping constant value of 5 as in pre-training.

See Appendix B for details of the downstream experiment setup.

### 4.3 Results and analysis

Our results can be seen in Figure 2. We observe the main patterns as follows. First and most importantly, datasets rewritten using a model that was pre-trained on the same dataset generally show better downstream results than datasets rewritten using a model pre-trained on a different dataset. For instance, at $\varepsilon = 1,000$, rewritten Snips from a model pre-trained on Snips has an $F_1$ score of $0.94$, while rewritten Snips from a model pre-trained on ATIS has only $0.20$. In fact, this is true even at $\varepsilon = \infty$ (non-private setting), without any added noise (e.g. $0.94$ $F_1$ pre-trained Snips, rewritten Snips vs. $0.18$ $F_1$ pre-trained ATIS, rewritten Snips), since for the latter case the model ends up rewriting the dataset that was pre-trained on, having memorized many of its examples. This can be seen in Figure 3, where the rewritten sentences appear to have no resemblance to the original dataset used for rewriting, but are very similar to the data used for pre-training.

Next, as expected, the results decrease for all configurations as the privacy budget $\varepsilon$ decreases, except for rewritten ATIS from a model pre-trained on Snips, where results are low for all $\varepsilon$ values, probably due to the same reasons as shown in Figure 3. At the lower $\varepsilon$ values of 10 and 1, performance is very low for all configurations. Since there is so much noise added to the encoder hidden representations, the utility of ADePT's rewriting is severely diminished, for any data inputs.

Finally, compared to running downstream experiments on the original dataset, Snips rewritten with a model pre-trained on Snips shows about the same results at high $\varepsilon$ values (e.g. $0.94$ $F_1$ pre-trained Snips, rewritten Snips vs. $0.95$ $F_1$ original Snips for $\varepsilon = \infty$). ATIS rewritten with a model pre-trained on ATIS shows lower results in this case (e.g. $0.73$ $F_1$ pre-trained ATIS, rewritten ATIS vs. $0.87$ $F_1$ original ATIS for $\varepsilon = \infty$). We speculate that since ATIS is a smaller dataset, there are less data points to effectively pre-train ADePT for the autoencoding task. We additionally report random and majority baselines in Appendix A on the original datasets for comparison.

We have thus shown that, despite fixing the theoretical privacy guarantees of ADePT, the pre-training procedure still results in privacy leakage, with rewritten datasets exposing a lot of information from the dataset used for pre-training. As a result, downstream performance is inflated if the datasets for pre-training and rewriting are the same.

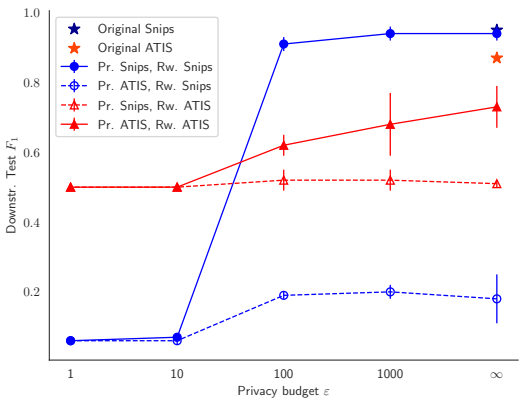


Figure 2: Downstream macro-averaged $F_1$ results for case study experiments with pre-trained and rewritten Snips/ATIS datasets, as well as comparisons with results on the original datasets ("Original Snips" and "Original ATIS"). Lower $\varepsilon$ corresponds to better privacy.

| **Snips rewritten from ATIS** $\varepsilon = 1000$ | |
|---|---|
| **Original Snips doc.** | listen to westbam alumb allergic on google music |
| **Rewritten Snips doc.** | how many people fly on after a turboprop airport |
| **ATIS doc. similar** | how many people fly on a turboprop airport |
| **ATIS rewritten from Snips** $\varepsilon = 1000$ | |
| **Original ATIS doc.** | what flights leave from phoenix |
| **Rewritten ATIS doc.** | start playing my disney spring |
| **Snips doc. similar** | start playing my disney playlist |

Figure 3: Sample rewritten texts showing memorization by ADePT model when pre-training and rewriting on different datasets. For a given document in the original dataset ("Original Snips/ATIS doc."), its rewritten version by the model ("Rewritten Snips/ATIS doc.") has no resemblance to it, but is very similar to another document from the pre-trained dataset ("ATIS/Snips doc. similar").

## 5 Conclusion

We introduced `DP-Rewrite`, an open-source framework for differentially private text rewriting experiments. We have demonstrated a sample use-case for our framework, which allows us to detect privacy leakage in the pre-training procedure of

the ADePT system, an example of how the modular and customizable nature of the software allows for transparency and reproducibility in DP text-rewriting research. `DP-Rewrite` is continuing to be under active development, and we are incorporating new datasets and private text rewriting methodologies as they are released. We welcome feedback from the community.

## Acknowledgements

The independent research group TrustHLT is supported by the Hessian Ministry of Higher Education, Research, Science and the Arts. This project was partly supported by the National Research Center for Applied Cybersecurity ATHENE. Thanks to Johannes Gaese and Jonas Rikowski who contributed in the initial implementation phases.

## References

Haohan Bo, Steven H. H. Ding, Benjamin C. M. Fung, and Farkhund Iqbal. 2021. ER-AE: Differentially private text generation for authorship anonymization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3997–4007, Online. Association for Computational Linguistics.

Alice Coucke, Alaa Saade, Adrien Ball, Théodore Bluche, Alexandre Caulier, David Leroy, Clément Doumouro, Thibault Gisselbrecht, Francesco Caltagirone, Thibaut Lavril, et al. 2018. Snips voice platform: an embedded spoken language understanding system for private-by-design voice interfaces. *arXiv preprint arXiv:1805.10190*.

Deborah A Dahl, Madeleine Bates, Michael K Brown, William M Fisher, Kate Hunicke-Smith, David S Pallett, Christine Pao, Alexander Rudnicky, and Elizabeth Shriberg. 1994. Expanding the scope of the atis task: The atis-3 corpus. In *Human Language Technology: Proceedings of a Workshop held at Plainsboro, New Jersey, March 8-11, 1994*.

Damien Desfontaines and Balázs Pejó. 2020. SoK: Differential privacies. *Proceedings on Privacy Enhancing Technologies*, 2020(2):288–313.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Cynthia Dwork and Aaron Roth. 2013. The Algorithmic Foundations of Differential Privacy. *Foundations and Trends® in Theoretical Computer Science*, 9(3-4):211–407.

Chih-Wen Goo, Guang Gao, Yun-Kai Hsu, Chih-Li Huo, Tsung-Chieh Chen, Keng-Wei Hsu, and Yun-Nung Chen. 2018. Slot-gated modeling for joint slot filling and intent prediction. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 753–757.

Ivan Habernal. 2021. When differential privacy meets NLP: The devil is in the detail. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1522–1528, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Ivan Habernal. 2022. How reparametrization trick broke differentially-private text representation learning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 771–777, Dublin, Ireland. Association for Computational Linguistics.

Steven Hill, Zhimin Zhou, Lawrence K Saul, and Hovav Shacham. 2016. On the (in) effectiveness of mosaicing and blurring as tools for document redaction. *Proc. Priv. Enhancing Technol.*, 2016(4):403–417.

Timour Igamberdiev and Ivan Habernal. 2022. Privacy-Preserving Graph Convolutional Networks for Text Classification. In *Proceedings of the Language Resources and Evaluation Conference*, pages 338–350, Marseille, France. European Language Resources Association.

Satyapriya Krishna, Rahul Gupta, and Christophe Dupuy. 2021. ADePT: Auto-encoder based differentially private text transformation. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2435–2439, Online. Association for Computational Linguistics.

Pierre Lison, Ildikó Pilán, David Sanchez, Montserrat Batet, and Lilja Øvrelid. 2021. Anonymisation models for text data: State of the art, challenges and future directions. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4188–4203, Online. Association for Computational Linguistics.

Justus Mattern, Benjamin Weggenmann, and Florian Kerschbaum. 2022. The Limits of Word Level Differential Privacy. *arXiv preprint*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Manuel Senge, Timour Igamberdiev, and Ivan Habernal. 2021. One size does not fit all: Investigating strategies for differentially-private learning across NLP tasks. *arXiv preprint*.

Benjamin Weggenmann, Valentin Rublack, Michael Andrejczuk, Justus Mattern, and Florian Kerschbaum. 2022. DP-VAE: Human-Readable Text Anonymization for Online Reviews with Differentially Private Variational Autoencoders. In *Proceedings of the ACM Web Conference 2022*, pages 721–731, Virtual Event. ACM.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

## A Detailed results of the case study

| Pretr. Dat. | Rewr. Dat. | $\varepsilon$ | Test $F_1$ |
|---|---|---|---|
| Snips | Snips | $\infty$ | 0.94 (0.02) |
| Snips | Snips | 1,000 | 0.94 (0.02) |
| Snips | Snips | 100 | 0.91 (0.02) |
| Snips | Snips | 10 | 0.07 (0.01) |
| Snips | Snips | 1 | 0.06 (0.00) |
| ATIS | Snips | $\infty$ | 0.18 (0.07) |
| ATIS | Snips | 1,000 | 0.20 (0.02) |
| ATIS | Snips | 100 | 0.19 (0.01) |
| ATIS | Snips | 10 | 0.06 (0.01) |
| ATIS | Snips | 1 | 0.06 (0.01) |
| Snips | ATIS | $\infty$ | 0.51 (0.01) |
| Snips | ATIS | 1,000 | 0.52 (0.03) |
| Snips | ATIS | 100 | 0.52 (0.03) |
| Snips | ATIS | 10 | 0.50 (0.01) |
| Snips | ATIS | 1 | 0.50 (0.01) |
| ATIS | ATIS | $\infty$ | 0.73 (0.06) |
| ATIS | ATIS | 1,000 | 0.68 (0.09) |
| ATIS | ATIS | 100 | 0.62 (0.03) |
| ATIS | ATIS | 10 | 0.50 (0.01) |
| ATIS | ATIS | 1 | 0.50 (0.01) |
| **Snips Orig.** | | | 0.95 (0.01) |
| **ATIS Orig.** | | | 0.87 (0.03) |
| **Snips Rand.** | | | 0.14 |
| **ATIS Rand.** | | | 0.01 |
| **Snips Maj.** | | | 0.03 |
| **ATIS Maj.** | | | 0.13 |

Table 1: Downstream macro-averaged $F_1$ results for case study experiments with pre-trained and rewritten Snips/ATIS datasets. We additionally show results on the original datasets, as well as random and majority baselines. Test $F_1$ shown as "mean (standard deviation)" over five runs with different random seeds. Lower $\varepsilon$ corresponds to better privacy.

## B Downstream experiment setup

For downstream experiments, we use a pre-trained BERT model (Devlin et al., 2018), with an additional feedforward layer that takes the mean of the last hidden states as input and predicts the output label. We use the rewritten training and validation sets for each configuration, and the original test sets for final evaluation. We run each configuration with five different random seeds and report the mean and standard deviation.