Privacy-Preserving Knowledge Transfer through Partial Parameter Sharing

Paul Youssef Jörg Schlötterer Christin Seifert
University of Duisburg-Essen
University of Marburg
{firstname.lastname}@uni-marburg.de

Abstract

Valuable datasets that contain sensitive information are not shared due to privacy and copyright concerns. This hinders progress in many areas and prevents the use of machine learning solutions to solve relevant tasks. One possible solution is sharing models that are trained on such datasets. However, this is also associated with potential privacy risks due to data extraction attacks. In this work, we propose a solution based on sharing parts of the model's parameters, and using a proxy dataset for complimentary knowledge transfer. Our experiments show encouraging results, and reduced risk to potential training data identification attacks. We present a viable solution to sharing knowledge with data-disadvantaged parties, that do not have the resources to produce high-quality data, with reduced privacy risks to the sharing parties. We make our code publicly available.[1]

1 Introduction

NLP research in many areas (e.g., healthcare) is hindered by the unavailability of publicly-available datasets. Even though such datasets might be available for some researchers, sharing them with the community is problematic in many cases due to privacy and copyright concerns (Liu et al., 2021).

De-identifying sensitive information in such datasets is a potential option. However, depending on the nature of the data, the utility of the data might be negatively affected (Jordon et al., 2021) when de-identifying the data. Sharing a model that is trained on the data instead of directly sharing the data itself is another option (Lehman et al., 2021). The shared model transfers knowledge gained from raw data and is beneficial in many cases (e.g., when an institute is interested in solving the same task, but lacks enough data). However, sharing the model is also associated with potential re-identification risks (Carlini et al., 2021).

Instead of directly sharing models or data, data-free knowledge distillation (DF-KD) aims to transfer the knowledge from a large teacher model to a smaller student model without relying on any task-specific data, i.e., data that has been used to train the teacher model. Instead, many approaches make use of a proxy dataset (Krishna et al., 2020) to facilitate the knowledge transfer.

In this work, we propose a solution to the problem of sharing knowledge between models in a privacy-preserving manner. Our solution depends on sharing parts of the model, and using a proxy dataset for complementary knowledge transfer. Partially sharing the model mitigates potential privacy risks. Further training on a proxy dataset helps compensating the loss caused by the absence of the non-shared parts of the model.

We experiment on two datasets for text classification from the clinical domain, **AP** (Gao et al., 2023) for relation classification and **MedNLI** (Romanov and Shivade, 2018) for natural language inference, and show that our approach substantially improves the performance of a student model trained only on a proxy dataset. Additionally, we show that the resulting model cannot be leveraged to reliably identify the original training data.

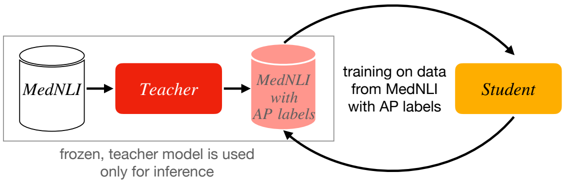
MedNLI
Teacher
MedNLI with AP labels
training on data from MedNLI with AP labels
Student
frozen, teacher model is used only for inference

Figure 1: The process of using the proxy dataset, **MedNLI**, to indirectly train the student model on the target task, **AP**. Left: **MedNLI** is labeled with **AP** labels using a teacher model that was trained on **AP** before. Right: Training the student model with the proxy dataset, i.e., **MedNLI** inputs and **AP** labels.

[1] https://github.com/paulyoussef/ppkt/

2 Related Work

Knowledge distillation (KD). The goal of knowledge distillation is to transfer knowledge from a large teacher model to a student model of a smaller size. Hinton et al. (2015) propose training the student model such, that its output distribution matches the output distribution of the teacher. In order to distill knowledge from BERT (Devlin et al., 2019) into a smaller transformer architecture, Sanh et al. (2019) additionally use the masked language modeling loss used to pre-train BERT and a cosine embedding loss in order to make the hidden representations from both models more similar on the original pre-trainig corpus of BERT. Haidar et al. (2022) randomly choose two intermediate layers from the teacher and the student and train the student's layer to produce similar representations to that of the teacher. In our method, we make use of the teacher's hard predictions, and do not assume access to its outputs distribution.

Data-free knowledge distillation (DF-KD). Even though the teacher's training data can be used in KD, the DF-KD setting assumes the unavailability of such data. Lopes et al. (2017) aim to reconstruct the teacher's training set using the teacher's activation records on the same data. Rashid et al. (2021) use an adversarial generator to generate out-of-domain data, on which the teacher and student disagree the most, and then use this data to train the student. Krishna et al. (2020) show that it is possible to extract a model using its predictions on nonsensical data, but put no restrictions on the size of the model. Our work assumes the availability of a proxy dataset from a related task and that the teacher and the student share the same architecture.

Data extraction from language models. Carlini et al. (2021) show that it is possible to extract training data from GPT-2 (Radford et al., 2019). Huang et al. (2022) experiment on GPT-Neo (Gao et al., 2020) and show that it could leak sensitive information, but the chances of extracting information about a specific user are small because of the model's weak association abilities. Similar work that targets BERT (Vakili and Dalianis, 2021; Lehman et al., 2021) suggests that extracting sensitive information from BERT is unlikely, but robustness against more sophisticated attacks cannot be guaranteed. Membership inference attacks, that aim to identify whether certain data instances have been used to train the model, show some success against BERT (Shejwalkar et al., 2021). We conduct a membership inference attack, in order to inspect if the student models we produce can be used to identify the teacher's training examples.

3 Problem Statement

Let T be a teacher model, trained for a specific task $target$ on training data D_{target} and S be a student model with the same architecture, but untrained. We are interested in transferring the knowledge captured by T on D_{target} to S without providing S any access to D_{target}. Ideally, S cannot be used to identify any data from $target$. S can be trained on any data that does not come from the same distribution as D_{target}. We refer to such data as D_{proxy}. T can provide predictions on D_{proxy} based on what it has learned on D_{target}. We measure the performance of both, T and S, using a held-out test set from $target$, which we refer to as $D^{'}_{target}$.

4 Method

Our method for transferring knowledge from T to S without using any task-specific data, consists of two parts: 1) partial parameter sharing, 2) finetuning on a proxy dataset.

Partial parameter sharing. Since T and S have the same architecture, we copy parameters from N non-adjacent layers of T, and use them directly in the corresponding position in S, in order to facilitate knowledge transfer from T to S. We consider sharing only non-adjacent layers from T, since having several consecutive layers in their initialized state might result in representations of lower quality. We keep the parameters from T fixed during the later finetuning step to avoid degrading to parameters of lower quality. Since the parameters from T reflect a compressed version of the data, we conjecture that partially sharing them provides S only with a distorted and partial view of D_{target}.

Finetuning on a proxy dataset. Sharing parameters in the first step only affects N layers from S, the rest of the layers in S are kept in their state from pre-training, and the task-specific parameters are randomly initialized. In order to make these layers contribute to the knowledge transfer as well, we finetune the model using the proxy dataset D_{proxy}. Note that D_{proxy} contains data that are not part of $target$, but that are artificially labeled using T. Hence, D_{proxy} can be unlabeled. This process is depicted in Figure 1. We only use hard predictions from T, i.e., we only use the class with the highest probability as label and do not use T's probability distribution over all classes. We leave experimenting with T's probability distribution over all classes for future work. To train the student model, we use the cross-entropy loss:

$$L_{CE} = -\sum_{c=1}^{C} y_{t,c} \log(y_{s,c}) \qquad (1)$$

where C is the number of classes, $y_{t,c} \in \{0, 1\}$ is the teacher's prediction, indicating if the input belongs to the c-th class or not, and $y_{s,c}$ is the students' model probability for class c.

5 Experimental Setup

In this section, we describe the data and the experiments we design to evaluate our proposed method for knowledge transfer.

5.1 Data

We use two datasets in our experiments. The first one, AP, acts as the target task, whose data should be kept private. The second dataset, MedNLI, is larger and we use it as a proxy dataset to transfer knowledge from the teacher model. Table 1 provides statistics on both datasets, and Table 2 shows an example from each dataset.

The **Assessment and Plan Relation Labeling (AP)** (Gao et al., 2023) dataset is based on clinical notes from MIMIC-III v1.4 (Johnson et al., 2016). Each instance consists of an assessment that describes the current state of the patient and her active health problems, a plan that handles a specific problem, and a label that describes the relation between the assessment and the plan (direct, indirect, neither or irrelevant). We set the training and test sets of AP to be D_{target} and D'_{target} respectively, i.e., AP is our target task.

The **Medical Natural Language Inference (MedNLI)** (Romanov and Shivade, 2018) is a dataset for medical language inference. Each instance consists of a premise, a hypothesis and a label belonging to one of three classes (entailment, neutral and contradiction) depending on whether the hypothesis can be entailed from the premise or not. The premise sentences are taken from MIMIC-III v1.3 (Johnson et al., 2016), whereas the hypothesis sentences were generated by clinicians. We set MedNLI to be D_{proxy}, i.e., MedNLI is the proxy dataset, that we label with the teacher, and use for complementary knowledge transfer.

	Training	**Dev**	**Test**	**len$_1$**	**len$_2$**
AP	4633	467	667	40	51.0
MedNLI	11232	1395	1422	20	5.8

Table 1: Dataset statistics. *len$_i$* refers to the average length of the i-th input in tokens. Note that we do not use the test set of MedNLI, the evaluation is done on AP's test set. We report the size of the test set for completeness.

	AP	
Input$_1$	64M with EtOH cirrhosis, Afib, admit with upper GI bleed...	**Label:** Direct
Input$_2$	Anemia. Predominary acute blood loss	
	MedNLI	
Input$_1$	She has cough with sputum, occasional blood streaks but no gross blood.	**Label:** Contradiction
Input$_2$	The patient has normal lungs	

Table 2: Examples from AP and MedNLI

5.2 Target Task Performance

The goal of this experiment is to compare the performance of the **teacher** model with the performance of several student models:

- **student-none:** a student that depends only on the proxy dataset, MedNLI, to learn the target task.
- **student-3:** a student model with 3 non-adjacent layers from the teacher. We select the first 3 layers with even indices.
- **student-6:** the same as student-3, but with 6 layers instead of 3.

We use BERT base-cased (Devlin et al., 2019), which consists of 12 encoder layers, as a base model for both the teacher and the student. Note that other domain-specific BERT-based models (e.g., BioClinicalBERT (Alsentzer et al., 2019)) perform better on both tasks. However, these models are pre-trained on data from MIMIC, and we wanted to avoid confounding our results by this factor. We initially train the teacher model on the AP training set for 3 epochs, with a learning rate of 5×10^{-5}, store a model checkpoint every 20 steps and select the checkpoint with the highest Macro-F1 on the validation set. Similarly, we finetune the student model for 1 epoch using the proxy train and validation sets after substituting some layers (in the case of **student-3** and **student-6**).

5.3 Training Data Identification

The goal of this experiment is to evaluate to what extent the different student models can be used to re-identify training data from the target task, AP, compared to the teacher model.

We create a synthetic dataset of positives (real training data from AP), and negatives (other data). To keep the task challenging, we create negatives by identifying medical entities in the positive example, and replacing these by other randomly chosen entities of the same type. We use a clinical NER model (Zhang et al., 2021) to annotate the entities of type: problem (e.g., diseases), treatment (e.g., medications), and test (e.g., diagnostic tests). We restrict the number of replacements to 4 in each instance (2 in each input part). Our final dataset consists of 100 positive and 100 negative examples.

We evaluate the capability of the models to identify training data after finetuning on the proxy dataset in case of the student models, and after finetuning on the AP dataset in case of the teacher model. We use the positive and negative examples as input to all models, and extract their respective representations of the `[CLS]` token from the last layer. This representation is often used as an input to a linear layer, which outputs the final predictions in classification tasks in BERT.

After extracting the representations for the positive and negative examples, we train a logistic regression model using 4-fold cross validation to predict whether the provided representations constitute real training data or not. Note, that this setting assumes the availability of labeled data to train the logistic regression model, i.e., access to original training data of the model under attack. However, this data should be difficult to acquire in practice. We follow other authors (e.g., (Shejwalkar et al., 2021)) in assuming the availability of such data.

	AP Performance (Macro-F1)	Identification (Accuracy)
majority	11.2	50.00
teacher	**76.9**	**67.40**
student-none	27.1	56.35
student-3	39.0	54.65
student-6	59.3	56.89

Table 3: Performance of all models on AP's test set (section 5.2), and the training data identification task (section 5.3). Majority refers to a majority baseline. The best performing model overall is **bold**. The best performing among the student models is underlined.

6 Results and Discussion

The results for the experiments explained in Sections 5.2 and 5.3 are shown in Table 3. The results show that the teacher model performs the best on AP's test set. This is not surprising, given that the teacher is trained on data that is quite similar to the test data. The gains in performance from training only on the proxy dataset from MedNLI, without sharing any parameters, are limited (see **student-none**). This might be attributed to the fact that the datasets are still different, even though they come from similar tasks (e.g., AP's inputs are much longer than MedNLI's, cf. Table 1). Grafting the student models with parameters from the teacher substantially improves the performance. This is especially apparent as the number of shared layers is increased to six.

However, the good performance of the teacher model on AP makes it more susceptible to the training data identification attack. Indeed, the results in the second column show that the representations from the teacher model are more helpful in identifying the training data than the representations extracted from the student models. The student models in general perform poorly in identifying the real training examples from AP, and their performance is close to that of the majority baseline. This suggests that sharing parameters with student models is harmless, as the representations we extract from them cannot be reliably used to identify the original training data of the teacher.

7 Conclusion

In this work, we presented an approach to tackle knowledge transfer between two parties: a teacher, that is trained on sensitive data, and a student model, that lacks enough data to be trained, but is interested in learning the same task. Our solution depends on the teacher partially sharing some of its parameters with the student, and providing it with predictions on an unlabeled proxy dataset that is different from the target dataset. Our experimental results indicate that the proposed solution is effective in knowledge transfer, and associated with reduced risks to potential training data identification attacks. In future work, we will look into using other model architectures, use more tasks for evaluation, take into account more advanced privacy attacks and consider cross-lingual settings, where the teacher and student use different languages.

References

Emily Alsentzer, John Murphy, William Boag, Wei-Hung Weng, Di Jindi, Tristan Naumann, and Matthew McDermott. 2019. Publicly available clinical BERT embeddings. In *Proceedings of the 2nd Clinical Natural Language Processing Workshop*, pages 72–78, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

Nicholas Carlini, Florian Tramer, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom B Brown, Dawn Song, Ulfar Erlingsson, et al. 2021. Extracting training data from large language models. In *USENIX Security Symposium*, volume 6.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, et al. 2020. The pile: An 800gb dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*.

Yanjun Gao, Dmitriy Dligach, Timothy Miller, John Caskey, Brihat Sharma, Matthew M. Churpek, and Majid Afshar. 2023. DR.BENCH: Diagnostic reasoning benchmark for clinical natural language processing. *Journal of Biomedical Informatics*, 138:104286.

Md Akmal Haidar, Nithin Anchuri, Mehdi Rezagholizadeh, Abbas Ghaddar, Philippe Langlais, and Pascal Poupart. 2022. RAIL-KD: RAndom intermediate layer mapping for knowledge distillation. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1389–1400, Seattle, United States. Association for Computational Linguistics.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

Jie Huang, Hanyin Shao, and Kevin Chen-Chuan Chang. 2022. Are large pre-trained language models leaking your personal information? In *Findings of the Association for Computational Linguistics: EMNLP 2022*.

Alistair EW Johnson, Tom J Pollard, Lu Shen, Li-wei H Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G Mark. 2016. MIMIC-III, a freely accessible critical care database. *Scientific data*, 3(1):1–9.

James Jordon, Daniel Jarrett, Evgeny Saveliev, Jinsung Yoon, Paul Elbers, Patrick Thoral, Ari Ercole, Cheng Zhang, Danielle Belgrave, and Mihaela van der Schaar. 2021. Hide-and-seek privacy challenge: Synthetic data generation vs. patient re-identification. In *Proceedings of the NeurIPS 2020 Competition and Demonstration Track*, volume 133 of *Proceedings of Machine Learning Research*, pages 206–215. PMLR.

Kalpesh Krishna, Gaurav Singh Tomar, Ankur P Parikh, Nicolas Papernot, and Mohit Iyyer. 2020. Thieves on sesame street! model extraction of bert-based apis. In *International Conference on Learning Representations*.

Eric Lehman, Sarthak Jain, Karl Pichotta, Yoav Goldberg, and Byron Wallace. 2021. Does BERT pretrained on clinical notes reveal sensitive data? In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 946–959, Online. Association for Computational Linguistics.

Yuang Liu, Wei Zhang, Jun Wang, and Jianyong Wang. 2021. Data-free knowledge transfer: A survey. *arXiv preprint arXiv:2112.15278*.

Raphael Gontijo Lopes, Stefano Fenu, and Thad Starner. 2017. Data-free knowledge distillation for deep neural networks. *ArXiv*, abs/1710.07535.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Ahmad Rashid, Vasileios Lioutas, Abbas Ghaddar, and Mehdi Rezagholizadeh. 2021. Towards zero-shot knowledge distillation for natural language processing. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6551–6561. Association for Computational Linguistics.

Alexey Romanov and Chaitanya Shivade. 2018. Lessons from natural language inference in the clinical domain.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Virat Shejwalkar, Huseyin A Inan, Amir Houmansadr, and Robert Sim. 2021. Membership inference attacks against NLP classification models. In *NeurIPS 2021 Workshop Privacy in Machine Learning*.

Thomas Vakili and Hercules Dalianis. 2021. Are clinical bert models privacy preserving? the difficulty of extracting patient-condition associations. In *HUMAN@ AAAI Fall Symposium*.

Yuhao Zhang, Yuhui Zhang, Peng Qi, Christopher D Manning, and Curtis P Langlotz. 2021. Biomedical and clinical English model packages for the Stanza Python NLP library. *Journal of the American Medical Informatics Association*.