Pretrained Language Models Rankers on Private Data: Is Online and Federated Learning the Solution?

Guido Zuccon

The University of Queensland, St Lucia, Australia

Abstract

Modern search engines rely upon extensive mining of users queries and interactions: this is the case also for recent advances in rankers based on pretrained language models. Users rightly worry about the privacy implications of this. For domains such as health, personal and enterprise search, sharing of data is strictly prohibited. In this paper we outline a possible solution for building effective and privacy preserving rankers based on state-of-the-art pretrained language models, and the problems that need to resolved for such a solution to successfully materialise.

Search engines are pervasive in our lives. They often are the first point of access to information and key to many of our decision making tasks; they also form an integral part of many data products. At the core of a search engine is the ranker – the model that computes a score for each item to be retrieved, and that ultimately dictates the order of items (i.e. the ranking) produced by the search engine. The technology underneath these rankers is at the height of a revolution, with the recent arise of retrieval and ranking methods that use pretrained language models (PLMs) [1] like BERT [2]. PLMs are based on the transformer architecture, a deep learning model that exploits the mechanism of self-attention to weight the importance of the input data [3]. PLMs are largely parametrised, e.g. BERT has 110M parameters, GPT-3 has 175B. Because of this, they are typically trained on very large extents of text in an self-supervised manner (a process called pre-training); then additional labelled data is used to train these models to specific tasks (a process called fine-tuning). In the case of rankers, fine-tuning is oriented towards learning the relevance of an item, or learning to rank a set of items, and relies upon extensive mining of users queries and interactions. For example, PLMs rankers used in Information Retrieval literature are typically fine-tuned on MS MARCO which comprises of 532,761 training queries with associated relevance labels. Rankers powered by PLMs have shown unprecedented improvements compared to the past keyword-based matching technology (e.g., BM25) [1], which is instead still commonly integrated in most commercial search engine platforms such as Lucene and Elasticsearch. Key to the high effectiveness of PLM rankers is the availability of a large quantity of labelled data for fine-tuning.

The requirement of a large quantity of labelled data for fine-tuning PLM rankers is a problem. Users rightly worry about the privacy implications of this. For domains such as health, personal

and enterprise search, sharing of data is strictly prohibited. Three main barriers are present:

1. Editorial labelling (i.e., labels provided by editors that are not the user/entity who owns the data) would pose a great privacy threat as it requires others to access private data, e.g., an editor reading your WeChat messages.
2. It is unlikely that users would provide a large quantity of explicit labels for their own data, e.g., it does not seem viable to ask you to label a large number of your own Messenger chats.
3. The data itself needs to be transferred to a central search service where the ranker would be trained and operated – and this is often not possible, e.g., consider extracting private medical records from a hospital and allowing a commercial search provider to index and search them.

An alternative is to fine-tune a private PLM ranker just on the (likely small) user data and with limited labels: this however will not produce an effective ranker. A key problem then arises: *How do we improve a PLM ranker capacity to be effective without large training data, while still preserving the users' privacy (i.e., without sharing or leaking of their data)?*

The intuition that I present in this paper is that a novel framework where PLM rankers are fine-tuned in a federated and online manner using implicit user interactions, such as queries and clicks, may provide solutions for creating effective PLM rankers on private, not shared data and with no need for explicit labels. The on-device federated learning of a ranker would allow the data and the ranker to reside directly on the user device while exploiting signals from multiple users for training the ranker in a privacy-preserving manner, i.e. without sharing the actual data. Instead of sharing data, clients fine-tune the ranker on their local data and then communicate to a central server the ranker updates they obtained. Then the server aggregates the local updates from clients to create a new global ranker, which is then shared back with each client. This process may be repeated throughout the lifetime of the search product. Exploiting data from multiple users is vital, as PLM rankers are data-hungry. The online learning process would allow learning rankers "on-the-fly" via continuous updates and by letting the ranker probe the search space to gain higher effectiveness and robustness to shifting query intents [4]. This online learning is coupled with the exploitation of user interactions, such as queries posed to the search service and clicks made on search results. This would make the unlikely labelling of user data not required, as implicit feedback, although noisier than labels, can be effectively exploited for learning rankers [5, 6].

Key to the realisation of the envisioned novel federated online framework for creating effective PLM rankers is the need to address the following challenges:

Challenge 1: How to learn PLM rankers federatively, in an effective and efficient manner? Our previous work on federated online learning to rank has contributed effective rankers and aggregation methods [7] but in the context of feature-based rankers. PLM rankers have been shown to be more effective than traditional feature-based learning to rank methods. The architecture of rankers based on PLMs is largely different, and it is unclear how advances in federated learning to rank apply to PLMs. Recent research has explored federated learning for the training of PLMs [8, 9], however: (1) these methods are for the training of the actual language model, not for the creation of rankers, (2) the loss attributed to the federated process VS. centralised learning is substantial [8]. Besides learning effective PLM rankers, a major challenge will be to do so in an efficient manner. BERT-based rankers are 400+ MB in size,

other PLMs are even larger (e.g. GPT-3 is 350 GB). It is then unimaginable in practical settings that require frequent real-time ranker updates to pass complete models within the federated online learning process after a few interactions (searches) have occurred. On the other hand, infrequent model updates may lead to low ranker effectiveness, as we have shown in the context of federated online learning to rank with feature-based rankers [7, 10].

Challenge 2: What are effective ways to handle noise and bias from clicks when training PLMs rankers? The envisioned framework relies on learning PLM rankers using interaction data, such as clicks, in place of editorial labels. PLM rankers in fact require a substantial amount of data for fine-tuning: the use of readily available interaction data would make up for the absence of labelled data. Learning from this noisy and biased feedback has been largely investigated in the context of learning to rank [11, 6] and online learning to rank [12]. Key to effective learning in the presence of such a signal are methods for de-biasing and de-noising clicks, e.g., online and counterfactual learning. Previous work has devised effective techniques for this problem in the context of feature-based rankers, including merging the counterfactual and online techniques [13]. However, aspects still unexplored are how this noisy and biased signal affects the learning of PLM rankers and how effective learning can take place in this context. Recent work has made initial inroads towards this problem, by creating a dense retrievers method that can exploit historical interaction data (however, not online data) [14].

Challenge 3: What is the effect of out-of-distribution data, e.g., non identically and independently distributed (non-IID) data? Our previous work on non-IID data in federated online learning to rank has shown the effect of out-of-distribution data on rankers and in which situations this is a problem [15]. This was done for feature-based rankers: whether these problems apply to PLM rankers and to what extent is hard to forecast. We note however that, in non-federated settings, PLM rankers have been already shown to suffer when fine-tuning and testing data are out-of-distribution [16, 17, 18].

Challenge 4: Are federated PLM rankers secure and is user privacy maintained? There are two components that can compromise the security of federated rankers and expose users' private data. One is the message passing between clients and server; this can generally be secured either via noise injection or via encryption – but their effect on PLM rankers needs investigation. The other component is the actual ranker. Assume a malicious agent joins the federation and accesses the PLM ranker. Can the ranker be "inspected" to gain knowledge about user data and queries? Unlike feature-based rankers, PLMs rankers directly learn from the raw user searchable data (and not its features) and queries. Previous work has shown that PLMs (not for ranking) can expose private information contained in the text they were learned from [19, 20]. How can we then ensure that the model parameters cannot be "reversed" to understand the data used to obtain them?

Note that the use of online learning and of federated learning is not new to Information Retrieval; in fact, the two have even been already combined in the context of learning to rank [21, 7, 10, 22, 15] – but no work has investigated how these apply to PLMs rankers. We also note that while federated learning has been shown highly effective in general machine learning tasks [23], in online ranking tasks federated learning still displays substantial gaps compared to centralised learning [7]. This may well be the case also for initial attempts at creating federated PLMs rankers. Similarly, federated learning has been applied to pretrained language models in the context of natural language processing tasks [9], but these works do not consider online

learning. We believe that while federated online learning of PLM rankers would likely share some of the challenges present also in the aforementioned areas, it also present specific twist to these challenges as well as challenges that are unique.

I believe that embracing the framework for PLM rankers envisioned in this paper and addressing the challenges outlined above, would allow to create PLM search engines that can be trained in a federated manner, without surrendering user data to a central search service, enabling search vendors to shift ranker creation directly to the user device, a vital requirement when user data should remain not shared.

Acknowledgments

References

[1] J. Lin, R. Nogueira, A. Yates, Pretrained transformers for text ranking: Bert and beyond, Synth. Lec. on Info. Conc., Ret. & Serv. 14 (2021).

[2] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, in: NAACL'19, 2019.

[3] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, in: NIPS'17, 2017.

[4] S. Zhuang, G. Zuccon, How do online learning to rank methods adapt to changes of intent?, in: SIGIR'21, 2021.

[5] K. Hofmann, S. Whiteson, M. de Rijke, Balancing exploration and exploitation in listwise and pairwise online learning to rank for information retrieval, Information Retrieval 16 (2013).

[6] R. Jagerman, H. Oosterhuis, M. de Rijke, To model or to intervene: A comparison of counterfactual and online learning to rank from user interactions, in: SIGIR, 2019.

[7] S. Wang, B. Liu, S. , G. Zuccon, Effective and privacy-preserving federated online learning to rank, in: ICTIR'21, 2021.

[8] P. Basu, T. S. Roy, R. Naidu, Z. Muftuoglu, S. Singh, F. Mireshghallah, Benchmarking differential privacy and federated learning for bert models, arXiv preprint arXiv:2106.13973 (2021).

[9] Y. Tian, Y. Wan, L. Lyu, D. Yao, H. Jin, L. Sun, Fedbert: When federated learning meets pre-training, TIST (2022).

[10] S. Wang, S. Zhuang, G. Zuccon, Federated online learning to rank with evolution strategies: A reproducibility study, in: ECIR'21, 2021.

[11] T. Joachims, A. Swaminathan, T. Schnabel, Unbiased learning-to-rank with biased feedback, in: WSDM'17, 2017, pp. 781–789.

[12] S. Zhuang, Z. Qiao, G. Zuccon, Reinforcement online learning to rank with unbiased reward shaping, Information Retrieval (2022 (To Appear).

[13] S. Zhuang, G. Zuccon, Counterfactual online learning to rank, in: ECIR'20, 2020.

[14] S. Zhuang, H. Li, G. Zuccon, Implicit feedback for dense passage retrieval: A counterfactual approach, in: SIGIR'22, 2022.

[15] S. Wang, G. Zuccon, Is non-iid data a threat in federated online learning to rank?, in: SIGIR'22, 2022.

[16] S. Zhuang, G. Zuccon, Dealing with typos for BERT-based passage retrieval and ranking, in: EMNLP'21, 2021.

[17] S. Zhuang, G. Zuccon, Characterbert and self-teaching for improving the robustness of dense retrievers on queries with typos, in: SIGIR'22, 2022.

[18] C. Sciavolino, Z. Zhong, J. Lee, D. Chen, Simple entity-centric questions challenge dense retrievers, in: EMNLP'21, 2021.

[19] T. Vakili, H. Dalianis, Are Clinical BERT Models Privacy Preserving? The Difficulty of Extracting Patient-Condition Associations (2021).

[20] C. Qu, W. Kong, L. Yang, M. Zhang, M. Bendersky, M. Najork, Natural language understanding with privacy-preserving bert, in: CIKM'21, 2021.

[21] E. Kharitonov, Federated online learning to rank with evolution strategies, in: WSDM'19, 2019, pp. 249–257.

[22] C. Li, H. Ouyang, Federated unbiased learning to rank, arXiv preprint arXiv:2105.04761 (2021).

[23] Q. Yang, Y. Liu, Y. Cheng, Y. Kang, T. Chen, H. Yu, Federated learning, Synthesis Lectures on Artificial Intelligence and Machine Learning 13 (2019) 1–207.