

# Hybrid Database Operations: Learned Operations for Seamless Querying of Textual and Tabular Data

Matthias Urban<sup>1</sup>, Carsten Binnig<sup>1,2</sup>

<sup>1</sup>Technical University of Darmstadt, Karolinenplatz 5, 64289 Darmstadt, Germany

<sup>2</sup>DFKI, 64289 Darmstadt, Germany

## Abstract

Many real-world applications in medicine, finance or other domains need to combine tabular and textual data. In this paper, we present a new approach called hybrid database operations which is a new class of learned database operations that allows users to seamlessly execute SQL queries over text and tabular data. As a main contribution to enable hybrid database operations, we show how state-of-the-art pre-trained language models such as BERT can be used to implement hybrid database operations such as joins or unions. In our initial evaluation, we report first promising results on real-world data sets which indicate that highly accurate hybrid operations can be realized with minimal training overhead.

## Keywords

ML for databases, hybrid database operations, pre-training, multimodal, language models

## 1. Introduction

Relational databases are the predominant approach used today in business and science for managing tabular data. One of the features contributing to their success is the query language SQL which allows structured data to be queried in a simple manner. However, today many applications need to deal with data sources beyond tabular data such as textual sources. While several extensions for textual data such as full-text search or pattern matching [1] have been integrated into relational databases and SQL, it can still not treat textual data sources as first class citizens and allow them to be queried in the same manner as tabular data.

To illustrate this by an example, think of a hospital database, which stores structured data of the patients such as age or gender together with medical reports that contain information about diagnoses and treatments. If all this data would be stored in structured tables, a data analyst could simply author a SQL query that finds out how many patients are currently in the hospital with the same diagnosis. Unfortunately, this is not possible if the information about the diagnoses is only available in textual reports about the patients. In this case, the data analyst today has to set up a complex data extraction pipeline to retrieve all diagnoses. That quickly becomes tedious, especially when the data that needs to be extracted changes over time.

In this paper, we thus present a new approach called *hybrid database operations* which is a new class of learned database operations that allows users to seamlessly execute SQL queries

---


LWDA'22: Lernen, Wissen, Daten, Analysen October 05–07, 2022, Hildesheim, Germany

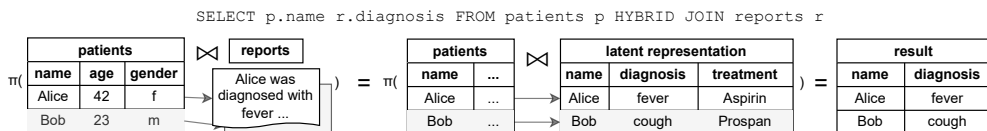
✉ matthias.urban@cs.tu-darmstadt.de (M. Urban); carsten.binnig@cs.tu-darmstadt.de (C. Binnig)

🆔 0000-0002-7418-6181 (M. Urban); 0000-0002-2744-7836 (C. Binnig)



© 2022 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

 CEUR Workshop Proceedings (CEUR-WS.org)



**Figure 1:** Example of a *hybrid JOIN*. We assume the reports can be represented in a *latent tabular representation*, with which we can join the patients table.

over text and tabular data. The basic idea of *hybrid database operations* is that we build on the recent breakthroughs of large pre-trained language models in NLP [2] that have shown to learn new tasks on textual data with only minimal training overhead. This line of work has also inspired researchers to use and extend these large language models towards models that can represent tabular data [3, 4]. However, while these models have been successfully used for NLP-centric tasks, only few papers have been trying to use them for database-related tasks such as running database queries over text collections [5].

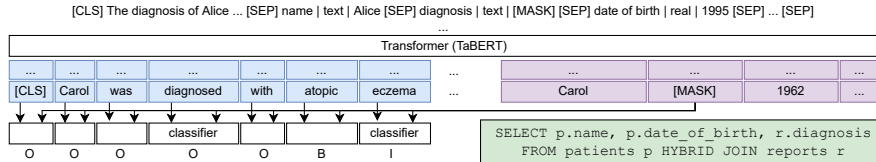
Hence, as a main contribution in this paper, we study how large pre-trained language models can be used to realize hybrid database operations. To be more precise, we show how hybrid database operations such as a *hybrid JOIN* or *hybrid UNION* can be realized as downstream tasks based on recent pre-trained language models that can jointly represent text and tabular data, like TaBERT [3]. These operations extract queried information from text and combine it with the rows of a table (*JOIN*) or add it as rows to a table (*UNION*). For example, as shown in Figure 1, a *hybrid JOIN* enables a data analyst to join the patients table with the patient reports containing information diagnoses without explicitly extracting the required information into a structured table in the first place. Most similar to our work are recent approaches to transform texts to tables [6] which, however, need to be trained from scratch for every new dataset. In contrast, we provide a large pre-training dataset, which allows our model to be usable on unseen domains with little to no training data.

To summarize, at the core, we present two major contributions to enable *hybrid database operations*: (1) We show how *hybrid database operations* can be realized as downstream tasks on top of TaBERT [3]. (2) While TaBERT has been pre-trained on corpora that span over text and tables, the pre-training objectives are not well suited to support SQL operations as downstream tasks. Hence, as a second contribution we present a new pre-training procedure that is better suited for enabling SQL over hybrid data.

## 2. Hybrid Database Operations

In this paper, we investigate *hybrid database operations* on a table  $T$  and a collection of documents  $D$ . For *hybrid JOINs*, each tuple  $t \in T$  is linked to documents  $D_t \subset D$  with a foreign-key relationship. As shown in Figure 1, we assume there is a *latent tabular representation*  $T_D$  of the document collection. Performing a *hybrid JOIN* means joining  $T$  with  $T_D$ . A *hybrid UNION* on the other hand adds the rows of  $T_D$  to  $T$ .

For this short paper, we want to mention that we introduce a couple of restrictions we aim to relax in the future: For the *hybrid JOIN*, we assume there is only one document linked to each tuple (i.e.  $D_t = \{d_t\}$ ), the *latent tabular representation*  $T_D$  is a single table only, and for each tuple  $t$ , only a single row needs to be extracted from the linked document  $d_t$ . For *UNIONs*, we assume



**Figure 2:** Example of a *hybrid JOIN* performed using our model. The linearized input consisting of documents, query attributes and tuples is first fed into TaBERT. Afterwards, in-out-between (IOB) tags are computed using the embeddings of text tokens (blue) and masked cells (purple).

only a single row per text is added to the table.

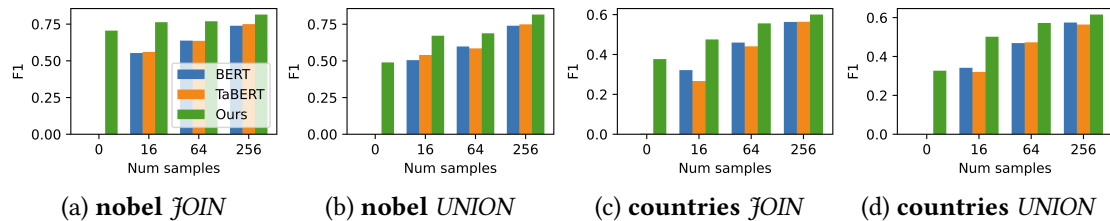
### 3. Model Design

As the basis of our model we choose TaBERT [3], a transformer-based model specifically designed for jointly representing tabular and textual data. In the following, we explain how *hybrid database operations* can be realized as a downstream task of TaBERT.

Figure 2 shows an overview of the model as used in a *hybrid JOIN*. To perform a *hybrid JOINs* with TaBERT, we pair each tuple  $t$  with its join partner  $d_t$ . Additionally, we add the query attributes from the user-provided SQL query to the tuple and add a mask token as the value. The task of the model is to recover the masked values by extracting them from the text and thereby computing the join. To perform *UNIONs* with TaBERT, we pair the text with a tuple, where all values are masked. To realize these *hybrid database operations* as a downstream task, we formulate them as a masked-attribute reconstruction problem. As explained before, the information that needs to be extracted from the textual source is represented as a masked attribute in the model input. To recover the masked attribute from the text, we extend TaBERT with a downstream model that can compute the answers spans in the text using so called in-out-between (IOB) tags. These answer spans are sequences in the text that are a possible value for the masked cells. This approach allows multiple answers per query and avoids so called hallucination (i.e., the model can not simply generate answers that are not existing in the text which is a typical problem of sequence-to-sequence language models).

### 4. Pre-training of Model

To be useful in practice, the model should work on new domains even with little to no training data. Intuitively, the model should learn the essential skills to perform database operations during pre-training. For both presented *hybrid database operations*, it is important to use signal from table context to extract queried information from text. As our main pre-training objective, we thus pair table rows and texts, mask random cell values and ask the model to reconstruct them from the text using IOB tags. A secondary objective aligns column embeddings with text embeddings. Due to the space constraints, we omit the details here. For these pre-training objectives, we constructed a new dataset using T-REx [7], a very large alignment of Wikipedia abstracts with Wikidata triples. We construct the tables by grouping Wikidata entities to tables and sampling Wikidata properties as columns. For each entity, we use the aligned Wikipedia abstract as text, and use the alignment as labels for pre-training.



(a) **nobel JOIN** (b) **nobel UNION** (c) **countries JOIN** (d) **countries UNION**  
**Figure 3:** F1 scores on the *JOIN* and *UNION* workloads of the **nobel** data, and for the **countries** workload. We vary the number of training samples for fine-tuning on the unseen domain.

## 5. Initial Results

In our initial experiments, we are interested how our model performs in unseen domains. For the evaluation, we use also data extracted from T-REx [7] — as for pre-training — but we make sure that the model has not seen any data from these domains. Specifically, we construct two evaluation datasets for our experiments: **nobel** and **countries**.

In our initial results, we fix the model architecture using the new model extensions and compare our pre-training scheme (green) to using the pre-trained weights of BERT [2] (blue) and TaBERT [3] (orange). The results are shown Figure 3. As we can see, due to our pre-training, our model is able to achieve impressive zero-shot performance for *hybrid JOINs*, outperforming both baselines. Despite being worse, our pre-training still performs better than baselines for *hybrid UNIONs* as well.

## 6. Conclusion

We show that large language models can be used to perform *hybrid database operations* having only limited training data. In future work, we aim to support more complex database operations and have a more thorough evaluation on non-Wikipedia texts.

## Acknowledgments

This research was funded by the Hochtief project AICO (AI in Construction). Moreover, we also want to thank hessian.AI at TU Darmstadt as well as DFKI Darmstadt.

## References

- [1] J. R. Hamilton, T. K. Nayak, Microsoft SQL server full-text search, *IEEE Data Eng. Bull.* 24 (2001) 7–10.
- [2] J. Devlin, M. Chang, K. Lee, K. Toutanova, BERT: pre-training of deep bidirectional transformers for language understanding, in: *Proceedings of NAACL-HLT 2019, Association for Computational Linguistics*, 2019, pp. 4171–4186.
- [3] P. Yin, G. Neubig, W. Yih, S. Riedel, Tabert: Pretraining for joint understanding of textual and tabular data, in: *Proceedings of ACL 2020, Association for Computational Linguistics*, 2020, pp. 8413–8426.
- [4] H. Iida, D. Thai, V. Manjunatha, M. Iyyer, TABBIE: pretrained representations of tabular data, in: *Proceedings of NAACL-HLT 2021, Association for Computational Linguistics*, 2021, pp. 3446–3456.
- [5] J. Thorne, M. Yazdani, M. Saeidi, F. Silvestri, S. Riedel, A. Y. Levy, From natural language processing to neural databases, *Proc. VLDB Endow.* 14 (2021) 1033–1039.
- [6] X. Wu, J. Zhang, H. Li, Text-to-table: A new way of information extraction, in: *Proceedings of ACL 2022, Association for Computational Linguistics*, 2022, pp. 2518–2533.
- [7] H. ElSahar, P. Vougiouklis, A. Remaci, C. Gravier, J. S. Hare, F. Laforest, E. Simperl, T-rex: A large scale alignment of natural language with knowledge base triples, in: *Proceedings of LREC 2018, European Language Resources Association (ELRA)*, 2018.