Technical Perspective of TURL: Table Understanding through Representation Learning

Paolo Papotti
papotti@eurecom.fr
EURECOM

Several efforts aim at representing tabular data with neural models for supporting target applications at the intersection of natural language processing (NLP) and databases (DB) [1–3]. The goal is to extend to structured data the recent neural architectures, which achieve state of the art results in NLP applications. Language models (LMs) are usually pre-trained with unsupervised tasks on a large text corpus. The output LM is then fine-tuned on a variety of downstream tasks with a small set of specific examples. This process has many advantages, because the LM contains information about textual structure and content, which are used by the target application without manually defining features.

Language models that consume tables and text can enable solutions that go beyond the limits of traditional declarative specifications built around SQL, such as answering queries expressed in natural language [3], computational fact-checking of textual claims [2], data integration [1], and text generation. Indeed, there is consensus across the two communities of the importance of developing models for representing and "understanding" tables accurately in applications that involve both structured data and natural language. Given the success of transformers in developing pre-trained LMs, such as BERT, there are several proposals to enhance their architecture for representing relational tables.

In this landscape, TURL stands as one of the reference approaches with three main contributions. First, it is among the first solutions extending the transformer architecture to consume structured data and text during pre-training. This is crucial, as it allows the generation of contextual embeddings, which naturally outperform in applications the previous methods based on static embeddings, e.g., tuples linearized and encoded with Word2Vec [1]. Second, it shows how to effectively model the structure in relational tables, with clear notions of tuples and columns. Information expressed in tables cannot be modeled without explicitly capturing such relationships across cell values. Third, it shows the benefits of the fine tuning approach with promising qualitative results over six traditional DB tasks. While the benefits for NLP tasks are well documented, the results over these DB target applications are especially important to show the potential of the LMs with structured data.

To achieve these results, the technical advancements in TURL span across the transformer architecture. To properly model the structure of data in tables, the original transformer is extended and updated by modifying components at different levels. At the *embedding level*, modifications include additional embeddings to explicitly model the table structure and differentiate entity types. At the *encoder level*, a masked self-attention module attends to structurally related elements, such as elements in a row or a column, unlike the traditional transformer where each element attends to all other elements in the sequence. Finally, in terms of *training*, it introduces a reconstruction task with the Masked Entity Recovery pre-training objective. The idea is to reconstruct the correct input from a corrupted one, i.e., randomly mask out entities in the table with the objective of recovering them based on the remaining entities and the sentences associated to the table, such as captions.

Stepping back to look at the bigger picture, the methods proposed in this paper enable the generation of data structure-aware LMs, which can be a fundamental tool to push progress in target applications using structured data. From this perspective, it is clear that TURL makes an important contribution to the general quest of bridging the gap between the NLP and the DB communities.

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

REFERENCES

[1] Riccardo Cappuzzo, Paolo Papotti, and Saravanan Thirumuruganathan. 2020. Creating Embeddings of Heterogeneous Relational Datasets for Data Integration Tasks. In *SIGMOD*.

[2] Rami Aly et al. 2021. FEVEROUS: Fact Extraction and VERification Over Unstructured and Structured information. In *NeurIPS - Datasets and Benchmarks Track*.

[3] Jonathan Herzig, Pawel Krzysztof Nowak, Thomas Mueller, Francesco Piccinno, and Julian Eisenschlos. 2020. TaPas: Weakly Supervised Table Parsing via Pre-training. In *ACL*. 4320–4333.