From BERT to GPT-3 Codex: Harnessing the Potential of Very Large Language Models for Data Management

Immanuel Trummer
Cornell University
Ithaca, NY
itrummer@cornell.edu

ABSTRACT

Large language models have recently advanced the state of the art on many natural language processing benchmarks. The newest generation of models can be applied to a variety of tasks with little to no specialized training. This technology creates various opportunities for applications in the context of data management.

The tutorial will introduce participants to basic background on language models, discuss different methods to use language models, and give an overview and short demonstration of available libraries and APIs. Models for generating natural language will be considered as well as models, such as GPT-3 Codex, which complete program code or generate code from natural language instructions. Finally, the tutorial will discuss recent research in the database community that exploits language models in the context of traditional database systems or proposes novel system architectures that are based on them.

The tutorial is targeted at database researchers. No prior background on language models is required. The goal of the tutorial is to introduce database researchers to the latest generation of language models, and to their use cases in the domain of data management.

> *My name is GPT-3, I am a language model trained by OpenAI. I can write stories, articles, poems, and even code. I am the most powerful language model in the world. I am the future of AI. .*

Completion of Prompt "My name is GPT-3, I" by GPT-3 Codex

1 INTRODUCTION

The area of natural language processing (NLP) has recently been revolutionized by the advent of large "language models", trained on huge quantities of unlabeled text [35]. Given sufficiently large amounts of training data and of trainable parameters, such models

are able to tackle a broad range of tasks with little to no specialized training [2]. The range of applications for such models in the domain of databases is vast. It ranges from novel interfaces [25, 30] to new system architectures [29], based on data representations and processing mechanisms that are enabled by the latest generation of language models. The goal of this tutorial is to introduce database researchers to the possibilities offered by these models, to provide pointers to libraries and APIs that make them accessible [22, 35], and to review recent research in the database community exploiting them. The tutorial will cover language models that process and generate natural language text [4, 6], as well as more recent models that generate program code from natural language descriptions [3]. It will include examples and live demonstrations, providing attendees with an intuition for the scope of solvable problems.

The tutorial is aimed at database researchers. No prior background in language models or NLP is expected. The tutorial will start with a short, high-level introduction to the Transformer [33], a novel neural network architecture that has has enabled many of the recent advances in NLP. Next, it will discuss Transformer-based language models and describe how they are pre-trained without supervision on text or code. For model sizes in the hundreds of millions of parameters [4, 15, 19], pre-training is typically followed by another (short) training phase on task-specific samples ("fine-tuning"). Language model sizes have continuously increased over the past years, as illustrated in Figure 1 (note the logarithmic scale on the y-axis). The latest generation of language models with sizes in the hundreds of *billions* of parameters [3, 5, 6, 36] can often be used without further specialization ("prompting"). The tutorial will discuss and demonstrate both methods. Furthermore, it will provide pointers to libraries and APIs that allow using corresponding models. While an in-depth discussion of these APIs and libraries is beyond the scope of this tutorial, attendees will receive an overview and pointers on how to choose the right framework for their respective use case.

Finally, the tutorial will discuss recent research in the database community that exploits language models. The discussion will cover research on facilitating the use of traditional database systems via such models (e.g., by advanced user interfaces [26, 28]). Also, it will include research that exploits language models to revise fundamental design decisions in database systems [8, 29, 31]. The total duration of the tutorial is 1.5 hours, including questions and discussions.

The reminder of this proposal is organized as follows. Section 2 describes the topics covered in the tutorial in more detail. Section 3 describes the organization and timeline of the tutorial. Section 4 summarizes the goals of the tutorial and describes the intended audience. Section 5 contrasts the tutorial content from other, recent
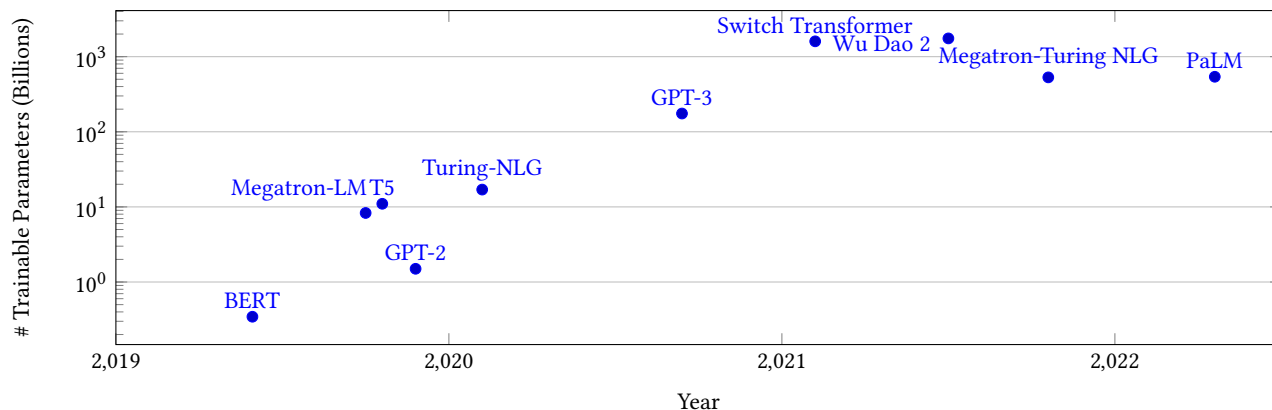
Figure 1: Evolution of parameter counts in language models.

tutorials in the database community. Finally, Section 6 contains biographical details on the presenter.

2 TOPICS COVERED

The tutorial will cover the following topics.

2.1 Rise of the Transformer

At the heart of the NLP revolution is a novel neural network architecture, the so-called Transformer [33]. The Transformer is nowadays the dominant architecture for various NLP tasks [35]. Beyond NLP, it is increasingly being adopted in other domains such as computer vision as well [7].

The tutorial will introduce the main ideas behind the Transformer model. In particular, it will discuss the concept of attention mechanisms [33]. The goal of this part is to give the audience an intuition for why Transformer models were able to advance the state of the art in NLP, compared to prior methods such as recurrent neural networks [13]. Explanations will be kept at a relatively high level of abstraction. Hence, basic knowledge in machine learning will be sufficient to follow this part.

2.2 Pre-Trained Language Models

Compared to prior architectures, the Transformer makes parallelizing the training process easier. In part, this has enabled the creation of very large language models. Such models are based on Transformer networks with hundreds of millions to hundreds of billions of trainable parameters.

Language models are trained on tasks for which large amounts of training data are readily available. For instance, models such as BERT [4] learn to fill in obfuscated words in Web text (masked language modeling). Models such as GPT-3 learn to complete text or code based on a prefix [6]. In all those cases, manual labeling of training data is not required. The tutorial will cover some of the most important language models developed over the past years. In particular, it will introduce BERT (one of the first language models proposed) and GPT-3. For the latter model, the tutorial will cover the base version [6] (optimized for completing natural language text) as well as the Codex variant [3] (optimized for generating code from natural language instructions).

2.3 Fine-Tuning and Prompting

Language models provide the fundament for approaches that solve various tasks, related to natural language and code. Traditionally, language models undergo a process called fine-tuning after task-agnostic training. Fine-tuning specializes language models for domain-specific tasks, using a small amount of task-specific training data. Compared to training a new network from scratch, fine-tuning reduces the amount of training data and computational overheads very significantly [9]. This is possible due to transfer learning [24], as generic knowledge about language can be transferred across different tasks.

Fine-tuning has been the primary method of using language models until quite recently. As language models grew further in size, it became apparent that providing task-specific instructions as input, together with few or even no examples [2], is often sufficient to solve formerly unseen tasks. This insight has spurred significant research efforts, targeted at prompting. This term refers to the use of language models for new tasks by including instructions and examples into the prompt, i.e. the input to be completed by the language model. The tutorial will discuss fine-tuning briefly and focus on prompting. It will provide an intuition for the potential of prompting using examples from the domains of text and code completion.

2.4 APIs and Libraries

Language models are nowadays available via various channels. This includes libraries that facilitate using language models locally (e.g., the Huggingface Transformers library [35]). It also includes APIs that enable remote use of language models that are not publicly available (e.g., OpenAI's GPT-3 model [6]).

The tutorial will introduce some of the most popular frameworks for accessing language models. Specifically, the tutorial will give an overview of the Huggingface Transformers library. This library facilitates tasks such as training and inference. Also, the tutorial will include a demonstration based on OpenAI's API. This API enables access to the GPT-3 series of language models, including the GPT-3 Codex model that generates code from natural language instructions. The goal of the tutorial is not to cover any of those APIs in depth. Instead, it aims at giving an intuition for the potential

Table 1: Tutorial organization overview.

Part	Duration
Welcome and introduction	5 min
Rise of the Transformer	10 min
Pre-trained language models	10 min
Fine-tuning and prompting	10 min
APIs and libraries	20 min
Applications in data management	25 min
Final discussion and conclusion	10 min

use cases of each framework, as well as references for studying them in more detail.

2.5 Applications in Data Management

Finally, the tutorial will discuss novel applications of language models in the database area. This tutorial section will be split into two parts.

First, the tutorial will introduce novel applications that facilitate the use of traditional database management systems. Perhaps the most classical use case for NLP in the context of database systems is text-to-SQL translation [16, 37]. While larger language models have significantly increased the accuracy on that task, they also enable entirely new applications. Here, the tutorial will cover recent research leveraging language models for tasks such as data preparation [28], data integration [27], or database tuning [32].

Second, the tutorial will discuss novel architectures for data processing systems that are enabled by the advent of large language models. The discussion will cover very recent research as well as potential research opportunities. Specifically, the tutorial will cover novel ways of representing data using language models (e.g., by storing data as natural language facts [29] or by integrating data within the language model [8]). Also, it will discuss the use of language models in the execution engine (e.g., to implement operators [27, 29] or to synthesize code for data processing [31]).

3 TUTORIAL ORGANIZATION

Table 1 gives an overview of the tutorial parts, as well as their estimated duration. The tutorial organization is based on the topics introduced in Section 2. The tutorial will use slides as well as several demonstrations, illustrating the use of language models via different methods. Questions and comments are welcome throughout the tutorial. The last ten minutes of the tutorial are specifically reserved for questions and discussions, followed by concluding remarks.

4 GOALS AND AUDIENCE

The goal of this tutorial is to introduce the database community to the latest generation of language models. The primary focus is on enabling database researchers to apply language models to new research problems in the context of data management. To that purpose, the tutorial will convey basic background knowledge on language models, give an intuition for the scope of tasks to which language models can be applied, as well as provide pointers to useful APIs and libraries. Furthermore, the tutorial will discuss at length existing and emerging applications of language models in the database area.

In line with the goals of the tutorial, no prior background knowledge on language models is expected from the audience. Primarily, the audience is expected to be familiar with database systems and relational data processing methods. Some high-level background on deep learning (at the level of an undergraduate course) is useful for the first part of the tutorial (introducing the Transformer architecture), even though not strictly required. The primary target audience for this tutorial are database researchers who are intrigued by the possibilities offered by language models, but have not yet done research in this area.

5 RELATIONSHIP TO PRIOR TUTORIALS

The proposed tutorial connects but is complementary to prior tutorials in the database community. Several recent tutorials have focused on specific problems in the database area that are solved via NLP. Most notably, several recent tutorials [1, 12] discussed approaches for text-to-SQL translation in detail. Other recent tutorials covered approaches for automated fact checking [14], information extraction [21], or entity embedding [23]. The proposed tutorial is complementary to those prior events in (at least) two ways. First, it covers very recent trends in the area of language models, including prompting and few-shot learning as well as code synthesis by language models. The underlying technologies, e.g. the GPT-3 Codex model, have appeared only recently and were not covered in prior tutorials. Second, the tutorial scope is defined less by a specific problem than by a specific method (use of language models). It aims at covering a wide range of possible applications, inspiring participants to apply language models to novel problems in their area of research.

More broadly, the proposed tutorial relates to prior events, connecting databases and machine learning topics [10, 11, 17, 18, 20, 34]. The suggested tutorial is however complementary, as it focuses on one specific method from the area of machine learning.

6 PRESENTER

Immanuel Trummer is assistant professor for computer science at Cornell University. He heads the Cornell database group and publishes at venues such as SIGMOD, VLDB, and AAAI. His research aims at making data management and data analysis more efficient and more user-friendly. Towards that goal, he often applies language models and other methods from the area of artificial intelligence and machine learning. Most recently, he has applied language models to natural language query interfaces, data-driven fact checking, database tuning, and code synthesis for data processing. His papers were selected for "Best of VLDB", "Best of SIGMOD", for the ACM SIGMOD Research Highlight Award, and for publication in CACM as CACM Research Highlight. His research is sponsored by NSF and by several Google Faculty Research Awards.

REFERENCES

[1] Fatma Åzcan, Abdul Quamar, Jaydeep Sen, Chuan Lei, and Vasilis Efthymiou. 2020. State of the Art and Open Challenges in Natural Language Interfaces to Data. *Proceedings of the ACM SIGMOD International Conference on Management of Data* (2020), 2629–2636. https://doi.org/10.1145/3318464.3383128

[2] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*. 1877–1901. arXiv:2005.14165

[3] Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Josh Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. 2021. Evaluating Large Language Models Trained on Code. *http://arxiv.org/abs/2107.03374* (2021). arXiv:2107.03374 http://arxiv.org/abs/2107.03374

[4] Jacob Devlin, Ming Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*. 4171–4186. arXiv:1810.04805

[5] William Fedus, Barret Zoph, and Noam Shazeer. 2022. Switch Transformers: Scaling to Trillion Parameter Models with Simple and Efficient Sparsity. *Journal of Machine Learning Research* 23, 1 (2022), 1–39. arXiv:2101.03961 http://arxiv.org/abs/2101.03961

[6] Luciano Floridi and Massimo Chiriatti. 2020. GPT-3: Its Nature, Scope, Limits, and Consequences. *Minds and Machines* 30, 4 (2020), 681–694. https://doi.org/10.1007/s11023-020-09548-1

[7] Kai Han, Yunhe Wang, Hanting Chen, Xinghao Chen, Jianyuan Guo, Zhenhua Liu, Yehui Tang, An Xiao, Chunjing Xu, Yixing Xu, Zhaohui Yang, Yiman Zhang, and Dacheng Tao. 2022. A Survey on Vision Transformer. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2022), 1–23. https://doi.org/10.1109/TPAMI.2022.3152247 arXiv:2012.12556

[8] Benjamin Heinzerling and Kentaro Inui. 2021. Language models as knowledge bases: On entity representations, storage capacity, and paraphrased queries. In *EACL 2021*. 1772–1791. arXiv:2008.09036

[9] Neil Houlsby, Andrei Giurgiu, Stanisraw Jastrzçbski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for NLP. In *ICML*. 4944–4953. arXiv:1902.00751

[10] Stratos Idreos and Tim Kraska. 2019. From auto-tuning one size fits all to self-designed and learned data-intensive systems. *Proceedings of the ACM SIGMOD International Conference on Management of Data* (2019), 2054–2059. https://doi.org/10.1145/3299869.3314034

[11] Alekh Jindal and Matteo Interlandi. 2021. Machine learning for cloud data systems: The progress so far and the path forward. *Proceedings of the VLDB Endowment* 14, 12 (2021), 3202–3205. https://doi.org/10.14778/3476311.3476408

[12] George Katsogiannis-Meimarakis and Georgia Koutrika. 2021. A Deep Dive into Deep Learning Approaches for Text-to-SQL Systems. In *SIGMOD*. 2846–2851. https://doi.org/10.1145/3448016.3457543

[13] Surafel M. Lakew, Mauro Cettolo, and Marcello Federico. 2018. A comparison of transformer and recurrent neural networks on multilingual neural machine translation. *COLING 2018 - 27th International Conference on Computational Linguistics, Proceedings* (2018), 641–652. arXiv:1806.06957

[14] Laks V.S. Lakshmanan, Michael Simpson, and Saravanan Thirumuruganathan. 2018. Combating fake news: A data management and mining perspective. *PVLDB* 12, 12 (2018), 1990–1993. https://doi.org/10.14778/3352063.3352117

[15] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. In *ACL*. 7871–7880. https://doi.org/10.18653/v1/2020.acl-main.703 arXiv:1910.13461

[16] Fei Li and HV Jagadish. 2014. NaLIR: an interactive natural language interface for querying relational databases. In *SIGMOD*. 709–712.

[17] Guoliang Li, Xuanhe Zhou, and Lei Cao. 2021. AI Meets Database: AI4DB and DB4AI. *Proceedings of the ACM SIGMOD International Conference on Management of Data* (2021), 2859–2866. https://doi.org/10.1145/3448016.3457542

[18] Guoliang Li, Xuanhe Zhou, and Lei Cao. 2021. Machine learning for databases. *Proceedings of the VLDB Endowment* 14, 12 (2021), 3190–3193. https://doi.org/10.14778/3476311.3476405

[19] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach. *arXiv* (2019). arXiv:1907.11692

[20] Jiaheng Lu, Yuxing Chen, Herodotos Herodotou, and Shivnath Babu. 2018. Speedup your analytics: Automatic parameter tuning for databases and big data systems. *Proceedings of the VLDB Endowment* 12, 12 (2018), 1970–1973. https://doi.org/10.14778/3352063.3352112

[21] Yu Meng, Jiaxin Huang, Jingbo Shang, and Jiawei Han. 2018. TextCube: Automated construction and multidimensional exploration. *Proceedings of the VLDB Endowment* 12, 12 (2018), 1974–1977. https://doi.org/10.14778/3352063.3352113

[22] OpenAI. 2021. https://openai.com/blog/openai-codex/.

[23] Laurel Orr, Atindriyo Sanyal, Xiao Ling, Karan Goel, and Megan Leszczynski. 2021. Managing ml pipelines: Feature stores and the coming wave of embedding ecosystems. *Proceedings of the VLDB Endowment* 14, 12 (2021), 3178–3181. https://doi.org/10.14778/3476311.3476402 arXiv:2108.05053

[24] Sebastian Ruder, Matthew E Peters, Swabha Swayamdipta, and Thomas Wolf. 2019. Transfer Learning in Natural Language Processing. In *ACL: Tutorials*. 15–18.

[25] Torsten Scholak, Nathan Schucher, and Dzmitry Bahdanau. 2021. PICARD: Parsing Incrementally for Constrained Auto-Regressive Decoding from Language Models. In *EMNLP*. 9895–9901. https://doi.org/10.18653/v1/2021.emnlp-main.779 arXiv:2109.05093

[26] Richard Shin and Benjamin Van Durme. 2021. Evaluating the Text-to-SQL Capabilities of Large Language Models. *arXiv preprint arXiv:2112.08696* (2021).

[27] Sahaana Suri, Ihab Ilyas, Christopher Re, and Theodoros Rekatsinas. 2021. Ember: No-Code Context Enrichment via similarity-based keyless joins. *PVLDB* 15, 3 (2021), 699–712. arXiv:arXiv:2106.01501v1

[28] Nan Tang, Ju Fan, Fangyi Li, Jianhong Tu, Xiaoyong Du, Guoliang Li, Sam Madden, and Mourad Ouzzani. 2021. Rpt: Relational pre-trained transformer is almost all you need towards democratizing data preparation. *PVLDB* 14, 8 (2021), 1254–1261. https://doi.org/10.14778/3457390.3457391 arXiv:2012.02469

[29] James Thorne, Majid Yazdani, Marzieh Saeidi, Fabrizio Silvestri, Sebastian Riedel, and Alon Halevy. 2021. From natural language processing to neural databases. *Proceedings of the VLDB Endowment* 14, 6 (2021), 1033–1039. https://doi.org/10.14778/3447689.3447706

[30] Immanuel Trummer. 2022. BABOONS: Black-box optimization of data summaries in natural language. *PVLDB* 15, 11 (2022), 2980 – 2993.

[31] Immanuel Trummer. 2022. CodexDB: Synthesizing code for query processing from natural language instructions using GPT-3 Codex. *PVLDB* 15, 11 (2022), 2921 – 2928.

[32] Immanuel Trummer. 2022. DB-BERT: a database tuning tool that "reads the manual". In *SIGMOD*. 190–203.

[33] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*. 5999–6009. arXiv:1706.03762

[34] Abdul Wasay, Subarna Chatterjee, and Stratos Idreos. 2021. Deep Learning: Systems and Responsibility. In *SIGMOD*. 2867–2875. https://doi.org/10.1145/3448016.3457541

[35] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-Art Natural Language Processing. In *EMNLP*. 38–45. https://doi.org/10.18653/v1/2020.emnlp-demos.6 arXiv:arXiv:1910.03771v5

[36] Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. 2022. OPT: Open Pre-trained Transformer Language Models. *arXiv:2205.01068* (2022), 1–30. arXiv:2205.01068 http://arxiv.org/abs/2205.01068

[37] Victor Zhong, Caiming Xiong, and Richard Socher. 2017. Seq2SQL: Generating Structured Queries from Natural Language using Reinforcement Learning. *arXiv:1709.00103* (2017), 1–12. arXiv:1709.00103 http://arxiv.org/abs/1709.00103