# Incorporating External Knowledge to Enhance Tabular Reasoning

**J. Neeraja***
IIT Guwahati
jneeraja@iitg.ac.in

**Vivek Gupta***
University of Utah
vgupta@cs.utah.edu

**Vivek Srikumar**
University of Utah
svivek@cs.utah.edu

## Abstract

Reasoning about tabular information presents unique challenges to modern NLP approaches which largely rely on pre-trained contextualized embeddings of text. In this paper, we study these challenges through the problem of tabular natural language inference. We propose easy and effective modifications to how information is presented to a model for this task. We show via systematic experiments that these strategies substantially improve tabular inference performance.

## 1 Introduction

Natural Language Inference (NLI) is the task of determining if a hypothesis sentence can be inferred as true, false, or undetermined given a premise sentence (Dagan et al., 2013). Contextual sentence embeddings such as BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019), applied to large datasets such as SNLI (Bowman et al., 2015) and MultiNLI (Williams et al., 2018), have led to near-human performance of NLI systems.

In this paper, we study the harder problem of reasoning about *tabular* premises, as instantiated in datasets such as TabFact (Chen et al., 2019) and InfoTabS (Gupta et al., 2020). This problem is similar to standard NLI, but the premises are Wikipedia tables rather than sentences. Models similar to the best ones for the standard NLI datasets struggle with tabular inference. Using the InfoTabS dataset as an example, we present a focused study that investigates (a) the poor performance of existing models, (b) connections to information deficiency in the tabular premises, and, (c) simple yet effective mitigatations for these problems.

We use the table and hypotheses in Figure 1 as a running example through this paper, and refer to the left column as its keys.[1] Tabular inference is challenging for several reasons: (a) **Poor table representation**: The table does not explicitly state the relationship between the keys and values. (b) **Missing implicit lexical knowledge** due to limited training data: This affects interpreting words like *'fewer'*, and *'over'* in H1 and H2 respectively. (c) **Presence of distracting information**: All keys except *No. of listings* are unrelated to the hypotheses H1 and H2. (d) **Missing domain knowledge about keys**: We need to interpret the key *Volume* in the financial context for this table.

| **New York Stock Exchange** | |
|---|---|
| **Type** | Stock exchange |
| **Location** | New York City, New York, U.S. |
| **Founded** | May 17, 1792; 226 years ago |
| **Currency** | United States dollar |
| **No. of listings** | 2,400 |
| **Volume** | US$20.161 trillion (2011) |

H1: NYSE has fewer than 3,000 stocks listed.
H2: Over 2,500 stocks are listed in the NYSE.
H3: S&P 500 stock trading volume is over $10 trillion.

Figure 1: A tabular premise example. The hypotheses H1 is entailed by it, H2 is a contradiction and H3 is neutral i.e. neither entailed nor contradictory.

In the absence of large labeled corpora, any modeling strategy needs to explicitly address these problems. In this paper, we propose effective approaches for addressing them, and show that they lead to substantial improvements in prediction quality, especially on adversarial test sets. This focused study makes the following contributions:

1. We analyse why the existing state-of-the-art BERT class models struggle on the challenging task of NLI over tabular data.
2. We propose solutions to overcome these challenges via simple modifications to inputs using existing language resources.

*The first two authors contributed equally to the work. The first author was a remote intern at University of Utah during the work.

[1] Keys in the InfoTabS tables are similar to column headers in the TabFact database-style tables.

3. Through extensive experiments, we show significant improvements to model performance, especially on challenging adversarial test sets.

The updated dataset, along with associated scripts, are available at https://github.com/utahnlp/knowledge_infotabs.

## 2 Challenges and Proposed Solutions

We examine the issues highlighted in §1 and propose simple solutions to mitigate them below.

**Better Paragraph Representation (BPR):** One way to represent the premise table is to use a universal template to convert each row of the table into sentence which serves as input to a BERT-style model. Gupta et al. (2020) suggest that in a table titled t, a row with key k and value v should be converted to a sentence using the template: "The k of t are v." Despite the advantage of simplicity, the approach produces ungrammatical sentences. In our example, the template converts the *Founded* row to the sentence *"The Founded of New York Stock Exchange are May 17, 1792; 226 years ago."*.

We note that keys are associated with values of specific entity types such as MONEY, DATE, CARDINAL, and BOOL, and the entire table itself has a category. Therefore, we propose type-specific templates, instead of using the universal one.[2] In our example, the table category is *Organization* and the key *Founded* has the type DATE. A better template for this key is "t was k on v", which produces the more grammatical sentence *"New York Stock Exchange was Founded on May 17, 1792; 226 years ago."*. Furthermore, we observe that including the table category information i.e. *"New York Stock Exchange is an Organization."* helps in better premise context understanding.[3] Appendix A provides more such templates.

**Implicit Knowledge Addition (KG implicit):** Tables represent information *implicitly*; they do not employ connectives to link their cells. As a result, a model trained only on tables struggles to make lexical inferences about the hypothesis, such as the difference between the meanings of *'before'* and *'after'*, and the function of negations. This is surprising, because the models have the benefit of being pre-trained on large textual corpora.

[2]The construction of the template sentences based on entity type is a one-time manual step.

[3]This category information is provided in the InfoTabS and TabFact datasets. For other datasets, it can be inferred easily by clustering over the keys of the training tables.

Recently, Andreas (2020) and Pruksachatkun et al. (2020) showed that we can pre-train models on specific tasks to incorporate such implicit knowledge. Eisenschlos et al. (2020) use pre-training on synthetic data to improve the performance on the TabFact dataset. Inspired by these, we first train our model on the large, diverse and *human-written* MultiNLI dataset. Then, we fine tune it to the InfoTabS task. Pre-training with MultiNLI data exposes the model to diverse lexical constructions. Furthermore, it increases the training data size by 433K (MultiNLI) example pairs. This makes the representation better tuned to the NLI task, thereby leading to better generalization.

**Distracting Rows Removal (DRR)** Not all premise table rows are necessary to reason about a given hypothesis. In our example, for the hypotheses H1 and H2, the row corresponding to the key *No. of listings* is sufficient to decide the label for the hypothesis. The other rows are an irrelevant distraction. Further, as a practical concern, when longer tables are encoded into sentences as described above, the resulting number of tokens is more than the input size restrictions of existing models, leading to useful rows potentially being cropped. Appendix F shows one such example on the InfoTabS. Therefore, it becomes important to prune irrelevant rows.

To identify relevant rows, we employ a simplified version of the alignment algorithm used by Yadav et al. (2019, 2020) for retrieval in reading comprehension.

First, every word in the hypothesis sentence is aligned with the most similar word in the table sentences using cosine similarity. We use fastText (Joulin et al., 2016; Mikolov et al., 2018) embeddings for this purpose, which preliminary experiments revealed to be better than other embeddings. Then, we rank rows by their similarity to the hypothesis, by aggregating similarity over content words in the hypothesis. Yadav et al. (2019) used inverse document frequency for weighting words, but we found that simple stop word pruning was sufficient. We took the top $k$ rows by similarity as the pruned representative of the table for this hypothesis. The hyper-parameter $k$ is selected by tuning on a development set. Appendix B gives more details about these design choices.

**Explicit Knowledge Addition (KG explicit):** We found that adding *explicit* information to enrich

keys improves a model's ability to disambiguate and understand them. We expand the pruned table premises with contextually relevant key information from existing resources such as WordNet (definitions) or Wikipedia (first sentence, usually a definition).[4]

To find the best expansion of a key, we use the sentential form of a row to obtain the BERT embedding (on-the-fly) for its key. We also obtain the BERT embeddings of the same key from WordNet examples (or Wikipedia sentences).[5] Finally, we concatenate the WordNet definition (or the Wikipedia sentence) corresponding to the highest key embedding similarity to the table. As we want the contextually relevant definition of the key, we use the BERT embeddings rather than non-contextual ones (e.g., fastText). For example, the key *volume* can have different meanings in various contexts. For our example, the contextually best definition is *"In capital markets,* ***volume****, is the total number of a security that was traded during a given period of time."* rather than the other definition *"In thermodynamics, the* ***volume*** *of a system is an extensive parameter for describing its thermodynamic state."*.

## 3 Experiment and Analysis

Our experiments are designed to study the research question: *Can today's large pre-trained models exploit the information sources described in §2 to better reason about tabular information?*

### 3.1 Experimental setup

**Datasets** Our experiments uses InfoTabS, a tabular inference dataset from Gupta et al. (2020). The dataset is heterogeneous in the types of tables and keys, and relies on background knowledge and common sense. Unlike the TabFact dataset (Chen et al., 2019), it has all three inference labels, namely entailment, contradiction and neutral. Importantly, for the purpose of our evaluation, it has three test sets. In addition to the usual development set and the test set (called $\alpha_1$), the dataset has two adversarial test sets: a contrast set $\alpha_2$ that is lexically similar to $\alpha_1$, but with minimal changes in the hypotheses and flip entail-contradict label, and a zero-shot set $\alpha_3$ which has long tables from different domains with little key overlap with the training set.

**Models** For a fair comparison with earlier baselines, we use RoBERTa-large (RoBERTa$_L$) for all our experiments. We represent the premise table by converting each table row into a sentence, and then appending them into a paragraph, i.e. the *Para* representation of Gupta et al. (2020).

**Hyperparameters Settings**[6] For the distracting row removal (+DRR) step, we have a hyperparameter $k$. We experimented with $k \in \{2, 3, 4, 5, 6\}$, by predicting on +DRR development premise on model trained on orignal training set (i.e. BPR), as shown in Table 1. The development accuracy increases significantly as $k$ increases from 2 to 4 and then from 4 to 6, increases marginally ( 1.5% improvement). Since our goal is to remove distracting rows, we use the lowest hyperparameter with good performance i.e. $k = 4$.[7].

| Train | Dev | k = 2 | k = 3 | k = 4 | k = 5 | k = 6 |
|---|---|---|---|---|---|---|
| BPR | DRR | 71.72 | 74.83 | 77.50 | 78.50 | 79.00 |

Table 1: Dev accuracy on increasing hyperparameter $k$.

### 3.2 Results and Analysis

Table 2 shows the results of our experiments.

| Premise | Dev | $\alpha_1$ | $\alpha_2$ | $\alpha_3$ |
|---|---|---|---|---|
| Human | **79.78** | **84.04** | **83.88** | **79.33** |
| Para | 75.55 | 74.88 | 65.55 | 64.94 |
| BPR | 76.42 | 75.29 | 66.50 | 64.26 |
| +KG implicit | **79.57** | 78.27 | 71.87 | 66.77 |
| +DRR | 78.77 | 78.13 | 70.90 | 68.98 |
| +KG explicit | 79.44 | **78.42** | **71.97** | **70.03** |

Table 2: Accuracy with the proposed modifications on the Dev and test sets. Here, + represents the change with respect to the previous row. Reported numbers are the average over three random seed runs with standard deviation of 0.33 (+KG explicit), 0.46 (+DRR), 0.61 (+KG implicit), 0.86 (BPR), over all sets. All improvements are statistically significant with $p < 0.05$, except $\alpha_1$ for BPR representation w.r.t to Para (Original). Here the Human and Para results are taken from Gupta et al. (2020).

[4]Usually multi-word keys are absent in WordNet, in this case we use Wikipedia. The WordNet definition of each word in the key is used if the multi-word key is absent in Wikipedia.

[5]We prefer using WordNet examples over definition for BERT embedding because (a) an example captures the context in which key is used, and (b) the definition may not always contain the key tokens.

[6]Appendix C has more details about hyperparameters.

[7]Indeed, the original InfoTabs work points out that no more than four rows in a table are needed for any hypothesis.

**BPR** As shown in Table 2, with BPR, we observe that the RoBERTa$_L$ model improves performance on all dev and test sets except $\alpha_3$. There are two main reasons behind this poor performance on $\alpha_3$.

First, the zero-shot $\alpha_3$ data includes unseen keys. The number of keys common to $\alpha_3$ and the training set is 94, whereas for, dev, $\alpha_1$ and $\alpha_2$ it is 334, 312, and 273 respectively (i.e., 3-5 times more). Second, despite being represented by better sentences, due to the input size restriction of RoBERTa$_L$ some relevant rows are still ignored.

**KG implicit** We observe that *implicit* knowledge addition via MNLI pre-training helps the model reason and generalize better. From Table 2, we can see significant performance improvement in the dev and all three test sets.

**DRR** This leads to significant improvement in the $\alpha_3$ set. We attribute this to two primary reasons: First, $\alpha_3$ tables are longer (13.1 keys per table on average, vs. 8.8 keys on average in the others), and DRR is important to avoid automatically removing keys from the bottom of a table due to the limitations in RoBERTa$_L$ model's input size. Without these relevant rows, the model incorrectly predicts the neutral label. Second, $\alpha_3$ is a zero-shot dataset and has significant proportion of unseen keys which could end up being noise for the model. The slight decrease in performance on the dev, $\alpha_1$ and $\alpha_2$ sets can be attributed to model utilising spurious patterns over irrelevant keys for prediction.[8] We validated this experimentally by testing the original premise trained model on the DRR test tables. Table 5 in the Appendix C shows that without pruning, the model focuses on irrelevant rows for prediction.

**KG explicit** With *explicit* contextualized knowledge about the table keys, we observe a marginal improvement in dev, $\alpha_1$ test sets and a significant performance gain on the $\alpha_2$ and $\alpha_3$ test sets. Improvement in the $\alpha_3$ set shows that adding external knowledge helps in the zero-shot setting. With $\alpha_2$, the model can not utilize spurious lexical correlations[9] due to its adversarial nature, and is forced to use the relevant keys in the premise tables, thus adding explicit information about the key improves performance more for $\alpha_2$ than $\alpha_1$ or dev. Appendix F shows some qualitative examples.

### 3.3 Ablation Study

We perform an ablation study as shown in table 3, where instead of doing all modification sequentially one after another (+), we do only one modification at a time to analyze its effects.

Through our ablation study we observe that: (a) **DRR** improves performance on the dev, $\alpha_1$, and $\alpha_2$ sets, but slightly degrades it on the $\alpha_3$ set. The drop in performance on $\alpha_3$ is due to spurious artifact deletion as explained in details in Appendix E. (b) **KG explicit** gives performance improvement in all sets. Furthermore, there is significant boost in performance of the adversarial $\alpha_2$ and $\alpha_3$ sets.[10] (c) Similarly, **KG implicit** shows significant improvement in all test sets. The large improvements on the adversarial sets $\alpha_2$ and $\alpha_3$ sets, suggest that the model can now reason better. Although, implicit knowledge provides most performance gain, all modifications are needed to obtain the best performance for all sets (especially on the $\alpha_3$ set).[11]

| Premise | Dev | $\alpha_1$ | $\alpha_2$ | $\alpha_3$ |
|---|---|---|---|---|
| Para | 75.55 | 74.88 | 65.55 | 64.94 |
| DRR | 76.39 | 75.78 | 67.22 | 64.88 |
| KG explicit | 77.16 | 75.38 | 67.88 | 65.50 |
| KG implicit | **79.06** | **78.44** | **71.66** | **67.55** |

Table 3: Ablation results with individual modifications.

## 4 Comparison with Related Work

Recently, there have been many papers which study several NLP tasks on semi-structured tabular data. These include tabular NLI and fact verification tasks such as TabFact (Chen et al., 2019), and InfoTabS (Gupta et al., 2020), various question answering and semantic parsing tasks (Pasupat and Liang, 2015; Krishnamurthy et al., 2017; Abbas et al., 2016; Sun et al., 2016; Chen et al., 2020; Lin et al., 2020, *inter alia*), and table-to-text generation and its evaluation (e.g., Parikh et al., 2020; Radev et al., 2020). Several, models for better representation of tables such as TAPAS (Herzig

[8] Performance drop of dev and $\alpha_2$ is also marginal i.e. (dev: 79.57 to 78.77, $\alpha_1$: 78.27 to 78.13, $\alpha_2$: 71.87 to 70.90), as compared to InfoTabS WMD-top3 i.e (dev: 75.5 to 72.55,$\alpha_1$: 74.88 to 70.38, $\alpha_2$: 65.44 to 62.55), here WMD-top3 performance numbers are taken from Gupta et al. (2020).

[9] The hypothesis-only baseline for $\alpha_2$ is 48.5% vs. $\alpha_1$: 60.5 % and dev: 60.5 % (Gupta et al., 2020)

[10] The KG explicit step is performed only for relevant keys (after DRR).

[11] We show in Appendix D, Table 6, that implicit knowledge addition to a non-sentential table representation i.e. Struc (Chen et al., 2019; Gupta et al., 2020) leads to performance improvement as well.

et al., 2020), TaBERT (Yin et al., 2020), and TabStruc (Zhang et al., 2020) were recently proposed. Yu et al. (2018, 2020) and Eisenschlos et al. (2020) study pre-training for improving tabular inference, similar to our MutliNLI pre-training.

The proposed modifications in this work are simple and intuitive. Yet, existing table reasoning papers have not studied the impact of such input modifications. Furthermore, much of the recent work focuses on building sophisticated neural models, without explicit focus on how these models (designed for raw text) adapt to the tabular data. In this work, we argue that instead of relying on the neural network to *"magically"* work for tabular structures, we should carefully think about the representation of semi-structured data, and the incorporation of both implicit and explicit knowledge into neural models. Our work highlights that simple pre-processing steps are important, especially for better generalization, as evident from the significant improvement in performance on adversarial test sets with the same RoBERTa models. We recommend that these pre-processing steps should be standardized across table reasoning tasks.

## 5 Conclusion & Future Work

We introduced simple and effective modifications that rely on introducing additional knowledge to improve tabular NLI. These modifications governs what information is provided to a tabular NLI and how the given information is presented to the model. We presented a case study with the recently published InfoTabS dataset and showed that our proposed changes lead to significant improvements. Furthermore, we also carefully studied the effect of these modifications on the multiple test-sets, and why a certain modification seems to help a particular adversarial set.

We believe that our study and proposed solutions will be valuable to researchers working on question answering and generation problems involving both tabular and textual inputs, such as tabular/hybrid question answering and table-to-text generation, especially with difficult or adversarial evaluation. Looking ahead, our work can be extended to include explicit knowledge for hypothesis tokens as well. To increase robustness, we can also integrate structural constraints via data augmentation through NLI training. Moreover, we expect that structural information such as position encoding could also help better represent tables.

## Acknowledgements

We thank members of the Utah NLP group for their valuable insights and suggestions at various stages of the project; and reviewers their helpful comments. We also thank the support of NSF grants #1801446 (SATC) and #1822877 (Cyberlearning) and a generous gift from Verisk Inc.

## References

Faheem Abbas, M. K. Malik, M. Rashid, and Rizwan Zafar. 2016. Wikiqa — a question answering system on wikipedia using freebase, dbpedia and infobox. *2016 Sixth International Conference on Innovative Computing Technology (INTECH)*, pages 185–193.

Jacob Andreas. 2020. Good-enough compositional data augmentation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7556–7566, Online. Association for Computational Linguistics.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A Large Annotated Corpus for Learning Natural Language Inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*.

Wenhu Chen, Hongmin Wang, Jianshu Chen, Yunkai Zhang, Hong Wang, Shiyang Li, Xiyou Zhou, and William Yang Wang. 2019. Tabfact: A large-scale dataset for table-based fact verification. In *International Conference on Learning Representations*.

Wenhu Chen, Hanwen Zha, Zhiyu Chen, Wenhan Xiong, Hong Wang, and William Wang. 2020. Hybridqa: A dataset of multi-hop question answering over tabular and textual data. *Findings of EMNLP 2020*.

Ido Dagan, Dan Roth, Mark Sammons, and Fabio Massimo Zanzotto. 2013. Recognizing textual entailment: Models and applications. *Synthesis Lectures on Human Language Technologies*, 6(4):1–220.

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*.

Julian Eisenschlos, Syrine Krichene, and Thomas Mueller. 2020. Understanding tables with intermediate pre-training. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pages 281–296.

Vivek Gupta, Maitrey Mehta, Pegah Nokhiz, and Vivek Srikumar. 2020. INFOTABS: Inference on tables as

semi-structured data. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2309–2324, Online. Association for Computational Linguistics.

Jonathan Herzig, Pawel Krzysztof Nowak, Thomas Müller, Francesco Piccinno, and Julian Eisenschlos. 2020. TaPas: Weakly supervised table parsing via pre-training. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4320–4333, Online. Association for Computational Linguistics.

Armand Joulin, Edouard Grave, Piotr Bojanowski, Matthijs Douze, Hérve Jégou, and Tomas Mikolov. 2016. Fasttext.zip: Compressing text classification models. *arXiv preprint arXiv:1612.03651*.

Jayant Krishnamurthy, Pradeep Dasigi, and Matt Gardner. 2017. Neural semantic parsing with type constraints for semi-structured tables. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1516–1526.

Xi Victoria Lin, Richard Socher, and Caiming Xiong. 2020. Bridging textual and tabular data for cross-domain text-to-sql semantic parsing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pages 4870–4888.

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 preprint arXiv:1907.11692*.

Tomas Mikolov, Edouard Grave, Piotr Bojanowski, Christian Puhrsch, and Armand Joulin. 2018. Advances in pre-training distributed word representations. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*.

Ankur P Parikh, Xuezhi Wang, Sebastian Gehrmann, Manaal Faruqui, Bhuwan Dhingra, Diyi Yang, and Dipanjan Das. 2020. ToTTo: A controlled table-to-text generation dataset. In *Proceedings of EMNLP*.

Panupong Pasupat and Percy Liang. 2015. Compositional semantic parsing on semi-structured tables. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1470–1480.

Yada Pruksachatkun, Jason Phang, Haokun Liu, Phu Mon Htut, Xiaoyi Zhang, Richard Yuanzhe Pang, Clara Vania, Katharina Kann, and Samuel R Bowman. 2020. Intermediate-task transfer learning with pretrained models for natural language understanding: When and why does it work? *arXiv preprint arXiv:2005.00628*.

Dragomir Radev, Rui Zhang, Amrit Rau, Abhinand Sivaprasad, Chiachun Hsieh, Nazneen Fatema Rajani, Xiangru Tang, Aadit Vyas, Neha Verma, Pranav Krishna, et al. 2020. Dart: Open-domain structured data record to text generation. *arXiv preprint arXiv:2007.02871*.

Huan Sun, Hao Ma, Xiaodong He, Wen-tau Yih, Yu Su, and Xifeng Yan. 2016. Table cell search for question answering. In *Proceedings of the 25th International Conference on World Wide Web*, pages 771–782.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A Broad-Coverage Challenge Corpus for Sentence Understanding through Inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

Vikas Yadav, Steven Bethard, and Mihai Surdeanu. 2019. Alignment over heterogeneous embeddings for question answering. 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 2681–2691.

Vikas Yadav, Steven Bethard, and Mihai Surdeanu. 2020. Unsupervised alignment-based iterative evidence retrieval for multi-hop question answering. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4514–4525, Online. Association for Computational Linguistics.

Pengcheng Yin, Graham Neubig, Wen-tau Yih, and Sebastian Riedel. 2020. TaBERT: Pretraining for joint understanding of textual and tabular data. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8413–8426, Online. Association for Computational Linguistics.

Tao Yu, Chien-Sheng Wu, Xi Victoria Lin, Bailin Wang, Yi Chern Tan, Xinyi Yang, Dragomir Radev, Richard Socher, and Caiming Xiong. 2020. Grappa: Grammar-augmented pre-training for table semantic parsing. *arXiv preprint arXiv:2009.13845*.

Tao Yu, Rui Zhang, Kai Yang, Michihiro Yasunaga, Dongxu Wang, Zifan Li, James Ma, Irene Li, Qingning Yao, Shanelle Roman, et al. 2018. Spider: A large-scale human-labeled dataset for complex and cross-domain semantic parsing and text-to-sql task. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3911–3921.

Hongzhi Zhang, Yingyao Wang, Sirui Wang, Xuezhi Cao, Fuzheng Zhang, and Zhongyuan Wang. 2020. Table fact verification with structure-aware transformer. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1624–1629, Online. Association for Computational Linguistics.

## A BPR Templates

Here, we are listing down some of the diverse example templates we have framed.

- For the table category *Bus/Train Lines* and key *Disabled access* with BOOL value YES, follow template: "*t* has *k*."

  **Orignal Premise Sentence** *"The Disabled access of Tukwila International Boulevard Station are Yes."*

  **BPR Sentence** *"Tukwila International Boulevard Station has Disabled access."*

- For the table category *Movie* and key *Box office* with MONEY type, follow template: "In the *k*, *t* made *v*."

  **Orignal Premise Sentence** *"The Box office of Brokeback Mountain are $178.1 million."*

  **BPR Sentence** *"In the Box office, Brokeback Mountain made $178.1 million."*

- For the table category *City* and key *Total* with CARDINAL type, follow template: "The *k* area of *t* is *v*."

  **Orignal Premise Sentence** *"The Total of Cusco are 435,114."*

  **BPR Sentence** *"The Total area of Cusco is 435,114."*

- For the table category *Painting* and key *Also known as*, follow template: "The *k* area of *t* is *v*."

  **Orignal Premise Sentence** *"The Also known as of Et in Arcadia ego are Les Bergers d'Arcadie."*

  **BPR Sentence** *"Et in Arcadia ego is Also known as Les Bergers d'Arcadie."*

- For the table category *Person* and key *Died* with DATE type , follow template: "*t k* on *v*."

  **Orignal Premise Sentence** *"The Died of Jesse Ramsden are November 1800 (1800-11-05) (aged 65) Brighton, Sussex."*

  **BPR Sentence** *"Jesse Ramsden Died on 5 November 1800 (1800-11-05) (aged 65) Brighton, Sussex."*

## B DRR: fastText and Binary weighting

**fastText:** For word representation, (Yadav et al., 2019) have used BERT and Glove embeddings. In our case, we prefer to use fastText word embeddings over Glove because fastText embedding uses sub-word information which helps in capturing different variations of the context words. Furthermore, fastText embeddings is also as better choice than BERT for our task because 1. Firstly, we are embedding single sentential form of diverse rows instead of longer context similar paragraphs, 2. Secondly, all words (especially keys) of the rows across all the tables are used only in one context, whereas BERT is useful when same word is used with different contexts across paragraphs, 3. Thirdly, in all tables, the number sentences to select from is bounded by maximum rows in the table, which is a small number (8.8 in train, dev, $\alpha_1$, $\alpha_2$ and 13.1 in $\alpha_3$), and 4. Lastly, using fastText is much faster to compute than BERT for obtaining embeddings.

**Binary weighting:** Since, we are embedding single sentential form of diverse rowsinstead of longer context related paragraphs, we found that using binary weighting 0 for stop words and 1 for others is more effective than the idf weighting, which is useful only for longer paragraph context with several lexical terms.

## C Hyperparameters $k$ vs test-sets accuracy

We also trained a model both train and tested on the DRR table premise for increasing values of the hyper parameter $k$, as shown in Table 1. We also test the model trained on the entire para on pruned para with increasing value of hyperparameters $k \in \{2, 3, 4, 5, 6\}$ for the test sets $\alpha_1$, $\alpha_2$, and $\alpha_3$. In all cases, except $\alpha_3$, the performance with larger $k$ is better. The increase in performance, even with $k > 4$, shows that the model is using more then required keys for prediction. Thus, the model is utlising the spurious pattern in irrelevant rows for the prediction.

| Train | Dev | k=2 | k=3 | k=4 | k=5 | k=6 |
|---|---|---|---|---|---|---|
| +DRR | +DRR | 77.61 | 77.94 | 78.16 | 78.38 | 79.00 |
| BPR | +DRR | 71.72 | 74.83 | 77.50 | 78.50 | 79.00 |

Table 4: Dev accuracy with increasing hyper parameter $k$ trained with both BPR and +DRR table.

| $k$ | $\alpha_1$ | $\alpha_2$ | $\alpha_3$ |
|---|---|---|---|
| 2 | 71.44 | 67.33 | 64.83 |
| 3 | 75.05 | 69.33 | 67.33 |
| 4 | 77.72 | 69.83 | 68.22 |
| 5 | 77.77 | 70.28 | **69.28** |
| 6 | **77.77** | **70.77** | 69.22 |

Table 5: Accuracy of model trained with orignal table but tested with DRR table with increasing hyper parameter $k$ on all test sets.

## D TabFact Representation Experiment

Table 6 implicit knowledge addition effect on non-para *Struc* representation i.e. a key value linearize representation as "key k : value v", rows separated by semicolon ";" (Gupta et al., 2020; Chen et al., 2019). Here too the implicit knowledge addition leads to improvement in performance on all the sets.

| Premise | Dev | $\alpha_1$ | $\alpha_2$ | $\alpha_3$ |
|---|---|---|---|---|
| Struc | 77.61 | 75.06 | 69.02 | 64.61 |
| + KG implicit | **79.55** | **78.66** | **72.33** | **70.44** |

Table 6: Accuracy on InfoTabS data for Struc representation of Tables. Here, + represents the change with respect to the previous row.

## E Artifacts and Model Predictions

In Table 7 we show percentage of example which were corrected after modification and vice versa. Surprisingly, there is a small percentage of examples which are predicted correctly earlier with original premise (Para) but predicted wrongly after all the modifications (Mod), although such examples are much lesser than opposite case. We suspect that earlier model was also relying on spurious pattern (artifacts) for correct prediction on these examples earlier, which are now corrupted after the proposed modifications. Hence, the new model struggle to predict correctly on such examples.

| Para | Mod | Dev | $\alpha_1$ | $\alpha_2$ | $\alpha_3$ |
|---|---|---|---|---|---|
| ✓ | × | 6.77 | 7.83 | 9.27 | 10.01 |
| × | ✓ | **10.94** | **12.55** | **14.33** | **16.05** |

Table 7: Correct vs Incorrect Predictions for Para model (Gupta et al., 2020) and the model after the modifcations (Mod).

In the next section F, we also shows qualitative examples, where modification helps model predict correctly. We also provide some examples via distracting row removal modification, where model fails after modification.

## F Qualitative Examples

In this section, we provide examples where model is able to predict well after the proposed modifications. We also provide some examples, where model struggles to make the correct prediction after distracting row removal (DRR) modification.

### F.1 BPR

**Original Premise** The Birth name of Eva Mendes are Eva de la Caridad Méndez. Eva Mendes was Born on March 5, 1974 (1974-03-05) (age 44) Miami, Florida, U.S.. The Occupation of Eva Mendes are Actress, model, businesswoman. The Years active of Eva Mendes are 1998 - present. The Partner(s) of Eva Mendes are Ryan Gosling (2011 - present). The Children of Eva Mendes are 2.

**Better Paragraph Premise** Eva Mendes is a person. The birth name of Eva Mendes is Eva de la Caridad Méndez. Eva Mendes was born on March 5, 1974 (1974-03-05) (age 44) Miami, Florida, U.S.. The occupation of Eva Mendes is Actress, model, businesswoman. The years active of Eva Mendes was on 1998 - present. The partner(s) of Eva Mendes is Ryan Gosling (2011 - present). The number of children of Eva Mendes are 2.

**Hypothesis** Eva Mendes has two children.

| Premise | Label |
|---|---|
| Human Label (Gold) | Entailed |
| Orignal Premise | Neutral |
| +BPR | Entailed |

Table 8: Prediction after BPR. Here, + represents the change with respect to the previous row.

**Result and Explanation** In this example from $\alpha_2$, the model predicts Neutral for this hypothesis with orignal premise. However, forming better sentences by adding the *"number of children are 2"* (highlighted as green) in case of CARDINAL type for the category PERSON helps the model understand the relation and reasoning behind the children and the number two and arrive at the correct prediction of entailment.

### F.2 KG implicit

**Original Premise** Janet Leigh is a person. Janet Leigh was born as Jeanette Helen Morrison (1927-07-06) July 6, 1927 Merced, California, U.S. Janet Leigh died on October 3, 2004 (2004-10-03) (aged 77) Los Angeles, California, U.S.. The resting place of Janet Leigh is Westwood Village Memorial Park Cemetery. The alma mater of Janet Leigh is University of the Pacific. The occupation of Janet Leigh are Actress, singer, dancer, author. The years active of Janet Leigh was on 1947-2004. The political party of Janet Leigh is Democratic. The spouse(s) of Janet Leigh are John Carlisle (m. 1942; annulled 1942), Stanley Reames (m. 1945; div. 1949), Tony Curtis (m. 1951; div. 1962), Robert Brandt (m. 1962). The children of Janet Leigh are Kelly Curtis, Jamie Lee Curtis.

**Hypothesis A** Janet Leigh's career spanned **over** 55 years long.

**Hypothesis B** Janet Leigh's career spanned **under** 55 years long.

| Premise | Label |
|---|---|
| Human Label (Gold) | Entailed |
| Orignal Premise | Entailed |
| + KG implicit | Entailed |

Table 9: Prediction on Hypothesis A. Here, + represents the change with respect to the previous row

| Premise | Label |
|---|---|
| Human Label (Gold) | Contradiction |
| Orignal Premise | Entailed |
| + KG implicit | Contradiction |

Table 10: Prediction on Hypothesis B (from $\alpha_2$). Here, + represents the change with respect to the previous row

**Result and Explanation** In this example from $\alpha_2$, the model without implicit knowledge and the model with implicit knowledge addition predict the correct label on the Hypothesis A. However for Hypothesis B which is an example from $\alpha_2$, and originally generated by replacing the word "**over**" to word "**under**" in the Hypothesis A and flipping gold label from entail to contradiction, the ealier model which is using artifacts over lexical patterns arrive to predict the original wrong label entail instead of contradiction. On adding implicit knowledge while training, the model is now able to reason rather than relying on artifacts and correctly predicts contradiction. Note, that both hypothesis A and hypothesis B require exactly same reasoning for inference i.e. they are equally hard.

### F.3 KG explicit

### F.4 DRR

**Original Premise** The pronunciation of Fluorine are (FLOOR-een, -in, -yn) and (FLOR-een, -in, -yn). The allotropes of Fluorine is alpha, beta. The appearance of Fluorine is gas: very pale yellow , liquid: bright yellow , solid: alpha is opaque, beta is transparent. The standard atomic weight are, std(f) of Fluorine is 18.998403163(6). The atomic number (z) of Fluorine is 9. The group of Fluorine is group 17 (halogens). The period of Fluorine is period 2. The block of Fluorine is p-block. The element category of Fluorine is Reactive nonmetal. The electron configuration of Fluorine is [He] 2s 2 2p 5. The electrons per shell of Fluorine is 2, 7. The phase at stp of Fluorine is gas. The melting point of Fluorine is (F-2) 53.48 K (-219.67 °C, -363.41 °F). The boiling point of Fluorine is (F 2 ) 85.03 K (-188.11 °C, -306.60 °F). The density (at stp) of Fluorine is 1.696 g/L. The when liquid (at b.p.) of Fluorine is 1.505 g/cm 3. The triple point of Fluorine is 53.48 K, 90 kPa. The critical point of Fluorine is 144.41 K, 5.1724 MPa. The heat of vaporization of Fluorine is 6.51 kJ/mol. The molar heat capacity of Fluorine is C p : 31 J/(mol·K) (at 21.1 °C) , C v : 23 J/(mol·K) (at 21.1 °C). The oxidation states of Fluorine is -1 (oxidizes oxygen). The electronegativity of Fluorine is Pauling scale: 3.98. Fluorine was ionization energies on 1st: 1681 kJ/mol, 2nd: 3374 kJ/mol, 3rd: 6147 kJ/mol, (more). The covalent radius of Fluorine is 64 pm. The van der waals radius of Fluorine is 135 pm. The natural occurrence of Fluorine is primordial. The thermal conductivity of Fluorine is 0.02591 W/(m·K). The magnetic ordering of Fluorine is diamagnetic (-1.2×10 -4 ). The cas number of Fluorine is 7782-41-4. The naming of Fluorine is after the mineral fluorite, itself named after Latin fluo (to flow, in smelting). The discovery of Fluorine is André-Marie Ampère (1810). The first isolation of Fluorine is Henri Moissan (June 26, 1886). The named by of Fluorine is Humphry Davy.

**Distracting Row Removal (DRR)** The first isolation of Fluorine is Henri Moissan (June 26, 1886). The group of Fluorine is group 17 (halogens). The discovery of Fluorine is André-Marie Ampère (1810). Fluorine was ionization energies on 1st: 1681 kJ/mol, 2nd: 3374 kJ/mol, 3rd: 6147 kJ/mol, (more).

**Hypothesis** Flourine was discovered in the 18th century.

| Premise | Label |
|---|---|
| Human Label (Gold) | Contradiction |
| Orignal Premise | Neutral |
| +DRR | Contradiction |

Table 11: Prediction after DRR. Here, + represents the change with respect to the previous row.

**Result and Explanation** In this example from the $\alpha_3$ set, removing distracting rows (sentence except the one in green and blue) definitely helps as there are irrelevant distracting noise and also make premise paragraph long beyond BERT maximum tokenization limits. Before DRR is applied, the model predicts neutral due to a) distracting rows and b) required information i.e. relevant keys-rows highlighted as green being removed due to maximum tokenization limitation (it's second last sentence). However, after DRR, the prune information retained is only the relevant keys highlighted as green and thus the model is able to predict the correct label.

**Negative Example** In some examples distracting row removal for DRR remove an relevant rows and hence the model failed to predict correctly on the DRR premise, as shown below:

**Original Premise** Et in Arcadia ego is a painting. Et in Arcadia ego is also known as Les Bergers d'Arcadie. The artist of Et in Arcadia ego is Nicolas Poussin. The year of Et in Arcadia ego is 1637 - 1638. The medium of Et in Arcadia ego is oil on canvas. The dimensions of Et in Arcadia ego is 87 cm 120 cm (34.25 in 47.24 in). The location of Et in Arcadia ego is Musee du Louvre.

**Distracting Row Removal (DRR)** Et in Arcadia ego is a painting. The artist of Et in Arcadia ego is Nicolas Poussin. The medium of Et in Arcadia ego is oil on canvas. The dimensions of Et in Arcadia ego is 87 cm 120 cm (34.25 in 47.24 in).

**Hypothesis** The art piece Et in Arcadia ego is **stored** in the **United Kingdom**

| Premise | Label |
|---|---|
| Human Label (Gold) | Contradiction |
| Orignal Premise | Contradiction |
| +DRR | Neutral |

Table 12: Prediction after DRR. Here, + represents the change with respect to the previous row.

**Result and Explanation** In this example from the Dev set, the DRR technique used removes the required key "*Location*" (highlighted in red) from the para representation. Hence, the model here predicts neutral as the information regarding where the painting is stored i.e. "*Location*" is removed in the DRR, which the model require for making the correct inference. While in original para, this information is still present and the model is able to arrive at the correct label. Another interesting observation is RoBERTa$_L$ knows Musee du Louvre is a museum in the United Kingdom, showing sign of world-knowledge.

**Negative Example** In another negative examples distracting row removal for DRR got the relevant rows correct but still the model failed to predict correct label due to spurious correlation, as shown below:

**Original Premise** Idiocracy is a movie. Idiocracy was directed by Mike Judge. Idiocracy was produced by Mike Judge, Elysa Koplovitz, Michael Nelson. Idiocracy was written by Etan Cohen, Mike Judge. Idiocracy was starring Luke Wilson, Maya Rudolph, Dax Shepard. Idiocracy was music by Theodore Shapiro. The cinematography of Idiocracy was by Tim Suhrstedt. Idiocracy was edited by David Rennie. The production company of Idiocracy is Ternion. Idiocracy was distributed by 20th Century Fox. The release date of Idiocracy is September 1, 2006. The running time of Idiocracy is 84 minutes. The country of Idiocracy is United States. The language of Idiocracy is English. The budget of Idiocracy is $2-4 million. In the box office, Idiocracy made $495,303 (worldwide).

**Distracting Row Removal (DRR)** Idiocracy was directed by Mike Judge. Idiocracy was produced by Mike Judge, Elysa Koplovitz, Michael Nelson. Idiocracy was written by Etan Cohen, Mike Judge. Idiocracy was edited by David Rennie.

**Hypothesis** Idiocracy was directed and written by the **same** person.

| Premise | Label |
|---|---|
| Human Label (Gold) | Entailed |
| Orignal Premise | Entailed |
| +DRR | Neutral |

Table 13: Prediction after DRR. Here, + represents the change with respect to the previous row.

**Result and Explanation** In this example from the Dev set, the model before DRR predicts the correct label but however on DRR, it predicts incorrect label of neutral. Despite the fact that both the relevant rows require for inference (highlighted in green) is present after DRR. This shows, that the model is looking at more keys than required in the initial case, which are eliminated in the DRR, which force the model to change it prediction. Thus, model is utilising spurious correlation from irrelevant rows to predict the label.

**Orignal Premise** Julius Caesar was born on 12 or 13 July 100 BC Rome. Julius Caesar died on 15 March 44 BC (aged 55) Rome. The resting place of Julius Caesar is Temple of Caesar, Rome. The spouse(s) of Julius Caesar are Cornelia (84-69 BC; her death), Pompeia (67-61 BC; divorced), Calpurnia (59-44 BC; his death).

**Orignal Premise + KG explicit** Julius Caesar died on 15 March 44 BC (aged 55) Rome. **The resting place of Julius Caesar is Temple of Caesar, Rome.** Julius Caesar was born on 12 or 13 July 100 BC Rome. The spouse(s) of Julius Caesar are Cornelia (84-69 BC; her death), Pompeia (67-61 BC; divorced), Calpurnia (59-44 BC; his death). KEY: Died is defined as pass from physical life and lose all bodily attributes and functions necessary to sustain life . KEY: Resting place is defined as a cemetery or graveyard is a place where the remains of dead people are buried or otherwise interred . KEY: Born is defined as british nuclear physicist (born in germany) honored for his contributions to quantum mechanics (1882-1970) . KEY: Spouse is defined as a spouse is a significant other in a marriage, civil union, or common-law marriage .

**Hypothesis** Julius Caesar was buried in Rome.

| Model | Label |
|---|---|
| Human Label (Gold) | Entailed |
| Original Premise | Neutral |
| + KG explicit | Entailed |

Table 14: Prediction after KG explicit addition. Here, + represents the change with respect to the previous row.

**Result and Explanation** In this example from $\alpha_2$, the model without explicit knowledge predicts neutral for the hypothesis as it is not able to infer that **resting place** is where people are **buried**, so it predicts neutral as it implicitly lack buried key understanding. On explicit KG addition (highlighted as blue+ green), we add the definition of resting place to be the place where remains of the dead are buried (highlighted as green). Now the model uses this extra information (highlighted as green) plus the original key related to death (highlighted in **bold**) to correctly infer that the statement Caesar is buried in Rome is entailed.