Re-TACRED: A New Relation Extraction Dataset

George Stoica
Carnegie Mellon University
5000 Forbes Ave,
Pittsburgh, PA 15213
gis@cs.cmu.edu

Emmanouil Antonios Platanios
Microsoft Semantic Machines
1 Microsoft Way,
Redmond, WA 98052
emplata@microsoft.com

Barnabás Póczos
Carnegie Mellon University
5000 Forbes Ave,
Pittsburgh, PA 15213
bapoczos@cs.cmu.edu

Abstract

TACRED [44] is one of the largest and most widely used sentence-level relation extraction datasets. However, a recent study [2] suggested that the dataset may have substantial quality issues. In this paper, we address these concerns by: (i) comprehensively studying the whole TACRED dataset, (ii) proposing and deploying an improved crowd-sourcing strategy to re-annotate the TACRED dataset, and (iii) thoroughly analyzing how correcting TACRED affects previously published results. After re-annotation, we observe that 22.1% of TACRED labels are different and that models evaluated on our revised dataset achieve an average f1-score improvement of 13%. Additionally, we publicly release our revised dataset, **Re-TACRED** to further enable reliable evaluation of relation extraction models.

1 Introduction

Many applications ranging from medical diagnostics to search engines rely on the ability to uncover relationships between seemingly disparate concepts based on existing knowledge. Relation extraction (RE) is a popular learning task aimed at extracting such relationships between concepts in plain text. For example, given the sentence "`John lives in Miami`", where "`John`" and "Miami" are the sentence subject and object respectively, the objective of an RE method is to infer the correct relation, `PERSON:LIVES_IN` , between them. TACRED [44], is one of the most widely used crowd-sourced RE datasets and consists of 106,264 sentences of varied complexity. Although just three years-old, a multitude of approaches have been proposed and evaluated using the dataset. Recently, methods have converged at $\sim$71.5% f1-score on the test data, raising the question of whether we have reached the maximum possible attainable performance on the TACRED dataset, and if so, why? [2] investigated these questions by performing a comprehensive review of the 5,000 most misclassified TACRED development and test split sentences among 49 existing RE methods. They observed that over 50% of the sentences were labeled incorrectly, leading to an average model performance improvement of over 8% after correcting these labels. However, the broader impact of their work is limited by two key factors. First, they restricted their dataset revisions to a biased small sample of TACRED. Thus, it is not clear whether their findings would be true for the whole TACRED dataset. Second, even after revision, the majority of the TACRED test split was uncorrected, making it challenging to identify if new errors made by the methods are primarily due to model capacity, data error, or a mixture of both.

In this paper, we address these shortcomings by performing a comprehensive re-annotation of the entire TACRED dataset. Our contributions can be summarized as follows. First, we deploy an improved and cost-efficient crowd-sourcing annotation strategy over the dataset. Our annotations achieve an average agreement rate of 82.3% and inter-annotator Fleiss' kappa of .77 (significantly higher than the .54 kappa achieved when TACRED was created [44]). Second, we thoroughly compare our revisions with the TACRED labels. Our revisions significantly improve model performance by an average of 13% f1-score, and reveal new types of model errors obscured by the original TACRED labels. Third, we publicly release our revised dataset labels.

4th Knowledge Representation and Reasoning Meets Machine Learning Workshop (KR2ML 2020), at NeurIPS.

2 Background

Each TACRED instance consists of a sentence and two non-overlapping contiguous spans representing a subject and an object, each with pre-assigned "types" (e.g., PERSON or CITY). For example, consider the sentence "John Doe was born in Miami." In this case, the subject is a PERSON and the object a CITY. Each instance is assigned one of 42 labels that describes the relationship between the subject and the object. These labels consist of 41 relations that describe the existence of some relationship between the two (e.g., CITY_OF_BIRTH), and a special NO_RELATION predicate to indicate the absence of a relationship. Moreover, all relations are *typed*: they only apply to a specific set of subject and object types. There are a total of 28 subject-object types pairs in TACRED.

TACRED instance labels were assigned using the Amazon Mechanical Turk (AMT) crowdsourcing platform. For each sentence, AMT workers were asked to select the appropriate label from a set of suggestions between highlighted subjects and objects. The suggestions included all labels that were compatible with the subject and object types, along with the special NO_RELATION label.

2.1 TACRED Quality

[44] assumed an acceptable level of label quality based on an observed high annotation accuracy of 93.3% from a random sample of 300 instances, and a Fleiss' Kappa of .54 over 761 randomly selected annotation pairs. However, recent work suggests that the annotation quality may be significantly lower than previously estimated. [2] manually verified the labels for the 5,000 most miss-classified sentences from the TACRED test and validation splits over 49 existing relation extraction methods via crowd-sourcing. While similar to [44], their annotation task exhibited two primary differences to improve quality. First, only workers trained in general linguistics and who passed a trial exam labeling 500 hand-picked TACRED sentences were allowed to participate. Second, they extended the label suggestion set to include predictions made by pre-trained relation extraction models. After annotation, they observed that *over 50% of TACRED labels in their sample were incorrect*. Their revised dataset improved model performance by an average of 8.1% f1-score, suggesting that TACRED model evaluation may lead to inaccurate conclusions. Moreover, their Fleiss Kappa were .80 and .87 for the validation and test sets, showing a high annotation quality.

While [2] demonstrated some of the shortcomings of the TACRED dataset, the broader impact of their work is restricted by both a small and biased sample set, and an analysis performed over a predominately uncorrected TACRED test dataset. Although correcting this small set of labels yielded significant impact on the evaluation of existing relation extraction models, it is difficult to generalize the results to the full dataset. Additionally, it is not clear if remaining model errors are due to their capability or further underlying data inaccuracies. These disadvantages raise several questions that are difficult to answer with their study. Can we design a cost-effective yet robust crowdsourced annotation task in order to correct the whole dataset and allow the research community to benefit from more accurate evaluations of novel methods? Can we expect similar performance improvements when re-annotating the whole dataset? How do model errors change under a revised dataset? These questions are difficult to answer based on the work of [2] and are our motivation for our work.

3 TACRED Revision

Labeling TACRED is challenging due to its large size and complex structure, making it difficult for crowd-sourced workers to identify the right relation among 42 choices. Thus, we initially reduce complexity by following an annotation template similar to [44, 2]. We first group TACRED sentences based on their corresponding subject and object types (e.g., "Jane loves her ring" is grouped together with sentences whose subject and object both have type PERSON), and assign each group a filtered candidate set of *type-compatible* labels (e.g., relations between people), and the special NO_RELATION label. Workers are then asked to choose the appropriate label for each sentence from the associated candidate set. However, we extend this template in three directions described below.

Wrong Type Handling. A preliminary analysis of 1,000 sentences revealed that 5% had incorrect subject and/or object type assignments (e.g., "Thomas More Law Center" tagged as PERSON instead of ORGANIZATION). This is problematic because such instances are placed in incorrect sentence groups and are assigned *type-incompatible candidate labels*. Therefore, if the types are wrong, workers would be forced to assign an incorrect label (correct relations must be type-compatible).

We address this issue in two parts. First, we merge sentence groups whose types are most confused with one another into eight "super-clusters", and define their candidate relation sets as the union of all associated sentence group candidate sets. This increases the probability that type-compatible relations exist for incorrectly-typed sentences. Second, we extend each cluster candidate set with an additional `WRONG_TYPE` relation. This enables workers to avoid selecting an incorrect label when all candidate relations are type-incompatible for sentences. We address these latter sentences by iteratively assigning them to other super-clusters until they are correctly annotated. We refer readers to Appendix A.1 for further details.

Label Definition Refinement. Similar to [44, 2], we defined many labels according to the TAC KBP[1] documentation. However, we observed that in certain cases the documentation was unintuitive and vague, confusing workers and resulting in poor annotation quality. Thus, we alter affected relation definitions to be better suited for the TACRED RE task. Overall, we refine 20 labels, and their refinements can be categorized into four groups: (i) explicitly enabling identity relationships between subjects and objects, (ii) merging significantly similar labels, (iii) relaxing challenging criteria, and (iv) enforcing label mutual-exclusivity (TACRED is a single-label RE dataset). Appendix A.2 describes each of these categories in further detail.

Quality Assurance. In order to obtain high annotation quality, we employ a two step quality assurance process similar to [22, 42] for our annotators. First, we specify three prerequisite criteria that workers must satisfy before annotating our dataset: (i) candidates must have had at least 500 previous tasks approved on AMT, (ii) have an overall approval rate $\left(\frac{\text{\# Annotations Approved}}{\text{\# Annotations Completed}}\right) \geq 95\%$, and (iii) pass custom "qualification tests" for each sentence super-cluster they annotate. The first two filters ensure that our annotators are both experienced and reliable. In conjunction, the latter exam gauges potential worker quality over our instances and *specializes/trains* them for each super-cluster annotation task. Second, following [42], for every five sentences a worker annotates, we include one control instance whose correct label is known. This enables us to track worker quality throughout annotation, and remove those who under-perform. Similar to [42], we only accept responses from annotators who correctly answer at least 80% of our control instances (separately computed for each super-cluster). On average, this eliminated approximately 10% of the annotators, and significantly improved the quality of the collected data. Note that, in aggregate we used approximately 2,000 unique control sentences for the annotation of the full TACRED dataset.

Our revised labels result in an 82.3% agreement rate and Fleiss Kappa of .77 between annotators throughout the *full dataset*, indicating high annotation quality. These metrics are significantly better than those reported by [44], which observed a Fleiss' Kappa of .54 across 761 sentences. We term the dataset resultant from these labels Re-TACRED.

4 TACRED and Re-TACRED Comparison

We additionally analyze the label distributional differences between Re-TACRED and TACRED, and examine the performance impact of Re-TACRED over TACRED. We perform our analysis using three existing relation extraction methods: PALSTM[44], C-GCN[45], and SpanBERT[17] —a State-of-the-Art model. We train all our TACRED-based models using the reported parameters by their respective contributors [44, 45, 17]. All hyperparemeter details for models trained on Re-TACRED can be found in Appendix B.1 and in our code repository[2]. Due to space restrictions, we summarize our key observations in this section, and leave further extensive details in Appendix B.

Distribution Differences. Overall, our revised labels disagree with 22.1% of TACRED sentences. Of these, 74.3% correspond to `NO_RELATION` that are switched to one of the other relations and 20.0% correspond to other relations switching to `NO_RELATION`. The remaining 5.7% correspond to switching between different non-negative relations.

Overall Impact. We present the evaluation results of the three models over TACRED and Re-TACRED in table 1. In addition, we record the improvement percentages of models evaluated on Re-TACRED have over those assessed on TACRED. All results were reported using micro-averaged f1-scores from the model with the median validation f1-score over five independent runs, as in prior literature [44, 45, 10]. Notably, we observe significant improvements across every metric for each

[1] https://tac.nist.gov/2017/KBP/index.html

[2] https://github.com/gstoica27/Re-TACRED.git

Table 1: Results for multiple RE models. We report result for TACRED obtained using our own experiments that may differ slightly from previously reported numbers. "Change %" indicates the performance difference between methods evaluated on TACRED and Re-TACRED.

Dataset	Metric	Models		
		PALSTM	**C-GCN**	**SpanBERT**
TACRED	Precision	68.1	68.5	70.1
	Recall	64.5	64.4	69.2
	F1	66.2	66.3	69.7
Re-TACRED	Precision	78.3	79.7	**84.6**
	Recall	77.6	78.5	**83.9**
	F1	77.9	79.1	**84.2**
Change %	Precision	+12.2	+11.2	**+14.5**
	Recall	+13.1	+14.1	**+14.7**
	F1	+11.7	+12.8	**+14.5**

of the three models. SpanBERT achieves the largest improvement in both f1-measure and precision by 14.5%, and a 14.7% improvement in recall. Interestingly, although PALSTM and C-GCN have similar f1-score increases, their recall and precision enhancements are complementary. C-GCN has larger recall improvement, while PALSTM displays a larger precision increase. In contrast to C-GCN and PALSTM, SpanBERT observes a larger improvement in all three metrics. These asymmetric model behavior differences indicate that improvement is not simply due to a revision offset or score scaling; instead, it is dependent on the characteristics of each model at reasoning over diverse data. In addition, these results suggest that existing models are under-evaluated on TACRED, and that their true capabilities—and performance margins—may be significantly better than reported.

Effect of Refined Labels. We observe significant performance improvements by as much as 73.5% over our four refined label categories mentioned in Section 3. Table 3 in Appendix B.3 shows the results from each of the three models. Importantly, while PALSTM and C-GCN yield complementary performances over each category on TACRED, C-GCN outperforms PALSTM on every category in Re-TACRED. Moreover, SpanBERT achieves significantly better f1-scores by at least 7.2% across every refinement category on Re-TACRED compared to TACRED. We attribute these improvements to our explicitly addressing diverse TACRED label nuances.

Effect of Non-Refined Labels. We also examine how models differ based on our *non-refined* label re-annotations. Non-refined relations are any for which we did not alter the TAC KBP relation definitions (i.e. the remaining 42-20=22 relations). Table 4 in Appendix B.4 shows our results. Overall, we observe similar trends as in our refined-labels: all methods significantly improve by as much as 9.1% f1-score on Re-TACRED compared to TACRED. Moreover, similar to the findings of [2], TACRED-trained models achieve up to 1.8% better f1-score when evaluated on Re-TACRED than on TACRED, illustrating how TACRED may be under-estimating model performance.

Re-TACRED Error Correction. In addition, we study how model errors change between TACRED and Re-TACRED. We conduct this analysis by training two separate SpanBERT instances on TACRED and Re-TACRED respectively, and evaluate both on the Re-TACRED test split. We then identify which sentences TACRED-trained SpanBERT classifies incorrectly, while Re-TACRED-trained SpanBERT answers correctly. Of these, 82.2% are due to TACRED-trained SpanBERT inferring `NO_RELATION` when the gold label is positive, 14.4% occur when the model predicts a positive relation when the correct label is negative, and the remaining 3.4% of errors arise when the method classifies the incorrect positive label. Table 5 in Appendix B.5 presents five sentences highlighting these errors.

5 Conclusion

We addressed the shortcomings of the TACRED dataset by performing a comprehensive verification of the complete dataset using crowd-sourcing. Our annotation strategy extended previous TACRED-label studies by accounting for type errors, label definition ambiguities, and additional quality control. Our results show significantly higher Fleiss' Kappa (.77) than original dataset annotations (.54), suggesting high annotation reliability. Moreover, our revised labels yield an average model improvement of 13% f1-score, and reveal new error types obscured by the original labels.

Broader Impact

Relation extraction (RE) plays a critical role in many recent innovations ranging from consumer products such as personal assistants and search engines, to enterprise operations such as automatic medical diagnostics. However, RE systems are only as powerful as their training-and-evaluation datasets. To this end, we propose a comprehensive revision addressing the shortcomings of one of the most widely used crowd-sourced and large (over 100,000 examples) RE datasets. Generated via crowd-sourcing, our new dataset exhibits significantly higher label quality than its predecessor. This higher quality translates to significantly better model performance than observed by the previous dataset, and uncovers new previously obscured model error types. Thus, our dataset provides an enhanced environment for developing RE systems by enabling informative and reliable model evaluation. In addition, our dataset provides an example for how to effectively use crowd-sourcing platforms to generate reliable data. We hope that the annotation strategies presented in this work may aid future research involving crowd-sourced data collection efforts.

References

[1] Christoph Alt, Marc Hübner, and Leonhard Hennig. Improving relation extraction by pre-trained language representations. *CoRR*, abs/1906.03088, 2019. URL `http://arxiv.org/abs/1906.03088`.

[2] Christoph Alt, Aleksandra Gabryszak, and Leonhard Hennig. Tacred revisited: A thorough evaluation of the tacred relation extraction task, 2020.

[3] Ivana Balazevic, Carl Allen, and Timothy Hospedales. TuckER: Tensor factorization for knowledge graph completion. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5185–5194, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1522. URL `https://www.aclweb.org/anthology/D19-1522`.

[4] Iz Beltagy, Kyle Lo, and Waleed Ammar. Improving distant supervision with maxpooled attention and sentence-level supervision. *CoRR*, abs/1810.12956, 2018. URL `http://arxiv.org/abs/1810.12956`.

[5] Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. Translating embeddings for modeling multi-relational data. In *Advances in Neural Information Processing Systems*, pages 2787–2795, 2013.

[6] Jun Chen, Robert Hoehndorf, Mohamed Elhoseiny, and Xiangliang Zhang. Efficient long-distance relation extraction with dg-spanbert, 2020.

[7] Rajarshi Das, Shehzaad Dhuliawala, Manzil Zaheer, Luke Vilnis, Ishan Durugkar, Akshay Krishnamurthy, Alex Smola, and Andrew McCallum. Go for a walk and arrive at the answer: Reasoning over paths in knowledge bases using reinforcement learning. In *International Conference on Learning Representations (ICLR)*, 2018.

[8] Tim Dettmers, Minervini Pasquale, Stenetorp Pontus, and Sebastian Riedel. Convolutional 2d knowledge graph embeddings. In *Proceedings of the 32th AAAI Conference on Artificial Intelligence*, pages 1811–1818, February 2018. URL `https://arxiv.org/abs/1707.01476`.

[9] Matt Gardner, Partha Pratim Talukdar, Bryan Kisiel, and Tom Mitchell. Improving learning and inference in a large knowledge-base using latent syntactic cues. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 833–838, 2013.

[10] Zhijiang Guo, Yan Zhang, and Wei Lu. Attention guided graph convolutional networks for relation extraction. *CoRR*, abs/1906.07510, 2019. URL `http://arxiv.org/abs/1906.07510`.

[11] Kelvin Guu, John Miller, and Percy Liang. Traversing knowledge graphs in vector space. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 318–327, 2015.

[12] Xu Han, Zhiyuan Liu, and Maosong Sun. Neural knowledge acquisition via mutual attention between knowledge graph and text. In *AAAI*, 2018.

[13] Xu Han, Hao Zhu, Pengfei Yu, Ziyun Wang, Yuan Yao, Zhiyuan Liu, and Maosong Sun. FewRel: A large-scale supervised few-shot relation classification dataset with state-of-the-art evaluation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4803–4809, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1514. URL https://www.aclweb.org/anthology/D18-1514.

[14] Iris Hendrickx, Su Nam Kim, Zornitsa Kozareva, Preslav Nakov, Diarmuid Ó Séaghdha, Sebastian Padó, Marco Pennacchiotti, Lorenza Romano, and Stan Szpakowicz. SemEval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 33–38, Uppsala, Sweden, July 2010. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/S10-1006.

[15] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8): 1735–1780, 1997.

[16] Guoliang Ji, Shizhu He, Liheng Xu, Kang Liu, and Jun Zhao. Knowledge graph embedding via dynamic mapping matrix. 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 687–696, 2015.

[17] Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer, and Omer Levy. Spanbert: Improving pre-training by representing and predicting spans. *CoRR*, abs/1907.10529, 2019. URL http://arxiv.org/abs/1907.10529.

[18] Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759*, 2016.

[19] Ni Lao, Tom Mitchell, and William W Cohen. Random walk inference and learning in a large scale knowledge base. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 529–539. Association for Computational Linguistics, 2011.

[20] Xi Victoria Lin, Richard Socher, and Caiming Xiong. Multi-hop knowledge graph reasoning with reward shaping. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3243–3253, 2018.

[21] Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. Learning entity and relation embeddings for knowledge graph completion. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, AAAI'15, pages 2181–2187. AAAI Press, 2015. ISBN 0-262-51129-0. URL http://dl.acm.org/citation.cfm?id=2886521.2886624.

[22] Angli Liu, Stephen Soderland, Jonathan Bragg, Christopher H. Lin, Xiao Ling, and Daniel S. Weld. Effective crowd annotation for relation extraction. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 897–906, San Diego, California, June 2016. Association for Computational Linguistics. doi: 10.18653/v1/N16-1104. URL https://www.aclweb.org/anthology/N16-1104.

[23] Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60, 2014. URL http://www.aclweb.org/anthology/P/P14/P14-5010.

[24] Arvind Neelakantan, Benjamin Roth, and Andrew McCallum. Compositional vector space models for knowledge base completion. In *ACL*, 2015.

[25] Thien Huu Nguyen and Ralph Grishman. Relation extraction: Perspective from convolutional neural networks. In *Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing*, pages 39–48, Denver, Colorado, June 2015. Association for Computational Linguistics. doi: 10.3115/v1/W15-1506. URL https://www.aclweb.org/anthology/W15-1506.

[26] Nanyun Peng, Hoifung Poon, Chris Quirk, Kristina Toutanova, and Wen tau Yih. Cross-sentence n-ary relation extraction with graph lstms, 2017.

[27] Matthew E. Peters, Mark Neumann, Robert L. Logan IV, Roy Schwartz, Vidur Joshi, Sameer Singh, and Noah A. Smith. Knowledge enhanced contextual word representations, 2019.

[28] Xiang Ren, Zeqiu Wu, Wenqi He, Meng Qu, Clare R. Voss, Heng Ji, Tarek F. Abdelzaher, and Jiawei Han. Cotype: Joint extraction of typed entities and relations with knowledge bases. *CoRR*, abs/1610.08763, 2016. URL http://arxiv.org/abs/1610.08763.

[29] Sebastian Riedel, Limin Yao, and Andrew McCallum. Modeling relations and their mentions without labeled text. In José Luis Balcázar, Francesco Bonchi, Aristides Gionis, and Michèle Sebag, editors, *Machine Learning and Knowledge Discovery in Databases*, pages 148–163, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg. ISBN 978-3-642-15939-8.

[30] Sebastian Riedel, Limin Yao, and Andrew Mccallum. Modeling relations and their mentions without labeled text. pages 148–163, 09 2010. doi: 10.1007/978-3-642-15939-8_10.

[31] Livio Baldini Soares, Nicholas FitzGerald, Jeffrey Ling, and Tom Kwiatkowski. Matching the blanks: Distributional similarity for relation learning. *CoRR*, abs/1906.03158, 2019. URL http://arxiv.org/abs/1906.03158.

[32] George Stoica*, Otilia Stretcu*, Emmanouil Antonios Platanios*, Barnabás Póczos, and Tom M. Mitchell. Contextual Parameter Generation for Knowledge Graph Link Prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2020.

[33] Kai Sheng Tai, Richard Socher, and Christopher D. Manning. Improved semantic representations from tree-structured long short-term memory networks. *CoRR*, abs/1503.00075, 2015. URL http://arxiv.org/abs/1503.00075.

[34] Kristina Toutanova, Victoria Lin, Wen-tau Yih, Hoifung Poon, and Chris Quirk. Compositional learning of embeddings for relation paths in knowledge base and text. In *ACL*, 2016.

[35] Théo Trouillon, Johannes Welbl, Sebastian Riedel, Éric Gaussier, and Guillaume Bouchard. Complex embeddings for simple link prediction. In *International Conference on Machine Learning (ICML)*, volume 48, pages 2071–2080, 2016.

[36] Guanying Wang, Wen Zhang, Ruoxu Wang, Yalin Zhou, Xi Chen, Wei Zhang, Hai Zhu, and Huajun Chen. Label-free distant supervision for relation extraction via knowledge graph embedding. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2246–2255, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1248. URL https://www.aclweb.org/anthology/D18-1248.

[37] Linlin Wang, Zhu Cao, Gerard de Melo, and Zhiyuan Liu. Relation classification via multi-level attention CNNs. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1298–1307, Berlin, Germany, August 2016. Association for Computational Linguistics. doi: 10.18653/v1/P16-1123. URL https://www.aclweb.org/anthology/P16-1123.

[38] R. Wang, B. Li, S. Hu, W. Du, and M. Zhang. Knowledge graph embedding via graph attenuated attention networks. *IEEE Access*, 8:5212–5224, 2020.

[39] Jason Weston, Antoine Bordes, Oksana Yakhnenko, and Nicolas Usunier. Connecting language and knowledge bases with embedding models for relation extraction. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1366–1371, Seattle, Washington, USA, October 2013. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/D13-1136.

[40] Bishan Yang, Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. Embedding entities and relations for learning and inference in knowledge bases. In *International Conference on Learning Representations (ICLR)*, 2015.

[41] Daojian Zeng, Kang Liu, Siwei Lai, Guangyou Zhou, and Jun Zhao. Relation classification via convolutional deep neural network. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 2335–2344, Dublin, Ireland, August 2014. Dublin City University and Association for Computational Linguistics. URL https://www.aclweb.org/anthology/C14-1220.

[42] Ce Zhang, Feng Niu, Christopher Ré, and Jude Shavlik. Big data versus the crowd: Looking for relationships in all the right places. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 825–834, Jeju Island, Korea, July 2012. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/P12-1087.

[43] Ningyu Zhang, Shumin Deng, Zhanlin Sun, Guanying Wang, Xi Chen, Wei Zhang, and Huajun Chen. Long-tail relation extraction via knowledge graph embeddings and graph convolution networks. *CoRR*, abs/1903.01306, 2019. URL http://arxiv.org/abs/1903.01306.

[44] Yuhao Zhang, Victor Zhong, Danqi Chen, Gabor Angeli, and Christopher D. Manning. Position-aware attention and supervised data improve slot filling. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 35–45, Copenhagen, Denmark, September 2017. Association for Computational Linguistics. doi: 10.18653/v1/D17-1004. URL https://www.aclweb.org/anthology/D17-1004.

[45] Yuhao Zhang, Peng Qi, and Christopher D. Manning. Graph convolution over pruned dependency trees improves relation extraction. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2205–2215, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1244. URL https://www.aclweb.org/anthology/D18-1244.

[46] Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu. ERNIE: Enhanced language representation with informative entities. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1441–1451, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1139. URL https://www.aclweb.org/anthology/P19-1139.

[47] Peng Zhou, Wei Shi, Jun Tian, Zhenyu Qi, Bingchen Li, Hongwei Hao, and Bo Xu. Attention-based bidirectional long short-term memory networks for relation classification. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 207–212, Berlin, Germany, August 2016. Association for Computational Linguistics. doi: 10.18653/v1/P16-2034. URL https://www.aclweb.org/anthology/P16-2034.

Appendices

A TACRED Revision Continued

Table 2: Mappings between super-clusters and sentence groups. Sentence groups are defined by the pair, (SUBJECT_TYPE, OBJECT_TYPE), which describes the subject and object type of all sentences in the group. The leftmost column denotes each super-cluster name. The middle column lists the two possible subject types (ORGANIZATION and PERSON), while the rightmost column shows the list of object types whose pairing with the corresponding subject type is an element of the respective super-cluster. For instance, (PERSON, TITLE) represents the sentence group where all sentence subject types are PERSON and all object types are TITLE. From the table, this group is an element of the per2miscmulti super-cluster.

Super-Cluster	Subject Type	Object Types
org2miscmulti	ORGANIZATION	URL, DATE, NUMBER, RELIGION, IDEOLOGY, MISC
org2locmulti		CITY, COUNTRY, STATE_OR_PROVINCE, LOCATION
org2org		ORGANIZATION
org2per		PERSON
per2miscmulti	PERSON	TITLE, DATE, CRIMINAL_CHARGE, RELIGION, NUMBER, CAUSE_OF_DEATH, DURATION, MISC
per2locmulti		NATIONALITY, COUNTRY, STATE_OR_PROVINCE, CITY, LOCATION
per2org		ORGANIZATION
per2per		PERSON

A.1 Wrong Type Handling

Our final super-clusters are show in Table 2. Each cluster contains at least one sentence group (i.e., sentences that correspond to a specific subject-object type pair), every sentence group belongs to a super-cluster, and there is not any group overlap between super-cluster.

However, our modified "super-cluster"-based sentence aggregation also increases the size of the candidate label set presented to annotators during annotation. While in many cases the resultant set is reasonably sized (under 9 relations), a minority of clusters have very large label sets, containing up to 14 relations. Large label sets can make it challenging for annotators to accurately and efficiently choose the most appropriate answer. To ensure that the candidate sets we present to the annotators are not too large, we impose a maximum size of 9 relations for each sentence. Clusters with corresponding label sets of size less than 9 are left intact and are annotated in a *single-stage* fashion, Larger clusters, however, are broken down into sub-clusters and are annotated using a *multi-stage* process. The single-stage annotation process consists of asking a single question for each sentence, where the candidate set of relations contains all of the corresponding super-cluster relations. The multi-stage annotation process consists of splitting a large cluster's label set into subsets such that each subset has fewer relations than our threshold (i.e., 9). Then, one of these subsets is selected and annotated in the same way as for the single-stage process. Afterwards, all sentences assigned to the special WRONG_TYPE relation (indicating that none of the relations in the candidate subset was plausible) are re-annotated using a different subset of relations. This process is repeated until either all of the subsets are exhausted, or all of the sentences are annotated with labels other than the special WRONG_TYPE relation.

A.2 Relation Definitions Refinement

Category (i) — *Explicitly enabling identity relationships*. We observed substantial label inconsistency in TACRED sentences whose subject and object refer to the same person (e.g., "Holly shows off a few pieces of her jewelry line here," where "Holly" is the subject and "her" is the object. Such sentences were inconsistently tagged as either PERSON:OTHER_FAMILY or NO_RELATION. Despite accounting for nearly 10% of TACRED, these sentences are difficult to annotate because they lie in a gray zone of the TAC KBP label guidelines: they are neither explicitly allowed nor disallowed. To this end, we opted to include these types of relationships in the PERSON:ALTERNATE_NAMES relation. Namely, we extended the definition of PERSON:ALTERNATE_NAMES to also explicitly account

for references to the same person, instead of only references using *different names*. Furthermore, in order to avoid confusion and incompatibilities between TACRED and Re-TACRED (our improved TACRED dataset), we renamed the PERSON:ALTERNATE_NAMES to PERSON:IDENTITY.

Category (ii) — *Merging very similar labels*: The relations ORGANIZATION:PARENTS and ORGANIZATION:MEMBER_OF, and their corresponding inverses, ORGANIZATION:SUBSIDIARIES and ORGANIZATION:MEMBERS, describe the relationship where the subject organization is a member (or part) of the object organization, and its inverse. Their sole distinction lies in the fact that ORGANIZATION:MEMBER_OF indicates an *autonomous* relationship between the subject and the object (i.e., the subject is a member of the object by choice), while ORGANIZATION:PARENTS indicates a dependent link where the subject is subsumed by the object (e.g., "LinkedIn" and "Microsoft"), and similarly for the second pair. While such fine-grained distinctions may be viable in a document-level relation extraction setting—The TAC KBP evaluations were defined as document-level relation extraction tasks—they can be extremely challenging (even impossible) at the sentence-level, where significantly less information is available. In fact, in multiple of the cases that we manually reviewed, the correct label could only be determined through a search on the Internet, rather than by relying on the provided sentences. Thus, we decided to merge the two pairs of relations into ORGANIZATION:MEMBER_OF and ORGANIZATION:MEMBERS, respectively.

Category (iii) — *Relaxing challenging criteria*: We also made alterations to ORGANIZATION:-LOCATION_OF_HEADQUARTERS relations, where LOCATION can be substituted for any type of location (e.g., CITY). Our initial annotation process for these relations resulted in substantial confusion due to syntactic ambiguities present throughout the data (e.g., does the phrase "ORGANIZATION from CITY" always imply that the specified organization is headquartered in the specified city? Based on the TAC KBP guidelines it can, but determining whether it does turned out to be particularly challenging for the annotators). Based on this observation, we decided to generalize the corresponding relation definitions to represent any location where an organization has a branch or office (rather than specifically where it is headquartered).

Category (iv) — *Enforcing label mutual-exclusivity*: Although TACRED is defined as a single-label relation extraction dataset (i.e., the relations are all *mutually-exclusive*), certain sentences can fit multiple relations. This is especially common among sentences which invoke a residential relationship between people and locations. For example, both relations PERSON:CITIES_OF_RESIDENCE and PERSON:CITY_OF_BIRTH apply to the sentence "He is a native of Potomac, Maryland." We account for these cases by altering the relation definitions to create clear boundaries for when one relation is more appropriate over another (e.g., any mention of the word "native" or any of its synonyms cannot be assigned a residence relation, such as PERSON:CITIES_OF_RESIDENCE).

A.3 Foreign Language Removal

In addition, we noticed that 1,058 TACRED sentences were not written in English (we automated this detection process by using the FastText [18] language identification model). Due to TACRED being defined in the English Language, we removed these sentences from the dataset, leaving us with 105,206 sentences.

B TACRED and Re-TACRED Comparison Continued

B.1 Model Hyperparameters

We train all our TACRED-based models using the reported hyperparameters by their respective contributors. All hyperparameter details for our Re-TACRED-based methods can be found below. Additionally, all code required to reproduce our results and our new dataset can be found in our repository at https://github.com/gstoica27/Re-TACRED.git. We train our PALSTM and C-GCN models on a single Nvidia Titan X GPU, and utilized a single Nvidia Tesla V100 GPU to train SpanBERT.

Re-TACRED PALSTM. We perform an extensive grid-search over LSTM hidden dimension sizes from {100, 150, 200, 250, 300}, LSTM depth of {1, 2, 3}, word dropout from {0.0, 0.01, 0.04, 0.1, 0.25, .5}, and position-encoding dimension size among {15, 20, 25, 30, 50, 75, 100}. However, we

observe the best performance with the hyperparameters reported by [44]. In addition, we employ the equivalent training strategy as they report in [44] (detailed under Appendix B of their publication).

Re-TACRED C-GCN. Similar to our observations experimenting with PALSTM, we find that keeping the majority of hyperparemeters equivalent to those reported by [45] yield the best results. The sole parameter we alter is increasing the residual neural network hidden dimension from 200 to 300. In addition, we use the same training procedure as [45] (described in Appendix A of their publication).

Re-TACRED SpanBERT. For SpanBERT, we perform a grid-search over learning rate sizes in {1e-6, 2e-6, 2e-5} and warm-up proportions in {.1, .2}. However, we observe the best performance using the reported parameters by [17]. We refer readers to [17] (detailed in Section 4.2 and Appendix B in their publication) for further details on training strategy.

B.2 Distribution Differences

In addition to revising more than 22% of TACRED labels, we observe significant distributional alterations between relations. For instance, we observed that 41.8% more sentences are labeled with `PERSON:CITY_OF_BIRTH` than in the original dataset. Of these, 55.2% were originally labeled as `PERSON:CITIES_OF_RESIDENCE`, illustrating the effect of improved label definitions. Moreover, we observed a 75.7% average increase in labels describing organizations in locations (e.g., `ORGANIZATION:CITY_OF_HEADQUARTERS`). Of these revisions, over 96% were originally labeled as `NO_RELATION`. We attribute this influx of assignments primarily due to our changes in the respective relation definitions described in Section 3, as well as our efforts to better handle wrong assignments of subject and object types.

While our revisions increase the presence of many labels, they also substantially decrease the presence of several others. For instance, we observed the largest reduction in `PERSON:CITIES_OF_RESIDENCE`, where 44.5% of the sentences were re-annotated with a different label. Interestingly, this complements our aforementioned increase in sentences labeled with `PERSON:CITY_OF_BIRTH`, suggesting a high rate of confusion between the two in the original TACRED dataset. This pattern is also mirrored for the `PERSON:COUNTRIES_OF_RESIDENCE` and `PERSON:STATES_OR_PROVINCES_OF_RESIDENCE` relations which changed to the `PERSON:COUNTRIES_OF_BIRTH` relation and the `PERSON:STATES_OR_PROVINCES_OF_BIRTH` relation, respectively. Additionally, we found a 39.9% decrease in sentences labeled with the `PERSON:OTHER_FAMILY`. We attribute this decrease due to our moving sentences with the `PERSON:IDENTITY` relation.

Table 3: Micro-averaged f1-score for all our refined labels in TACRED and Re-TACRED. Categories are defined as described in Appendix A.2. In addition, Re-TACRED performance improvements are listed for each model under the "Change %" rows.

Model	Dataset	Refined Labels			
		Category (i)	Category (ii)	Category (iii)	Category (iv)
PALSTM	TACRED	46.7	21.2	55.9	51.9
	Re-TACRED	87.6	48.8	68.8	53.4
	Change %	+30.9	+27.6	+12.9	+1.5
C-GCN	TACRED	14.6	22.7	56.7	51.5
	Re-TACRED	88.1	51.9	73.7	54.2
	Change %	+73.5	+29.2	+17.0	+2.7
SpanBERT	TACRED	44.1	51.9	66.8	55.9
	Re-TACRED	91.7	65.1	74.0	69.8
	Change %	+56.6	+13.2	+7.2	+13.9

B.3 Effect of Refined Labels.

Table 3 reports the micro-averaged f1-scores for each refined-category on TACRED and Re-TACRED. Our refinements are defined as Section A.2. Overall, our label refinements yield significant performance improvements across all models *by as much as 73.5%*. While PALSTM and C-GCN performances are difficult to distinguish on TACRED, C-GCN exhibits substantially better perfor-

Table 4: Results for multiple RE models (leftmost column) on different train-and-evaluation combinations (represented by the second and third columns). The remaining columns show metric results.

Model	Train Split	Test Split	Metrics		
			F1	Precision	Recall
PALSTM	$\texttt{TACRED}_{\texttt{train}}$	$\texttt{TACRED}_{\texttt{test}}$	72.3	71.3	73.3
	$\texttt{TACRED}_{\texttt{train}}$	$\texttt{Re-TACRED}_{\texttt{test}}$	73.3	76.7	70.2
	$\texttt{Re-TACRED}_{\texttt{train}}$	$\texttt{TACRED}_{\texttt{test}}$	68.3	65.9	70.9
	$\texttt{Re-TACRED}_{\texttt{train}}$	$\texttt{Re-TACRED}_{\texttt{test}}$	75.9	75.8	76.1
C-GCN	$\texttt{TACRED}_{\texttt{train}}$	$\texttt{TACRED}_{\texttt{test}}$	72.6	71.1	74.3
	$\texttt{TACRED}_{\texttt{train}}$	$\texttt{Re-TACRED}_{\texttt{test}}$	73.2	76.0	70.6
	$\texttt{Re-TACRED}_{\texttt{train}}$	$\texttt{TACRED}_{\texttt{test}}$	69.2	68.5	69.8
	$\texttt{Re-TACRED}_{\texttt{train}}$	$\texttt{Re-TACRED}_{\texttt{test}}$	77.3	78.2	76.5
SpanBERT	$\texttt{TACRED}_{\texttt{train}}$	$\texttt{TACRED}_{\texttt{test}}$	75.0	74.7	75.3
	$\texttt{TACRED}_{\texttt{train}}$	$\texttt{Re-TACRED}_{\texttt{test}}$	76.8	81.2	72.8
	$\texttt{Re-TACRED}_{\texttt{train}}$	$\texttt{TACRED}_{\texttt{test}}$	74.1	70.9	77.7
	$\texttt{Re-TACRED}_{\texttt{train}}$	$\texttt{Re-TACRED}_{\texttt{test}}$	84.1	85.0	83.1

mance than PALSTM after label refinement. Similarly, SpanBERT achieves significantly better f1-scores, by at least 7.2% in every category. These results highlight the added clarity our refinements have of labels. All methods achieve the largest gain in category (i) refinements. This indicates that their robustness at detecting same-person relationships is significantly higher than could be observed in TACRED. Interestingly, both C-GCN and PALSTM exhibit very small improvement on category (iv) refinements. We hypothesize that this is due to the complexity of "residence" relations within this group. Namely, characterizations of residence are diverse in the TAC KBP documentation. For instance, "grew up", "lives", "has home", "from", etc. . . are just a few of many valid residence indications. Additionally, we observe significant improvements in category (ii) refinements, illustrating their difficulties in distinguishing between the subtle label differences in each group. By addressing these nuances, we observe significant f1-score increase on Re-TACRED.

B.4 Effect of Non-Refined Labels

Table 4 shows the results of our study investigating the performance differences between TACRED and Re-TACRED over non-refined labels. We conduct this analysis by comparing model performance over different combinations of train and test splits from TACRED and Re-TACRED. We denote train splits using $[\cdot]_{\texttt{train}}$ and test splits using $[\cdot]_{\texttt{test}}$, where $[\cdot]$ is either TACRED or Re-TACRED (e.g., $\text{TACRED}_{\texttt{train}}$). All models are then trained on $\text{TACRED}_{\texttt{train}}$ or $\text{Re-TACRED}_{\texttt{train}}$, and evaluated on $\text{TACRED}_{\texttt{test}}$ or $\text{Re-TACRED}_{\texttt{test}}$.

The results show several interesting differences between TACRED and Re-TACRED. First, all methods trained and evaluated on TACRED obtain significantly higher performance on the non-refined labels than over the full label set. We attribute this increase to the fact that these relations are less ambiguous compared than the refined ones. Second, methods trained on $\text{TACRED}_{\text{train}}$ achieve better performance on $\text{Re-TACRED}_{\text{test}}$ than on $\text{TACRED}_{\text{test}}$. This is consistent with the findings in [2], and suggests that TACRED may be under-estimating model performance, and large improvements can be obtained simply by evaluating models on higher quality annotations. Third, methods trained on $\text{Re-TACRED}_{\text{train}}$ and evaluated on $\text{TACRED}_{\text{test}}$ perform worse than those evaluated on $\text{Re-TACRED}_{\text{test}}$. A deeper inspection of the data reveals that such models exhibit significantly fewer correct positively labeled predictions in $\text{TACRED}_{\text{test}}$ than in $\text{Re-TACRED}_{\text{test}}$, resulting in substantially lower scores. For instance, SpanBERT trained on $\text{Re-TACRED}_{\text{train}}$ exhibits 16.5% fewer correct positively labeled instances in $\text{TACRED}_{\text{test}}$ compared to $\text{Re-TACRED}_{\text{test}}$. This highlights the effects of our label changes described in Section 4: many positively labeled sentences in Re-TACRED are either negatively labeled or assigned another positive relation in TACRED. Fourth, models trained and evaluated on Re-TACRED perform significantly better than any other combination. Thus, while methods trained on $\text{TACRED}_{\text{train}}$ achieve performance boosts when testing on $\text{Re-TACRED}_{\text{test}}$ (compared to evaluating on $\text{TACRED}_{\text{test}}$), training on $\text{Re-TACRED}_{\text{train}}$ is critical to achieving the strongest performance on $\text{Re-TACRED}_{\text{test}}$.

Table 5: Five handpicked sentences from the Re-TACRED test split that a TACRED-trained SpanBERT model misclassifies but a Re-TACRED-trained SpanBERT method correctly classifies. Sentence subjects and objects are defined as in Section 1, and the complete TACRED-trained SpanBERT predictions and gold labels are provided. ORG represents the ORGANIZATION type and PER denotes the PERSON type.

Sentence	TACRED Prediction	Correct Label
"...his posts as Cephalon's chairman and chief executive."	NO_RELATION	PER:EMPLOYEE_OF
"...Pakistani journalist and Taliban expert Ahmed Rashid,..."	NO_RELATION	PER:TITLE
"...National Taiwan Symphony Orchestra (NTSO) ...an NTSO..."	NO_RELATION	ORG:ALTERNATE_NAMES
"His therapist told him to politely decline, 'which helped."	PER:TITLE	NO_RELATION
"...her stepchildren, Susan, ..., Stephen and Maggie Mailer; ..."	PER:SIBLINGS	PER:CHILDREN

Table 6: Micro-averaged f1-score for each category in TACRED and Re-TACRED. PER stands for PERSON and ORG for ORGANIZATION.

Model	Dataset	Categories					
		PER:*	ORG:*	PER:ORG	ORG:PER	PER:PER	ORG:ORG
PALSTM	TACRED	66.8	65.2	65.3	72.6	59.9	59.3
	Re-TACRED	79.0	74.4	62.9	85.1	85.2	70.3
	Change %	+12.2	+8.8	-2.4	+14.3	+25.3	+11.0
C-GCN	TACRED	66.5	65.9	66.4	72.2	49.9	61.6
	Re-TACRED	79.9	76.7	65.3	85.3	85.3	72.2
	Change %	+12.6	+10.6	+8.9	+13.1	+35.4	+10.6
SpanBERT	TACRED	69.7	69.5	68.9	74.8	61.2	68.1
	Re-TACRED	85.6	80.8	78.6	88.6	88.8	79.0
	Change %	+15.9	+11.3	+9.7	+13.8	+7.6	+10.9

B.5 Re-TACRED Error Correction

Table 5 shows several sentences highlighting the types of prediction errors TACRED-trained SpanBERT makes that Re-TACRED trained SpanBERT is able to correct for. Based on the observations described in Section 4, we argue that TACRED-trained SpanBERT's erroneous NO_RELATION predictions are primarily due to implicit negative bias TACRED-trained methods have as a result of TACRED's severe NO_RELATION data skew (79.6% of sentences are negatively labeled). In contrast, Re-TACRED trained SpanBERT is able to better recognize instances where NO_RELATION) is not appropriate, potentially due to Re-TACRED containing substantially fewer negatively labeled instances (68.0%).

B.6 Performance Change Across Label Types.

To better understand the overall impact of Re-TACRED, we also analyze model quality over several relation categories between TACRED and Re-TACRED. Each category examines particular relation types, and is defined similar to [2]. Namely, PER:* and ORG:* represents all relations whose subject types are PERSON and ORGANIZATION respectively, while those denoted by X:Y symbolize relations whose subject type is X and object type is Y. We choose these categories due to the diversity of specific relations they represent, and their overall coverage of the relation-space. For each category, we compute the micro-averaged f1-score based on the scores and supports from its relations. We report our results in Table 6.

The results indicate that C-GCN and PALSTM exhibit a complementary relationship over many categories with TACRED labels. While C-GCN beats PALSTM in ORGANIZATION:*, the reverse is true with PERSON:*. Moreover, PALSTM significantly outperforms C-GCN by 10% on PERSON:PERSON relationships. However, this compatibility disappears when the two are compared on our revised dataset. Notably, C-GCN outscores PALSTM in every category. Thus, while TACRED paints these methods as very being comparable, Re-TACRED reveals that C-GCN is a much stronger model. SpanBERT consistently beats PALSTM and C-GCN in both TACRED and Re-TACRED evaluations, illustrating its robustness.