

# Re-TACRED: Addressing Shortcomings of the TACRED Dataset

George Stoica <sup>1</sup>, Emmanouil Antonios Platanios <sup>2</sup>, Barnabás Póczos <sup>1</sup>

<sup>1</sup> Carnegie Mellon University

<sup>2</sup> Microsoft Semantic Machines

gis@cs.cmu.edu, emplata@microsoft.com, bapoczos@cs.cmu.edu

## Abstract

TACRED is one of the largest and most widely used sentence-level relation extraction datasets. Proposed models that are evaluated using this dataset consistently set new state-of-the-art performance. However, they still exhibit large error rates despite leveraging external knowledge and unsupervised pre-training on large text corpora. A recent study suggested that this may be due to poor dataset quality. The study observed that over 50% of the most challenging sentences from the development and test sets are incorrectly labeled and account for an average drop of 8% f1-score in model performance. However, this study was limited to a small biased sample of 5k (out of a total of 106k) sentences, substantially restricting the generalizability and broader implications of their findings. In this paper, we address these shortcomings by: (i) performing a comprehensive study over the whole TACRED dataset, (ii) proposing an improved crowd-sourcing strategy and deploying it to re-annotate the whole dataset, and (iii) performing a thorough analysis to understand how correcting TACRED affects previously published results. After verification, we observe that 22.1% of TACRED labels are incorrect. Moreover, evaluating several models on our revised dataset yields an average f1-score improvement of 13% and also helps uncover stronger relationships between the different models (rather than simply offsetting or scaling their scores by a constant factor). Finally, aside from our analysis we also release **Re-TACRED**, a new and completely re-annotated version of the TACRED dataset that can be used to perform 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 diverse concepts. Relation extraction (RE) is a popular learning task aimed at extracting such relationships between entities in plain text. For example, given the sentence “[William Shakespeare]<sub>SUB</sub> was born in [England]<sub>OBJ</sub>,” where “William Shakespeare” and “England” are the subject and object entities respectively, the objective of a RE method is to infer the correct relation (e.g., PERSON:BORN\_IN\_COUNTRY) between the subject and the object. Developing successful RE methods requires robust ways

to evaluate their qualities. TACRED (Zhang et al. 2017) is one of the largest and most widely used such datasets.

TACRED consists of 106,264 sentences of varied complexity that were annotated using Amazon Mechanical Turk (AMT). Although just three years-old, a multitude of approaches have been proposed and evaluated using the TACRED dataset. These approaches typically leverage an assortment of different knowledge: (i) auxiliary named entity recognition (NER) and part-of-speech (POS) tag information—Zhang et al. (2017); Zhang, Qi, and Manning (2018); Guo, Zhang, and Lu (2019), (ii) sentence dependency parses—Zhang, Qi, and Manning (2018); Guo, Zhang, and Lu (2019), (iii) fine-tuned pre-trained language representations—Soares et al. (2019); Peters et al. (2019); Alt, Hübner, and Hennig (2019); Joshi et al. (2019); Chen et al. (2020); Zhang et al. (2019), or (iv) even external training data—Soares et al. (2019); Peters et al. (2019). Recently, methods have converged at ~71.5% f1-score on the test data, which raises the question of whether we have reached the maximum possible attainable performance on the TACRED dataset, and if so, why? (Alt, Gabryszak, and Hennig 2020) 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 in the ground truth, leading to an average model performance improvement of over 8% after correcting these labels. Furthermore, they identified several error categories that describe model mistakes on their revised test split. However, the broader impact of their work is limited by two key factors. First, they restricted their dataset revisions to a small and biased 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 aim to address these shortcomings by performing a re-annotation of the entire TACRED dataset. Our contributions can be summarized as follows:

- Annotation: We propose an improved and cost-efficient crowd-sourcing annotation strategy that we subsequently deploy to re-annotate the full TACRED dataset. Our task design tackles an important flaw in the original TACRED

- data collection process, refines existing relation definitions to be better suited for the TACRED dataset, and also uses quality assurance mechanisms in order to ensure increased annotation quality (and thus accuracy). Our annotators achieve an average agreement rate of 82.3% and inter-annotator Fleiss' kappa of .77, which is significantly higher than the .54 kappa achieved by [Zhang et al. \(2017\)](#).
- **Analysis:** We perform a thorough comparison of the TACRED labels and our new re-annotated labels. We analyze both their qualitative differences, and their impact on the evaluation and comparison of existing RE models. Our results show that our corrections significantly improve model performance by an average of 13% f1-score, and also indicate that prior analysis on the types of errors that the models make may have been misguided due to the wrong TACRED labels.
  - **Dataset Release:** We release our newly corrected TACRED labels publicly online<sup>1</sup>. Due to licensing restrictions, we cannot release complete dataset, but similar to [Alt, Gabryszak, and Hennig \(2020\)](#), we provide a patch containing all our revisions. We term the corrected dataset Revised-TACRED (Re-TACRED).

## 2 Background

The TAC relation extraction dataset (TACRED), introduced by [Zhang et al. \(2017\)](#), is one of the largest and most widely used datasets for sentence-level relation extraction. It consists of over 106,000 sentences collected from the 2009-2014 TAC knowledge base population (KBP) evaluations, with those between 2009-2012 used for training, 2013 for development, and 2014 for testing. 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). Furthermore, each instance is assigned one of 42 labels that describes the relationship between the subject and the object. These labels consist of 41 relation types that describe the existence of some relationship between the subject and the object (e.g., PERSON:CITY\_OF\_BIRTH), and a special NO\_RELATION predicate to indicate the absence of a relationship. For example, consider the sentence “[John Doe]<sub>SUB</sub> lives in [Miami]<sub>OBJ</sub>.<sup>1</sup>” In this case, the subject is a PERSON and the object a CITY. In TACRED, all relations are *typed*, meaning that they only apply to a specific subject and object type. The subject type is always either PERSON or ORGANIZATION, and there exist 17 unique object types. There are a total of 27 subject-object types pairs with corresponding candidate relations.

Instances in the original TACRED dataset were annotated with labels using the Amazon Mechanical Turk (AMT) crowdsourcing platform. The AMT workers were provided sentences with their subject and object spans highlighted, and were asked to choose the appropriate label from a set of suggestions (i.e., the annotation task was framed as a multiple choice task). The suggestions included all labels that were compatible with the subject and object types, along with the special NO\_RELATION label.

<sup>1</sup><https://github.com/gstoica27/Re-TACRED>

## 2.1 TACRED Quality

[Zhang et al. \(2017\)](#) manually verified TACRED annotation quality over a random sample of 300 instances. They reported that they observed a high annotation accuracy of 93.3%, with respect to what they considered as the correct labels for these instances. Coupled with a moderate Fleiss' kappa of .54 over 761 randomly selected annotation pairs, they assumed an acceptable level of label quality. However, recent work suggests that the annotation quality may be significantly lower than previously estimated. [Alt, Gabryszak, and Hennig \(2020\)](#) used crowd-sourcing to manually verify labels for the five thousand most miss-classified sentences from 49 existing relation extraction methods. Their annotation task was designed similar to that of [Zhang et al. \(2017\)](#), with two primary differences to help identify potential issues. First, only workers with prior training in general linguistics were allowed to participate, and these workers were further pruned by asking them to correctly label 500 manually chosen and hand-labeled sentences from the original TACRED development set. Second, the set of possible choices presented to the workers also included the set of predictions made by pre-trained (on the original TACRED dataset) relation extraction models. These predictions may have included type-incompatible relations to help identify cases of wrongly-assigned types. Using this re-annotation procedure, they observed that *over 50% of the TACRED annotations in their sample were incorrect*. Among the wrongly-annotated instances, they found that 36% were erroneously labeled as NO\_RELATION, 49% were incorrectly assigned relations other than NO\_RELATION, and 15% were assigned the wrong label among non-NO\_RELATION labels. Notably, their revised dataset resulted in an average f1-score improvement of 8.1% over the unaltered TACRED dataset, suggesting that using TACRED for evaluating methods may potentially result in inaccurate conclusions. Moreover, their Fleiss' kappa for the new annotations was 0.80 for the development set and 0.87 for the test set, suggesting high annotation quality. These were much higher than computed by the original TACRED labels—.54.

While [Alt, Gabryszak, and Hennig \(2020\)](#) demonstrated several shortcomings of the TACRED dataset, the broader impact of their work is restricted by both their small and biased sample set, and analysis performed over a predominately uncorrected TACRED 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. 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 form the main motivation for this paper.

## 3 TACRED Revision

We propose a new crowdsourcing task design that improves upon previous approaches along the following directions:

1. **Wrong Type Handling:** We performed a manual analysis of 1,000 randomly selected instances and found that about 5% of them have incorrect types for the subject, the object, or both (e.g., “Thomas More Law Center” tagged as a PERSON instead of an ORGANIZATION). This is important because the task design of [Zhang et al. \(2017\)](#) only presented the annotators candidate relations that matched the pre-specified subject and object types. Therefore, if the types were wrong, the annotators had no possible way of choosing the right relation. Section 3.2 proposes a cost-effective modification to the previous task design that addresses this issue.
2. **Relation Definition Refinements:** Similar to [Zhang et al. \(2017\)](#); [Alt, Gabryszak, and Hennig \(2020\)](#), we initially defined all possible relations according to the TAC KBP documentation<sup>2</sup>. However, we observed that the documentation is ambiguous or unintuitive in a small number of cases. This leads to worker confusion and poor annotation quality. We address this problem by altering problematic relation definitions, described in Section 3.3.
3. **Quality Assurance:** In order to ensure high-quality annotations, we employ a two-step quality assurance process for our annotators. This is described in Section 3.4.
4. **Foreign Language Removal:** During our quality analysis for TACRED, we discovered sentences that were not written in the English language. We address this issue in Section 3.5.

The following sections describe our overall crowdsourcing task design, as well as our approach along each of these directions in detail. Note that we *re-annotate the full TACRED dataset* using the Amazon Mechanical Turk (AMT) platform. (as opposed to a small fraction of it like [Alt, Gabryszak, and Hennig \(2020\)](#)). Finally, in Section 4 we perform an analysis of the resulting changes in the TACRED dataset and their impact in evaluating existing relation extraction methods.

### 3.1 Task Design

Labeling TACRED is challenging due to its large size and complex structure. Sentences contain variable amounts of syntactic and lexical ambiguity, making it difficult for crowd-workers to identify the right relation among 42 choices. In order to reduce annotation complexity, we follow a similar approach to [Zhang et al. \(2017\)](#); [Alt, Gabryszak, and Hennig \(2020\)](#). We first group TACRED sentences based on their corresponding subject and object types (e.g., the sentence “[Holly]<sub>SUB</sub> showed off [her]<sub>OBJ</sub> jewelry” is grouped together with sentences whose subject and object both have type PERSON), and we then assign each group a filtered candidate set of labels that consists only of relations that are *type-compatible* (e.g., relations between people), along with the special NO\_RELATION label. Given that the provided types may be wrong (as mentioned earlier), we also allow the annotators to select a special WRONG\_TYPES label for each instance. This is because, if either of the types is incorrect, then the candidate label set may no longer be truly type-compatible, thus only providing implausible options to the annotators. This ought to further reduce confusion in cases when the

types are incorrect, because annotators are made explicitly aware of their possibility and are provided with an option for them.

### 3.2 Wrong Type Handling

The inclusion of the WRONG\_TYPE label implies that the original candidate label sets of affected sentences are not compatible with the sentence. To find the correct relation, each sentence must be re-labeled according to different label sets until a match is found. A potential approach to this problem would be to iteratively consider all possible pairs of subject and object types until annotators agree on a relation other than WRONG\_TYPE. However, such a solution would be prohibitively expensive as in the worst case 27 separate annotation tasks would need to be performed for a single sentence. If just 5% of TACRED sentences have wrong types (our estimate based on a 1,000 sentence sample), then the worst case annotation cost would increase by  $\sim 130\%$ .

We address this issue by defining 8 *super-clusters* over relations, such that each super-cluster contains at least one sentence group (i.e., sentences that correspond to a specific subject-object type pair), and every sentence group belongs to exactly one super-cluster. To illustrate, let one such super-cluster describe all sentences that exhibit a relationship between a PERSON and a LOCATION. Sentence groups within this cluster describe different aspects of the super-cluster relationship (e.g., relationships between people and cities), and do not appear in any other super-cluster. We specify each cluster by aggregating sentence groups whose types were most confused with one another from a random sample of a 1,000 sentences. Our final super-clusters are shown in Table 4 under Appendix A<sup>3</sup>. We define each cluster’s candidate label set as the union of the candidate sets for each of its sentence group members. This increases the probability that type-compatible relations exist for incorrectly-typed sentences within a super-cluster. Moreover, this approach reduces the worst-case overall annotation cost by a factor of  $27/8 = 3.4$ .

However, our modified “super-cluster”-based sentence aggregation also increases the size of the candidate label set presented to workers 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 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

<sup>2</sup><https://tac.nist.gov/2017/KBP/index.html>

<sup>3</sup>Section 5 includes information on how to access the Appendix.

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 were 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.

### 3.3 Relation Definition Refinements

In this section, we describe the changes we made to the original TAC KBP relation definitions in order to make them more clear and intuitive.

**PERSON:IDENTITY:** We observed substantial inconsistency in TACRED between the relations `PERSON:OTHER_FAMILY` and `NO_RELATION` in sentences whose subject and object refer to the same person (e.g., “[Holly]<sub>SUB</sub> shows off a few pieces of [her]<sub>OBJ</sub> jewelry line here,” where the subject and object are denoted as described in Section 1. Despite accounting for nearly 10% of TACRED, these sentences are difficult to annotate due ambiguity in the TAC KBP label guidelines. To this end, we extended the definition of a similar relation `PERSON:ALTERNATE_NAMES` to include any identity references. 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`. Additional details can be found in Appendix F<sup>4</sup>.

**ORGANIZATION:{MEMBER\_OF/MEMBERS}:** The relations `ORGANIZATION:MEMBER_OF` and `ORGANIZATION:PARENTS` describe the relationship where a subject organization is a member (or part) of an object organization. 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]<sub>SUB</sub>” and “[Microsoft]<sub>OBJ</sub>”). 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 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 merged these two relations into one: `ORGANIZATION:MEMBER_OF`. Additionally, we similarly merged their inverses (`ORGANIZATION:MEMBERS` and `ORGANIZATION:SUBSIDIARIES`) into `ORGANIZATION:MEMBERS`.

**Single-Label vs Multi-Label:** 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]<sub>SUB</sub> is a

native of [Potomac]<sub>OBJ</sub>, 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`).

**ORGANIZATION:\*\_OF\_HEADQUARTERS:** We also made alterations to the `ORGANIZATION:*_OF_HEADQUARTERS` relations, where “\*” is a placeholder for location types (e.g., `CITY`). Our initial annotation process for these relations resulted in substantial confusion due to semantic ambiguities present throughout the data. For example, 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 may, but determining whether it does turned out to be particularly challenging for our annotators. Based on this observation, we generalized the corresponding relation definitions to be valid as long as an organization has a branch or office in the label’s location type (rather than forcing it to be headquartered there). For example, `ORGANIZATION:CITY_OF_HEADQUARTERS` became `ORGANIZATION:CITY_OF_BRANCH`.

### 3.4 Quality Assurance

In order to ensure high-quality annotations, we employed a two-step quality assurance process similar to the gated-instruction technique introduced by Liu et al. (2016) for our crowd annotators. The first step, which we call the *trial*, is conducted prior to the data annotation process, and is used to filter out annotators that perform poorly before they are able to label our data. The second stage, which we call the *control*, is performed during our data annotation process in order to ensure consistent high-quality annotations.

**Trial:** We specify several prerequisite criteria that workers must satisfy before annotating our dataset. First, candidates must have had at least 500 previous tasks approved on Amazon Mechanical Turk (AMT), and an overall approval rate ( $\frac{\# \text{ Annotations Approved}}{\# \text{ Annotations Completed}}$ )  $\geq 95\%$ . These filters help ensure that our annotators are both experienced and reliable. In addition, we constructed custom “qualification tests” for all eight of our sentence super-clusters. Since all sentences within a super-cluster are assigned the same set of candidate relations, we made sure that each test contained the definitions of all candidate relations assigned to the respective super-cluster, along with a series of questions aimed at testing a worker’s understanding of each of these relations. A perfect score of 100% was required to pass. These tests serve two purposes: (i) gauge annotator quality, and (ii) *specialize/train* annotators for each super-cluster annotation task. Only annotators that passed these tests were allowed to provide annotations.

**Control:** Although our prerequisites were sufficient to eliminate many untrustworthy workers, we observed several incidents where annotators would devote effort to pass our trial criteria, and then randomly annotate sentences to save time while getting paid. While such events may be easy to detect at small scales where a comprehensive manual review of each annotation is viable, it is infeasible to do so at a large scale in-

<sup>4</sup>Section 5 includes information on how to access the Appendix.

volving tens of thousands of sentences. Thus, we handpicked and manually labeled a set of *control* sentences, and mixed them with the unannotated sentences presented to annotators. Following the work of [Zhang et al. \(2012\)](#), for every five sentences presented to annotators, we made sure that one was a control sentence whose true label was known. This allowed us to estimate the annotator accuracy, which in turn enabled us to impose a filter that only accepted responses from annotators with accuracy higher than 80% (separately computed for each one of our super-clusters). We choose this threshold based off that used by [Zhang et al. \(2012\)](#) throughout their experiments. 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.

### 3.5 Foreign Language Removal

We noticed that 1,058 of the TACRED sentences were not written in English (we automated this detection process by using FastText by [Joulin et al. \(2016\)](#)). Since the task is defined in the English language, we removed these sentences from the dataset, leaving us with 105,206 sentences.

## 4 TACRED and Re-TACRED Comparison

After revision, Re-TACRED consists of 105,206 sentences split amongst 40 different relations. In order to maintain a similar evaluation environment as TACRED, Re-TACRED contains the same train, development, and test splits as TACRED.

In this section we first provide a qualitative comparison between TACRED and Re-TACRED. Then, we provide an empirical analysis for how our re-annotation efforts affect model performance and potentially influence conclusions that were previously drawn from their TACRED evaluations.

### 4.1 Qualitative Comparison

Overall, our Re-TACRED labels achieved an average agreement rate of 82.3% between annotators throughout the whole dataset. Moreover, our inter-annotator Fleiss' Kappa over all annotations is .77, indicating high quality. Our labels disagree with the original TACRED labels in 22.1% of sentences. Out of the modified labels, 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. Our revisions also substantially alter the distribution of relations in TACRED. 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 at defining concrete bounds between the two relations. Moreover, we observed a 75.7% average increase in labels describing organizations in locations. 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 3.3 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` relation. We attribute this decrease due to our moving sentences with the `PERSON:IDENTITY` relation.

### 4.2 Model Performance Comparison

We examine how our changes impact the evaluation of three existing relation extraction models (neither of which are our own):

- PA-LSTM ([Zhang et al. \(2017\)](#)): This model infers relations by applying a one-directional long short-term memory (LSTM) network and a custom position-aware attention mechanism over sentences. It also incorporates sentence token named-entity recognition (NER) tags, part-of-speech (POS) tags, and positional offsets from subjects and objects in its reasoning. We refer readers to [Zhang et al. \(2017\)](#) for further information.
- C-GCN ([Zhang, Qi, and Manning \(2018\)](#)): This model labels sentences by applying a graph-convolution network (GCN) over sentence dependency tree parses. Similar to PA-LSTM, the model first encodes sentences using a bi-directional LSTM network, before processing the outputs over a graph implied by a pruned version of the sentence dependency tree parse. In particular, C-GCN computes the least common ancestor (LCA) between the subject and the object, and removes tree branches that are more than a pre-specified degree away from the LCA. The resulting GCN output representations are finally processed by a multi-layer perceptron to predict relations. We refer readers to [Zhang, Qi, and Manning \(2018\)](#) for further information.
- SpanBERT ([Joshi et al. \(2019\)](#)): This is one of the state-of-the-art models at the time of writing. SpanBERT is similar to BERT ([Devlin et al. \(2019\)](#)), but is instead pre-trained using a span prediction objective, making it better suited to the relation extraction task. SpanBERT also differs from BERT in terms of how the token masking is performed during pre-training, in that it masks contiguous token spans instead of individual tokens. We refer readers to [Joshi et al. \(2019\)](#) for further information.

**Overall Performance Impact.** Table 1 presents the evaluation results of the three models on both TACRED and Re-TACRED. In addition, we mark their performance differences between the two datasets. All results were reported

Dataset	Metric	Models		
		PA-LSTM	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	<b>84.6</b>
	Recall	77.6	78.5	<b>83.9</b>
	F1	77.9	79.1	<b>84.2</b>
Change %	Precision	+12.2	+11.2	<b>+14.5</b>
	Recall	+13.1	+14.1	<b>+14.7</b>
	F1	+11.7	+12.8	<b>+14.5</b>

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.

using micro-averaged f1-scores from the model with the median validation f1-score over five independent runs, as in prior literature. Notably, we observe significant improvements across every metric for each 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 PA-LSTM and C-GCN have similar f1-score increases, their recall and precision enhancements are complementary. C-GCN has larger recall improvement, while PA-LSTM displays a larger precision increase. In contrast to C-GCN and PA-LSTM, 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.

**Performance Change Across Label Types.** To better understand these performances, we also analyze model quality over several relation categories. Each category examines particular relation types, and is defined similar to Alt, Gabryszak, and Hennig (2020). Namely, `PER:*` and `ORG:*` represent 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 from its relations. We report our results in Table 2.

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

in both TACRED and Re-TACRED evaluations, illustrating its robustness.

**Effect of Refined Labels.** We also examine how impactful our label refinements are across different models. Table 3 reports the micro-averaged f1-scores for each label refinement-category on TACRED and Re-TACRED. Categories are defined as in Section 3.3, with a few additions. Namely, we group all `PERSON`, `RESIDENCE`, `BIRTH`, and `DEATH` types into respective `PERSON:RESIDENCE`, `PERSON:BIRTH`, and `PERSON:DEATH` categories. In a similar manner, `ORGANIZATION:LOCATION` marks all location-type relations (e.g., `ORGANIZATION:CITY_OF_BRANCH`) describing the place of an `ORGANIZATION`’s branch or office. `REST` denotes the set of all remaining labels.

Overall, our label refinements yield significant performance improvements across all models *by as much as 88.3%*. While PA-LSTM and C-GCN performances are difficult to distinguish on TACRED, C-GCN exhibits substantially better performance than PA-LSTM after label refinement. Similarly, SpanBERT achieves significantly better f1-scores, by at least 11.6% in every category. Its best improvement is on `PERSON:IDENTITY`, showing an over 470% increase in f1-measure, highlighting the added clarity of our label refinements. Moreover, all methods achieve the largest gain in `PERSON:IDENTITY`<sup>5</sup> classifications, and two—PA-LSTM and C-GCN—improve performance from 0.0% to more than 87.0%. This indicates that their robustness is at detecting same-person relationships is significantly higher than could be observed in TACRED. Interestingly, all models exhibit the least improvement on `PERSON:RESIDENCE` labels. We hypothesize that this is because their relations are more much more complex than similar labels such as birth and death. Specifically, whereas lexical variation describing places of birth and death is limited, characterizations of locations of residences are diverse in the TAC KBP documentation. For instance, “grew up”, “lives”, “has home”, “from”, etc. . . are just a few of many valid indications. Moreover, we observe substantial improvements in `ORGANIZATION:MEMBERS` and `ORGANIZATION:MEMBER_OF`. Both categories yielded among the lowest scores for models evaluated on TACRED, 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.

**Re-TACRED Error Correction.** We further investigate 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 SpanBERT trained on Re-TACRED answers correctly. We choose SpanBERT because it is the best performing model on both TACRED and Re-TACRED out of our three. Overall, we find 2,820 total such sentences. Of these, 82.2% are due to TACRED-trained SpanBERT inferring `NO_RELATION` when the gold label is positive, 14.4%

<sup>5</sup> `PERSON:IDENTITY` refers to `PERSON:ALTERNATE_NAMES` in TACRED.

Model	Dataset	Categories					
		PER:*	ORG:*	PER:ORG	ORG:PER	PER:LOCATION	PER:PER
PA-LSTM	TACRED	66.8	65.2	65.3	72.6	51.9	59.9
	Re-TACRED	79.0	74.4	62.9	85.1	53.4	85.2
C-GCN	TACRED	66.5	65.9	66.4	72.2	51.5	49.9
	Re-TACRED	79.9	76.7	65.3	85.3	54.2	85.3
SpanBERT	TACRED	69.7	69.5	68.9	74.8	55.9	61.2
	Re-TACRED	85.6	80.8	78.6	88.6	69.8	88.8

Table 2: Micro-averaged f1-score for each category in TACRED and Re-TACRED. PER stands for PERSON and ORG for ORGANIZATION. “\*” indicates all object types.

Model	Dataset	Refined Labels					
		ORG:MEMBERS	ORG:MEMBER_OF	PER:RESIDENCE	PER:BIRTH	PER:DEATH	ORG:LOCATION
PA-LSTM	TACRED	23.5	22.6	54.1	31.0	26.7	55.9
	Re-TACRED	48.8	42.7	55.2	40.5	57.1	68.1
	Difference	+15.3	+20.1	+1.1	+8.5	+30.4	+12.2
C-GCN	TACRED	24.0	24.6	54.1	30.0	25.0	56.7
	Re-TACRED	51.9	41.7	55.3	43.7	60.0	73.7
	Difference	+27.9	+17.1	+1.2	+13.7	+35.0	+88.3
SpanBERT	TACRED	52.2	50.3	57.6	50.0	26.7	62.7
	Re-TACRED	64.2	64.0	69.2	68.8	82.1	74.7
	Difference	+12.0	+14.3	+11.6	+18.8	+55.4	+12.5
							+71.4

Table 3: Micro-averaged f1-score for all our refined labels in TACRED and Re-TACRED. PER stands for PERSON and ORG for ORGANIZATION, and the refined relations are grouped according to their type, and are defined as in Section 3.3. Additionally, PER:RESIDENCE, PER:BIRTH, and PER:DEATH represent all LOCATION types of residence, birth, and death respectively, for type PER. ORG:LOCATION is the aggregate of all LOCATION types for ORG subjects.

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. 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%). Table 5 in Appendix B shows several sentences highlighting the types of prediction errors TACRED-trained SpanBERT makes that Re-TACRED trained SpanBERT is able to correct for.

## 5 Supplemental Materials

All supplemental material can be accessed at [https://github.com/gstoica27/Re-TACRED/blob/master/supplemental\\_material.pdf](https://github.com/gstoica27/Re-TACRED/blob/master/supplemental_material.pdf)

## 6 Conclusion

We conducted a comprehensive review of the TACRED dataset. We addressed the limitations of previous work by re-annotating the complete dataset using crowd-sourcing. Our annotation strategy extended previous studies by accounting for data errors, label definition ambiguity, and annotation quality control. Our results show significantly higher inter-annotator agreement rate and Fleiss’ Kappa (.77) than original dataset annotations, suggesting clearer task descriptions and high label annotation reliability. Moreover, we performed

a thorough analysis of how existing relation extraction methods compare between datasets, and how errors change between them. Perhaps most notably, we observed an average improvement of 13% f1-score from three models on our revised dataset.

## 7 Acknowledgements

This work has been partly funded by the DARPA Data-Driven Discovery of Models (D3M) program.

## References

- Alt, C.; Gabryszak, A.; and Hennig, L. 2020. TACRED Revisited: A Thorough Evaluation of the TACRED Relation Extraction Task.
- Alt, C.; Hübner, M.; and Hennig, L. 2019. Improving Relation Extraction by Pre-trained Language Representations. *CoRR* abs/1906.03088. URL <http://arxiv.org/abs/1906.03088>
- Chen, J.; Hoehndorf, R.; Elhoseiny, M.; and Zhang, X. 2020. Efficient long-distance relation extraction with DG-SpanBERT.
- Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.
- Guo, Z.; Zhang, Y.; and Lu, W. 2019. Attention Guided Graph Convolutional Networks for Relation Extraction. *CoRR* abs/1906.07510. URL <http://arxiv.org/abs/1906.07510>
- Joshi, M.; Chen, D.; Liu, Y.; Weld, D. S.; Zettlemoyer, L.; and Levy, O. 2019. SpanBERT: Improving Pre-training by Representing and Predicting Spans. *CoRR* abs/1907.10529. URL <http://arxiv.org/abs/1907.10529>

<sup>6</sup>Section 5 includes information on how to access the Appendix.

Joulin, A.; Grave, E.; Bojanowski, P.; and Mikolov, T. 2016. Bag of Tricks for Efficient Text Classification. *arXiv preprint arXiv:1607.01759*.

Liu, A.; Soderland, S.; Bragg, J.; Lin, C. H.; Ling, X.; and Weld, D. S. 2016. 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*, 897–906. San Diego, California: Association for Computational Linguistics. doi:10.18653/v1/N16-1104. URL <https://www.aclweb.org/anthology/N16-1104>.

Peters, M. E.; Neumann, M.; IV, R. L. L.; Schwartz, R.; Joshi, V.; Singh, S.; and Smith, N. A. 2019. Knowledge Enhanced Contextual Word Representations.

Soares, L. B.; FitzGerald, N.; Ling, J.; and Kwiatkowski, T. 2019. Matching the Blanks: Distributional Similarity for Relation Learning. *CoRR* abs/1906.03158. URL <http://arxiv.org/abs/1906.03158>.

Zhang, C.; Niu, F.; Ré, C.; and Shavlik, J. 2012. 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)*, 825–834. Jeju Island, Korea: Association for Computational Linguistics. URL <https://www.aclweb.org/anthology/P12-1087>

Zhang, Y.; Qi, P.; and Manning, C. D. 2018. Graph Convolution over Pruned Dependency Trees Improves Relation Extraction. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2205–2215. Brussels, Belgium: Association for Computational Linguistics. doi:10.18653/v1/D18-1244. URL <https://www.aclweb.org/anthology/D18-1244>

Zhang, Y.; Zhong, V.; Chen, D.; Angeli, G.; and Manning, C. D. 2017. Position-aware Attention and Supervised Data Improve Slot Filling. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 35–45. Copenhagen, Denmark: Association for Computational Linguistics. doi:10.18653/v1/D17-1004. URL <https://www.aclweb.org/anthology/D17-1004>.

Zhang, Z.; Han, X.; Liu, Z.; Jiang, X.; Sun, M.; and Liu, Q. 2019. ERNIE: Enhanced Language Representation with Informative Entities. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 1441–1451. Florence, Italy: Association for Computational Linguistics. doi:10.18653/v1/P19-1139. URL <https://www.aclweb.org/anthology/P19-1139>.