Re-TACRED: Addressing Shortcomings of the TACRED Dataset

George Stoica [1], **Emmanouil Antonios Platanios** [2], **Barnabás Póczos** [1]
[1] Carnegie Mellon University
[2] 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 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 "`[William Shakespeare]`$_{\texttt{SUB}}$ `was born in [England]`$_{\texttt{OBJ}}$," where "`William Shakespeare`" and "`England`" are the sentence subject and object respectively, the objective of an RE method is to infer the correct relation, `PERSON:BORN_IN_COUNTRY`, between the subject and the object. In developing successful RE methods we need robust ways to evaluate how good our methods are, and two widely used benchmarks are the SemEval 2010 Task 8 (Hendrickx et al. 2010) dataset, and the significantly larger and more widely used TACRED (Zhang et al. 2017) dataset. However, as Alt, Gabryszak, and Hennig (2020) show, it suffers from flaws that may affect conclusions drawn from using it as a means to evaluate. In this paper, we propose a new crowdsourcing task design to re-annotate TACRED, resulting in Re-TACRED, a fully re-annotated version of TACRED. The resulting dataset is of significantly higher quality and results in an interesting adjustment of prior results that were obtained using TACRED (Section 4 describes this in more detail).

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 $\sim$71.5% f1-score on the test data, which recently raised 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, 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 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 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 a couple 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, analyzing both their qualitative differences, but also 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 and that matter may have been somewhat misguided due to the wrong TACRED labels.
– Dataset Release: We release our newly corrected TACRED labels publicly online[1]. 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 relations that describe the existence of some relationship between the subject and the object (e.g., `CITY_OF_BIRTH`), and a special `NO_RELATION` predicate to indicate the absence of a relationship. For example, consider the sentence "`[John Doe]`$_{\texttt{SUB}}$ `lives in [Miami]`$_{\texttt{OBJ}}$." 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. Moreover, 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.

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

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.

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 over 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 relation extraction models (on the original TACRED dataset—note that these predictions may include type-incompatible relations which ought 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. Additionally, these were much higher that computed by the original TACRED labels—.54.

While Alt, Gabryszak, and Hennig (2020) 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 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 crowd-

sourced 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 Alt, Gabryszak, and Hennig (2020) and form our main motivation for the work presented in this paper.

3 TACRED Revision

We propose a new crowdsourcing task design aimed at improving over the 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 both Zhang et al. (2017) and Alt, Gabryszak, and Hennig (2020) 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. In Section 3.2, we propose a modification to the previous task design that addresses this issue in a cost-effective manner.
2. Relation Definitions Refinement: Similar to Zhang et al. (2017); Alt, Gabryszak, and Hennig (2020) we initially defined all possible relations according to the TAC KBP documentation[2]. However, we observed that in a small number of cases the documentation is somewhat unintuitive and leaves substantial grey-areas into whether or not certain relations apply for a sentence. This results in confusing the annotators and thus in bad 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 employed a two-step quality assurance process for our annotators, which is described in Section 3.4.
4. Miscellaneous Revisions: During our quality analysis for TACRED we also discovered additional miscellaneous issues (e.g., several sentences are not in the English language). We discuss and address these issues 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 hard for the annotators to identify the right relation among 42 choices. Furthermore, the intended meaning of certain relations is unclear and oftentimes confusing. To this end, in order to reduce annotation complexity, we follow a similar approach to Zhang et al. (2017); Alt, Gabryszak, and Hennig (2020). We first group the TACRED sentences based on the corresponding subject and object types (e.g., the sentence "[Holly]$_{\text{SUB}}$ showed off [her]$_{\text{OBJ}}$ 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 consider all possible pairs of subject and object types and start going through them until the 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 does not overlap with any other super-cluster. Moreover, every sentence group belongs to a super-cluster. We specify these clusters by aggregating groups whose types were most confused with one another in our preliminary investigation over a 1,000-sized random sample of TACRED sentences. Our final super-clusters are shown in Table 4 under Appendix A. We then define each cluster's candidate label set as the union of the candidate sets for each of its sentence group members. This method reduces the worst-case overall annotation cost by a factor of $27/8 = 3.4$. Furthermore, we hypothesize that the cost reduction will be higher empirically because of the fact that the super-clusters are constructed based on the most often confused types and are thus likely to contain the correct relation for cases where the subject and object types are incorrect.

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

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

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.

3.3 Relation Definitions Refinement

In this section, we describe the changes we made to the original TAC KBP relation definitions in order to make them more intuitive and remove any potential sources of confusion for the annotators.

`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]`$_{\texttt{SUB}}$ `shows off a few pieces of [her]`$_{\texttt{OBJ}}$ `jewelry line here`," where the subject and object are denoted as described in Section 2). 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`.

`ORGANIZATION:{MEMBER_OF/MEMBERS}:` The relations `ORGANIZATION:MEMBER_OF` and `ORGANIZATION:PARENTS`, 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]`$_{\texttt{SUB}}$" and "`[Microsoft]`$_{\texttt{OBJ}}$"), 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.

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]`$_{\texttt{SUB}}$ `is a native of [Potomac]`$_{\texttt{OBJ}}$`, 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:LOCATION_OF_HEADQUARTERS:` 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).

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 very 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 $\left(\frac{\text{\# Annotations Approved}}{\text{\# Annotations Completed}}\right) \geq 95\%$. 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, so that they 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 involving tens of thousands of sentences. Thus, we decided to handpick a set of *control* sentences, which we manually annotated ourselves, and mix them with the un-annotated sentences presented to the 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 Miscellaneous Revisions

In addition to the aforementioned modifications to the crowdsourcing task design, we also discovered a couple of issues with the TACRED datasets that we addressed separately. First, we noticed that 1,058 of the TACRED sentences are not written in English (we automated this detection process by using FastText (Joulin et al. 2016), and, 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

In this section we first provide a qualitative comparison between TACRED and our re-annotated version, 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 by using TACRED for evaluating models.

4.1 Qualitative Comparison

In our re-annotated dataset, called Re-TACRED, we have an average agreement rate of 82.3% between the annotators throughout the whole dataset. Moreover, our inter-annotator Fleiss' Kappa over all annotations is .77, indicating high quality. Overall our labels disagreed 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 (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 3.3, 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.

4.2 Model Performance Comparison

We examine how our changes impact the evaluation of several existing relation extraction models and discuss the conclusions reached based on that evaluation. Specifically, we perform an analysis using three existing relation extraction models:

- PALSTM: This model, by Zhang et al. (2017), 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) and part-of-speech (POS) tags, and positional offsets from subjects and object in its reasoning. We refer readers to Zhang et al. (2017) for further information.
- C-GCN: This model, by Zhang, Qi, and Manning (2018), labels sentences by applying a graph-convolution network (GCN) over sentence dependency tree parses. Similar to PALSTM, 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

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**

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.

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: This is a state-of-the-art (SoTA) model is similar to BERT Joshi et al. (2019), but it 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. We present the evaluation results of the three models when using both TACRED and Re-TACRED datasets 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. 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 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 do 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:*` 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 2.

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.

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 `LOCATION` types into respective `PERSON:RESIDENCE`, `PERSON:BIRTH`, `PERSON:DEATH` categories. In a similar manner, `ORGANIZATION:LOCATION` marks all `LOCATION` type relations describing place of `ORGANIZATION` 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 PALSTM and C-GCN performances are difficult to distinguish on TACRED, C-GCN exhibits substantially better performance than PALSTM 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 our refinements have of labels. Moreover, all methods achieve the largest gain in `PERSON:IDENTITY` classifications, and two – PALSTM 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 model 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 the lowest scores for models evaluated on TACRED, illustrating

Model	Dataset	Categories						
		PER:*	ORG:*	PER:ORG	ORG:PER	PER:LOCATION	PER:PER	ORG:ORG
PALSTM	TACRED	66.8	65.2	65.3	72.6	51.9	59.9	59.3
	Re-TACRED	79.0	74.4	62.9	85.1	53.4	85.2	70.3
C-GCN	TACRED	66.5	65.9	66.4	72.2	51.5	49.9	61.6
	Re-TACRED	79.9	76.7	65.3	85.3	54.2	85.3	72.2
SpanBERT	TACRED	69.7	69.5	68.9	74.8	55.9	61.2	68.1
	Re-TACRED	85.6	80.8	78.6	88.6	69.8	88.8	79.0

Table 2: Micro-averaged f1-score for each category in TACRED and Re-TACRED. PER stands for PERSON and ORG for ORGANIZATION.

Model	Dataset	Refined Labels						
		ORG:MEMBERS	ORG:MEMBER_OF	PER:RESIDENCE	PER:BIRTH	PER:DEATH	ORG:LOCATION	PER:IDENTITY
PALSTM	TACRED	23.5	22.6	54.1	31.0	26.7	55.9	0.0
	Re-TACRED	48.8	42.7	55.2	40.5	57.1	68.1	87.8
	Difference	+15.3	+20.1	+1.1	+8.5	+30.4	+12.2	+87.8
C-GCN	TACRED	24.0	24.6	54.1	30.0	25.0	56.7	0.0
	Re-TACRED	51.9	41.7	55.3	43.7	60.0	73.7	88.3
	Difference	+27.9	+17.1	+1.2	+13.7	+35.0	+88.3	+88.3
SpanBERT	TACRED	52.2	50.3	57.6	50.0	26.7	62.7	19.1
	Re-TACRED	64.2	64.0	69.2	68.8	82.1	74.7	90.5
	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.

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% 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 Conclusion

In this paper, we conducted a comprehensive review of the TACRED dataset. We built upon the limitations of previous work by re-annotating the complete dataset using crowdsourcing. Our annotation strategy extended previous label-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 clear task descriptions and high label annotation reliability. Moreover, we perform a thorough analysis of how existing relation extraction methods compare between datasets, and how errors change between them. Perhaps most notably, we observe an average improvement of 13% f1-score of three models on our revised dataset.

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.

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.

Hendrickx, I.; Kim, S. N.; Kozareva, Z.; Nakov, P.; Ó Séaghdha, D.; Padó, S.; Pennacchiotti, M.; Romano, L.; and Szpakowicz, S. 2010. SemEval-2010 Task 8: Multi-Way Classification of Semantic Relations between Pairs of Nominals. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, 33–38. Uppsala, Sweden:

Association for Computational Linguistics. URL https://www.aclweb.org/anthology/S10-1006.

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.

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.

Appendices

A Super-Clusters

Table 4 (below) shows our eight super-clusters over TACRED sentence groups described in Section 3.2. Each group appears in exactly one cluster.

B Re-TACRED Corrected Prediction Errors

Table 5 (below) presents five handpicked sentences showing different types of errors that a SpanBERT model trained on TACRED makes on the Re-TACRED test split, and that the same model trained on Re-TACRED is able to correct on the same split.

C 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 for (i.e. any label not discussed in Section 3.3). 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 $\text{Re-TACRED}_{\texttt{test}}$. Our results are shown in Table 6.

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 Alt, Gabryszak, and Hennig (2020), and suggests that TACRED may be under-estimating model performance, and large improvements can be obtained simply by evaluating models on higher quality annotations.

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

Table 6: Results for multiple RE models (leftmost column) on different train-and-evaluation combinations. Each combination is represented by a pair of the form ("train split", "test split"). For instance, ($\text{TACRED}_{\text{train}}$, $\text{Re-TACRED}_{\text{test}}$) indicates that a method is trained on the TACRED train partition and evaluated on the Re-TACRED test split. The remaining columns show metric results.

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

Table 4: 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.

Error Type	Sentence	TACRED Prediction	Correct Label
Neg → Pos	"…leave of absence from [his]$_{\text{SUB}}$ posts as [Cephalon]$_{\text{OBJ}}$'s chairman and chief executive."	NO_RELATION	PERSON:EMPLOYEE_OF
	"…Pakistani [journalist]$_{\text{OBJ}}$ and Taliban expert [Ahmed Rashid]$_{\text{SUB}}$, from Madrid."	NO_RELATION	PERSON:TITLE
	"…[National Taiwan Symphony Orchestra]$_{\text{SUB}}$ (NTSO) …an [NTSO]$_{\text{OBJ}}$ spokesman…"	NO_RELATION	ORGANIZATION:ALTERNATE_NAMES
Pos → Neg	"[His]$_{\text{SUB}}$ [therapist]$_{\text{OBJ}}$ told him to politely decline, 'which helped."	PERSON:TITLE	NO_RELATION
Pos → Pos	"…[her]$_{\text{SUB}}$ stepchildren, Susan, …, Stephen and [Maggie]$_{\text{OBJ}}$ Mailer; …."	PERSON:SIBLINGS	PERSON:CHILDREN

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. Additionally, each sentence is marked by a specific "error type" (the leftmost column) decribing whether the error is due to predicting a negative sentence label when the correct relation is positive (Neg → Pos), predicting a positive relation when the correct label is negative (Pos → Neg), or inferring the incorrect positive label (Pos → Pos).

Third, methods trained on Re-TACRED$_{\text{train}}$ and evaluated on TACRED$_{\text{test}}$ perform worse than those evaluated on Re-TACRED$_{\text{test}}$. A deeper inspection of the data reveals that such models exhibit significantly fewer correct positively labeled predictions in TACRED$_{\text{test}}$ than in Re-TACRED$_{\text{test}}$, resulting in substantially lower scores. For instance, SpanBERT trained on Re-TACRED$_{\text{train}}$ exhibits 16.5% fewer correct positively labeled instances in TACRED$_{\text{test}}$ compared to Re-TACRED$_{\text{test}}$. This highlights the effects of our label changes described in Section 4.1: 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 TACRED$_{\text{train}}$ achieve performance boosts when testing on Re-TACRED$_{\text{test}}$ (compared to evaluating on TACRED$_{\text{test}}$), training on Re-TACRED$_{\text{train}}$ is critical to achieving the strongest performance on Re-TACRED$_{\text{test}}$.

D 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. 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 Zhang et al. (2017). In addition, we employ the equivalent training strategy as they report in Zhang et al. (2017) (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 Zhang, Qi, and Manning (2018) 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 Zhang, Qi, and Manning (2018) (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 Joshi et al. (2019). We refer readers to Joshi et al. (2019) (detailed in Section 4.2 and Appendix B in their publication) for further details on training strategy.