Question Answering with Additive Restrictive Training (QuAART)

Question Answering for the Rapid Development of New Knowledge Extraction Pipelines

Harper, C.A.; Daniel, R.; Groth, P.

DOI
10.1007/978-3-031-17105-5_4

Publication date
2022

Document Version
Final published version

Published in
Knowledge Engineering and Knowledge Management

Question Answering with Additive Restrictive Training (QuAART): Question Answering for the Rapid Development of New Knowledge Extraction Pipelines

Corey A. Harper[1,2(✉)], Ron Daniel Jr.[1], and Paul Groth[2]

[1] Elsevier Labs, Suite 800, 230 Park Avenue, New York, NY 10169, USA
{c.harper,r.daniel}@elsevier.com
[2] University of Amsterdam, Postbus 94323, 1090 GH Amsterdam, The Netherlands
{c.a.harper,p.t.groth}@uva.nl

Abstract. Numerous studies have explored the use of language models and question answering techniques for knowledge extraction. In most cases, these models are trained on data specific to the new task at hand. We hypothesize that using models trained only on generic question answering data (e.g. SQuAD) is a good starting point for domain specific entity extraction. We test this hypothesis, and explore whether the addition of small amounts of training data can help lift model performance. We pay special attention to the use of null answers and unanswerable questions to optimize performance. To our knowledge, no studies have been done to evaluate the effectiveness of this technique. We do so for an end-to-end entity mention detection and entity typing task on HAnDS and FIGER, two common evaluation datasets for fine grained entity recognition. We focus on fine-grained entity recognition because it is challenging scenario, and because the long tail of types in this task highlights the need for entity extraction systems that can deal with new domains and types. To our knowledge, we are the first system beyond those presented in the original FIGER and HAnDS papers to tackle the task in an end-to-end fashion. Using an extremely small sample from the distantly-supervised HAnDS training data – 0.0015%, or less than 500 passages randomly chosen out of 31 million – we produce a CoNNL F1 score of 73.72 for entity detection on FIGER. Our end-to-end detection and typing evaluation produces macro and micro F1s of 45.11 and 54.75, based on the FIGER evaluation metrics. This work provides a foundation for the rapid development of new knowledge extraction pipelines.

Keywords: Question answering · Named entity recognition · Fine grained entity typing · Knowledge extraction

1 Introduction

It is common to encounter new knowledge extraction tasks for new product lines or projects [19]. New extractions are often needed in domains which are

O. Corcho et al. (Eds.): EKAW 2022, LNAI 13514, pp. 51–65, 2022.
https://doi.org/10.1007/978-3-031-17105-5_4

either too new (e.g. carbon capture and sequestration) or too niche (e.g. material properties for engineering) to have relevant training data or hand-annotated labels. Creating new training data for such tasks is costly and difficult [17].

To tackle this problem, we propose using Question Answering (QA) as a strategy for low cost knowledge extraction with little to no additional training data. While numerous studies have explored the use of language models and question answering techniques for knowledge extraction, in most cases, these models are trained or fine-tuned on data specific to the new task [9,10,12].

In contrast, we start from the hypothesis that using pre-trained QA models with little to no additional training can effectively bootstrap domain specific entity extraction. We investigate this hypothesis, and explore how the addition of small amounts of training data could help lift model performance. This use of incremental addition of training data allows users to understand the trade-off between effectiveness of the model and the need to obtain more data.

Concretely, we start from a QA model trained on SQuAD 2.0 [15], and convert entity extraction and entity typing tasks into a QA format compatible with SQUAD for inference and for additional training. Importantly, to achieve this goal, we design and provide an open-source implementation of a framework for systematically applying QA to solve entity extraction tasks that deals in particular with both null and multiple answers.

To systematically evaluate the performance of QA models and the impact of additional training data for knowledge extraction, we use the task of fine-grained entity recognition and typing [11]. The aim of this task is to determine entity mentions and then assign them a type from a large set of potential predefined types. This task is appropriate as it provides a challenging proxy for real world environments where new long-tail entities need to be recognized.

The contributions of this paper are as follows:

- A framework that maps entity recognition tasks to question answering supporting BIO-type span tagging and that is able to use transformer-based QA models for the prediction of multiple answers per question that effectively deals with nulls. We address entity mention and type detection as an end-to-end problem, a very challenging task that is rarely covered in the literature.
- Measurement of the incremental gains achieved by small amounts of task-specific training data compared to a base SQuAD2.0 trained model in a fine-grained entity recognition setting.

This article is organized as follows. Section 2 describes related work. Section 3 introduces the datasets we employ. Section 4 continues with a discussion of our models, evaluation, and results. In Sects. 5 and 6 we provide a more detailed analysis of our results and reflect on the implications of our work.

2 Related Work

Information extraction in areas with little to no training data is a research area of growing importance [4]. Much work in this area focuses on distant or weak

supervision. We test Question Answering for such low-resources situations, and use Fine-Grained Entity Recognition to evaluate our results. We discuss these two areas in-turn.

Question Answering: Question answering techniques are increasingly being used for information extraction. Perhaps the best known Question Answering dataset is SQuAD, the Stanford Question Answering Dataset [15]. SQuAD 1.1 consists of over 100,000 question answer pairs crowdsourced from hundreds of Wikipedia articles. These are typically used to train systems designed to extract information from text or perform other Natural Language Understanding and Reading Comprehension tasks. SQuAD asks questions about historical events, sports, geography, politics, and many other popular topics. Other datasets have followed, such as Discrete Reasoning Over Paragraphs (DROP) [5], which poses questions about sporting events that include numerical reasoning and comparison, and QAngeroo [20], which requires multi-Hop reasoning across multiple documents to assemble answers. He, et al. [7] reformulated Semantic Role Labeling as such a task. Levy, et al. [9] demonstrate the use of templated question answering for relation extraction. Most closely related to our work, Qi, et al. [12] and Li, et al. [10] show how multi-hop or multi-turn questions can allow machine comprehension models to resolve complex dependencies and compile multiple pieces of related information. We build on these ideas to tackle Fine-grained Entity Mention Detection and Type Detection as a single, end-to-end task.

Fine Grained Entity Mention Detection and Type Detection: Fine-grained Entity Mention Detection and Type Detection is a class of entity recognition task originating in Ling and Weld's 2012 Fine-Grained Entity Recognition (FIGER) paper, which observed that most Entity Recognition datasets were based on a very small number of entity types, plus a catch-all category of MISC [11]. Even some of the larger type vocabularies of the time, such as OntoNotes, only had a few dozen entity types [8]. FIGER addresses this by developing a type vocabulary of 112 much finer grained types, grouped into a two-level hierarchy.

FIGER includes hand-annotated gold data as well as distantly-supervised training data. Additionally, Ling and Weld develop a fine-grained entity recognition system. They report their systems performance on their gold data for end-to-end entity detection and typing, and also report the results of their model given the gold-data segmentation. Most subsequent research that is evaluated on the FIGER gold data only addresses the Fine-Grained Entity *Type Detection* task [2,18]. Additional work has expanded significantly on these type vocabularies, but has again focused only on the entity typing task [3]. Our work builds on the subset of research that uses the FIGER evaluation data to evaluate end-to-end entity *Mention Detection* and *Type Detection* pipelines. This includes Heuristics Allied with Distant Supervision (HAnDS), whose training data we build on [1]. More recently, Rodríguez, et al., also split the task into entity mention detection and type detection, but they treat them as distinct tasks and do not attempt an end-to-end solution [16].

3 QuAART Framework

Figure 1 illustrates our overall Question Answering with Additive Restrictive Training (QuAART) framework. Given a new type, the first step is to construct questions from templates based on the "type" of entity or property sought. Specifically, the question template generates questions in the form of "What was the [*type*]?" for each type in the vocabulary. The resulting questions are then fed to the question answering model with the associated passages of text. The answer to the question are a set of spans of text identifying the entity of the given type encoded in the question.

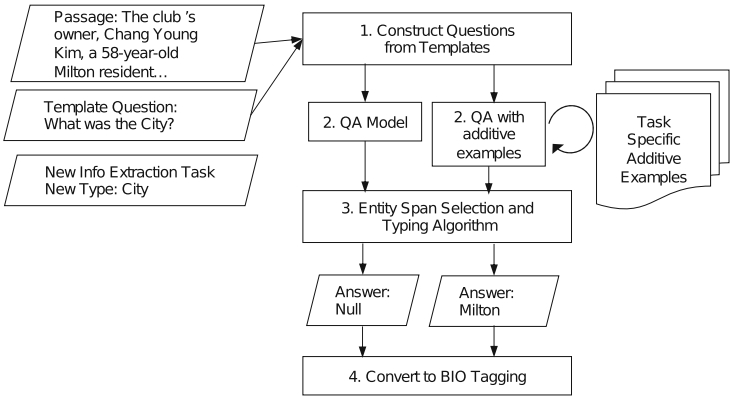


Fig. 1. The QuAART framework

A central component of the framework is knowing when *not* to answer the question, since we ask many questions for which we expect null results. Given the passage in Fig. 1, and the question "What was the spacecraft?", the question is unanswerable because there is no spacecraft mentioned in the passage. An unanswerable question returns a null result. With hundreds of types, QuAART poses hundreds of questions for each passage. It is critical to only produce answers where confidence is high.

To tackle this problem, we devised an algorithm to filter and select the most appropriate spans. The algorithm shown in Listing 1, uses heuristics to remove long answer spans (Line 7) that are likely not the names of entities and prefers entities that appear frequently (Line 13). Note, that this selection is done per text and per templated question.

An important input to the algorithm is a confidence threshold given to the model. QA models typically are not designed for entity detection tasks, so their model confidence thresholds for the predictions are set too low for this task. This results in many null answers for questions. To address this, we empirically determine an appropriate confidence threshold by using a small development set of labelled data. We note this confidence threshold does not necessarily need to be tuned for every new domain.

After entity recognition, the framework converts the results to the standard Begin-Inside-Outside (BIO) tagging system for evaluation. To this point, we have described the framework's use in a setting with a given QA model. However, the framework is also designed to enable the systematic retraining of datasets with task specific training data. Here, the key component is reformatting entity recognition datasets in a format that can be used to fine-tune QA models. We now describe the datasets used in our experiments based on this framework.

Algorithm 1: Entity span selection and typing

input : `QAModel`- A question answering model, that returns a set of answer spans given a passage of text, a question, and a confidence threshold;
`Overlap`- Given a set of answer spans, find and return pairs of spans which overlap;
c - A confidence threshold;
D - Data in the form of a set of text passages;
T - A set of types to recognize

output: R - a map, $D \rightarrow \{(S, T)\}$, that maps each passage to an entity answer span and its associated type. S is the set of possible answer spans.

begin
  $R \leftarrow \emptyset$;
  for $d \in D$ **do**
    for $t \in T$ **do**
      $q \leftarrow$ question template parameterized by t
      $A \leftarrow$ `QAModel` (d, q, c) ;
      `// Remove long answer spans`
      for $a \in A$ **do**
        for $x \in A \setminus a$ **do**
          for $y \in A \setminus a$ **do**
            if $|x| < |a|$ **and** $|y| < |a|$
            and a *is overlapping with* x *and* y **then**
              $A \leftarrow A \setminus a$
      `// Pick a preferred overlapping span`
      foreach $(x, y) \in$ `Overlap`(A) **do**
        if *freq(x)* $>$ *freq(y)* **then** $A \leftarrow A \setminus y$;
        else if *freq(y)* $>$ *freq(x)* **then** $A \leftarrow A \setminus x$;
        else if *freq(y)* $=$ *freq(x)* **then**
          if $|x| > |y|$ **then** $A \leftarrow A \setminus y$;
          else $A \leftarrow A \setminus x$;
      foreach $a \in A$ **do**
        `// Update result with selected type and answer`
        $R[d] \leftarrow +(a, t)$

4 Datasets

Fine grained entity recognition tasks – especially when performed end to end – provide a challenging context for evaluating our framework. The evaluation data from FIGER data is among the most commonly used evaluation datasets in this research space [11]. The HAnDS dataset builds on FIGER, has evaluation data that uses similar types to FIGER, and, importantly, has a distantly-supervised training dataset that corresponds exactly to the type vocabulary in their evaluation data.

We provide a statistical description on FIGER and HAnDS below. Table 1 summarizes this information and Sect. 4 briefly describes the derivative datasets used in our experiments.

Table 1. Statistical summary of FIGER and HAnDS datasets

Dataset	Passage Count	Number of Entities	Distinct Types
FIGER Gold	434	563	43
HAnDS Gold	982	2,420	117
HAnDS Train	31,896,989	37,734,727	117

FIGER Data: The FIGER gold evaluation data consists of 434 sentences tagged with 563 entities using 43 entity types. FIGER also provides distantly-supervised training data generated from Wikipedia anchor texts [11]. This training dataset consists of two million passages. The mentions labeled in these passages use 8,566 distinct types, but not one of these passages limit mentions to the 113 official FIGER types. Given that QuAART only ask questions for, and can therefore only predict, in-vocabulary types, we do not use the FIGER training data. This is in-line with other approaches that use alternative training data and evaluate on FIGER [13].

HAnDS Data: HAnDS uses a type vocabulary of 118 types as opposed to FIGER's 113. The HAnDS types are not an exact superset of FIGER's: nine HAnDS classes are not present in FIGER, while four FIGER classes are not present in HAnDS.

The HAnDS evaluation data consists of 982 passages, split into a dev and test set of 446 and 536 passages respectively. The total evaluation dataset includes 2,420 entities tagged using 117 out of 118 types. The HAnDS training data is much larger than FIGER's, consists of 31 million passages, again from Wikipedia, but with entities tagged using the same 117 types as the evaluation data. Again, the training data is tagged using distant supervision.

Derived Question Answering Training Data: We construct a set of training data useful for fine tuning question answering for entity recognition. Specifically, we randomly select a tiny fraction – less than 0.0015% – of the HAnDS training data to build Question Answering data in a format that is compatible with SQuAD 2.0. This data is built in incremental chunks, adding 87 training *contexts/passages* at a time for 5 sets, totalling 435 passages. After compiling the

first 5 sets, a 6th set was created adding another 34 passages. This additional set was to ensure that the final training data set included positive, answerable examples for all 118 types.

As per the QuAART framework, 118 questions are created, one per type in the HAnDS type vocabulary. The vast majority of the questions are not answerable and have null answers. Since SQuAD does not support multiple correct answers per question, this conversion is not lossless. In cases where there are more than one span of a given type in the source data, the resulting SQuAD-like will be missing some types and may even be missing entire entities. If there are two entities in the passage tagged with the */person* type, only one will be in the training data. Similarly, if there is a *person* entity co-occurring with another entity tagged */person* and */person/artist*, the second entity will only appear for the "Who was the artist?" question.

Table 1 below shows statistical distributions of the 6 training data files. There are always 118 questions per passage, but the vast majority of questions have null answers. The "Non-null questions" questions column counts the questions that have non-null answers. Similarly, non-null types counts the types that are effectively covered by non-null questions in the training set.

Table 2. Counts of HAnDS-specific Passages, Questions, "Possible" Questions, and "Possible" Types

Model	Passage count	Questions	Non-null answers	Non-null Types
SQuAD Only	0	0	0	0
SQuAD + 87	87	10266	159	51
SQuAD + 174	174	20532	320	67
SQuAD + 261	261	30798	517	77
SQuAD + 348	348	41064	725	83
SQuAD + 435	435	51330	889	84
SQuAD + 468	468	55342	1045	117

5 Experimental Method and Results

We run two sets of experiments. Data source information, data conversion scripts, and evaluation scripts as well as information on model training and inference can be found on the QuAART GitHub Repository.[1] In *Experiment 1*, we fine-tune against HAnDS training data and evaluate against both the FIGER and HAnDS evaluation sets.

The HAnDS training data is distantly supervised. In production settings, small amounts of gold labeled data may be more available than large corpora of distantly supervised data. Therefore, it is important to understand the impact of using hand labeled gold data for training. In *Experiment 2*, we construct

[1] https://github.com/elsevierlabs-os/quaart.

train/dev/test splits out of the existing hand annotated FIGER evaluation data. For both experiments, we report two sets of scores:

1. Entity Mention Detection scores - this determines how well the model performs in detecting mentions of entities in text ignoring types. Specifically, we use the Conference on Natural Language Learning (CoNNL) F1 metric treating every entity as type MISC.
2. Entity Type Detection scores - this is the end-to-end performance on the task of recognizing entity mention and assigning an appropriate type. Here, we report FIGER's Strict, Loose Macro F1, and Loose Micro F1 scores, as implemented in Shimaoka, et al. [11,18].

5.1 Experiment 1: Incremental Training with HAnDS

A RoBERTa model fine tuned on SQuAD 2.0 is taken as a base. Progressively larger sets of HAnDS training data are added and the model is fine-tuned from the base for each increment of data. Given the length of training, we only perform one sampling. We provide our splits in the GitHub repository. After each model retraining, predictions are run against dev splits of both HAnDS and FIGER.

At training time, we use max sequence length increased to 512 to support the longer passages found in both the HAnDS and FIGER datasets. Inference also uses a max_seq_length of 512, and an n_best of 10 to slightly constrain the possible sets of answers produced.

As noted in Sect. 3, the HAnDS evaluation data already comes split into dev and test sets. This is *not* the case for FIGER, so a dev split is generated containing slightly more than 10% of the overall evaluation data. For each of the models above, predictions are run against the FIGER and HAnDS dev splits. As discussed in Sect. 3, these dev sets are used to tune post-processing routines and heuristics for generating BIO tagged sequences from the SQuAD Question Answering Results.

Specifically, the standard SQuAD predictions do not fit the use case of fine-grained entity mention detection and type detection, as they assume one answer per passage. Additionally, model confidence thresholds for the predictions are far too low, resulting in almost entirely null answers. For vanilla SQuAD, these thresholds are slightly higher, but they drop significantly after being exposed to thousands of additional null answer examples from the HAnDS training data. Instead of using the predictions as is, we process the n_best prediction sets. This allows for a tuneable prediction threshold that can vary from model to model. More significantly, this provides a mechanism for potentially generating more than one answer per question in cases where multiple entities of the same type exist in one passage. As noted previously, this multiple-entity scenario is common in both the HAnDS and FIGER datasets.

The confidence threshold with the best performance for each model on the dev sets is used when running predictions for the full evaluation sets for both FIGER and HAnDS.

Results: Tables 3 and 4 give the results for both evaluation datasets on both the Entity Mention Detection task, and the Entity and Type Detection task. As a reminder these results are using the HAnDS training data.

Table 3. Entity Mention Detection F1 scores for both FIGER and HAnDS.

Model	F1 FIGER	F1 HAnDS
SQuAD only	0.37	0.47
SQuAD + 87	0.70	0.54
SQuAD + 174	0.72	0.58
SQuAD + 261	0.66	0.59
SQuAD + 348	0.70	0.59
SQuAD + 435	0.74	0.62
SQuAD + 468	0.70	0.63

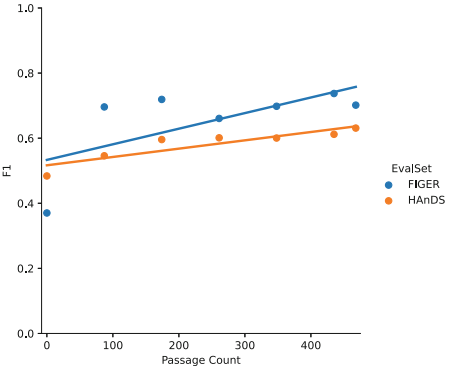


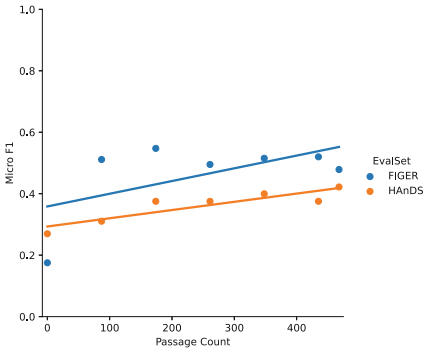
Fig. 2. Mention Detection scores steadily increase with additional training data.

For Entity Mention Detection evaluated on FIGER, the initial increment of training data nearly doubles the scores achieved by the QA model trained on SQuAD alone. Subsequent additions of data offer less improvement, and in some cases lower performance. In the HAnDS dataset, though the initial boost is smaller, the results do continue to rise with each progressive addition of training data. The same trends hold true for the end-to-end detection plus typing results in Table 4.

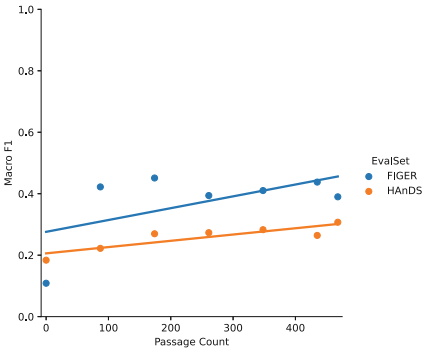
Table 4. End-to-end Detection and Typing scores on FIGER and HAnDS.

	FIGER Evaluation			HAnDS Evalution		
Model	Strict F1	Micro F1	Macro F1	Strict F1	Micro F1	Macro F1
SQuAD Only	0.04	0.18	0.11	0.05	0.26	0.18
SQuAD + 87	0.27	0.51	0.42	0.09	0.31	0.22
SQuAD + 174	0.30	0.55	0.45	0.11	0.36	0.26
SQuAD + 261	0.27	0.50	0.39	0.11	0.36	0.26
SQuAD + 348	0.24	0.52	0.41	0.13	0.39	0.27
SQuAD + 435	0.27	0.52	0.44	0.11	0.37	0.26
SQuAD + 468	0.19	0.48	0.39	0.13	0.42	0.30

To further understand the impact of incremental data, Figs. 2, 3a, 3b fit a linear regression to distributions for CoNNL, FIGER Micro and FIGER Macro F1 Scores. On all three metrics, adding the first iteration of HAnDS training data creates substantive increases in score. This is especially pronounced for the FIGER evaluation scores, which largely level out and even decrease slightly for some of the training data additions.



(a) FIGER Micro F1 scores generally increase with additional training data.



(b) FIGER Macro F1 scores generally increase with additional training data.

Fig. 3. FIGER regressions

These results clearly demonstrate that just using SQuAD provides a reasonable and very low effort starting point for entity detection and fine-grained entity typing in new domains. Only a list of types needs to be prepared. With even a small addition of training data the quality of information extraction improves. Further additions of training data, while still useful, only marginally improve results.

We limited our experiments to the addition of up to 500 passages due to the increasing run-time required for training models. Our largest model takes two to three hours to train. Since each passage results in an addition of one question *per type*, training data can have tens of thousands of questions for only hundreds of passages.

Table 5 shows our best performing models (SQuAD + 174 for FIGER and SQuAD + 468 for HAnDS) compared against the original FIGER and HAnDS papers, which are the only other two studies we are aware of that attempt perform end-to-end Entity Mention Detection and Entity Type Detection on these datasets. We refer to the FIGER model from Ling and Weld [11] as Distant Supervision (DS), and the HAnDS model from Abhishek, et al. [1] as Distant Supervision with Heuristics (DSH). DS uses 2 million training examples and DSH uses 31 million examples to achieve their results. The QuAART approach uses a fraction of that data: 174 passages (0.0005%) of HAnDS training data in our top performing FIGER model, and 468 (0.0015%) when evaluating on HAnDS.

Table 5. End-to-end Detection and Typing scores situated against other systems.

FIGER Evalution Data			
Model	Strict F1	Macro F1	Micro F1
DS [11]	0.47	0.62	0.60
DSH [1]	0.56	0.71	0.68
SQuAD + 174	0.30	0.55	0.45

HAnDS Evalution Data			
Model	Strict F1	Macro F1	Micro F1
DSH [1]	0.53	0.68	0.69
SQuAD + 468	0.13	0.42	0.31

5.2 Experiment 2 – Training with Gold Data

Given that it took remarkably few passages to start seeing viable results in Experiment 1, we wanted to investigate the use of gold training data that was not produced using distant supervision. Experiment 2 was conducted using the FIGER evaluation dataset, which we further subdivided into separate training, development, and test sets. We kept the bulk of the data in test (326 of the 434 total passages), and used dev and training sets of 54 passages each. The training test was further subdivided into 9 random batches of 6 passages each. This splitting was done 3 different times as to limit the effect of specific training examples on the data ablations.

Results: Figure 4 shows our Entity Mention Detection F1, and end-to-end Mention & Typing Macro and Micro F1 scores through all of these shuffles. Similar to above, the incremental addition of small amounts of training data improve performance. It is noteworthy that some of the higher scores come with very little training data. The top mention detection scores are from models trained with only 18 passages of text. As passages are added, mention detection scores drop slightly, while end-to-end mention and type detection scores gain slightly.

6 Discussion

QuAART discusses *restrictive* examples because much of the benefit of the added data is in reigning in false positives on the end-to-end task. For example, the SQuAD only model might predict /person, /person/actor, and /person/musician for an entity with a gold type of only /person. The presence of large numbers of null answers for these rarer types, in the additive examples, reduces the likelihood of these false positives.

Beyond the original SQuAD v.2 paper, which introduced the null answers, little has been written about the value of this null response [14]. To our knowledge, no further investigations have been done into how the generation of null response questions can improve the results of other information extraction tasks.

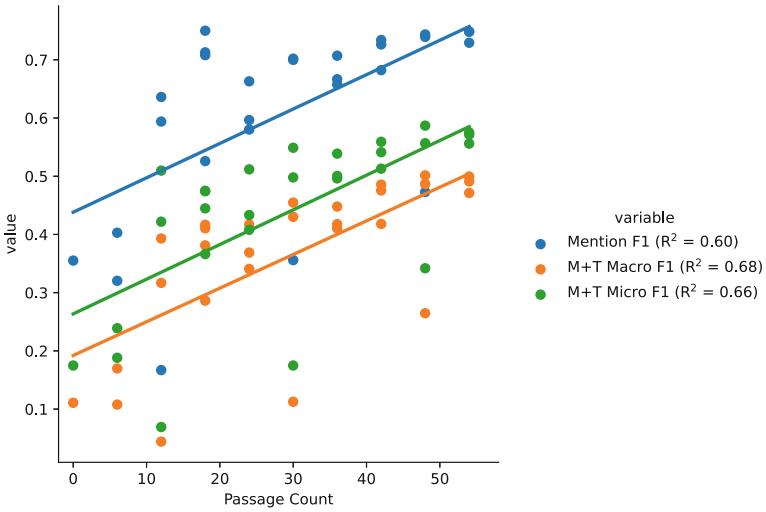


Fig. 4. Cross-validated FIGER scores trained on small amounts of gold data. F1 scores are entity mention detection scores, while Micro and Macro F1 are for end to end entity mention detection and type detection (M+T).

As shown in Table 2, the overwhelming majority of our additive examples are negative examples. On average, only around 1.5% of the questions generated from the HAnDS data have an answer. This tends to reduce the overall number of total predictions, increasing precision, though slightly reducing recall.

While our fine-tuned models achieve much higher scores overall, they also predict fewer classes. On FIGER, our SQuAD only model, predicts 1,090 spans across 94 classes, but only matches 164 spans correctly. SQuAD + 174 predicts 599 spans across only 28 classes, but matches 367 of those spans correctly.

Table 6 isolates scores for a few types of "organization" before and after adding 174 HAnDS passages. For this set of examples, the F1 scores go up for *every* class, regardless of whether the number of predicted spans increase or decrease. Both precision and recall improve substantively.

Table 7 looks more closely at a specific example. The SQuAD only model misses more entities, predicts an erroneous entity, and specifically overpredicts types. The addition of a mere 174 passages of HAnDS examples results in predictions that are much closer to the gold data, and the errors produced by the model – such as location/city for Utah – make much more intuitive sense.

These results show promise for future work. Specifically, we aim to investigate whether using the relationship between fine grained types and more generic types can improve performance. Additionally, there is scope to applying this approach to extract other important knowledge such as relations or attributes [6].

Table 6. Scores for "organization" types before and after adding 174 HAnDS passages (Preds is count of predictions, Matches is count of matching predictions)

Model	Type	Preds	Matches	Strict F1	Micro F1	Macro F1
SQuAD Only	Organization	69	25	0.05	0.14	0.11
SQuAD+174	Organization	123	65	0.14	0.38	0.28
SQuAD Only	Company	25	6	0.00	0.10	0.09
SQuAD + 174	Company	11	5	0.09	0.22	0.14
SQuAD Only	Ed. Institution	24	6	0.02	0.14	0.11
SQuAD + 174	Ed. Institution	5	3	0.04	0.17	0.10
SQuAD Only	Sports League	5	1	0.00	0.04	0.06
SQuAD + 174	Sports League	3	3	0.00	0.32	0.22

Table 7. Example passage, gold data, and predictions from FIGER eval dataset.

Passage: The biggest cause for concern for McGuff is the bruised hamstring Regina Rogers suffered against Utah last Saturday .	
Gold	**McGuff**: /person **Regina Rogers**: /person, /person/athlete **Utah**: /organization,/organization/sports_team **Saturday**: /time
SQuAD Only	**bruised hamstring**: /product,/event/attack, /medical_treatment,/symptom **Regina Rogers**: /person,/person/actor, /person/artist,/person/athlete,/person/soldier **Utah**: /product/game
SQuAD + 174	**McGuff**: /person **Regina Rogers**: /person, /person/athlete **Utah**: /organization,/location,/location/city,/time

7 Conclusion

We present QuAART, a framework for mapping entity recognition tasks to question answering tasks. QuAART includes the construction of questions from templates, an algorithm for selecting high-confidence answers, and a system for mapping back to BIO tags for evaluation. The framework is used to test the performance of question answering models for the task fine-grained entity mention and type detection. We start from a model trained on SQuAD 2.0, and iteratively add small amounts of training data from HAnDS, tracking the improvements achieved through each iteration. We run a second experiment using a small training split of the hand-labeled FIGER evaluation data, which more closely approximates real-world information extraction tasks.

Our results show that question answering can be a viable approach for quickly constructing new knowledge extraction pipelines. Users need only formulate a list of entity types and generate questions in order to extract new information. This is faster and simpler than labelling data or constructing large distantly supervised corpora. Importantly, we show that with only a small amount of domain specific question answering training data performance can be improved allowing users to find a balance between quick construction of a pipeline and extraction performance.

Acknowledgments. The authors would like to thank Curt Kohler and Antony Scerri for various discussions and reviews of this work. This project was funded in-part by Elsevier's Discovery Lab.

References

1. Abhishek, A., Taneja, S.B., Malik, G., Anand, A., Awekar, A.: Fine-grained entity recognition with reduced false negatives and large type coverage. In: Automated Knowledge Base Construction (AKBC) (2019)
2. Chen, Y., et al.: An empirical study on multiple information sources for zero-shot fine-grained entity typing. In: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP 2021), pp. 2668–2678, November 2021
3. Choi, E., Levy, O., Choi, Y., Zettlemoyer, L.: Ultra-fine entity typing. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL 2018), pp. 87–96, July 2018. https://doi.org/10.18653/v1/P18-1009
4. Deng, S., Zhang, N., Chen, H., Xiong, F., Pan, J.Z., Chen, H.: Knowledge extraction in low-resource scenarios: Survey and perspective (2022). https://arxiv.org/abs/2202.08063
5. Dua, D., Wang, Y., Dasigi, P., Stanovsky, G., Singh, S., Gardner, M.: DROP: a reading comprehension benchmark requiring discrete reasoning over paragraphs. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics, pp. 2368–2378, June 2019
6. Harper, C., Cox, J., Kohler, C., Scerri, A., Daniel Jr., R., Groth, P.: SemEval-2021 task 8: MeasEval - extracting counts and measurements and their related contexts. In: Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021), pp. 306–316, August 2021
7. He, L., Lewis, M., Zettlemoyer, L.: Question-answer driven semantic role labeling: Using natural language to annotate natural language. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pp. 643–653, September 2015
8. Hovy, E., Marcus, M., Palmer, M., Ramshaw, L., Weischedel, R.: Ontonotes: the 90% solution. In: Proceedings of the Human Language Technology Conference of the NAACL. NAACL-Short 2006, USA, pp. 57–60 (2006)
9. Levy, O., Seo, M., Choi, E., Zettlemoyer, L.: Zero-shot relation extraction via reading comprehension. In: Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017), pp. 333–342, August 2017
10. Li, X., et al.: Entity-relation extraction as multi-turn question answering. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pp. 1340–1350, July 2019

11. Ling, X., Weld, D.S.: Fine-grained entity recognition. In: Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2012, pp. 94–100. AAAI Press (2012)
12. Qi, P., Lin, X., Mehr, L., Wang, Z., Manning, C.D.: Answering complex open-domain questions through iterative query generation. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language (EMNLP-IJCNLP 2019), pp. 2590–2602, November 2019
13. Qian, J., et al.: Fine-grained entity typing without knowledge base. In: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP 2021), Online and Punta Cana, Dominican Republic, pp. 5309–5319, November 2021
14. Rajpurkar, P., Jia, R., Liang, P.: Know what you don't know: unanswerable questions for SQuAD. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, pp. 784–789, July 2018
15. Rajpurkar, P., Zhang, J., Lopyrev, K., Liang, P.: SQuAD: 100,000+ questions for machine comprehension of text. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, November 2016
16. Rodríguez, A.J.C., Castro, D.C., García, S.H.: Noun-based attention mechanism for fine-grained named entity recognition. Expert Syst. Appl. **193** (2022). https://doi.org/10.1016/j.eswa.2021.116406
17. Roh, Y., Heo, G., Whang, S.E.: A survey on data collection for machine learning: a big data-AI integration perspective. IEEE Trans. Knowl. Data Eng. **33**, 1328–1347 (2021). https://doi.org/10.1109/TKDE.2019.2946162
18. Shimaoka, S., Stenetorp, P., Inui, K., Riedel, S.: Neural architectures for fine-grained entity type classification. In: Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, pp. 1271–1280, April 2017. https://aclanthology.org/E17-1119
19. Surdeanu, M., McClosky, D., Smith, M., Gusev, A., Manning, C.: Customizing an information extraction system to a new domain. In: Proceedings of the ACL 2011 Workshop on Relational Models of Semantics, pp. 2–10, June 2011
20. Welbl, J., Stenetorp, P., Riedel, S.: Constructing datasets for multi-hop reading comprehension across documents. Trans. Assoc. Comput. Linguist. **6**, 287–302 (2018)