Few-Shot Upsampling for Protest Size Detection

Andrew Halterman
Massachusetts Institute of Technology
ahalt@mit.edu

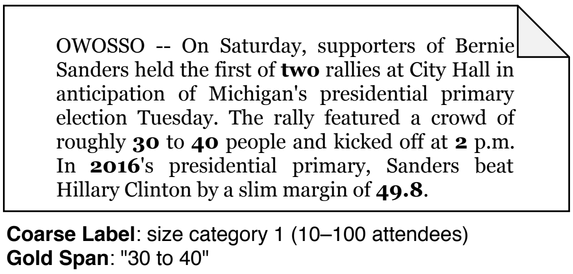
Benjamin J. Radford
UNC Charlotte
benjamin.radford@uncc.edu

Abstract

We propose a new task and dataset for a common problem in social science research: “upsampling” coarse document labels to fine-grained labels or spans. We pose the problem in a question answering format, with the answers providing the fine-grained labels. We provide a benchmark dataset and baselines on a socially impactful task: identifying the exact crowd size at protests and demonstrations in the United States given only order-of-magnitude information about protest attendance, a very small sample of fine-grained examples, and English-language news text. We evaluate several baseline models, including zero-shot results from rule-based and question-answering models, few-shot models fine-tuned on a small set of documents, and weakly supervised models using a larger set of coarsely-labeled documents. We find that our rule-based model initially outperforms a zero-shot pre-trained transformer language model but that further fine-tuning on a very small subset of 25 examples substantially improves out-of-sample performance. We also demonstrate a method for fine-tuning the transformer span on only the coarse labels that performs similarly to our rule-based approach. This work will contribute to social scientists’ ability to generate data to understand the causes and successes of collective action.

1 Introduction

A common data collection task in social science is applying fine-grained labels to documents, including extracting specific passages from text. In many cases, social scientists already have many coarsely-labeled documents and a small number of hand-annotated documents. An automated technique for “upsampling” from coarse labels to more detailed information could help researchers produce better tailored datasets. However, this process does not fit the tools that applied researchers have access to: training a document classifier on coarse labels will not produce the fine-grained answers. Innovations in zero-shot and few-shot classifiers and information extraction (IE) techniques show promise, but new methods are required that can also draw on the existing coarse document annotations to improve fine-grained extraction.


OWOSSO -- On Saturday, supporters of Bernie Sanders held the first of **two** rallies at City Hall in anticipation of Michigan's presidential primary election Tuesday. The rally featured a crowd of roughly **30** to **40** people and kicked off at **2** p.m. In **2016**'s presidential primary, Sanders beat Hillary Clinton by a slim margin of **49.8**.

Coarse Label: size category 1 (10–100 attendees)
Gold Span: "30 to 40"

Figure 1: Documents in our corpus have “coarse labels” reporting the order of magnitude of the protest size and “gold spans” reporting the exact size of the protest. The frequency of number words (in bold) shows why this task is not trivial.

We introduce a new task and dataset for improving information extraction systems’ performance when given many coarsely-labeled documents and a small number of documents annotated with the spans of interest.[1] We draw on a dataset on dissent and collective action (hereafter, “protests”) in the United States compiled by the Crowd Counting Consortium (2020) (CCC) to construct our training and evaluation data. Protests are an important avenue for social change and of major interest for social science researchers. Current work suggests that attendance is a major factor in the success of a protest movement (Chenoweth and Margherita, 2019), but good data on protest attendance is difficult to collect. CCC compiles structured data about protests from expert annotators using news report-

[1] Replication archive available at https://github.com/benradford/few-shot-upsampling-for-protest-size-detection.

Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, pages 3713–3720
August 1–6, 2021. ©2021 Association for Computational Linguistics

ing, including the exact text span from the article that describes the protest's size and the order of magnitude of the crowd size. An example is given in Figure 1. The task we propose is to locate the span within a document that reports the size of a protest, given a training set of documents labeled with the order of magnitude of the protest ("*coarse labels*") and a small number of document pieces (25) with exact span information ("*gold spans*").

Drawing on recent work in question answering, we repurpose existing models to generate fine-grained labels given a large set of coarsely-labeled documents and a small set of documents with fine-grained labels. We provide results from three baseline models, finding that a heuristic, rule-based system outperforms a zero-shot transformer-based question-answering (QA) model. Fine tuning on a small set (25) of gold spans substantially improves performance. We also introduce a new multitask model that reaches equivalent performance despite fine-tuning on *no* gold spans.

2 Task and Data

For each protest in the CCC dataset, we collect the following data: the raw article text (scraped from the CCC-provided URLs), the exact string reporting the protest size, and a "size category" provided by CCC that reports the order of magnitude size of the crowd. The task is to predict the size text string, given plentiful training data with the size category and the gold spans for a small set of partial documents (25 paragraphs). The test set includes only the full article texts and order-of-magnitude information. To make the task tractable, we exclude protests that are coded from multiple documents and documents from which multiple protests are coded. From 48,736 total protests reported by CCC between January 21, 2017 and October 31, 2020, we eliminate multi-document/multi-protest reports and successfully scrape text for 11,005 protests. We eliminate documents where the CCC-reported size text is not located within the document, leaving 3,849 protests/documents. We split these data into four parts:

- **Coarse label training set**: text with coarse, order-of-magnitude labels {0,1,2,3} but no exact answer spans (2,694 full articles).
- **Gold span training set**: short texts with exact answer spans but no order-of-magnitude labels (25 paragraphs).
- **Validation set**: documents with order-of-magnitude labels and exact answer spans (200 full articles).
- **Test set**: documents with order-of-magnitude labels and exact answer spans (930 full articles).

The task is challenging because models are not evaluated on the largest portion of the data (coarse document labels) but rather on a fine-grained span prediction task for which only limited data is available. The task can thus be framed in several ways, depending on which parts of the data are used and in what ways:

- **Zero shot**: use an off-the-shelf model to detect protest sizes without any fine tuning on our data, either coarse or fine.
- **Few-shot on gold spans**: fine tune a baseline model on the small number of gold span labelled data.
- **Coarse labels**: use a coarse-to-fine model to identify spans given only document-level labels.
- **Coarse labels + gold spans**: train a model using both coarse order-of-magnitude labels and limited fine-grained span data.

3 Related Work

The task we propose relates to several strands of research. One framing is as a *question-answering* task (QA), where the same question ("How many people protested?") is asked about each document. A large set of NLP tasks can be framed as question-answering models (McCann et al., 2018) and QA models trained on language models can generalize to new domains with few or no labeled examples (Brown et al., 2020; Radford et al., 2019). QA models have also been successfully used when the training data is noisy (Lin et al., 2018). Given the flexibility of QA models and their strong performance in new domains, we use one as the base of our models.

A different framing is as a *"rationale"* problem for a document classifier. Lei et al. (2016) train a classifier on document-level labels and use attention weights to extract rationales for the classification. Our task differs from the canonical document classification task because a responsive model is evaluated on the extracted spans, not on the coarse label prediction task.

Distant supervision uses noisy labels, often applied automatically or with heuristic labels, to train systems (Ratner et al., 2017). The classic example of distant supervision uses a database of relations to label binary relations in text (Mintz et al., 2009). Weak supervision, more generally, uses labels that are noisy or coarse to train fine-grained models (Khetan et al., 2018; Robinson et al., 2020). Some work on "noisy labels" relates to our task, where labels are presented at a higher level of aggregation rather than with noise. Nayak et al. (2020) propose a model that uses coarse, document-level sentiment labels to train a fine-grained, sentence-level sentiment classifier. Their task differs from ours in the nature of their labels: in moving from document-level to sentence-level labels, they predict labels of the same type (sentiment scores). In our task, we also change the labels themselves, from a crowd size order of magnitude to a token-level label of whether a word describes the exact protest size.

4 Modeling Strategy

We first attempt the task using a rule-based model (the "heuristic keyword model") and an off-the-shelf zero-shot QA system. We then introduce a multi-task neural network model based on a pre-trained transformer language model. We fine-tune and evaluate this model on the coarse labels and gold spans, as well as on noisy labels we generate through a rule-based procedure.

The two standard performance metrics for question answering tasks are exact match and F_1 (Rajpurkar et al., 2018). We compute exact match as the sum of exact matches (predicted spans exactly matched in the set of correct target spans) divided by the total number of documents. We compute F_1 per document based on token-level precision and recall, then average across documents.

4.1 Heuristic Keyword Model

Our heuristic model is a rule-based system that uses keyword matching and dependency parses to return a single number-containing phrase from the article. We first locate all number-containing phrases (digits or number words) in the text with regular expressions. Using a rule-based system, we convert these number phrases to a numeric form (e.g. "several dozen" → 36) and then compare the phrase's numerical value to the protest's reported order of magnitude. If the phrase does not match the order of magnitude, we eliminate it from our candidate list. To further reduce the candidate list, we look for number phrases that occur within the same sentence as a set of keywords such as "crowd", "gathered", or "protesters".[2] If multiple sentences have keyword matches, we return the first one. The CCC data's size spans include modifiers alongside the raw numerical values (e.g. "*about* 20", "*more than* 50"). We use dependency parse information generated by spaCy to extract the wider span.[3]

4.2 Zero-Shot QA Model

We begin with a pre-trained RoBERTa model (Liu et al., 2019) that we subsequently fine-tune for question answering using the Stanford Question Answering Dataset (SQuAD) 2.0 as described in Appendix A (Rajpurkar et al., 2018).[4] The QA model architecture is depicted on the left side of Figure 2. Because we do not tune this model on our dataset, we consider its predictions to be zero-shot.

Model	**Exact**	**F_1**
Heuristic rules	0.54	0.61
RoBERTa QA		
zero-shot	0.17	0.27
+ gold spans	0.67	0.65
+ *coarse labels*	0.48	0.54
+ *coarse labels* + *heuristic spans*	0.66	0.63

Table 1: Exact match ("Exact") and F_1 performance on test set data. All RoBERTa QA and multitask models are fine-tuned on SQuAD 2.0. Multitask models italicized. Full results given in Appendix B.

4.3 Fine-tuned QA Model

To use the coarse labels, we add an additional objective to the QA model that is trained to predict the crowd size order of magnitude. The model first predicts the start and end token vectors for a given context-question pair. We compute the cumulative sum (over tokens) of the predicted start token vector and the reverse cumulative sum for the predicted end token vector. The resulting vectors are element-wise multiplied to produce an attention mask with high values in the range of tokens between the predicted start and end tokens. We apply

[2] The complete list is "protesters", "demonstrators", "gathered", "crowd", "rallied", "attended", "picketed", "protest".

[3] Specifically, (1) for each sentence matching a keyword (2) identify the word in the sentence that is a number word or numeric, and (3) also include child nodes that had the following labels: adjectival modifier, modifier of quantifier, compound, adverbial modifier. We used spaCy version 2.3.2 with the `en_core_web_lg` model to perform the dependency parsing and sentence segmentation.

[4] We use `roberta-base` from HuggingFace (Wolf et al., 2020).

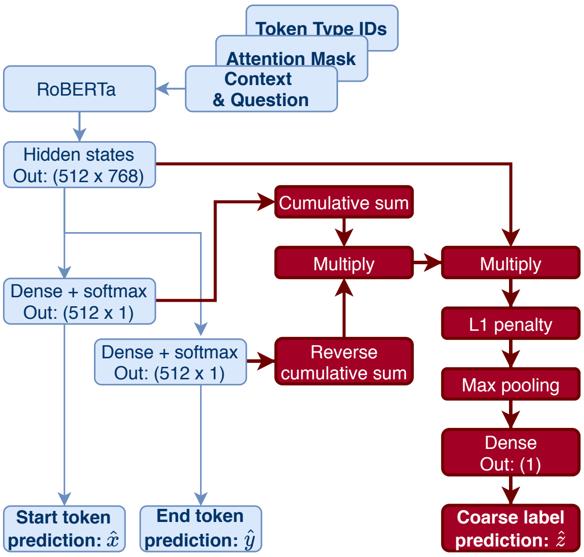

Token Type IDs
Attention Mask
RoBERTa
Context & Question
Hidden states Out: (512 x 768)
Cumulative sum
Multiply
Multiply
Dense + softmax Out: (512 x 1)
L1 penalty
Dense + softmax Out: (512 x 1)
Reverse cumulative sum
Max pooling
Dense Out: (1)
Start token prediction: $\hat{x}$
End token prediction: $\hat{y}$
Coarse label prediction: $\hat{z}$

Figure 2: Multitask model architecture: standard RoBERTa QA (left) and attention mask-based regression for coarse label prediction (right).

an L1 penalty to this mask to ensure the attention focuses on a small number of tokens. The attention mask is then element-wise multiplied with the token hidden states produced by RoBERTa. Global max pooling and a single linear regression layer applied to these attended-to hidden states predict the coarse label (as shown in the right side of Figure 2).

The loss function for the multitask model, an unweighted combination of cross-entropy loss and mean squared error, is $-\sum_{i=1}^{n}\{x_i\log(\hat{x}_i) + y_i\log(\hat{y}_i)\} + (\hat{z} - z)^2$, where $x_i \in \{0, 1\}$ indicates whether token i is the start of an answer span, $y_i \in \{0, 1\}$ indicates whether token i is the end of an answer span, z is the document's coarse label, and n is the number of tokens (512, here). The model can be fit to data including any combination of these three targets.

5 Results

Results on the test set are given in Table 1. RoBERTa QA refers to RoBERTa fine-tuned on SQuAD 2.0. With only fine-tuning on SQuAD 2.0, the model scores 17% exact match accuracy and 27% F_1. On their own, the heuristic-derived spans outperform zero-shot RoBERTa QA. "+ Heuristic spans" indicates that the given model was fine-tuned on the spans identified by the heuristic model.

Fine-tuning the multitask model on the coarse labels alone results in a 180% increase in exact match accuracy and 100% increase in F-score. An example prediction made by the multitask coarse labels model is shown in Figure 3.[5] However, the highest scores are achieved by fine-tuning the RoBERTa QA model on just the 25 gold spans: 67% exact match accuracy and 65% F-score.

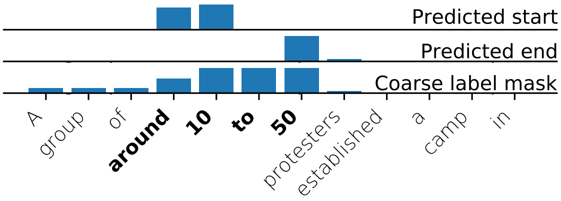


Figure 3: Example target span from document excerpt with predicted start tokens (top), predicted end tokens (middle), and attention mask (bottom). Results from model c in Table 1. Actual span in bold.

The greatest performance by a multitask model without any gold spans is achieved by the model fine-tuned on both the coarse labels and the heuristic spans: 66% exact match and 63% F_1, just below the top performing model with access to the gold spans. We interpret the success of this model and the coarse labels model over the base RoBERTa QA model as evidence that our attention masking strategy was successful at upsampling from coarse document-level labels to specific token-level spans.

6 Discussion and Conclusion

Social scientists often find themselves with coarsely-labeled text data for which upsampling may provide valuable additional information. We anticipate applications in extracting fine-grained policy proposals from party manifestos with document-level annotations (Lehmann et al., 2017), the specific armed actors engaged in civil war violence from documents labeled with "rebel" or "government" (Lyall, 2010), or the specific phrases in news text that lead to their censorship (King et al., 2013). We also see applications in upsampling ranges of causalities from NGO reports or Wikipedia articles to the exact sizes, upsampling years to more specific dates, or using rounded numbers from financial disclosures or government reports as coarse supervision for extracting the exact amount from text.

Improvements in zero- and low-shot models should encourage applied researchers to explore computational approaches to text analysis even when training data is scarce, noisy, or coarse—

[5] The model just misses an exact match by omitting "around" from the predicted span.

common challenges that are often perceived as intractable. At the same time, NLP researchers should continue to improve models that can learn to extract fine-grained information given coarse training data. Multitask QA models show promise in doing so, but future work can further integrate work from the weak/distant supervision literature, including modeling the noisiness of the labels.

Acknowledgments

We thank Jay Ulfelder of the Crowd Counting Consortium for helpful discussions. Halterman gratefully acknowledges the support of an NSF Graduate Research Fellowship.

Impact Statement

Studies of protests have the potential for serious ethical concerns. Some tasks, such as identifying or de-anonymizing the participants in a protest could produce major harms. Our application, identifying the number of attendees at a protest, has less potential for harm. Our collection of information on the size of protests will generally accord with the desires of protesters. Social scientists have long seen protests as an important tool for social movements to overcome collective action problems: by making support for a position visible in the streets, a protest assures potential supporters of the protest that their opinions are held by others and that the group could potentially achieve its ends with more support (Kuran, 1989; Petersen, 2001; Tarrow, 2011). Providing better information on the size of protests furthers the signalling and information-disseminating objectives of the protesters themselves. While we might not agree with the causes of all protesters in the United States, we believe that on-balance, our work benefits those with less power more than it does those with greater power, who can likely already collect the information they seek manually.

The data that we draw on was collected by the Crowd Counting Consortium, which relies on volunteers and paid research assistants to collect the data. Their protocol was reviewed by the University of Denver IRB and deemed exempt because they do not collect personally identifiable information and use only public data.[6]

A second consideration in our work involves the role of copyrighted news text in our project. Our method uses copyrighted news text that we scraped from the web. While scraping websites is legal in the United States,[7] redistributing copyrighted text is more difficult to justify and depends on how the use fits into the fair use doctrine. Balancing copyright holders' rights with public and educational benefit is at the core of the fair use doctrine.[8] Our attempt to balance the harms to copyright holders and the harms to broader public and scientific benefit is to publish a URL list and scraper so that our corpus can be re-created by future researchers. Additionally, in cases where a researcher is attempting to replicate our work for educational purposes, we will make our scraped corpus available for the narrow purpose of replicating our work.

References

Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. 2015. TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org.

Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.

Erica Chenoweth and Belgioioso Margherita. 2019. The physics of dissent and the effects of movement momentum. *Nature Human Behaviour*, 3(10):1088–1095. Copyright - 2019© The Author(s), under exclusive licence to Springer Nature Limited 2019; Last updated - 2020-03-25.

Crowd Counting Consortium. 2020. [link].

Ashish Khetan, Zachary C Lipton, and Animashree Anandkumar. 2018. Learning from noisy singly-labeled data. In *International Conference on Learning Representations*.

Gary King, Jennifer Pan, and Margaret E Roberts. 2013. How censorship in china allows government criticism but silences collective expression. *American Political Science Review*, 107(2):326–343.

[6] https://sites.google.com/view/crowdcountingconsortium/faqs

[7] https://www.eff.org/deeplinks/2019/09/victory-ruling-hiq-v-linkedin-protects-scraping-public-data

[8] https://www.copyright.gov/fair-use/more-info.html

Timur Kuran. 1989. Sparks and prairie fires: A theory of unanticipated political revolution. *Public choice*, 61(1):41–74.

Pola Lehmann, Theres Matthieß, Nicolas Merz, Sven Regel, and Annika Werner. 2017. Manifesto corpus. version: 2017b. Technical report, Berlin: WZB Berlin Social Science Center.

Tao Lei, Regina Barzilay, and Tommi Jaakkola. 2016. Rationalizing neural predictions. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 107–117.

Yankai Lin, Haozhe Ji, Zhiyuan Liu, and Maosong Sun. 2018. Denoising distantly supervised open-domain question answering. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1736–1745.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Jason Lyall. 2010. Are coethnics more effective counterinsurgents? evidence from the second chechen war. *American Political Science Review*, 104(01):1–20.

Bryan McCann, Nitish Shirish Keskar, Caiming Xiong, and Richard Socher. 2018. The natural language decathlon: Multitask learning as question answering. *arXiv preprint arXiv:1806.08730*.

Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2*, pages 1003–1011. Association for Computational Linguistics.

Apoorv Nandan. 2020. Text extraction with bert.

Guruprasad Nayak, Rahul Ghosh, Xiaowei Jia, Varun Mithafi, and Vipin Kumar. 2020. Semi-supervised classification using attention-based regularization on coarse-resolution data. In *Proceedings of the 2020 SIAM International Conference on Data Mining*, pages 253–261. SIAM.

Roger D Petersen. 2001. *Resistance and rebellion: lessons from Eastern Europe*. Cambridge University Press.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for squad. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789.

Alexander Ratner, Stephen H Bach, Henry Ehrenberg, Jason Fries, Sen Wu, and Christopher Ré. 2017. Snorkel: Rapid training data creation with weak supervision. In *Proceedings of the VLDB Endowment. International Conference on Very Large Data Bases*, volume 11, page 269. NIH Public Access.

Joshua Robinson, Stefanie Jegelka, and Suvrit Sra. 2020. Strength from weakness: Fast learning using weak supervision. In *International Conference on Machine Learning*, pages 8127–8136. PMLR.

Sidney G Tarrow. 2011. *Power in movement: Social movements and contentious politics*. Cambridge University Press.

Thomas Wolf, Julien Chaumond, Lysandre Debut, Victor Sanh, Clement Delangue, Anthony Moi, Pierric Cistac, Morgan Funtowicz, Joe Davison, Sam Shleifer, et al. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45.

A Fine-tuning RoBERTa on SQuAD 2.0

A.1 SQuAD 2.0 Fine-Tuning

In order to facilitate extensions to the standard QA model, we perform the fine-tuning of RoBERTa on SQuAD 2.0 ourselves (Abadi et al., 2015). We fine-tune on the SQuAD 2.0 training set for three epochs using the settings recommended by Nandan (2020). We use a batch size of 12 due to memory limitations. We use the Adam optimizer with a learning rate of $5e-5$. Our model achieves 0.78 and 0.74 exact match on the training and evaluation sets, respectively. We use this model only as a basis for subsequent fine-tuning and therefore do not attempt to match state-of-the-art performance on the SQuAD 2.0 evaluation set. The model is trained on two RTX 2080 Ti GPUs. Model size and training time details are provided in Table 2.

We allow the QA model to identify impossible-to-answer questions by predicting the sequence start token (“<s>”) as both the answer span start and end token.

To fit within the RoBERTa base model’s 512 token limit, we pre-process all text inputs via a shingling procedure. We limit contexts to 450 tokens thereby allowing questions of up to 62 tokens in length. We then pad to a uniform 512 tokens. When contexts exceed 450 tokens, we use a sliding window of 450 tokens that we step through the context 225 tokens at a time. We guarantee all samples generated from large contexts contain precisely 450 tokens by adjusting the first and last window positions such that they do not extend before or after the first or last context token, respectively. We aggregate predictions across shingles by assuming one predicted span per document and selecting the predicted span from the shingle for which $\max_{i\in[1,\ldots,512]}(\hat{x_i})+\max_{i\in[1,\ldots,512]}(\hat{y_i})$ is the greatest.

A.2 Task-Specific Fine-Tuning

The selection of learning rate for these models, 5e-6 (exactly one order of magnitude lower than the default used for SQuAD fine-tuning), was due to our sensitivity to overfitting on the very small set of span examples. All models were trained for 150 batches, each batch comprising 12 samples chosen from the training datasets with replacement. When multiple datasets are used to train the same model, batches alternate between them. We selected the number of batches for training by observing exact match accuracy on the validation set over a range of iteration steps from 1 to 400 and selecting the earliest batch iteration at which validation set accuracy appeared to plateau.

B Results

The full set of fine-tuning data combinations is given in Table 3. All models c through i are trained using the same hyperparameters and strategy (Adam optimizer, 5e-6 learning rate, and 150 batches of size 12 examples each).

		Parameters	Training Time
(a)	Heuristic rules	–	–
(b)	RoBERTa + SQuAD 2.0 (zero-shot)	1.25M	200 min
(c)	+ coarse labels	1.25M	+ 20 min
(d)	+ heuristic spans	1.25M	+ 20 min
(e)	+ coarse labels + heuristic spans	1.25M	+ 20 min
(f)	+ *gold spans*	1.25M	+ 20 min
(g)	+ *gold spans* + coarse labels	1.25M	+ 20 min
(h)	+ *gold spans* + heuristic spans	1.25M	+ 20 min
(i)	+ *gold spans* + coarse labels + heuristic spans	1.25M	+ 20 min

Table 2: Model size in parameters. Training time (approximate) on 2× RTX 2080 Ti GPUs. "+ 20 min" indicates the model takes an additional 20 minutes to fine-tune after the initial fine-tuning on SQuAD 2.0. These estimates may be high due to our validation set performance evaluation between batches.

		Test set		Validation set	
		Exact Match	F_1	Exact Match	F_1
(a)	Heuristic rules	0.54	0.61		
(b)	RoBERTa + SQuAD 2.0 (zero-shot)	0.17	0.27	0.19	0.27
(c)	+ coarse labels	0.48	0.54	0.54	0.58
(d)	+ heuristic spans	0.51	0.50	0.56	0.51
(e)	+ coarse labels + heuristic spans	0.66	0.63	0.72	0.66
(f)	+ *gold spans*	0.67	0.65	0.71	0.68
(g)	+ *gold spans* + coarse labels	0.67	0.65	0.68	0.66
(h)	+ *gold spans* + heuristic spans	0.62	0.61	0.66	0.62
(i)	+ *gold spans* + coarse labels + heuristic spans	0.65	0.64	0.72	0.67

Table 3: Exact match and token-level F_1 performance by each model on test and validation set data.