Reference-based Weak Supervision for Answer Sentence Selection using Web Data

Vivek Krishnamurthy[*], **Thuy Vu,** and **Alessandro Moschitti**
University of California, Los Angeles, CA, USA
Amazon Alexa AI, Manhattan Beach, CA, USA
vivek.k.murthy93@gmail.com, {thuyvu; amosch}@amazon.com

Abstract

Answer Sentence Selection (AS2) models are core components of efficient retrieval-based Question Answering (QA) systems. We present the Reference-based Weak Supervision (RWS), a fully automatic large-scale data pipeline that harvests high-quality weakly-supervised answer sentences from Web data, only requiring a question-reference pair as input. We evaluated the quality of the RWS-derived data by training TANDA models, which are the state of the art for AS2. Our results show that the data consistently bolsters TANDA on three different datasets. In particular, we set the new state of the art for AS2 to P@1=90.1%, and MAP=92.9%, on WikiQA. We record similar performance gains of RWS on a much larger dataset named Web-based Question Answering (WQA).

1 Introduction

Creating datasets for AS2 (Wang et al., 2007), a core task for QA, requires expensive hand-labeling work. We propose the Reference-based Weak Supervision (RWS), a fully automatic data pipeline to harvest high quality answers from the Web. RWS operates in two stages: (i) collecting answer candidates from Web documents, and (ii) automatically assigning them correct or incorrect labels. More specifically, we build a large index of more than 100MM Web documents from Common Crawl's crawls. Given a question-reference pair, the question is used as a query to retrieve a set of relevant documents from the index. Then, we extract sentences from those documents to build a large pool of answer candidates, which are finally scored by an automatic evaluator based on the provided reference. We use the AVA approach, which we recently introduced in Vu and Moschitti (2021) for automatic evaluation of AS2.

[*]Work done while the author was an intern at Amazon Alexa AI.

We show that RWS complements the original data (question/answer pairs) by measuring the improvement over the state-of-the-art AS2 models on WikiQA and TREC-QA datasets. The experimental results suggest that the weakly supervised data produced by RWS adds new *supervision capacity* to the original dataset, enabling models to advance the state of the art.

In a nutshell, our contributions include: (i) a pipeline for processing large-scale data, which generates labeled question-answer pairs using publicly available Web data, i.e., Common Crawl; and (ii) a large automatically labelled dataset derived from the data and labels of ASNQ (Garg et al., 2020) with RWS.

2 Background

In this section we provide the background of our work. We first describe AS2 task formally, and then introduce TANDA, the current state-of-the-art model for AS2 (Garg et al., 2020). Finally, we present AVA employed in our pipeline.

2.1 Answer Sentence Selection (AS2)

AS2 can be modeled with a classifier scoring the candidate sentences as follows: Let q be a question, $T_q = \{t_1, \ldots, t_n\}$ be a set of answer candidates for q, we define a ranking function, $\mathcal{R}$, which orders the candidates in T_q according to a score, $p\left(q, t_i\right)$, indicating the probability of t_i to be a correct answer for q. Popular methods modeling $\mathcal{R}$ include Compare-Aggregate (Yoon et al., 2019), inter-weighted alignment networks (Shen et al., 2017), and Transformers (Garg et al., 2020).

2.2 TANDA: Fine-tuning for AS2

Fine-tuning a general pre-trained model to a target application is a recent topic of interest (Gururangan et al., 2020). Specifically, for AS2, Garg et al. (2020) introduced TANDA, a fine-tuning technique using multiple datasets. TANDA transfers a general

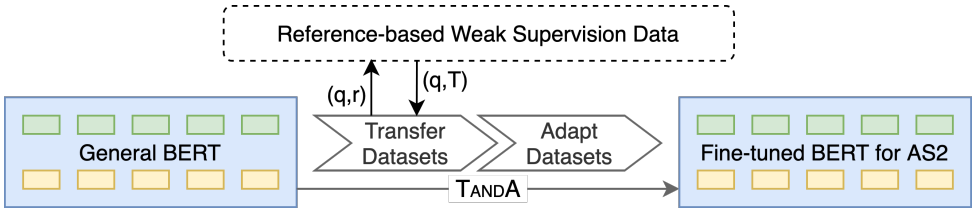

Figure 1: RWS's generated data applied in TANDA.

q:	Where is the world second largest aquarium?
r:	Located in the Southeast Asian city-state of *Singapore*, Marine Life Park contains twelve million gallons of water, making it the second-largest aquarium in the world.
t:	The Marine Life Park, situated in southern *Singapore*, was the largest oceanarium in the world from 2012 to 2014, until it was surpassed by Chimelong Ocean Kingdom.

Table 1: A sample input for the automatic evaluator, which compares the semantic similarity between a reference r and an answer candidate t, biased by q.

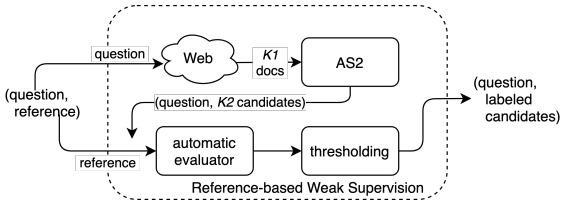

Figure 2: RWS takes as input a (question, reference) pair and produces weakly supervised (question, answer) pairs. It consists of 4 steps: retrieval, candidate selection, automatic evaluation, and thresholding.

pre-trained Transformer model to one, specialized to AS2 a target domain. Then, with a second fine-tuning, it transfers the obtained model to a specific domain. This approach achieved state-of-the-art results on multiple AS2 benchmarks. Thus, we study and validate the impact of RWS in the TANDA setting to compare with the best models.

Figure 1 describes how RWS is integrated in TANDA. In short, given a Transformer, e.g., BERT, we first fine-tune it with general datasets, including weakly supervised data, and then adapt it to the target domain using the AS2 domain specific data.

Semantic Evaluator for AS2 AVA is a recent approach to automatically measure the *correctness* of an answer t_i with respect to a question q, using a reference answer r. Formally, it is modeled as a function: $\mathcal{A}(q, r, t_i) \rightarrow \{0, 1\}$, where the output is a binary correct/incorrect label. Table 1 shows an example input for $\mathcal{A}$.

Weakly Supervised Data Creation Distant supervision has gained success in creating *weakly labeled data* for both relation extraction (Mintz et al., 2009; Jiang et al., 2018; Qin et al., 2018) and machine reading (Joshi et al.; Kočiský et al., 2018), using curated entity relation database. Unlike others, we use abundant Web data and reference answers to create weakly label data. We also argue that we are the first to address this research in AS2 context.

3 Reference-based Weak Supervision

Data Generation Pipeline We describe our proposed RWS pipeline for AS2. The process starts from q, and r, i.e., a valid response to q.

First, we retrieve top K_1 documents relevant to q from an index of Web data. The documents are split into sentences, which are later re-ranked by a reranker.

Second, we select the top K_2 sentences as candidate, $T_q = \{t_1, \ldots, t_n\}$. We create the triples of (q, r, t_i) $\forall t_i \in T_q$ to be input to AVA, which in turns provides the scores for them.

Finally, we apply a threshold on the scores of t_i to generate its positive or negative label. The entire process is exemplified by Figure 2.

AVA as an Automatic Labeler AVA is designed to classify an answer to a question as correct or incorrect like an AS2 model does, but it exploits the semantic similarity between t and r, conditioned by q.

We studied multiple configurations to optimize AVA for our task of generating weakly supervision. In our experiments, we use the best setting we found in (Vu and Moschitti, 2021), which uses a Transformer-based approach with Peer-Attention, to model the interaction among q, t, and r.

We built AVA using a dataset of 245 questions, each having roughly 100 annotated answers. The number of correct and incorrect answers are 5.3K and 20.7K, respectively. This generates approximately 500K point-wise training examples for AVA.

We verified that our training set is disjoint with respect to all datasets studied in this paper to generate weakly supervised data.

4 Experiments

We study the efficacy of RWS by testing its impact on TANDA models for AS2. We first describe our experimental setup, datasets, and then apply RWS to AS2-NQ. We report the results of TANDA when RWS's data is used during the transfer stage.

4.1 Setup

Large Web Index Having the ability to query a large index of Web documents is required in our data pipeline. In particular, we need to retrieve a large number of documents, given a question, and we also process hundreds of thousands of questions. As public search engines do not allow for such large-scale experimentation, we created our search engine constituted by a large index of more than 100MM English documents, collected from 19 Common Crawl's crawls from 2013 to 2020. We will make this index available to the community to enable similar retrieval activities.

Parameter Settings We employ two standard pre-trained models in our experiments: RoBERTa (Liu et al., 2019) and ELECTRA (Clark et al., 2020). We verify our findings on both Base and Large configurations. We use HuggingFace's Transformer library (Wolf et al., 2020) and set the learning-rates to $1e{-}6$ and $1e{-}5$ for the transfer and adapt stages of TANDA, respectively, across all experiments. The other hyper-parameters are set to default values. Specifically, all experiments share the same hyper-parameter setting, including the default random seed of the transformers library (i.e., 42). We also performed the experiments with 5 random seeds and averaged the results.

4.2 Datasets

We evaluated the impact of RWS on AS2 using the two most popular public datasets: WikiQA and TREC-QA. Additionally, we measured the impact of RWS on a larger dataset we built internally, and we created AS2-NQ by extending ASNQ. AS2-NQ has 47% more questions than ASNQ, taken from the NQ dataset (Kwiatkowski et al., 2019). We execute RWS with question-reference pairs from AS2-NQ and name the produced dataset RWS for simplicity.

Dataset	Split	#Q	#A	#A$^+$	#A$^-$
WikiQA	Train	873	8,672	1,040	7,632
	Dev	121	1,126	140	990
	Test	237	2,341	293	2,058
TREC-QA	Train	1,227	53,417	6,403	47,014
	Dev	65	1,117	205	912
	Test	68	1,442	248	1,194
WQA	Train	4,978	206,249	42,963	163,286
	Dev	904	22,600	6,157	16,443
	Test	1,000	24,953	6,366	18,587

Table 2: Statistics for WikiQA, TREC-QA, and WQA dataset: total number of questions (#Q), answers (#A), correct and incorrect (#A$^+$ and #A$^-$) for each split: Train, Dev, and Test.

TREC-QA is a traditional benchmark for the AS2 task (Wang et al., 2007). We use the standard split used in previous work, e.g., (Tan et al., 2015; Rao et al., 2016; Garg et al., 2020).

WikiQA The dataset, introduced by Yang et al. (2015), consists of questions from Bing query logs and answers extracted from a *user-clicked* Wikipedia page returned by Bing. We follow the standard setting used in previous work, e.g., (Yoon et al., 2019; Tay et al., 2017; Garg et al., 2020).

Web-based Question Answering (WQA)[1]. We built the dataset as part of the effort to improve understanding and benchmarking in open-domain QA systems. The creation process includes the following steps: (i) given a set of questions we collected from the web, a search engine is used to retrieve up to 1,000 web pages from an index containing hundreds of millions of pages. (ii) From the retrieved documents, all candidate sentences are extracted and ranked using AS2 models. Finally, (iii) top candidates for each question are manually assessed as correct or incorrect by human judges. This allowed for obtaining a higher average number of correct answers with a richer variety from multiple sources, as shown in Table 2.

AS2-NQ Current public benchmark datasets for AS2, e.g., TREC-QA and WikiQA, are relatively small and mainly used in the adapting step of TANDA. The prior step, transferring from general pre-trained Transformer models, requires a significant large and accurate general domain dataset to be effective. We created AS2-NQ by extending ASNQ (Garg et al., 2020) in order to maximize the potential at the transferring step in TANDA.

[1]The public version of WQA will be released in the short-term future. Please search for a publication by Thuy Vu and Alessandro Moschitti, with title *WQA: A Dataset for Web-based Question Answering Tasks* on arXiv.org.

Dataset	#Q	#A	#A$^+$	#A$^-$
ASNQ	57,242	20,745,240	60,285	20,684,955
AS2-NQ	84,121	27,208,065	86,756	27,121,309
RWS	84,089	2,103,027	69,945	2,033,082

Table 3: Total number of questions (#Q), answers (#A), correct and incorrect (#A$^+$ and #A$^-$) of ASNQ, AS2-NQ, and the weakly-supervised dataset generated from AS2-NQ via our RWS pipeline.

Specifically, we extracted question-answer candidate pairs from NQ, a large scale dataset intended for machine reading (MR) task. Each question in NQ is associated with a Wikipedia page, a long answer paragraph (`long_answer`) containing the answer extracted from the page. Each `long_answer` may contain answer phrases annotated as `short_answer`. A `long_answer` consists of multiple sentences, thus NQ is not directly applicable for AS2.

To obtain an AS2 dataset, for each question, we consider the sentences that occur in the long answer paragraphs in NQ and contain *annotated* short answers, as correct answers . The remaining sentences from the document are labeled as negative for the target question. The negative examples can be of the following types:

1. Sentences from the document that are in the `long_answer` but do not have annotated short answers. It is possible that these sentences might contain strings matched with the `short_answer`.

2. Sentences from the document that are not in the `long_answer` but contain the `short_answer` string, that is, such occurrences are plausible but mainly irrelevant.

3. Sentences from the document that are neither in the `long_answer` nor contain the `short_answer`. Since this set is extremely large, we sub-sampled to an amount equivalent to the previous sets.

As a result, AS2-NQ has more than ∼84K questions, i.e., 27K more questions than ASNQ, each having typically one reference answer. The dataset will be released together with the paper. The first two rows in Table 3 show the statistics of ASNQ and AS2-NQ, respectively.

We verified the quality of the new dataset by comparing TANDA models trained with ASNQ and AS2-NQ. In particular, Table 4 reports the results of the models when transferred on ASNQ or AS2-NQ, measured on WikiQA and TREC-QA. The results suggest that the end-to-end performance gain given by AS2-NQ is negligible, although 47% more data is added. This indicates that the accuracy gain with respect to the increase of the amount of training data (from NQ) has reached a plateau. However, in Sec. 4.3, we show that our weakly supervised data from RWS improves accuracy.

TANDA	Transfer on	WikiQA		TREC-QA	
		MAP	MRR	MAP	MRR
RoBERTa-Base	ASNQ (2020)	0.889	0.901	**0.914**	**0.952**
	AS2-NQ	**0.898**	**0.910**	0.908	0.938
	% diff.	+1.01	+0.99	-0.66	-1.52
RoBERTa-Large	ASNQ (2020)	0.920	0.933	**0.943**	0.974
	AS2-NQ	**0.923**	**0.935**	0.936	**0.975**
	% diff.	+0.33	+0.23	-0.73	+0.15

Table 4: TANDA's performance on two datasets ASNQ and AS2-NQ using RoBERTa Base and Large. % diff. reports the percentage differences.

RWS We apply RWS to AS2-NQ following these steps: First, we collect question-reference pairs from AS2-NQ by using only pairs with correct answers. We set K_1 and K_2 at 1,000 and 25, i.e., for each question, we run a query and select 1,000 relevant documents from our Elasticsearch index. This typically generates a set of 10,000 candidates. Then, we select the 25 most probable candidates using an off-the-shelf AS2 reranker tuned on ASNQ by Garg et al. (2020). While a large number of questions are shared between ASNQ and AS2-NQ, the candidates from our index are disjoint. We apply AVA to label each triple, (q, r, t_i), thus generating labelled pairs, (q, t_i). A pair is labeled as correct if its AVA score, produced by $\mathcal{A}(q, r, t_i)$, is at least 0.9, otherwise it is labeled as incorrect.

4.3 Integrating RWS into TANDA

We study the contribution of RWS in fine-tuning models for AS2. Specifically, we compare the following transfer configurations for TANDA. First, we report the baselines using (i) vanilla **BERT** Base and Large models without transferring data; and (ii) TANDA-RoBERTa transferred with **ASNQ**. We then replace ASNQ (iii) by **AS2-NQ** and (iv) by **RWS** at transfer stage, measuring the results of each transfer. Finally, we use both datasets, AS2-NQ and RWS, at transfer stage in the following orders: **AS2-NQ→RWS** and **RWS→AS2-NQ**. We use precision at 1 (P@1), mean average precision (MAP), and mean reciprocal rank (MRR) as evalu-

PT	Transfer on	WikiQA		TREC-QA		WQA	
		P1	MAP	P1	MAP	P1	MAP
BERT-Base (2020)		-	0.813	-	0.857	-	-
Roberta-Base	ASNQ (2020)	-	0.893	-	0.914	-	-
	AS2-NQ	0.852	0.898	0.882	0.908	0	0
	RWS	0.716	0.809	0.868	0.878	−0.76	−2.14
	RWS→AS2-NQ	0.852	0.897	0.897	0.903		
	% diff.	0.00	−0.09	+1.67	−0.58	**+1.13**	+0.38
	AS2-NQ→RWS	**0.864**	**0.907**	**0.926**	**0.916**		
	% diff.	+1.43	+1.00	+4.95	+0.88	+0.76	**+0.71**
Electra-Base	ASNQ (2020)	-	-	-	-	-	-
	AS2-NQ	0.831	0.887	0.882	0.886	0	0
	RWS	0.712	0.807	0.838	0.827	−3.15	−3.54
	RWS→AS2-NQ	**0.864**	**0.900**	**0.912**	**0.911**		
	% diff.	+3.97	+1.47	+3.40	+2.82	0.00	**+1.24**
	AS2-NQ→RWS	0.835	0.89	**0.912**	0.893		
	% diff.	+0.48	+0.34	+3.40	+0.79	−0.56	+0.03
BERT-Large (2020)		-	0.836	-	0.904	-	-
Roberta-Large	ASNQ (2020)	-	0.904	-	0.943	-	-
	AS2-NQ	0.893	0.923	**0.956**	0.936	0	0
	RWS	0.802	0.871	0.941	0.918	−1.25	−1.49
	RWS→AS2-NQ	**0.901**	**0.929**	0.912	0.918		
	% diff.	+0.90	+0.65	−4.60	−1.92	**+0.36**	−0.22
	AS2-NQ→RWS	0.889	0.922	**0.956**	**0.94**		
	% diff.	−0.45	-0.11	0.00	+0.43	−1.07	−0.09
Electra-Large	ASNQ (2020)	-	-	-	-	-	-
	AS2-NQ	0.872	0.909	0.941	0.941	0	0
	RWS	0.844	0.894	0.897	0.922	−0.18	−0.27
	RWS→AS2-NQ	**0.885**	**0.92**	0.926	0.938		
	% diff.	+1.49	+1.21	−1.59	−0.32	+0.72	**+0.75**
	AS2-NQ→RWS	0.885	0.918	**0.956**	**0.944**		
	% diff.	+1.49	+0.99	+1.59	+0.32	**+1.26**	+0.64

Table 5: Experimental results of different TANDA settings on WikiQA, TREC-QA, and WQA. **% diff.** indicates the relative performance (in %) compared to the TANDA fine-tuned on the same AS2-NQ dataset. For WQA dataset, we report only the relative performance to comply with customer data handling guidance.

ation metrics.

General results Table 5 shows that RWS used alone does not improve the baselines trained on ASNQ or AS2-NQ. This is intuitive as the quality of weakly supervised data is supposed to be lower than supervised data. However, when RWS is used as the first level of fine-tuning (i.e., TANDA approach), for any dataset and any model (see model RWS→*), we observed a significant improvement. In particular, when RWS→AS2-NQ is used with RoBERTa-Large, the model establishes the new state of the art in AS2.

WikiQA RWS achieves additional performance gains when combining it with AS2-NQ during the transfer steps. In particular, we note 1%–4% performance gains over the TANDA transferred on AS2-NQ. On WikiQA, it seems better using RWS before AS2-NQ, i.e., RWS→AS2-NQ.

TREC-QA Using RWS during the transfer step improves the performance on TREC-QA. While the measures are better over the baselines, i.e., using ASNQ or AS2-NQ alone, we observe a different transferring trend. Specifically, it seems more beneficial to transfer RWS later, i.e., AS2-NQ→RWS. We conjecture that this is due to the differences between WikiQA and TREC-QA. That is, the former is very similar to AS2-NQ and ASNQ, thus the best accuracy on WikiQA should be obtained by using RWS first. In contrast, TREC-QA is more general, thus it can better benefit from having RWS, a similar dataset, in the second step of fine-tuning.

The absolute improvement is low as it is obtained over the highest results (state of the art) on datasets that are popular and well studied: these numbers are rather high, 85-95%. This means that the improvement of RWS measured as relative error reduction is large. For example, with RoBERTa-Large model we achieve an improvement of 1.5 absolute points, from 94.1% to 95.6%, with an error reduction of 25%.

WQA We also record similar performance gains of RWS when combining it with AS2-NQ during the transfer steps benchmarked on this much larger dataset. In particular, there is ∼1% relative performance gains over the TANDA transferred on AS2-NQ. This indicates the quality of the data with respect to a harder benchmark.

5 Conclusion

We have presented RWS a fully automatic data pipeline for AS2 that creates a large amount of weakly labeled question-answer pairs from question-reference pairs. This data is showed to benefit AS2 models. Specifically, we recorded significant performance gains on both popular public benchmarks, WikiQA and TREC-QA, and our internal dataset WQA, which is several times larger. In a nutshell, the key motivation of RWS is to make use of abundant Web data to find more relevant answers for a question. We believe RWS can benefit other applications besides AS2.

We will make our three new datasets, AS2-NQ, WQA and RWS, as well as our index using CommonCrawl data available at `github.com/alexa/wqa_dataset`. We believe this data will enable further research on retrieval-based QA and data creation with weakly supervised techniques.

References

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. Electra: Pre-

training text encoders as discriminators rather than generators. In *ICLR 2020*.

Siddhant Garg, Thuy Vu, and Alessandro Moschitti. 2020. TANDA: transfer and adapt pre-trained transformer models for answer sentence selection. In *AAAI 2020*.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *ACL 2020*.

Tingsong Jiang, Jing Liu, Chin-Yew Lin, and Zhifang Sui. 2018. Revisiting distant supervision for relation extraction. In *LREC 2018*.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *ACL 2017*.

Tomáš Kočiský, Jonathan Schwarz, Phil Blunsom, Chris Dyer, Karl Moritz Hermann, Gábor Melis, and Edward Grefenstette. 2018. The NarrativeQA reading comprehension challenge. *TACL 2018*, 6:317–328.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Matthew Kelcey, Jacob Devlin, Kenton Lee, Kristina N. Toutanova, Llion Jones, Ming-Wei Chang, Andrew Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: a benchmark for question answering research. *TACL 2019*.

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.

Mike Mintz, Steven Bills, Rion Snow, and Daniel Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *ACL-IJCNLP 2009*, Suntec, Singapore.

Pengda Qin, Weiran Xu, and William Yang Wang. 2018. Robust distant supervision relation extraction via deep reinforcement learning. In *ACL 2018*.

Jinfeng Rao, Hua He, and Jimmy Lin. 2016. Noise-contrastive estimation for answer selection with deep neural networks. CIKM 2016.

Gehui Shen, Yunlun Yang, and Zhi-Hong Deng. 2017. Inter-weighted alignment network for sentence pair modeling. In *EMNLP 2017*, pages 1179–1189, Copenhagen, Denmark.

Ming Tan, Bing Xiang, and Bowen Zhou. 2015. Lstm-based deep learning models for non-factoid answer selection. *CoRR*, abs/1511.04108.

Yi Tay, Anh Tuan Luu, and Siu Cheung Hui. 2017. Enabling efficient question answer retrieval via hyperbolic neural networks. *CoRR*, abs/1707.07847.

Thuy Vu and Alessandro Moschitti. 2021. AVA: an automatic evaluation approach to question answering systems. In *NAACL 2021*.

Mengqiu Wang, Noah A. Smith, and Teruko Mitamura. 2007. What is the Jeopardy model? a quasi-synchronous grammar for QA. In *EMNLP-CoNLL 2007*, Prague, Czech Republic.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *EMNLP 2020: System Demonstrations*.

Yi Yang, Wen-tau Yih, and Christopher Meek. 2015. Wikiqa: A challenge dataset for open-domain question answering. In *EMNLP 2015*, pages 2013–2018.

Seunghyun Yoon, Franck Dernoncourt, Doo Soon Kim, Trung Bui, and Kyomin Jung. 2019. A compare-aggregate model with latent clustering for answer selection. *CoRR*, abs/1905.12897.