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.

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, \( R \), which orders the candidates in \( T_q \) according to a score, \( p(q, t_i) \), indicating the probability of \( t_i \) to be a correct answer for \( q \). Popular methods modeling \( R \) 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.

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 \( q \).

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 $1 \times 10^{-6}$ and $1 \times 10^{-5}$ 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.

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

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 significantly 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.
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.

### Integrating RWS into TANDA

We study the contribution of RWS in fine-tuning the 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-

### 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.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>#Q</th>
<th>#A</th>
<th>#A^+</th>
<th>#A^-</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASNQ</td>
<td>57,242</td>
<td>20,745,240</td>
<td>60,285</td>
<td>20,684,955</td>
</tr>
<tr>
<td>AS2-NQ</td>
<td>84,121</td>
<td>27,208,065</td>
<td>86,756</td>
<td>27,121,309</td>
</tr>
<tr>
<td>RWS</td>
<td>84,089</td>
<td>2,103,027</td>
<td>69,945</td>
<td>2,033,082</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>TANDA</th>
<th>Transfer on</th>
<th>WikiQA</th>
<th>TREC-QA</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>MAP</td>
<td>MRR</td>
</tr>
<tr>
<td>RoBERTa-Base</td>
<td>ASNQ (2020)</td>
<td>0.889</td>
<td>0.901</td>
</tr>
<tr>
<td></td>
<td>AS2-NQ</td>
<td>0.898</td>
<td>0.910</td>
</tr>
<tr>
<td>% diff.</td>
<td>+1.01</td>
<td>+0.99</td>
<td>-0.66</td>
</tr>
<tr>
<td>RoBERTa-Large</td>
<td>ASNQ (2020)</td>
<td>0.920</td>
<td>0.933</td>
</tr>
<tr>
<td></td>
<td>AS2-NQ</td>
<td>0.923</td>
<td>0.935</td>
</tr>
<tr>
<td>% diff.</td>
<td>+0.33</td>
<td>+0.23</td>
<td>-0.73</td>
</tr>
</tbody>
</table>
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.

### 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-*


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

