Answer Quality Aware Aggregation for Extractive QA Crowdsourcing

Peide Zhu
Delft University of Technology
p.zhu-1@tudelft.nl

Zhen Wang
Delft University of Technology
z.wang-42@student.tudelft.nl

Claudia Hauff and **Jie Yang** and **Avishek Anand**
Delft University of Technology
{c.hauff,j.yang-3,avishek.anand}@tudelft.nl

Abstract

Quality control is essential for creating extractive question answering (EQA) datasets via crowdsourcing. Aggregation across answers, i.e. word spans within passages annotated, by different crowd workers is one major focus for ensuring its quality. However, crowd workers cannot reach a consensus on a considerable portion of questions. We introduce a simple yet effective answer aggregation method that takes into account the relations among the answer, question, and context passage. We evaluate answer quality from both the view of question answering model to determine how confident the QA model is about each answer and the view of the answer verification model to determine whether the answer is correct. Then we compute aggregation scores with each answer's quality and its contextual embedding produced by pre-trained language models. The experiments on a large real crowdsourced EQA dataset show that our framework outperforms baselines by around 16% on precision and effectively conduct answer aggregation for extractive QA task. The code is available at https://github.com/zpeide/Answer-Quality-Aware-Aggregation.

1 Introduction

Extractive Question answering (EQA) is a fundamental task in natural language processing (Parsing, 2009). With access to large-scale datasets, deep neural models have achieved significant advances in the EQA task (Lewis et al., 2019; Devlin et al., 2018; Zhang et al., 2020). Creating large-scale high-quality datasets is one of the essential factors driving progress (Rogers et al., 2021). Currently, a prevalent method for creating EQA datasets is crowdsourcing (Rajpurkar et al., 2016, 2018; Trischler et al., 2016; Yang et al., 2018; Talmor et al., 2018) thanks to its efficiency and scalability due to the availability of crowd workers. Yet, answers collected from crowd workers often contain a substantial amount of noise due to the reliability issue of crowd workers affected by their varying expertise, skills, and motivation (Kazai et al., 2011; Geva et al., 2019).

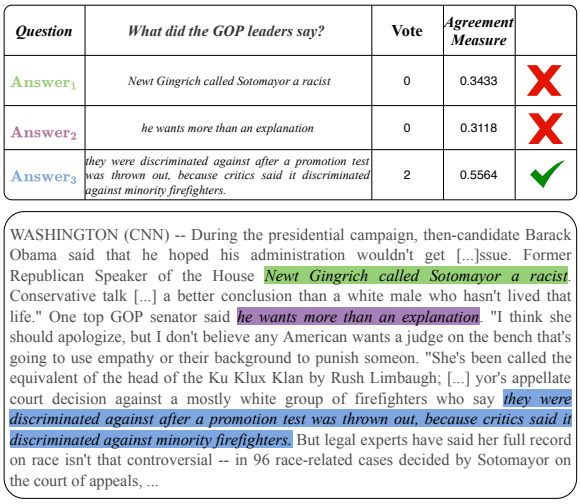


Figure 1: An example of answer aggregation for QA crowdsourcing. In this example, three crowd workers are asked to select a word span in the passage as the answer to the question. The gold answer can be aggregated from the disagreed answers by asking another group of workers for answer selection (vote) or using answer aggregation models (aggregation measure).

To reduce noise in crowdsourced data, a widely-adopted solution in previous crowdsourcing research is to assign each instance to multiple crowd workers to crate redundant annotations (Trischler et al., 2016; Yang et al., 2018; Talmor et al., 2018). Aggregation across answers provided by different crowd workers thus becomes one primary focus for crowdsourcing EQA datasets. Major voting is a simple and widely adopted aggregation method (Zheng et al., 2017) which elects answers that most crowd workers agree with. However, most of these major voting based methods are for categorical labels where the label space is small enough such that workers will more likely produce the same label (Passonneau and Carpenter, 2014;

Findings of the Association for Computational Linguistics: EMNLP 2022, pages 6176 - 6188
December 7-11, 2022

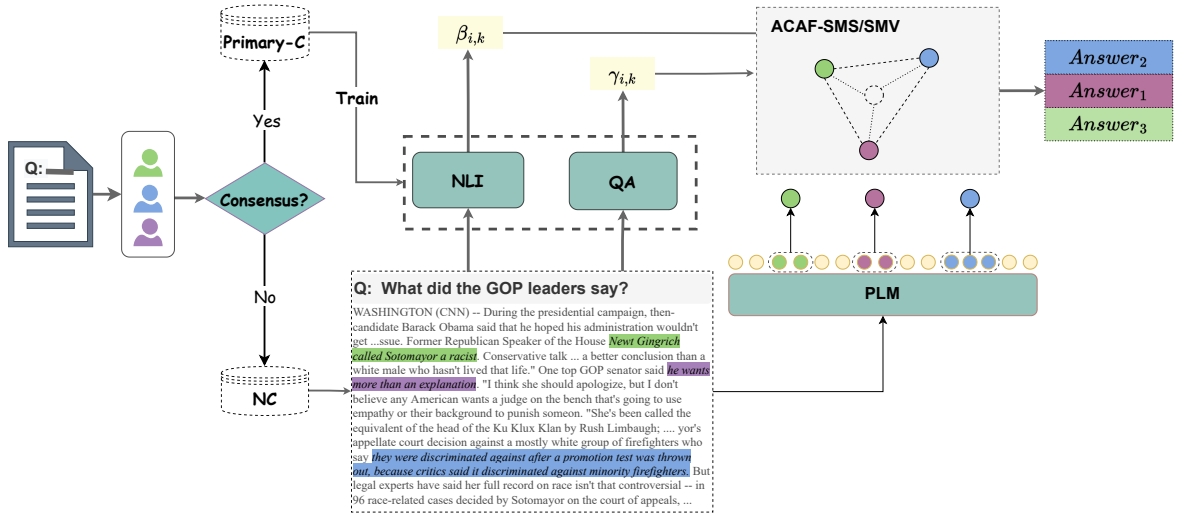


Figure 2: System overview and an example of automatic answer aggregation. Crowd workers are asked to label answer spans in passages for the given questions. If they achieve consensus, the QA pairs are used to fine-tune the natural language inference (NLI) based answer correctness evaluation model and the question answering (QA) model. Then we sort the non-consensus answers based on their encoding using a pre-trained language model (PLM), the answer correctness ($\beta_{i,k}$) and the question answering confidence ($\gamma_{i,k}$).

Lakkaraju et al., 2015; Nguyen et al., 2017; Zhang et al., 2021a). They cannot apply to this EQA task where the answer candidates are word spans rather than a limited number of categorical labels, due to the huge number of words in the dictionary. There are some methods for automatic aggregating text sequences (Li, 2020; Li and Fukumoto, 2019), but they only apply to free text sequence tasks such as translation. Unlike free text sequence tasks, answer candidates are word spans within context passages and their quality is related to both the question and the context passage. The previous methods do not consider these dependencies. Therefore, answer aggregation for EQA is commonly performed by having a second group of workers selects and verify answers (Trischler et al., 2016; Welbl et al., 2017). As the example in Figure 2 shows, crowd workers provide three distinct answer spans for the same instance. Another three crowd workers are then asked to vote for each answer annotation. $Answer_3$ got 2 votes and is selected as the ground-truth answer for the question. This method requires more resources and human efforts.

In this paper, we first model the candidate answer as a text sequence aggregation problem (Li and Fukumoto, 2019). Previous methods aggregate the best answer based on inter-answer distances of their vector representation. As answers for EQA are word spans within context passages, we adapt previous methods by presenting answers using contextual vector embedding produced by pre-trained language models (Wolf et al., 2020). In previous research, answer quality is evaluated by estimating worker reliability. However, we argue that in EQA, answer quality can also be evaluated based on its relation to the context passage and the question. We investigate answer quality evaluation from both the view of question answering (*Answer Confidence* measure) by using QA models and from the view of answer verification (*Answer Correctness* measure) by using natural language inference (NLI) models. We further propose a novel joint framework to incorporate the answer quality measures with the inter-answer distances based answer aggregation methods for EQA.

With this work we make following contributions:

- We propose a simple yet effective novel aggregation framework for aggregating crowdsourced answer annotations for EQA.
- We explore two answer quality measures *Answer Confidence* and *Answer Correctness* using weak heuristic question answering signal and NLI models and illustrate their effectiveness.
- The comprehensive experiments on a real large-scale crowdsourced QA dataset suggest the effectiveness of the proposed answer quality measures and the proposed answer aggregation methods. The results show that our framework can effectively leverage the rich information of context passage, questions and answer candidates for an-

swer aggregation and achieve an improvement of around 15% on precision to baseline methods.

2 Background

2.1 Crowdsourcing for QA Dataset Creation

Quality control in crowdsourcing has attracted intensive research (Snow et al., 2008; Kazai et al., 2011; Yang et al., 2019; Geva et al., 2019; Sayin et al., 2021). To reduce the noises of crowdsourced data, each data instance is commonly assigned to multiple workers to crate redundant annotations to infer the hidden ground truth by aggregation (Trischler et al., 2016; Yang et al., 2018; Talmor et al., 2018). In contrast to classification or categorical crowdsourcing tasks (Sun et al., 2014; Nguyen et al., 2017; Zhang et al., 2021a; Simpson et al., 2020; Lin et al., 2021) which have small label spaces, it is harder for crowd workers to achieve consensus on the answer for the same question.

What signals the disagreement contains and how to effectively use them is an interesting research question (Aroyo and Welty, 2015; Northcutt et al., 2021). Most existing work on this question focuses on classification problems. Some work (Min et al., 2019; Chen et al., 2022) found that it is possible to use noisy answers as week supervision signals to improve QA performance especially in low-resource domains. However, they still rely on the existence of ground-truth answers which is obtained by crowdsourcing. In practice, multi-stage methods are commonly adopted for answer aggregation in QA (Trischler et al., 2016; Welbl et al., 2017; Kwiatkowski et al., 2019). For example, a four-stage collection process is utilized for collecting NewsQA (Trischler et al., 2016). Each item is assigned to multiple crowd workers(avg. 2.73) to make answer annotations. Then another group (avg. group size is 2.48) is asked to validate distinct answer annotations collected in the previous stage). The Google Natural Questions dataset (Kwiatkowski et al., 2019) evaluates non-null answer correctness with consensus judgments from 4 "experts" and the k-way annotations (with k = 25) on a subset. This approach leads to more cost of human efforts, time and money.

2.2 Crowdsourced Text Sequence Aggregation

Majority Voting is the most common and simplest aggregation method. It assumes most workers have comparable accuracy and reliability on the task. Thus some workers will produce the same answer for the same question, especially for categorical label tasks where the label space is small enough. However, it can perform poorly on complex sequence labeling tasks such as translation, summarization, and question answering. The number of words in the dictionary is so huge that it is difficult for workers to produce the same answer so that the ground truth answer can be found. Therefore multi-stage crowdsourcing patterns are used to resolve disagreements by selecting, verifying, or correcting answers like the fore-mentioned methods in the last subsection. Several automation methods have been proposed to reduce human labor. (Li and Fukumoto, 2019; Li, 2020) converted the answer texts into embeddings and extracted the potential optimal answer by estimating the embeddings of the true answer, considering both worker reliability and sequence representation. (Braylan and Lease, 2020) proposed a single, general annotation and aggregation model by modeling label distances to support diverse tasks such as translation and sequence labels. (Braylan and Lease, 2021) proposed to perform answer aggregation on complex annotations such as sequence labeling and multi-object image annotation by matching and merging different labels. Although the proposed methods have achieved great advantage in complex answer aggregation, little research focuses on the question answering crowdsourcing.

3 Method

3.1 Problem Definition

For the extractive answer labeling task, each instance D_i assigned to crowd workers is a tuple containing a *context passage* P_i and a *question* Q_i, i.e. $D_i = (P_i, Q_i)$. The worker k is asked to select a word span $A_{i,k}$ from the context passage $A_{i,k} = (A^s_{i,k}, A^e_{i,k})$, s, e indicates the start and end position of the answer in the passage, or `NULL` if no answer is present in the passage. Then we get a set of answers for question Q_i: $\mathcal{A}_i = \{A_{i,k}\}_1^K$ from K workers. The answer aggregation model aims to select one answer from $\mathcal{A}_i$ as the golden answer or reject all answers. In this work, we focus on designing an effective automation answer aggregation model to reduce human labor for multi-stage answer selection and verification, especially when none of them agree with each other. We achieve this goal by making a ranked list of all answers, so the answers with the highest evaluation score are ranked in front.

3.2 Text Sequence Aggregation for Answer Aggregation

As word spans from context passages, we first model the answer aggregation problem as a free text sequence aggregation problem and adopt the free text sequence aggregation methods *Sequence Major Voting (SMV)* and *Sequence Maximum Similarity (SMS)* on it (Li and Fukumoto, 2019). These methods perform text sequence aggregation based on answers' vector representations.

Answer Representation Different from the text sequence aggregation problems like translation, the answer correctness depends not only on the answer word span, but also on its context. Therefore, to produce a single vector representation of each answer, instead of encoding the answer independently, we get the answer's contextual embedding by encoding the passage containing the answer with transformers-based pre-trained language models. Then we use the mean value of all answer token embeddings as the embedding of the answer. Formally, we define the passage which consists a sequence of words as $P_i = \{p_j\}_{j=1}^{|P_i|}$ (with $|P_i|$ being the length of the passage and p_j being the tokens in the passage), the language model as E and the token-wise encoding as:

$$\{\hat{p}_1, \hat{p}_2, \cdots, \hat{p}_{|P_i|}\} = E(\{p_1, p_2, \cdots, p_{|P_i|}\})$$

then the answer representation $\hat{a_{i,k}}$ is produced by: $\hat{a}_{i,k} = mean(\{\hat{p}_{A^s_{i,k}} : \hat{p}_{A^e_{i,k}}\})$

Sequence Majority Voting (SMV) by Li and Fukumoto (2019) is the direct adaptation of majority voting to the sequence label problem. *SMV* estimates the true answer embedding $\hat{e}_i$ as the mean vector of all answer vector representations:

$$\hat{e}_i = mean(\hat{a}_{i,1}, \hat{a}_{i,2}, \cdots, \hat{a}_{i,K}) \quad (1)$$

and ranks answer candidates according to their similarity to $\hat{e}_i$ and extracts the golden answer $\hat{z}_i$ as the answer candidate with the most semantic similarity to $\hat{e}_i$:

$$s_{i,k} = sim(\hat{a}_{i,k}, \hat{e}_i) \quad (2)$$

Sequence Maximum Similarity (SMS) (Li and Fukumoto, 2019). SMS method was first proposed for unsupervised ensemble of outputs of multiple text generation models (Kobayashi, 2018). It selects the gold output by selecting a majority-like output close to other outputs by using cosine similarity, which is an approximation of finding the maximum density point by kernel density estimation. Li and Fukumoto (2019) adopts SMS for crowdsourcing translation data which are generated by crowd workers instead of text generation models. However, they only use it on free text sequences. In this paper, we further adopt it to extractive QA task. We produce answer representation as fore-mentioned, and extract the golden answer $\hat{z}_i$ as the answer candidate with the largest sum of similarity $s_{i,k}$ with other answer annotations of the same question:

$$s_{i,k} = \frac{1}{|\mathcal{A}_i| - 1} \sum_{k_1 \neq k} sim(\hat{a}_{i,k_1}, \hat{a}_{i,k}) \quad (3)$$

3.3 Answer Quality Aware Answer Aggregation

The answer representations concentrate on answer contextual representation only, but the quality of each answer also depends on whether it can answer the question based on the context passage. The answer text sequence aggregation methods cannot fully utilize the rich information of both the context and question. Therefore, we further propose to aggregate crowdsourced answers in an answer quality aware way. We first propose to evaluate answer quality from the view of question answering model (*Answer Confidence*) and the view of answer verification model (*Answer Correctness*). Due to the lack of labeled data for training the QA and NLI models, prediction of these models are noisy and inaccurate. However, they can still provide hints on answer quality. Then we propose a novel aggregation method to strengthen the influences of possible high-quality answers (**ACAF-SMS/SMV**).

3.3.1 Answer Quality Evaluation

Answer Confidence (AF) We use BERT-QA (Devlin et al., 2018) as our QA model. It consists of two parts, the BERT encoder and the answer classifier. The answer classifier predicts the distributions of the start position and the end position separately based on the outputs of the BERT encoder. As argued by Xie et al. (2020); Zhu and Hauff (2021), the QA model should be quite confident about the prediction of answer start/end span to the answerable question. Thus the prediction probability distribution should peak on both $A^s_{i,k}$ and $A^e_{i,k}$. Therefore, the geometric average of these start position probability ($\Pr_s(s|P_i, Q_i)$) and end

position probability ($\Pr_e(e|P_i, Q_i)$) distributions can be used as a heuristic of the confidence of the answer prediction. Formally, We define the answer confidence $\gamma_{i,k}$ as follows:

$$\gamma_{i,k} = \max_{A^s_{i,k}-w \le b \le c \le A^e_{i,k}+w} \sqrt{\Pr_s(b|P_i, Q_i) \cdot \Pr_e(c|P_i, Q_i)}. \quad (4)$$

where w is search window size.

Answer Correctness (AC) QA models often lack the ability to verify the correctness of the predicted answer (Chen et al., 2021). One way to address this issue is to reformulate it to a textual entailment problem (Harabagiu and Hickl, 2006; Richardson et al., 2013; Chen et al., 2021) by viewing the answer context as the premise and the QA pair as the hypothesis. Then we use a natural language inference (NLI) system to verify whether the candidate answer proposed by crowd workers satisfies the entailment criterion. We use the transformers-based pre-trained sequence classification model for answer correctness verification. We treat the answer candidate as a short text sequence (answer-text), and formulate the input to the model in the format " `[CLS]` *question* `[SEP]` *passage* `[SEP]` *answer-text* `[SEP]` ". We truncate passages longer than the maximum 512 tokens and only keep the sentences containing the answer span. The embedding of the `[CLS]` token is used as the pooling encoding of the sequence, and a linear classification layer has performed the encoding. Finally, according to the passage, we use the $softmax$ function to get the final probability that an answer candidate is correct.

$$\beta_{i,k} = \mathbf{V}(P_i, Q_i, A_{i,k}) \quad (5)$$

Above, $\mathbf{V}$ represents the NLI model to verify the answer correctness. $\beta_{i,k}$ is the probability that the answer $A_{i,k}$ to question Q_i is correct.

We then propose to combine the answer confidence and the answer correctness probability for answer quality evaluation. Assuming these two measures are complementary, to make the method simple, we combine them as simple sum:

$$\upsilon_{i,k} = \gamma_{i,k} + \beta_{i,k}. \quad (6)$$

3.3.2 The Joint Method (ACAF-SMS/SMV)

We propose to join NLI model, QA model and contextual answer vector representations for answer aggregation by incorporating the answer correctness probability and answer confidence with sequence aggregation methods **SMV** and **SMS** to strengthen the influence of high-quality answers further. The joint sequence majority voting (**ACAF-SMV**) method computes the answer aggregation measure $s_{i,k}$ as:

$$s_{i,k} = \frac{\upsilon_{i,k}}{\sum_k \upsilon_{i,k}} sim(\hat{a}_{i,k}, \hat{e_i}) \quad (7)$$

and the joint sequence maximum similarity (**ACAF-SMS**) method as:

$$s_{i,k} = \upsilon_{i,k} \frac{\sum_{k_1 \neq k} \upsilon_{i,k_1} \cdot sim(\hat{a}_{i,k_1}, \hat{a}_{i,k})}{\sum_{k_1 \neq k} \upsilon_{i,k_1}} \quad (8)$$

The **AF-SMS** algorithm and **AF-SMV** algorithms are similar to the above mentioned methods by replacing answer correctness probability $\beta_{i,k}$ with answer confidence $\gamma_{i,k}$ or $r_{i,k}$. Figure 2 illustrates the proposed method.

4 Experimental Setup

4.1 Dataset

We evaluate the proposed method with the NewsQA dataset because it provides all crowdsourced raw answer annotations. The creation process of NewsQA demonstrates the challenges of QA dataset crowdsourcing and the importance and necessity of answer aggregation. Answers in the NewsQA are collected through a two-stage process: the primary stage (answer sourcing) and the validation stage. In the primary stage, each question solicits answers from avg. 2.73 crowdworkers. 56.8% of questions have consensus answers between at least two answers on the primary stage. 37.8% of questions got consensus answers after the validation stage. Crowdworkers do not come to a consensus for the rest 5.3% questions.

In this paper, we split NewsQA into four subsets: the ***primary consensus (Primary-C)*** set, which contains all passages, questions and their answers from the training set that achieve answer agreement on the primary stage; the ***primary non-consensus (Primary-NC)*** which contains all passages, questions and answer candidates that only achieve agreement after an additional round of answer validation from the training set; ***test consensus (Test-C)***

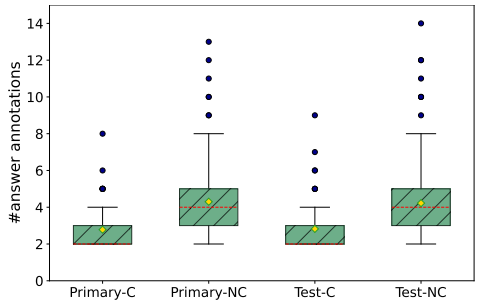


Figure 3: Number of answer annotations for each question in the four datasets we use.

Data	$\lvert P\rvert$	$\lvert Q\rvert$	$\lvert A_C\rvert$	$\lvert A_W\rvert$
Primary-C	11,469	61171	93,842	76,163
Primary-NC	11,469	40713	52,941	122,071
Test-C	634	3393	2,306	1,906
Test-NC	637	2273	2,980	6,620

Table 1: Statistics of the datasets; number of passages $|P|$; number of answerable questions $|Q_A|$; number of unanswerable questions $|Q_U|$; number of correct answers $|A_C|$ and number of wrong answers $|A_W|$.

set which contains passages, questions and answers that achieve consensus from the test set, and the ***test non-consensus (Test-NC)*** set which contains data items that only reach consensus after an additional round of answer validation from the test set. Figure 3 shows the boxplot of the number of crowdsourced answers for each question. There are more than four distinct answers per question in non-consensus sets. The Primary-C and Test-C sets are gold answers that can be used for training and evaluating the NLI and QA models used for answer aggregation. The Primary-NC and Test-NC sets are used for evaluating the proposed method. Passages in the training set do not contain passages in the test set, making our evaluation generative. Table 1 shows the statistics of our data.

4.2 Baselines

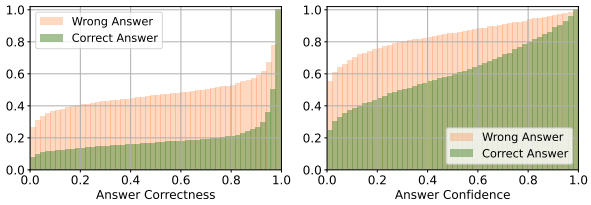
Random Selection (RS) The baseline is to rank answer annotations randomly for each question. We report the RS performance as the average over five random trials.

Context-Free (CF) SMS/SMV This baseline is to produce answer representation by treating answers as free text sequences without considering the context passages, i.e., the original SMS/SMV methods proposed by Li and Fukumoto (2019).

4.3 Evaluation

For each question, we sort the answers by the proposed aggregation methods. We evaluate the results in terms of widely used rank-aware metrics, including Precision@1 (P@1), Recall@1 (R@1), Mean Average Precision (MAP) and normalized discounted cumulative gain(NDCG). We choose the implementation of the information retrieval evaluation toolkit Pytrec_eval (Van Gysel and de Rijke, 2018) library.

5 Results and Analysis



(a) (b)

Figure 4: Cumulative answer correctness(a) and answer confidence(b) distributions on correct answers and incorrect answers.

5.1 Effectiveness of Answer Quality Evaluation Methods

Performance of AC on Answer Classification We train the NLI model for producing AC using the BERT for sequence classification implementation from the Huggingface Transformers library (Wolf et al., 2020) on the Primary-C set. It achieves 80.65% in accuracy and 87.59% in F1 on the Test-C set. On the Test-NC set, it performs 62.57% in accuracy and 64.52%, which is much worse than its performance on the Test-C set. The results indicate answers to questions that achieve consensus on the first sourcing stage are relatively more distinguishable and show the difficulty of specifying the correctness of disagreed answers. Figure 4a and Figure 5 show that AC is an effective metric to distinguish correct and wrong answers, which achieves 0.70 in AOC.

Performance of AF on Answer Classification We train the QA model using the BERT-QA implementation from the Huggingface Transformers library on the Primary-C set and adopt the exact match (EM) and F1 score (F1) to evaluate its performance. The QA model achieves 27.94% and 60.89% in EM and F1 respectively on the Test-C set. In contrast, its performance on the Test-NC

	Method	Primary-NC P@1	Primary-NC R@1	Primary-NC MAP	Primary-NC NDCG	Test-NC P@1	Test-NC R@1	Test-NC MAP	Test-NC NDCG
Baselines	RS	0.4728	0.3574	0.6550	0.7385	0.4782	0.3610	0.6600	0.7429
	CF-SMV	0.4660	0.3818	0.6536	0.7334	0.4765	0.3940	0.6629	0.7409
	CF-SMS	0.4683	0.3800	0.6545	0.7339	0.4831	0.3952	0.6645	0.7419
Answer Quality	AC	0.5703	0.4364	0.7280	0.7902	0.5820	0.4451	0.7369	0.7973
	AF	0.5796	0.4423	0.7310	0.7917	0.5878	0.4512	0.7376	0.7971
	AC+AF	0.6022	0.4595	0.7471	0.8034	0.6128	0.4677	0.7546	0.8096
Sequence Aggregation	SMV	0.5685	0.4124	0.7194	0.7822	0.5816	0.4234	0.7278	0.7894
	SMS	0.5701	0.4087	0.7190	0.7816	0.5851	0.4225	0.7282	0.7892
Joint Method	AC-SMV	0.6036	0.4467	0.7400	0.7985	0.6124	0.4528	0.7472	0.8047
	AF-SMV	0.6009	0.4544	0.7434	0.7997	0.6106	0.4634	0.7507	0.8057
	AC-SMS	0.6008	0.4450	0.7393	0.7978	0.6194	0.4598	0.7526	0.8084
	AF-SMS	0.6011	0.4538	0.7449	0.8007	0.6190	0.4687	0.7563	0.8099
	ACAF-SMV	**0.6213**	0.4646	**0.7533**	**0.8079**	0.6274	0.4698	0.7606	0.8140
	ACAF-SMS	0.6165	**0.4647**	0.7530	0.8076	**0.6304**	**0.4762**	**0.7635**	**0.8159**

Table 2: Experimental results of baselines and the proposed framework of answer agreement on *Primary-NC* and *Test-NC* set using the BERT-base-uncased model.

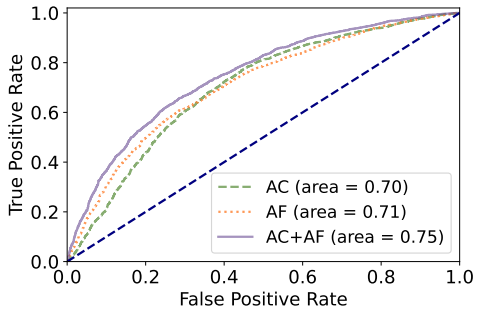


Figure 5: ROC Curve and area under the cure (AOC) of different answer classification methods, including answer correctness (AC), answer confidence (AF) and their combination.

set is 9.15% and 37.22% in EM and F1, which is much worse than performance on Test-C and demonstrates the difficulty of automatically answering these questions. Although its performance is poor due to the lack of enough training data, we observe that the AF score is an effective metric for correct answer classification as shown in Figure 4b and Figure 5 and achieves 0.71 in AOC, which is slightly better than AC. The combination of AC and AF (AC+AF) improves answer classification performance by up to 4% by a simple sum.

Performance of Answer Quality Evaluation on Answer Aggregation In Table 2, the rows **AC**, **AF** and **AC+AF** show the experimental results of performing answer aggregation by ranking answers according to AC, AF or by combining them(AC+AF). AC and AF have comparable performance; both achieve over 57% on P@1 and around 10% improvement over baselines, which shows the effectiveness of the proposed signals. By combining the NLI model prediction and the QA model heuristic signal, we can further improve the P@1 performance by around 3% on both Primary-NC and Test-NC sets, which shows the complementary strengths of the two signals.

5.2 Effectiveness of Answer Text Sequence Aggregation

As shown in Table 2, ***SMV*** and ***SMS*** can achieve similar performance to AC and AF by using the pre-trained BERT-base model as encoder without any fine-tuning. This suggests the effectiveness of modeling answer aggregation for extractive QA task as a sequence answer aggregation problem. These methods outperform the context-free sequence aggregation baselines by about 10%, which proves the importance of contextual embedding. Since both SMV and SMS are based on the latent semantic similarity among answer candidates, the effectiveness of these methods implies the crowdsourced answers bear some common knowledge or contextual information which can be further explored.

We then conduct experiments by combining AC, AF with SMS and SMV separately (**AC-SMV**, **AF-SMV**, **AC-SMS** and **AF-SMS**). Results in Table 2 show that the proposed joint methods achieve around 3% absolute performance improvement on P@1, around 5% on R@1 than using SMS and SMV only and similar to AC+AF (only slightly worse). By combining AC+AF with SMS or SMV (ACAF-SMS / ACAF-SMV), the system performance is further improved by around 2% on P@1 and around 1% on other metrics. These findings

Model	ACAF-SMV				ACAF-SMS			
	P@1	R@1	MAP	NDCG	P@1	R@1	MAP	NDCG
BERT-base	0.6274	0.4698	**0.7606**	**0.8140**	**0.6304**	0.4762	0.7635	0.8159
BERT-large	0.6238	0.4670	0.7575	0.8117	0.6247	0.4726	0.7607	0.8140
Roberta-base	0.6247	0.4653	0.7570	0.8111	0.6300	**0.4786**	**0.7638**	**0.8162**
BART-base	**0.6291**	**0.4724**	0.7545	0.8098	0.6304	0.4750	0.7633	0.8155

Table 3: Results of answer aggregation using different encoders.

first suggest the effectiveness of the joint aggregation method. They also demonstrate that the system can achieve better performance by combining unsupervised contextual answer representation and the weak learned signals.

5.3 Influence of Encoders

Table 3 show the performance of the joint methods ACAF-SMV and ACAF-SMS on Test-NC set using different types of pre-trained encoders `BERT-base`, `BERT-large`, `Roberta-base` and `BART-base`. The results first show the performance of both methods is robust alongside different encoders with different model sizes, types and pre-training methods, demonstrating the effectiveness and stability of the proposed methods. Second, ACAF-SMS outperforms ACAF-SMV with all kinds of encoders on the Test-NC set.

5.4 Case Study

As shown in Table 4, we conduct a case study to examine the performance of the proposed framework. In this case, AC, AC+AF and SMS suggest *waste* is the correct answer. However its answer confidence is very low(0.0025). AF points *great pacific garbage patch that stretches* is the best answer. Only **ACAF-SMS** ranks the golden answer *of the pacific ocean* as the best answer, even though the AC and AF scores of this answer are not the highest.

context	The American photographed the remains of albatross chicks that had died from consuming plastic waste found in the surrounding oceans. According to the artist, not a single piece of plastic in any of the photographs was moved, placed or altered in any way. The nesting babies had been fed the plastic by their parents, who collected what looked to them like food to bring back to their young. From cigarette lighters to bottle caps, the plastic is found in what is now known as the great Pacific garbage patch that stretches across thousands of miles of the Pacific Ocean.
Question	Plastic was found across thousands of miles of what
Answer Candidates	great pacific garbage patch that stretches 0.0081 0.7406 0.0053 0.4904
	of 0.0837 0.7406 0.0453 0.4737
	of the pacific ocean. 0.7745 0.0898 0.3658 0.5306
	waste 0.9175 0.0025 0.4142 0.4457
	in the 0.0129 0.0017 0.0085 0.0091

Table 4: An example from NewsQA dataset. There are 7 different answer annotations for the question. Some of the answers are overlapped. For each answer we report its ranking scores with AC AF SMS ACAF-SMS.

6 Conclusion

In this paper, we propose a novel answer annotation aggregation method for EQA crowdsourcing. We show that without any fine-tuning, our methods can achieve comparable performance with the trained QA and NLI model using *limited training data*. We introduce a novel algorithm for combining the NLI model, QA model and contextual text embedding for answer text sequence aggregation. The experiments on a real large-scale crowdsourced EQA dataset show the effectiveness and stability of the proposed method. The proposed methods outperform the baseline single metric method by around 16% absolute improvement on P@1 and 10% improvement on other ranking metrics. For future work, we will further explore methods incorporating crowd worker reliability and question answerability for better answer aggregation. We will also explore the applicability of our approaches to other tasks that deal with collecting extractive texts (DeYoung et al., 2020; Zhang et al., 2021b).

7 Limitations

While many automatic answer aggregation methods take crowd worker's reliability into consideration (Tian and Zhu, 2015; Li and Fukumoto, 2019),

to keep the proposed framework simple and concise, we focus on the influence of answer quality and ignore the worker reliability. Moreover, we only use NewsQA to evaluate the proposed method. Although it is possible to consider more real or simulated datasets, as shown by the experiments on SQuAD and Natural Questions in Appendix A.3, NewsQA is the only large extractive QA dataset that provides all actual annotations to the best of our knowledge. Besides, this paper assumes there is only one correct answer for each question, while it is possible that there are multiple correct answers in some applications. We will explore crowd worker reliability aware answer aggregation methods and extend our work to multi-answer settings in future research.

Acknowledgement

This research has been partially supported by the China Scholarships Council (CSC), NWO project SearchX (639.022.722) and NWO Aspasia (015.013.027).

References

Lora Aroyo and Chris Welty. 2015. Truth is a lie: Crowd truth and the seven myths of human annotation. *AI Magazine*, 36(1):15–24.

Alexander Braylan and Matthew Lease. 2020. Modeling and aggregation of complex annotations via annotation distances. In *Proceedings of The Web Conference 2020*, pages 1807–1818.

Alexander Braylan and Matthew Lease. 2021. Aggregating complex annotations via merging and matching. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pages 86–94.

Jifan Chen, Eunsol Choi, and Greg Durrett. 2021. Can nli models verify qa systems' predictions? *arXiv preprint arXiv:2104.08731*.

Nuo Chen, Linjun Shou, Ming Gong, and Jian Pei. 2022. From good to best: Two-stage training for cross-lingual machine reading comprehension. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 10501–10508.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Jay DeYoung, Sarthak Jain, Nazneen Fatema Rajani, Eric Lehman, Caiming Xiong, Richard Socher, and Byron C. Wallace. 2020. ERASER: A benchmark to evaluate rationalized NLP models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4443–4458, Online. Association for Computational Linguistics.

Adam Fisch, Alon Talmor, Robin Jia, Minjoon Seo, Eunsol Choi, and Danqi Chen. 2019. MRQA 2019 shared task: Evaluating generalization in reading comprehension. In *Proceedings of 2nd Machine Reading for Reading Comprehension (MRQA) Workshop at EMNLP*.

Mor Geva, Yoav Goldberg, and Jonathan Berant. 2019. Are we modeling the task or the annotator? an investigation of annotator bias in natural language understanding datasets. *arXiv preprint arXiv:1908.07898*.

Sanda Harabagiu and Andrew Hickl. 2006. Methods for using textual entailment in open-domain question answering. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 905–912.

Gabriella Kazai, Jaap Kamps, and Natasa Milic-Frayling. 2011. Worker types and personality traits in crowdsourcing relevance labels. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, pages 1941–1944.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Hayato Kobayashi. 2018. Frustratingly easy model ensemble for abstractive summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4165–4176.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. 2019. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466.

Himabindu Lakkaraju, Jure Leskovec, Jon Kleinberg, and Sendhil Mullainathan. 2015. A bayesian framework for modeling human evaluations. In *Proceedings of the 2015 SIAM International Conference on Data Mining*, pages 181–189. SIAM.

Patrick Lewis, Ludovic Denoyer, and Sebastian Riedel. 2019. Unsupervised question answering by cloze translation. *arXiv preprint arXiv:1906.04980*.

Jiyi Li. 2020. Crowdsourced text sequence aggregation based on hybrid reliability and representation. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1761–1764.

Jiyi Li and Fumiyo Fukumoto. 2019. A dataset of crowdsourced word sequences: Collections and answer aggregation for ground truth creation. In *Proceedings of the First Workshop on Aggregating and*

Analysing Crowdsourced Annotations for NLP, pages 24–28.

Jianzhe Lin, Tianze Yu, and Z Jane Wang. 2021. Rethinking crowdsourcing annotation: Partial annotation with salient labels for multi-label image classification. *arXiv preprint arXiv:2109.02688*.

Sewon Min, Danqi Chen, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2019. A discrete hard em approach for weakly supervised question answering. *arXiv preprint arXiv:1909.04849*.

An T Nguyen, Byron C Wallace, Junyi Jessy Li, Ani Nenkova, and Matthew Lease. 2017. Aggregating and predicting sequence labels from crowd annotations. In *Proceedings of the conference. Association for Computational Linguistics. Meeting*, volume 2017, page 299. NIH Public Access.

Curtis Northcutt, Lu Jiang, and Isaac Chuang. 2021. Confident learning: Estimating uncertainty in dataset labels. *Journal of Artificial Intelligence Research*, 70:1373–1411.

Constituency Parsing. 2009. Speech and language processing.

Rebecca J Passonneau and Bob Carpenter. 2014. The benefits of a model of annotation. *Transactions of the Association for Computational Linguistics*, 2:311–326.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for squad. *arXiv preprint arXiv:1806.03822*.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*.

Matthew Richardson, Christopher JC Burges, and Erin Renshaw. 2013. Mctest: A challenge dataset for the open-domain machine comprehension of text. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 193–203.

Anna Rogers, Matt Gardner, and Isabelle Augenstein. 2021. Qa dataset explosion: A taxonomy of nlp resources for question answering and reading comprehension. *ArXiv*, abs/2107.12708.

Burcu Sayin, Evgeny Krivosheev, Jie Yang, Andrea Passerini, and Fabio Casati. 2021. A review and experimental analysis of active learning over crowdsourced data. *Artificial Intelligence Review*, pages 1–23.

Edwin Simpson, Jonas Pfeiffer, and Iryna Gurevych. 2020. Low resource sequence tagging with weak labels. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8862–8869.

Rion Snow, Brendan O'connor, Dan Jurafsky, and Andrew Y Ng. 2008. Cheap and fast–but is it good? evaluating non-expert annotations for natural language tasks. In *Proceedings of the 2008 conference on empirical methods in natural language processing*, pages 254–263.

Chong Sun, Narasimhan Rampalli, Frank Yang, and AnHai Doan. 2014. Chimera: Large-scale classification using machine learning, rules, and crowdsourcing. *Proceedings of the VLDB Endowment*, 7(13):1529–1540.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2018. Commonsenseqa: A question answering challenge targeting commonsense knowledge. *arXiv preprint arXiv:1811.00937*.

Tian Tian and Jun Zhu. 2015. Max-margin majority voting for learning from crowds. *Advances in neural information processing systems*, 28.

Adam Trischler, Tong Wang, Xingdi Yuan, Justin Harris, Alessandro Sordoni, Philip Bachman, and Kaheer Suleman. 2016. Newsqa: A machine comprehension dataset. *arXiv preprint arXiv:1611.09830*.

Christophe Van Gysel and Maarten de Rijke. 2018. Pytrec_eval: An extremely fast python interface to trec_eval. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, pages 873–876.

Johannes Welbl, Nelson F Liu, and Matt Gardner. 2017. Crowdsourcing multiple choice science questions. *arXiv preprint arXiv:1707.06209*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi 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 M. Rush. 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, Online. Association for Computational Linguistics.

Yuxi Xie, Liangming Pan, Dongzhe Wang, Min-Yen Kan, and Yansong Feng. 2020. Exploring question-specific rewards for generating deep questions. *arXiv preprint arXiv:2011.01102*.

Jie Yang, Alisa Smirnova, Dingqi Yang, Gianluca Demartini, Yuan Lu, and Philippe Cudré-Mauroux. 2019. Scalpel-cd: leveraging crowdsourcing and deep probabilistic modeling for debugging noisy training data. In *The World Wide Web Conference*, pages 2158–2168.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W Cohen, Ruslan Salakhutdinov, and Christopher D Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. *arXiv preprint arXiv:1809.09600*.

Xin Zhang, Guangwei Xu, Yueheng Sun, Meishan Zhang, and Pengjun Xie. 2021a. Crowdsourcing learning as domain adaptation: A case study on named entity recognition. *arXiv preprint arXiv:2105.14980*.

Zhuosheng Zhang, Junjie Yang, and Hai Zhao. 2020. Retrospective reader for machine reading comprehension. *arXiv preprint arXiv:2001.09694*.

Zijian Zhang, Koustav Rudra, and Avishek Anand. 2021b. Explain and predict, and then predict again. In *WSDM '21, The Fourteenth ACM International Conference on Web Search and Data Mining, Virtual Event, Israel, March 8-12, 2021*, pages 418–426. ACM.

Yudian Zheng, Guoliang Li, Yuanbing Li, Caihua Shan, and Reynold Cheng. 2017. Truth inference in crowdsourcing: Is the problem solved? *Proceedings of the VLDB Endowment*, 10(5):541–552.

Peide Zhu and Claudia Hauff. 2021. Evaluating bert-based rewards for question generation with reinforcement learning. In *ICTIR '21: The 2021 ACM SIGIR International Conference on the Theory of Information Retrieval, Virtual Event, Canada, July 11, 2021*, pages 261–270. ACM.

A Appendix

A.1 Detailed Experimental Setup

Hyper-parameters for Training The NLI Model Adam optimizer (Kingma and Ba, 2014) with warming-up and linear schedule is used for fine-tuning the answer verification model. We set the maximum learning rate (lr) as $lr = 2e^{-5}$ and $\epsilon = 1e^{-8}$ and the warmup steps of 1000. The models are trained on a server using 4 GTX-1080 GPUs for 20,000 iterations where each iteration is a batch size of 32 and use the best performing checkpoint.

Method	Test-C		Test-NC	
	Accuracy	F1	Accuracy	F1
Bert-base	0.8065	0.8759	0.6257	0.6452

Table 5: NLI Performance on answer verification.

Hyper-parameters for Training The QA Model The QA model is trained on the same server consisting of 4 GeForce GTX 1080 gpus with a batch size of 32, the maximum learning rate of $1e^{-5}$ with adam as the optimizer for 10 epochs and take the epoch with the best validation accuracy as the final model.

Method	Test-C		Test-NC	
	Exact	F1	Exact	F1
Bert-base	27.94	60.89	9.15	37.22
Bert-large	31.21	62.21	12.23	37.33
Roberta-base	32.24	66.65	13.11	43.94

Table 6: QA model performance.

A.2 Evaluation with More Metrics

Besides the rank-aware metrics, we also compare method performance of the top-1 answer using two evaluation metrics: Exact Match, and the macro-averaged F1 score. Exact Match and F1 measures overlap between a bag-of-words representation of the ground truth and top-1 answers. We use the implementation of Eact Match and F1 from MRQA (Fisch et al., 2019)[1].

Method	Primary-NC		Test-NC	
	EM	F1	EM	F1
RS	0.4666	0.5246	0.4656	0.5292
CF-SMV	0.4640	0.5690	0.4729	0.5750
CF-SMS	0.4669	0.5696	0.4773	0.5749
AC	0.5638	0.6140	0.5689	0.6182
AF	0.5751	0.6300	0.5829	0.6337
AC+AF	0.5933	0.6426	0.5970	0.6454
SMV	0.5584	0.6179	0.5693	0.6287
SMS	0.5626	0.6202	0.5733	0.6309
AC-SMV	0.5980	0.6478	0.6027	0.6525
AF-SMV	0.5900	0.6449	0.5944	0.6459
AC-SMS	0.5957	0.6445	0.6089	0.6546
AF-SMS	0.5896	0.6423	0.6036	0.6492
ACAF-SMV	**0.6132**	**0.6626**	0.6146	**0.6652**
ACAF-SMS	0.6085	0.6568	**0.6168**	0.6622

Table 7: Performance of answer agreement on *Primary-NC* and *Test-NC* using the BERT-base-uncased model in terms of Exact Match (EM) and F1.

A.3 Answer Aggregation Results on Other Datasets

SQuAD and Natural Questions datasets only provide multiple annotations for dev sets. We performed experiments on by treating the training set as Primary-C and selecting questions with multiple different annotations and one consensus answer as Primary-NC. To train the NLI models needed for answer verification, besides the ground truth answers, we create negative answers by sampling different word spans with the same named entity types if possible, or word spans with the most similar part-of-speech(POS) structures.

[1] https://github.com/mrqa/MRQA-Shared-Task-2019

Method	P@1	R@1	MAP	NDCG
	SQuAD			
SMS	0.6251	0.4829	0.8064	0.8573
SMV	0.8150	0.4787	0.8074	0.8580
ACAF-SMS	0.8597	**0.5245**	0.9265	0.9460
ACAF-SMV	**0.8602**	0.5244	**0.9266**	**0.9460**
	Natural Questions			
SMS	0.4725	0.4183	0.7159	0.7894
SMV	0.4636	0.4094	0.7118	0.7864
ACAF-SMS	**0.7563**	**0.5233**	**0.8654**	**0.9008**
ACAF-SMV	0.7474	0.5141	0.8587	0.8959

Table 8: Results on SQuAD and Natural Questions.

A.4 Impact of Answer Selection on QA Performance

To explore the impact of answer selection on QA performance on the NewsQA dataset. We first train BERT-base-QA models on data where answers are selected by our method ACAF-SMS and ACAF-SMV, against answers selected by humans. We observe that f1-scores from our methods 59.68 (ACAF-SMS), and 60.56 (ACAF-SMV) are very close to the original data 61.12. We further investigate the effectiveness of our method by using them as additional voters for selecting the best answers in combination with human voting. Results show that the QA performance can be improved to 61.63 (ACAF-SMS as the voter), and 62.27 (ACAF-SMV as the voter), both surpassing the human-selection-only setting. These results show that the data quality improved by our methods can indeed improve QA models.

Method	Exact	F1
GroundTruth	28.00	61.12
ACAF-SMS	25.94	59.68
ACAF-SMV	26.37	60.56
ACAF-SMS$_{\text{voter}}$	27.44	61.63
ACAF-SMV$_{\text{voter}}$	**28.55**	**62.27**

Table 9: QA model performance.

A.5 Examples of Answer Aggregation Results

Context	Question	Answer	AC	AF	ACAF-SMS
Editor's note: Bryan Batt, who plays the closeted art director Salvatore Romano in the Emmy award-winning cable TV series "Mad Men," has acted in nine Broadway and nine Off-Broadway productions, such as "Sunset Boulevard," "Beauty and the Beast," "Jeffrey" and "Starlight Express." Batt, who is 45, has been acting for 23 years. He spoke to CNN.com about being an openly gay actor. "We have to work toward acceptance on all levels," says actor Bryan Batt, who is openly gay.	who is bryan batt	plays the closeted art director Salvatore Romano	0.0681	0.0023	0.0533
		plays	0.0681	0.0023	0.0023
		"Mad Men,"	0.9317	0.0018	0.5939
		closeted art director Salvatore Romano in	0.0033	0.0004	0.9745
		plays the closeted art director Salvatore Romano in the Emmy award-winning	0.0047	0.0023	0.0053
		cable TV series "Mad Men," has acted in nine Broadway and nine Off-Broadway productions, such as "Sunset Boulevard," "Beauty and the Beast," "Jeffrey" and "Starlight Express."	0.0031	0.0018	0.0036
		an openly gay actor.	0.0015	0.9934	0.5455
Malnutrition has left this baby born in Zimbabwe fighting for her life. She is the face of an unfolding crisis in a country once known as Africa's bread basket.[. . .] But the World Health Organization (WHO) says the desperate situation has triggered a widening cholera outbreak that has killed 775 people and infected more than 15,000.	What is the outbreak part of?	unfolding crisis	0.9737	0.0009	0.4904
		cholera	0.9848	0.1766	0.4737
		Africa's bread basket.	0.9894	0.0283	0.5306
		the desperate situation	0.9938	0.1811	0.4456

Table 10: A positive example (top) and a negative example (bottom) from NewsQA dataset.