Answer Consolidation: Formulation and Benchmarking

Wenxuan Zhou[1]*, **Qiang Ning**[2], **Heba Elfardy**[2], **Kevin Small**[2], **Muhao Chen**[1]
[1]University of Southern California, [2]Amazon
{zhouwenx,muhaoche}@usc.edu {qning,helfardy,smakevin}@amazon.com

Abstract

Current question answering (QA) systems primarily consider the single-answer scenario, where each question is assumed to be paired with one correct answer. However, in many real-world QA applications, multiple answer scenarios arise where consolidating answers into a comprehensive and non-redundant set of answers is a more efficient user interface. In this paper, we formulate the problem of *answer consolidation*, where answers are partitioned into multiple groups, each representing different aspects of the answer set. Then, given this partitioning, a comprehensive and non-redundant set of answers can be constructed by picking one answer from each group. To initiate research on answer consolidation, we construct a dataset consisting of 4,699 questions and 24,006 sentences and evaluate multiple models. Despite a promising performance achieved by the best-performing supervised models, we still believe this task has room for further improvements.[1]

1 Introduction

Open-domain question answering (QA) systems (Voorhees, 1999) aim to answer natural language questions using large collections of reference documents, contributing to real-world applications such as intelligent virtual assistants and search engines. Current QA systems (Zhu et al., 2021) usually adopt a three-stage pipeline consisting of: (1) a *passage retriever* (Yang et al., 2019; Karpukhin et al., 2020) that selects a small set of passages relevant to the question, (2) a *machine reader* that examines the retrieved passages and extracts (Wang et al., 2017, 2019) or abstracts (Lewis et al., 2020) the candidate answers, and (3) an *answer reranker* (Wang et al., 2018; Kratzwald et al., 2019) that fuses features from previous stages to either select one final answer or return the top-ranked answer from the previous stage.

Current QA research (Joshi et al., 2017; Kwiatkowski et al., 2019) primarily examines the case where each question is assumed to have a single correct answer. However, in practice, many questions can have multiple correct answers. For example, the question "*Is coffee good for your health?*" can be answered with respect to different aspects (e.g., "*coffee can help you with weight loss*", "*coffee can cause insomnia and restlessness*"). To correctly identify different aspects of answers to the same question while mitigating aspect-level redundancy, it is important to *consolidate* the answers. Answer consolidation is particularly desirable for applications such as intelligent assistants, where responses are desired to be both comprehensive and concise. Additionally, in scenarios where QA is used for knowledge extraction (Bhutani et al., 2019; Du and Cardie, 2020) or claim verification (Yin and Roth, 2018; Zhang et al., 2020), consolidation is also an essential step to identify salient knowledge or evidence while mitigating duplication.

To effectively recognize multiple aspects of answers in QA systems, our *first* contribution is to introduce and formalize the *answer consolidation* problem. Specifically, given a question paired with multiple answer snippets, answer consolidation first partitions the snippets into groups where each group represents a single aspect within the answer space. Once partitioned, the final answer set is produced by returning a representative snippet from each group. In this formulation, the answer consolidation task is a post-processing stage that takes predicted answer-mentioning snippets (in this work, sentences) from previous QA stages and produces an answer set that maximizes answer aspect coverage while minimizing answer duplication.

To foster research on the answer consolidation problem, our *second* contribution is the collec-

*This work was conducted when the first author was doing an internship at Amazon.

[1]The contributed resources and implementation are available at https://github.com/amazon-research/question-answer-consolidation.

Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 4314 - 4325

tion of a new dataset, namely, QUASI (Question-Answer consolidation). QUASI consists of 4,699 questions such that each question is paired with multiple answer-mentioning sentences grouped according to different aspects. Starting with a Quora-based question source (Chen et al., 2018), noting the potential for multi-aspect answers, we first retrieve 10 relevant answer sentences from the web for each question. These sentences are then examined by three crowd-sourced workers, who exclude sentences that do not contain an answer and group the remaining ones. Finally, individual sentence groupings from different workers are aggregated into a single partitioning. QUASI consists of 24,006 sentences and 19,676 groups, corresponding to an average number of 4.18 aspects per question and 1.22 sentences per group.

Our *third* contribution is a comprehensive benchmarking for the answer consolidation problem based on QUASI. Specifically, we consider two evaluation settings: (1) *classification*, where the model predicts whether two sentences are in the same group. (2) *sentence grouping*, where the model groups the answer sentences. We evaluate a wide selection of zero-shot and supervised methods, including various SoTA sentence embedding models (Reimers and Gurevych, 2019; Gao et al., 2021), cross-encoders (Devlin et al., 2019; Liu et al., 2019), and a newly proposed answer-aware cross-encoder model. In the supervised setting, the answer-aware cross-encoder achieves the best results based on the Matthew correlation coefficient (MCC) score of 87.8% (classification setting) and an adjusted mutual information (AMI) score of 68.9% (sentence grouping setting). As this performance is notably far from perfect, our findings indicate the need for future investigation on this meaningful, but challenging, task.

2 Related Work

QA with multiple answers. Many QA datasets have assumed that a question has a single correct answer (Joshi et al., 2017; Kwiatkowski et al., 2019), while in real scenarios, many questions can have multiple correct answers. Fewer datasets for QA or machine reading comprehension (MRC) have been proposed with the consideration of multiple answers. In extractive MRC, MASH-QA (Zhu et al., 2020) allows one question to be answered by multiple non-consecutive text spans. In abstractive MRC/QA, MS MARCO (Campos et al., 2016) simply treats different workers' answers as different answers. DuReader (He et al., 2018) merges similar answers during data construction. QReCC (Anantha et al., 2021) allows one worker to provide multiple different answers. Beyond these, WebQuestions (Berant et al., 2013) and GooAQ (Khashabi et al., 2021) include lists of diverse answers, and TREC-QA (Baudiš and Šedivỳ, 2015) uses regular expressions to capture multiple answers. However, none of the aforementioned efforts have investigated effective consolidation of multiple answers. In this work, we formally define and collect a dataset for the answer consolidation problem as a complement to previous work. From another perspective, AmbigQA (Min et al., 2020) focuses on the case where a question can be interpreted in different ways, leading to the question disambiguation task. This is fundamentally different from our work that partitions answers to the same question into different coherent subsets. Stance detection (Liu et al., 2021) is concerned with the focused problem of collecting approving/disapproving opinions for a yes-no question, unlike our studied problem where all multi-answer questions are considered and the answers are not limited to binary opinions.

Answer Summarization. Questions with multiple answers are common in online communities. For example, Liu et al. (2008) observe that no more than 48% of best answers on Yahoo! Answers are unique. Many efforts (Song et al., 2017; Chowdhury and Chakraborty, 2019; Fabbri et al., 2021) have been devoted to summarizing reusable answers in community QA. Particularly, AnswerSumm (Fabbri et al., 2021) proposes a dataset where different answers are rewritten to bullet points by humans. While training on summarization data may enable the model to return salient and non-redundant answers, such training only works for abstractive machine readers. A more related work is BERT-DDP (Fujita et al., 2020), which considers the problem of getting a diverse and non-redundant answer set. They construct a dataset based on Yahoo! Chiebukuro where workers are asked to provide an answer set given a question. However, the correct answer set is not unique when answers are equivalent. As they treat all but the annotated answer set as wrong, both training and inference are prone to false negatives. In this paper, we group the answers with respect to their aspects and provide a discriminant rule, such that the correct group assignment is unique.

Diverse passage retrieval. Many information retrieval efforts address the problem of retrieving diverse documents for a query (Clarke et al., 2008; Fan et al., 2018; Abdool et al., 2020). In QA, Min et al. (2021) examine answer diversity in passage retrieval and propose a self-supervised dynamic oracle training objective. However, as passages may contain irrelevant information to the question, the retriever faces the challenge of identifying and integrating answers in passages when assessing answer diversity. In this work, we consider a dedicated task of answer consolidation and leave the problem of identifying answers to previous QA stages.

3 Answer Consolidation Task

Motivation. Many questions can have multiple correct answers, including questions explicitly asking for a multi-answer list (e.g., *What are the symptoms of flu?*) or questions where different people have different opinions (e.g., debate questions), amongst others. To provide users with a comprehensive view of the answers, the QA system needs to actively identify different answers as opposed to only returning the most popular or top-ranked answer. Additionally, as the same answer may be repeated or paraphrased many times in the reference corpus (e.g., web), the QA system may also need to eliminate the redundant answers. We address these requirements within answer consolidation.

Basic concepts. When answering a specific question, different answers may be given regarding different perspectives, opinions, angles, or parts of the overall answer. We regard such answers as those pertaining to different *aspects*. Furthermore, we refer to two sentences as *equivalent* if they contain the same answer aspect(s) and *distinct* if they express different answer aspect(s). To better identify equivalent/distinct sentences, we propose the following operational discriminant rule: Given two answer-mentioning sentences s_1, s_2 for the same question q, we can rewrite the answers contained in s_1 and s_2 into yes-no questions q_1' and q_2', which can be answered by yes/no/irrelevant.[2] Then, if s_1 and s_2 give coherent answers of yes/no[3] to q_2' and q_1', respectively, then s_1 and s_2 are considered to represent equivalent aspects. Otherwise, they are considered distinct from each other.

[2]A general process for changing a sentence to questions can be found at https://www.wikihow.com/Change-a-Statement-to-Question.

[3]Answers of irrelevant are not considered coherent.

We take the following example:

Q: Is coffee good for your health?

S1: Coffee can make you slim down.

S2: Coffee can relieve headache.

S3: Coffee can help with weight loss.

Then we rewrite the answers contained in sentences as the following questions:

Q'1: Can coffee make you slim down?

Q'2: Can coffee relieve headache?

Q'3: Can coffee help you with weight loss?

We can tell that S1 and S3 are equivalent, as they both give coherent answers (yes) to each others' yes-no questions. We can also tell that S2 is distinct from S1 and S3, as it gives irrelevant answers to Q'1 and Q'3.

Task definition. A formal definition for answer consolidation is that given a question and a set of answer-mentioning sentences, answer consolidation aims at putting sentences into groups such that: (1) each sentence belongs to exactly one group, and (2) sentences from the same/different groups are equivalent/distinct. In this way, each sentence group corresponds to the same answer aspect(s). We show in §4.3 that although this definition may fail for a pair of sentences if they are partially relevant, it only occurs for 2.6% of sentences, which shows that our operational task definition works well for the majority of sentences in practice.

In this work, we treat answer consolidation as a stand-alone process applied after QA retrieval such that we only take the question and answer-mentioning sentences as input. In this way, the answer consolidation model is independent of the retriever and the reader architectures, and can flexibly adapt to different QA systems. We show in §6.3 that taking sentences instead of answer spans as input leads to better performance.

4 Question-Answer Consolidation Dataset (QUASI)

In this section, we describe the creation of our Question-Answer consolidation dataset (QUASI) including corpus collection (§4.1) and dataset annotation (§4.2). We then provide statistical and quantitative analysis of QUASI (§4.3).

4.1 Corpus

We created QUASI based on the Quora question pairs (QQP) corpus (Chen et al., 2018), which consists of 364k questions pairs, originally designed for predicting whether pairs of questions have the

same meaning. We st
questions have a high
ing multiple correct an
removed questions cc
non-English words usi
questions containing p
{"I", "you", "we", "my
that such questions fre
cific personal experie
may not necessarily co
opinions, or facts in th

Next, we retrieved s
contain multiple answ
a question, we retriev
the web using a SoTA i
each sentence was asso
and a URL. To ensure
answers, we first remo
`quora.com` and kep
vance scores of the to
were larger than speci
old. Finally, we kept t
remaining question an
sentence group annota

4.2 Annotation

We used Amazon Mechanical Turk (AMT) to annotate QUASI. Each AMT HIT consisted of a question and 11 sentences (including the top-10 relevance scores and one additional *attention-check* sentence) where the crowd workers were required to: (1) identify sentences that actually contain answers to the question and (2) put answer-mentioning sentences into sentence groups with respect to their aspects. The workers were allowed to skip an answer-mentioning sentence if it was hard to put it into any groups (e.g., sentences containing more than one aspect).

The AMT interface is shown in Figure 1. Annotation was performed by dragging the sentences between blue boxes. The sentence groups could be added or removed using the two buttons. To submit the HIT, workers needed to put all sentences into boxes corresponding to either a specific sentence group, *not an answer*, or *hard to put into groups*.

Cost. Each HIT was assigned to three annotators with pay of $0.50/HIT, leading to target an hourly pay rate of $15. We randomly sampled 5k questions from §4.1 for HIT submission to AMT.

[4] https://abiword.github.io/enchant/

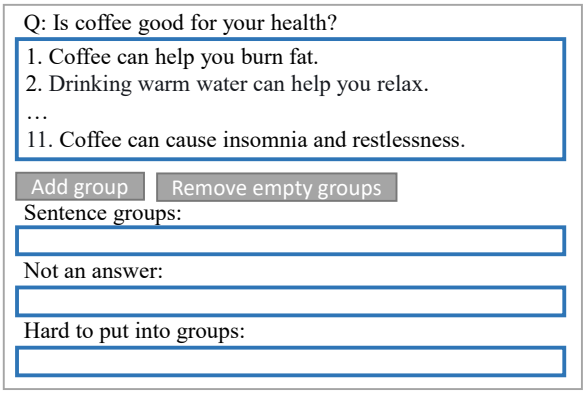

Figure 1: The interface used to collect the dataset. The second sentence is an attention-check sentence.

Quality Control. We used three strategies to ensure the annotation quality:

1. **Workers selection.** We only allowed crowd workers with acceptance rate $\geq 98\%$ and had completed at least 5k hits to work on the task. We provided annotation guidelines and examples of 3 annotated hits to instruct the workers.
2. **Qualification test.** We manually annotated three hits as the qualification test. Workers were required to practice on these three hits and get the correct sentence groups on at least 2 hits to continue working on the task. We pay $0.05 for each submitted hit in the qualification test. Although we had provided detailed instructions, only 24% of workers passed the qualification test.
3. **Attention checker.** For all hits, we added an attention-check sentence that was randomly sampled from other questions, so that it was unlikely to answer the question. As a part of the task, the worker needed to identify that this attention-check sentence did not involve an answer, otherwise she/he would be blocked from continuing to work on the task.

Label aggregation. To ensure data quality, we aggregated worker annotations. To derive the sentence set for answer consolidation, we begin by only considering sentences put into any sentence group(s) by all crowd workers as eligible, keeping 37,588 out of 50k sentences. Next, we derived the aggregated sentence groups from AMT annotations. As we are not aware of existing methods for this process, we proposed the following algorithm for constructing new sentence groups. First, we sort the sentences by their relevance scores and create a sentence group with the most relevant sentence

Type	Description	Example	Percentage
Formatting	Sentences only differ in letter case, punctuation, or short forms.	Q: How small can a black hole be? S1: scientists think the smallest black hole are as small as just one atom. S2: Scientists think the smallest black holes are as small as just one atom.	9%
Exact match	Sentences are different but answer spans are the same.	Q: What are some good stories of revenge in relationship? S1: Shakespeare's "Hamlet" is one of the most famous plays of revenge. S2: The play, Hamlet by William Shakespeare explores the concept of revenge.	47%
Lexical variation	Answer spans differ in articles, verb tenses, or have synonym substitutions.	Q: Is it hard to get a job as a fashion buyer? S1: Fashion jobs in merchandising can be very challenging. S2: Picking one out of many fashion jobs generally is an overwhelming challenge.	11%
Semantic variation	Answers are paraphrased, or identification requires commonsense reasoning.	Q: How does the respiratory system work? S1: The respiratory system works by getting the good air in and the bad air out. S2: The Respiratory System a simple system designed to get oxygen into the body, and to get rid of carbon dioxide and water.	30%
Ambiguous	We do not agree with the crowd workers' annotation.	Q: Is Zeus more powerful then Odin? (Not the same aspects.) S1: Zeus is 10 ton more than Odin. S2: In DC, Zeus is higher than Odin.	3%

Table 1: Types of equivalent sentences annotated by crowd workers. We randomly sampled 100 sentence pairs in the same group, manually annotated the answer span (underlined), and categorized them into different types.

being the only member. We then iterate over the remaining sentences with the following procedure. For a sentence s, there are three possible cases:

1. If there existed one group $\mathcal{G}$ such that $\forall s' \in \mathcal{G}$, s and s' were put into the same group by all workers, and for all already added sentences $s^* \notin \mathcal{G}$, s and s^* were put into different groups by all workers, we added s to $\mathcal{G}$.
2. If for all already added s^*, s and s^* were put into different groups by all workers, we created a new group with s being the only member.
3. Otherwise, we discarded sentence s, since there was disagreement on this sentence.

Finally, we keep each question for which the number of preserved sentences was larger than one. Our aggregation algorithm produced sentence groups on a subset of sentences, on which all workers agree on each pair of sentences about whether they belong to the same group or not. After this process, 4,699 out of 5,000 questions and 24,006 out of 37,588 sentences were kept.

4.3 Dataset Analysis

We provide statistical and qualitative analyses regarding QUASI in this section.

Annotation quality. We first analyze the quality of data annotation before label aggregation. In the first annotation task of identifying whether a sentence contains an answer, AMT workers achieved an inter-annotator Fleiss' kappa of 0.62, an average agreement rate of 90.2%, and a worker agreement with aggregate (WAWA) of 82.5% in F_1. WAWA is used to compare the majority vote with all workers' annotations. In the second annotation task of sentence grouping, we first get the set of sentences that all workers put into some groups. We then calculate the workers' agreement on each pair of sentences regarding whether they belong to the same group or not. The inter-annotator Fleiss' kappa, average agreement rate, WAWA F_1 are 0.46, 84.8%, and 75.9%, respectively. These results show that the overall annotation quality is usable, but with room for improvement. Accordingly, to further improve the data quality, we only keep group annotations on which all workers agree (as stated in §4.2).

Dataset statistics. Our final dataset consists of 4,699 questions, 24,006 sentences, and 19,676 groups. On average, there are 4.18 groups per question, and 1.22 sentences per group. Specifically, 97.7% of questions have multiple aspects (sentence groups), and 45.4% of questions have at least one pair of equivalent sentences. In terms of sentence groups, 86.6% of groups have only one sentence, 8.8% of groups have two sentences, and the remaining 4.6% of groups have 3 or more sentences. Overall, this analysis shows that our dataset contains both multi-aspect and redundant answers that align with the challenges of answer consolidation.

Types of equivalent sentences. To get a better understanding of the required knowledge to identify equivalent sentences, we randomly sampled 100 sentence pairs in the same group and manually labeled the pairs with the types shown in Table 1. We observed that if the machine reader has the correct answer spans, 56% (formatting and ex-

act match) of the equivalent sentences could be directly identified by string comparison. Another 11% of the equivalent sentences only differed at the lexical level, which may be identified using lemmatization, removal of stop words, or a dictionary of synonym words. 30% of equivalent sentences are semantic variations such that identifying equivalence requires understanding of their meanings and potentially even commonsense reasoning. E.g., for the example given in Table 1, the answer consolidation model needs to understand that *oxygen* is *good air* and *carbon dioxide* corresponds to *bad air*. For the remaining 3% of pairs, we do not agree with the annotation. Either the sentences do not answer the question, or they do not contain the same aspect(s).

Limits of the task definition. During data annotation, 2.6% of answer-mentioning sentences are denoted as "*hard to put into groups*". After inspection, we find that these sentences contain more than one aspect of answers. For example, given the question *What are the best places to visit and things to do in San Diego, CA?*, one sentence may be *The San Diego Zoo, Balboa Park, and SeaWorld are the top tourist attractions in San Diego.*, which contains 3 different answers. This sentence overlaps with multiple groups and thus cannot be placed in a single group. Given the low prevalence, we leave consideration of such sentences to future work.

5 Approach

In this section, we first tackle the classification setting of answer consolidation (§5.1). Given a question and answer-mentioning sentences, the task is to predict for a pair of sentences whether they are in the same group. We consider different types of models, including sentence embedding models, cross-encoders, and answer-aware cross-encoders.

Then we consider the sentence grouping setting, presenting the method of transforming pairwise predictions to sentence groups (§5.2). For all methods, we use RoBERTa$_{\text{LARGE}}$ (Liu et al., 2019) as the encoder, noting that other pretrained language models (PLMs) can easily be incorporated as part of these methods.

5.1 Sentence Pair Classification

Sentence embedding models. Sentence embedding models (Reimers and Gurevych, 2019; Gao et al., 2021) produce for each sentence an embedding vector, with which we can use metrics such as cosine to calculate their similarity. Specifically, given a question q and a sentence s, we first tokenize them to X_q and X_s using the RoBERTa tokenizer, and then concatenate them as inputs:

<s>X_q X_s</s>

Following Gao et al. (2021), we take the <s> embedding in the last layer of PLM as the sentence embedding. Then for a pair of sentences, whether they are in the same group is decided by the cosine similarity of the sentence embedding. The similarity can be converted to binary predictions using the best threshold that is selected on the validation set.

The sentence embedding models can work in both zero-shot and supervised settings. In the zero-shot setting, we directly use the pretrained sentence embedding model to make predictions without fine-tuning. In the supervised setting, given a pair of sentence embedding $\boldsymbol{h}_1$, $\boldsymbol{h}_2$, and label $y \in \{0, 1\}$, where 0 and 1 mean not in/in the same group respectively,[5] we fine-tune the PLM based on the following regression objective of sentence-transformers (Reimers and Gurevych, 2019):

$$\mathcal{L}_{\text{reg}} = \left(\cos\left(\boldsymbol{h}_1, \boldsymbol{h}_2\right) - y\right)^2 .$$

Cross-encoders. Cross-encoders (Devlin et al., 2019; Liu et al., 2019) take a pair of sentences as the input and predict whether they are in the same group or not. Given a question q and two answer-mentioning sentences s_1 and s_2, we first tokenize them as X_q, X_{s_1}, and X_{s_2} using the RoBERTa tokenizer, and then take X_q X_{s_1} and X_q X_{s_2} as two segments of inputs, following the input formats of sentence pair classification tasks (Liu et al., 2019):

<s>X_q X_{s_1}</s></s>X_q X_{s_2}</s>

Prediction is independently performed on sentence pairs using a binary classifier on the first special (classification) token <s> embedding in the last layer of the PLM. The cross-encoders work in both zero-shot and supervised settings. In the zero-shot setting, we fine-tune the model on the MNLI (Williams et al., 2018) dataset and take *entailment* as *in the same group*. In the supervised setting, given the sentence pair embedding (obtained from <s>) $\boldsymbol{h}$ and the label y, we fine-tune

[5] We find that using 0 for *not in the same group* achieves better results than using -1.

the model using the binary cross-entropy loss:

$$p = \sigma\left(\boldsymbol{w}^\intercal \boldsymbol{h}\right),$$
$$\mathcal{L}_{\text{bce}} = -\left(y \log p + (1-y)\log\left(1-p\right)\right),$$

where p is the probability that the sentences are in the same group, σ is the sigmoid function, $\boldsymbol{w}$ is a parameter of the classifier. In inference, we convert p to binary predictions using the best threshold selected on the validation set. The cross-encoders require predicting on all sentence pairs and have higher computational costs than sentence embedding models. However, we observe in experiments that cross-encoders consistently outperform sentence embedding models in the supervised setting.

Answer-aware (A^2) cross-encoders. §4.3 shows that 56% of equivalent sentences can be directly identified if the model knows the answer spans. Therefore, we provide the answer consolidation models with answers generated from the $\text{UnifiedQA}_{\text{LARGE}}$ model (Khashabi et al., 2020). As the UnifiedQA is trained on both extractive and abstractive datasets, the answers may not be text spans of sentences. Specifically, given a question q, two sentences s_1 and s_2, and the generated answers a_1 and a_2, we first tokenize them as X_p, X_{s_1}, X_{s_2}, X_{a_1}, and X_{a_2}, respectively, then construct the input to cross-encoders as:

<s>X_q X_{s_1} X_{a_1}</s></s>X_q X_{s_2} X_{a_2}</s>

The training process and inference process are the same as the cross-encoders.

5.2 Sentence Grouping

In answer consolidation, our ultimate goal is to obtain the consolidated sentence groups. This is done in a two-step approach. The first step is to get the matrix of distances $\mathcal{D}$ between pairs of sentences for a question. We perform this step using models trained in the classification setting. For sentence embeddings, $\mathcal{D}$ is adopted as the pairwise cosine distance matrix. For cross-encoders, each entry of $\mathcal{D}$ equals to 1 minus the predicted probability for a sentence pair. As $\mathcal{D}$ derived in this way may not be always symmetric, we use $\frac{1}{2}\left(\mathcal{D} + \mathcal{D}^\intercal\right)$ as the distance matrix instead.

The next step is to transform the distance matrix into sentence groups. Here we apply agglomerate clustering (Han et al., 2011). It uses a bottom-up strategy, starting from letting each sentence form its own cluster, and then recursively merging the clusters if their distance is smaller than a threshold. We use the average distance of sentence pairs as the inter-cluster distance measure and select the best threshold on the validation set. Agglomerate clustering stops when the distances between all clusters are larger than the threshold.

6 Experiments

In this section, we present the experimental setup (§6.1), show the main results (§6.2), conduct an ablation study (§6.3), and provide error analysis (§6.4).

6.1 Experimental Setup

Dataset. We randomly split the 4,699 questions into an 80/10/10 split, which serves as the training, validation, and test set, respectively.

Evaluation metrics. We use different evaluation metrics in the two evaluation settings. For the classification setting, we first use the micro F_1 measure. Considering that classes in the dataset are highly imbalanced (only 11% of sentence pairs are in the same group), we additionally use the Matthews correlation coefficient (MCC; Matthews 1975), which is considered a more class-balanced metric. For the sentence grouping setting, we use clustering metrics including adjusted rand index (ARI; Rand 1971) and adjusted mutual information (AMI; Nguyen et al. 2009). These two metrics take the predicted grouping and the ground-truth grouping, and measure the similarity between them. For all metrics, larger values indicate better performance, and a value of 100% indicates perfect classification/grouping.

Configuration. We implement the models using Huggingface's Transformers (Wolf et al., 2020). The models are optimized with Adam (Kingma and Ba, 2015) using a learning rate of 1e−5, with a linear decay to 0. We fine-tune all models for 10 epochs with a batch size of 32 questions (including all associated sentence pairs). The best model checkpoint and thresholds are selected based on the validation set. We report the average results on 5 runs of training using different random seeds.

Models. We use $\text{RoBERTa}_{\text{LARGE}}$ (Liu et al., 2019) as the encoder for all models. For sentence embedding models, we try RoBERTa fine-tuned on two intermediate tasks: 1) SRoBERTa (Reimers and Gurevych, 2019) is fine-tuned on natural language inference (NLI) datasets, achieving better

Model	Classification		Grouping	
	F_1	MCC	ARI	AMI
Zero-shot Sentence Embedding				
RoBERTa	22.2	9.2	55.8	33.3
SRoBERTa	41.9	35.1	57.8	37.7
SimCSE-RoBERTa	**53.2**	**47.6**	66.1	46.1
Zero-shot Cross-Encoders				
RoBERTa-MNLI	52.4	47.0	**69.0**	**48.5**
A^{2}RoBERTa-MNLI	40.4	34.6	60.9	40.0
Supervised Sentence Embedding				
RoBERTa	73.5	70.3	79.2	58.5
SRoBERTa	80.7	78.8	85.3	64.3
SimCSE-RoBERTa	81.1	79.0	85.7	64.9
Supervised Cross-Encoders				
RoBERTa	86.8	85.9	88.4	66.9
A^{2}RoBERTa	88.2	86.8	89.6	68.2
RoBERTa-MNLI	88.7	87.4	89.6	68.2
A^{2}RoBERTa-MNLI	**89.0**	**87.8**	**90.4**	**68.9**

Table 2: Main results on the test set of QUASI. The best results in the zero-shot and supervised settings are highlighted in **bold**.

results on semantic textual similarity (STS) tasks, and 2) SimCSE-RoBERTa (Gao et al., 2021) is fine-tuned in a self-supervised fashion, taking a sentence and predicting itself using a contrastive learning objective. For cross-encoders, in addition to directly running supervised fine-tuning on our data, we also try supplementary training on an intermediate labeled-data task (Phang et al., 2018), which fine-tunes cross-encoders on MNLI before supervised fine-tuning. Particularly in the latter setting, we observe it being necessary to re-initialize the classifier before supervised fine-tuning to obtain more promising performance.

6.2 Main Results

The experimental results on both the pairwise classification and sentence grouping settings are reported in Table 2. We observe that in the zero-shot setting, intermediate-task training improves answer consolidation, while the performance remains far behind supervised models. In the supervised setting, cross-encoders consistently outperform the sentence embedding models. Overall, the answer-aware cross-encoder intermediately tuned on MNLI (A^{2}RoBERTa-MNLI) achieves the best results on all metrics, showing that intermediate-task training on MNLI improves performance. Besides, we find that answer-aware cross-encoders outperforms regular cross-encoders, showing that answers generated by the machine reader provide additional information that helps consolidation.

Model	Classification		Grouping	
	F_1	MCC	ARI	AMI
Supervised Sentence Embedding				
SimCSE-RoBERTa (Q+A)	61.6	57.6	63.4	51.6
SimCSE-RoBERTa (S)	72.1	69.0	78.2	58.1
SimCSE-RoBERTa (S+A)	77.6	75.0	80.4	60.0
SimCSE-RoBERTa (Q+S)	81.1	79.0	**85.2**	**64.9**
SimCSE-RoBERTa (Q+S+A)	**82.5**	**80.4**	85.1	64.6
Supervised Cross-Encoders				
RoBERTa-MNLI (Q+A)	66.7	62.9	75.1	53.4
RoBERTa-MNLI (S)	83.8	81.9	85.3	65.0
RoBERTa-MNLI (S+A)	85.7	84.1	87.5	66.7
RoBERTa-MNLI (Q+S)	88.7	87.4	89.6	68.2
RoBERTa-MNLI (Q+S+A)	**89.0**	**87.8**	**90.4**	**68.9**

Table 3: Results with different input formats on the test set. Q, S, A denotes question, sentences, and answers (generated by UnifiedQA), respectively. RoBERTa-MNLI (Q+S+A) is equivalent to A^{2}RoBERTa-MNLI.

6.3 Ablation Study

In this section, we study the model performance based on different input information given to supervised models. We denote the questions, sentences as Q, S, and answers generated by UnifiedQA as A. The results are shown in Table 3. Overall, models trained on all inputs (Q+S+A) achieve better results than those that have observed only a subset of the available inputs on most metrics. Removing the sentences leads to the largest drops in performance (e.g., 20.9% in F_1 for SimCSE-RoBERTa and 22.3% in F_1 for RoBERTa-MNLI), which shows that sentences provide useful information for answer consolidation. Using sentences only leads to the second-largest drop in performance, showing that without grounding to questions and answers, consolidation is not simply addressed only with the sentences. Besides, removing questions also leads to more significant drops in performance than removing answers (e.g., 4.9% in F_1 for SimCSE-RoBERTa and 2.3% in F_1 for RoBERTa-MNLI). This shows that it is necessary to understand the answer equivalence within the question context in order to consolidate answers.

6.4 Error Analysis

To get a sense of what knowledge is needed to further improve model performance, we examined sentence pairs incorrectly classified as *not in the same group* by A^{2}RoBERTa in the validation and test sets, where 318 out of 2,614 pairs (12.2%) are wrongly classified. We randomly sample 50 such error cases and categorize them by the answer-equivalence type as defined in Table 1. Of the 50

Cause	Description	Example
Entailment (16.7%)	One answer is entailed by the other.	Q: What makes successful people different from average people? S1: Wealthy people are not afraid of failure, unlike average people who often do not even try. S2: The difference between average people and achieving people is their perception of and response to failure.
Entity/commonsense knowledge (23.8%)	The answers refer to the same entity or are equivalent by common sense.	Q: What are some best Hollywood romantic movies to watch? S1: When the subject of romantic movies comes up, one of the first that comes to mind in any list of all-time greats is Casablanca. S2: Humphrey Bogart and Ingrid Bergman's film about love and loss during WWII is basically required viewing for anyone who enjoys romantic movies.
Semantic equivalence (57.1%)	The answers have the same semantic meaning but are expressed using different words.	Q: What is the equity risk premium? S1: Equity Risk Premium is the difference between returns on equity/individual stock and the risk-free rate of return. S2: Let us start with defining the equity risk premium: the Equity Risk Premium is the average extra return demanded by investors, on top of a risk free rate, as a compensation for investing in equity securities with average risk.
Spelling errors (2.4%)	There are spelling errors in the answers.	Q: Are all psychopaths narcissists? S1: I came across that all psychopats are narcissists, but not all narcissists are psychopats. S2: I have read it summed up this way: Not all narcissists are psychopaths, but all psychopaths are narcissists.

Table 4: Different causes of wrongly classified positive pairs.

pairs, 1 (2%) is from *exact match*, 8 (16%) are ambiguous, and the remaining 41 (82%) are from the *semantic variation* category, showing that it is the most challenging type to tackle.

We further study the specific causes of errors on the 42 unambiguous pairs. Examples of distinct error causes are described in Table 4 We find that 16.7% of the falsely classified sentence pairs contain one answer that entails the other instead of expressing the exact same answers, which should however be considered redundant answers by our definition. 80.9% of pairs are equivalent but require understanding the semantic meanings or entity-specific/commonsense knowledge. The rest 2.4% contain spelling errors that negatively affect model inference.

7 Conclusion

In this paper, we formulate and propose the answer consolidation task that seeks to group answers into different aspects. This process can be used to construct a final set of answers that is both comprehensive and non-redundant. We contribute the Question-Answer consolidation dataset (QUASI) for this task and evaluate various models, including sentence embedding models, cross-encoders, and answer-aware cross-encoders. While the best-performing supervised models have achieved promising performance, without that abundant annotation, unsupervised methods still remain far from perfect. This suggests room for further studies on more robust and generalizable solutions for answer consolidation that would largely benefit real-world open-domain QA systems.

Acknowledgement

We appreciate the reviewers for their insightful comments and suggestions. Wenxuan Zhou and Muhao Chen are supported by the National Science Foundation of United States Grant IIS 2105329.

References

Mustafa Abdool, Malay Haldar, Prashant Ramanathan, Tyler Sax, Lanbo Zhang, Aamir Manaswala, Lynn Yang, Bradley Turnbull, Qing Zhang, and Thomas Legrand. 2020. Managing diversity in airbnb search. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2952–2960.

Raviteja Anantha, Svitlana Vakulenko, Zhucheng Tu, Shayne Longpre, Stephen Pulman, and Srinivas Chappidi. 2021. Open-domain question answering goes conversational via question rewriting. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 520–534, Online. Association for Computational Linguistics.

Petr Baudiš and Jan Šedivỳ. 2015. Modeling of the question answering task in the yodaqa system. In *International Conference of the cross-language evaluation Forum for European languages*, pages 222–228. Springer.

Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. 2013. Semantic parsing on Freebase from question-answer pairs. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1533–1544, Seattle, Washington, USA. Association for Computational Linguistics.

Nikita Bhutani, Yoshihiko Suhara, Wang-Chiew Tan, Alon Halevy, and H. V. Jagadish. 2019. Open information extraction from question-answer pairs. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2294–2305, Minneapolis, Minnesota. Association for Computational Linguistics.

Daniel Fernando Campos, Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, Li Deng, and Bhaskar Mitra. 2016. Ms marco: A human generated machine reading comprehension dataset. *ArXiv*, abs/1611.09268.

Zihan Chen, Hongbo Zhang, Xiaoji Zhang, and Leqi Zhao. 2018. Quora question pairs. *URL https://www. kaggle. com/c/quora-question-pairs*.

Tanya Chowdhury and Tanmoy Chakraborty. 2019. Cqasumm: Building references for community question answering summarization corpora. In *Proceedings of the ACM india joint international conference on data science and management of data*, pages 18–26.

Charles LA Clarke, Maheedhar Kolla, Gordon V Cormack, Olga Vechtomova, Azin Ashkan, Stefan Büttcher, and Ian MacKinnon. 2008. Novelty and diversity in information retrieval evaluation. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 659–666.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Xinya Du and Claire Cardie. 2020. Event extraction by answering (almost) natural questions. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 671–683, Online. Association for Computational Linguistics.

Alexander R Fabbri, Xiaojian Wu, Srini Iyer, and Mona Diab. 2021. Multi-perspective abstractive answer summarization. *arXiv preprint arXiv:2104.08536*.

Yixing Fan, Jiafeng Guo, Yanyan Lan, Jun Xu, Chengxiang Zhai, and Xueqi Cheng. 2018. Modeling diverse relevance patterns in ad-hoc retrieval. In *The 41st international ACM SIGIR conference on research & development in information retrieval*, pages 375–384.

Shogo Fujita, Tomohide Shibata, and Manabu Okumura. 2020. Diverse and non-redundant answer set extraction on community QA based on DPPs. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5309–5320, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. SimCSE: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jiawei Han, Jian Pei, and Micheline Kamber. 2011. *Data mining: concepts and techniques*. Elsevier.

Wei He, Kai Liu, Jing Liu, Yajuan Lyu, Shiqi Zhao, Xinyan Xiao, Yuan Liu, Yizhong Wang, Hua Wu, Qiaoqiao She, Xuan Liu, Tian Wu, and Haifeng Wang. 2018. DuReader: a Chinese machine reading comprehension dataset from real-world applications. In *Proceedings of the Workshop on Machine Reading for Question Answering*, pages 37–46, Melbourne, Australia. Association for Computational Linguistics.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. 2017. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1601–1611, Vancouver, Canada. Association for Computational Linguistics.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online. Association for Computational Linguistics.

Daniel Khashabi, Sewon Min, Tushar Khot, Ashish Sabharwal, Oyvind Tafjord, Peter Clark, and Hannaneh Hajishirzi. 2020. UNIFIEDQA: Crossing format boundaries with a single QA system. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1896–1907, Online. Association for Computational Linguistics.

Daniel Khashabi, Amos Ng, Tushar Khot, Ashish Sabharwal, Hannaneh Hajishirzi, and Chris Callison-Burch. 2021. GooAQ: Open question answering with

diverse answer types. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 421–433, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Bernhard Kratzwald, Anna Eigenmann, and Stefan Feuerriegel. 2019. RankQA: Neural question answering with answer re-ranking. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6076–6085, Florence, Italy. Association for Computational Linguistics.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474. Curran Associates, Inc.

Siyi Liu, Sihao Chen, Xander Uyttendaele, and Dan Roth. 2021. MultiOpEd: A corpus of multi-perspective news editorials. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4345–4361, Online. Association for Computational Linguistics.

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. *arXiv preprint arXiv:1907.11692*.

Yuanjie Liu, Shasha Li, Yunbo Cao, Chin-Yew Lin, Dingyi Han, and Yong Yu. 2008. Understanding and summarizing answers in community-based question answering services. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, pages 497–504, Manchester, UK. Coling 2008 Organizing Committee.

Brian W Matthews. 1975. Comparison of the predicted and observed secondary structure of t4 phage lysozyme. *Biochimica et Biophysica Acta (BBA)-Protein Structure*, 405(2):442–451.

Sewon Min, Kenton Lee, Ming-Wei Chang, Kristina Toutanova, and Hannaneh Hajishirzi. 2021. Joint passage ranking for diverse multi-answer retrieval. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6997–7008, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Sewon Min, Julian Michael, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2020. AmbigQA: Answering ambiguous open-domain questions. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5783–5797, Online. Association for Computational Linguistics.

Xuan Vinh Nguyen, Julien Epps, and James Bailey. 2009. Information theoretic measures for clusterings comparison: is a correction for chance necessary? In *ICML*.

Jason Phang, Thibault Févry, and Samuel R. Bowman. 2018. Sentence encoders on stilts: Supplementary training on intermediate labeled-data tasks. *ArXiv*, abs/1811.01088.

William M Rand. 1971. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical association*, 66(336):846–850.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Hongya Song, Zhaochun Ren, Shangsong Liang, Piji Li, Jun Ma, and Maarten de Rijke. 2017. Summarizing answers in non-factoid community question-answering. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*, pages 405–414.

Ellen M. Voorhees. 1999. The trec-8 question answering track report. In *TREC*.

Shuohang Wang, Mo Yu, Jing Jiang, Wei Zhang, Xiaoxiao Guo, Shiyu Chang, Zhiguo Wang, Tim Klinger, Gerald Tesauro, and Murray Campbell. 2018. Evidence aggregation for answer re-ranking in open-domain question answering.(2018). In *Proceedings of the 6th International Conference on Learning Representation, Vancouver, Canada, 2018 April 30-May*, volume 3, pages 1–14.

Wenhui Wang, Nan Yang, Furu Wei, Baobao Chang, and Ming Zhou. 2017. Gated self-matching networks for reading comprehension and question answering. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 189–198, Vancouver, Canada. Association for Computational Linguistics.

Zhiguo Wang, Patrick Ng, Xiaofei Ma, Ramesh Nallapati, and Bing Xiang. 2019. Multi-passage BERT: A globally normalized BERT model for open-domain question answering. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5878–5882, Hong Kong, China. Association for Computational Linguistics.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122, New Orleans, Louisiana. Association for Computational Linguistics.

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 *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Wei Yang, Yuqing Xie, Aileen Lin, Xingyu Li, Luchen Tan, Kun Xiong, Ming Li, and Jimmy Lin. 2019. End-to-end open-domain question answering with BERTserini. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 72–77, Minneapolis, Minnesota. Association for Computational Linguistics.

Wenpeng Yin and Dan Roth. 2018. TwoWingOS: A two-wing optimization strategy for evidential claim verification. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 105–114, Brussels, Belgium. Association for Computational Linguistics.

Wenxuan Zhang, Yang Deng, Jing Ma, and Wai Lam. 2020. AnswerFact: Fact checking in product question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2407–2417, Online. Association for Computational Linguistics.

Fengbin Zhu, Wenqiang Lei, Chao Wang, Jianming Zheng, Soujanya Poria, and Tat-Seng Chua. 2021. Retrieving and reading: A comprehensive survey on open-domain question answering. *arXiv preprint arXiv:2101.00774*.

Ming Zhu, Aman Ahuja, Da-Cheng Juan, Wei Wei, and Chandan K. Reddy. 2020. Question answering with long multiple-span answers. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3840–3849, Online. Association for Computational Linguistics.