MQAG: Multiple-choice Question Answering and Generation for Assessing Information Consistency in Summarization

Potsawee Manakul, Adian Liusie, Mark J. F. Gales
ALTA Institute, Department of Engineering, University of Cambridge
pm574@cam.ac.uk, al826@cam.ac.uk, mjfg@eng.cam.ac.uk

Abstract

State-of-the-art summarization systems can generate highly fluent summaries. These summaries, however, may contain factual inconsistencies and/or information not present in the source. Hence, an important component of assessing the quality of summaries is to determine whether there is information consistency between the source and the summary. Existing approaches are typically based on lexical matching or representation-based methods. In this work, we introduce an alternative scheme based on standard information-theoretic measures in which the information present in the source and summary is directly compared. We propose a Multiple-choice Question Answering and Generation framework, MQAG, which approximates the information consistency by computing the expected statistical distance between summary and source answer distributions over automatically generated multiple-choice questions. This approach exploits multiple-choice answer probabilities, as predicted answer distributions can be compared. We conduct experiments on four summary evaluation datasets: QAG-CNNDM/XSum, XSum-Hallucination, Podcast Assessment, and SummEval. Experiments show that MQAG, using models trained on SQuAD or RACE, outperforms existing evaluation methods on the majority of tasks.[1]

1 Introduction

The objective of summary evaluation is to quantify the quality of summaries, either on a relative or an absolute scale. Accurate and reliable automatic summary evaluation systems are useful to researchers, as they provide an easy and cheap way to compare new summarization models to existing ones. Although current summarization systems have improved dramatically in the last decade, and are capable of generating highly fluent outputs (Lewis et al., 2020; Zhang et al., 2020a; Brown et al., 2020), it has been shown that generated summaries are prone to exhibit factual errors or hallucinations (Kryscinski et al., 2019; Huang et al., 2021; Cao et al., 2022; Ji et al., 2022). Thus, information consistency between the summary and source is an important assessment criterion.

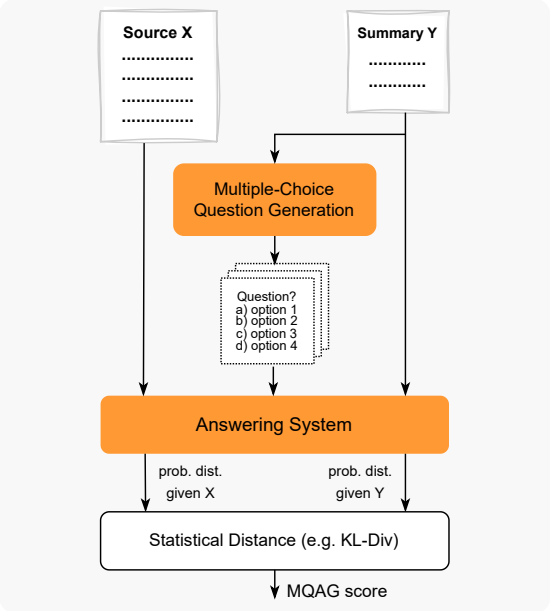


Figure 1: Multiple-choice Question Answering and Generation (MQAG) framework. The answers are represented by probability distributions over choices instead of text spans in existing question-answering approaches.

Existing methods that measure information consistency generally perform lexical matching, either directly such as ROUGE (Lin, 2004) and BLEU (Papineni et al., 2002), or indirectly using more complex representations such as triple matching (Goodrich et al., 2019). Some recent approaches adopt question answering (QA) pipelines to detect factual inconsistencies (Chen et al., 2018; Wang et al., 2020; Durmus et al., 2020; Deutsch et al., 2021; Nan et al., 2021). They are based on the assumption that if the source extracted answer is consistent with the summary extracted answer then

[1]Code and model weights are available at https://github.com/potsawee/mqag0.

the summary and source are consistent. The answers are compared using either lexical matching (Scialom et al., 2019; Wang et al., 2020; Durmus et al., 2020; Scialom et al., 2021) or representation-based matching (Deutsch and Roth, 2022). These span-based QA approaches may have lexical biases, and struggle with highly abstractive summaries or when dealing with multiple answer spans.

In this work, a measure of consistency between the source and summary is defined from an information-theoretic perspective. We propose a Multiple-choice Question Answering and Generation framework, MQAG, where instead of comparing text-based answer spans, multiple-choice questions are generated and the resulting answer distributions from the source and summary are compared. The main contributions of this paper are:

- We provide an alternative and novel question answering-based approach for assessing information consistency. Our approach can represent the answers via probability distributions instead of lexical or embeddings.
- We show that our approach, MQAG, achieves state-of-the-art performance on four out of six summary evaluation tasks.

2 Background and Related Work

Standard summary evaluation metrics such as ROUGE (Lin, 2004) and METEOR (Banerjee and Lavie, 2005) are designed to assess summaries against ground-truth summaries, i.e. reference summaries. However, these metrics have been shown to have a low correlation with human judgements (Fabbri et al., 2021). In practice, there is no ground-truth summary to be used as the reference, and evaluation methods need to compare the summary against the source. Therefore, the scope of this work is assessing the summary against the source.

Although there are several aspects of good summaries, including fluency, coherency, coverage or consistency, generation systems are becoming much more capable of generating fluent texts, so the fluency/coherency aspects are less of a concern compared to consistency and hallucination problems (Ji et al., 2023). Thus, this work focuses on *consistency*. Because the definition of consistent information can depend on one's interpretation, we follow the definition of 'faithfulness' in Maynez et al. (2020) such that we determine if the information in the summary is consistent with information in the source, and we do not consider 'factuality' where valid external facts are acceptable. Existing unsupervised evaluation methods are categorized and explained in the following part.[2]

Textual overlap scores

n-gram based metrics, including BLEU (Papineni et al., 2002), ROUGE (Lin, 2004), and METEOR (Banerjee and Lavie, 2005) measure n-gram overlap between two texts. Instead of n-grams, BERTScore (Zhang et al., 2020b) and BLEURT (Sellam et al., 2020) compare texts in their representation space. These metrics measure textual similarity, so they are not necessarily a good measure of consistency. We note that the original works that proposed these metrics compare the summary against the ground-truth summary, but this work focuses on the scenario where there is no ground-truth summary, and these metrics are used as baselines to compare the summary against the source.

Knowledge representation

Goodrich et al. (2019) assess factual consistency by comparing relation triples from the source and the summary. The relation triples are in the format of Subject-Relation-Object and can be obtained using a model-free method such as OpenIE (Etzioni et al., 2008) or using a trained relation extraction model. The factual accuracy score based on the triple matching approach is then defined as,

$$\text{Score} = \frac{|\mathtt{T}_x \cap \mathtt{T}_y|}{|\mathtt{T}_y|}$$

where $\mathtt{T}_x$ and $\mathtt{T}_y$ are relation triples extracted from the source and the summary, respectively.

Textual Entailment

Simulated data, such as real or fake summaries created by pre-defined transformations, have been used to train classifiers to detect inconsistent summaries (Kryscinski et al., 2020; Bao et al., 2022). Alternatively, (Maynez et al., 2020) trained a textual entailment classifier on the Multi-NLI (MNLI) dataset (Williams et al., 2018). Given a context, the entailment model is to classify the hypothesis into one of the three classes (entail/neutral/contradict). When applied to assess summaries, the context is the source document and the hypothesis is the summary. The probability of being the entail class is

[2] Supervised approaches, with systems trained on human evaluation annotations, are outside the scope of this work.

then used as the consistency score,

$$\text{Score} = P(\text{entail}|x, y) \quad (1)$$

Span-based Question Answering (SpanQAG)

A question-answering approach consists of a question-generation model and an answering model. Given automatically generated questions, the first answer is derived from the source and the second answer is derived from the evaluated summary, and then the two answers are compared.

For example, Eyal et al. (2019) proposed a QA-based method where questions are generated from the ground-truth summary. QAGS (Wang et al., 2020) and FEQA (Durmus et al., 2020) generate questions from the evaluated summary, so these two methods are designed to measure the amount of information in the summary that is consistent with the source. In contrast, SummaQA (Scialom et al., 2019) generates questions from the source document, so it assesses the coverage of the summary. As an extension to the ideas in QAGS/FEQA and SummQA, QuestEval (Scialom et al., 2021) generates questions from both the source and the summary separately to obtain a precision score and a recall score. QuestEval also assigns a weighting function to take into account the importance of each query/question.

Nevertheless, existing QA methods are *span-based* where the answering system extracts answer spans before two answer spans are compared. Due to the nature of span-based answers, answer verification (i.e. answer comparison) is typically through exact matching, token F1, BERTScore, or a learned metric (Deutsch and Roth, 2022). This answer verification illustrates a drawback of the existing QA methods that they have to compare the similarity between two texts. To avoid span-based answer verification, we propose an alternative question answering-based approach where multiple-choice question generation and answering systems are used where the answers are now in the form of probability distributions rather than text spans.

3 Multiple-choice Question Answering and Generation (MQAG)

3.1 Motivation and Theory

Since current summarization systems generate highly fluent summaries, this work focuses on assessing whether summaries contain the same information as that of the source, or whether it is contradictory. One way to view information would be to consider the set of questions that are answerable given a certain passage. If a summary is consistent with the source, then one would expect the set of answerable questions by the summary to overlap with those of the source and yield similar answers. Though span-based QA approaches are similarly motivated, existing span-based frameworks use text similarity measures, either in the form of lexical or representation space. In contrast, we attempt to measure information using multiple-choice questions, which allows for a more abstract understanding of information and enables convenient use of standard information-theoretic measures.

3.2 MQAG Score

Let x = source, y = summary, q = question, and $\mathbf{o}$ = options associated with the question q. We define information inconsistency as,

$$\begin{aligned}&\mathcal{I}(x, y) = \\ &\int_{q,\mathbf{o}} D\left(P_{\mathtt{A}}(\mathbf{o}|q, x), P_{\mathtt{A}}(\mathbf{o}|q, y)\right) P_{\mathtt{G}}(q, \mathbf{o}|y)\mathrm{d}\mathbf{o}\mathrm{d}q \\ &\approx \frac{1}{N}\sum_{i=1}^{N} D\left(P_{\mathtt{A}}(\mathbf{o}^{(i)}|q^{(i)}, x), P_{\mathtt{A}}(\mathbf{o}^{(i)}|q^{(i)}, y)\right)\end{aligned} \quad (2)$$

where $\{q^{(i)}, \mathbf{o}^{(i)}\}$ is sampled from $P_{\mathtt{G}}(q, \mathbf{o}|y)$, the question-option generation model, $P_{\mathtt{A}}(\mathbf{o}^{(i)}|q^{(i)}, x)$ and $P_{\mathtt{A}}(\mathbf{o}^{(i)}|q^{(i)}, y)$ are the option distributions given the source and summary respectively, and D is a statistical distance such as KL-divergence. Based on the information inconsistency score in Equation 2, we define the MQAG score as,[3]

$$\text{MQAG-Score}(x, y) = 1 - \mathcal{I}(x, y) \quad (3)$$

We refer to Equation 3 as the **MQAG-Sum** score as the questions are generated from the summary. Furthermore, it is possible to generate questions, $\{q, \mathbf{o}\}$ using the source x instead of the summary y, $\{q^{(i)}, \mathbf{o}^{(i)}\}$ is sampled from $P_{\mathtt{G}}(q, \mathbf{o}|x)$. We will refer to this variant as the **MQAG-Src** score. MQAG-Src is expected to measure the amount of source information present in the summary, i.e. the coverage of the summary, while MQAG-Sum is expected to measure the consistency of the summary with respect to the source. To account for consistency and coverage, we also consider a simple combination,

$$\text{MQAG-F1} = 2 \cdot \frac{\text{MQAG-Sum} \times \text{MQAG-Src}}{\text{MQAG-Sum} + \text{MQAG-Src}} \quad (4)$$

[3] If $D > 1$, for example, when using KL-divergence, the MQAG score can be negative, but the maximum value is 1.0.

3.3 Statistical Distances D

Given two probability distributions over options $\mathbf{o}$ (e.g. one conditioned on source x, and the other conditioned on summary y), a statistical distance D measures the distance between the probability distributions. There are multiple distances, which can be used, and in this work, we consider some of the main distances and investigate their properties as well as their empirical performance in our MQAG framework as follows,

- KL-Divergence:

$$D_{\texttt{KL}} = \sum_{o \in \mathbf{o}} P_{\texttt{A}}(o|q,x) \log \left(\frac{P_{\texttt{A}}(o|q,x)}{P_{\texttt{A}}(o|q,y)} \right)$$

- One-Best (i.e. argmax matching):

$$D_{\texttt{OB}} = \begin{cases} 0, & \text{if } o_x = o_y \\ 1, & \text{otherwise} \end{cases}$$

  where $o_x = \arg\max_o P_{\texttt{A}}(o|q,x)$ and $o_y = \arg\max_o P_{\texttt{A}}(o|q,y)$. $D_{\texttt{OB}}$ simply determines whether the two answers match or not.

- Total Variation:

$$D_{\texttt{TV}} = \frac{1}{2} \left\| P_{\texttt{A}}(\mathbf{o}|q,x) - P_{\texttt{A}}(\mathbf{o}|q,y) \right\|_1$$

- Hellinger:

$$D_{\texttt{HL}} = \frac{1}{\sqrt{2}} \left\| \sqrt{P_{\texttt{A}}(\mathbf{o}|q,x)} - \sqrt{P_{\texttt{A}}(\mathbf{o}|q,y)} \right\|_2$$

KL divergence is unbounded, which means the value can be exceedingly large. In contrast, one-best is bounded but discontinuous. Both total variation and Hellinger distance are bounded and continuous. We illustrate examples of the properties of these statistical distances on Bernoulli distributions in Figure 4 in the appendix.

4 Experimental Setup

4.1 System Development Data

RACE (Lai et al., 2017) is a multiple-choice reading comprehension dataset where each example consists of context, question, answer, and 3 distractors (i.e. incorrect options). SQuAD (Rajpurkar et al., 2016) is a collection of question-answer pairs derived from Wikipedia articles, and the correct answers can be any sequence of tokens in the given context. The statistics are provided in Table 1 where *abstractiveness* is measured by 1.0 minus the length of the longest sequence that exists in both the context and the answer per the answer length, i.e. $1.0 - \text{ROUGE-L}_{\text{Precision}}(\text{Answer}, \text{Context})$.

Dataset	Size	Length		Abstractive
		Context	Answer	
SQuAD	98.2k	317.8	11.0	0.0%
RACE	97.7k	138.3	11.3	39.1%

Table 1: Statistics of datasets for training MQAG systems. Length = #words. Abstractiveness of 0% indicates that in SQuAD the answer always exists in the context.

4.2 Evaluation Data

We evaluate the performance by measuring the correlation against human judgements at the summary level on QAG-(CNNDM (Hermann et al., 2015), XSum (Narayan et al., 2018)), XSum-Hallucination and at the system level on Podcast Assessment and SummEval, and the definitions of summary-level and system-level correlations are provided in Appendix C. The statistics are provided in Table 2.

Eval Dataset	Size	Length	
		Source	Summary
QAG-CNNDM	235	355.8	54.4
QAG-XSum	239	403.7	19.7
XSum-H	2500	442.1	20.5
Podcast	*20×179	5950	88.3
SummEval	*16×100	404.0	63.7

Table 2: Statistics of evaluation datasets. Length is the number of words calculated using the NLTK tokenizer. *#systems×documents.

QAG. Wang et al. (2020) annotated 235 CNNDM summaries of the system in Gehrmann et al. (2018) and 239 XSum summaries of fine-tuned BART (Lewis et al., 2020). The annotation was performed at the sentence level indicating if hallucination occurs or not. Subsequently, for each summary, the faithfulness (or consistency) score is then obtained by averaging all sentence-level human scores.

XSum-Hallucination (XSum-H). Maynez et al. (2020) annotated 2500 XSum summaries using 3 crowd-sourced workers on two metrics: 1) Faithfulness = whether the information is faithful w.r.t. the source at the token level. The judgements are then averaged; 2) Factuality = whether the summary level is factual w.r.t source and external knowledge.

Podcast Assessment. Manakul and Gales (2022) compiled 3580 podcast summaries of abstraction and extractive summarization systems from Spotify Podcast Challenge 2020 (Jones et al., 2021). The

human evaluation was performed on a 4-point scale considering a combination of consistency, coverage, and fluency.

SummEval. Fabbri et al. (2021) assessed 1600 CNNDM summaries from 16 different summarization systems on four aspects, including relevancy, consistency, coherency, and fluency. In this work, we use the consistency scores.

4.3 Baselines

All of the considered methods compare the summary y against the source document x without the ground-truth summary, and we implement these methods as described in Section 2 using code/repository from the relevant previous works.

ROUGE. We use the ROUGE-1 (F1) score in the `rouge-score` Python package.

OpenIE-TripleMatch. The relation extraction is based on an open scheme, and we use the implementation in FactSumm (Heo, 2021).

BERTScore. We use DeBERTa-base (He et al., 2021) fine-tuned to MNLI as the backbone.

Entailment model. Following the method in Maynez et al. (2020), we trained BERT-large (Devlin et al., 2019) on MNLI and we use the probability of the source being entailed by the summary as the assessment score as shown in Equation 1.

Span-based QAG Baselines. We use three existing span-based question-answering methods as our baselines: QAGS proposed by Wang et al. (2020), FEQA proposed by Durmus et al. (2020), and QuestEval proposed by Scialom et al. (2021).

4.4 MQAG Implementation

Question Generation (`G1`, `G2`)

The multiple-choice question generation is implemented in two stages.[4] First model `G1` generates the question q and answer a, then model `G2` generates the distractors $\mathbf{o}_{\setminus a}$ given q and a.

$$P_{\texttt{G}}(q, \mathbf{o}|y) = P_{\texttt{G2}}(\mathbf{o}_{\setminus a}|q, a, y)P_{\texttt{G1}}(q, a|y) \quad (5)$$

where $\mathbf{o} = \{a, \mathbf{o}_{\setminus a}\}$ denotes all options/choices. We set the number of options to four. Both `G1` and `G2` are sequence-to-sequence T5-large models (Raffel et al., 2020). The question-answer generation system `G1` is fine-tuned to either RACE or SQuAD, and the distractor generation system `G2` is fine-tuned to RACE.

Question Answering (`A`)

The answering stage contains one model `A`, which is Longformer-large (Beltagy et al., 2020) with a multiple-choice setup following Yu et al. (2020); Raina and Gales (2022). The input to the model is a concatenation of context, question and option. The answering model `A` is fine-tuned to RACE.

Answerability of Generated Questions

Because not all generated questions are of high quality, we consider filtering out low-quality questions through question-context answerability measures (Kundu and Ng, 2018; Hu et al., 2019). We consider a simple answerability measure based on the entropy of the probability distribution over the options. We define the effective number of options,

$$\mathcal{N}_y(q, \mathbf{o}) = 2^{\mathcal{H}[P_{\texttt{A}}(\mathbf{o}|q,y)]} \quad (6)$$

where $\mathcal{H}(.)$ is base-2 entropy, so $\mathcal{N}_y(q, \mathbf{o})$ ranges from 1.0 to the number of options, e.g. 4.0. When q is generated from y but $\mathcal{N}_y(q, \mathbf{o})$ is high, this question q should be deemed *unanswerable* as it is not answerable even when using the same context. As a result, we use $\mathcal{N}_y(q, \mathbf{o})$ as an answerability criterion to *reject* questions which have $\mathcal{N}_y(q, \mathbf{o})$ higher than a threshold denoted by $\mathcal{N}_y^\tau$.

5 Experimental Results

5.1 Analysis of the Components in MQAG

In this subsection, we carry out experiments to find the best configuration of MQAG, including the analysis of statistical distances, variants of MQAG, and answerability. We build two MQAG variants: $\text{MQAG}_{\text{SQuAD}}$ and $\text{MQAG}_{\text{RACE}}$, which differ in the training data of the question+answer generator `G1`, while the distractor generator `G2` and answering system `A` are both trained on RACE.

Statistical Distances

In Table 3, our results compare statistical distances. It can be seen that in both configurations, KL-divergence yields lower correlations than other distances, and on average total variation slightly outperforms Hellinger and one-best distances. Hence, total variation will be used as the main distance. The next observation is that $\text{MQAG}_{\text{SQuAD}}$, despite generating more extractive questions, achieves higher correlations than $\text{MQAG}_{\text{RACE}}$ on most tasks except on Podcast and SummEval.

[4] The motivation is based on our initial experiments that a single generation system (generating the question and 4 options together) often gave low-quality distractors, and using two generation systems improved the quality of distractors.

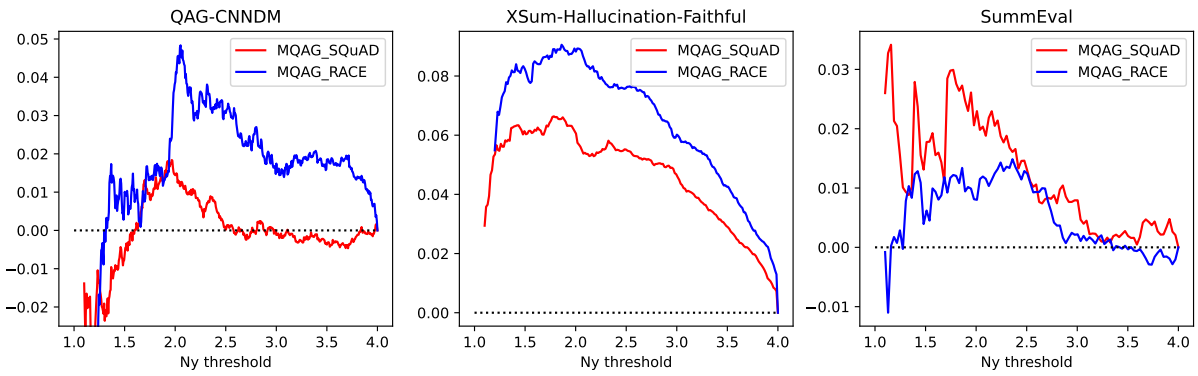


Figure 2: ΔPCC of MQAG-Sum with total variation (i.e. PCC − $\text{PCC}_{N_y^\tau=4.0}$) against the answerability threshold $\mathcal{N}_y^\tau$ on X-axis. MQAG without answerability is equivalent to setting $N_y^\tau = 4.0$, and the results at this operating point can be seen on the right-most point in each plot. As we reduce the threshold ($N_y^\tau \downarrow$), more questions are rejected. The results on QAG-XSum and Podcast are provided in Figure 5 in the appendix.

D	QAG CNN	QAG XSum	XSum-H Faith	XSum-H Fact	Podc	SumE
MQAG-Sum, G1 = SQuAD						
D_{KL}	0.478	0.374	0.177	0.226	0.251	0.936
D_{OB}	0.476	0.354	0.295	0.254	0.677	0.872
D_{TV}	0.508	0.396	0.269	0.267	0.225	0.870
D_{HL}	0.499	0.399	0.266	0.269	0.201	0.870
MQAG-Sum, G1 = RACE						
D_{KL}	0.450	0.283	0.135	0.179	0.789	0.954
D_{OB}	0.453	0.225	0.240	0.221	0.839	0.928
D_{TV}	0.462	0.309	0.221	0.244	0.770	0.933
D_{HL}	0.473	0.323	0.215	0.244	0.751	0.927

Table 3: Comparison of Statistical Distances using MQAG-Sum without answerability.

	QAG CNN	QAG XSum	XSum-H Faith	XSum-H Fact	Podc	SumE
G1 = SQuAD, D = Total Variation						
Sum	0.508	0.396	0.269	0.267	0.225	0.870
Src	0.272	0.017	0.093	0.037	0.470	0.707
F1	0.490	0.393	0.286	0.261	0.475	0.863
G1 = RACE, D = Total Variation						
Sum	0.462	0.309	0.221	0.244	0.770	0.933
Src	0.233	0.143	0.069	0.087	0.144	0.588
F1	0.468	0.301	0.217	0.252	0.731	0.866

Table 4: Comparison of MQAG-Src, MQAG-Sum, and MQAG-F1 without answerability.

MQAG-Sum, MQAG-Src, MQAG-F1

Here, we compare three variants of MQAG scores. Our results in Table 4 show that MQAG-Src, which assesses how much source information is contained in the summary by generating questions from the source, achieves lower PCCs than MQAG-Sum on all datasets. This finding aligns with our expectation, as the summaries were graded by humans predominantly on the consistency aspect (which MQAG-Sum was designed to measure) rather than the quantity of source information present (which MQAG-Src measures). When combining MQAG-Src and MQAG-Sum into MQAG-F1, we only observe a small gain on two test settings. Therefore, MQAG-Sum is selected as our main MQAG configuration for the remaining investigations.

Answerability

In Figure 2, the answerability is swept from 4.0 (keeping all questions) to 1.0 (only keeping those that the answering system A is highly confident). It can be seen that as we filter out high-entropy questions, there is an upward trend in performance across all tasks. In addition, as shown in the figure, setting $\mathcal{N}_y^\tau$ at 2.0 seems to be a reasonable answerability threshold. At this threshold, $\mathcal{N}_y^\tau = 2.0$, out of 50 automatically generated questions, about 36 questions are kept for $\text{MQAG}_{\text{SQuAD}}$ and about 30 questions are kept for $\text{MQAG}_{\text{RACE}}$. The number of remaining questions is similar across all datasets as shown in Table 9 in the appendix. Thus, we set $\mathcal{N}_y^\tau = 2.0$, and the performance of MQAG using this answerability criterion is presented and compared against baseline systems in Table 5.

5.2 Comparison Against Existing Baselines

The baseline and MQAG results are shown in Table 5. The observation is that MQAG achieves a higher correlation than the best SpanQAG on 5 out of 6 tasks. When compared to all existing baselines, MQAG achieves state-of-the-art performance on 4 out of 6 tasks. To investigate the impact of the abstractiveness of summaries on the performance,

Method	QAG		XSum-H		Podcast	SumEvl
	CNNDM	XSum	Faithful	Factual		
Baselines: Other Approaches						
ROUGE-1	0.337	0.012	-0.050	0.008	0.326	0.458
OpenIE-TripleMatching	0.381	0.131	0.019	-0.020	0.706	0.548
BERTScore	0.584	0.008	0.185	0.154	0.718	0.645
Entailment (BERT Model)	0.159	0.169	0.362	0.209	0.228	0.619
Baselines: SpanQAG						
QAGS	0.437	0.200	0.101	0.080	0.464	0.812
FEQA	0.322	0.283	0.297	0.171	0.603	0.464
QuestEval	0.250	0.173	0.421	0.197	0.579	0.838
Multiple-choice Question Answering and Generation (MQAG)						
$\text{MQAG}_{\text{SQuAD}}$	0.519	0.407	0.324	0.292	0.502	0.890
$\text{MQAG}_{\text{RACE}}$	0.502	0.313	0.306	0.270	0.855	0.945

Table 5: Pearson Correlation Coefficient (PCC) between the scores of summary evaluation methods and human judgements. PCCs are computed at the summary level on QAG and XSum-H, and at the system level on Podcast and SummEval. PCCs on Podcast are computed on 15 abstractive systems. Our best performing MQAG configuration consists of (i) generation stage G generates questions from summary y (i.e. MQAG-Sum), (ii) statistical distance is total variation, (iii) the answerability threshold $\mathcal{N}_y^\tau$ is set to 2.0. Underline denotes where MQAG outperforms the best SpanQAG system, which is 5 out of 6 tasks. When compared to all baselines, MQAG achieves the highest PCC on 4 out of 6 tasks. The results of all MQAG configurations are provided in Table 10, and Spearman's correlation results are provided in Table 11 in the appendix.

we split QAG-XSum and XSum-H datasets[5] into two portions of the same size by abstractiveness as measured by the longest sequence in the summary that exists in the source per the summary length (i.e. ROUGE-L precision of summary y using source x as the reference). The results in Table 6 show that although $\text{MQAG}_{\text{RACE}}$ achieves lower PCCs than $\text{MQAG}_{\text{SQuAD}}$ (in Table 5), when evaluated on the more abstractive split, the performance $\text{MQAG}_{\text{RACE}}$ is much closer to that of $\text{MQAG}_{\text{SQuAD}}$. In addition, compared to MQAG, SpanQAG methods show a larger drop in PCCs in the more abstractive split. This finding further illustrates the benefits of comparing answer distributions rather than text spans.

Method	QAG-XSum		XSum-H	
	Low	High	Low	High
QAGS	0.190	0.184	0.101	0.159
FEQA	0.296	0.163	0.290	0.124
QuestEval	0.215	0.061	0.398	0.326
$\text{MQAG}_{\text{SQuAD}}$	0.431	0.328	0.334	0.254
$\text{MQAG}_{\text{RACE}}$	0.277	0.295	0.319	0.249

Table 6: Performance as measured by Pearson correlation coefficient on the *low* abstractiveness and *high* abstractiveness of QAG-XSum and XSum-H (Faithful). The results on the entire datasets are in Table 5.

6 Ablation Studies

6.1 Number of Questions (N)

We analyse the impact of the number of generated questions on the performance of MQAG. The mean and standard deviation are presented in Figure 3. The results show a smooth increase in correlation, which is as expected because the framework is based on a Monte-Carlo approximation (in Equation 2), and a similar finding was also observed in QAGS (Wang et al., 2020). Figure 3 also shows that the variance decreases with N, showing the stability of the approach. Though the performance curve has not completely plateaued at N=50, since the computational cost of MQAG scales linearly with N, 50 questions seem to be a reasonable compromise between computational efficiency and performance. An interesting next step would be to investigate if the same or similar performance can be achieved with as low N as possible, for example, by generating a smaller but more diverse set of questions and options such as varifocal question generation where questions are generated based on different focal points (Ousidhoum et al., 2022).

[5]XSum summaries are more abstractive than CNNDM summaries, so using XSum should enable us to investigate the impact of abstractiveness better than CNNDM.

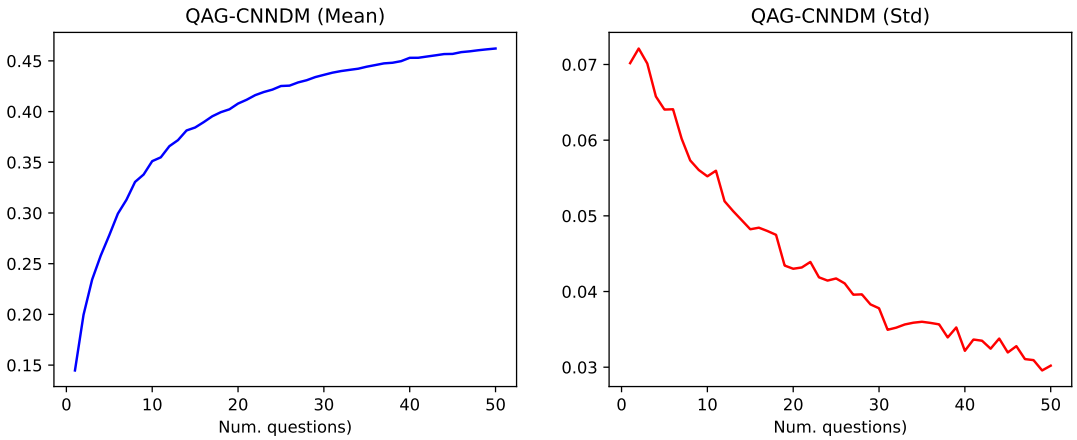


Figure 3: Mean and standard deviation of Pearson correlation (Y-axis) of $\text{MQAG}_{\text{RACE}}$ on QAG-CNNDM when the number of generated questions N is varied from 1 to 50 (X-axis). Standard deviation is obtained via bootstrapping. The results on other datasets are provided in Figure 6 in the appendix.

6.2 Model Choices

Pre-trained Backbone

We investigate model choices by swapping to less capable models, e.g. T5-large → T5-base for generation, and Longformer(4096) → RoBERTa(512) (Liu et al., 2019) for answering. The results in Table 8 in the appendix show: (1) For generation stage, using a smaller model does not result in lower performance. This could be because T5-base has higher perplexity, and yields more diverse questions. (2) In contrast, for answering stage, when using RoBERTa, with a shorter input length, the performance on SummEval (the input length is mostly shorter than 512) remains almost the same. However, as the input length is longer in other datasets, we observe a drop in PCC when using RoBERTa.

Zero-shot Multiple-choice Question Generation

Given the impressive results of large language models (LLMs) across natural language generation tasks, we investigate the performance of LLMs in a zero-shot fashion instead of using fine-tuned T5 for multiple-choice question generation. Specifically, we use OpenAI GPT-3 (Brown et al., 2020) (text-davinci-003) where we query 50 questions and 4 options using the following prompt format:

```
Write 50 diverse multiple-choice questions with 4 options from the following context: {context}.
```

We found that GPT-3 generated 50 questions as specified in the prompt around 26% of the examples and the remaining only have 20 questions. The majority of questions (more than 95%) have 4 options, while the remaining have 2 options. In Table 7, the results show that zero-shot GPT-3 performs worse than our fine-tuned T5 systems in both multiple-choice question generation tasks. This illustrates that there is some sensitivity due to the quality of generated questions, and using our fine-tuned T5 is a better option than zero-shot GPT-3.

Backbone	QAG	
	CNNDM	XSum
T5 (SQuAD)	0.508	0.396
T5 (RACE)	0.462	0.309
GPT-3	0.392	0.130

Table 7: GPT-3 versus fine-tuned T5 using D_{TV} without answerability for multiple-choice question generation.

7 Conclusion

This work proposes MQAG – a novel scheme for assessing information consistency between source and summary based on the distance between multiple-choice answer distributions instead of text-based answer spans in existing question-answering methods. Our experiments demonstrate the potential of this alternative approach which outperforms existing techniques on various datasets. The realization of the framework exploits current multiple-choice question generation and answering systems. Its performance is expected to increase as backbone systems improve, for example, the diversity of questions generated and the selection of options. Also, the framework is highly interpretable, allowing more insight into summary assessment.

Limitations

Domain. Our approach is designed to assess the information content, so it may not work well with other aspects of summary evaluation such as fluency or coherency. Our analysis is based on the systems trained on RACE, which is collected from English examinations in China. Hence, the generated questions and answer distributions could be biased towards the style of the examinations.

Efficiency. Given the realization of the MQAG framework where two generators `G1` and `G2` are adopted, the MQAG framework can be slow when using old infrastructure, for example, it takes around 3 seconds per question on one NVIDIA P100 GPU. To address this issue, future work could explore a more efficient realization of MQAG.

Acknowledgments

This work is supported by Cambridge University Press & Assessment (CUP&A), a department of The Chancellor, Masters, and Scholars of the University of Cambridge, and the Cambridge Commonwealth, European & International Trust. We would like to thank the anonymous reviewers for their helpful comments.

References

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.

Forrest Bao, Ge Luo, Hebi Li, Minghui Qiu, Yinfei Yang, Youbiao He, and Cen Chen. 2022. SueNes: A weakly supervised approach to evaluating single-document summarization via negative sampling. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2450–2458, Seattle, United States. Association for Computational Linguistics.

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Meng Cao, Yue Dong, and Jackie Cheung. 2022. Hallucinated but factual! inspecting the factuality of hallucinations in abstractive summarization. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3340–3354, Dublin, Ireland. Association for Computational Linguistics.

Ping Chen, Fei Wu, Tong Wang, and Wei Ding. 2018. A semantic qa-based approach for text summarization evaluation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Daniel Deutsch, Tania Bedrax-Weiss, and Dan Roth. 2021. Towards question-answering as an automatic metric for evaluating the content quality of a summary. *Transactions of the Association for Computational Linguistics*, 9:774–789.

Daniel Deutsch and Dan Roth. 2022. Benchmarking answer verification methods for question answering-based summarization evaluation metrics. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 3759–3765, Dublin, Ireland. Association for Computational Linguistics.

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.

Esin Durmus, He He, and Mona Diab. 2020. FEQA: A question answering evaluation framework for faithfulness assessment in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5055–5070, Online. Association for Computational Linguistics.

Oren Etzioni, Michele Banko, Stephen Soderland, and Daniel S Weld. 2008. Open information extraction from the web. *Communications of the ACM*, 51(12):68–74.

Matan Eyal, Tal Baumel, and Michael Elhadad. 2019. Question answering as an automatic evaluation metric for news article summarization. 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 3938–3948, Minneapolis, Minnesota. Association for Computational Linguistics.

Alexander R. Fabbri, Wojciech Kryściński, Bryan McCann, Caiming Xiong, Richard Socher, and Dragomir Radev. 2021. SummEval: Re-evaluating summarization evaluation. *Transactions of the Association for Computational Linguistics*, 9:391–409.

Sebastian Gehrmann, Yuntian Deng, and Alexander Rush. 2018. Bottom-up abstractive summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4098–4109, Brussels, Belgium. Association for Computational Linguistics.

Ben Goodrich, Vinay Rao, Peter J. Liu, and Mohammad Saleh. 2019. Assessing the factual accuracy of generated text. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '19, page 166–175, New York, NY, USA. Association for Computing Machinery.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. Deberta: Decoding-enhanced bert with disentangled attention. In *International Conference on Learning Representations*.

Hoon Heo. 2021. Factsumm: Factual consistency scorer for abstractive summarization. `https://github.com/Huffon/factsumm`.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc.

Minghao Hu, Furu Wei, Yuxing Peng, Zhen Huang, Nan Yang, and Dongsheng Li. 2019. Read+ verify: Machine reading comprehension with unanswerable questions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6529–6537.

Yi-Chong Huang, Xiachong Feng, Xiaocheng Feng, and Bing Qin. 2021. The factual inconsistency problem in abstractive text summarization: A survey. *ArXiv*, abs/2104.14839.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *ACM Comput. Surv.*, 55(12).

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Yejin Bang, Andrea Madotto, and Pascale Fung. 2022. Survey of hallucination in natural language generation. *ArXiv*, abs/2202.03629.

Rosie Jones, Ben Carterette, Ann Clifton, Maria Eskevich, Gareth JF Jones, Jussi Karlgren, Aasish Pappu, Sravana Reddy, and Yongze Yu. 2021. Trec 2020 podcasts track overview. *arXiv preprint arXiv:2103.15953*.

Wojciech Kryscinski, Nitish Shirish Keskar, Bryan McCann, Caiming Xiong, and Richard Socher. 2019. Neural text summarization: A critical evaluation. 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 540–551, Hong Kong, China. Association for Computational Linguistics.

Wojciech Kryscinski, Bryan McCann, Caiming Xiong, and Richard Socher. 2020. Evaluating the factual consistency of abstractive text summarization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9332–9346, Online. Association for Computational Linguistics.

Souvik Kundu and Hwee Tou Ng. 2018. A nil-aware answer extraction framework for question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4243–4252, Brussels, Belgium. Association for Computational Linguistics.

Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. 2017. RACE: Large-scale ReAding comprehension dataset from examinations. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 785–794, Copenhagen, Denmark. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. 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*.

Potsawee Manakul and Mark JF Gales. 2022. Podcast Summary Assessment: A resource for evaluating summary assessment methods. *arXiv preprint arXiv:2208.13265*.

Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. 2020. On faithfulness and factuality in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1906–1919, Online. Association for Computational Linguistics.

Feng Nan, Cicero Nogueira dos Santos, Henghui Zhu, Patrick Ng, Kathleen McKeown, Ramesh Nallapati, Dejiao Zhang, Zhiguo Wang, Andrew O. Arnold, and Bing Xiang. 2021. Improving factual consistency of abstractive summarization via question answering. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6881–6894, Online. Association for Computational Linguistics.

Shashi Narayan, Shay B. Cohen, and Mirella Lapata. 2018. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1797–1807, Brussels, Belgium. Association for Computational Linguistics.

Nedjma Ousidhoum, Zhangdie Yuan, and Andreas Vlachos. 2022. Varifocal question generation for fact-checking. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 2532–2544, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*.

Vatsal Raina and Mark Gales. 2022. Answer uncertainty and unanswerability in multiple-choice machine reading comprehension. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1020–1034, Dublin, Ireland. Association for Computational Linguistics.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Thomas Scialom, Paul-Alexis Dray, Sylvain Lamprier, Benjamin Piwowarski, Jacopo Staiano, Alex Wang, and Patrick Gallinari. 2021. QuestEval: Summarization asks for fact-based evaluation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6594–6604, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Thomas Scialom, Sylvain Lamprier, Benjamin Piwowarski, and Jacopo Staiano. 2019. Answers unite! unsupervised metrics for reinforced summarization models. 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 3246–3256, Hong Kong, China. Association for Computational Linguistics.

Thibault Sellam, Dipanjan Das, and Ankur Parikh. 2020. BLEURT: Learning robust metrics for text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7881–7892, Online. Association for Computational Linguistics.

Alex Wang, Kyunghyun Cho, and Mike Lewis. 2020. Asking and answering questions to evaluate the factual consistency of summaries. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5008–5020, Online. 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.

Weihao Yu, Zihang Jiang, Yanfei Dong, and Jiashi Feng. 2020. Reclor: A reading comprehension dataset requiring logical reasoning. In *International Conference on Learning Representations*.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter Liu. 2020a. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In *International Conference on Machine Learning*, pages 11328–11339. PMLR.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020b. Bertscore: Evaluating text generation with BERT. In *International Conference on Learning Representations*.

A More Details about Models and Data

Training QG and QA systems

We train the question+answer generation model (`G1`) on RACE or SQuAD, and train the distractor generation model (`G2`) and the answering model (`A`) on RACE. We do early stopping when the performance on the validation set does not improve. We use batch size 8 for `G1` and `G2` models (T5) and 2 for `A` model (Longformer). The learning rate is set to 1e-6, and we use the Adam optimizer. We carried out training on one NVIDIA A100-80GB GPU. Training one generation model (T5-large) takes around 8 hours, and training the answering

model (Longformer-4096) takes up to 2 days. Running MQAG inference with generation=T5-large and answering=Longformer-4096 on one NVIDIA P100 GPU takes around 3 seconds per question.

Licenses

The licenses of the datasets are CC-BY-4.0 for XSum-Hallucination and Podcast Assessment, and MIT license for SummEval. For QAG, we were unable to find its license. The licenses of T5 and Longformer backbone models are apache-2.0.

Open-Sourcing Trained Models

To allow the trained models in MQAG to be used for *research* purposes in other question generation and answering tasks, we have made them available online. The links to these models on HuggingFace can be found on our project page at https://github.com/potsawee/mqag0.

B Statistical Distances

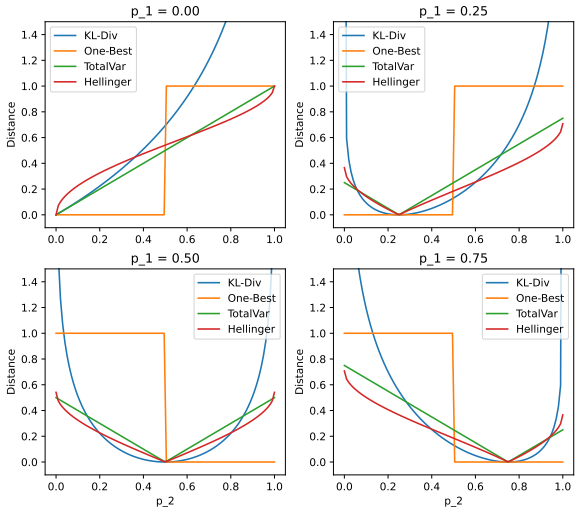


Figure 4: Statistical distances between two Bernoulli distributions $\mathbf{p}_1 = [p_1; 1-p_1]$ and $\mathbf{p}_2 = [p_2; 1-p_2]$ at different values of p_1. We show 4 plots of different values of p_1 = 0.00, 0.25, 0.50, 0.75, and Y-axis represents distance D and X-axis represents p_2. It can be seen that KL divergence is unbounded, which means the value can be exceedingly large. One-best, in contrast, is bounded between 0.0 and 1.0; however, one-best is discontinuous. Total variation and Hellinger distance are continuous and bounded between 0.0 and 1.0.

C Computing Correlation

Following the notation in Deutsch et al. (2021), let z_i^j and $\bar{z}_i^j$ be two scores of metrics Z and $\bar{Z}$ for the summary output by system $i \in \{1, ..., N\}$ on the document $j \in \{1, ..., M\}$. In this work, Z is the evaluation method, and $\bar{Z}$ is the human judgement. The correlations, e.g. Pearson or Spearman's rank correlation coefficient, are defined as follows:

- System-level (i.e. Corpus-level)

$$\rho = \text{Corr}\left(\left\{\frac{\sum_j z_i^j}{M}, \frac{\sum_j \bar{z}_i^j}{M}\right\}_{i=1}^{N}\right)$$

- Summary-level (i.e. Sentence-level)

$$\rho = \frac{1}{M}\sum_j \text{Corr}\left(\left\{z_i^j, \bar{z}_i^j\right\}_{i=1}^{N}\right)$$

D Additional Results

D.1 Ablation: Model Choices

For generation models, we measure cross-entropy losses on RACE-testset:

- T5-base (223M): G1 = 1.612, G2 = 1.875
- T5-large (738M): G1 = 1.478, G2 = 1.741

where G1 denotes question+answer generation, and G2 denotes distractor generation. For answering models, we measure accuracy on RACE-testset:

- Roberta (355M): Accuracy = 84.84
- Longformer (435M): Accuracy = 81.67

Model		Pearson Corr.		
Generation	Answering	SumE	QAG-X	Podc
T5-base	RoBERTa	0.949	0.242	0.471
T5-base	Longformer	0.949	0.293	0.647
T5-large	RoBERTa	0.930	0.211	0.350
T5-large	Longformer	0.930	0.229	0.772

Table 8: Ablation on model choices in MQAG using N=20. SumE = SummEval (Consistency aspect), QAG-X = QAG-XSum, Podc = Podcast Assessment.

D.2 MQAG Results

Here, we provide results that are complementary to those presented in the main text. Figure 5 illustrates the answerability results on QAG-XSum and Podcast, and Figure 6 illustrates the impact of N on the remaining datasets not presented in the main text. Table 10 shows the results of all MQAG configurations. Table 11 shows the Spearman's rank correlation coefficient of the main results.

Method	QAG-CNNDM	QAG-XSum	XSum-H	Podcast	SummEval
$\text{MQAG}_{\text{SQuAD}}$	35.0	37.4	34.0	34.7	37.0
$\text{MQAG}_{\text{RACE}}$	30.5	30.0	30.0	30.5	31.1

Table 9: The number remaining questions at $\mathcal{N}_y^\tau = 2.0$.

MQAG Configuration				QAG		XSum-H		Podcast	SumEvl
G's Inp.	G1-trained	Dist.	Ans.	CNNDM	XSum	Faithful	Factual		
Src x	SQuAD	$D_{\texttt{KL}}$	✗	0.219	0.008	0.070	0.027	0.432	0.726
Src x	SQuAD	$D_{\texttt{OB}}$	✗	0.264	0.003	0.165	0.064	0.788	0.703
Src x	SQuAD	$D_{\texttt{TV}}$	✗	0.272	0.017	0.093	0.037	0.470	0.707
Src x	SQuAD	$D_{\texttt{HL}}$	✗	0.266	0.010	0.081	0.032	0.517	0.713
Sum y	SQuAD	$D_{\texttt{KL}}$	✗	0.478	0.374	0.177	0.226	0.251	0.936
Sum y	SQuAD	$D_{\texttt{OB}}$	✗	0.476	0.354	0.295	0.254	0.677	0.872
Sum y	SQuAD	$D_{\texttt{TV}}$	✗	0.508	0.396	0.269	0.267	0.225	0.870
Sum y	SQuAD	$D_{\texttt{HL}}$	✗	0.499	0.399	0.266	0.269	0.201	0.870
F1	SQuAD	$D_{\texttt{KL}}$	✗	0.508	0.361	0.197	0.213	0.531	0.921
F1	SQuAD	$D_{\texttt{OB}}$	✗	0.416	0.161	0.296	0.199	0.825	0.869
F1	SQuAD	$D_{\texttt{TV}}$	✗	0.490	0.393	0.286	0.261	0.475	0.863
F1	SQuAD	$D_{\texttt{HL}}$	✗	0.481	0.387	0.274	0.255	0.487	0.862
Sum y	SQuAD	$D_{\texttt{KL}}$	$\mathcal{N}_y$	0.483	0.396	0.229	0.249	0.545	0.943
Sum y	SQuAD	$D_{\texttt{OB}}$	$\mathcal{N}_y$	0.517	0.385	0.286	0.256	0.711	0.914
Sum y	SQuAD	$D_{\texttt{TV}}$	$\mathcal{N}_y$	0.519	0.407	0.324	0.292	0.502	0.890
Sum y	SQuAD	$D_{\texttt{HL}}$	$\mathcal{N}_y$	0.512	0.413	0.323	0.299	0.385	0.889
Src x	RACE	$D_{\texttt{KL}}$	✗	0.143	0.097	0.088	0.054	0.321	0.599
Src x	RACE	$D_{\texttt{OB}}$	✗	0.226	0.091	0.160	0.091	0.534	0.612
Src x	RACE	$D_{\texttt{TV}}$	✗	0.233	0.143	0.069	0.087	0.144	0.588
Src x	RACE	$D_{\texttt{HL}}$	✗	0.221	0.148	0.056	0.083	0.222	0.592
Sum y	RACE	$D_{\texttt{KL}}$	✗	0.450	0.283	0.135	0.179	0.789	0.954
Sum y	RACE	$D_{\texttt{OB}}$	✗	0.453	0.225	0.240	0.221	0.839	0.928
Sum y	RACE	$D_{\texttt{TV}}$	✗	0.462	0.309	0.221	0.244	0.770	0.933
Sum y	RACE	$D_{\texttt{HL}}$	✗	0.473	0.323	0.215	0.244	0.751	0.927
F1	RACE	$D_{\texttt{KL}}$	✗	0.480	0.266	0.156	0.198	0.830	0.908
F1	RACE	$D_{\texttt{OB}}$	✗	0.379	0.192	0.268	0.206	0.796	0.815
F1	RACE	$D_{\texttt{TV}}$	✗	0.468	0.301	0.217	0.252	0.731	0.866
F1	RACE	$D_{\texttt{HL}}$	✗	0.472	0.317	0.206	0.252	0.693	0.858
Sum y	RACE	$D_{\texttt{KL}}$	$\mathcal{N}_y$	0.460	0.302	0.208	0.206	0.857	0.961
Sum y	RACE	$D_{\texttt{OB}}$	$\mathcal{N}_y$	0.466	0.233	0.266	0.226	0.822	0.954
Sum y	RACE	$D_{\texttt{TV}}$	$\mathcal{N}_y$	0.502	0.313	0.306	0.270	0.855	0.945
Sum y	RACE	$D_{\texttt{HL}}$	$\mathcal{N}_y$	0.501	0.328	0.305	0.273	0.860	0.936

Table 10: Pearson correlation coefficients of all MQAG configurations. Our MQAG results are based on N=50. When applying the answerability mechanism, the threshold $\mathcal{N}_y^\tau$ is set to 2.0.

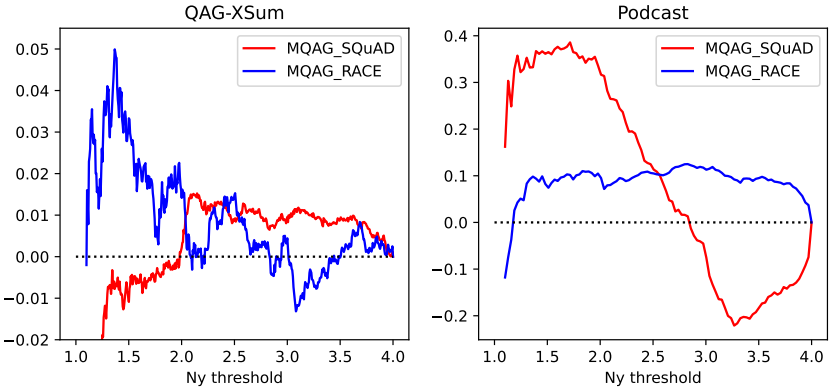


Figure 5: ΔPCC of MQAG-Sum with total variation against the answerability threshold $\mathcal{N}_y^\tau$ on the X-axis. This figure extends Figure 2 in the main text.

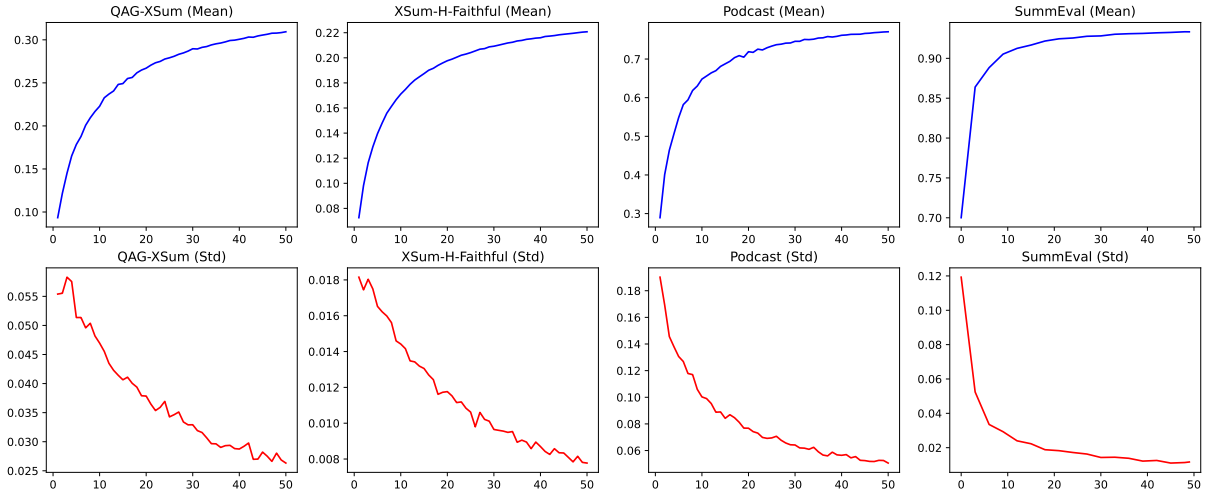


Figure 6: Mean (top row) and standard deviation (bottom row) of Pearson correlation (Y-axis) of $\text{MQAG}_{\text{RACE}}$ when the number of generated questions N is varied from 1 to 50 (X-axis). This figure extends Figure 3 in the main text.

Method	QAG		XSum-H		Podcast	SumEvl
	CNNDM	XSum	Faithful	Factual		
Baselines: Other Approaches						
ROUGE-1	0.318	0.053	-0.030	0.001	0.282	0.627
OpenIE-TripleMatching	0.337	0.130	0.019	-0.025	0.700	0.671
BERTScore	0.523	0.018	0.183	0.153	0.686	0.835
Entailment (BERT Model)	0.167	0.190	0.380	0.202	0.207	0.141
Baselines: SpanQAG						
QAGS	0.341	0.166	0.085	0.052	0.357	0.421
FEQA	0.275	0.277	0.300	0.155	0.504	0.270
QuestEval	0.181	0.175	0.415	0.176	0.425	0.812
Multiple-choice Question Answering and Generation (MQAG)						
$\text{MQAG}_{\text{SQuAD}}$	0.470	0.409	0.335	0.284	0.441	0.773
$\text{MQAG}_{\text{RACE}}$	0.460	0.308	0.322	0.266	0.779	0.920

Table 11: Spearman's rank correlation coefficient between the scores of summary evaluation methods and human judgements. This table is complementary to Table 5 which reports Pearson's correlation coefficient results.

Source: A G4S security van has been robbed outside a branch of royal bank of Scotland in Glasgow city centre. Police said three armed men took a five-figure sum from the vehicle in the city's Sauchiehall street on Monday at about 21:45. A spokesman said no-one had been injured although two security guards aged 47 and 49 were left badly shaken. The area around the bank, which is near the Buchanan galleries shopping centre, has been cordoned off by police. Police said the security guards had been making their delivery when they were approached by the three armed men, who threatened them and demanded they hand over a box of money. It is understood the cash taken was in the region of £50,000. Following the robbery, the three men got into a white seat Leon car, which sped off along west Nile street towards the cowcaddens area. [...]
Summary: Two security guards have been threatened during a robbery at a bank in Edinburgh.
Generated question (using summary): The robbery happened in _ . **Generated options (using summary)**: (1) Edinburgh (2) a bank (3) a shop (4) a small town.
Prob. over options given Source: `0.077, 0.895, 0.018, 0.010` **Prob. over options given Summary**: `0.687, 0.295, 0.000, 0.018`

Table 12: Example from QAG-XSum (documentID=1). Factual inconsistency in the summary is highlighted in red.