Two is Better than Many? Binary Classification as an Effective Approach to Multi-Choice Question Answering

Deepanway Ghosal[♣] **Navonil Majumder**[♣]
Rada Mihalcea[♠] **Soujanya Poria**[♣]
[♣] DeCLaRe Lab, Singapore University of Technology and Design, Singapore
[♠] University of Michigan, USA
{deepanway_ghosal@mymail.,navonil_majumder@,sporia@}sutd.edu.sg
mihalcea@umich.edu

Abstract

We propose a simple refactoring of multi-choice question answering (MCQA) tasks as a series of binary classifications. The MCQA task is generally performed by scoring each (question, answer) pair normalized over all the pairs, and then selecting the answer from the pair that yield the highest score. For *n* answer choices, this is equivalent to an *n*-class classification setup where only one class (true answer) is correct. We instead show that classifying (question, true answer) as positive instances and (question, false answer) as negative instances is significantly more effective across various models and datasets. We show the efficacy of our proposed approach in different tasks – abductive reasoning, commonsense question answering, science question answering, and sentence completion. Our DeBERTa binary classification model reaches the top or close to the top performance on public leaderboards for these tasks. The source code of the proposed approach is available at https://github.com/declare-lab/TEAM.

1 Introduction

Starting with the early Text Retrieval Conference (TREC) community-wide evaluations of textual question answering (Voorhees et al., 1999), all the way to the recent work on multimodal question answering (Lei et al., 2018; Tapaswi et al., 2016; Jang et al., 2017; Castro et al., 2020) and commonsense question answering (Sap et al., 2019; Talmor et al., 2019), the task has become a staple of the natural language processing research community. One of the major challenges encountered in question answering is the evaluation, which often requires human input to evaluate the textual answers thoroughly. Because of this, the alternative that has been proposed is that of *multi-choice question answering*, where the correct answer is provided together with other incorrect answers. The task is thus transformed into that of answer classification, where a system has to select one answer from the choices provided. While there are drawbacks associated with this evaluation metric, it has been widely adopted because of its benefit of providing a clear evaluation methodology.

In this paper, we reformulate the task of multi-choice question answering as a binary classification task and show that this re-framing leads to significant performance improvements on several datasets. Importantly, this formulation brings flexibility to the overall question-answering setup, as it reduces the dependence on the up-front availability of multiple candidate answers. Using our method – TEAM (**T**wo is b**E**tter th**A**n **M**any), candidate answers can be produced and evaluated for correctness on the fly, and thus the answer classification component can be also used in conjunction with more natural settings that use open-ended answer generation (Castro et al., 2022; Sadhu et al., 2021).

2 Methodology

Let q be a question for which multiple answer choices $\mathcal{A} = \{a_1, \ldots, a_n\}$ are given. Optionally, there is some context c which could be helpful for answering the question. The objective is to select the correct answer a_k from the answer set $\mathcal{A}$.

For some of the datasets used in the paper, the question q is not provided, and the answer is based only on the context c. For example, SWAG and HellaSwag are two such datasets where the task is to choose the best possible ending for sentence completion, as shown in Table 1. In this case, the question q can be assumed as implicit: *What is the best possible ending for the context?* The sentence to be completed is considered as the context c.

We discuss how the MCQA task is generally performed using transformer language models in §2.1. We denote this approach as **Score-based Method** or Score method . We then discuss our proposed **Binary Classification-based Method**, TEAM in §2.2.

Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 10158–10166
December 7-11, 2022

2.1 Score-based Method (`Score`)

We use the notation introduced earlier in §2. Given question q, optional context c, and the answer choices $\mathcal{A} = \{a_1, a_2, \ldots, a_n\}$, n different input sequences are constructed each containing the concatenation of the question q, context c, and one possible answer choice a_i. The sequences are independently encoded through a pre-trained transformer language model such as RoBERTa (Liu et al., 2019) or DeBERTa (He et al., 2021). A score s_i is predicted for each input sequence which is then normalized with a softmax layer across the n outputs to obtain score q_i.

The cross-entropy loss is used to train the encoder model. Assuming the answer a_k is correct, the loss can be obtained as follows:

$$\mathcal{L} = -\sum_{i=1}^{n} p_i log(q_i) = -log(q_k) \quad (1)$$

where p_i are considered as the class labels. The class p_k corresponding to the gold answer a_k is valued as 1, and all other classes are valued as 0. The loss is equivalent to the cross-entropy loss in a n-class classification setup. The normalization of the scores using the softmax layer to obtain a distribution over the answer choices is also analogous to the probability distribution over the different classes in the multi-class classification setup.

The choice providing the highest score is the predicted answer during inference. The `Score` method was used for the SWAG task in BERT (Devlin et al., 2019), StoryCloze task in GPT (Radford et al., 2018) and has been used for all MCQA tasks in the huggingface transformers[1] framework.

2.2 Classification-based Method (`TEAM`)

For our proposed classification-based method, we first extend the pre-trained language model by adding a classification head with two nodes. The values of these two nodes will denote the unnormalized scores for the negative and positive classes in our classification setup.

Now, similar to the previous `Score` method, we first construct n different input sequences by concatenating the question q, the optional context c, and each possible answer choice a_i. We then obtain the unnormalized negative and positive scores s_i^- and s_i^+ for each sequence by independently encoding them through the modified language model. We normalize each pair of scores through a softmax layer to obtain probabilities of negative and positive classes: q_i^- and q_i^+, respectively.

We consider the sequence corresponding to the gold answer a_k as positive, and all the other sequences as negative. Therefore, the loss function takes the following form:

$$\begin{aligned}\mathcal{L} &= -\sum_{i=1}^{n}(p_i^+ log(q_i^+) + p_i^- log(q_i^-)) \\ &= -log(q_k^+) - \sum_{i=1, i\neq k}^{n} log(q_i^-)\end{aligned} \quad (2)$$

where p_i^+ and p_i^- are considered as the class labels. As a_k is the gold answer, we use $p_k^+ = 1$, $p_k^- = 0$ and $p_i^+ = 0$, $p_i^- = 1$, when $i \neq k$.

Although Eq. (2) is a suitable loss function for single correct answer cases, it can be easily extended for instances or datasets with multiple correct answers. This can be done by changing the class labels p_i^+ and p_i^- to positive and negative appropriately for the additional correct answers.

During inference, we choose the answer with the highest positive class probability as the predicted answer. We will show later in §4 that the `TEAM` method generally outperforms the `Score` method across several datasets for the same choice of transformer models.

3 Experimental Datasets

We experiment with the following datasets:

Abductive NLI (Bhagavatula et al., 2020). Given two observations o_1 and o_2 (considered as context c), the goal is to select the more plausible intermediate event among hypotheses h_1 and h_2. We use the sequences $\{o_1, h_1, o_2\}$ and $\{o_1, h_2, o_2\}$ as input for both the `Score` and `TEAM` method. Assuming h_1 is the gold answer, we classify $\{o_1, h_1, o_2\}$ as positive; $\{o_1, h_2, o_2\}$ as negative.

CommonsenseQA (Talmor et al., 2019) or CQA is a dataset for commonsense QA based on knowledge encoded in ConceptNet (Speer et al., 2017). Given a question, there are five possible choices , among which only one is correct. We do not use any additional knowledge or context for this task.

CommonsenseQA 2.0 (Talmor et al., 2021) or CQA2 is a recent challenging QA dataset collected with a model-in-the-loop approach. The dataset contains commonsense questions from various reasoning categories with either *yes* or *no* answer.

QASC (Khot et al., 2020) or Question Answer-

[1] https://github.com/huggingface/transformers

Dataset	Instance
CQA	**Question:** Where on a river can you hold a cup upright to catch water on a sunny day?
	Choice 1: Waterfall **Choice 2:** Bridge ... **Choice 5:** Mountain
QASC	**Question:** Differential heating of air can be harnessed for what?
	Choice 1: electricity production **Choice 2:** running and lifting **Choice 3:** animal survival ... **Choice 8:** reducing acid rain
SWAG	**Partial Event:** On stage, a woman takes a seat at the piano. She
	Ending 1: sits on a bench as her sister plays with the doll. ... **Ending 4:** nervously sets her fingers on the keys.
PIQA	**Goal:** To separate egg whites from the yolk using a water bottle, you should
	Solution 1: Squeeze the water bottle and press it against the yolk. Release, which creates suction and lifts the yolk. **Solution 2:** Place the water bottle and press it against the yolk. Keep pushing, which creates suction and lifts the yolk.

Table 1: Illustration of some of the datasets used in this work. The answers highlighted in green are the correct answers. CQA: Commonsense QA, PIQA: Physical IQA.

ing via Sentence Composition task requires fact retrieval from a large corpus and composing them to answer a multi-choice science question. Each question q has eight choices, among which one is correct. We use the question and choices without any retrieved facts for this task. We evaluate another task setup **QASC-IR** (information retrieval) where we use two-step IR retrieved facts as in Khot et al. (2020) as additional context c.

SWAG, HellaSwag (Zellers et al., 2018, 2019) are two datasets for grounded commonsense inference, where the objective is to find the correct ending given a partial description of an event. We consider the partial description as the context c. The correct ending is to be chosen from a pool of four possible choices.

Social IQA (SIQA) (Sap et al., 2019) is a dataset for commonsense reasoning about social interactive situations. Given a question about a social situation context, the objective is to select the correct answer from three possible choices.

Physical IQA (PIQA) (Bisk et al., 2020) is designed to investigate physical knowledge of language models. The task is to select the correct solution for a goal from two given choices.

CosmosQA (Huang et al., 2019) is a QA dataset for commonsense-based reading comprehension. Given a question about a paragraph (c), the task is to select the correct answer among four choices.

CICERO v1, v2 (Ghosal et al., 2022; Shen et al., 2022) are datasets for contextual commonsense reasoning in dialogues. Given the dialogue and a question about an utterance, the task is to choose the correct answer among multiple choices. We modify the original datasets to use them in a MCQA setup. More details are in the appendix.

4 Results

We use the RoBERTa Large (Liu et al., 2019) and DeBERTa Large (He et al., 2021) model to benchmark the `Score` and `TEAM` method across the experimental datasets. We report the accuracy for the validation set in Table 2 and accuracy of leaderboard submissions for the test set in Table 3. We also report results for other QA systems such as UnifiedQA (Khashabi et al., 2020) and UNICORN (Lourie et al., 2021) for the test set (wherever available) in Table 3.

Our main finding is that the `TEAM` method improves over the `Score` method for most of the datasets except Social IQA, Physical IQA, and CICERO v1. We observe this result for both the RoBERTa and DeBERTa models.

Abductive Reasoning: The improvement is consistently large for both validation and test set in the Abductive NLI (ANLI) dataset. The problem of intermediate hypothesis selection transforms into a problem of plausible story selection as we use the sequence $\{o_1, h, o_2\}$ as our input. In this formulation, the `TEAM` method is significantly better than the `Score` method for both RoBERTa and DeBERTa models.

Science QA: We also observe considerable improvements in the QASC dataset without and with the additional retrieved knowledge. The RoBERTa-`TEAM` model is more than 7% better in the test set when retrieved knowledge is not used. The difference in performance is around 3% and 4.5% in the validation and test set when the retrieved knowledge is used. For DeBERTa, we observe the most significant improvement in the test results of the QASC-IR setting, where the `TEAM` method is 3.7% better than the `Score` method.

Commonsense QA and Sentence Ending Prediction: The `TEAM` method is also better than the `Score` method for commonsense question-answering in CommonsenseQA and CommonsenseQA 2.0 across most settings. One notable instance is the 3% superior score of the DeBERTa `TEAM` in the CommonsenseQA 2.0 validation set. We observe a similar trend in results for sentence-ending prediction in SWAG and HellaSwag. The improvement in performance for the `TEAM` method is between 0.85-1.9% in the test set. We also notice improvements in the test set results for reading comprehension QA in CosmosQA.

Model	Method	ANLI	CQA	CQA2	QASC	QASC-IR	SWAG	H-SWAG	SIQA	PIQA	CosmosQA	CICERO* v1	CICERO* v2
RoBERTa Large	Score	85.25	73.63	54.76	53.46	77.21	89.23	83.89	**78.15**	**78.89**	**80.44**	**80.33**	85.25
	TEAM	**87.47**	**75.32**	**55.83**	**57.24**	**80.35**	**89.49**	**84.52**	76.49	76.71	80.37	77.54	**86.53**
DeBERTa Large	Score	89.75	**83.75**	66.63	74.41	89.31	93.14	94.67	**80.82**	**87.81**	86.13	**86.60**	89.06
	TEAM	**92.23**	83.34	**69.57**	**75.33**	**91.09**	**93.27**	**95.47**	80.27	86.07	**86.35**	84.48	**90.59**

Table 2: Accuracy on the validation split of the datasets. All numbers are the average of five runs with different seeds.

Model	Method	ANLI	CQA2	QASC	QASC-IR	SWAG	H-SWAG	SIQA	PIQA	CosmosQA	CICERO* v1	CICERO* v2
RoBERTa Large	Score	83.91	55.44	46.52	73.26	88.97	81.70	**76.70**	**79.40**	80.71	**83.28**	89.61
	TEAM	**87.04**	**56.73**	**53.80**	**77.93**	**89.88 (7)**	**83.63**	75.96	74.55	**80.84**	79.94	**89.81**
DeBERTa Large	Score	89.74	67.37	71.74	85.65	92.37 (2)	94.72 (4)	**80.18**	**87.41 (4)**	85.51	**88.04**	92.67
	TEAM	**92.20 (1)**	**68.38 (9)**	**74.35**	**89.35 (3)**	**94.12 (1)**	**95.57 (2)**	79.89	85.90 (5)	**86.86 (5)**	86.84	**93.25**
UnifiedQA 11B	-	-	-	**78.50**	**89.60**	-	-	81.40	89.50	-	-	-
UNICORN 11B	-	87.30	**70.20**	-	-	-	93.90	**83.20**	**90.10**	**91.80**	-	-

Table 3: Accuracy on the test split of the datasets. Numbers on the parentheses indicate rank on the leaderboard (if in the top 10) at the time of submission to the leaderboard. Numbers in purple indicate results for RoBERTa Large as reported in the UNICORN paper (Lourie et al., 2021). We do not report results for CommonsenseQA (CQA) test set as test labels are not publicly available and there is no automated submission leaderboard.

Dialogue Commonsense Reasoning: We observe contrasting results in CICERO v1 and v2. The Score method outperforms the TEAM method by around 2-3% in CICERO v1. However, the TEAM method is better in CICERO v2 for both RoBERTa and DeBERTa models. We analyze the results in more detail in §5.1.

Negative Results: The Score method outperforms the TEAM method in Physical IQA (PIQA) and CICERO v1. These two datasets contain answer choices that are lexically close together and subtly different from each other (example in Table 1). We analyze the results in more detail in §5.1. The Score method is also the better performing method in SIQA, with small improvements over the TEAM method in DeBERTa and comparatively large improvements in RoBERTa. We surmise that the Score method is better because the dataset contains complex social commonsense scenarios, for which learning by directly comparing the options is more effective.

State-of-the-Art Models and Leaderboard Submissions: We also report the results for UnifiedQA and UNICORN 11B models for the test set in Table 3. We compare these results against our best-performing model: DeBERTa Large in classification setup (DeBERTa-TEAM). DeBERTa-TEAM maintains parity with UnifiedQA 11B in QASC-IR, despite being 36 times smaller. UNICORN 11B outperforms DeBERTa-TEAM by a large margin on SIQA, PIQA, and CosmosQA. It is an expected result as UNICORN is trained on multiple datasets for commonsense reasoning starting from the T5-11B checkpoint and then fine-tuned on each target dataset. DeBERTa-TEAM is, however, considerably better in Abductive NLI and HellaSwag. DeBERTa-TEAM also reached the top or close to the top of the leaderboard (at the time of submission to the leaderboard) in Abductive NLI, SWAG, HellaSwag, and QASC.

5 Analysis

5.1 How Does Similar Answer Choices Affect Performance?

We analyze the similarity between the correct and incorrect choices to understand why the TEAM method is better than the Score method in most of the datasets and vice-versa in the others. We report the lexical similarity with BLEU (Papineni et al., 2002), ROUGE-L (Lin, 2004), and semantic similarity with *all-mpnet-base-v2* sentence transformer (Reimers and Gurevych, 2019) in Table 4. We also report the difference in performance between TEAM and Score models for RoBERTa and DeBERTa in the Δ columns.

The similarity measurements in Table 4 indicate that the datasets can be clearly segregated into two groups – one with low to medium similarity, and the other with very high similarity. Interestingly, the Δ values are mostly positive for the low to medium similarity group, and all negatives for the high similarity group. We surmise that the difference between the very similar correct and incor-

Dataset	BLEU1	BLEU4	ROUGE	Sem-Sim	Δ_1	Δ_2
ANLI	21.84	7.81	24.61	46.02	2.22	2.48
CQA	1.48	0	1.31	30.75	1.69	-0.41
QASC	3.15	0.95	2.08	25.71	3.14	1.78
SWAG	12.78	0.81	11.61	30.47	0.26	0.13
H-SWAG	18.55	1.18	16.14	46.95	0.63	0.80
SIQA	12.56	3.99	10.41	29.17	-1.66	-0.55
CosmosQA	32.37	13.31	24.66	35.29	-0.07	0.22
CICEROv2	30.00	7.50	33.85	44.23	1.28	1.53
PIQA	81.97	72.77	74.01	82.50	-2.18	-1.74
CICEROv1	73.17	53.96	74.98	74.12	-2.79	-2.12

Table 4: Average similarity between correct and incorrect answer choices in the validation set for different datasets. Numbers are shown on a scale of 0-100. Δ_1 and Δ_2 indicate difference in performance between TEAM and Score methods for RoBERTa and DeBERTa in validation set.

rect choices are better captured through the softmax activation over the answers in the Score method. However, this aspect is not captured in the TEAM method, as sequences corresponding to the correct and incorrect choices are separately classified as positive or negative. Thus, the Score method is more effective when the answer choices are very similar, as in PIQA or CICERO v1.

5.2 How Accurate is the Binary Classifier?

We evaluate how often input sequences corresponding to correct and incorrect answers are predicted accurately with DeBERTA-TEAM binary classification model in Table 5. The binary classifier model is more likely to predict all answers as negative than all answers as positive, as it learns from more negative choices in most datasets. Interestingly, however, the model predicts all positive answers for 25.63% instances in PIQA, which is significantly higher than all the other datasets. This is one of the sources of error in PIQA, as the model often predicts both choices as positive, but assigns a higher positive probability to the incorrect choice. We also report the % of instances for which the correct answer is predicted as positive and all incorrect answers are predicted as negative in the **Accurate** column. The accuracy is highest in HellaSWAG and lowest in QASC, which corelates well with the highest performance in HellaSWAG and second lowest performance in QASC across the datasets in Table 2 and Table 3.

5.3 Error Analysis

We show some examples of incorrect predictions for the DeBERTa-TEAM model in the CommonsenseQA and PIQA dataset in Table 6. The erroneously predicted answers in CommonsenseQA are often very close in meaning to the correct answers. Furthermore, the incorrectly predicted answer could also be argued as correct for some instances (second example in Table 6), as the incorrect choice is also equally plausible. In PIQA however, the model make mistakes where complex scientific and physical world knowledge is required. The incorporation of external knowledge is likely necessary to answer these questions accurately.

Dataset	DeBERTa-TEAM Predicted All				
	Neg	Pos	Incor as Neg	Cor as Pos	Accurate
CQA	17.69	0.08	70.35	76.99	52.66
CQA2	1.81	6.53	65.17	69.89	63.36
QASC	37.37	0.0	80.45	55.29	43.09
SWAG	13.2	0.05	86.97	85.0	73.77
H-SWAG	15.63	0.01	94.69	83.39	79.06
SIQA	20.93	2.61	73.69	72.36	52.76
PIQA	19.37	25.63	70.46	76.71	51.09
CosmosQA	19.33	0.2	78.32	76.21	58.99
CICEROv1	22.62	0.37	80.60	71.80	57.44
CICEROv2	11.26	2.64	79.40	85.71	68.14

Table 5: DeBERTA-TEAM binary classification results. The **Neg** and **Pos** column indicate % of instances for which all answer choices are predicted as negative or positive. The **Incor as Neg**, **Cor as Pos**, and **Accurate** column indicate % of instances for which all incorrect answers are predicted as negative, the correct answer is predicted as positive, and all answers are predicted accurately as negative or positive. **Accurate** is the intersection of **Incor as Neg** and **Cor as Pos**.

Dataset: CommonsenseQA. **Question**: Though the thin film seemed fragile, for it's intended purpose it was actually nearly what? **Correct Answer**: Indestructible. **Predicted Answer**: Unbreakable.
Dataset: CommonsenseQA. **Question**: She was always helping at the senior center, it brought her what? **Correct Answer**: Happiness. **Predicted Answer**: Satisfaction.
Dataset: PIQA. **Goal**: To discourage house flies from living in your home, **Correct Answer**: keep basil plants in the kitchen or windows. **Predicted Answer**: keep lavender plants in the kitchen or window.
Dataset: PIQA. **Goal**: To cook perfectly golden pancakes, **Correct Answer**: keep the temperature low for a longer time. **Predicted Answer**: keep the temperature high and cook quickly.

Table 6: Some examples of incorrect predictions in CommonsenseQA and PIQA.

6 Conclusion

In this paper, we introduced a simple binary classification method as an alternative way to address multi-choice question answering (MCQA) tasks. Through evaluations on ten different MCQA benchmarks, we showed that this simple method generally exceeds the performance of the score-based method traditionally used in the past. We believe this approach can also be used in the more natural open-ended answer generation setups, thus providing a "bridge" between the MCQA and answer generation frameworks for question answering.

7 Limitations

Although the method we introduced is more flexible than the answer scoring approach typically used for MCQA, it still lacks the full flexibility of open-ended question answering and assumes the availability of a candidate answer that it can classify as correct or incorrect.

Additionally, even if our approach outperforms the score-based methods for most of the benchmarks we considered, there are still some datasets (e.g., SIQA, PIQA, CICERO v1), where the score-based method performs best. We leave it for future work to identify a principled approach for selecting the best methodology to use for a given dataset.

Acknowledgement

This research/project is supported by the National Research Foundation, Singapore, and the Ministry of National Development, Singapore under its Cities of Tomorrow R&D Programme (CoT Award COT-V2-2020-1). Any opinions, findings, and conclusions, or recommendations expressed in this material are those of the author(s) and do not reflect the views of the National Research Foundation, Singapore, and the Ministry of National Development, Singapore. This research is also supported by A*STAR under its RIE 2020 AME programmatic grant RGAST2003 and the Ministry of Education, Singapore, under its AcRF Tier-2 grant (Project no. T2MOE2008, and Grantor reference no. MOET2EP20220-0017). Any opinions, findings, conclusions, or recommendations expressed in this material are those of the author(s) and do not reflect the views of the Ministry of Education, Singapore.

References

Chandra Bhagavatula, Ronan Le Bras, Chaitanya Malaviya, Keisuke Sakaguchi, Ari Holtzman, Hannah Rashkin, Doug Downey, Wen-tau Yih, and Yejin Choi. 2020. Abductive commonsense reasoning. In *ICLR*.

Yonatan Bisk, Rowan Zellers, Jianfeng Gao, Yejin Choi, et al. 2020. Piqa: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 7432–7439.

Santiago Castro, Mahmoud Azab, Jonathan Stroud, Cristina Noujaim, Ruoyao Wang, Jia Deng, and Rada Mihalcea. 2020. LifeQA: A real-life dataset for video question answering. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 4352–4358, Marseille, France. European Language Resources Association.

Santiago Castro, Ruoyao Wang, Pingxuan Huang, Ian Stewart, Oana Ignat, Nan Liu, Jonathan Stroud, and Rada Mihalcea. 2022. FIBER: Fill-in-the-blanks as a challenging video understanding evaluation framework. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2925–2940, 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.

Deepanway Ghosal, Siqi Shen, Navonil Majumder, Rada Mihalcea, and Soujanya Poria. 2022. Cicero: A dataset for contextualized commonsense inference in dialogues. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5010–5028.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2021. Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing. *arXiv preprint arXiv:2111.09543*.

Lifu Huang, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2019. Cosmos qa: Machine reading comprehension with contextual commonsense reasoning. 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 2391–2401.

Yunseok Jang, Yale Song, Youngjae Yu, Youngjin Kim, and Gunhee Kim. 2017. TGIF-QA: Toward spatio-temporal reasoning in visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2758–2766.

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.

Tushar Khot, Peter Clark, Michal Guerquin, Peter Jansen, and Ashish Sabharwal. 2020. Qasc: A dataset for question answering via sentence composition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8082–8090.

Jie Lei, Licheng Yu, Mohit Bansal, and Tamara Berg. 2018. TVQA: Localized, compositional video question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1369–1379, Brussels, Belgium. Association for Computational Linguistics.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

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*.

Ilya Loshchilov and Frank Hutter. 2018. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Nicholas Lourie, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2021. Unicorn on rainbow: A universal commonsense reasoning model on a new multitask benchmark. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 13480–13488.

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.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

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*. Association for Computational Linguistics.

Arka Sadhu, Kan Chen, and Ram Nevatia. 2021. Video question answering with phrases via semantic roles. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2460–2478, Online. Association for Computational Linguistics.

Maarten Sap, Hannah Rashkin, Derek Chen, Ronan Le Bras, and Yejin Choi. 2019. Social iqa: Commonsense reasoning about social interactions. 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 4463–4473.

Siqi Shen, Deepanway Ghosal, Navonil Majumder, Henry Lim, Rada Mihalcea, and Soujanya Poria. 2022. Multiview contextual commonsense inference: A new dataset and task. *arXiv preprint arXiv:2210.02890*.

Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Thirty-first AAAI conference on artificial intelligence*.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. Commonsenseqa: A question answering challenge targeting commonsense knowledge. 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 4149–4158.

Alon Talmor, Ori Yoran, Ronan Le Bras, Chandra Bhagavatula, Yoav Goldberg, Yejin Choi, and Jonathan Berant. 2021. Commonsenseqa 2.0: Exposing the limits of ai through gamification. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*.

Makarand Tapaswi, Yukun Zhu, Rainer Stiefelhagen, Antonio Torralba, Raquel Urtasun, and Sanja Fidler. 2016. MovieQA: Understanding stories in movies through question-answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4631–4640.

Ellen M Voorhees et al. 1999. The trec-8 question answering track report. In *Trec*, volume 99, pages 77–82.

Rowan Zellers, Yonatan Bisk, Roy Schwartz, and Yejin Choi. 2018. SWAG: A large-scale adversarial dataset for grounded commonsense inference. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 93–104, Brussels, Belgium. Association for Computational Linguistics.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. Hellaswag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4791–4800.

A Experimental Details

We train all the score-based and classification-based models with the AdamW (Loshchilov and Hutter, 2018) optimizer with a learning rate of 1e-6, 3e-6, 5e-6, 1e-5, 3e-5. We train all the models for 8 epochs. The best models are chosen based on the results on the validation set. The RoBERTa-Large and DeBERTa-Large models have 355M and 304M parameters, respectively.

B Computational Resources

We use a single Quadro RTX 8000 GPU for our experiments. Training takes between 30 minutes to 8 hours for the different datasets used in the paper.

C Dataset Details

All datasets used in this paper are in English language. The datasets are available in the corresponding leaderboard websites[2] or through the huggingface datasets hub[3].

The number of MCQA instances in the training, validation and test set of the various datasets are shown in Table 7. Some example instances from the datasets are shown in Table 8.

Dataset	Train	Validation	Test
Abductive NLI	169,654	1,532	3,040
Commonsense QA	9,741	1,221	1,140
Commonsense QA 2.0	9,264	2,541	2,473
QASC / QASC IR	8,134	926	920
SWAG	73,546	20,006	20,005
HellaSwag	39,905	10,042	10,050
PIQA	16,113	1,838	3,446
SIQA	33,410	1,954	2,059
CosmosQA	25,262	2,985	6,963
CICERO v1	27,225	9,470	9,064
CICERO v2	13,496	2,806	4,150

Table 7: Number of MCQA instances in the train, validation, and test set for the experimental datasets.

D Modifications in CICERO

CICERO v1 and v2 both contain instances with either one or more than one correct answer choices. We make the following modifications in the original datasets to use them in our MCQA setup here, as we assume only one answer is correct for a given MCQA instance:

v1: We only consider instances which has one annotated correct answer. Each instance in CICERO v1 has five possible answer choices. Thus, the instances selected for our experiments in all the three sets (training, validation, and test split) has one correct answer and four incorrect answers.

v2: All instances in CICERO v2 has at-least two correct answers. We consider instances with at-least one incorrect answer and create the MCQA dataset as follows:

- If the original CICERO v2 instance has n correct answers, then we will create n MCQA instances from it, each having one of the correct answers and three incorrect answers.
- The three incorrect answers will be chosen from the incorrect answers of the original instance. We perform oversampling (some incorrect answers repeated) to create three incorrect answers if there are less than three incorrect answers in the original instance.

For example, an instance in CICERO v2 has answer choices: $\{c_1, c_2, i_1, i_2\}$. The correct answers are $\{c_1, c_2\}$ and the incorrect answers are $\{i_1, i_2\}$. We create two MCQA instances from the original instance – i) with answer choices $\{c_1, i_1, i_2, i_1\}$, and ii) with answer choices $\{c_2, i_1, i_2, i_2\}$.

[2] https://leaderboard.allenai.org/
[3] https://huggingface.co/datasets

Dataset	Task	Instance
ANLI	Intermediate Event Selection	**Event 1:** Jenny cleaned her house and went to work, leaving the window just a crack open. **Event 2:** When Jenny returned home she saw that her house was a mess!
		Choice 1: A thief broke into the house by pulling open the window. **Choice 2:** At work, she opened her window and the wind blew her papers everywhere.
CommonsenseQA	Answer Selection	**Question:** Where on a river can you hold a cup upright to catch water on a sunny day?
		Choice 1: Waterfall **Choice 2:** Bridge **Choice 3:** Valley **Choice 4:** Pebble **Choice 5:** Mountain
CommonsenseQA 2.0	Answer Selection	**Question:** The peak of a mountain almost always reaches above the the tree line.
		Choice 1: No **Choice 2:** Yes
QASC	Answer Selection	**Question:** Differential heating of air can be harnessed for what?
		Choice 1: electricity production **Choice 2:** running and lifting **Choice 3:** animal survival . . . **Choice 8:** reducing acid rain
SWAG	Ending Prediction	**Partial Event:** On stage, a woman takes a seat at the piano. She
		Ending 1: sits on a bench as her sister plays with the doll. **Ending 2:** smiles with someone as the music plays. **Ending 3:** is in the crowd, watching the dancers. **Ending 4:** nervously sets her fingers on the keys.
HellaSwag	Ending Prediction	**Partial Event:** A woman is outside with a bucket and a dog. The dog is running around trying to avoid a bath. She
		Ending 1: rinses the bucket off with soap and blow dry the dog's head. **Ending 2:** uses a hose to keep it from getting soapy. **Ending 3:** gets the dog wet, then it runs away again. **Ending 4:** gets into a bath tub with the dog.
Social IQA	Answer Selection	**Context:** Alex spilled the food she just prepared all over the floor and it made a huge mess. **Question:** What will Alex want to do next?
		Choice 1: taste the food **Choice 2:** mop up **Choice 3:** run around in the mess
Physical IQA	Solution Selection	**Goal:** To separate egg whites from the yolk using a water bottle, you should
		Solution 1: Squeeze the water bottle and press it against the yolk. Release, which creates suction and lifts the yolk. **Solution 2:** Place the water bottle and press it against the yolk. Keep pushing, which creates suction and lifts the yolk.
CosmosQA	Answer Selection	**Context:** : It's a very humbling experience when you need someone to dress you every morning, tie your shoes, and put your hair up. Every menial task takes an unprecedented amount of effort. It made me appreciate Dan even more. But anyway I shan't dwell on this (I'm not dying after all) and not let it detract from my lovely 5 days with my friends visiting from Jersey **Question:** What's a possible reason the writer needed someone to dress him every morning?
		Chocie 1: The writer doesn't like putting effort into these tasks. **Chocie 2:** The writer has a physical disability. **Chocie 3:** The writer is bad at doing his own hair. **Chocie 4:** None of the above choices.
CICERO v2	Answer Selection	**Dialogue:** A: Dad, why are you taping the windows? B: Honey, a typhoon is coming. A: Really? Wow, I don't have to go to school tomorrow. B: Jenny, come and help, we need to prepare more food. A: OK. Dad! I'm coming. **Target:** Jenny, come and help, we need to prepare more food. **Question:** What subsequent event happens or could happen following the target?
		Chocie 1: Jenny and her father stockpile food for the coming days. **Chocie 2:** Jenny and her father give away all their food. **Chocie 3:** Jenny and her father eat all the food in their refrigerator. **Chocie 4:** Jenny and her father eat all the food in their refrigerator.

Table 8: Illustration of the different datasets used in this work. The answers highlighted in green are the correct answers.