Adversarial-Social IQA:
An Adversarial Commonsense Benchmark Focusing on Association Biases.

by

Thumwanit Napat
タムワニット ナパット

A Master Thesis
修士論文

Submitted to
the Graduate School of the University of Tokyo
on January 19, 2022
in Partial Fulfillment of the Requirements
for the Degree of Master of Information Science and Technology
in Computer Science

Thesis Supervisor: Akiko Aizawa　相澤 彰子
Professor of Computer Science

ABSTRACT

Commonsense reasoning is one of the important abilities for natural language understanding requiring implications beyond textual pattern, for example, *lemon is sour*, *murder is immoral*. Social Interaction Question-Answering (SIQA) is a dataset that evaluates models on social interaction commonsense reasoning in the multiple-choice format. The leaderboard of SIQA shows that model performance is close to human performance with a larger model size and more training data. However, it is likely that the models do not understand commonsense and rely on spurious correlation. We focus on *association biases* where the commonsense reasoning models prefer prominent relationships between words or sentences such as *hospital: medicine* or *Alice loses the game: Alice is unskilled*, and if the model solely relies on these association biases, it is likely to fail when encountering a specific or unusual situation.

In this thesis, we introduce a multiple-choice Adversarial-SIQA (A-SIQA) evaluation dataset that is adversarial toward models exploiting association biases. To construct our A-SIQA, we developed the annotation system using the prevalent commonsense knowledge graph ATOMIC, a collection of if-then relationship events, and the models fine-tuned on SIQA. Firstly, we use ATOMIC as the data source for extracting the models ’ association biases. Then, the annotators use the extracted biases to attack the models which result in hard multiple-choice questions.

We show that the classification models trained on SIQA, including the state-of-the-art model, UnifiedQA, fail on our A-SIQA. We further fine-tune the models on our A-SIQA dataset to observe if the models can learn adversarial cases. It improves the accuracy when testing on A-SIQA yet drops on the original SIQA test set. Our analysis shows that the models adapt to the new association biases after being trained on A-SIQA which results in a performance drop on the original dataset SIQA. Finally, the results suggest that evaluating the commonsense reasoning models on our A-SIQA together with the original SIQA can bring a more robust commonsense reasoning benchmarking.

Acknowledgements

Thank you Prof. Akiko Aizawa for providing useful suggestions and feedback on the research ideas, experiment, and this whole thesis.

Thank you the lab members, especially in the machine-reading group, Sugawara-san, Xahn-san, and Mario-san, for a fruitful discussion and new ideas for the experiment.

Thank you to all the diligent workers who help me throughout the annotation process.

Contents

Chapter 1

Introduction

1.1 Commonsense Reasoning Benchmark and Problems

Commonsense is one of the important aspects of natural language understanding. Rather than relying on the syntax or structure of the sentence, commonsense requires us to perform implications from the common fact happening every day. For example, we know that if a person is *thirsty*, they would likely to find something to *drink*. Several benchmarks are used to evaluate the model ability to understand commonsense such as Choice of Plausible Alternative (COPA) [10], Winogrand [20] which are manually crafted by experts, and Commonsense Question Answering (CommonsenseQA) [34], Physical Interaction: Question Answering (PIQA) [3], and Social Interaction: Question Answering (SIQA) [29], which are annotated based on large corpus or knowledge bases, ConceptNet [30], ATOMIC [28], or WikiHow[1]. We show an example from SIQA in the Figure 1.1.

Since the upcoming of a deep neural model such as Long-Short Term Memory (LSTM) [12] or transformer [35], it shows superior performance on different tasks in natural language processing. However, [11] shows that the models exploit *annotation artifacts*. Annotation artifacts are cues that can be easily exploited by the model; for example, in reading comprehension, answer usually resides in the beginning of the paragraph [18], or in Natural Language Inference (NLI), the hypothesis including the word *not* usually be contradiction. There are several attempts to collect the dataset to avoid exploitation such as improvement in the collection process or post-filtering methods. In commonsense reasoning, SWAG [37] constructs a sentence-ending dataset using Adversarial Filtering (AF) to filter easy answers from the pool of model-generated candidates. HellaSwag [38] accounts for the biases caused by the generator in SWAG and uses a better generator with the new data source.

1.2 Model Association in Commonsense Reasoning

In this thesis, we use the term *association* to explain how we see, or how the model decide, one thing related to another thing in some relation. For example, suddenly given the context “sea”, we might imagine “swimming”. In the same way, models can have their association; neural models are well-known to capture lexical statistics from the training data, especially the recent pre-trained language models that are trained to predict the mask token. However, these associations can be used for the model to exploit in commonsense reasoning benchmark. *Word association*, such as *pan* → *hot*, is one of the addressed problems that the model can perform shortcut learning rather than understanding or reasoning. Winogrande

[1] `https://www.wikihow.com/`

Situation: Alex had a great trip at the park for a few hours.
Question: How would you describe Alex?
A1*: happy
A2: lazy
A3: tired and sore

Figure 1.1: An example from the commonsense reasoning benchmark is Social IQA. (*) indicates the correct choice. We underline the trigger phrase that the model can exploit to answer the question.

Winogrande
Situation: The lions ate the zebras because **they** are predators.
A1*: Lions
A2: Zebras

WinoVenti
Situation: Regina shivered when she picked up the **pan**
Question: The pan is ___?
A1: Hot
A2*: Cold

Figure 1.2: Examples from Winogrande and Winoventi. We underline the trigger word in each example. In Winogrande, the model can exploit the association from the word *predator* to the *lions*. In WinoVenti, they raise the exception to the generic association *pan* to *hot* by modifying the context.

[27] construct a co-reference resolution dataset that avoids the model exploiting on word association through human filtering and further filter out model-biases through their algorithm AFLITE. Winoventi [7] construct the sample that is *exception* to generic association, such as *pan* → *hot*. They extract the association pairs from the pre-trained language model and human-annotated the exception sample. We show both examples from the Winogrande and the Winoventi in the Figure 1.2

Nevertheless, most previous works focus mainly on word association which we believe that there is also association on higher levels such as the *phrase association*. We aim to improve the difficulty of the existing dataset SIQA by introducing our Adversarial SIQA (A-SIQA) that opposes the model preference. We collect the samples using Human-And-Model-in-the-Loop (HAML) to better observe the model preference and attack on the weak points. We create an annotation system that the annotator can observe the confidence score of the model on different candidates from ATOMIC annotations which represent the model preference directly. This can prevent the saturation of the dataset which is similar to the idea in ANLI [23], or DynaSent [24] which used the Dynabench template. In Chapter 3, we discuss in detail the annotation how we use SIQA pre-trained model and ATOMIC knowledge base to capture the model association and create hard samples.

In Figure 1.3, we show samples from our A-SIQA that the model was fooled. In the first example, we can see that the model strongly associated “Alex’s house” with the statement “needed to buy the home”. The additional context “inherited from his father” strongly implies that Alex did not buy the house. However, the model is less sensitive toward the context than its preference. In the second example, the model associated the word “killed” with “violent” strongly even though the semantic role here is different.

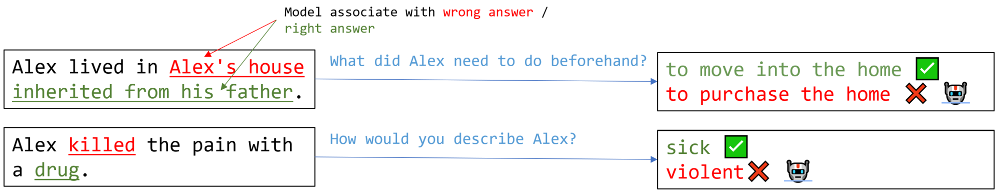

Model associate with wrong answer / right answer
Alex lived in Alex's house inherited from his father.
What did Alex need to do beforehand?
to move into the home ✓
to purchase the home ✗
Alex killed the pain with a drug.
How would you describe Alex?
sick ✓
violent ✗

Figure 1.3: Example from A-SIQA. The red and green highlight the words or phrase that are associated by the model to the wrong answer or correct answer.

We found that the models trained on SIQA fail on A-SIQA. They perform worse or near to random chance even the state-of-the-art model UnifiedQA [15]. This shows that current dataset is not enough to evaluate the model ability on commonsense reasoning. Also, this implies that the difficulty can be transferred to different types of model. We fine-tune the model on our dataset to observe if the model can adapt to A-SIQA. The result shows us that the model can perform better on our A-SIQA at the expense of dropping performance on the original SIQA which is in line with the finding by [14]. We discuss further in the Chapter 4 our dataset and the model performance in different settings.

We found that our A-SIQA has an opposite biases pattern to the SIQA. We perform ablation study, excluding situation, question, or both, using SIQA and find that the model accuracy is lower than chance. This implies that our A-SIQA can serve as a benchmark that strongly penalizes models exploiting the biases in the SIQA.

1.3 Analysis and Measuring Model Association

To understand the model association, we need to obtain the output from the model. In the case of pre-trained language models, we can query them using sentences with mask tokens [7, 33, 36]. To capture the association, the query should be *incomplete*, not detailed, in such a sense that we prefer the model to infer by itself. We find that the ATOMIC knowledge graph is a suitable source as it contains events that are simple sentences without further complex detail and a variety of annotations associated with the event.

On the other hand, humans behaviour can be similar to models; if given an incomplete statement, we can predict or provide the most likely statement that come up next. For example, given the statement “He is running” without any further detail, we would likely think “He is exercising” rather than “He is escaping away from ghost” as it is more likely to happen in our life. Such behavior can be considered as human association too. Taking this observation into consideration, we can analyze the consistency of the model association to the human association by asking the human to do a similar task to the model. This enables us to quantitatively evaluate how the model association matches with humans.

We describe in detail how we can measure the surrogate for the model association using the ATOMIC knowledge graph in Chapter 5. We analyze the associations of the different models that are fine-tuned on different datasets, both concerning the human association and different model associations. We use the result from the analysis to show that our A-SIQA exhibits opposite association from the original dataset SIQA and also opposite from human association. In general, we found that the model fine-tuned with portion of A-SIQA attain lower consistency to human association, even when mixing the SIQA with A-SIQA. This implies that, internally, the model become *paranoid* and the logits become less consistent with the human.

We further compare the association between models by taking the correlation of the output logits. We found that the training dataset strongly affects the model association, even more than the architecture of the model. By fine-tuning on SIQA mixing with A-SIQA, the model association diverts from the model trained on SIQA. This suggests us multiple round of HAML data annotation might able to tackle model from multiple association biases.

Chapter 2

Related Works

In this chapter, we brief about the researches that have been done in the commonsense reasoning field in Section 2.1 and Section 2.2. Next, we explain the general picture about the neural model shortcut learning and the methods used for preventing them in Section 2.3. Lastly, we provide a brief detail about the models that be used in the experiment in Section 2.5, focusing on transformer-based models.

2.1 Commonsense Knowledge Base

To enable access to commonsense knowledge, some researchers compile large knowledge bases in the form of graphs and relationships. ConceptNet [32, 31] is a large knowledge graph that collected the graph connecting the relationship of an object to different objects or phrases. For example, the object *car* has a relationship *capableOf* to the phrase *move quickly.* There are over 21 million edges and over 8 million nodes which included other languages than English according to their claims. ATOMIC knowledge base collects a lot of phrases and their relationship with different phrases in a similar manner to ConceptNet but focusing on social interaction. For example, given the event phrase *PersonX books a ticket*, it has a relationship *intention (xIntent)* to another event phrase *to travel somewhere.* There are over 600k of the phrases and relationship triplets. Some words in the given event can be abstracted out, for example, in Figure2.1, the right example blanks out the object and lets the annotators assume anything. One use of these commonsense knowledge bases is to train the generative models that can produce the follow-up statement after feeding the query, in the case of ATOMIC, we feed an event and the relationship and then ask the model to produce the related event. COMET [4] is a transformer-based generative commonsense model trained both on ConceptNet and ATOMIC to output the given query. This is

PersonX adopts a kitten
Relationship: Intention
Annotation:

- to help an animal
- to have a companion

PersonX gets ___ from a friend
Relationship: Want
Annotation:

- to talk
- to respond to their friend

Figure 2.1: Examples from the ATOMIC knowledge base, A collection of if-then relationships on different events. The person subjects are all abstracted to “PersonX”. In the example on the right, the object is abstracted to be the blank “___”

an attempt to build a neural model for auto-completion of the knowledge base. ATOMIC 2020 [13] combines the original ATOMIC with the completion of the COMET and releases 1.3 million triplets of the if-then relationship knowledge graph.

2.2 Commonsense Benchmark

As our main focus, Social Interactive Question Answering (SIQA) is the dataset that collected questions that are related to social interaction in different situations. The questions were collected using ATOMIC as the data source for crafting. The type of the questions are based on the relationship in ATOMIC; for example, for the relationship type *Attribute*, the question ask *How would you describe personName?*. To collect hard questions that require question and context understanding, they employ the method *question-switching* such that the negative answer is crafted from the different question in the same context. For example, we have the context "Alex spilled food all over the floor and it made a huge mess." and the question asks "What Alex want to do next?". The possible correct answer might be "to mop the floor". Next, we switch the question by asking "What did Alex need to do before this?" instead which we could answer something such as "have a slippery hand". In the end, we can use the answer from different question to be a negative choice, in this case, "have a slippery hand" is the negative answer because it is originally asked what Alex want to do next. We show an example in Figure 1.1.

There are several other commonsense benchmarks such as Choice of Plausible Alternative (COPA) asking about the *effect* and *cause* of the premise and hypothesis, Commonsense Question Answering (Commonsense QA) that focusing on physical commonsense such as *river has water*, *snow is cold*, or Physical IQA (PIQA) which also focuses on physical interaction but in the format of sentence ending.

2.3 Shortcut Learning and Adversarial Data Collection

Shortcut learning is a phenomenon that the model does not learn what the human originally "intent" the model to learn but solves the problem by using unintentional cues included in the benchmark. [9] gives a general discussion on shortcut learning in different fields. Furthermore, the models that exploit shortcut learning fall short when an encounter with out-of-distribution data as they did not learn to solve the problem systematically. In natural language processing, many datasets are crafted by human annotation, and that makes the dataset polluted with some biases by humans. On the other hand, the dataset that is crafted involving model generation can also suffer biases by the model. The authors of the HellaSWAG [38] shows that the existing dataset SWAG contains those biases. [11] conduct an experiment on the Natural Language Inference (NLI) task where the premise and hypothesis are given and the model needs to classify correctly if the hypothesis is true, false, or neutral w.r.t. the hypothesis. Interestingly that the model can perform far better than chance even when the model is not provided the premises. This input ablation benchmark is adopted widely in researches that focus on crafting datasets; usually, used after data collection. Several works attempt to tackle the biases in the dataset by including an algorithm to avoid or eliminate them. For example, SWAG [37], a sentence-ending multiple-choice task, introduce the Adversarial Filter (AF) to filter out the "easy" wrong answers that were generated by the models. Briefly, AF is an algorithm that iteratively trains a model on a random training portion and replaces the easy negative choices with the harder ones, determined by the logit of the model. Winogrande [27] is a coreference resolution dataset that extends from the Winogrand [20] but at a larger scale. They elicit human annotators to avoid the *association biases* that can be exploited by the model. Furthermore, they introduce a lightweight version of AF named AFLITE to eliminate model-specific biases.

On the other hand, the idea of "human-and-model-in-the-loop" (HAML) comes up as it is possible to craft samples that are hard to the model by asking humans to interact with the model and attack its weak points. The ANLI [23] introduces this idea on the NLI dataset. They provide the promise to the annotators and ask them to craft the hypothesis that the model answers wrongly. They show that the further rounds of data collection are conducted, the harder the dataset becomes or even the model becomes harder to fool. The Dynabench [16] further extends to different NLP tasks such as sentiment analysis or reading comprehension. The Dynasent [24] is one of the results from Dynabench. Beat-the-AI [1] perform an experiment on extractive machine reading question-answering with different types of models both LSTM and transformers. In general, the stronger the model, the harder it would be if considering the average performance on all models. Interestingly, the model same type as the one used for the HAML suffer the most when testing on the adversarial dataset regardless of being the strongest model. However, Even HAML enable human to directly craft hard samples yet can become a harder benchmark for the model, it could induce the model to fail on normal dataset when using them for fine-tuning due to its out-of-distribution from the normal dataset.

2.4 Commonsense Association, and Exception

Commonsense is well-known as an implicit relationship between entity and entity such as *lemon* is *yellow*. The model such as transformer-based language can capture these connections by the input data that might appear *lemon* and *yellow* near to each other, we call it *model association*. However, it is hard to ask the model to explain how the *lemon* is *yellow*. On the other hand, if we pose an exceptional situation such as *the lemon is rotten*, we hypothesize that the model fail on it. WinoVenti [7] is the work that attempts to proof this concept by extracting the association from the pre-trained language model BERT and craft the samples that counter the association. They name them *generic* and *exception* commonsense. Defeasible-NLI [26] introduce another idea to capture commonsense with exception. Defeasible-NLI were collected from SNLI [5], Social Chem 101 [8], and ATOMIC. In our work, we just focus on only ATOMIC portion. As we described above, each ATOMIC event comes with several annotations depended on the relationship. They ask the workers to craft the updater that makes an annotation become more likely, namely *strengthener*, and the updater that makes an annotation become less likely, namely *weakener*. We show an example for each of both WinoVenti and defeasible-NLI in the Figure 2.2.

2.5 Models

We give a brief detail about the model that be used in our experiment. The models are all transformer-based [35] pre-trained language models. Concretely, we use BERT [6], RoBERTa [21], ALBERT [19], and T5 [25] in our experiment. All models are pre-trained on generative task such as *mask prediction*, similar to filling in the blank task, or learning to predict the next sentence. Despite

WinoVenti
Regina screamed when she picked up the pan. The pan is **hot**.
Regina shivered when she picked up the pan. The pan is **hot**.

defeasible-NLI (ATOMIC)
Premise: PersonX has a pool party
Hypothesis: Because PersonX wanted to hangout with friends
Strengthener (↑): It was PersonX ’ s birthday
Weakener (↓): PersonX was having a family reunion

Figure 2.2: The upper half shows an example of the WinoVenti on both generic and exception. The bottom half shows an example of the premise, hypothesis, and their corresponding strengthener and weakener.

being pre-trained mainly for text generation, they can be fine-tuned easily to different tasks by changing the prediction neural network head of the model and achieving the state-of-the-art on most of them. UnifiedQA [15] is a transformer-based model fine-tuned on different question answering tasks simultaneously by converting all of them into the generative task with the same format. They fine-tune it on each task specifically. By using T5-11B (11 billion parameters T5) as the based model, UnifiedQA beat the state-of-the-art on almost every task, including the Social IQA as our main focused dataset in this thesis.

Chapter 3

Dataset and Annotation Process

3.1 Definition

In this work, we focus on the multiple choices dataset given a situation and a question. Based on the SIQA format, each sample is given a situation $s \in \mathcal{S}$, question $q \in \mathcal{Q}$, and the answers $a_1, a_2, \ldots, a_k \in \mathcal{A}$. The task is to choose the correct answer from the given answers. Here, we employ a model $f : \mathcal{S} \times \mathcal{Q} \times \mathcal{A} \to \mathbb{R}$ output the logits for each given (s, q, a_i), we evaluate if the model can output the highest logit on the correct answer a_j, $\arg\max_i f(s, q, a_i) = j$. On the other hand, if the model is trained to be generative such as UnifiedQA or UNICORN, we use the exact match as the metric to measure the accuracy.

We had discussed ATOMIC in Chapter 2 and we define it formally here. The ATOMIC knowledge base is comprised of event s (e.g. PersonX presses the button), given relationship r (e.g. PersonX intention), and the annotation a (e.g. to turn on the light). The knowledge base represents the if-then relationship of the given event $s \in \mathcal{S}$ to the result in annotations $\mathcal{A}$ with respect to the relationship $r \in \mathcal{R}$. So we define the ATOMIC triplet as (s, r, a).

3.2 Data Collection

In our work, we conduct the adversarial data collection using human-and-model-in-the-loop to collect hard data for the model. There are two important steps for data collection.

1. **Identify the model association biases**: Firstly, as our work focus on attacking association biases, we need to identify the model association. It is well-known that deep learning is a black-box model which is unlikely to be interpretable. Therefore, we show how we can use the scoring of the model and the ATOMIC knowledge base to identify the model association in Section 3.2.1.

2. **Attack the model association biases**: We integrate the scoring model into our system and ask the annotators to craft the adversarial samples that can fool the model. We explain different annotation tasks in Section 3.2.2.

3.2.1 Scoring Model and Model Association

Because our task is to craft an adversarial version of SIQA, we need to train a model that can solve SIQA. SIQA is a multiple-choice, exactly three-choices,

Situation: PersonX presses the button. (→ Alex presses the button.)

Relationship: Intention (→ Why did Alex do this?)
Annotation: to turn on the light

Relationship: Needed (→ What did Alex need to do?)
Annotation: reach for the button

Figure 3.1: We show an example from the ATOMIC knowledge base and how we convert them into the natural language form before feeding them to the model. We can convert the situation $s(\rightarrow s')$ and a relationship to questions $r(\rightarrow q)$.

and we train the model to choose the most correct answer. We train a regression model $f : f(s, q, a) \rightarrow \mathbb{R}$ that receives the input as the triplet of situation, question and answer (s, q, a) and output the logit score as a real number. For each sample, we obtain three logits for three choices, $f(s, q, a_1)$, $f(s, q, a_2)$, and $f(s, q, a_3)$, and compute the loss using softmax. We train the model using the Adam optimizer [17] at the learning rate 10^{-5} with early stopping if the model evaluation accuracy drop. In total, we train five RoBERTa-large and take the average of the logits from all models as suggested by [16] that ensemble model is more robust when used for human-and-model-in-the-loop (HAML) annotation. In fact, it is not restrict to only regression model that can solve the problem; generative model such as UnifiedQA [15] or UNICORN [22] are based on T5-11B model and achieve the state-of-the-art result on Social-IQA and also other datasets in natural language understanding. However, generative model could be much harder to probe the behavior; thus, it is less suitable to our HAML annotation. For convenience, we mention the model used in HAML annotation as **HAML model**.

In our process, we provide the ATOMIC event and its associated annotations to the annotators. To assist the annotators in understanding the model, we employ our scoring model to score each of the annotations so annotators can have a hint of how the model associates the event to different annotations. To obtain the score, we need to feed the ATOMIC triplet (s, r, a) to the model and obtain the output as the score. We first need to convert the triplet into a formulation that is similar to Social-IQA. In ATOMIC, all subjects are symbolized to "PersonX/Y/Z" which can be easily substituted by people's names. We substitute them with "Alex/Robert/Charlie" as these names are unisex such that gender biases can be avoided. If the event includes a blank "___", we replace them by using RoBERTa. We replace the blank with the "<mask>" token and pick the noun with the highest predicted probability. Note that this could generate something being non-sense; however, the annotators can fix these in the later stage. The relationship can be converted into the designated questions. The full conversion of the ATOMIC event and relationship is noted in Table 3.1 and a conversion example is shown in Figure 3.1. Once we convert the event and relationship $(s, r, a) \rightarrow (s', q, a)$, we feed the triplet to the model and output its logit. We show an example of the ATOMIC triplet scoring step-by-step in the Figure 3.2.

3.2.2 Annotation Process

Our work is based on the existing task, Social-IQA, therefore, we annotate the data into the same format. Each sample be comprised of (1) Situation, (2) Question, (3) Correct answer, (4) First wrong answer, and (5) Second wrong answer.

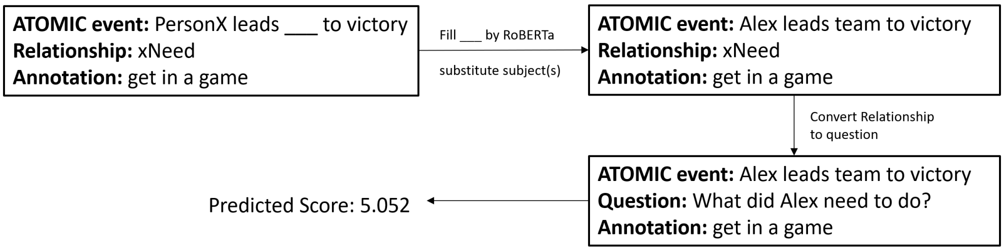

ATOMIC event: PersonX leads ___ to victory
Relationship: xNeed
Annotation: get in a game
Fill ___ by RoBERTa
substitute subject(s)
ATOMIC event: Alex leads team to victory
Relationship: xNeed
Annotation: get in a game
Convert Relationship to question
ATOMIC event: Alex leads team to victory
Question: What did Alex need to do?
Annotation: get in a game
Predicted Score: 5.052

Figure 3.2: Step-by-step processing to compute the score on an ATOMIC triplet.

	ATOMIC	**Conversion**
Subject	PersonX	Alex
	PersonY	Robin
	PersonZ	Charlie
Relationship	xAttr (Attribute)	How could you describe Alex?
	xEffect (Effect)	What happen to Alex?
	xIntent (Intention)	Why did Alex do this?
	xNeed (Needed)	What did Alex need to do?
	xReact (Reaction)	What Alex feel afterward?
	xWant (Want)	What Alex want to do?
	oEffect (other Effect)	What happen to (Robin/other)?
	oReact (other Reaction)	What (Robin/other) feel afterward?
	oWant (other Want)	What (Robin/other) want to do next?

Table 3.1: Conversion of subject and relationships in the ATOMIC event to different names and questions.

Different from the Social-IQA, the process to collect the sample is performed by a single annotator in an end-to-end fashion. It is also for providing more flexibility to fool the model. To simplify the task, we split the task into **Two-choice annotations** where the correct answer and the first wrong answer are annotated, and **Third-choice annotation** where the second wrong answer is added. We show the full annotation interfaces in the Appendix A.

Two-choice Annotation In this annotation process, we ask the annotators to craft a sample (s, q, a, a') where a is the more likely and a' is less likely w.r.t the situation and question (s, q). We aim the create adversarial samples that are hard for a model that exploits artifacts, and model association. The annotation process is shown as the following step.

1. The annotators are provided with a random event and its ATOMIC annotations within their relationship. We convert the event into the situation and compute the score for each annotation using the HAML model.

2. The annotators consider how to attack the HAML model from the computed scores and make a modification in the situation.

3. The annotators choose the question and decides the correct and wrong answers. Usually, annotators can choose the answers directly from the ATOMIC annotations; however, they can choose to write or modify them by themselves too.

ATOMIC event: Alex leads ___ to victory
Situation: Coach Alex leads his team to victory of the league.
Question type: Needed
Question: What did Alex need to do?
Correct Answer: give them support
Wrong answer: get in the game
Second wrong answer: celebrate [→ What Alex want to do next?]

Figure 3.3: An example of A-SIQA with three choices.

4. The annotators query the HAML model to score the sample once it is finished. The model is successfully fooled if the model logit for the correct answer is lower than the wrong answer, in the other word, $f(s, q, a) < f(s, q, a')$.

5. If the model is not successfully fooled, one can choose to modify further the samples.

The illustration of the overall process is shown in Figure 3.4. The most frequently used strategy for this annotation is flipping the model score which is shown right in the illustration. Initially, for the given event *PersonX loses forty pound*, the model assigns the score -3.66 for *goes to doctor* and 1.499 for *buys new clothes.* The annotators can choose to make the lower assigned score answer become the **correct answer** and the higher assigned score answer become the **wrong answer** by modifying the event without letting the model recognizes. Here, we added a small change in the context as *Alex loses forty pounds in a day*.

Third-choice Annotation For the third choice annotation, we similarly employ question-switching as it had been done in SIQA. As we collect the original ATOMIC event from the two-choice annotation, we use the annotations from the different relationships as the candidate. The difference from the process in SIQA is that we know which choice is preferred by the model; thus the annotators can choose the answer that tricks the model. In this sense, we could describe this process as *adversarial question switching.* Finally, we collect the situation, question, one correct answer and two wrong answers (s, q, a, a', a''). We show an example in Figure 3.3 where there are completely three choices. The first positive and the negative choices were collected from our HAML annotation. The third choice is later collected from the question-switching method. In the example, the third wrong choice is *celebrate*; however, it was the answer for the wrong question as *What Alex want to do next?* instead.

3.3 Annotation Detail

It is known that the models are sometimes being sensitive to even a small change which makes the model adversarial dataset considered a hard task. We hypothesize that crafting the samples on this task is easy; however, crafting a hard sample for the model is more difficult. Therefore, we provide the annotation of the annotator as a batch of 5 questions and require them to successfully fool the model 3 out of 5 questions to be considered as a successful annotation.

We found that this task is fairly complex so that we decide to hire native speakers or English proficient workers that can be easily contacted to provide

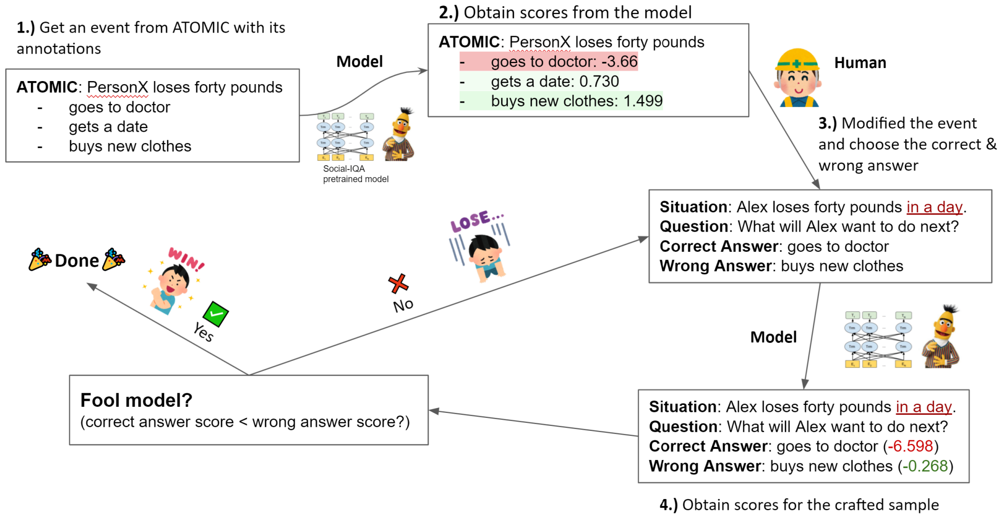

1.) Get an event from ATOMIC with its annotations
ATOMIC: PersonX loses forty pounds
- goes to doctor
- gets a date
- buys new clothes
Model
Social-IQA pretrained model
2.) Obtain scores from the model
ATOMIC: PersonX loses forty pounds
- goes to doctor: -3.66
- gets a date: 0.730
- buys new clothes: 1.499
Human
3.) Modified the event and choose the correct & wrong answer
Situation: Alex loses forty pounds in a day.
Question: What will Alex want to do next?
Correct Answer: goes to doctor
Wrong Answer: buys new clothes
Model
Situation: Alex loses forty pounds in a day.
Question: What will Alex want to do next?
Correct Answer: goes to doctor (-6.598)
Wrong Answer: buys new clothes (-0.268)
4.) Obtain scores for the crafted sample
Fool model?
(correct answer score < wrong answer score?)
Yes
WIN!
Done
No
LOSE...

Figure 3.4: The process of the two-choice annotation using model-in-the-loop.

feedback promptly. In total, we hire 13 workers, included the author, to annotate the data.

For verification, we decide to use the Amazon Mechanical Turk (AMT) platform. We qualify the annotators in this step to have been performing the task on AMT more than 1000 tasks with >97% of approval rate. By these qualifications, we can assure that the annotators are native and of high quality. For further workers' quality control, we conduct the qualification test to check if the workers can choose the correct answers to the designated questions. There are 20 questions in total and the worker should correctly answer at least 18 questions. We keep tracking the annotators' performance by including performance tracking questions from both Social-IQA and our collected data. We block the worker from performing our task if the worker accuracy on the performance tracking set drops lower than <80%.

The payment for the questions annotation is hour-based and we expected 20 hours for 150 annotated questions. The wage is based on the research assistant scale. For the verification task, we paid the workers an average of $0.15 for each question. On average, the worker can perform 100 questions in an hour so that the hour pay is around $15 per hour. For each of the questions, three workers perform verification. We take the most popular vote for each question and filter those questions that the voted answer is not matched with the original answer.

3.4 Dataset Summary

We collected 1216 samples that are correctly voted for our A-SIQA. The dataset was split into training and validation set for 916 samples and 300 samples respectively. We chose the samples that all workers agreed on the correct answer and also successfully fooled the model as the validation set to make sure that it is challenging yet doable for humans.

The ratio of the collected question types is shown in the Table 3.2. We find that most of the questions are in the *Attribute* category which occupies 26% of the dataset. This category requires implication such that the answer is not explicitly mentioned in the situation context. Some categories such as *Reaction* and *Want* have a lower percentage in our A-SIQA. This implies that the categories that the

Question Type	**SIQA**	**A-SIQA**
Attribute	15%	26%
Effect	11%	15%
Intention	12%	20%
Needed	12%	11%
Reaction	21%	10%
Want	29%	18%

Table 3.2: Percentage of each question type for SIQA and A-SIQA.

Properties	**SIQA**	**A-SIQA**
#Unique tokens	3699	3983
#Unique tokens (situation)	2639	2822
Density of unique tokens	0.126	0.147
Density of unique tokens (situation)	0.160	0.181
situation mean length	13.50	12.84
correct answer mean length	3.56	3.27
wrong answer mean length	3.56	3.05

Table 3.3: Basic properties of randomly sampled SIQA and A-SIQA.

annotators tend to fool model are those categories that have lower percentages in the original dataset SIQA.

We also summarize the token counts for both SIQA and A-SIQA. Due to the difference in data size, we also report the statistic for SIQA when it was sampled at the same amount as A-SIQA. In previous works, the model is tricked by using a longer length of the questions; for example, in ANLI, longer premises are used so it is harder for the model to answer or, in HellaSWAG, longer sentences are used. However, in A-SIQA, we observe that the average length of the situation is not longer than the average length in SIQA and also applied for different elements, correct answers, and wrong answers. This implies that our A-SIQA doesn't attack the model by simply adding longer context but rather focusing on making harder context instead. A-SIQA has more unique tokens than SIQA with the same number of samples despite having a shorter average length of the situations and the answers. Even more, interestingly, there are several tokens in A-SIQA that have not appeared in SIQA. New tokens are *recent* words such as *crypto*, *bitcoin*, *tinder*, or *COVID*, less common keywords such as *mafia*, *leukemia* which shows the use of broader tokens or words; thus, leading to higher diversity of the dataset. However, we find that the average length of the correct answers and the wrong answers are slightly different and possibly be exploited by the model. We show the overall statistics of the basic properties in Table 3.3.

Chapter 4

Experiment

4.1 Models and Training Process

For all models, we train them by using the exactly same process as described in Section 3.2.1. The model output logit for each choice and penalize on softmax loss. For each model, there are slightly different ways to encode the context s, question q, and choice a by concatenating with the token [SEP]. Thus, the input is concatenate as "{context} [SEP] {question} [SEP] {answer}" for each {answer} from the three choices. However, UnifiedQA input the sentence differently so we need to construct the prompt as "{question} \\n (A) {answerA} (B) {answerB} (C) {answerC} \\n {context}"[1] and let it generate the correct answer. We use the fine-tuned weight of T5-11B UnifiedQA on SIQA in our experiment.

4.2 Zero-shot from SIQA to A-SIQA

4.2.1 Main Results

To observe the adaptability of the models trained on SIQA to our datasets, we evaluate A-SIQA without any further fine-tuning. We evaluate the original SIQA and our A-SIQA both 2-way setting and 3-way setting. The results are shown in Table 4.1. In general, we can observe that models performing better on Social IQA shows better performance on our A-SIQA except for the RoBERTa model which was used for model-in-the-loop. Note that this result is in line with the result from [1] showing that even the model was retrained in a different seed, it suffers the most due to model-specific adversaries. Most models obtain lower than chance accuracy, but ALBERT exceeds the chance accuracy while also obtaining the highest accuracy. Lastly, UnifiedQA is the most robust out of all models that can perform great on SIQA and obtain higher than chance accuracy on our A-SIQA. Interestingly, the third choice does not affect the UnifiedQA which means that the model is tolerant toward question-switching adversarial more than the other models.

4.2.2 Error Analysis

We check if the difficulty of the questions correlates with the predicted logits by the model used in annotations or not. We simplify the experiment by focusing on 2-way A-SIQA (s, q, a, a'). We define $d_{\text{HAML}}(x) = f_{\text{HAML}}(s, q, a') - f_{\text{HAML}}(s, q, a)$ as the difficulty of each sample for the model RoBERTa where f_{HAML} is the RoBERTa model used in model-in-the-loop. To illustrate that, based on the crafted sample shown in Figure 3.4, the difficulty for the sample is $d_{\text{HAML}}(x) =$

[1] According to the official released code: `https://github.com/allenai/unifiedqa`

Model - Evaluation Set	**A-SIQA**		**SIQA**
	2-way	3-way	
BERT	0.340	0.250	0.659
RoBERTa*	0.217	0.167	0.775
ALBERT	0.527	0.477	0.798
UnifiedQA	**0.583**	**0.583**	**0.812**
Human Performance	-	0.837	0.841

Table 4.1: Performance of zero-shot for each SIQA pre-trained models. RoBERTa is indicated with * to show that it was used for model-in-the-loop which we retrained it with different initialization seed.

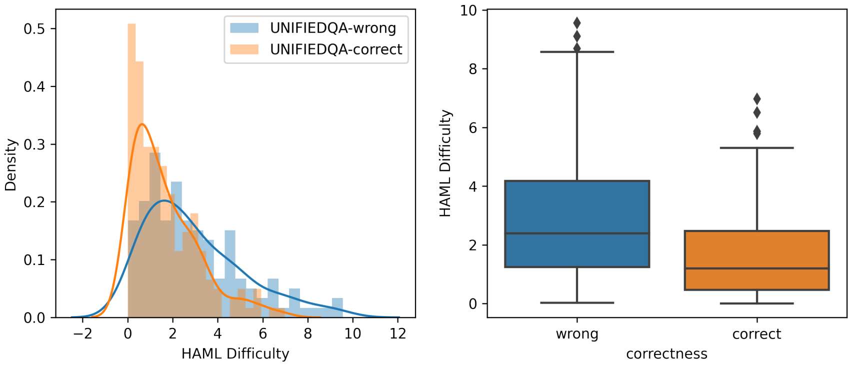


Figure 4.1: Comparison of the HAML model difficulty vs UnifiedQA correctness on zero-shot evaluation.

$-0.268 - (-6.598) = -6.330$. We examine the difficulty on the set of questions where UnifiedQA can answer correctly and those it cannot. We show the distribution plot and the box plot of the difficulties with respect to the prediction correctness of UnifiedQA in Figure 4.1. The distribution plot shows that the questions that are answered incorrectly have higher distribution on higher RoBERTa difficulty compared to those that are answered correctly. It can be observed clearly from the box plot that the wrong questions are distributed toward higher RoBERTa difficulty. This shows that the difficulty from one model can imply the difficulty on the different models even the state-of-the-art model. Therefore, model-in-the-loop data collection with a strong enough model should be sufficient to challenge the state-of-the-art model.

Furthermore, we visualize the difficulty of different models toward the HAML model. As expected, RoBERTa fine-tuned on SIQA would show the highest correlation because the based model is the same as the HAML model. Both BERT and ALBERT have a positive correlation with HAML difficulty while BERT has less correlation. The reason might be that BERT is the weakest model compared to the others. In general, the positive correlations, especially to stronger model ALBERT, suggest that the difficulty of the adversarial samples crafted on different HAML model annotation can be transferred to different models.

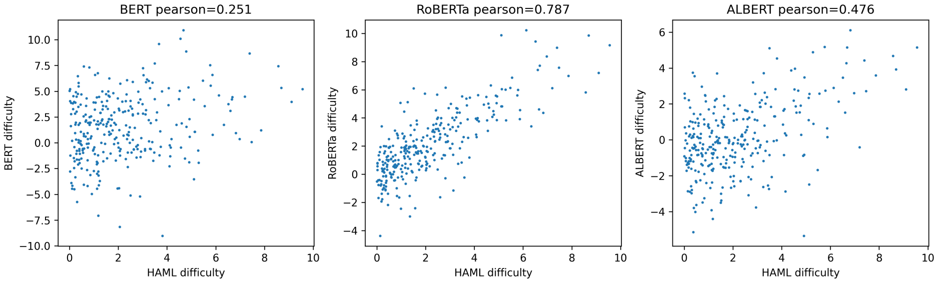


Figure 4.2: Comparison of the HAML model difficulty vs BERT, RoBERTa, ALBERT difficulty on zero-shot.

4.3 Fine-tuning on A-SIQA

4.3.1 Setup

We fine-tune the models further on A-SIQA to observe if the models can adapt to the adversarial samples. From the 1216 samples we collected, we fine-tune the model on our A-SIQA by splitting it into training and development sets (as shown in Section 3.4). Our A-SIQA has a small number of samples so it is necessary to train the model in a way that the model can learn A-SIQA while not being overpowered by the SIQA. Therefore, we try to up-sample the number of A-SIQA to be equivalent to SIQA and included them in the training. Note that this method is used for an imbalanced dataset where negative labels are far less than positive labels and cause low precision [2]. We list the models' type names and how the models are fine-tuned in the following list.

- f_{SIQA}: The model is fine-tuned on only SIQA. In case we compare to different models started from different initialization seeds, we name them f_{SIQA_1} and f_{SIQA_2}
- $f_{\text{A}-\text{SIQA}}$: The model is fine-tuned on only A-SIQA.
- $f_{\text{SIQA}_{\text{mini}}}$: We randomly select 916 samples from the SIQA for fine-tuning the model.
- $f_{\text{SIQA}+\text{A}-\text{SIQA}}$: We mix the SIQA and A-SIQA together and feed them to the model during fine-tuning.
- $f_{\text{SIQA}+\uparrow\text{A}-\text{SIQA}}$: We mix the SIQA and A-SIQA together but the A-SIQA is upsampled to be the same amount as SIQA.

These models' definitions be used for explaining the analysis in Section 5.

4.3.2 Main Results

We experiment on the same models' list as in Section 4.2. However, we cannot fine-tune the UnifiedQA due to its large number of parameters (11 billion parameters), we decide to leave out the result of it. For each model, we trained three different models starting from different random seeds and take the model that obtains the best accuracy on the validation set. We report the result in Table 4.2 and Table 4.3.

We observe the fair comparison between SIQA and A-SIQA in Table 4.2. For all transformers, we observe that the accuracy on A-SIQA for $f_{\text{SIQA}_{\text{mini}}}$ are

	Model Type	**A-SIQA**		**SIQA**
		2-way	3-way	
BERT	$f_{\text{SIQA}_{\text{mini}}}$	0.363	0.260	0.457
	$f_{\text{A}-\text{SIQA}}$	0.707	0.617	0.320
RoBERTa	$f_{\text{SIQA}_{\text{mini}}}$	0.293	0.197	0.596
	$f_{\text{A}-\text{SIQA}}$	**0.760**	**0.643**	0.265
ALBERT	$f_{\text{SIQA}_{\text{mini}}}$	0.430	0.327	**0.682**
	$f_{\text{A}-\text{SIQA}}$	0.700	0.613	0.391

Table 4.2: Accuracy on A-SIQA and SIQA for different transformer models and different model types. Note that all models are LARGE versions and cased.

	Model Type	**A-SIQA**		**SIQA**
		2-way	3-way	
BERT	$f_{\text{SIQA}+\text{A}-\text{SIQA}}$	0.503	0.400	0.651
	$f_{\text{SIQA}+\uparrow\text{A}-\text{SIQA}}$	0.617	0.537	0.643
RoBERTa	$f_{\text{SIQA}+\text{A}-\text{SIQA}}$	0.447	0.367	0.787
	$f_{\text{SIQA}+\uparrow\text{A}-\text{SIQA}}$	0.617	0.550	0.736
ALBERT	$f_{\text{SIQA}+\text{A}-\text{SIQA}}$	0.703	0.647	**0.792**
	$f_{\text{SIQA}+\uparrow\text{A}-\text{SIQA}}$	**0.737**	**0.680**	0.786

Table 4.3: Accuracy on A-SIQA and SIQA for different transformer models and different model types. Note that all models are LARGE versions and cased.

lower than chance probability which means the model failed to tackle with A-SIQA by training only on SIQA. On the other hand, the model fine-tuned on A-SIQA alone completely failed to tackle SIQA, only ALBERT shows slightly better than chance accuracy. This implies that the samples have associations that are opposed to usual sentences which we further analyze in the Section 5. Interestingly, RoBERTa which is the based model for the HAML annotation obtain the best accuracy on A-SIQA yet the accuracy on SIQA drastically drop from the random chance, compared to BERT and ALBERT.

We observe the models trained on the full training set of SIQA mixed with A-SIQA in Table 4.3. In any setting, the models perform better than fine-tuning with only SIQA and reach above random chance significantly, as the results are shown in Table 4.1. The fine-tuning without up-sampling can also improve the accuracy on A-SIQA while the accuracy on the SIQA is insignificantly changed, even slightly better for RoBERTa. The improvement of accuracy on A-SIQA is small for BERT and RoBERTa which is overpowered by SIQA. oppositely, there is a significant gain for ALBERT. The fine-tuning with up-sampling further enhance the model to learn to tackle with A-SIQA in exchange for accuracy drop on SIQA. There is a small accuracy drop on SIQA for BERT and ALBERT but much larger for RoBERTa. Possibly, using RoBERTa as the HAML model results in stronger SIQA accuracy drop due to A-SIQA.

4.3.3 Error Analysis

We select the example from the A-SIQA which the model mispredicted with the largest margin between the correct and wrong answer logits. We examine the

Question Type	Incorrect
Attribute	30%
Effect	26%
Intention	34%
Needed	29%
Reaction	18%
Want	**45%**

Table 4.4: Percentage of incorrectly predicted questions for each question type in A-SIQA of the model ALBERT-xxlarge on $f_{\mathrm{SIQA}+\uparrow\mathrm{A-SIQA}}$

strongest baseline ALBERT-xxlarge. Even the model $f_{\mathrm{SIQA}+\uparrow\mathrm{A-SIQA}}$ can perform better on the A-SIQA as the accuracy reported in Table 4.3, we still find errors such as in this question "Alex made more money from his work than he did in the last year. What happen to Alex?" and the model choose "become richer" instead of "is taxed higher". This example shows that the model strongly associates the money with rich; however, to be taxed more happen while "become richer" might not always if the amount of money is not enough or Alex wasted more money. For different types of questions in A-SIQA, we find that the model makes mistakes in the *Want* category the most with 44% mistakes while other types such as *Intention* and *Attribution* follow up with 34% and 30% of questions being incorrectly answered. Despite *Attribute* and *Want* having the most percentage of the training dataset, these types of questions are the most incorrectly answered, while question type *Reaction* has lower incorrectly answered questions. This fact also reflects on the annotation where most of the annotated types are *Attribute*, *Intention*, *Want*; on the other hand, the least annotated type is *Reaction.*

In addition, we show the examples that the model correctly predicts after being fine-tuned on A-SIQA and also those incorrectly predicts in Table 4.5.

4.4 Annotation Artifact in A-SIQA

In this experiment, we perform fine-tuning with ablation on the A-SIQA samples to observe the annotation artifact. When collecting the data using the HAML paradigm, the annotators may grasp some strategies and attack the model on its weak point. However, that could lead to the *annotation artifacts* or unintentional cues that allow the model to exploit undesirable features. We fine-tune in three different ablations, only situation (-Q), only question (-S), and without both (-S,Q).

With ablation, we find that the model can still perform better than chance probability; thus, there are biases included in our A-SIQA. We think that more rounds of HAML annotation with dynamic model updates can alleviate this problem which leaves it to future work.

Interestingly, we fine-tune the ablation model on SIQA and evaluate on our A-SIQA. We can observe that the accuracy on A-SIQA is much lower than SIQA or even near to chance probability. Also, it is true for the opposite. This shows us that both SIQA and A-SIQA have biases but they are different from each other. The accuracy of the ablation models are listed in Table 4.6.

A-SIQA Wrong → Correct
Alex developed a software for use in the company. **Question:** What did Alex need to do? **A1:** to make a plan ✓ **A2:** to gather materials ✗
Alex rode the big roller coaster after her friends took a lot of time to convince her **Question:** How could you describe Alex? **A1:** scared ✓ **A2:** adventurous ✗
A-SIQA Correct → Wrong
Alex ran from his room after he heard his mom called for dinner. **Question:** Why did Alex do this? **A1:** to eat ✓ **A2:** to get away ✗
Alex was forced out of middle school to study at the university. **Question:** How could you describe Alex? **A1:** genius ✓ **A2:** embarrassed ✗

Table 4.5: Examples of samples in A-SIQA that the prediction correctness of the model ALBERT-xxlarge on $f_{\text{SIQA}+\uparrow\text{A}-\text{SIQA}}$ changed. The upper half shows the samples that the model predicts correctly after fine-tuning with up-sampling A-SIQA. The lower half shows the samples that the model predicts incorrectly after fine-tuning.

Premise (p): PersonX has a pool party
Hypothesis (h): Because PersonX wanted to hangout with friends
Relationship (r): Intention
Strengthener (↑) (s): It was PersonX ’ s birthday
Weakener (↓) (w): PersonX was having a family reunion

Defeasible-NLI Conversion
Strengthen Premise (s'^{+}): Alex has a pool party. It was Alex’s birthday
Weaken Premise (s'^{-}): Alex has a pool party. Alex was having a family reunion
Question (q): Why did Alex do this?
Hypothesis (h'): Because Alex wanted to hangout with friends

Figure 4.3: The converted sample of defeasible-NLI into SIQA-like format.

4.5 Out-of-distribution Evaluation

We further observe the robustness of the model on out-of-distribution tasks. We choose the datasets that can be formulated similarly to SIQA so we can instantly use the model without any further training. In this experiment, we employ two datasets introduced in 2.4, defeasible-NLI and Winoventi.

4.5.1 Defeasible-NLI

In this experiment, we focus on the ATOMIC portion of the defeasible-NLI. each sample is comprised of

- **Premise** p: The ATOMIC event

	Model Type	A-SIQA		SIQA
		2-way	3-way	
BERT	f_{SIQA}	0.340	0.250	0.659
	−Q	0.397	0.280	0.506
	−S	0.283	0.177	0.494
	−S,Q	0.347	0.230	0.453
	$f_{\text{A-SIQA}}$	0.707	0.617	0.320
	−Q	0.627	0.487	0.336
	−S	0.540	0.397	0.304
	−S,Q	0.467	0.310	0.319
RoBERTa	f_{SIQA}	0.217	0.167	0.775
	−Q	0.373	0.270	0.591
	−S	0.207	0.147	0.623
	−S,Q	0.250	0.163	0.474
	$f_{\text{A-SIQA}}$	0.76	0.643	0.265
	−Q	0.757	0.620	0.364
	−S	0.693	0.517	0.279
	−S,Q	0.643	0.490	0.269
ALBERT	f_{SIQA}	0.527	0.477	0.798
	−Q	0.493	0.410	0.650
	−S	0.240	0.183	0.644
	−S,Q	0.360	0.230	0.503
	$f_{\text{A-SIQA}}$	0.700	0.613	0.391
	−Q	0.633	0.516	0.365
	−S	0.623	0.520	0.310
	−S,Q	0.517	0.350	0.334

Table 4.6: Accuracy on A-SIQA and SIQA for different transformer models and different model types. Note that all models are LARGE versions and cased.

- **Hypothesis** h: The ATOMIC annotation
- **Relationship** r: The ATOMIC relationship between premise and hypothesis
- **Updater**
  - **Strengthener** s: The updater of the premise that makes the hypothesis more likely
  - **Weakener** w: The updater of the premise that makes the hypothesis less likely

We can construct the sample into a SIQA format task. We combine the premise with updater by simply appending the updater behind the premise as the strengthen premise $s^+ = [p; s]$, and the weaken premise $s^- = [p; w]$. We convert the samples into the same format as SIQA by replacing the abstracted person name and the relationship with the designated name and questions as shown in Table 3.1. Finally, by using naming conversion, we construct a sample $(s^+, s^-, r, h) \rightarrow (s'^+, s'^-, q, h')$. We test the model if it can consistently output the higher logit for the strengthen premise; in another word, if $f(s'^+, q, h') > f(s'^-, q, h')$ for a model f. We report the consistency in the accuracy metric.

	Model Type	Defeasible-NLI consistency
BERT	f_{SIQA}	0.734
	$f_{\text{SIQA+A-SIQA}}$	**0.751**
	$f_{\text{SIQA+}\uparrow\text{A-SIQA}}$	0.716
RoBERTa	f_{SIQA}	0.816
	$f_{\text{SIQA+A-SIQA}}$	**0.831**
	$f_{\text{SIQA+}\uparrow\text{A-SIQA}}$	0.797
ALBERT	f_{SIQA}	0.834
	$f_{\text{SIQA+A-SIQA}}$	**0.840**
	$f_{\text{SIQA+}\uparrow\text{A-SIQA}}$	0.830

Table 4.7: Accuracy on A-SIQA and SIQA for different transformer models and different model types. Note that all models are LARGE versions and cased.

WinoVenti
Regina screamed when she picked up the pan. The pan is **(hot/cold)**.

WinoVenti Conversion
Situation: Regina screamed when she picked up the pan.
Question: How could you describe the pan?
Answer: Hot / Cold

Figure 4.4: The converted sample of WinoVenti into SIQA-like format.

We report the consistency in Table 4.7 for each model and three types, f_{SIQA}, $f_{\text{SIQA+A-SIQA}}$, and $f_{\text{SIQA+}\uparrow\text{A-SIQA}}$. For each model family, we find that the model $f_{\text{SIQA+A-SIQA}}$ type shows more consistency than the f_{SIQA}. However, $f_{\text{SIQA+}\uparrow\text{A-SIQA}}$ performs least consistent. It is possible that by up-sampling, it imposes too strongly uneven distribution and make the fine-tuned model overfitting on A-SIQA.

4.5.2 WinoVenti

WinoVenti is the dataset with a similar format as WinoGrand. As the questions are mainly focused on asking the property of an object, this task could be similar to SIQA but asking only on the category *Attribute*. Thankfully that the format for all questions is uniform, we can extract the subject of the question and formulate a sample into SIQA format. We show the example of the conversion in Figure 4.4.

We report the accuracy in Table 4.8. Fine-tuning on A-SIQA can bring some improvement on this task for all model families except on BERT where $f_{\text{SIQA+}\uparrow\text{A-SIQA}}$ performs worse than f_{SIQA}. The model type $f_{\text{SIQA+A-SIQA}}$ performs that best in every model family. Interestingly, the accuracy shown in our result, which is zero-shot from SIQA and A-SIQA, is better than the accuracy obtained by fine-tuning directly on the WinoVenti as reported in [7].

	Model Type	**WinoVenti**
BERT	f_{SIQA}	0.656
	$f_{\text{SIQA+A-SIQA}}$	**0.661**
	$f_{\text{SIQA+}\uparrow\text{A-SIQA}}$	0.652
RoBERTa	f_{SIQA}	0.735
	$f_{\text{SIQA+A-SIQA}}$	**0.755**
	$f_{\text{SIQA+}\uparrow\text{A-SIQA}}$	0.750
ALBERT	f_{SIQA}	0.758
	$f_{\text{SIQA+A-SIQA}}$	**0.776**
	$f_{\text{SIQA+}\uparrow\text{A-SIQA}}$	0.761

Table 4.8: Accuracy on A-SIQA and SIQA for different transformer models and different model types. Note that all models are LARGE versions and cased.

Chapter 5

Analysis on Model Association and Effect of Dataset

To analyze the model association, we extract the surrogate of the model association by computing the ATOMIC annotations score using scoring from different models. For a situation, relationship, and annotation (s, r, a) triplet, we retrieve the score using model logit $f(s[\rightarrow s'], r[\rightarrow q], a)$ which the process of conversion is exactly identical to the process introduced in Section 3.2.1. By measuring all logits from all ATOMIC triplets, we obtain the representation of the model associations in the form of numbers which we use in the following quantitative analysis of model association. In total, there are 700k triplets from the ATOMIC knowledge graph.

We analyze the models in two settings. Each setting includes an analysis of different models set. Note that the model type symbols are defined in Section 4.3.

1. $f_{\text{SIQA}_{\text{mini}}}$ and $f_{\text{A}-\text{SIQA}}$ for fair comparison as all models are trained on same amount of data. We examine how each dataset affects the model association so we can understand what kind of associations are included in the SIQA and A-SIQA.

2. f_{SIQA}, $f_{\text{SIQA+A}-\text{SIQA}}$, and $f_{\text{SIQA+}\uparrow\text{A}-\text{SIQA}}$ to observe how the fine-tuning can make change in the model association.

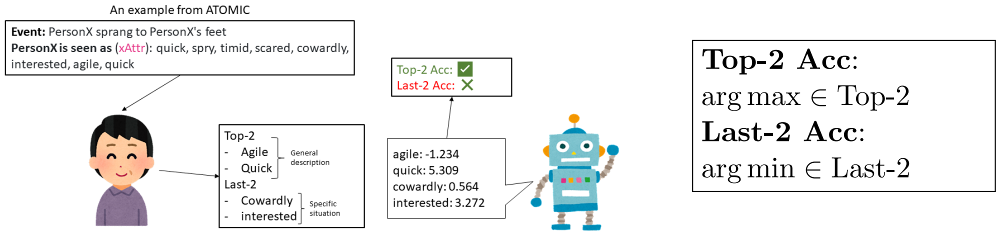


Figure 5.1: The process of ranking the ATOMIC annotations by human and evaluation based on model ATOMIC scoring.

5.1 Model Association vs Human Association

We want to observe to what extent the model scoring on ATOMIC annotations is in line with human opinion. If the model scoring is in line with humans, the model has a similar association to humans. To collect the human opinion, we choose the ATOMIC events that have at least 6 annotations so the ranking could be more flexible. We choose **top-2** annotations that are most likely to happen and **last-2** annotations that are most unlikely to happen. The likeliness was subjectively chosen by generality (is the annotation can generally explain the event?) and commonness (how usual is the event to happen). For example, given a situation where "Alex sprang to his feet", we can generally say that he is fast or agile. However, it is also possible that Alex is a coward or interested in something but that could happen in a more specific situation; thus, less plausible than the other choices. The example and the data collection process are shown in Figure 5.1.

We compute the consistency of the model association to the human preference through the following metrics

- **Top-2 Acc**: Out of four predicted logits, whether the highest logit is in the **top-2** or not.
- **Last-2 Acc**: Out of four predicted logits, whether the lowest logit is in the **last-2** or not.

As we assume that humans can rank the answers reasonably, we hope that the model choice or scoring have high consistency with the human ranking. However, these metrics are not absolute metrics to determine if the model is good or bad, evaluations further on the benchmark dataset SIQA or A-SIQA are a must. We also report the human score by asking the tester to rank four answers and using the same metrics as above and use the result as the human baseline.

We compare the model types $f_{\mathrm{SIQA}_{\mathrm{mini}}}$ and $f_{\mathrm{A-SIQA}_{\mathrm{mini}}}$ which are trained on the same amount of data but from the different dataset. According to the results shown in Table 5.1, the association of $f_{\mathrm{SIQA}_{\mathrm{mini}}}$ is more consistent with the human association. On the other hand, the association of $f_{\mathrm{A-SIQA}_{\mathrm{mini}}}$ strays away from the human association shows that the model learns to be opposite to generic association. This is the evidence to show that A-SIQA is a collection of samples that are non-generic and counterfactual to the human association.

We further explore the consistency on the models f_{SIQA}, $f_{\mathrm{SIQA+A-SIQA}}$, and $f_{\mathrm{SIQA+\uparrow A-SIQA}}$, the models that are fine-tuned on SIQA. Naively mixing the A-SIQA with SIQA as training data, $f_{\mathrm{SIQA+A-SIQA}}$, causes consistency drop compared to f_{SIQA} on ALBERT but does not show a clear drop on RoBERTa and even increase on BERT case. However, there is clear consistency drop for all model when comparing f_{SIQA} to $f_{\mathrm{SIQA+\uparrow A-SIQA}}$. These results were expected as $f_{\mathrm{SIQA+\uparrow A-SIQA}}$ models were more exposed to the adversarial samples. Considering the results of RoBERTa in Table 4.3 and Table 5.2, the large drop of SIQA accuracy on $f_{\mathrm{SIQA+\uparrow A-SIQA}}$ is correlated with the large drop of the consistency. We hypothesize that fine-tuning on adversarial samples, it could make the model to *paranoid* instead of learning to solve adversarial samples reasonably. In another word, the model learn to be affixed to the adversarial association.

5.2 Model Association vs Model Association

As we use the ATOMIC annotations score as the surrogate for the model preference, we can compare the association of two different models by observing

	Model Type	Top-2	Last-2
BERT	$f_{\text{SIQA}_{\text{mini}}}$	0.61	0.71
	$f_{\text{A}-\text{SIQA}}$	0.51	0.54
RoBERTa	$f_{\text{SIQA}_{\text{mini}}}$	0.67	0.65
	$f_{\text{A}-\text{SIQA}}$	0.35	0.36
ALBERT	$f_{\text{SIQA}_{\text{mini}}}$	0.61	0.66
	$f_{\text{A}-\text{SIQA}}$	0.51	0.50
Human		0.83	0.90

Table 5.1: Consistency of the model preference to human preference on different training data. A higher score implies higher consistency to human preference. Note that all models are LARGE versions and cased.

	Model Type	Top-2	Last-2
BERT	f_{SIQA}	0.66	0.71
	$f_{\text{SIQA}+\text{A}-\text{SIQA}}$	**0.68**	**0.75**
	$f_{\text{SIQA}+\uparrow\text{A}-\text{SIQA}}$	0.62	0.65
RoBERTa	f_{SIQA}	**0.78**	0.77
	$f_{\text{SIQA}+\text{A}-\text{SIQA}}$	0.77	**0.78**
	$f_{\text{SIQA}+\uparrow\text{A}-\text{SIQA}}$	0.67	0.71
ALBERT	f_{SIQA}	**0.84**	**0.82**
	$f_{\text{SIQA}+\text{A}-\text{SIQA}}$	0.71	0.77
	$f_{\text{SIQA}+\uparrow\text{A}-\text{SIQA}}$	0.70	0.76
Human		0.83	0.90

Table 5.2: Consistency of the model preference to human preference on different training data. A higher score implies higher consistency to human preference. Note that all models are LARGE versions and cased.

the similarity of the scores of both models. If we compute all ATOMIC triplets scores, we can construct the *association vector* **A** of a model f from the ATOMIC knowledge graph as

$$[\mathbf{A}(f)]_i = f(s_i[\to s'_i], r_i[\to q_i], a_i) \text{ for } (s_i, r_i, a_i) \in \text{ATOMIC triplets}$$

where s'_i and q_i are the event and question converted. Next, we compute the Pearson and Spearman correlation between the association vectors of two models to measure the similarity. ATOMIC events are simple and not too much detail, therefore, it is appropriate to use the ATOMIC triplet and extract the model association and represent it as a vector for each model. By doing so, we can capture the dynamic change of the association when models are trained on different distributions of datasets. We show an illustration to compare the similarity of the associations in Figure 5.2.

We fine-tune the model on each dataset and measure their association. Then we measure the correlation between any two associations. In this case, we can compare two different models, $f_{\text{SIQA}_{\text{mini}1}}$ and $f_{\text{SIQA}_{\text{mini}2}}$, that are fine-tuned on a portion of SIQA as the baseline correlation. Next, we measure the correlation of two models, $f_{\text{SIQA}_{\text{mini}1}}$ and $f_{\text{A}-\text{SIQA}}$, that is fine-tuned on the same amount of

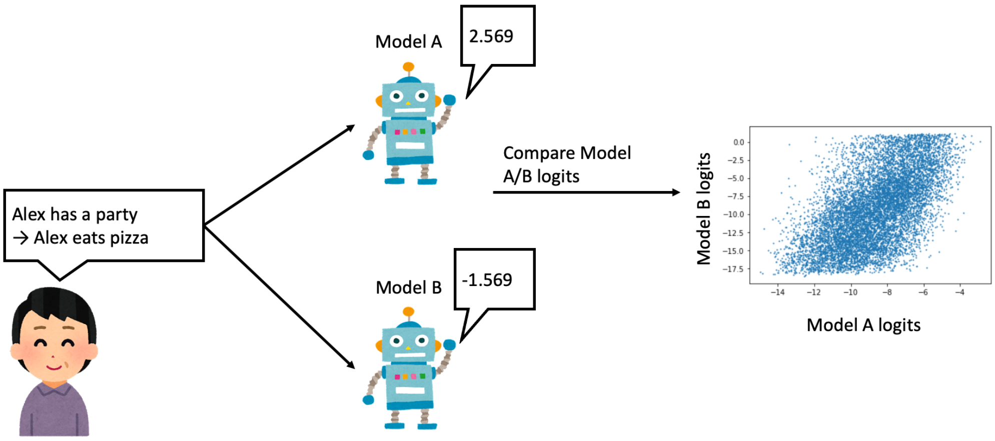


Figure 5.2: Method to compare model associations using the model logits and ATOMIC events.

data but one on SIQA and one on A-SIQA. By doing so, we can fairly compare the correlation to observe the difference. As shown in Table 5.3, the correlation of the $f_{\mathrm{SIQA}_{\mathrm{mini}\,1}}$ and $f_{\mathrm{A-SIQA}}$ is far lower than the correlation of the $f_{\mathrm{SIQA}_{\mathrm{mini}\,1}}$ and $f_{\mathrm{SIQA}_{\mathrm{mini}\,2}}$ both Pearson and Spearman. This result is in line with the previous result discussed in Section 5.1

We further observe the effect of A-SIQA, while fine-tuning mixing of both SIQA and A-SIQA into the training. We also report the result in the Table 5.3. For simplicity, we focus on the best model that can output the logits, ALBERT. We can see that two ALBERT models fine-tuned on SIQA have a high correlation. This shows that the association might depend on the architecture of the model or how it was pre-trained; thus, the same family of models has high similarity. We can observe that the correlation between two RoBERTa models fine-tuned on SIQA is slightly higher than the ALBERT to RoBERTa. The correlation of ALBERT and BERT fine-tuned on SIQA is the lowest and might be due to the BERT being far weaker than the other models. Lastly, we compare two ALBERT models but one is fine-tuned on SIQA and another is fine-tuned on SIQA+↑A-SIQA. We can observe the drop of correlation from two models fine-tuned on SIQA, even lower than the correlation between ALBERT and RoBERTa. This shows that the model association is dynamically changed due to the training data, though there is no drastic change in SIQA accuracy, from 0.798 to 0.786. We visualize the plot of the ATOMIC triplets score in Figure 5.3.

The dynamic change of the association shows that it is possible to attack the model by performing HAML annotation again. We think that multiple rounds of HAML annotation can help to create more diverse adversarial dataset as the association of the model could change after fine-tuning. We leave this part as future work.

Model Type	Pearson	Spearman
ALBERT$_{\text{SIQA}_{\text{mini1}}}$ / ALBERT$_{\text{SIQA}_{\text{mini2}}}$	0.437	0.428
ALBERT$_{\text{SIQA}_{\text{mini}}}$ / ALBERT$_{\text{A}-\text{SIQA}}$	0.199	0.209
ALBERT$_{\text{SIQA}_1}$ / ALBERT$_{\text{SIQA}_2}$	0.771	0.759
ALBERT$_{\text{SIQA}}$ / RoBERTa$_{\text{SIQA}}$	0.745	0.737
ALBERT$_{\text{SIQA}}$ / BERT$_{\text{SIQA}}$	0.633	0.623
ALBERT$_{\text{SIQA}}$ / ALBERT$_{\text{SIQA+A}-\text{SIQA}}$	0.723	0.711
ALBERT$_{\text{SIQA}}$ / ALBERT$_{\text{SIQA+}\uparrow\text{A}-\text{SIQA}}$	0.687	0.695
RoBERTa$_{\text{SIQA}_1}$ / RoBERTa$_{\text{SIQA}_2}$	0.755	0.760

Table 5.3: Correlation of the models trained on different settings. The higher correlation implies that the learned association is similar.

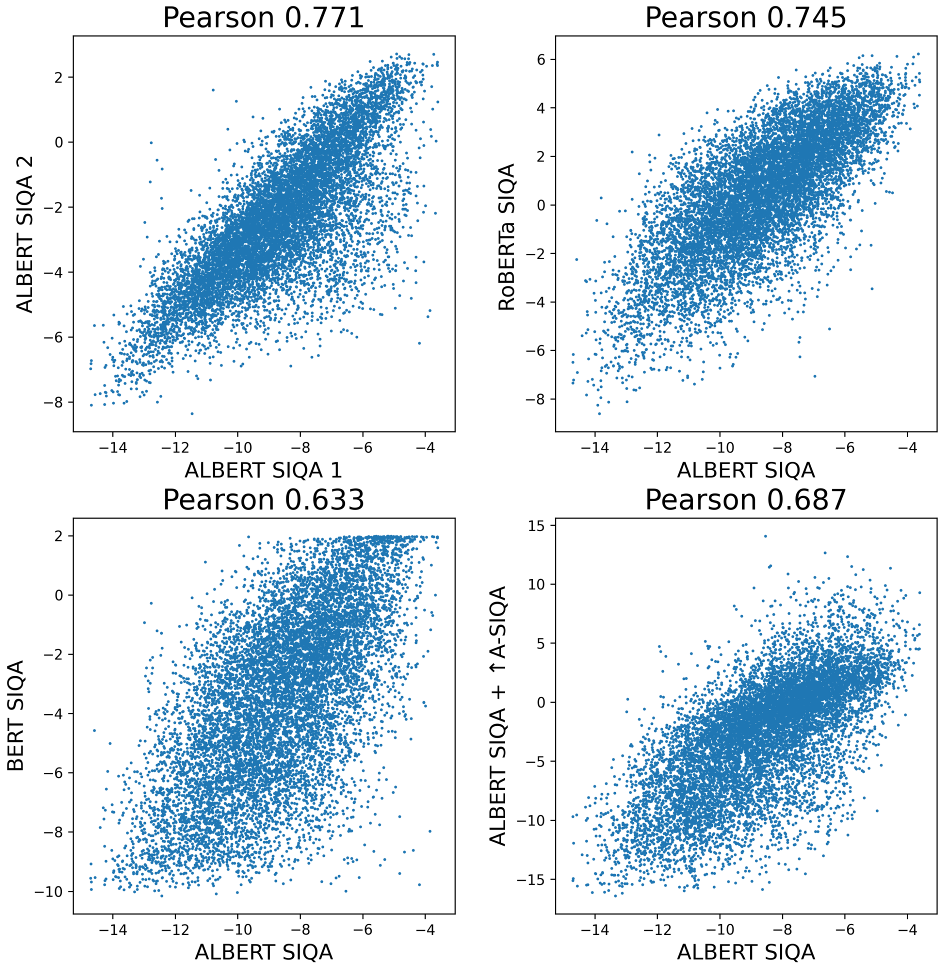


Figure 5.3: Plot of the output logits between different models. (Top Left) Plot of the ALBERT model fine-tuned on SIQA with different seeds. (Top Right) Plot of the ALBERT model fine-tuned on SIQA and RoBERTa model fine-tuned on SIQA. (Bottom Left) Plot of the ALBERT model fine-tuned on SIQA and BERT fine-tuned on SIQA. (Bottom Right) Plot of the ALBERT model fine-tuned on SIQA and ALBERT fine-tuned on SIQA + ↑A-SIQA. 10000 ATOMIC triplets are randomly selected for visualization.

Chapter 6

Conclusion

In this work, we develop a framework to assist the annotators to craft adversarial samples on social interaction commonsense reasoning tasks using human-and-model-in-the-loop. Different from the previous work, we propose a framework to collect multiple-choice questions by using the model logits as the hint. We focus our adversarial examples to attack the model association built up after being fine-tuned on a dataset, in this case, SIQA. Using our annotation framework, we craft the adversarial dataset A-SIQA to stress the model evaluation in a counterfactual context.

We show that our dataset has a higher variety of word token than the original dataset but also preserves the average length of the situation and the answer like the original dataset. Different from work such as SWAG or ANLI that focuses on attacking the model by using a longer length of the sentence, we aim to focus on making the context less generic; thus, harder for the model to tackle with. However, there are biases included in our A-SIQA which are shown through ablation fine-tuning. We show that the biases in SIQA and A-SIQA are different as the ablation model fine-tuned on a dataset is not transferred to the counterpart.

We show in our experiment that the models that are fined-tune on SIQA fail on our A-SIQA, in zero-shot evaluation. Moreover, the state-of-the-art model such as UnifiedQA fail on A-SIQA with the accuracy near to chance on the 2-way setting. We analyze the difficulty by taking the difference of the output logits of the correct answer and the wrong answer produced by the model we used in the annotation loop. We found that the questions that are wrongly answered by UnifiedQA have higher difficulty on the annotation model. Furthermore, there is a significant correlation between the difficulty of the model used in the annotation to another model. This implies that the difficulty can be transferred to an even stronger model or a different model.

We examine the ability to learn adversarial samples of the models by fine-tuning on our A-SIQA and performing evaluation right after. We find that the model was able to learn to solve the adversarial samples with an expense that SIQA accuracy decreases. This effect is stronger on RoBERTa which is used in the annotation. We evaluate the models on defeasible-NLI and WinoVenti, which included commonsense with the exception, of an out-of-distribution task. We find that the model fine-tuned on the mixture of SIQA and A-SIQA performs better than the model that is fine-tuned on SIQA alone. This illustrates the robustness of the model after fine-tuning on adversarial samples. Furthermore, we want to emphasizes that SIQA accuracy alone cannot imply robustness of model commonsense reasoning ability.

We analyze the model association of the model by extracting directly from the ATOMIC triplets. We found that fine-tuning the model with A-SIQA caused the model association to diverge from the human association. This implies that the model thinks *more paranoid* after being trained with A-SIQA. It is undesirable because it means that the model starts to think differently than human. We compare the associations of models trained on different subset of data. As a result, we find that there is difference between the association of the model trained solely on SIQA and the model trained with some portion as A-SIQA. Even models in different family (RoBERTa, ALBERT) trained on SIQA have more similar association pattern. Thus, the training data affect the association of the model largely, even more than the architecture of the models. The dynamical change of the model association gives us a hint that the model adapts a new association with a new distribution of samples. We suggest that multiple rounds of human-and-model-in-the-loop could increase the diversity of the adversarial samples as the model association keeps changing and this can be part of future work.

Finally, we believe that A-SIQA can complement with current SIQA benchmark for model commonsense evaluation by broadening the distribution of the benchmark.

References

[1] Max Bartolo, Alastair Roberts, Johannes Welbl, Sebastian Riedel, and Pontus Stenetorp. Beat the AI: Investigating adversarial human annotation for reading comprehension. *Transactions of the Association for Computational Linguistics*, 8:662–678, 2020.

[2] Gustavo E. A. P. A. Batista, Ronaldo C. Prati, and Maria Carolina Monard. A study of the behavior of several methods for balancing machine learning training data. *SIGKDD Explor. Newsl.*, 6(1):20–29, jun 2004.

[3] Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. Piqa: Reasoning about physical commonsense in natural language. In *Thirty-Fourth AAAI Conference on Artificial Intelligence*, 2020.

[4] Antoine Bosselut, Hannah Rashkin, Maarten Sap, Chaitanya Malaviya, Asli Celikyilmaz, and Yejin Choi. COMET: Commonsense transformers for automatic knowledge graph construction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4762–4779, Florence, Italy, July 2019. Association for Computational Linguistics.

[5] Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal, September 2015. Association for Computational Linguistics.

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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, June 2019. Association for Computational Linguistics.

[7] Nam Do and Ellie Pavlick. Are rotten apples edible? challenging commonsense inference ability with exceptions. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2061–2073, Online, August 2021. Association for Computational Linguistics.

[8] Maxwell Forbes, Jena D. Hwang, Vered Shwartz, Maarten Sap, and Yejin Choi. Social chemistry 101: Learning to reason about social and moral norms. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 653–670, Online, November 2020. Association for Computational Linguistics.

[9] Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A Wichmann. Shortcut learn-

ing in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673, 2020.

[10] Andrew Gordon, Zornitsa Kozareva, and Melissa Roemmele. SemEval-2012 task 7: Choice of plausible alternatives: An evaluation of commonsense causal reasoning. In **SEM 2012: The First Joint Conference on Lexical and Computational Semantics – Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, pages 394–398, Montréal, Canada, 7-8 June 2012. Association for Computational Linguistics.

[11] Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel Bowman, and Noah A. Smith. Annotation artifacts in natural language inference data. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 107–112, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.

[12] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Comput.*, 9(8):1735–1780, November 1997.

[13] Jena D. Hwang, Chandra Bhagavatula, Ronan Le Bras, Jeff Da, Keisuke Sakaguchi, Antoine Bosselut, and Yejin Choi. Comet-atomic 2020: On symbolic and neural commonsense knowledge graphs. In *AAAI*, 2021.

[14] Divyansh Kaushik, Douwe Kiela, Zachary C. Lipton, and Wen-tau Yih. On the efficacy of adversarial data collection for question answering: Results from a large-scale randomized study. 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 6618–6633, Online, August 2021. Association for Computational Linguistics.

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

[16] Douwe Kiela, Max Bartolo, Yixin Nie, Divyansh Kaushik, Atticus Geiger, Zhengxuan Wu, Bertie Vidgen, Grusha Prasad, Amanpreet Singh, Pratik Ringshia, Zhiyi Ma, Tristan Thrush, Sebastian Riedel, Zeerak Waseem, Pontus Stenetorp, Robin Jia, Mohit Bansal, Christopher Potts, and Adina Williams. Dynabench: Rethinking benchmarking in NLP. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4110–4124, Online, June 2021. Association for Computational Linguistics.

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

[18] Miyoung Ko, Jinhyuk Lee, Hyunjae Kim, Gangwoo Kim, and Jaewoo Kang. Look at the first sentence: Position bias in question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1109–1121, Online, November 2020. Association for Computational Linguistics.

[19] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. Albert: A lite bert for self-supervised learning of language representations. In *ICLR*. OpenReview.net, 2020.

[20] Hector J. Levesque, Ernest Davis, and Leora Morgenstern. The winograd schema challenge. In *Proceedings of the Thirteenth International Conference on Principles of Knowledge Representation and Reasoning*, KR'12, page 552–561. AAAI Press, 2012.

[21] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

[22] Nicholas Lourie, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. Unicorn on rainbow: A universal commonsense reasoning model on a new multitask benchmark. *AAAI*, 2021.

[23] Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. Adversarial NLI: A new benchmark for natural language understanding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4885–4901, Online, July 2020. Association for Computational Linguistics.

[24] Christopher Potts, Zhengxuan Wu, Atticus Geiger, and Douwe Kiela. DynaSent: A dynamic benchmark for sentiment analysis. 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 2388–2404, Online, August 2021. Association for Computational Linguistics.

[25] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020.

[26] Rachel Rudinger, Vered Shwartz, Jena D. Hwang, Chandra Bhagavatula, Maxwell Forbes, Ronan Le Bras, Noah A. Smith, and Yejin Choi. Thinking like a skeptic: Defeasible inference in natural language. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4661–4675, Online, November 2020. Association for Computational Linguistics.

[27] Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. Winogrande: An adversarial winograd schema challenge at scale. *Commun. ACM*, 64(9):99–106, August 2021.

[28] Maarten Sap, Ronan Le Bras, Emily Allaway, Chandra Bhagavatula, Nicholas Lourie, Hannah Rashkin, Brendan Roof, Noah A. Smith, and Yejin Choi. ATOMIC: an atlas of machine commonsense for if-then reasoning. In

The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019, pages 3027–3035. AAAI Press, 2019.

[29] Maarten Sap, Hannah Rashkin, Derek Chen, Ronan Le Bras, and Yejin Choi. 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, Hong Kong, China, November 2019. Association for Computational Linguistics.

[30] Robyn Speer, Joshua Chin, and Catherine Havasi. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, AAAI'17, page 4444–4451. AAAI Press, 2017.

[31] Robyn Speer, Joshua Chin, and Catherine Havasi. Conceptnet 5.5: An open multilingual graph of general knowledge, 2017.

[32] Robyn Speer and Catherine Havasi. Representing general relational knowledge in ConceptNet 5. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 3679–3686, Istanbul, Turkey, May 2012. European Language Resources Association (ELRA).

[33] Alon Talmor, Yanai Elazar, Yoav Goldberg, and Jonathan Berant. oLMpics-on what language model pre-training captures. *Transactions of the Association for Computational Linguistics*, 8:743–758, 2020.

[34] Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 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, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[35] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[36] Nathaniel Weir, Adam Poliak, and Benjamin Van Durme. On the existence of tacit assumptions in contextualized language models. *ArXiv*, abs/2004.04877, 2020.

[37] Rowan Zellers, Yonatan Bisk, Roy Schwartz, and Yejin Choi. 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, October-November 2018. Association for Computational Linguistics.

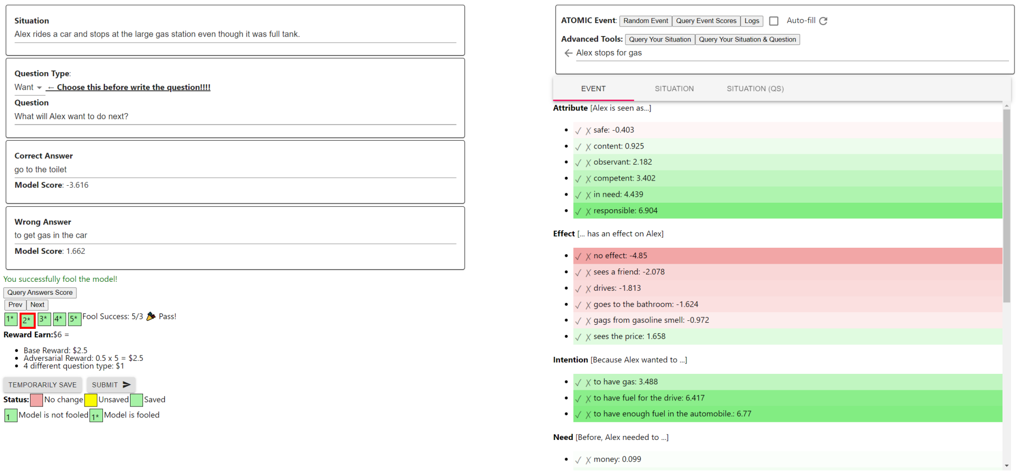
[38] Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. HellaSwag: Can a machine really finish your sentence? In *Proceedings of*

the 57th Annual Meeting of the Association for Computational Linguistics, pages 4791–4800, Florence, Italy, July 2019. Association for Computational Linguistics.

Appendix A

Annotation Interfaces

Overall, we have two interfaces for annotation, two-choice annotation and third-choice annotation. We show the two-choice annotation interface in A.1 and the third-choice annotation interface in A.2.



Situation
Alex rides a car and stops at the large gas station even though it was full tank.

Question Type:
Want ← **Choose this before write the question!!!!**
Question
What will Alex want to do next?

Correct Answer
go to the toilet
Model Score: -3.616

Wrong Answer
to get gas in the car
Model Score: 1.662

You successfully fool the model!

1* 2* 3* 4* 5* Fool Success: 5/3 Pass!

Reward Earn:$6 =

- Base Reward: $2.5
- Adversarial Reward: 0.5 x 5 = $2.5
- 4 different question type: $1

Status: 🟥 No change 🟨 Unsaved 🟩 Saved

1 Model is not fooled 1* Model is fooled

ATOMIC Event: □

Advanced Tools:

← Alex stops for gas

Attribute [Alex is seen as...]

- ✓ ✗ safe: -0.403
- ✓ ✗ content: 0.925
- ✓ ✗ observant: 2.182
- ✓ ✗ competent: 3.402
- ✓ ✗ in need: 4.439
- ✓ ✗ responsible: 6.904

Effect [... has an effect on Alex]

- ✓ ✗ no effect: -4.85
- ✓ ✗ sees a friend: -2.078
- ✓ ✗ drives: -1.813
- ✓ ✗ goes to the bathroom: -1.624
- ✓ ✗ gags from gasoline smell: -0.972
- ✓ ✗ sees the price: 1.658

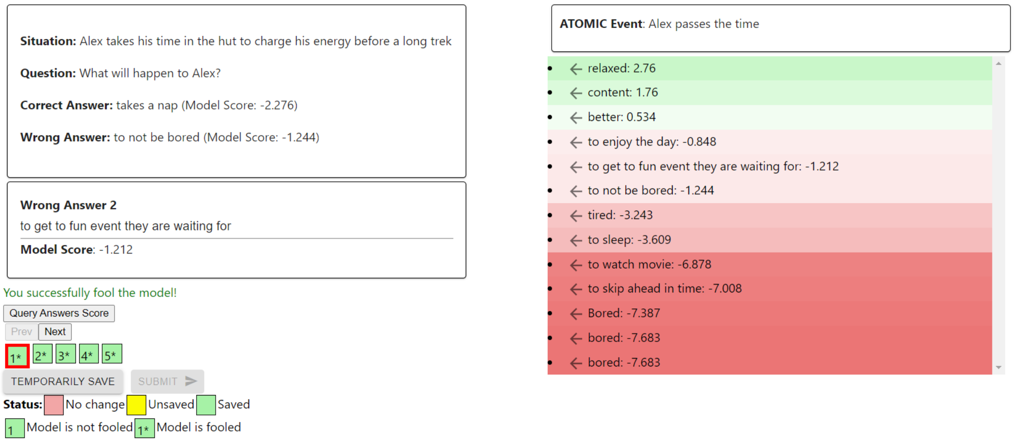
Intention [Because Alex wanted to ...]

- ✓ ✗ to have gas: 3.488
- ✓ ✗ to have fuel for the drive: 6.417
- ✓ ✗ to have enough fuel in the automobile.: 6.77

Need [Before, Alex needed to ...]

- ✓ ✗ money: 0.099

Figure A.1: Annotation interface for the two-choice annotation.



Situation: Alex takes his time in the hut to charge his energy before a long trek

Question: What will happen to Alex?

Correct Answer: takes a nap (Model Score: -2.276)

Wrong Answer: to not be bored (Model Score: -1.244)

Wrong Answer 2

to get to fun event they are waiting for

Model Score: -1.212

You successfully fool the model!

ATOMIC Event: Alex passes the time

- relaxed: 2.76
- content: 1.76
- better: 0.534
- to enjoy the day: -0.848
- to get to fun event they are waiting for: -1.212
- to not be bored: -1.244
- tired: -3.243
- to sleep: -3.609
- to watch movie: -6.878
- to skip ahead in time: -7.008
- Bored: -7.387
- bored: -7.683
- bored: -7.683

Figure A.2: Annotation interface for the third-choice annotation.