

Institute for Visualization and Interactive Systems

University of Stuttgart
Universitätsstraße 38
D-70569 Stuttgart

Masterarbeit

Recreating False-Belief Tests as Visual Question Answering Tasks

Michael Erdemann

Course of Study:	Softwaretechnik
Examiner:	Prof. Dr. Andreas Bulling
Supervisor:	Susanne Hindennach, Matteo Bortoletto
Commenced:	January 17, 2023
Completed:	July 17, 2023

Abstract

Theory of Mind (ToM) represents a key aspect of human intelligence, but it is still unclear whether Artificial Intelligence (AI) can learn this ability. Previous works attempted to test the ToM ability on AI models by using different implementations like text or images but none of them did follow a Visual Question Answering (VQA) approach. This work presents the new data set CLEVR-ToM, which for the first time represents false-belief tests as VQA tasks. By using a VQA approach, it addresses two important human senses with natural language (text) and visual (image) information. Especially for the Sally-Anne test, which tests a location false-belief, this VQA version appears very beneficial as it shows many similarities to the original form of the test. For the testing, this work extends the CNN+LSTM+RN model to a new model CNN+2LSTM+RN to better fit the new CLEVR-ToM data set. The CNN+2LSTM+RN model delivered outstanding results on the CLEVR-ToM data set with an accuracy of almost 98%, achieving higher results than the original model. This work proves for the first time that it is possible to implement the false-belief test in a VQA fashion and that the models can handle the tasks very well. This lays the foundation for further tests of other, even more challenging ToM types, that can be built on this basis.

Contents

1	Introduction	13
2	Related Work	15
2.1	Computational Theory of Mind Tests	15
2.2	Visual Question Answering (VQA)	16
2.3	VQA Models	18
3	CLEVR-ToM data set	19
3.1	Methods	19
3.2	Generation of the data set	32
3.3	Statistics	34
4	User study	39
4.1	Structure and procedure	39
4.2	Evaluation	41
5	Models & Results	47
5.1	Relational Network (CNN+LSTM+RN)	47
5.2	Extended Relational Network (CNN+2LSTM+RN)	49
5.3	Rule-based models	50
5.4	Results	51
6	Discussion	55
6.1	Future Work	58
7	Conclusion	59
	Bibliography	61

List of Figures

3.1	Sally-Anne test	20
3.2	Examples of the CLEVR-ToM data set	21
3.3	Examples of the distinction between <i>normal</i> - and <i>distractor-tasks</i>	31
3.4	CLEVR-ToM generation pipeline	32
3.5	Distribution of the question-types	35
3.6	Distribution of the answer-options	36
4.1	Example of a task from the user study	40
4.2	Statistics of the different belief-types	41
4.3	Correctness comparison of the <i>relational//non-relational questions</i>	42
4.4	Difficulty comparison of the <i>relational//non-relational questions</i>	43
4.5	Difficulty comparison of the correct/incorrect answer sets for the <i>relational questions</i>	44
5.1	CNN+2LSTM+RN model architecture	48
5.2	Comparison of the loss-graphs	52
5.3	Comparison of the accuracy-graphs	53
5.4	Confusion matrix	54

List of Tables

2.1	Object attributes in CLEVR	18
3.1	Example instantiation of the change-colour-action	28
5.1	Accuracy values for the different question-types	51
5.2	Accuracy values for further types	51

Acronyms

AI Artificial Intelligence. 3, 13

ANN artificial neural network. 13

CNN Convolutional Neural Network. 18

LSTM long short-term memory. 18

MLP multi-layer perceptron. 49

POMDP Partially observable Markov decision process. 15

ReLU Rectified Linear Unit. 49

RN relation network. 18

ToM Theory of Mind. 3, 13, 19

VQA Visual Question Answering. 3, 13

XAI Explainable AI. 13

1 Introduction

In the real world, it is important not only to have one's own perspective, but also to perceive the world from other perspectives and to put ourselves in the position of others in order to better understand the environment. This requires the so-called Theory of Mind (ToM) ability, which most people develop early in childhood and which is known primarily from the field of psychology [WP83]. This ability enables us humans to understand ourselves and other people as actors who have subjective mental states such as beliefs, desires and intentions. By understanding mental states, it is possible to understand and predict actions of oneself or others. ToM is an umbrella term that can be divided into different types and sub-types. These can relate to emotions, perceptions, knowledge or belief. In recent years, ToM has been a frequently discussed topic that has been researched in various directions. For example the influence of the educational environment on ToM [SSG20] or the connection between ToM and reading comprehension [DAGH18] was examined.

In the last decade in particular, attempts have been made to make Artificial Intelligence (AI) similar to humans and to teach it human behaviour and thinking. Of course, this also includes ToM abilities, as they are one of the important characteristics that make us human ([Whi93] [as cited in Bar01]). For this reason, attempts have often been made to integrate this type of thinking in machine learning and artificial neural networks (ANNs) [Jar19; RPS+18]. For example, this could be especially useful to increase the human trust in AI, as the Explainable AI (XAI) framework [ALS+19] shows, or it could be helpful for better human-robot interaction [Win18].

While most people acquire this ability, it is not yet clear whether machines, or more precisely AI, can fully learn this ability and really understand and solve tasks that require ToM. To prove this ability, various tests were used that can only be solved by using ToM [BLGB20]. These skill testing tasks come in a variety of forms and were originally designed for people, especially children, to see if they have acquired the ToM ability. A well-known test is the false-belief test or more specifically the *Sally-Anne test* [BLF85], which requires the subject to distinguish between the real location of an object and a person's assumption about its position.

There were already works that have created data sets consisting of textual stories describing the process of the Sally-Anne test and questions about Sally's beliefs. Question answering models were then used for the prediction of the belief [GNG17]. Besides the work with texts, there were also works that used images to test the ability of AI models to detect false-belief cases [EVT16]. But a combination of text and images has not yet been used and tested for these purposes.

An implementation of the task as Visual Question Answering (VQA), i.e. pictures combined with text questions, would however be an important next step, because it serves two key human senses, which are important for people to perceive and assess states of the world. Visual information also appears to play a major role in the development of ToM in humans [MHB98]. Furthermore, the original Sally-Anne test was also posed in a VQA fashion, in which a scene was shown in images or with puppets and a question was asked at the end [BLF85]. This was especially important for the children so that they could understand the task. VQA has also received a lot of attention in the past and many models have been developed for it [WTW+17].

For this reason, the new data set *CLEVR-ToM* is presented in this work, which contains ToM tests, more precisely false-belief tests, implemented as VQA tasks. Specifically, this data set tests three different types of false-belief: The false-belief can be regarding the existence of an object, the attributes of an object or, as in the Sally-Anne test, the location of an object.

The well-known VQA data set *CLEVR* [JHV+17] was taken as a basis, which tested visual and relational reasoning in artificial images and was mastered by various models [PSD+18; SRB+17]. Since *CLEVR* itself was a synthetic data set, it also allowed an automatic creation of the *CLEVR-ToM* data set with all its advantages, including reducing potential biases.

The realisation of the false-belief test using *CLEVR* was achieved through a new task design that used modified questions and added actions to *CLEVR*, going beyond similar work such as *CLEVR_HYP* [SKYB21], which also integrated actions into *CLEVR*. This work also investigated whether models were able to distinguish between several different points in time and were also able to differentiate between the state from the image and the changed states through textual actions and use the correct one to answer the question.

Subsequently, the VQA model CNN+LSTM+RN [SRB+17], which had delivered outstanding results at *CLEVR*, was trained on the new data set *CLEVR-ToM*. In addition to the CNN+LSTM+RN model, this work presents CNN+2LSTM+RN, an extended version of the model with an additional text input. A comparison is also made between the results from a user study, the subsequent work and the influence of the model extension.

In summary, the following contributions resulted from this work:

- The first representation of a false-belief test as VQA-tasks was implemented. Thereby, different time levels were added to the *CLEVR* tasks for the first time.
- A new large VQA data set called *CLEVR-ToM* was created, which uses these false-belief tests and which was verified in a user study.
- Based on the CNN+LSTM+RN model, an extended model CNN+2LSTM+RN was created to fit the *CLEVR-ToM* data set, which was trained and tested alongside various baseline models on the new data set *CLEVR-ToM*. This new model predicted the correct answers for this data set with a high accuracy, suggesting that it can handle one type of ToM ability.

2 Related Work

In this section, the focus is on the three different areas that are particularly relevant to this work. These include, firstly, previous work that has developed ToM tests for AI models, as in this work. In addition, the area of VQA data sets will be introduced, since the CLEVR-ToM data set developed in this thesis is also a VQA data set and was built upon an existing VQA data set. The last part of this section describes several interesting VQA models, as these were all considered as potential baseline models for CLEVR-ToM.

2.1 Computational Theory of Mind Tests

There have been several attempts to develop or test ToM tests and specific false-belief tests for AI. For this reason, there were already different approaches to implement ToM tests, for which different types of ToM were tested and different implementation types were used.

The first work that presented false-belief tests using natural language sentences was published by Grant et al. [GNG17]. They examined whether existing question answering models, e.g. end-to-end memory networks, can deal with false-belief tests such as the Sally-Anne test. Therefore, they designed an automatically generated data set containing tasks using sentences to explain a story line. These sentences could describe actions or beliefs. In these scenarios, Sally could have a true- or a false-belief about the state of the scene. Sally's belief was then tested with a corresponding question. In addition to the different belief-types used, the variation has been increased with the help of different templates and interchangeable objects, whereby the actions only changed the position of the objects.

An extension of this work was presented by Nematzadeh et al. [NBG+18]. In addition to first-order beliefs, second-order beliefs were also considered, which described the belief about the belief of another. Furthermore, additional questions were used in each task that not only ask about a person's belief, but also about the actual state of the scene or the original one. This approach was chosen to test the understanding of the model over the whole scenario.

Similar to the new data set CLEVR-ToM, false-belief tasks were used for the data sets. But, they were implemented as purely textual tasks, which is in contrast to the VQA approach of CLEVR-ToM. In addition, both works used just the location false-belief, changing only the positions of the objects in the story line, whereas for the new data set CLEVR-ToM several types were used.

In the work of Rabinowitz et al. [RPS+18] a different approach was chosen. The ToM topic was regarded as a meta-learning problem and a grid world based on a Partially observable Markov decision process (POMDP) was defined in which agents carried out actions with a specific policy. A global observer that had access to the state of the POMDP and the actions of the agents, but not to the agent's policies, should predict the behaviour of the agents. Among other tests, the Sally-Anne test was successfully tested on an ANN.

The work of Rabinowitz et al. [RPS+18] differed from previous publications and CLEVR-ToM by only working in the grid world and not using natural language sentences or images. However, they also chose false-belief tests (including the Sally-Anne test) for the tasks.

Another work was published by Eysenbach et al. [EVT16] and combined ToM with images by considering the ToM problem of false-belief as a visual task. They created a data set that contained abstract graphic scenes, of eight images each, in which the recognition of people with a false-belief was examined.

By using a visual approach to implementing ToM tasks, the work of Eysenbach et al. [EVT16] shows many similarities to the approach used in this study. However, it did not examine the Sally-Anne test. Furthermore, no text information was used in addition to the images, so it was not a VQA data set like CLEVR-ToM. In addition, the work only addressed recognising the timing and the person who has a false-belief and does not query the false-belief itself.

Besides the well-known false-belief test, there are also other types of tests that cover other areas of ToM. Like for the false-belief tests, there have been attempts to implement them for AI. One example is the work of Labash et al. [LAM+20], which investigated whether ANNs can deal with perspective taking tasks and learn the necessary skills. The way in which the test was implemented is similar to that of Rabinowitz et al. [RPS+18], because a grid world in which agents perform actions was also used. The work showed that at least in some cases it is possible for machines to learn perspective taking skills using reinforcement learning.

As this selection of papers shows, computational representations of ToM tests for AI models are an emerging topic. Especially the false-belief test has been addressed by many papers, but so far they have been implemented without VQA. In some cases, they used the individual components such as texts and images but not the combination of both.

Since the final data set CLEVR-ToM is a VQA data set and CLEVR was used as a basis for this work, it might be worthwhile to look at a selection of the most important VQA data sets.

2.2 Visual Question Answering (VQA)

VQA extends the classic textual question answering by combining two modalities: In addition to an image as a source of information, textual questions are asked that relate to the image.

By using the visual information (image) and natural language (text), it addresses two senses, which play an important role in the development of ToM in humans [MHB98]. For this reason VQA was very interesting for the developing of a ToM test and was taken as the approach for the data set. In addition, this form of image and question is also similar to the original Sally-Anne test [BLF85]. This connection of images and text, which is very natural and easy for humans to understand, was a challenge for previous AI. For this reason, various data sets were created, which were the basis for the development of new models.

One of the first large data sets was the data set VQA [AAL+15], which consisted of natural and artificial images and open-ended natural language questions about the image. The questions, which were created by humans, required working with the image, the identification of objects in an image and in most cases a further interpretation of the scene.

The COCO-QA data set [RKZ15], on the other hand, consisted only of natural images, but the questions and answers were generated artificially. To enable this artificial generation, existing textual descriptions of the images were used. In this data set, the questions were mainly aimed at recognising the scene with the objects, counting and identifying the colour of objects and determining spatial positions.

These works pushed the development of VQA and the associated models. Nevertheless, they had the problem of data generation. Even though artificial images have been used to some extent, or the question- and answer-generation has been automated, human interaction were still necessary. Hence, the generation of large data sets still required a huge amount of manual work by a human operator. Furthermore, the possibility to adapt the tasks to the needs and to reduce it to the most essential was much more limited. However, if one can modify the tasks appropriately, the likelihood of biases that allow the model to take shortcuts can be reduced as the balance of the data can be ensured. Especially with natural images, adaptability is difficult.

For this reason, an artificial generation of the data set including images, questions and answers is useful and was also used for the generation of the CLEVR-ToM data set.

A well-known VQA data set consisting of artificially generated images and questions is CLEVR [JHV+17]. In this data set, questions about different three-dimensional geometric bodies in a room were used as tasks. Due to various properties of the objects (listed in table 2.1), many questions could be formed, which query these various characteristics of objects. The questions could be simple counting and comparison tasks, but could also require logical thinking as well as memory and query object data indirectly through visual reasoning. For this purpose, in addition to *non-relational questions*, in which the question concerned the entire scene, *relational questions* were also used. In these questions, it was necessary to determine the relevant part of the image for the question by using an object specified in the question (*relational object*) together with a direction.

CLEVR was an important basis for this work, as it represented a powerful data set, which was less susceptible to biases due to the artificial generation and could therefore be controlled more easily. In addition, the structure of the images offered possibilities for the illustration of ToM tests. Another advantage was the variety of extensions available for CLEVR.

CLEVR_HYP [SKYB21] was one of the extensions of the CLEVR data set and added another level to the tasks. This hypothetical layer was formed from textual actions applied to the image. In addition to these action-texts, the questions have been adjusted so that they no longer referred directly to the image but to the hypothetical level. Models must therefore be able to reproduce this hypothetical state to answer the question correctly.

This work built also on the idea of CLEVR_HYP because this additional layer allowed extensions to add temporal context to CLEVR. Since the Sally-Anne test consists of a sequence of actions, a static image like CLEVR was not sufficient to represent this test. Therefore, the application of actions to the scene and the creation of different states of the scene over time was necessary.

This overview of important VQA data sets shows that there has been a lot of development work has been done in in this area recently. Furthermore, it indicates that CLEVR with the extension CLEVR_HYP brings many benefits and was thus an important basis for the development of the new data set for false-belief tests.

2 Related Work

Attribute	Value space
Shape	cube, sphere, cylinder
Material	rubber, metal
Colour	blue, brown, cyan, gray, green, purple, red, yellow
Size	small, big

Table 2.1: Object attributes in CLEVR [JHV+17].

2.3 VQA Models

In addition to the VQA data sets, a variety of models have been developed which are designed for VQA tasks and can particularly master existing VQA data sets. Since a model was also tested on the CLEVR-ToM data set in this work, it was necessary to deal with the various models and their characteristics.

The challenging CLEVR data set released in 2017 had a strong impact on the development of new models. In addition to combinations of long short-term memories (LSTMs) [HS97] and Convolutional Neural Networks (CNNs) [JHV+17], new models have been developed that rely on relation networks (RNs) [SRB+17] or transformers [TB19] to deal with the *relational questions* and the relationships between the objects. Earlier models (e.g. [NSH16]) used CNNs as the main component. However, these did not manage to cope with the relational relationships in the tasks of CLEVR.

In the same year as CLEVR, DeepMind introduced a new model CNN+LSTM+RN [SRB+17], which achieved state-of-the-art performance for CLEVR at the time of publication. It achieved the excellent performance not only for the *non-relational questions*, but also for the *relational questions*. The advantage of this model was the integration of a RN alongside the CNN and LSTM, which was critical to answer the *relational questions* in the CLEVR data set. This network made it possible to combine different retrieved objects from the image by a CNN, which allowed relationships between objects to be included in the calculation.

In 2019, a transformer-based approach was presented with the LXMERT model [TB19], which was not developed directly for CLEVR, but can be adapted and used for VQA tasks. This work was chosen as the second baseline in CLEVR_HYP and fine-tuned for this purpose. The model used specific encoders for the objects extracted from the image and the question as textual input. They were connected to each other with a special cross-modality encoder, allowing the relationship between the objects in the image and the words of the question to be taken into account.

In addition, many other VQA models have been created, such as FiLM [PSD+18], abstracting from the relational calculations as in the work of Santoro et al. [SRB+17] by using general-purpose components in multiple layers. This model was able to learn the ability of visual thinking and thus achieve a good generalisation ability.

Since these models have achieved good results for CLEVR or CLEVR_HYP, they are suitable candidates for CLEVR-ToM. In this work, the model proposed by DeepMind [SRB+17] was used.

3 CLEVR-ToM data set

3.1 Methods

3.1.1 Basis of this work

Theory of Mind (ToM) incorporates several different aspects and can be broken down to multiple sub-types. For this reason, there is no test that examines all aspects of ToM. This is also the reason why this work only focuses on the area of beliefs, or more precisely the distinction between a true- and a false-belief for which there is the so-called false-belief test. This test examines whether a person can assess another person’s perspective and understand the status of a scene from the other person’s point of view. A false-belief occurs when the state of a person does not correspond to the actual state of the scene. This term is again an umbrella term, as false-beliefs can refer to different areas and occur in different ways.

A good example for this kind of test is the location false-belief test or more specifically the Sally-Anne test, which was developed primarily for children [BLF85]. This test consisted of a scene in which the position of a ball changed. In addition, questions were asked that query Sally’s belief about the current position of the ball, which could differ from the actual position. A test person had to assess the perspective of the person Sally in this scene in order to answer the question correctly. Figure 3.1 shows the scenario of the Sally-Anne test. In this scenario, Sally first hides a ball in a basket and leaves the room. Anne changes then the location of the ball by putting the ball in the box. Afterwards, Sally comes back into the room to get the ball. The question asked to the test person then was: Where is Sally looking for the ball: In the basket or in the box? In the example, Sally has a false-belief, as she thinks the ball is in the basket since she didn’t see the change of location of the ball. However, if she had been in the room during the swap, she would have a true-belief, knowing that the ball is now in the box.

There have already been implementations of this test for machine learning models, but so far none that were formulated as VQA tasks. As stated in Section 2.1, previous work has only used text or images, but has not used a combination of both types as input.

To make this kind of test suitable for machine learning models in a VQA fashion, the well-known CLEVR data set [JHV+17] was used as a basis for the new data set. CLEVR introduced three-dimensional objects with the properties colour, shape, size and material, which lie in a three-dimensional space in an image. By using a fixed perspective, it enabled spatial relationships between the objects in the space, e.g. in front, behind, left and right. The potential values of the object properties can be seen in table 2.1. These images made it possible to define a scene at least for a specific point in time, which consisted of concrete locations and attributes of the individual objects. In addition to the resulting images, the CLEVR data set also consisted of textual questions

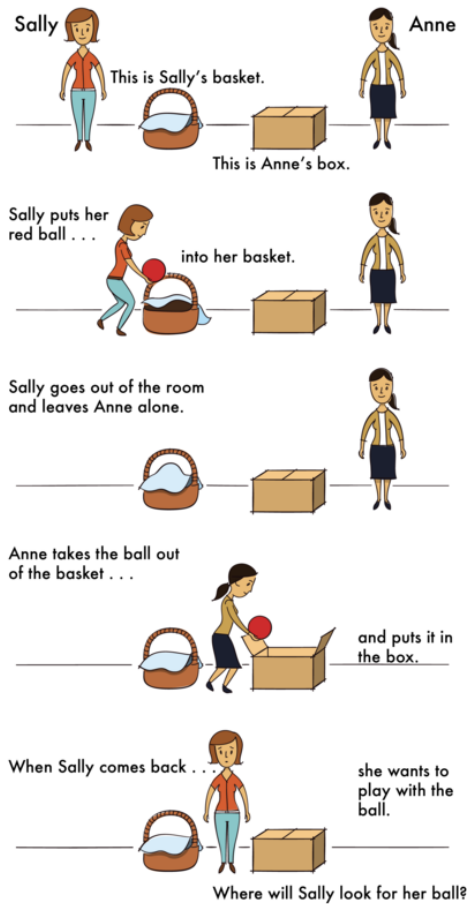
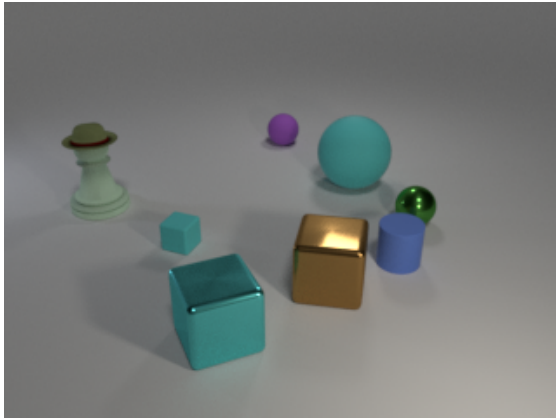


Figure 3.1: Example of the Sally-Anne test [Mal22].

that relate to the scene and required visual reasoning. These questions allowed to query the status of the scene and could concern the locations or attributes of objects. This was very important for this work as it helped to detect potential false-belief cases.

Since CLEVR was used as basis, the components of the false-belief test had to be converted into a suitable form and the structure of the CLEVR tasks had to be adapted and extended. The final data set from this work, called CLEVR-ToM, is presented in more detail below. Examples of this can be seen in figure 3.2.



Remove-action + Existence-question (non-relational)

- False belief:
 - A: Remove the agent. Then take the small die out of the scene.
 - Q: Does the agent expect that there is a cyan die made of rubber at the end? ?
 - Answer: True
- True belief:
 - A: Take the small cube out of the image. Then the agent leaves the scene.
 - Q: Does the agent expect that there is a cyan die made of rubber at the end?
 - Answer: False

(a) Example task with true- or false-belief about the existence of an object by using a *remove-action* and a *existence-question*.

Swap-action + Count-question (relational)

- False belief:
 - A: Take the agent out of the image. Then the positions of the rubber cylinder and the large ball are swapped.
 - Q: What number of objects does the person expect have a big size and are in front of the green metallic object at the end? ?
 - Answer: 2
- True belief:
 - A: Change the position of the rubber cylinder with the big ball. Then the person exits the image.
 - Q: What number of objects does the person expect have a big size and are in front of the green metallic object at the end?
 - Answer: 3

(b) Example task with true- or false-belief about the position of an object by using a *swap-action* and a *count-question*.

Change-action (Colour) + Attribute-question (relational)

- False belief:
 - A: Remove the agent. Then change the color of the little cylinder to green.
 - Q: What is the person's opinion on the color of the tiny rubber thing that is to the right of the purple rubber object at the end?
 - Answer: blue
- True belief:
 - A: Make the small cylinder green. Then take the agent out of the image.
 - Q: What is the person's opinion on the color of the tiny rubber thing that is to the right of the purple rubber object at the end?
 - Answer: green

(c) Example task with true- or false-belief about the attribute (colour) of an object by using a *change-colour-action* and a *attribute-colour-question*.

Figure 3.2: Examples of the CLEVR-ToM data set.

For each task an image (the same for all tasks in the example), an action-text (A) and a question (Q) with the corresponding answer were given. The action-text describes various actions that are carried out one after the other in the scene of the image. The agent can only see the state of the scene if that agent is in the scene at that time. For this reason, the agent's belief about the state can then differ from reality (false-belief) or correspond to it (true-belief). The spatial relations in the questions always refer to the perspective of the image and not to that of the agent.

3.1.2 Similarity to the Sally-Anne test

Time dimension

An important feature of the CLEVR-ToM data set was the introduction of actions to CLEVR. The aim of this idea was to apply the actions described in sentences to the image one after the other, thus creating several time levels and states of the scenario. Since the Sally-Anne test [BLF85] consisted of a continuous story-line with multiple actions and changes, it was also important that the data set in this work could represent this plot with its several time levels. Since the original CLEVR data set had no time component and this test could not be represented in a static image, the introduction of the actions had been a necessary design decision.

The method was chosen for this work because it is easy for humans to understand and because there was already a related work, CLEVR_HYP [SKYB21], which added so-called action-texts to CLEVR to create another hypothetical state from the image. These actions could remove objects, change object attributes, or even move the objects within the scene.

In contrast to CLEVR_HYP, which changed the image by the action and asked questions about this new state, in this work there should be three different temporal levels that can all be queried. For this purpose, a second action was added to the action-text: In addition to the time point represented by the image, there were two more time points after the first and second action. This sums up to three time levels per scene. The individual actions were placed in a temporal sequence with the help of the term „Then...“. This word was also used in the CLEVR_HYP data set to connect actions, but in this case, it indicated a strict temporal sequence.

The composition of the two actions in the action-text was not random and consisted of two actions of different kinds. Exactly one of these actions was about removing the agent (the chess pawn) from the scene. This was a fixed action in each task. In contrast, the other action could affect any object (not the agent) or object pair and could remove it, swap it, or change one of the attribute values of the object. The sequence of both actions was thereby crucial for the occurrence of a false- or a true-belief.

The figure 3.2 shows three examples from the final CLEVR-ToM data set. Each of the examples represents a different action. In figure 3.2a, an object was removed in the action-text, while in figure 3.2b, two objects were swapped with each other. In the last example (3.2c), an attribute (more precisely the colour) of an object was changed.

Agent

The implemented actions replaced the concrete character Anne in this test, as she was only important to perform actions in the scenario. These actions were now performed in text form, like those from the CLEVR_HYP data set, without reference to a concrete person. This change did not alter the aim of the test.

In addition to the actions, a chess pawn wearing a hat was integrated into the image in the CLEVR-ToM data set. This figure was supposed to be the equivalent of the concrete person Sally. This is why it also resembles a human being due to its shape, but the figure was still an object in the scene and thus should not create too much contrast to the original CLEVR objects. This chess pawn has been given a hat that humanised it for the viewer and made it more clearly recognisable as a typical

agent, so that people should better understand the questions. As with the other CLEVR objects, there were different colours for the agent. However, these were not from the fixed colour palette of CLEVR, but could have any colour across the RGB-spectrum separately for the figure body and hat. The purpose of this was that the variation was large and the distinction from normal objects became clearer. Size and material of the agent were fixed and could not take on different values. The integration of the agent or the character Sally into the test was very important because she was to observe the scenario and her belief was essential to answering the question.

To ensure that the data set was general, the concrete name *Sally* was not used, but only the words *agent* or *person*. These names also allowed clear recognition of the agent by people who do not know the original test and only have the image at hand. The name *agent* is also used in the following as a synonym for Sally in this context.

In the examples in figure 3.2 this design decision can be observed along with the addition of the actions. The agent is shown as a chess pawn with a hat in the image on the left and is referenced in the actions and questions with the words „agent“ or „person“. Like the classic test, the aim was to be able to assess the level of knowledge of the agent, i.e., to understand the agent’s belief about the position of an object. In this case, the belief was influenced by whether the agent had noticed the change in location of an object.

In contrast to the original Sally-Anne test, the agent did not perform any action itself, i.e., the object asked for was already in a place and has not been placed there by the agent in the beginning. This made no difference to the test itself, as the agent could see the object and thus know the current position of the object and its state. It is important to note that in the image of the task it did not matter at which place the agent appears in the scene, the variation was only for learning the recognition of the agent. For the agent, all objects were visible if the agent was in the scene, even if an object would seem to block the direct view from the agent according to the image.

3.1.3 Diversity of variants

False-belief & action-types

Several types of actions and questions were also used for the data set to test three different types of false-belief on the models. In the classic Sally-Anne test, an object was moved from one place to another without the person noticing the action, creating a false-belief about the object’s location. In this data set, in addition to the location false-belief, a false-belief regarding the existence of an object and about an attribute of an object was also used. The different types are presented in more detail below.

On the one hand, the false-belief, in this case more precisely a location false-belief, could be caused by a change of location of an object. However, in this case, not only was an object moved to a different location, but the object was swapped with another. This action is called a *swap-action* in the following. In this scenario, this had the advantage that the new location of the object did not have to be named explicitly but was implicitly determined by the swapping of the objects. The possible false-belief in this case even concerned two objects, which could both be referenced in the question.

Unlike the Sally-Anne test where the object was not visible before and after the action (because it was in the basket or box), the queried object and the current location of it was always visible to the

agent and not hidden when the agent was in the same scene. This modification, however, did not change the intention of the test.

This type of false-belief can be seen in the example in figure 3.2b where a *swap-operation* is performed in the action-text. As with all false-belief cases (in contrast to the true-belief case), the agent leaves the scene as the first action. The second action, which is executed afterwards, is the actual *swap-action*. This action indicates that now two objects swap positions, which are specified with the help of certain attribute values. In this case, the position of the „rubber cylinder“ (the small blue cylinder on the right of the image) is swapped with that of the „large ball“ (the large cyan sphere in the background of the image). Only if the agent had noticed the swapping of the objects would they know the current positions of the two objects. However, since the agent in the false-belief version left the scene before the change was carried out, we have a false-belief for questions involving the position of the objects.

Another type of false-belief concerned more generally the existence of an object (existence false-belief), i.e., whether an object was in the scene at the end or not. This can also be compared to a Sally-Anne test in which the new location would not be in the scene. For example, Anne can take the ball and remove it from the room. Sally would still look in the room where she had last seen it. Again, we have a false-belief.

For this, the *remove-action* was added, which removes an object that was already in the scene. In this case, there is a time when the object was in the scene and a time when it was not in the scene. Depending on whether the agent has noticed this removal, a false-belief might have arisen here.

An example in which an existence false-belief occurs with the help of the *remove-action* is shown in figure 3.2a. The object to remove is described here in more detail with two attributes. In this case, the „small die“, i.e. the small cyan-coloured cube, is removed from the scene with the action. Also in this case, the agent did not notice the removal of the object in the false-belief version, as they left the scene before the action.

Initially, an *add-action* had also been planned that would create the same type of false-belief, but as is also mentioned in Section 3.1.4, it was decided against.

The third type of false-belief, the attribute false-belief, was caused by the so-called *change-action* and concerned the attribute values of an object. Relating this to the Sally-Anne test, this would be, for example, that Anne recolours the red ball with a blue colour without Sally seeing it. When Sally then re-enters the room, she believes the ball is still red and would look for it. Even if the location is the same, she would have a false-belief about the ball.

In the data set, the *change-action* could affect all attribute-types. That means it was possible to change the colour, as in the example, but it was also possible to change the size, the material or even the shape of an object. It may be a little difficult to understand how to enlarge an object or change the material, but it did not influence the resulting false-belief. Because of this variety of possibilities, the flexibility of the data set was very high and the richness of the possibilities of the CLEVR data set were used to build potential false-belief cases.

Another example can be seen in figure 3.2c, which shows an attribute false-belief using the *change-colour-action*. In this particular example, the colour of the „little cylinder“ is changed to green by applying the action, whereby in the false-belief case the agent did not notice this colouring.

The actions executed in the false-belief version of the task did not necessarily have to result in a false-belief, because a distinction was made between *normal-* and *distractor-tasks*. Only in the case of *normal-tasks* the changed object was the object that was queried and could lead to a false-belief of the agent when answering. In *distractor-tasks*, the changed and the queried objects

were independent of each other, which means that in all cases there was a true-belief, but the action sequence could still correspond to that of a false-belief. More details on this and the reasons for the distinction between the tasks are explained in the Section 3.1.4. The examples in figure 3.2 are all *normal-tasks*, as the answer between the true- and false-belief case is different.

Question-types

In addition to the different actions, there were also different types of questions. The questions were used to query the state of the scene from the agent's point of view and to identify potential false-beliefs from the answers. Compared to the Sally-Anne test, the questions were somewhat more indirect. For example, they did not ask where the object is, but asked for this information with the help of reference points and different question-types. The questions were divided into three types.

On the one hand, there was the direct question about the *existence* of an object. This object was determined in the question by various attribute values, whereby not all attribute-types had to be specified, but enough so that the object could be clearly identified. The *existence-question* asked whether such an object with these attribute values existed in the scene according to the agent's belief at the end. The answer differed for the *normal-task* between false-belief, where the agent has leaved the scene before the action was applied, and the true-belief, where the agent exited the scene after the action. With the *existence-question*, this object could either exist at the time or be missing in the scene.

This behaviour can be seen in the example in figure 3.2a, where the question is used in combination with the *remove-action*. In this case, the question asks the belief in the existence of a „cyan die made of rubber“. This is also the identical object that was removed with the *remove-action*. Since with the false-belief case the agent leaves the scene before the actual *remove-action*, the agent knows nothing about the removal and believes that the final state of the scene is the same as in the image. Since the queried object still existed in this state, the correct answer here is *true*. As this answer does not correspond to the real correct state, we have an actual (existence) false-belief in this case. With the true-belief, the actions are executed in the other order. That is, first the actual *remove-action* took place and then the agent leaves the scene. In this case, the agent noticed the removal of the object and also thinks that this object is no longer in the scene at the end. For this reason, the answer in this case is *false* and thus corresponds to the correct end state of the scene, which is why the agent does have a true-belief.

This question could be asked for all possible action-types. This means that not only the *remove-action* that changes the existence of objects was possible, but also by changing the position or the attribute values this question was applicable. For example, before changing the colour of an object, the asked *red cube* is present, but when it is coloured *blue*, it is no longer.

The answer-options for this question-type were *true* or *false* and both options could occur in the true-belief as well as the false-belief case.

Another type of question was the *count-question*. This extended the *existence-question* with further answer-options. Now not only *true* and *false* were possible as answers, but all integers between 0 and 6, which means that there were seven different answer-options.

The question and the attribute values specified in it were no longer aimed at a single object but could concern several objects or even all objects in the scene. Thus, the general question about the number of all objects in the scene was also a valid option at least in combination with the *remove-action*. In general, however, the question-type was again applicable in combination with all types of actions.

An example of this question type can be seen in figure 3.2b. In this case it is combined with the *swap-action*. The question here is aimed at the objects that „are in front of the green metallic object“ (the green sphere) and have a large size. This is true for three objects if we look at the whole scene, i.e., the cyan coloured large cube, the large brown cube and the large cyan sphere. However, since the sphere is behind the green object, it is not considered in the question and the answer is 2. In the true-belief case, however, the agent again sees the *swap-action* and notices the new state where the cyan sphere is in front of the green ball. Thus, the answer from the agent’s point of view is now 3.

The third question-type was the question about a specific *attribute* value. This question-category itself is divided into four different sub-types for each of the attribute-types. Thus, in addition to the question about the colour of an object, the material, the size or even the shape of the object can be queried. For each of these sub-types, the number of answer-options is different and corresponds to the possible attribute values for each of the types (see therefore table 2.1).

This question-type was different from the other two types, because in this case the object asked for must be always present in the scene. As with the *existence-question*, the attribute values given in the question targeted exactly one object. To made the question non-trivial to solve, the value for the requested attribute-type was not given. Due to the nature of the question, it was only applicable to the *change-action* and not to the other two action-types. The subcategory of the question also had to match that of the action, as the question should query specifically the attribute-type that was changed in the action.

This can be seen for example in figure 3.2c. In this case it is a *change-colour-action* in combination with an *attribute-colour-question*. The question in this example is aimed at the objects „to the right of the purple rubber object“ (the small purple sphere, which is in the background). The object we are looking for is a *small cylinder*. Such an object exists in the searched area (which is a necessary condition for this kind of question) and is in this case the blue cylinder in the right part of the image. In the false-belief case, where the agent initially left the scene, the agent only remembered the original colour of the cylinder, which was *blue*. In the true-belief case, however, the agent saw the changed colour and knows that the last colour of the object was *green*.

In addition to the general types of questions, a distinction was also made between *relational* and *non-relational questions*. *Non-relational questions* always targeted the entire scene, while *relational questions* further delimited the objects concerned in the scene. This separation was dependent on a *relational object*, which was clearly specified as a reference point in the question with the attribute values and was always in the scene. This object was not affected by the actual action, except in some cases for the *swap-action*. In addition, a direction was specified, which could be left, right, front, or back. These then limited the scene to the part and the objects that were in this direction to the *relational object*. The indication of the direction referred to the perspective of the image and was not dependent on the position of the agent. For this reason, the agent also did not have a face with a line of sight.

In the examples in figure 3.2b and 3.2c, *relational questions* are also present. As is usual with a *relational question*, the questions again specify a *relational object*. For example, in figure 3.2b this *relational object* is the „green metallic object“ (the small green sphere on the right side of the image). The question targets all the objects that are in front of this green sphere and have a large size. Objects that do not lie in the relation, i.e. in this case lie behind the green sphere, are not considered for the question.

The *relational questions* were an important part of the original CLEVR data set, which is why models that performed well on the CLEVR data set were also good at understanding these relations. In this data set, they were used not only to increase variation, but also because it was possible to ask

questions about the position of an object. This was because, unlike the Sally-Anne test, where it was possible to ask directly about the location of the ball, it was harder to ask for the exact location in this case. In the *non-relational questions*, the answer was the same whether the object was on the far right or on the front left. With *relational questions*, this distinction was possible. This made it possible to use the *swap-action*, as it only changed the location of an object within the scene and not the overall state of the scene. The *swap-action* was therefore a special case that could only be used in combination with *relational questions*.

Templates

To structure the natural language sentences, templates were used that specify parts of the sentence and made them variable with placeholders. Usually, several alternative sentence constructions were also offered. The choice of the templates played an important role in this work. These templates could not be taken directly from the CLEVR data set, as they were not targeted at the agent's belief. It was important that they could be clearly understood by humans and that they did not contain any ambiguities or obscurities. The templates specified how a question, or an action is structured and contained placeholders for variable places such as attribute values. For each of the question- and action-types there were also several template groups, which in turn could consist of several templates and allowed a wide variation of sentences. The resulting texts could again be modified with synonyms.

As there were no actions in the original CLEVR data set, it was necessary to create completely new templates for the actions. These templates were created for each of the action-types: *remove*, *change* and *swap*. Selected examples from the CLEVR_HYP data set served as inspiration, as the creation code of this data set with its templates was not available. However, unlike CLEVR_HYP, the action-text consisted of two concatenated actions and other actions such as removing the agent were added. An example template for an action is: „Remove <Z><C><M><S> from the scene.“. The letters in the brackets indicate different variables, e.g.: Z=Size, C=Colour, M=Material and S=Shape.

Two actions were then linked together with the word „Then“. This results in the following scheme for the action-text: „<Action1>. Then <Action2>“, for example: „Remove <Z><C><M><S> from the scene. Then paint the <Z2><C2><M2><S2> with <C3> color.“

For some of the questions, texts and templates could also be taken from CLEVR and CLEVR_HYP. However, since the CLEVR and CLEVR_HYP questions only targeted the final state of the scene and did not address beliefs of others, the question-templates were modified to directly target the agent's belief in the scene. For example, a template for a *count-question* can look like this: „How many <Z><C><M><S> does the agent think are there at the end?“.

A collection of possible question-types for the *change-colour-action* and a possible instantiation of a template for both the action and the question can be seen in the table 3.1.

In order to avoid using only the same terms in the templates, synonyms were added for some words. It was important that these did not change the meaning and were clear and precise. For example, the terms „expect“ and „assume“ were added as synonyms for „think“. For „agent“, the term „person“ was added, as this character also reminds of a person. The name „Sally“ was not applied, as this would be too specific and it is not directly clear who Sally is. To ensure the quality of the sentences, they were not only checked by three people but also tested in a user study (see Chapter 4).

Action	<i>change-colour-action</i>
Possible question-types	<i>existence-question / count-question / attribute-question</i>
Template action-text	The agent leaves the scene. Then change the color of the <Z><C><M><S> to <C2>.
Template question	What does the agent think is the color of the <Z2><C2><M2><S2> [that is] <R> the <Z><C><M><S> at the end?
Example action-text	The agent leaves the scene. Then change the color of the big metal object to blue.
Example question	What does the agent think is the color of the cylinder that is behind the red cube at the end?

Table 3.1: Example of an instantiation of the *change-colour-action* with the *attribute-question*. The letters in the brackets indicate different variables, e.g.: Z=Size, C=Colour, M=Material, S=Shape and R=Relation. Different objects are delimited with numbers in the brackets. There is no direct relationship between the values of the question and the action.

3.1.4 Prevention of text leakage

In order for CLEVR-ToM to truly be considered as a VQA data set, an important feature was that the image must actually be used to answer the questions and that no information can be derived from the text alone.

To identify this problem, a pure textual variant of the CNN+2LSTM+RN model presented in Section 5.2 was tested on the data set. This model took only the textual information, i.e. the actions and questions as input, but not the image. However, during testing an previous version of the data set, where actions had always affected the answer to the question, with this textual model, very high accuracy values came out for certain categories. These indicated that for the *existence-* and *attribute-questions*, the answer could often be read directly from the action-text.

On closer examination of the tasks, this behaviour was obvious. For example, if the colour of an object was changed to *red*, the answer to the *attribute-colour-question* in the true-belief case, that is when the action is considered, was the same as the new colour from the action. In this case, the answer *red* would have been correct. If the possible answer-options only consist of two values, it was also possible to determine the answer for the false-belief case, as this always represents the other value than the one from the true-belief. This was the case here for the *attribute-questions* on material and size.

The same problem existed with the *existence-question* in combination with the *remove-action*. If an object was removed with the *remove-action*, for example, the answer was always *false* in the true-belief and *true* in the false-belief case, because the queried object matched the deleted one.

In other combinations with the *existence-question*, e.g., with a *change-action*, this problem was less prominent, as the answer was not always constant depending on the belief-type. Nevertheless, it was possible to get the correct answer if, for example, the changed attribute value from the action was present in the question or not. The problem did not exist with the *count-question*, because

only the probability of guessing could be improved by including the action, but no general solution strategy existed only from the textual information. To solve this problem, an extension of the data set was necessary.

For this reason, so-called *distractor-tasks* were added to the existing data set. These tasks were meant to be like normal tasks in the data set, but in these tasks the action had no effect on the answer to the question.

The aim with the original tasks, later called *normal-tasks*, was that the answers for the true- and false-belief case were different so that the understanding of the different belief-types could be tested. This behaviour can also be seen in figure 3.3, where the responses differ for the different belief-types. For this behaviour to occur, the question always targeted the action, which meant that the action always changed the answer. This had the consequence that examples as shown in the beginning of the section were possible, i.e. that the answer to the question about the colour of an object was already named in the action. For this reason, no knowledge from the image input was required to get the right answer to the question.

With these *distractor-tasks*, however, this was different. In this case, the action could also change the colour of an object to *red*. But the answer to the question about the belief of the colour of a certain object was then mostly not *red*. The answer could sometimes be correct, but only with a random probability. In most cases, the answer was wrong because the object that was changed with the action did not correspond to the object asked for in the question. This means that there was no indication of the previous or the new colour of the object in the action and this information could only be deduced from the image. The answer was identical for both sequences of actions and there was also not a real false-belief case, as the answer always corresponded to a true-belief of the scene. The figure 3.3 shows an example with a similar task both as a *normal-task* and as a *distractor-task*. It is noticeable that in the normal task, the answer from the true-belief case matches the value of the action and is both times *cube* or the synonym *die*. In contrast, the answer in the *distractor-task* is different from the new form from the action and is constant across both belief-types.

The aim of the extension of the data set was that the image must be used by the model to solve all the tasks and thus fulfils the conditions of a VQA data set. This required that the model was not able to recognise these *distractor-tasks* and distinguish them from the *normal-tasks*.

If a separation were possible for the model, it could adapt the strategy for solving the tasks to the respective task-type. This would mean that only the textual information would still be used to answer the *normal-tasks*. In the case of *distractor-tasks*, the pure image information could then be used, as is done in the false-belief case.

If the model could not do the distinction, then it could also not achieve excellent performance using only the text source and it could only achieve very good performance for one of the two parts (i.e., for the *distractor-tasks* or the *normal-tasks*) and in the other case only achieve performance that corresponds to chance. For a better performance, it was necessary to distinguish these two tasks from each other and use the matching solving strategy, but this required working with the image to decide whether the object from the action corresponds to the one from the question. Thus, for each task, the image would first have to be used to check which case occurs.

A distinction without the image information could be possible if, for example, the object attributes from the affected object were compared with those from the question. If they matched, the likelihood that this was a *normal-task* would be very high. On the other hand, if hardly any or no attributes match, it would most likely be a *distractor-task*.

For the tasks to be perceived as similar as possible, the generation of actions and questions was adapted. An important change was that actions always use exactly two attributes to identify an object. Furthermore, questions should also use at least two attributes, of which exactly one should correspond to those of the action. There should also be no contradictory values, e.g., the action referred to a red object and the question referred to a blue one. This was to ensure that no indication of the connection between the action and the question can be inferred from the object description of the action. However, the adopted attribute was never of the type colour, as the variance of the answer-options was too large compared to the other types and this case would then occur much less frequently for the *distractor-tasks* than for the *normal-tasks*.

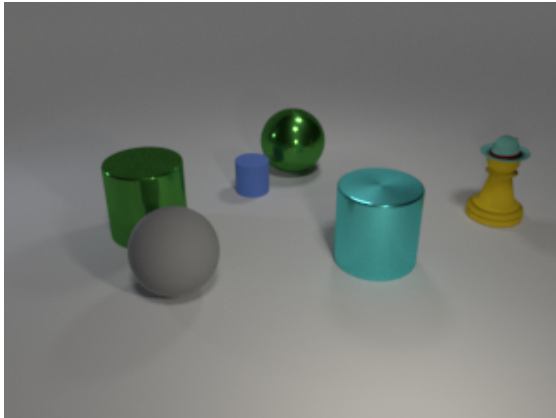
This similarity of the questions can also be seen in the example in the figure 3.3, as there is always exactly the same attribute for both tasks (marked in green), but nevertheless one represents a *normal-task* and the other a *distractor-task*.

Furthermore, the *add-action* was not used for the final CLEVR-ToM data set because the attribute values of the new object were derived from the action alone and could not be found in the image, since the object did not initially exist in the scene. For this reason, the correct answer could be read from the text alone and the image would not even be considered as a source of information.

To check whether a model could recognise the *distractor-tasks*, an additional modified data set was created. In this data set, the *distractor-tasks* had a constant answer value for each of the question-types. For example, the answer for the *count-question* was always 6. Thus, a model that could clearly distinguish between these two types of actions could achieve an accuracy of almost 100%. To be more precise, it would then score close to 100% on the *normal-tasks* (where the answer was in the action-text) and on the *distractor-tasks* (where only the constant answer must be returned). A good reference for general performance were the results for the *attribute-size-* or *attribute-material-questions*, as these tasks could be solved entirely by the text in the case of a *normal-task* for both the true- and false-belief case, as there were only two answer choices for both cases.

In the case of the final CLEVR-ToM data set, if the model failed in distinguishing between the tasks, this would lead to an accuracy of about 75%. This is because if the model only followed one of the strategies, then it achieved 100% for half the tasks (e.g., for the *normal-tasks*) and still achieved the random value of 50% for the other half (e.g. for the *distractor-tasks*). Since the number of tasks was not completely identical for the two cases, the actual value for the best case was somewhat higher. For this reason, results close to 75% would show that the model fails in distinguishing between the tasks, whereas results close to 100% would indicate a clear assignment. Results in between show that in certain cases an assignment was possible.

The model that uses only the text information achieved on the final CLEVR-ToM data set an accuracy between 80 and 82% for the *attribute-material-questions* and for the *attribute-size-questions*. The results are similar for the other types of *attribute-questions*. For example, the results for the *attribute-colour-question* are around 57%, which also suggests an almost random result, as there are eight possible answer-options for this type of question. These results were determined for the final data set, with the constant answers for the *distractor-tasks*. Thus, in most cases, the distraction tasks are hardly distinguishable from the normal tasks. Nevertheless, the accuracy values indicate that for some categories up to a quarter of the cases a distinction was possible for the model. However, since the results of the purely textual model also decreased with this modification (see the results in Section 5.4) and very good accuracy values can no longer be achieved without the image, this extension fulfilled the expectations.



Normal-task

- False belief:
 - *A*: Take the agent out of the scene. Then reshape the **matte** sphere to be a **block**.
 - *Q*: There is a gray **matte** thing; what shape does the agent think it to be at the end?
 - Answer: **sphere**
- True belief:
 - *A*: Convert the **matte** ball into a **die**. Then take the agent out of the scene.
 - *Q*: There is a gray **matte** thing; what shape does the agent think it to be at the end?
 - Answer: **cube**

(a) Example of a *normal-task*

Distractor-task

- False belief:
 - *A*: Remove the agent from the scene. Then make the tiny **matte** thing **die**-shaped.
 - *Q*: There is a gray **matte** thing; what shape does the agent think it to be at the end?
 - Answer: **sphere**
- True belief:
 - *A*: Make the small **rubber** object **cube**-shaped. Then remove the agent from the scene.
 - *Q*: There is a gray **matte** thing; what shape does the agent think it to be at the end?
 - Answer: **sphere**

(b) Example of a *distractor-task*

Figure 3.3: Examples of the distinction between *normal-* and *distractor-task* in the CLEVR-ToM data set. The first example used a *normal-action* and the second used a *distractor-action*. Both tasks used the same image and question. For the *normal-task*, the changed object from the action matches the queried object from the question. This is not the case with the *distractor-task*.

The adopted attribute value between the action and the question is marked in *green*. The new attribute value from the action and the answer is highlighted in *yellow*.

3.2 Generation of the data set

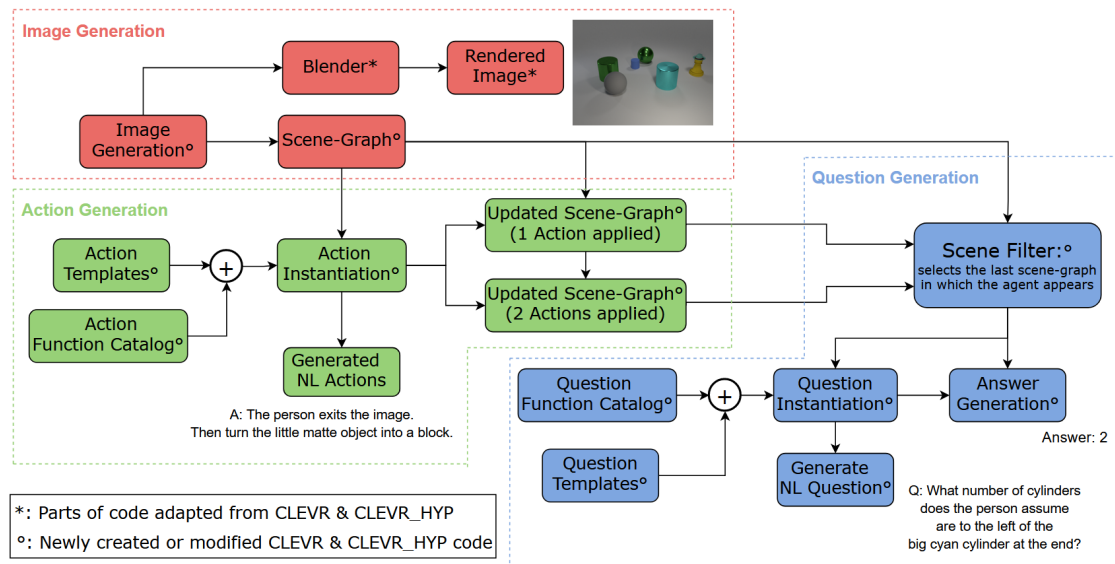


Figure 3.4: Simplified pipeline showing the most important constructs of the generation of the CLEVR-ToM data set. Deviations from the code of CLEVR and CLEVR_HYP are marked. The figure is based on the image of the pipeline from CLEVR_HYP [SKYB21]. In the pipeline, less important constructs such as the *functional programs* have been omitted for the purpose of clarity.

A big advantage of CLEVR was the use of only synthetic images and an automatic image- and question-generation. In other works (like VQA [AAL+15] or COCO-QA [RKZ15]) the images and questions often had to be selected by people or taken from existing sources, which limited the influence on the selection. The automatic generation allowed a much better control over the resulting tasks, as it was possible to arrange the components of the images individually and reduce the image elements to the most essential. In this way, potential interference fields could be avoided and it was possible to create balanced data sets, thus also reducing the risk of potential biases in the data set. Furthermore, images could be generated more easily and significantly larger data sets were possible. For these reasons, a similar automated generation approach to CLEVR was used for the CLEVR-ToM data set, without the need for human interaction.

A simplified overview of the generation pipeline can be seen in figure 3.4. This describes the rough concepts of how the various important parts that later serve as input to the model were generated and build on each other. The creation of the complete data set with the images, actions and the questions with the answers generally proceeds as follows.

First, the images are generated. This is shown in red in the figure 3.4. The process can be started with a command that calls the generation code. In this command, the number of images to be generated can be specified, as well as other possible settings that are to be considered during generation, e.g., the maximum number of objects in the scene.

The image-generation process is then as follows: Random objects are created, i.e., with random attribute values for colour, material, size, and shape (see for possible attribute values the table 2.1)

and with a random position in the scene. Furthermore, the agent is also placed in a random position with a random colour combination for the body and hat. In order for the image to be rendered, the images must meet certain quality criteria, otherwise they will be discarded. For example: the objects must not directly overlap and be at least partially visible in the image. This quality assurance test was also used in the original CLEVR generation code, but improved for this data set so that small objects are more visible and relations between objects are clearer. When the test is passed the image is rendered and saved using Blender [Com18]. In addition, a textual description of the scene, also called „scene graph“, is created. This file contains the exact data for the image, with the position of the camera (used for the generation of the images) and those of the individual objects, including those of the agent. This *scene graph* represents the perfect view of the scene and is useful for later steps. It also contains the relations between the individual objects and possible free positions in the image for future added objects. This file is then saved together with the image. When the specified number of images have been generated, all *scene graphs* are saved bundled into a single file.

The action- and question-generation can then be executed either bundled in one script or separately. In all cases, the action-generation happens first before the question-generation. The action-generation, which is shown in green in the figure 3.4, takes place in a loop over all images for which actions are to be generated. The *scene graph* of the image serves as input in each case. For each of the images, actions are then generated for the different action-types. For each action-type, an attempt is made to create two different instances so that an alternative option is available for each action-type during question-generation. This means that two instances are created for each of the seven action-types (including the different types of the *change-action*), but only one ends up in the final data set for each of the types. In every action instantiation, a random template is taken as input and filled with random values for the attributes. However, the values can also take the null value for an attribute type if it is not specified. Care is taken to ensure that the attribute values always refer to exactly one object in the scene to avoid creating ambiguities. In addition, by creating potential attribute assignments for the questions in advance, it was ensured that both questions for the *normal-tasks* and questions for the *distractor-tasks* can be generated during the later question-generation.

Each generated action-text consists of two individual sub-actions that are executed one after the other. One is the actual change to the objects in the scene and the other is the agent leaving the scene. After each of the individual actions, the newly generated scene is saved as a new *scene graph*, resulting in two *scene graphs* (in addition to the original one) being saved for each action. The changes to the *scene graph* are also contained in a *functional program*, which is also saved with the action and describes the meaning of the action as a sequence of functions defined in a function catalog. From this, the effect of the action on a scene can be traced and reproduced. The affected object(s) of the scene are also explicitly saved in the action-file to simplify question-generation. The generation is always carried out for the true- and the false-belief case with the same objects affected by the action. Only the order of the actions, the possible different choice of words and the resulting *scene graphs* distinguish the two cases. For this reason, there are separate fields for these values for both belief cases in the action-file.

The question-generation, shown in blue in figure 3.4, is then the next step. This is executed separately for the creation of the questions for the *normal-tasks*, as well as for the *distractor-tasks*. The generation code iterates itself over the individual scenes of the images and takes the corresponding action-file as input. As with the action, several questions are instantiated for each action, although this does not necessarily cover all question-types. The selection of the question-type and whether it

is a *relational question* or *non-relational question* in this case is randomised.

The question takes as input the three different *scene graphs* (the original one from the image and the two modified ones from the action) that describe the different points in time of the scene. A selection procedure then chooses the correct scene that corresponds to the belief. For this data set, the last scene in which the agent is in the scene was always taken. The correct answer is then determined with this selected *scene graph*. Answers that have appeared less frequently are preferred by tracking the distribution of answers during the training and rejecting answers that occur too frequently. When determining the attribute values that are asked for, these are also directly limited through a filter in such a way that a meaningful answer can be found and that this answer differs between the true- and false-belief case for the *normal-tasks*. This ensures that the questions for the *normal-tasks* refer to the changed object by the action and ask for the difference between the true- and false-belief case. With the *distractor-tasks*, exactly the opposite is ensured. For the filtering, the attribute assignments previously created during the action generation are also partly used. This prior filtering also speeds up the generation considerably.

The answer is calculated by a sequence of filters and small programs defined in a specific function catalogue. The sequence of these calculation steps is also stored in the question file in a *functional program*, which makes it comprehensible. The creation of the *functional program* and the answer is done for the true- and false-belief case. First the answer for the true-belief case is calculated and then the answer for the false-belief case is calculated with the same calculation steps. The only difference here is the different *scene graph* as input. The answers for both belief-types are then added together as one field to the question file. The question file again also contains information about the templates used and the action including the action-texts, as the final data set only contains the images and the questions with the answers and does not directly use the separate action-files. If no question instantiation is possible because no attribute assignment in the template meets the requirements, other templates for the same question-type are tried first. If this does not lead to a result either, the action is exchanged for another one, as a reserve action has been generated in each case. If no instantiation is possible, this action-type is skipped for this image.

3.3 Statistics

For the final data set, 12 actions were generated for each image, i.e., two for each action-type (remove, swap and change for each of the four attribute-types), whereby only six of the 12 actions were used for the final questions. The rest were used only as reserves. Note, however, that one action contained the case for the true- and false-belief, which is why these would correspond to twice the number.

Since *distractor-tasks* were added, the following question-generation was carried out twice. For each of the actions, an attempt was made to generate 10 questions. The type of question was determined randomly. This led to 60 questions per image. Because of the two parts, namely for the *normal-tasks* and the *distractor-tasks*, this resulted in 120 tasks. There were even 240 tasks in the data set for each image, because in this case the tasks were split for the true- and the false-belief case. Due to the way of generation, the number of tasks for the true-belief case exactly matched those for the false-belief case. It should be noted here, however, that the false-belief case for the *distractor-tasks* only represented an apparent false-belief with the specific action sequence, but otherwise also corresponded to a true-belief.

Since the final data set consisted of 10.000 images, this resulted in a total number of more than

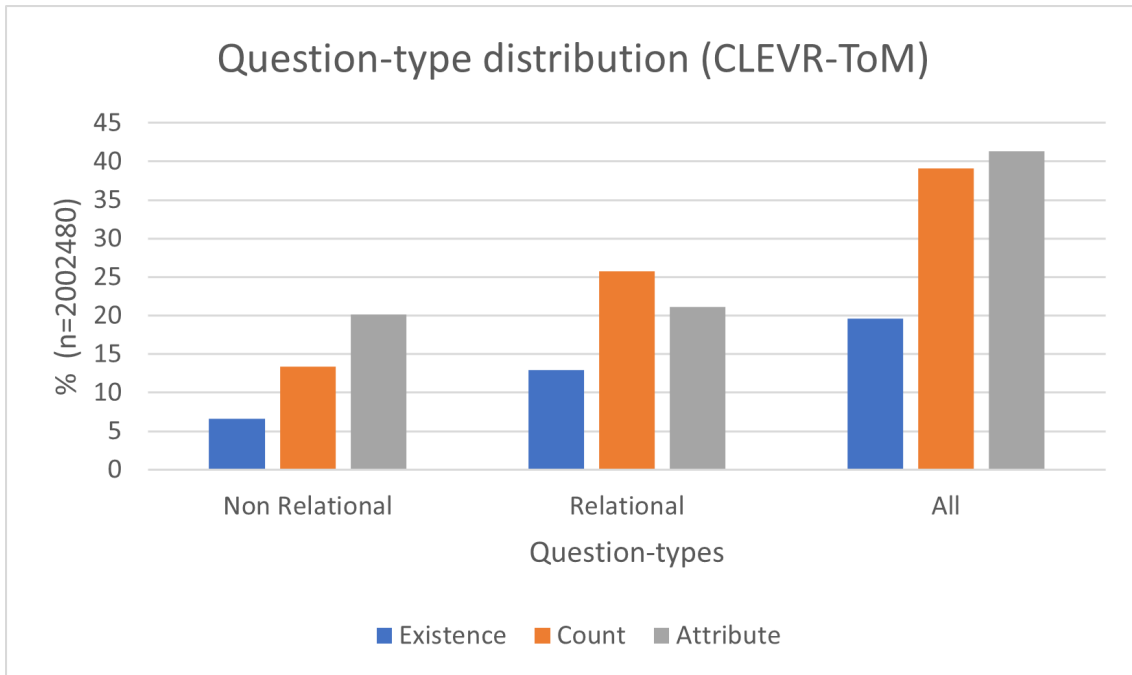


Figure 3.5: Bar chart showing the distribution of the individual question-types in the final CLEVR-ToM data set. In addition to the individual question-types, a distinction is also made between *relational* and *non-relational* questions.

two million tasks, or more precisely 2.002.480. Compared to the expected number of 2.400.000, slightly less than 17% of the tasks were missing. This was because for some images there were no tasks found for certain types of questions and actions.

To generate the data set with the modification, a time of about one and a half day was needed using a *NVIDIA GTX 1080* with *CUDA* activated (only important for image-generation). This included 17 hours for the generation of the images, 0.5 hours for the action-generation, 3 hours for the generation of the *normal-questions* and 15 hours for the generation of the *distractor-questions*.

In order to avoid potential biases, similar questions were combined into question-families in CLEVR and CLEVR_HYP, which should be represented equally in the data set. This idea was extended for this work for further types and sub-types and also to the different types of actions, which were introduced in Section 3.1.3. By attempting a balanced use of the three different action-types, the different types of tested false-belief (location, existence and attribute) should also be equally represented.

The generation approach described in Section 3.2 was intended to ensure that the different action- and question-types occur evenly in the data set. Since the *attribute-questions* could only be asked for the *change-actions*, the probability for this type of question was increased for these actions. The distribution of the question-types can be seen in figure 3.5. It is noticeable that the *existence-question* occurred only half as often as the other types. The reason for this was that this question-type offered only two answer-options. In order to ensure that the probability of the answers *true* and *false* did not exceed all other answer-options, the proportion of *existence-questions* in general was reduced.

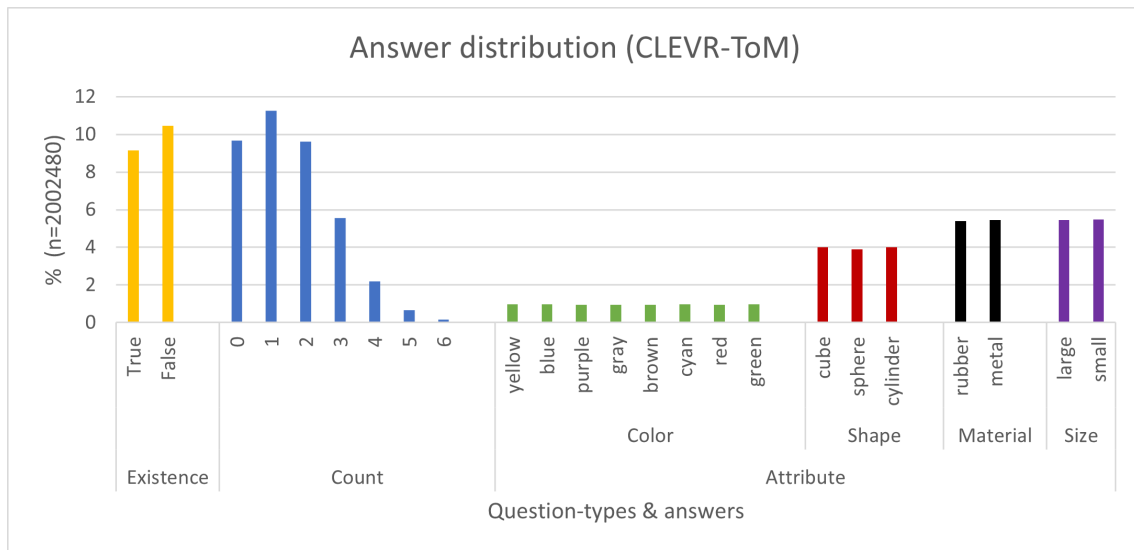


Figure 3.6: Bar chart showing the distribution of the different answer-options in the final CLEVR-ToM data set. The answer-options are shown bundled for each question-type in a uniform colour.

The equal use of *relational questions* and *non-relational questions* should also allow the model to think spatially and work with the image. The objects were also not named concretely in the *relational questions*, but only indirectly, which encouraged a proper interaction with the image. But as the figure 3.5 shows, the number of *relational questions* was significantly higher compared to the *non-relational questions* and were approximately in a 2:1 ratio at least for the *existence-* and *count-questions*. For the *attribute-question*, the ratio remained balanced. This ratio did not cause a problem, however, as there was no difference between the number of *relational questions* and *non-relational questions* for the *remove-action* and the *change-action* and the effect only existed because of the *swap-action* which always occurred with a *relational question*.

It was also important that the number of tasks for the true-belief was balanced with those for the false-belief case, so that the model could learn both types and not just focus on one. In this way, it would also learn to deal with the different sequences of actions, and learn the connections to the true- and false-beliefs while doing so. When considering only the order of action, there was an equal number of true- and false-beliefs in the final data set. However, by using *distractor-tasks* and looking at the actual belief of the agent and not just the sequence of actions, there were now more cases where the belief was true. To be more precise, 75% of the tasks represented a real true-belief case and 25% a false-belief case. By marking these *distractor-tasks*, however, the results for the normal tasks could be calculated, where 50% of the tasks still represented a real true- or false-belief. This made it easy to see whether the model understands and can deal with the difference between the two belief-types with their different answers.

For each of the question-types the answer-options were then also used as balanced as possible, e.g. the answers for the *existence-question* should come close to a 50:50 distribution for *true* or *false*. However, this was not an easy undertaking, since, for example, in the *count-question*, the higher numbers were significantly less likely than the lower ones. This was because the maximum number of total objects in the scene was eight, although the number could change by one caused by the

remove-action. The minimum number of objects was four. This means that in some images, even if the *count-question* targeted all objects, only four was the highest number that could be achieved. In a *relational question*, the number was even lower by at least one, because the *relational object* was not counted. Furthermore, most questions specify attributes for the objects searched for, which severely limited the number of possible objects. For this reason, only the numbers between zero and three were similarly highly probable and the higher numbers were less likely in the data set, as can be seen in the answer distribution in figure 3.6. However, this was not a major problem for the training, as the results in Section 5.4 show.

For the other question-types, the answer-options were balanced among themselves. For example, in the question about the colour of an object, there were a similar number of questions with the answer *red* as with the answer *cyan*.

Overall, there were differences. The low numbers for the *count-question* and the answers *true* and *false* for the *existence-question* were clearly more probable than the answers for the *attribute-questions*. However, as the results in Section 5.4 show, the differentiation between the various question-groups was not a problem for the model, which is why the balance between the answer-options of a question-group was clearly more important than a global balance.

Nevertheless, this difference should not become too extreme. As there are even several sub-types for the *attribute-question* in which each object attribute could be queried, the number of answer-options for this type was also significantly higher than for the other types. Since the *attribute-questions* were also only applicable in combination with the *change-action* and these, moreover, offer the greatest variation of possible answer-options, the proportion of these questions for the *change-action* was increased. $\frac{2}{3}$ of all questions were from this category for the *change-action*. $\frac{2}{9}$ were reserved for the *count-questions* and $\frac{1}{9}$ only for the *existence-questions*. Furthermore, the *change-actions* were not considered as one group with the same proportion as the *add-* and *remove-actions*. Instead, this group has been split into the different sub-groups for each of the attribute-types that the action changes. Each of these sub-groups got the same ratio as the other. This made it possible to get the ratio of *attribute-questions* to a similar level as that of the *count-questions*. However, since there were many answer-options for the attributes for each of the attribute-types, they were less likely to be answered in total than those for the existence and *count-questions*. There were also differences between the *attribute-questions*, as each attribute-type gets the same probability, but there were differences in the number of answer-options for each type. For example, for size and material there were only two options, but for the *attribute-colour-question* there were seven options. But as said before, even if there were these differences in this data set, they were still acceptable without affecting the training of the models too much (see Section 5.4).

4 User study

4.1 Structure and procedure

In addition to training & testing the data set with models, a user study was also conducted. This offered the opportunity to test the data set also on people, and thus on the original target group of the Sally-Anne test. Furthermore, it gave the opportunity to get feedback for possible changes in order to correct and improve the templates of the actions and questions or the structure of the tasks in general. An important concern was that the different answers for the true- and false-belief case are understood by the participants and that they can also understand the meaning of the different kinds of actions and questions. In this way, the templates for the actions and questions were also tested. Furthermore, it allowed to acquire a human baseline from the results of the user study that can be compared with the results of the model.

For the creation and execution of the user study, the application *LimeSurvey* [Lim12] was used because it allows an easy creation, sharing and conducting of the user study. To make sure that the results are not biased, a few prerequisites were defined in order to participate in this user study. The first requirement was that all participants have an understanding of English with a language level of at least B1. This was to ensure that all participants could understand the textual actions and questions correctly and that the results did not represent language comprehension. Furthermore, the minimum age was 18, which meant that no young children could take part in the study whose ToM ability was not yet fully developed. Another important restriction was that the participants should not be colour blind, as this would be a problem in recognising the colours of the objects, as this is an important feature of the CLEVR and the CLEVR-ToM tasks.

The structure of the final survey was then as follows:

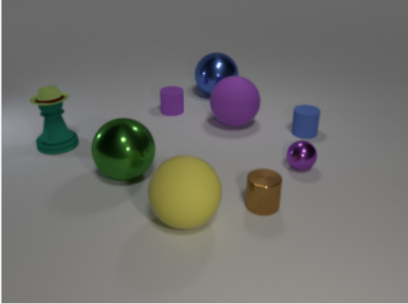
On the welcome page, before the actual start of the user study, the participants were informed about the conditions of participation. In addition, the task was roughly described in text form with the most important notes on how to work on the tasks, e.g. that the agent itself did not count in the *count-question* and did not represent an object in the image. Furthermore, an example task with the correct answer was given. The user study was anonymous and only recorded the age of the person, which had to be entered at the beginning of the user study. The user study was then conducted in two parts.

First, the understanding of the classic Sally-Anne test was examined. In this part, the participant was shown an image of the test procedure of the Sally-Anne test and asked to indicate where Sally would look for the ball. The answer had to be chosen from a list of two possible locations („basket“ or „box“).

The second part then consisted of examples from a previous version of the CLEVR-ToM data set, which did not contain the extension with the *distractor-tasks* discussed in Section 3.1.4. The individual tasks for this data set consisted of an image of the data set with the corresponding action.

Remove Action 1

Image:



Action: Remove the agent from the scene. Then take the blue shiny thing out of the scene.

*Question: How many spheres does the person expect have a big size and are behind the small metallic cylinder at the end?

📌 Your answer must be between 0 and 9
📌 Only an integer value may be entered in this field.

*How difficult was the previous task to solve on a scale from 1 (easy) to 5 (impossible)?

1 2 3 4 5

Figure 4.1: Example of a typical task from the user study. In this case, the task represents a false-belief for the *remove-action* and the *count-question*.

In the user study, the first question always represented a question from the data set and the second asked how difficult it felt for the participant to answer the question.

Below this, two questions were posed. Such an example can be seen in figure 4.1.

Each participant was given a task for each combination of action-type and question-type. In this case, the *change-action* was only counted as one action and not split according to the different attribute-types. Thus, in addition to the Sally-Anne test task, each participant had to answer nine tasks from the CLEVR-ToM data set. It was randomised whether each task represented a true- or false-belief and how the question-types were combined with the tasks. This ensured that the results for individual question-types and belief-types were not dependent on the image.

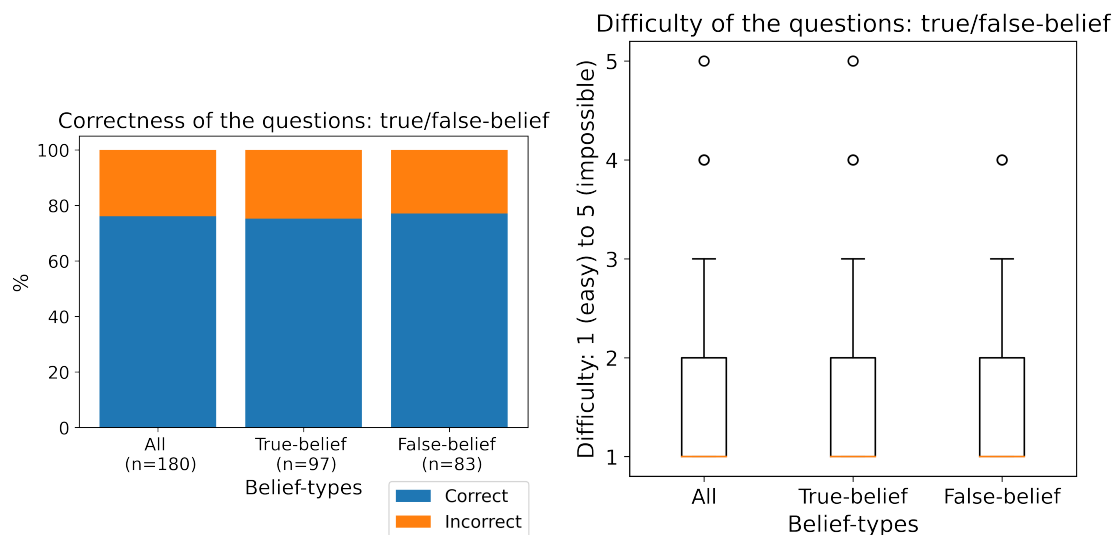
The first question was a question from the data set that belonged to the task (for which the image and action were already present) and had to be answered correctly. All answer-options of the specific question-type were available as possible answers for this question, e.g. all possible colours for the question about the colour of an object. The answer is given either by a number field, where only

integers between zero and nine can be entered for the *count-question*, by a choice of „true“ and „false“ for the *existence-question* and by a list of attributes for the *attribute-question*.

The second question, on the other hand, only targeted the perceived difficulty of the previous question for the participant. Here, the participant should enter the difficulty on a scale from one for „easy“ to five for „impossible to answer“. This helped to get a sense of the perceived difficulty of the data set, as it should not be too difficult for adults to solve ToM tests.

A time of about 10 minutes was planned for conducting the user study, although there was no time limit when answering.

4.2 Evaluation



- (a) Shows the percentage of correct and incorrect answers for the different belief-types in the form of bar charts. The number of correct answers is shown in blue and the number of incorrect answers is shown in orange colour.
- (b) Shows the perceived difficulty in solving the tasks as boxplots. A scale from one (easy) to five (impossible) was used for the difficulty. In this chart, the boxes show the range between the lower quartile and the upper quartile, i.e. the range between 25% and 75% of the data points. The orange line in the box marks the value of the median. The whiskers show the normal range of the data points, with outliers marked as individual dots in the image.

Figure 4.2: Statistics of the tasks regarding the belief-types.

In total, 20 people participated with a complete answer set. This resulted in a total number of 180 solved tasks. Of these, 97 were a true-belief and 83 a false-belief task.

All participants successfully answered the first task with the Sally-Anne test. In the following, the results of the second part with the tasks from the data set will be discussed.

4 User study

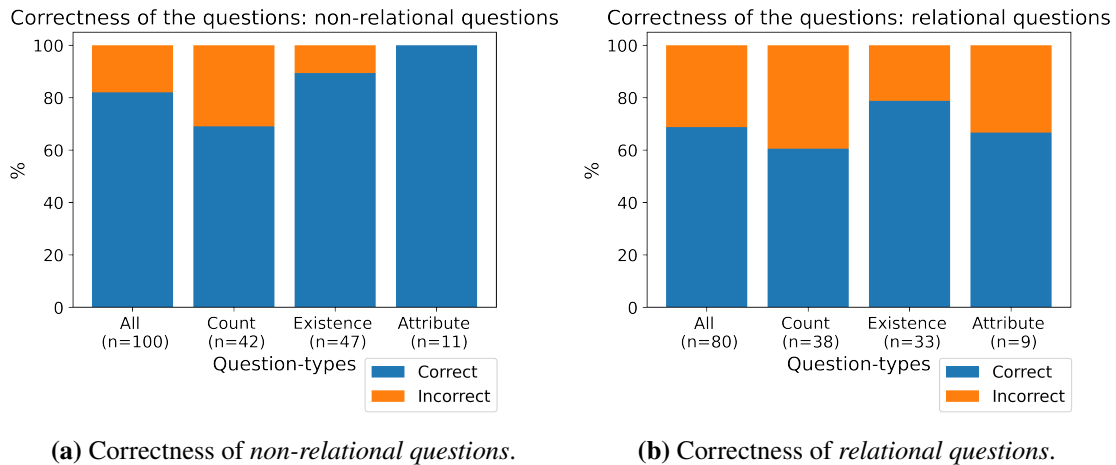


Figure 4.3: Correctness comparison of the *relational/non-relational* questions regarding the question-types. Shows the percentage of correct and incorrect answers for the different question-types in the form of bar charts. The number of correct answers is shown in blue and the number of incorrect answers is shown in orange colour.

As the bar chart in figure 4.2a shows, the number of correct answers for the true- and false-belief were very similar, which means that correctness did not depend on the belief-type. The measured difficulty was also very low on average for all tasks, as the box plot in figure 4.2b suggests. This shows that the tasks generally did not present any problems to the participants. Nevertheless, over 20% of the tasks were answered incorrectly and there was also a significant variation in the perceived difficulty of the tasks. This could be an indication that problems nevertheless existed with certain tasks. However, this was independent from the belief-type of the tasks.

The perceived difficulty and reasons for incorrect answers might be explained by the complexity of the question. The results of the two diagrams in figure 4.3 show that especially the *relational* questions posed a difficulty compared to the *non-relational* questions and that there were significantly more false answers for these.

The fact that *relational* questions are generally more difficult to answer than *non-relational* ones can be seen in the example from the user study in figure 4.1. In this example, we have a *relational-question*, because it limits the queried area with the help of a *relational object* and a direction. In order to solve this question, it was not enough to look at the whole image and search for the queried objects, but it must be checked for each of the objects that it is also in the specified relation to the *relational object*. In the example, all objects „behind the small metallic cylinder“ are in the area of interest and it is asked there for the number of big spheres. While for a *non-relational question* the answer is 4 because there are four large spheres in the scene, for the *relational question* it must be considered that the yellow sphere is not behind the *relational object* („small metallic cylinder“), so the answer is here 3. This means that with the *relational* questions, one more analytical step must be taken.

But that was not the only reason, as could be inferred from the feedback that came back from different people about the user study. In this feedback, it was often described that it was not clear from which perspective the *relational* questions had to be answered.

The relations in this data set are to be understood from the perspective of the image. This means that

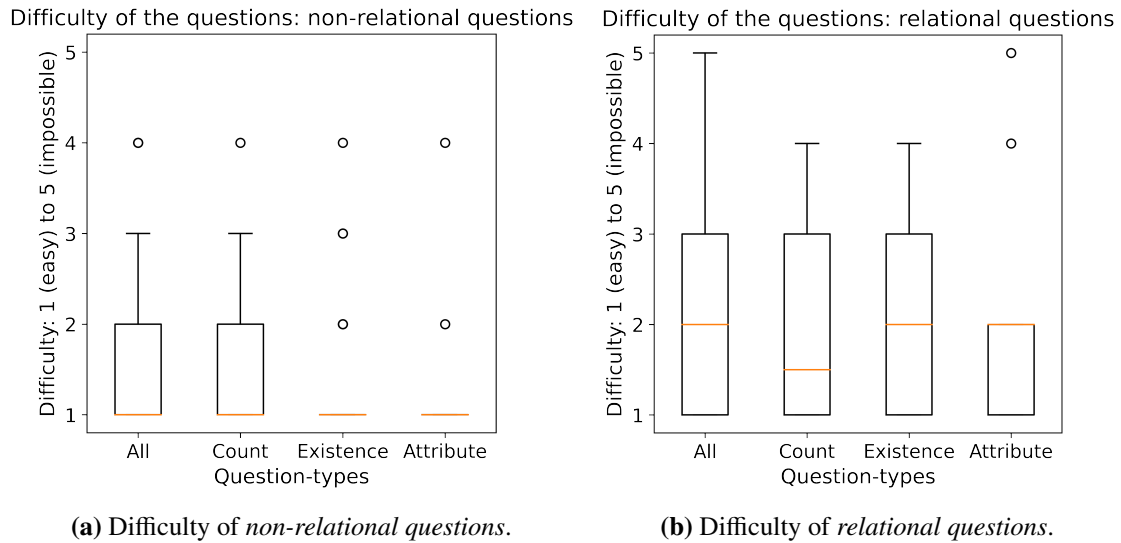


Figure 4.4: Difficulty comparison of the *relational/non-relational* questions regarding the question-types. Shows the perceived difficulty in solving the tasks as boxplots. A scale from one (easy) to five (impossible) was used for the difficulty. In this chart, the boxes show the range between the lower quartile and the upper quartile, i.e. the range between 25% and 75% of the data points. The orange line in the box marks the value of the median. The whiskers show the normal range of the data points, with outliers marked as individual dots in the image.

in the example in figure 4.1, the statement „behind the small metallic cylinder“ means all objects except the yellow sphere, since in the image only the yellow sphere is slightly in front of the brown cylinder. Another approach, however, would be to take the perspective of the agent, which some participants used in the user study. This expectation is something very normal in humans because we humans naturally take the perspective of another person and this is also a kind of ToM ability, which is also called „visual perspective-taking“ [APK+06]. In this case, the statement „behind the small metallic cylinder“ would not include, for example, the green sphere, since it is closer to the agent than the brown cylinder. However, this perspective is much more difficult to handle, since taking a perspective other than the one given from the image is more demanding. It is also not clear in which direction the agent is looking, as he has no face to indicate the direction of his gaze, and it is not clear whether in the example, for instance, the blue cylinder is in the relationship or not. This would then also explain the higher perceived difficulty in the *relational* questions from the comparison in figure 4.4.

When the tasks are further divided into correct and incorrect solutions, this difference in perceived difficulty is even more pronounced. This case can be seen in the box plots of figure 4.5. For the wrong answers, the perceived difficulty was significantly greater than for the correct answers, at least when looking at the individual question-types. This supports the assumption that the agent’s perspective was frequently used for the wrong answers.

However, this problem only affects humans because we have already acquired this ToM ability and it is natural for us. For a machine or model, however, it is not given, and it learns to use the perspective dictated by the correct answers. Since CLEVR and CLEVR_HYP also only used the image-perspective and these models had no problems with this, this use should also not affect the

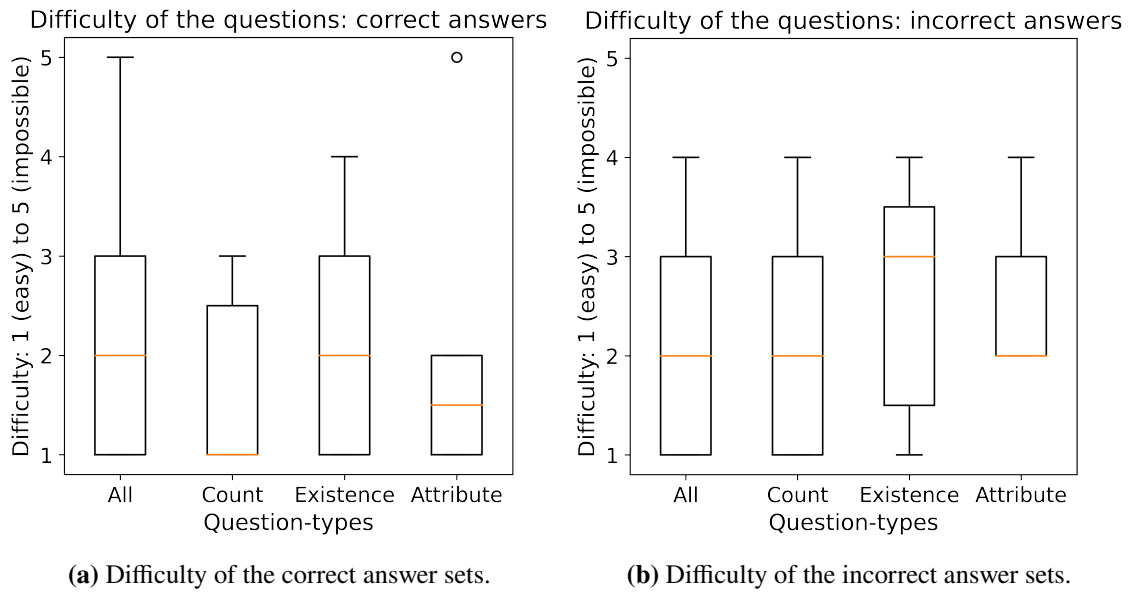


Figure 4.5: Difficulty comparison of the correct and incorrect answer sets for the *relational questions* regarding the question-types. Shows the perceived difficulty in solving the tasks as boxplots. A scale from one (easy) to five (impossible) was used for the difficulty. In this chart, the boxes show the range between the lower quartile and the upper quartile, i.e. the range between 25% and 75% of the data points, i.e. the range between 25% and 75% of the data points. The orange line in the box marks the value of the median. The whiskers show the normal range of the data points, with outliers marked as individual dots in the image.

CLEVR-ToM data set.

It would also have been possible to implement the data set for this other perspective (agent-perspective), but then several types of ToM would be implemented in the data set, and it would then be harder to get conclusions about the individual types of ToM. For this reason, this work was limited to the use of the false-belief test and the image-perspective.

In general, it can be observed that the *count-questions* led to the most incorrect answers (see figure 4.3). On the one hand, this was due to the larger number of answer-options, which made guessing much more difficult. Furthermore, there was also feedback that some attributes were more difficult to identify in the image. For example, it was difficult for some participants to see whether an object was made of „metal“ or „rubber“, or whether an object in the background was „large“ or „small“. This could also lead to these problems with the *count-questions*, as this made miscounting much more likely, as it was not always clear whether an object belonged to the set of objects in the question or not.

The same problem may also be responsible for the higher perceived difficulty in the *existence-questions*. In these questions, however, the limited number of possible answer-options made a correct guess more likely.

However, this should not be a problem for the data set and for the models, as the attributes came from the original CLEVR data set, and the image-generation had only been slightly modified

(see Section 3.2 for more on this). As the models coped very well with the original data set, this recognition seems to be a problem mainly for humans. As the data set was based on CLEVR, which used it as core elements, no adjustment was made to these attributes either, so they still match.

In general, the user study showed that the people understood the tasks (except for the partly unclear perspective) and could also deal with the templates and sentence constructions used and knew, for example, which change should be made to the scene. The user study also showed that participants were able to identify the different belief cases and took the necessary steps to answer the question, which was very important for the data set. In general, all problems encountered should not apply to the models, so that no changes to the structure of the tasks were necessary.

5 Models & Results

5.1 Relational Network (CNN+LSTM+RN)

In order to test the new data set it was important to find models that could handle the input consisting of image, action-text and question. The models should also be able to use the chronological order of the individual actions of the action-texts, since these influence the answer. For this reason, especially LSTMs [HS97] and transformers were considered for the processing of the action texts.

An interesting model for this work was the CNN+LSTM+RN model [SRB+17] that was created for the CLEVR data set and performed excellently for this.

The advantage of this model over previous models was the ability to handle *relational questions* and relationships between objects using a RN. Previous models had difficulty solving these tasks [JHV+17], but by identifying possible objects using a CNN and processing these discovered objects in pairs using the question, the model not only managed to recognise and correctly answer *relational questions* but did so with a comparatively very simple structure. Since the CLEVR-ToM data set presented here also used *relational questions* alongside *non-relational questions*, this model was a good choice for this data set.

An additional benefit was that the model used a LSTM for the textual input, which allowed the processing of long sentences and considered the spatial relations in the text. This also allowed a possible recognition of the sequence of actions and thus the temporal levels in relation to the actions. This was essential for the solution of this data set, as only by including the sequence of actions it was possible to recognise the true- or false-belief. In CLEVR and CLEVR_HYP, on the other hand, this ability was not necessary, as they had only used one temporal level.

Another advantage was the easy expandability of the model. This was used in this work to also create an extended form of the model, which was specifically designed for the data set with the added actions. This extension will be presented later in Section 5.2.

A simplified illustration of the whole model can be seen in figure 5.1, where in the original model CNN+LSTM+RN, without the second added LSTMs, only the image and the question serve as input. The action and the LSTM shown in green are not part of the original model.

As the name of the model CNN+LSTM+RN suggests, the model is composed of three parts, which will now be presented one by one.

CNN

The first part of the model is the CNN, which was responsible for processing the images. This CNN received an pixel based representation of an image as input and convolved it into k feature maps for the k different kernels. For this purpose, the CNN used four convolutional layers. The CNN was used in the model to identify different objects in the image. The „objects“ did not necessarily have

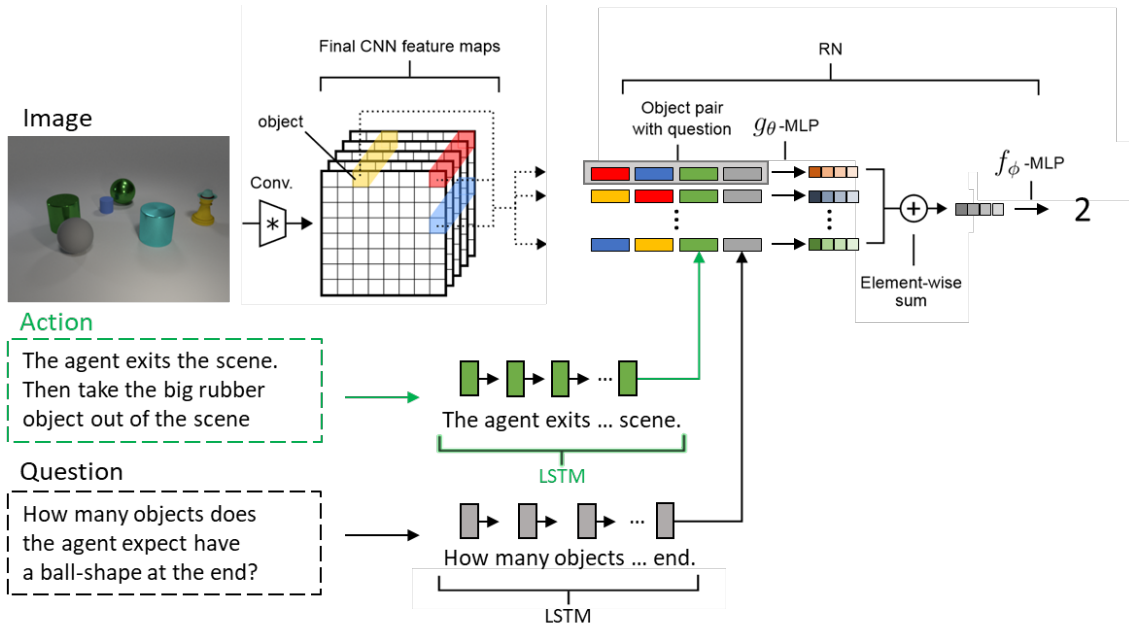


Figure 5.1: Architecture of the Extended Relational Network model (CNN+2LSTM+RN). The added input for the action-text is shown in green [SRB+17].

to be real objects, but generally conspicuous features. An object was described by a k -dimensional cell of the feature maps, as shown in the colours yellow, red and blue in the figure 5.1. In addition to these values, the relative spatial position in the image was also included for each of the individual objects. This served the purpose of later being able to determine a spatial relation between two objects. For example, whether one object was to the left of another.

LSTM

Besides the CNN, an LSTM was used, which was responsible for representing the textual question as question embedding, which served as input for the RN in addition to the objects from the CNN. The question embedding was very important for the final step, because the information about the relation and the queried object was contained in the question. Without the question, the combinations of the objects in the image would be meaningless, as it would not be clear whether this relation is relevant to the task or not. In order for the textual question to serve as input for the LSTM, the individual words in the text were replaced by unique numbers, which indexed a learnable look-up embedding. This gave the LSTM a single word embedding as input every time-step. The final state of the LSTM was then the question embedding.

Since in the CLEVR-ToM data set, in addition to the questions, an action-text was present in each task, which was important for answering the question, this had to be integrated into the model. A possibility without changing the model was to concatenate the action to the textual question. This was already done in CLEVR_HYP for the baseline models [SKYB21]. The input for the LSTM was then composed of action-text + question.

RN

The third part is the RN, which was the core of the model. The RN had the advantage that the architecture of the network was already designed to handle relationships. In this implementation, the RN used different combinations of the objects of the CNN and LSTM as input. In the illustration 5.1 the combination of the objects is shown as an example, where each colour represents an object from the CNN. In addition to the object pairs, the output of the LSTM for the question was also included, which was also handled as an object. This procedure had the advantage that relations from the questions are recognised and the relevant objects from the image can be taken into account. For example, if all red objects to the left of the green cylinder are searched for, combinations of a red object with a green cylinder that also lie in the correct relation could be given a higher weight for the later calculation.

The set of object combinations was then used as an input for the RN, which calculated the probabilities of the answers via a combination of two functions g, f : $r = f_\phi(\sum_{i,j} g_\theta(o_i, o_j, q))$. The function f took as input a sum over the different pairs of objects (o_i is here the i^{th} object) with the question embedding q from function g . Multi-layer perceptrons (MLPs) with learnable weights ϕ, θ were used for the functions f and g here. The output of the model r was then determined by a softmax layer that returns the probability over the different possible answer-options.

Configuration

The exact parameters used largely correspond to those of the original work. For example, four convolutional layers with 24 kernels, Rectified Linear Unit (ReLU) as the activation function, and batch normalisation each were used for the CNN. The LSTM for the question used 128 units and 32 unit word-lookup embedding. For the g-MLP, four layers with 256 units per layer with ReLU were applied. In contrast, for the f-MLP, only three layers were used, in which 50% dropout was applied in the second layer. The first two layers used 256 units and the third one 24 units with ReLU. A linear layer was chosen as the final layer. This layer returned logits over the answer vocabulary, which were normalised using softmax and optimised with a cross-entropy loss function using the Adam optimiser with a learning rate of $2.5e-4$.

The training took for one model around six days with an *NVIDIA GTX 1080* and a batch size of 64.

5.2 Extended Relational Network (CNN+2LSTM+RN)

Since the existing CNN+LSTM+RN model was only designed for CLEVR and not for CLEVR-ToM, the model only used two separate input channels for the image and the question. Since concatenating the action-text and the question and using the combination as the text input was only a workaround, extending the inputs to handle the actions separately was an interesting approach.

By adding another text input, it was possible to handle the question and the action separately. The advantage of this was that separating these two components would allow the model to deal with the different inputs in the LSTMs separately and thereby react more precisely to the different features. In addition, the separation of these components from the concatenated text by the model would be more effortful than if it had already been done. Furthermore, the concatenated text would be very

long, which is not advantageous for an LSTM.

Therefore, a new LSTM was added for the action-text, which corresponded to the one for the question input. The result of the LSTM, the action embedding, was then used as further input for the RN. The added parts of the extension of the model can be seen highlighted in green in figure 5.1. The output of the LSTM for the action, like that for the question, represented an object which served as input for the g-MLP model. More precisely, for each pair of objects extracted from the image by a CNN, in addition to the output of the LSTM for the question, that for the action-texts was also added and used together as an input to the RN. The resulting function was then: $r = f_{\phi}(\sum_{i,j} g_{\theta}(o_i, o_j, a, q))$, where the letter a stands for action embedding.

The configuration for the extended model was the same as for the original CNN+LSTM+RN model. For the additional LSTM for the action-text, the same values were used as for the LSTM for the questions.

In addition to this extended model CNN+2LSTM+RN, a purely textual version of the model was also trained, which corresponded to this model, but in which only the actions and questions served as input and the image with the CNN was not taken into account.

5.3 Rule-based models

For a better comparison of the results, it was worth looking at the results of different rule-based baselines. In contrast to CLEVR or CLEVR_HYP, the CLEVR-ToM data set was not a 27-class classification but a 24-class classification problem, since only the values from 0 to 6 were possible for the *count-question*. The model could choose from all possible answer-options for each question, which is why it was also possible for a model to give a nonsensical answer to a question, e.g. the answer 5 for the question about the colour. A model that always returns a random answer would therefore have an accuracy of $\frac{1}{24}$, i.e. about 4%. Since the distribution of the answers was not uniform, a distinction was made between four different variants of rule-based models.

The first variant only outputted a constant answer for all questions, which was the answer with the highest overall probability in the data set. As can be seen from the answer distribution in figure 3.6, the most frequent answer was the number 1, which the model returned on any question. This model is referred to as *Rule-constant*.

The second variant with the name *Rule-question* distinguished between the different question-types and *relational* and *non-relational questions* and outputted a constant answer for each of them, i.e. the model did not output any nonsensical answers here and outputted the most probable answer for each question-type, e.g. *cylinder* for the *relational attribute-shape-question*.

Furthermore, there was a variant that in addition to the different questions-types also distinguished between the different types of beliefs and always gave the best constant answer for each part. This made it possible to test whether there were biases and irregularities between the belief-types. This variant is listed in the work under the name *Rule-belief*.

The last and fourth variant called *Rule-extension* extended the third *Rule-belief* variant by the distinction between the *normal-* and the *distractor-actions*. Thus, this model had the most flexibility of the rule-based models and should also achieve the highest accuracy values among them.

5.4 Results

The CLEVR-ToM data set was divided into three parts. 70% of the data was intended for training. The remaining 30% was shared equally between the validation- and test-set. The results of the models in this section are based on the test-set from the CLEVR-ToM data set except for the human baseline.

Model	existence	count	colour	shape	material	size	Total
Rule-constant	0.000	0.286	0.000	0.000	0.000	0.000	0.112
Rule-question	0.532	0.286	0.139	0.338	0.502	0.501	0.376
Rule-belief	0.572	0.286	0.139	0.341	0.506	0.504	0.385
Rule-extension	0.598	0.287	0.141	0.345	0.508	0.505	0.392
Human baseline*	0.850	0.605	1.000	1.000	-	0.666	0.761
CNN+LSTM+RN	0.985	0.931	0.977	0.983	0.994	0.993	0.965
CNN+2LSTM+RN	0.990	0.952	0.986	0.989	0.996	0.995	0.976
*-textual version	0.689	0.436	0.355	0.584	0.784	0.784	0.573

Table 5.1: Accuracy values of different models on the test-set of the CLEVR-ToM data set for the different question-types.

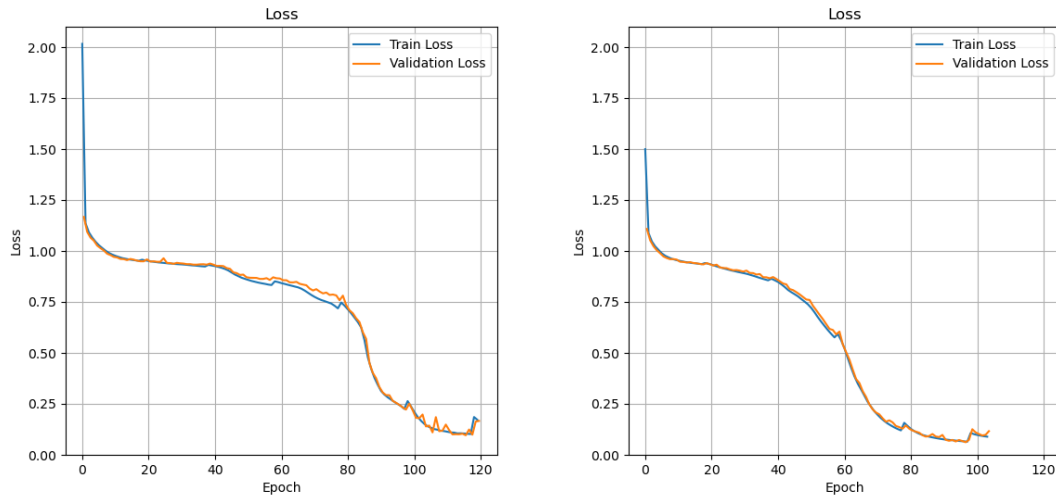
*The human baseline results, which correspond to the results of the user study, are included here as a model for completeness, although the results are based on a previous version of the data set.

Model	true-belief	false-belief	non-relational	relational	Total
Rule-constant	0.112	0.111	0.092	0.125	0.112
Rule-question	0.389	0.364	0.371	0.380	0.376
Rule-belief	0.390	0.380	0.391	0.381	0.385
Rule-extension	0.392	0.391	0.394	0.390	0.392
Human baseline*	0.753	0.771	0.820	0.688	0.761
CNN+LSTM+RN	0.965 (0.968)	0.965 (0.967)	0.972	0.961	0.965
CNN+2LSTM+RN	0.974 (0.972)	0.978 (0.978)	0.981	0.973	0.976
*-textual version	0.609 (0.752)	0.537 (0.677)	0.611	0.548	0.573

Table 5.2: Accuracy values of different models on the test-set of the CLEVR-ToM data set for the belief-types and the *relational* and *non-relational* questions.

The values of the *normal-tasks* for the two belief-types are shown in brackets.

*The human baseline results, which correspond to the results of the user study, are included here as a model for completeness, although the results are based on a previous version of the data set.



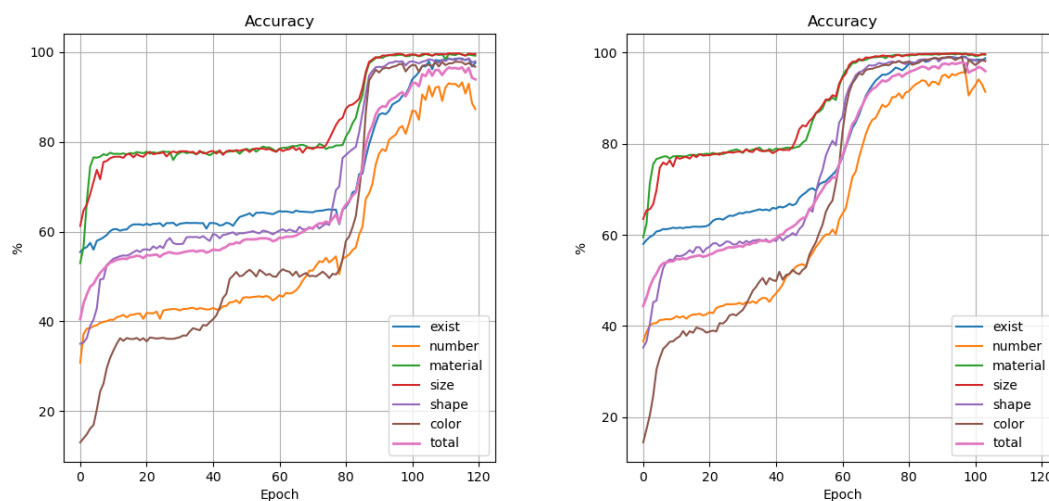
(a) Loss-graph of the Relational Network (CNN+LSTM+RN) model.

(b) Loss-graph of the Extended Relational Network (CNN+2LSTM+RN) model.

Figure 5.2: Comparison of the loss-graphs of the original Relational Network (CNN+LSTM+RN) model with the Extended Relational Network (CNN+2LSTM+RN) model. These show the progression of the training- and validation-loss over the course of the epochs.

The results in table 5.1 show that the *Rule-constant* model delivered the lowest accuracy, as expected. This was because with the constant answer *I* it only had a chance of achieving a correct answer for the *count-questions*. The difference between the other rule-based models was quite small. This shows that the answer distributions between the individual belief-types and the *normal-* and *distractor-tasks* in CLEVR-ToM were very similar. Even if a model could recognise the *distractor-tasks*, which was not the case due to many adjustments (see chapter Section 3.1.4), it would not be possible for the model to achieve high accuracy values. However, the accuracy depended strongly on the number of possible answer-options. For example, the accuracy values for the *existence-question* was 60% and for the *attribute-colour-question* 14%. Since there were only two answer-options for the *existence-question* and eight for the *attribute-colour-question*, these results were only slightly above chance.

A similar behaviour can be seen in the second table 5.2, in which accuracy values for further categories are listed. In this table, the values for the different belief-types and between *relational* and *non-relational questions* hardly differ for the different rule-based models. The models were also unable to achieve high accuracy values in any of the categories. This indicates that there were only minor differences in the distribution of answers in these categories and that high values for accuracy could not be achieved without taking into account the additional information from the image and text. It is important to know that *distractor-tasks* in which the agent initially leaves the scene were also counted as false-belief in this case. Even if the question with the answer itself was a true-belief, the sequence of actions was that of a false-belief.



(a) Accuracy-graph of the Relational Network (CNN+LSTM+RN) model.

(b) Accuracy-graph of the Extended Relational Network (CNN+2LSTM+RN) model.

Figure 5.3: Comparison of the accuracy-graphs of the original Relational Network (CNN+LSTM+RN) model with the Extended Relational Network (CNN+2LSTM+RN) model. These show the progression of the validation accuracy for different categories over the course of the epochs.

In contrast to the rule-based models, the CNN+LSTM+RN model achieved significantly better accuracy values and, with the exception of the *count-question* category, managed to achieve accuracy values above 96% in all other categories. But even in the *count-question* category it still achieved values of over 93%.

The extended model CNN+2LSTM+RN managed to increase the values of the original model even further. In general, the accuracy results of the extended model were slightly more than one percent higher than those of the original model. The difference was most noticeable in the *count-questions* where the accuracy values differ by more than two percent. Furthermore, it can be seen in the accuracy-graph in figure 5.3 that the accuracy increased significantly faster over the epochs. This may be due to the fact that this modification took work off the model by separating the different types of inputs. This allowed it to more quickly adapt the treatment of the two inputs in a targeted way and to address differences in the treatment. The advantage of faster learning can also be seen in the loss-graphs in figure 5.2. In both graphs it is visible that the learning progress for the original model only increased rapidly at epoch 80. In contrast, the process in the extended model already took place at epoch 40, although this increase was not as steep as in the original model.

The loss graphs 5.2 also shows that sometimes the validation loss was lower than the training loss. This may be due to the regularisation through the used dropout, which is only applied during training.

Figure 5.4 shows a confusion matrix which indicates how well the model predicts individual answer values. It is noticeable that the model coped very well with all values and even predicted the higher numerical values in the *count-question* in most cases correctly. Nevertheless, the higher numbers had a higher probability that the model returned a wrong answer in comparison to the

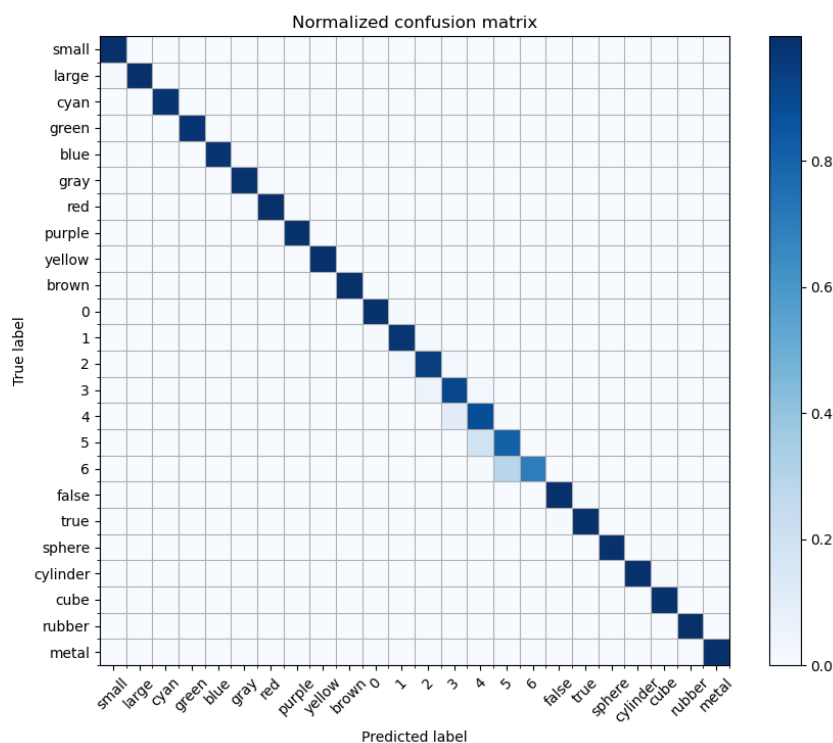


Figure 5.4: Confusion-matrix shows the probabilities of the combinations of predicted labels and true labels for each response option. The values are taken from the results of the CNN+2LSTM+RN model on the test set after 92 epochs of training.

other answer-options. This was not surprising given the fact that these values made up only a very small part of the *count-question* and therefore rarely occurred in the data set, as can be seen in the figure 3.6 showing the answer distribution. However, the difference to the correct value was normally not more than one unit.

The accuracy values in table 5.1 show almost perfect results for the extended model CNN+2LSTM+RN. Even though there were eight different answer-options for the *attribute-colour-question* and the different rule-based models did not achieve more than 15%, the extended model managed to reach results around 99%. The model only scored values below 98% for the *count-question* when considering the different question-types. But it still achieved more than 95% accuracy there. Like the rule-based models, there were hardly any differences between the results of the extended model for the different belief-types and *relational* and *non-relational questions*, as can be seen in table 5.2. There were also no differences in the belief-types, considering only the *normal-tasks* (values are in brackets in the table). Compared to the results from the user study, the results were also significantly better in most cases.

The textual model, which only used the textual action and question as input and not the image, managed to clearly beat the rule-based models for accuracy and to score particularly well with the *attribute-questions*. Nevertheless, the textual model did not reach by far the accuracy values of the extended model.

6 Discussion

The results from Section 5.4 show that the CNN+LSTM+RN model can deal well with this CLEVR-ToM data set, especially the extended model CNN+2LSTM+RN. In particular, the model managed to distinguish between true- and false-believe cases (see table 5.2) with the different answers. In the following, the results and differences to the other models and works are discussed in more detail.

6.0.1 Comparison with human baseline

If the results of the CNN+2LSTM+RN model are compared with the results of the user study (see tables 5.1 and 5.2), it is easy to see that the accuracy values of the model were higher in most areas. Similar to the user study, the proportion of incorrect answers was highest for the count question. This may be due to the many answer-options and the simple miscounting of one. While the participants in the user study mainly had problems with recognising the size of objects in the *attribute-questions*, this phenomenon did not occur with the model. This is because the model had also performed well in CLEVR and thus could also recognise the attributes in this data set properly. It can be concluded that even if objects were further in the background and thus appeared smaller, the model could still reliably distinguish the size of the object, whereas this was more difficult for humans.

The general difficulty with *relational questions* was not as pronounced in the model. The accuracy values differed only marginally here, which was also since the model generally delivered almost perfect values. This is because the model managed to learn the correct perspective for the *relational questions* that were needed to solve the tasks. As in the user study, the values between the true- and false-belief were almost identical, which indicated a good recognition of the belief-types. This shows that the problems that people had in solving the tasks (see Section 4.2) did not have any effect on the performance of the model, as expected.

6.0.2 Importance of image information

When comparing the results of the CNN+2LSTM+RN model with the textual version of the model, it becomes clear that the tasks could not be solved unambiguously without the information of the image. The results of both models can again be seen in the tables 5.1 and 5.2. The fact that the results of the textual model were nevertheless significantly higher than those of the rule-based models was due not only to the higher information content of the text, but also to the problem that the correct answers were partly contained in the action or could be derived from it. This problem was presented and discussed in more detail in chapter Section 3.1.4, where *distractor-tasks* were also introduced as a solution. Since the results of the CNN+2LSTM+RN model achieved very high accuracy values compared to the textual model, it also shows that the model could differentiate

between the *distractor-* and the *normal-tasks* with the image information. Consequently, the model almost always had to work with the image to recognise whether an action was relevant to the question or not. Therefore, the image had to be considered for answering the question, making this data set really a collection of VQA tasks.

6.0.3 Comparison with CLEVR & CLEVR_HYP

In general, it can be observed in the tables 5.1 and 5.2 that the CNN+2LSTM+RN model achieved very high accuracy values in all areas. This is especially surprising since the results of CLEVR_HYP, which only made changes to the image scene and asked questions about this modified scene, achieved significantly lower accuracy values for the different models tested. This could be due to the choice of models, but also to the simplified structure of the tasks in the CLEVR-ToM data set.

Simplifications and improvements of the CLEVR-ToM data set

In comparison with the CLEVR_HYP data set, no relations were used in the actions, and these were only used in the questions. Adding these relations to the actions might significantly increase the complexity of the task but was not necessary for the purpose of testing for a false-belief test.

Another simplification in the CLEVR-ToM data set was that actions only change one object, as is the case with the classic Sally-Anne test. In the CLEVR_HYP data set, it was possible for actions to affect multiple objects and make changes to all of them. Overall, in the CLEVR_HYP data set there were also more action- and question-types, such as moving one object on top of another as an action or comparing attributes or numbers in a question.

In general, the images and the generation of them had been improved compared with CLEVR and CLEVR_HYP, e.g., the test for occlusion of objects now also took the object size into account. Before, the same visible area of an object was always taken as a measure, but since this means that the percentage of the area was significantly smaller for large objects and a certain percentage was necessary for a correct recognition of the shape, such a constant was a drawback.

Furthermore, when selecting the questions for the *relational questions*, a higher significance of the relation was required. This means that the objects referring to a *relational question* must lie more clearly in the relation or clearly in the inverse relation. More precisely, no object should be at a similar level to the *relational object* and make it difficult to assign them to the relation.

This may also explain why the results of the model on this data set were also slightly better than those on the original CLEVR data set. There, the CNN+LSTM+RN model achieved an accuracy of 95.5%.

6.0.4 Limitations

As the results show, the CNN+2LSTM+RN model could distinguish between the two belief-types and managed to predict the correct answer in almost all cases. But it must also be considered that this work was an implementation of a rather simple test. By looking at the sequence of actions, a potential false-belief of the agent could be quickly identified. It only must be determined whether the agent leaves the scene as the first action or in the second action. However, this was a general

problem of this test, which also affected the textual implementations in the works of Grant et al. [GNG17] and Nematzadeh et al. [NBG+18]. In these works, it was also possible to recognise the belief-type from the sequence or the type of actions. However, as this problem originated in the original Sally-Anne test, these works showed that it is possible to solve a well-known false-belief test with AI models.

6.0.5 Task design

In comparison with other implementations of false-belief tests, the special feature of this data set was the combination of image and text and the use of both information to solve the tasks. This way it was much closer to the original Sally-Anne test and more in line with human thinking, which uses a combination of both senses (visual and natural language).

Furthermore, this data set also showed that it was possible to implement different time levels with CLEVR and that these were also understood by a model. CLEVR_HYP only modified the scene with the actions and queried this modified scene with questions. With the original CLEVR data set, the questions targeted the scene of the image directly. In this new CLEVR-ToM data set, however, both points in time were considered when answering the questions. In a false-belief case, the state of the scene from the image was needed as the source of information for the correct answer, but in a true-belief case, it was the scene modified by the actions. There were even three different time points in this data set, but one of them only differed in the removal of the agent. These very high accuracy values for the results show that the model could deal with these different time points and states.

6.0.6 Comparison with other false-belief data sets

Even if no direct comparison between different approaches is possible, it is also worthwhile to look at the other comparable related work, e.g. the textual implementations of the Sally-Anne test.

Compared with the data set presented by Grant et al. [GNG17], the results from this work with the CNN+2LSTM+RN model were significantly better. In the work from Grant et al. [GNG17], it was only possible to achieve higher accuracy values if the model received information in the tasks about whether an action was visible to a certain person, which allowed it to consider only the relevant actions. But even with this information, the accuracy values were lower than in this work. In contrast, in the work of Nematzadeh et al. [NBG+18] it was possible to achieve an accuracy of 100% in the presented simplified data set (ToM-easy). However, the data set also contained information on whether individual actions were visible to a specific person.

In general, the model achieved excellent results despite the combination of different inputs and thus more complicated processing for a model. The fact that the results were not the same between the various models and approaches is nothing uncommon and can even be related to humans, where the differences in the degree of the ToM ability also differ [HJH+05].

6.1 Future Work

These good results of the extended model CNN+2LSTM+RN on the CLEVR-ToM data set also give confidence that it is possible to integrate further actions and points in time into CLEVR and thus represent more complex processes and sequences. While the task used in this data set was still relatively simple, it could now also be implemented in a more complicated form based on the positive results of this work. In this case, further actions can be carried out one after the other in addition to the two actions. In this way, the agent could not be in the scene at first and then enter and leave the scene in later actions, thus further increasing the variance of the tasks.

In addition to extensions of the false-belief tests through further actions, it would also be possible to represent a „second-order false-belief test“ with this data set as a basis.

Furthermore, the test can be extended to the ToM ability „visual perspective-taking“ [APK+06] by integrating the perspective from the agent’s point of view into this test or by creating a new test. The CLEVR data set would also be well suited as a basis for this because of the three-dimensional scene. With CLEVR as a basis, a similar test has already been presented in the data set called *CLEVR-MRT* [BWG+23], which required a different perspective to solve the relational question. However, the perspective was given with the help of coordinates, so the model did not have to derive the correct perspective from the image. For this reason, it was not a ToM test. But if the agent from our data set is given a concrete direction of gaze, for example through indicated eyes, a real ToM test could be created based on CLEVR-ToM.

7 Conclusion

In this thesis, the new data set CLEVR-ToM was presented, which for the first time implemented a false-belief test similar to the Sally-Anne test [BLF85], as a VQA task. This closed the gap between the purely textual data sets and those that only use images. The implementation as VQA tasks had the advantage of being closer to the original Sally-Anne test, which in most cases also used images and text/language and thus was used in a VQA fashion [BLF85]. Furthermore, with visual information (image) and natural language (textual actions and questions), it addressed two important human senses for the ToM ability [MHB98].

The CLEVR-ToM data set also managed to combine several types of false-belief and deliver a wide variation of tasks. Besides the location false-belief as in the Sally-Anne test, an existence and an attribute false-belief were integrated and tested with this data set. By using CLEVR as a basis, this data set could be generated completely synthetically without the need for human interaction.

In addition to the first implementation of false-belief tests as VQA tasks, several different time levels were integrated into CLEVR for the first time, which only made the implementation of such a false-belief test possible.

The type of tasks of the data set were also tested in a user study for suitability and no serious problems for the later application in the models were found and an average relatively low difficulty to solve the tasks for people was determined.

The results of the CNN+LSTM+RN model show that it is capable of handling this type of task and had no problems distinguishing between the true- and false-beliefs and drawing the correct conclusions to answer the question. Furthermore, this work introduced an extension to the model called CNN+2LSTM+RN, which was better suited to the data set and allowed the separation of the action-text from the question in the input. This had further improved the very good results of the model and was an important technical decision to achieve the best results on this data set.

Since the model was able to achieve these high accuracy values, it also showed that the model could handle false-belief tests and multiple time levels and that the CLEVR data set was a suitable basis for the CLEVR-ToM data set. This means that even more challenging tests for ToM-types could be implemented in later work.

Bibliography

- [AAL+15] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C.L. Zitnick, D. Parikh. “Vqa: Visual question answering”. In: *Proceedings of the IEEE international conference on computer vision*. 2015, pp. 2425–2433 (cit. on pp. 16, 32).
- [ALS+19] A. R. Akula, C. Liu, S. Saba-Sadiya, H. Lu, S. Todorovic, J. Y. Chai, S.-C. Zhu. “X-tom: Explaining with theory-of-mind for gaining justified human trust”. In: *arXiv preprint arXiv:1909.06907* (2019) (cit. on p. 13).
- [APK+06] M. Aichhorn, J. Perner, M. Kronbichler, W. Staffen, G. Ladurner. “Do visual perspective tasks need theory of mind?” In: *Neuroimage* 30.3 (2006), pp. 1059–1068 (cit. on pp. 43, 58).
- [Bar01] S. Baron-Cohen. “Theory of mind in normal development and autism”. In: *Prisme* 34.1 (2001), pp. 74–183 (cit. on p. 13).
- [BLF85] S. Baron-Cohen, A. M. Leslie, U. Frith. “Does the autistic child have a “theory of mind”?” In: *Cognition* 21.1 (1985), pp. 37–46 (cit. on pp. 13, 16, 19, 22, 59).
- [BLGB20] C. Beaudoin, É. Leblanc, C. Gagner, M. H. Beauchamp. “Systematic review and inventory of theory of mind measures for young children”. In: *Frontiers in psychology* 10 (2020), p. 2905 (cit. on p. 13).
- [BWG+23] C. Beckham, M. Weiss, F. Golemo, S. Honari, D. Nowrouzezahrai, C. Pal. “Visual question answering from another perspective: CLEVR mental rotation tests”. In: *Pattern Recognition* 136 (2023), p. 109209 (cit. on p. 58).
- [Com18] B. O. Community. *Blender - a 3D modelling and rendering package*. Blender Foundation. Stichting Blender Foundation, Amsterdam, 2018. URL: <http://www.blender.org> (cit. on p. 33).
- [DAGH18] R. A. Dore, S. J. Amend, R. M. Golinkoff, K. Hirsh-Pasek. “Theory of mind: A hidden factor in reading comprehension?” In: *Educational Psychology Review* 30.3 (2018), pp. 1067–1089 (cit. on p. 13).
- [EVT16] B. Eysenbach, C. Vondrick, A. Torralba. “Who is mistaken?” In: *arXiv preprint arXiv:1612.01175* (2016) (cit. on pp. 13, 16).
- [GNG17] E. Grant, A. Nematzadeh, T. L. Griffiths. “How Can Memory-Augmented Neural Networks Pass a False-Belief Task?” In: *CogSci*. 2017 (cit. on pp. 13, 15, 57).
- [HJH+05] C. Hughes, S. R. Jaffee, F. Happé, A. Taylor, A. Caspi, T. E. Moffitt. “Origins of individual differences in theory of mind: From nature to nurture?” In: *Child development* 76.2 (2005), pp. 356–370 (cit. on p. 57).
- [HS97] S. Hochreiter, J. Schmidhuber. “Long short-term memory”. In: *Neural computation* 9.8 (1997), pp. 1735–1780 (cit. on pp. 18, 47).

Bibliography

- [Jar19] J. Jara-Ettinger. “Theory of mind as inverse reinforcement learning”. In: *Current Opinion in Behavioral Sciences* 29 (2019), pp. 105–110 (cit. on p. 13).
- [JHV+17] J. Johnson, B. Hariharan, L. Van Der Maaten, L. Fei-Fei, C. Lawrence Zitnick, R. Girshick. “Clevr: A diagnostic dataset for compositional language and elementary visual reasoning”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 2901–2910 (cit. on pp. 14, 17–19, 47).
- [LAM+20] A. Labash, J. Aru, T. Matiisen, A. Tampuu, R. Vicente. “Perspective taking in deep reinforcement learning agents”. In: *Frontiers in Computational Neuroscience* 14 (2020), p. 69 (cit. on p. 16).
- [Lim12] LimeSurvey Project Team / Carsten Schmitz. *LimeSurvey: An Open Source survey tool*. LimeSurvey Project. Hamburg, Germany, 2012. URL: <http://www.limesurvey.org> (cit. on p. 39).
- [Mal22] B. Malle. *Theory of mind*. In R. Biswas-Diener & E. Diener (Eds), *Noba textbook series: Psychology*. Champaign. <http://noba.to/a8wpytg3>. Accessed: 2022-11-22. 2022 (cit. on p. 20).
- [MHB98] M. Minter, R. P. Hobson, M. Bishop. “Congenital visual impairment and ‘theory of mind’”. In: *British Journal of Developmental Psychology* 16.2 (1998), pp. 183–196 (cit. on pp. 13, 16, 59).
- [NBG+18] A. Nematzadeh, K. Burns, E. Grant, A. Gopnik, T. L. Griffiths. “Evaluating theory of mind in question answering”. In: *arXiv preprint arXiv:1808.09352* (2018) (cit. on pp. 15, 57).
- [NSH16] H. Noh, P. H. Seo, B. Han. “Image question answering using convolutional neural network with dynamic parameter prediction”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 30–38 (cit. on p. 18).
- [PSD+18] E. Perez, F. Strub, H. De Vries, V. Dumoulin, A. Courville. “Film: Visual reasoning with a general conditioning layer”. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 32. 1. 2018 (cit. on pp. 14, 18).
- [RKZ15] M. Ren, R. Kiros, R. Zemel. “Exploring models and data for image question answering”. In: *Advances in neural information processing systems* 28 (2015) (cit. on pp. 17, 32).
- [RPS+18] N. Rabinowitz, F. Perbet, F. Song, C. Zhang, S. A. Eslami, M. Botvinick. “Machine theory of mind”. In: *International conference on machine learning*. PMLR. 2018, pp. 4218–4227 (cit. on pp. 13, 15, 16).
- [SKYB21] S. K. Sapat, A. Kumar, Y. Yang, C. Baral. “CLEVR_HYP: A challenge dataset and baselines for visual question answering with hypothetical actions over images”. In: *arXiv preprint arXiv:2104.05981* (2021) (cit. on pp. 14, 17, 22, 32, 48).
- [SRB+17] A. Santoro, D. Raposo, D. G. Barrett, M. Malinowski, R. Pascanu, P. Battaglia, T. Lillicrap. “A simple neural network module for relational reasoning”. In: *Advances in neural information processing systems* 30 (2017) (cit. on pp. 14, 18, 47, 48).
- [SSG20] J. Smogorzewska, G. Szumski, P. Grygiel. “Theory of mind goes to school: Does educational environment influence the development of theory of mind in middle childhood?” In: *Plos One* 15.8 (2020), e0237524 (cit. on p. 13).

- [TB19] H. Tan, M. Bansal. “Lxmert: Learning cross-modality encoder representations from transformers”. In: *arXiv preprint arXiv:1908.07490* (2019) (cit. on p. 18).
- [Whi93] A. Whiten. “Evolving a theory of mind: the nature of non-verbal mentalism in other primates”. In: *Understanding other minds: Perspectives from autism* (1993), pp. 367–396 (cit. on p. 13).
- [Win18] A. F. Winfield. “Experiments in artificial theory of mind: From safety to story-telling”. In: *Frontiers in Robotics and AI* 5 (2018), p. 75 (cit. on p. 13).
- [WP83] H. Wimmer, J. Perner. “Beliefs about beliefs: Representation and constraining function of wrong beliefs in young children’s understanding of deception”. In: *Cognition* 13.1 (1983), pp. 103–128 (cit. on p. 13).
- [WTW+17] Q. Wu, D. Teney, P. Wang, C. Shen, A. Dick, A. Van Den Hengel. “Visual question answering: A survey of methods and datasets”. In: *Computer Vision and Image Understanding* 163 (2017), pp. 21–40 (cit. on p. 13).

All links were last followed on July 12, 2023.

Declaration

I hereby declare that the work presented in this thesis is entirely my own and that I did not use any other sources and references than the listed ones. I have marked all direct or indirect statements from other sources contained therein as quotations. Neither this work nor significant parts of it were part of another examination procedure. I have not published this work in whole or in part before. The electronic copy is consistent with all submitted copies.

place, date, signature