Solving Diagrammatic Reasoning Problems Using Deep Learning

Himanshu Choudhary[1], Debi Prosad Dogra[1], and Arif Ahmed Sekh[2](✉)

[1] Indian Institute of Technology, Bhubaneswar, India
{hc11,dpdogra}@iitbbs.ac.in
[2] XIM University, Bhubaneswar, India
skarifahmed@gmail.com

Abstract. Diagrammatic Reasoning (DR) questions are very common in competitive examinations. However, construction of interesting and fresh DR questions can be a tedious job even for the experts. We explore the possibility of using Artificial Intelligence (AI) and computer vision (CV) for construction and solving DR problems. In this paper, we have proposed a new deep learning-based framework that can be used to solve certain types of DR problems. The research also shows that a similar framework can be used to generate new DR problems of similar characteristics. We formulate the DR problem with an extension of conventional 4×1 Raven's Progressive Matrix (RPM) by keeping 4 outputs. Thus, each problem sample has eight images, where the first four images are part of the input in a sequence and the last four images are options for the correct output. The first four images create a valid sequence and the target is to choose the fifth image from the next four images. To find the correct option, we have proposed a deep learning framework that consists of an LSTM, an Encoder and a fully connected classifier unit. The framework has also been used to generate new DR problems. We have tested our framework on Rotational DR problems. A new DR dataset has been generated using automated scripts to train the framework. The framework performs better as compared to SOTA deep learning frameworks.

Keywords: Diagrammatic Reasoning · Raven's Progressive Matrix · LSTM · Encoder · Image Analysis

1 Introduction

Diagrammatic reasoning (DR) problems are well known. A DR problem consists of a sequence of images with some logical relation in between them. The goal is to choose the next image from a set of given options that fits correctly into the sequence. Thus, solving DR problems using AI and CV requires visual representations of the objects or diagrams. It involves understanding of the concepts and ideas from images with the patterns that are used in visual IQ tests.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2023
D. Gupta et al. (Eds.): CVIP 2022, CCIS 1777, pp. 377–387, 2023.
https://doi.org/10.1007/978-3-031-31417-9_29

Solving diagrammatic reasoning problems using artificial intelligence can help to understand complex patterns of objects. It can also be used to generate new DR problems that can be used in tests. We have chosen to solve a class of diagrammatic reasoning problems that involve rotated objects. Such problems are referred to as rotational DR problems. In such DR problems, an object is rotated by certain angle to create a valid sequence. We are given with four options to choose the correct one in the sequence. The problems we chose contain a sequence of 4 images and we need to find out the 5th image from the given four options.

1.1 Related Work

Reasoning is the ability to make sense of things by verifying facts and applying logic. We refer to machine learning-based methods for reasoning as artificial reasoning (AR). AR uses knowledge completion, value approximation, and goal-oriented reasoning to solve different forms of reasoning [3]. Zhou et al. [4] have explored the use of knowledge graphs, that capture general or commonsense knowledge, to augment the information extracted from images by the state-of-the-art methods for image captioning. Value approximation is a method for extracting numeric facts. It is used in quantitative question answering from natural language texts and images [5]. Goal-oriented reasoning is a top-down approach that heuristically searches for a solution to achieve a goal. It is popular in robotics, intelligent agent, and case-based reasoning [6]. Data and knowledge-driven statistical methods [7], logic programming [8], and neural network-based approaches [9] are also popular for solving various reasoning problems.

Artificial reasoning methods are complex in nature and such methods require a logical representation of data, common sense, statistical information, and learning techniques. In the past few years, Deep learning has been widely used to learn and represent the features. However, majority of the existing representations rely on low-level features and they do not consider high-level representations such as logic or knowledge, etc. Recently, Serafini et al. [10] have proposed a logic tensor network (LTN) to learn the data-driven logic. LTN converts real logic formulas into TensorFlow computational graphs. Such formulas can express complex queries about the data. Kazemi et al. [11] have also proposed a deep neural network known as relational neural networks (RelNNs) to learn the reasoning directly from the FOL. Garcez et al. [12] have proposed a neural-symbolic computing approach to combine neural networks with symbolic representation and a reasoning-based learning approach. Mao et al. [13] have proposed a Neuro-Symbolic Concept Learner, a model that learns visual concepts, words, and semantic parsing of sentences without explicit supervision. This model learns by simply looking at images and reading paired questions and answers.

However, visual reasoning is not straightforward as compared to the other types of reasoning due to the difficulty in interpreting the objects and relations between them. Therefore, logical and statistical AI methods cannot directly be applied to solve visual reasoning problems. Two similar domains of reasoning that have received the attention of the CV research community are visual question answering and visual reasoning. Visual question answering consists of images and questions that can be answered from the images. To answer the questions, we may require prior knowledge about the objects, their color, position, etc. In addition to these features, visual reasoning may also require shape information, count, orientation, etc. Johnson et al. [5] have released a CLVR dataset that tests a range of visual reasoning abilities. It is used for reasoning color, shape, quantity, and size. Visual IQ questions that are based on RPM [2] vary in nature and are diverse in complexities. Answers to RPM-based reasoning require common sense, the idea about the shapes, and knowledge of mathematics. A recent work in this field by Arif et al. [1] introduces a new deep learning-based approach to solve DR problems. A knowledge acquisition module has been used that constructs a knowledge-base from the sequence of images given in the problem and answer options. The authors have used relation features such as rotation, counting, and scaling to prepare the knowledge-base. After this, the active features are chosen. Based on the active features, an LSTM network is used to find the correct option. For other types of DR problems, a ConvLSTM network has been used.

1.2 Contributions

The existing work discussed earlier have some limitations. For example, the work proposed in [1] cannot generate new sets of DR problems. Moreover, the architecture proposed cannot handle complex DR problems. Even for the rotation-related DR problems, the maximum accuracy has been reported to be around 76.2% using RF-LSTM framework. This can be further improved with encoder-decoder architecture. To mitigate some of the aforementioned problems, we have made the following technical contributions in this paper:

(a) We have proposed a new deep learning-based architecture that can solve rotational DR problems with better accuracy. The model predicts a score between [0, 1] for each options given. The correct option is the one that gets the highest score.
(b) We have created a new DR dataset using automated scripts. The dataset contains newly generated 1500 rotation DR problems. Each sample contains 8 images of size $(64 \times 64 \times 3)$. We have implemented and tested the model architecture and the new dataset. We have achieved better accuracy as compared to other known models suitable for solving rotational DR problems.

The rest of the paper is organized as follows. In Sect. 2, we present the proposed method. Section 3 summarizes the dataset and the experiment results. We conclude in Sect. 4.

2 Proposed Framework

We present the proposed architecture in this section. There are three main components in the architecture, namely a VGG16 feature extractor module, an LSTM module to encode the spation-temporal relation of the sequence of images, and a classifier. The first four images are given as inputs in the question sequence. The last image is one of the given options. The idea is to first extract the relation from this sequence of 5 images and then classify it into a valid or invalid relation.

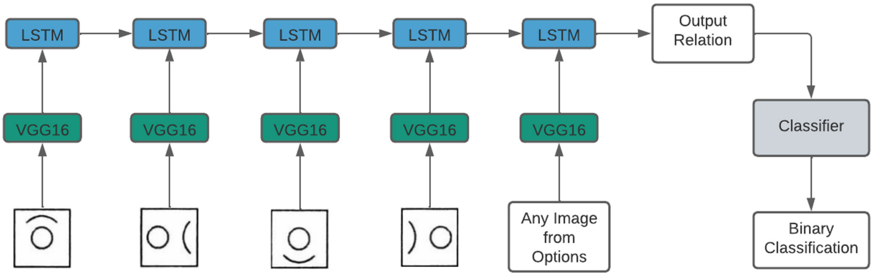

Fig. 1. The architecture of the proposed DR problem solving framework.

The VGG16 module has been used in the image encoder part. It takes a $64 \times 64 \times 3$ dimension image as input and produces a feature vector of size 2048. Once the encoder-decoder is trained, the decoder is discarded during the testing. The LSTM module takes the feature vector produced using the VGG16 as input and produces a relation vector of size 128. The last module (classifier) is a fully connected neural network. It takes a relation vector extracted by LSTM as input and performs a binary classification. It predicts a score in the range of [0,1]. The higher score an option gets, better the option. We then choose the option with the highest score as the correct option.

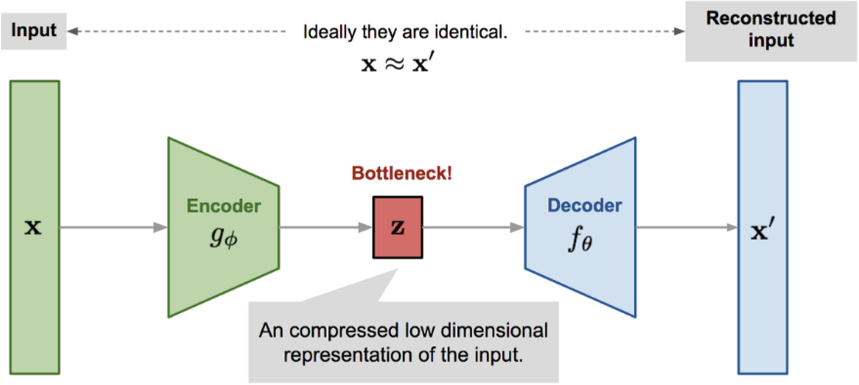
Input
Ideally they are identical.
$\mathbf{x} \approx \mathbf{x}'$
Reconstructed input
$\mathbf{x}$
Encoder
g_ϕ
Bottleneck!
$\mathbf{z}$
Decoder
f_θ
$\mathbf{x}'$
An compressed low dimensional representation of the input.

Fig. 2. The Encoder-Decoder training architecture.

2.1 Encoder-Decoder Training

We have trained the VGG16 encoder using a total of 12000 (1500×8) images. This has ensured that the module learns to encoder the shapes and arrangement of images. Figure 2 depicts the encoder-decoder architecture that has been used to train the encoder model from the scratch. A reconstruction loss as formulated using (1–6), has been used to regenerate the encoded images. Since this is an unsupervised step, we have trained the encoder on every image of the training set.

$$g_\psi : \text{ Encoder (VGG16)} \tag{1}$$

$$f_\theta : \text{ Decoder} \tag{2}$$

$$z = g_\psi(x) \tag{3}$$

$$x' = f_\theta(z) \tag{4}$$

$$\mathcal{L}(x, x') = ||x - x'||^2 \tag{5}$$

$$\mathcal{L}_{\text{total}} = \frac{1}{n} \sum_{\forall x \in \text{dataset}} \mathcal{L}(x, x') \tag{6}$$

2.2 LSTM Training

After training the VGG16-based encoder-decoder architecture, we have fixed the weights of VGG16 model. In the next step, we have trained the LSTM and the classifier parts of the model using the training set. Each sample in the dataset contains 8 images. Out of which, the first four are given as the input sequence and the other four are given as options from which the correct answer is selected. First, we encode a given sequence images using the VGG16-based encoder and the encoded features are given as input to the LSTM as shown in Figure 1. After this, we chose one of the options at a time and gave it's encoding as input to the LSTM. We take the output vectors of the LSTMs. These time-step outputs are used as inputs to the classifier. It performs as a binary classifier to check if the given sequence of 5 images is correct or not. The expected output of the classifier is 1 if the chosen option is correct, else 0. We have trained the classifier using binary cross-entropy loss. Since there is only 1 correct option and 3 incorrect options, we have multiplied the loss of the correct option by 3 to solve the class imbalance problem. The whole process is described in (7–13).

$$I_i : i^{th} \text{ image from the given question} \tag{7}$$

$$Op_i : i^{th} \text{ image from the given options} \tag{8}$$

$$[z_1^i, z_2^i, z_3^i, z_4^i] = [En(I_1^i), En(I_2^i), En(I_3^i), En(I_4^i)] \tag{9}$$

$$[\bar{z}_1^i, \bar{z}_2^i, \bar{z}_3^i, \bar{z}_4^i] = [En(Op_1^i), En(Op_2^i), En(Op_3^i), En(Op_4^i)] \tag{10}$$

$$R = LSTM(z_4^i, LSTM(z_3^i, LSTM(z_2^i, LSTM(z_1^i)))) \tag{11}$$

$$y_1^i = LSTM(\bar{z}_1^i, R), \quad y_2^i = LSTM(\bar{z}_2^i, R) \tag{12}$$

$$y_3^i = lstm(\bar{z}_3^i, R), \quad y_4^i = lstm(\bar{z}_4^i, R) \tag{13}$$

The loss is calculated using (14–15) when the first option is correct.

$$\mathcal{L}'_i = 3 * BCE(y_1^i, 1) + \sum_{j=2}^{4} BCE(y_j^i, 0) \tag{14}$$

$$\mathcal{L}_{total} = \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}'_i \tag{15}$$

3 Experiments and Results

3.1 Dataset

Figure 3 is a sample from our created dataset. It contains 8 images. First four images are given as a question which create a sequence of rotating images. These images and an image from options (one at a time) will be given as input to LSTM after encoding to encode the relation used to created this sequence. Our goal is to find the correct option from given options which fits correctly as 5th image in the sequence. So the option for which our model will predict the most score will be chosen as correct option.

Our dataset contains total 1500 samples, each of which has 8 images. We have used 1050 (70%) samples for training and rest 450 (30%) were used for testing the model.

3.2 Accuracy

We have achieved an accuracy of 94.66% on train set and 85.55% on test set. Table 1 contains the our model accuracy and size of train and test dataset.

3.3 Model Predictions

Below are some predictions from our model. We have included few of correctly and incorrectly predicted examples from our test set. Figures 4 and 5 are correctly predicted by our model a high score for the correct option. Figures 7 and 6 are negative samples in which wrong option is getting more score than correct option. Also in Fig. 7, we can see that option 1 and option 4 are very similar and option 4 is getting a slightly better score than option 1.

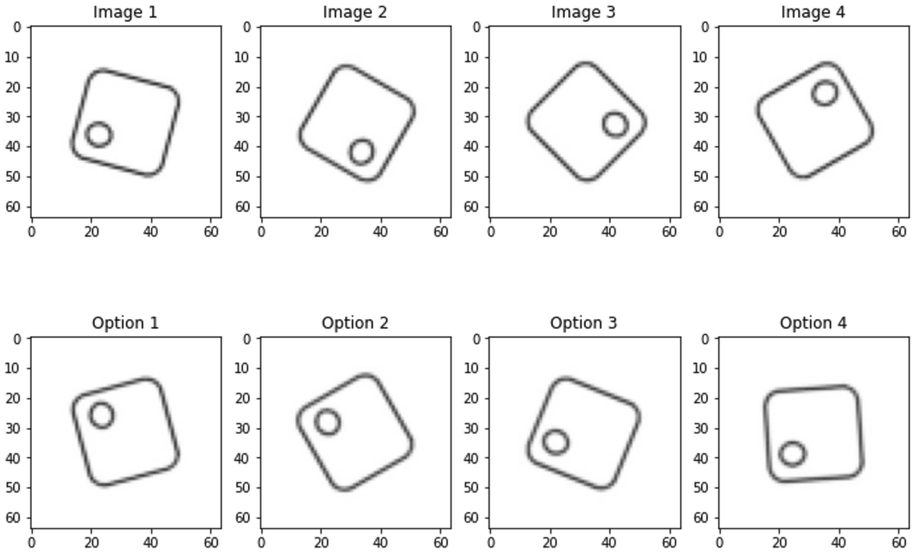

Fig. 3. Dataset Example

Table 1. Train and Test Set Description

	Train Set	Test Set
Accuracy	94.66 %	85.55 %
Size	1050	450

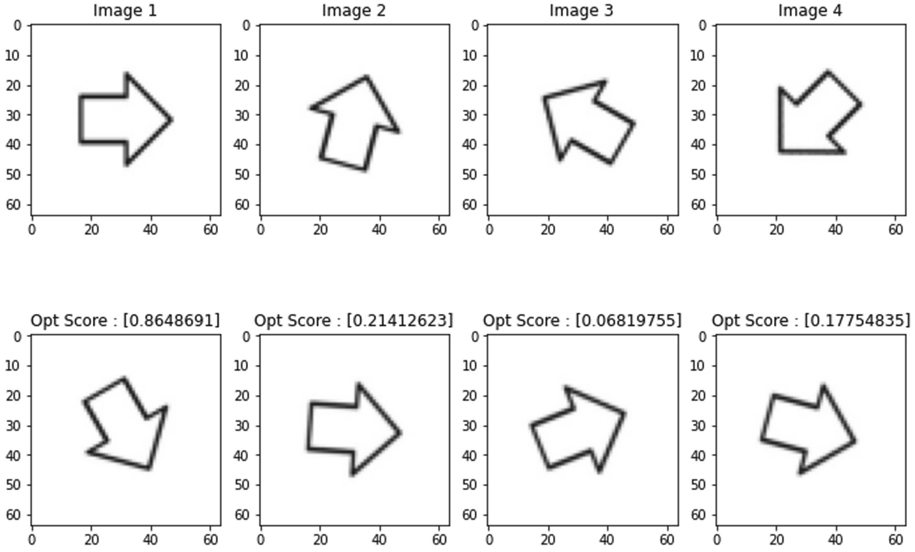

Fig. 4. Correctly Predicted Example

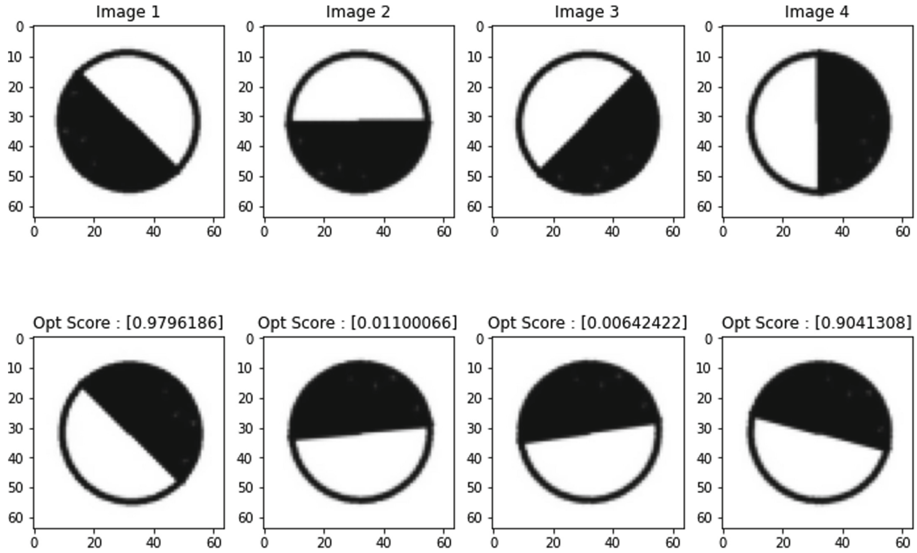

Fig. 5. Correctly Predicted Example

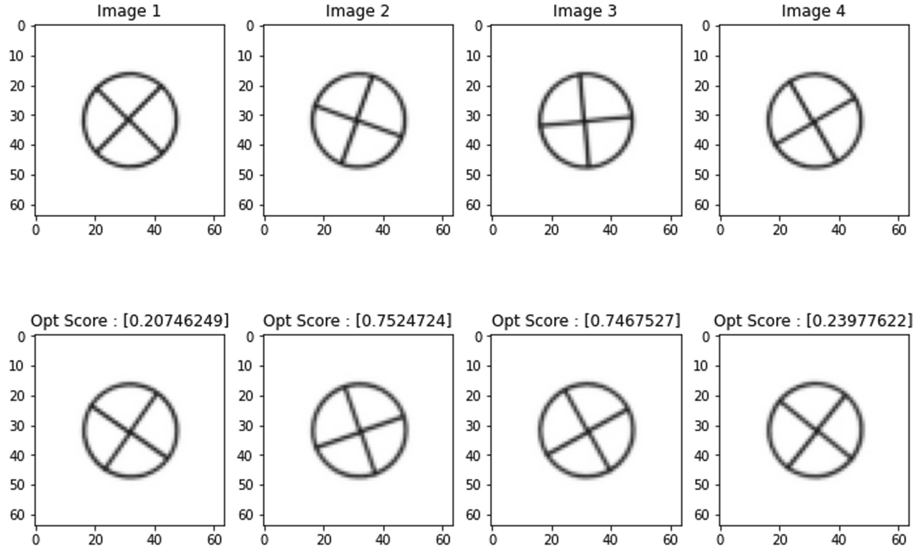

Fig. 6. Incorrectly Predicted Example

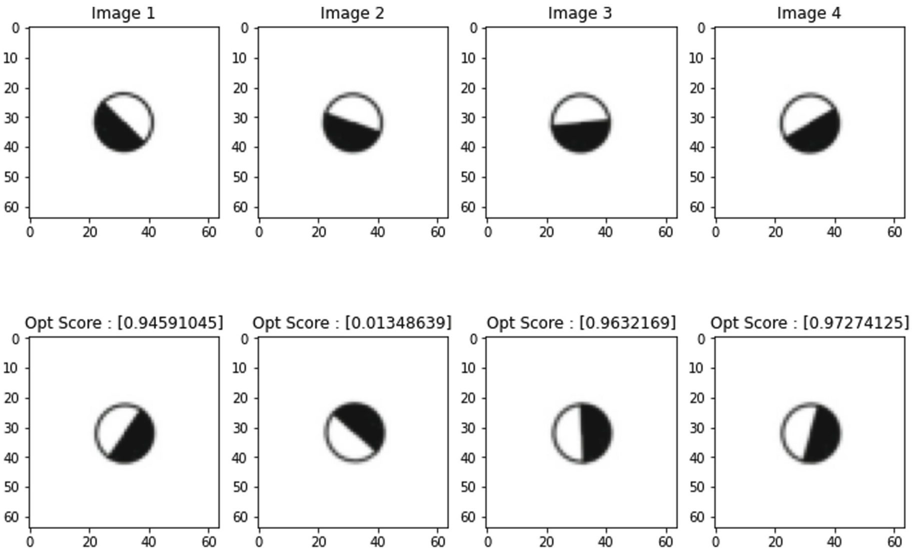

Fig. 7. Incorrectly Predicted Example

4 Conclusion

This article contributes towards the challenges and the possibilities of cognitive learning. We considered Raven's Progressive Matrix for experiment. The article proposed a solution towards imitating the visual cognitive reasoning of a person. We solve a simple visual reasoning problem using AI. There are many questions and possibilities, we hope the research will attract more CV researchers in this domain.

References

1. Dogra, D.P., Sekh, A.A., Kar, S., Roy, P.P., Prasad, D.K.: Can we automate diagrammatic reasoning, October (2020)
2. Burke, H.R.: Raven's progressive matrices: a review and critical evaluation. J. Genet. Psychol. **93**(2), 199–228 (1958)
3. Diamantini, C., Freddi, A., Longhi, S., Potena, D., Storti, E.: A goal-oriented, ontology-based methodology to support the design of AAL environments. Expert Syst. Appl. **64**, 117–131 (2016)
4. Zhou, Y., Sun, Y., Honavar, V.: Improving image captioning by leveraging knowledge graphs. In: IEEE Winter Conference on Applications of Computer Vision, pp. 283–293 (2019)

5. Johnson, J., Hariharan, B., van der Maaten, L., Fei-Fei, L., Zitnick, C.L., Girshick, R.: CLEVR: a diagnostic dataset for compositional language and elementary visual reasoning. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1988–1997 (2017)
6. Giorgini, P., Mylopoulos, J., Nicchiarelli, E., Sebastiani, R.: Reasoning with goal models. In: Spaccapietra, S., March, S.T., Kambayashi, Y. (eds.) ER 2002. LNCS, vol. 2503, pp. 167–181. Springer, Heidelberg (2002). https://doi.org/10.1007/3-540-45816-6_22
7. Mineau, G.W., Godin, R.: Automatic structuring of knowledge bases by conceptual clustering. IEEE Trans. Knowl. Data Eng. **7**(5), 824–829 (1995)
8. Raedt, L.D., Kersting, K., Natarajan, S., Poole, D.: Statistical relational artificial intelligence: logic, probability, and computation. Synth. Lect. Artif. Intell. Mach. Learn. **10**(2), 1–189 (2016)
9. Shin, C.-U., Cha, J.-W.: End-to-end task dependent recurrent entity network for goal-oriented dialog learning. Comput. Speech Lang. **53**, 12–24 (2019)
10. Serafini, L., d'Avila Garcez, A.S.: Learning and reasoning with logic tensor networks. In: Adorni, G., Cagnoni, S., Gori, M., Maratea, M. (eds.) AI*IA 2016. LNCS (LNAI), vol. 10037, pp. 334–348. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-49130-1_25
11. Kazemi, S.M., Poole, D.: ReINN: a deep neural model for relational learning. In: 32nd AAAI Conference on Artificial Intelligence, pp. 6367–6375 (2018)
12. Garcez, A., Gori, M., Lamb, L., Serafini, L., Spranger, M., Tran, S.: Neural-symbolic computing: an effective methodology for principled integration of machine learning and reasoning. J. Appl. Logics **6**(4), 611–632 (2019)
13. Mao, J., Gan, C., Kohli, P., Tenenbaum, J.B., Wu, J.: The neuro-symbolic concept learner: interpreting scenes, words, and sentences from natural supervision. In: International Conference on Learning Representations, pp. 1–28 (2019)
14. Wang, J., Wang, W., Wang, L., Wang, Z., Feng, D.D., Tan, T.: Learning visual relationship and context-aware attention for image captioning. Pattern Recogn. **98**, 107075–107086 (2020)
15. Wang, W., Huang, Y., Wang, L.: Long video question answering: a matching-guided attention model. Pattern Recogn. **102**, 107–248 (2020)
16. Santoro, A., Hill, F., Barrett, D., Morcos, A., Lillicrap, T.: Measuring abstract reasoning in neural networks. In: International Conference on Machine Learning, pp. 4477–4486 (2018)
17. Hill, F., Santoro, A., Barrett, D., Morcos, A., Lillicrap, T.: Learning to make analogies by contrasting abstract relational structure. In: International Conference on Learning Representations, pp. 1–14 (2019)
18. Kunda, M., McGreggor, K., Goel, A.: Addressing the ravens progressive matrices test of general intelligence. In: AAAI Fall Symposium Series, pp. 22–27 (2009)
19. Lovett, A., Forbus, K., Usher, J.: A structure-mapping model of raven's progressive matrices. In: Proceedings of the Annual Meeting of the Cognitive Science Society, vol. 32, pp. 2761–2766 (2010)
20. Ragni, M., Neubert, S.: Solving Raven's IQ-tests: an AI and cognitive modeling approach. In: Proceedings of the 20th European Conference on Artificial Intelligence, pp. 666–671. IOS Press (2012)
21. Lovett, A., Forbus, K.: Modeling visual problem solving as analogical reasoning., Psychol Rev. **124**(1), 60 (2017)
22. Zhang, C., Gao, F., Jia, B., Zhu, Y., Zhu, S.-C.: Raven: a dataset for relational and analogical visual reasoning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5317–5327 (2019)

23. Sutskever, I., Vinyals, O., Le, Q.V.: Raven, sequence to sequence learning with neural networks. Accessed 10 Sep 2014
24. Bank, D., Koenigstein, N., Giryes, R.: Autoencoders, v1. Accessed 12 Mar 2020
25. Ronneberger, O., Fischer, P., Brox, T.: U-Net: convolutional networks for biomedical image segmentation (2021)