WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Neuro-Symbolic AI: Bringing a new era of Machine Learning

***Rabinandan Kishor*[1]**

[1]***www.rabinandan.in Sapporo, Japan***

Corresponding Author: Rabinandan Kishor, **E-mail**: hi@rabinandan.in

DOI: https://doi.org/10.55248/gengpi.2022.31271

ABSTRACT

Processing Natural Language using machines is not a new concept. Back in 1940 researchers estimated the importance of a machine that could translate one language to another. Further, during 1957-1970 researchers split into two divisions concerning NLP: symbolic and stochastic. This paper presents an extensive review of recent breakthroughs in Neuro Symbolic Artificial Intelligence (NSAI), an area of AI research which seeks to combine traditional rules-based AI approaches with modern deep learning techniques. Neuro Symbolic models have already demonstrated the capability to outperform state-of-the-art deep learning models in domains such as image and video reasoning. Such models not only performed better when trained on a fraction of dataset compared with traditional machine learning models, but also solved an underlined issue called generalization of deep neural network systems. We also find that symbolic models are good in visual question answering (VQA). In this paper, we also review research results related to Neuro Symbolic AI with the objective of exploring the importance of such AI systems and how it would shape the future of AI as a whole. We discuss different types of dataset of Visual Question Answering(VQA) tasks based on NSAI and extensive comparison of performance of different NSAI models. Later, the article focuses on the contemporary real time application of NSAI systems and how NSAI is shaping the world's different sectors including finance, healthcare, cyber security.

Keywords : Neuro Symbolic AI, Machine Learning, Natural Language Processing, NSCL

1. Introduction

Over the past decade, Artificial Intelligence and, in particular deep learning have attracted media attention, have become the focus of increasingly large research endeavors, and have changed businesses.This led to influential debates on the impact of AI both on academia and industry[1][2] Turing, in his paper called 'Computing Machinery and Intelligence' in 1950, proved that machines could be intelligent. However, even at that time, the word artificial intelligence did not appear directly, and there was no specific content. However, six years later, John McCarthy of Dartmouth University mentioned the term 'Artificial Intelligence' at a Dartmouth conference(McCarthy et al., 2006) and suggested specific discussions and research, and research on artificial intelligence has started.From there two fields of AI have been initiated - Connectionism and Symbolisms.

Systems	Advantages	Disadvantages
Symbolic system (Good at deductive reasoning)	Strong generalization ability Interpretability Knowledge driven	Not good at handling unstructured data Weak robustness Slowly reasoning
Neural system (Good at inductive learning)	Good at handling unstructured data Strong robustness Fastly learning	Weak generalization ability No interpretability Requiring a large amount of label data

Figure 1: Pros and cons of symbolic and neural systems.

Despite many attempts, either symbolic or neural systems were not able to solve many limitations of language models. Fortunately, the Neural-Symbolic AI (NSAI) approach, which is the combination of both neural and symbolic approaches, can solve some of those issues. By neural we mean approaches based on artificial neural networks—sometimes called connectionist or subsymbolic approaches—and in particular, this includes deep learning, which has provided very significant breakthrough results in the recent decade and is fueling the current general interest in AI. By symbolic, we mean approaches that rely on the explicit representation of knowledge using formal languages—including formal logic—and the manipulation of language items ('symbols') by algorithms to achieve a goal. Mostly, neuro-symbolic AI utilizes formal logic as studied in the knowledge representation and reasoning subfield of AI, but the lines blur, and tasks such as general term rewriting or planning, that may not be framed explicitly in formal logic, bear significant similarities and should reasonably be included.

The objective of NSAI is to extract features from data using DL approaches, then manipulate these features using symbolic approaches. Neuro-Symbolic models have outperformed pure DL models on image and video question answering tasks, and have proven to converge far quicker, with as little as 1/10 of the training data needed for accuracy.

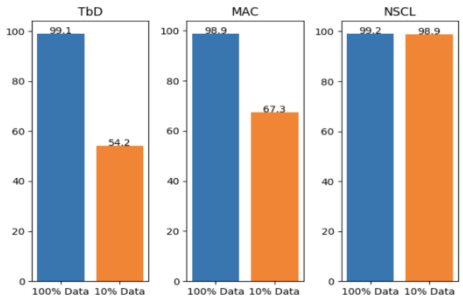

Figure 2: The neuro-symbolic model (NSCL [3]) achieves good accuracy even when only trained with 10% of the data (CLEVR dataset) whereas the accuracy of neural models (TbD [4] and MAC [5]) deteriorates when the training set is small. Data is drawn from [3].

In the figure 2 above a comparison of the Neuro-Symbolic based NSCL model with other pure deep learning models - TbD and MAC. While the accuracy of all three models are similar when training on 100% of the dataset: however, NSCL performed dramatically better even though we provided 10% of the dataset for the training. Hence, the Neural-Symbolic based model can be a better fit when we do not have data available for training or some contrarian of the collection of training data. In this paper, we explore the performance of different Neuro-Symbolic models. Those models have mainly been trained on recently developed datasets called CLEVR [6] and CLEVRER [7]. In this paper we cover different models including Neuro-Symbolic Concept Learner (NSCL) and Neuro-Symbolic Dynamic Reasoning (NS-DR), are composed of multiple independent "submodels", which extract problem features before passing them as input to a final symbolic submodel, and Neural Logic Machines (NLM), is an end-to-end model without independent submodels. Most of the recent work towards the improvement of NSAI is being carried out by the MIT and IBM team. In 2019, the Neuro-Symbolic Concept Learner proved the viability of combining symbolic AI with deep learning techniques by parsing input questions and scenes into symbolic programs[3]. The Concept Learner demonstrated a novel technique for parsing input scenes and questions into a semantic program, and introduced new techniques for training this parsing at the same time as the symbolic execution engine. The Concept Learner model was followed the next year by the Visual Concept-Metaconcept Learning (VCML) model, which used embeddings to learn object properties with less supervision[8].

2. Literature Review

At IBM, Neuro-Symbolic(NS) AI researchers aim to conceive a fundamentally new methodology for AI, to address the gaps remaining between today's state-of-the-art and the full goals of AI, including AGI. In particular it is aimed at augmenting (and retaining) the strengths of statistical AI (machine learning) with the complementary capabilities of symbolic AI (knowledge and reasoning). It is aimed at a construction of new paradigms rather than a superficial synthesis of existing paradigms, and revolution rather than evolution. Their NS research directly addresses long-standing obstacles including imperfect or incomplete knowledge, the difficulty of semantic parsing, and computational scaling. NS is oriented toward long-term science via a focused and sequentially constructive research program, with open and collaborative publishing, and periodic spinoff technologies, with a small selection of motivating use cases over time. The primary ones currently include the pursuit of true natural language understanding via the proxy of question answering; automatic data science, programming, and mathematics; and financial trading/risk optimization as ways to showcase the fundamental principles being developed. Research in NS is inherently multi-disciplinary and includes (among many other things) work in learning theory and foundations, optimization and algorithms, knowledge representation and acquisition, logic and theorem proving reinforcement learning, planning, and control, and multi-task/meta/transfer learning[9]. IBM research on Neuro-symbolic AI is categorized into 8 different sections as follows[10]:

1. Logical Neural Network(LNN)

The novel LNNs framework has been developed by IBM recently designed to seamlessly provide key properties of both neural nets (learning) and symbolic logic (knowledge and reasoning). Every neuron has a meaning as a component of a formula in a weighted real-valued logic, yielding a highly interpretable disentangled representation. Inference is omnidirectional rather than focused on predefined target variables, and corresponds to logical reasoning, including classical first-order logic theorem proving as a special case. The model is end-to-end differentiable, and learning minimizes a novel loss function capturing logical contradiction, yielding resilience to inconsistent knowledge. It also enables the open-world assumption by maintaining bounds on truth values which can have probabilistic semantics, yielding resilience to incomplete knowledge[11].

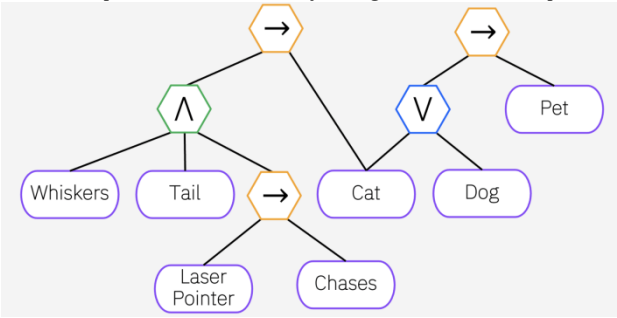

Figure 3: IBM LNN : An Interpretable Neural Reasoner [12]

Different types of LNN models are being explored by the IBM team which includes FOL-LNN (First-order logic LNN), Logical Optimal Actions (LOA), Logical Formula Embedder, and Tensor-LNN.

2. Natural language processing via reasoning (NLP)

This system works to predict the knowledge base types of the answer to a given natural language question[13]. Input of this system is 'a neural network question' while the output will be the following in the figure below[14]

Answer Category	Allowed Answer Types
Resource	Any number of DBpedia types
Literal	Number, Date or String
Boolean	Boolean

Figure 4: Expected output of IBM Answer type prediction[14]

Many other variants of reasoning NLP have been developed by IBM which include SLING(Relation linking framework), Sem Rel (Simple transformer-based Relation linking), GenRL (Relation linking as a generative problem), Logic Embeddings, and Knowledge-Enabled Textual-Entailment.

3. Knowledge foundation (KF)

The Knowledge foundation focuses on generating a "knowledge hub" and arranging it into graphs. One popular model The Expressive Reasoning Graph Store (ERGS), in this category, deals with is an OWL[16] reasoner and an RDF triple store built on top of a Property Graph architecture. ERGS uses Janus Graph as the underlying graph store which can be replaced by any Apache TinkerPop compliant graph store[15].

4. Learning with less (LwL)

Learning with less and generalization is a key focus area under neuro-symbolic AI research at IBM. Less can be interpreted in various ways. Some of the goals of this area of research are: 1) learn with 10x less computational resources 2) Multi-task learning with 10x less data or interactions 3) Compositional generalization for minimal recursive semantics (MRS) with LNNs 4) Defining a quantitative measure of intelligence for multiple tasks[17].

5. Knowledge augmented sequential decision making (SDM)

In many reinforcement learning settings, the choice of the underlying algorithm is not obvious. The RLAPSE[18] framework provides the choice whether to use a lightweight myopic algorithm (e.g., Q-learning with discount factor 0) or more complicated (e.g., Q-learning with discount factor close to 1) reinforcement learning algorithm. The framework utilizes (i) the likelihood ratio test, and (ii) a variant of Q-learning analyzed by (Even-Dar et al. 2003))[19]. Based on the collected statistics about the environment, the RLAPSE orchestrator switches to a more appropriate algorithm, if deemed necessary[18].

6. Human in the loop (HIL)

Human-in-the-Loop (HITL) enables human verification and corrections to ensure accuracy of data extracted by Document AI processors before it is used in critical business applications. It provides a workflow and UI for humans (referred to as labelers in HITL) to review, validate and correct the data extracted from documents by Document AI processors. It is used across Financial Services, Health, Manufacturing, Government and other industries[20].

7. Datasets and environments (DS)

Life Jacket dataset, Power plant dataset, TextWorld Commonsense (Game engine for cleaning room), and Logical Twins (Textworld-to-PDDLGym) are examples of DS at IBM[10].

8. Related advances (RA)

This category consists of the s wide variety of AI-models including Binary Matrix Factorization, Axiom Verification, Ethical AI Platform, Ethical AI Platform, Ethical AI Demo, Formal ML, GraphSEIR_aPCE, MER, Policy Gradient Algorithm for Learning to Learn in Multiagent RL, Framework for collecting moral preference data, Online Alternating Minimization, Unsupervised Learning of Graph Hierarchical Abstractions, Sobolev Independence Criterion, Persistence Homology for Link Prediction, and TM-GCN[10].

3. Methodology

In the section we discuss different datasets and models we have examined for the research.

3.1 Dataset Overview

Although many datasets are being used for the development of Neuro-Symbolic model: however two of them are extremely popular and brough the NSAI to the next level. One of them is the CLEVR dataset. It is a diagnostic dataset that tests a range of visual reasoning abilities. It contains minimal biases and has detailed annotations describing the kind of reasoning each question requires. They use this dataset to analyze a variety of modern visual reasoning systems, providing novel insights into their abilities and limitations[21]. CLEVR provides a dataset that requires complex reasoning to solve and that can be used to conduct rich diagnostics to better understand the visual reasoning capabilities of Visual Question Answering(VQA) systems.

This requires tight control over the dataset, which we achieve by using synthetic images and automatically generated questions. The images have associated ground-truth object locations and attributes, and the questions have an associated machine-readable form.

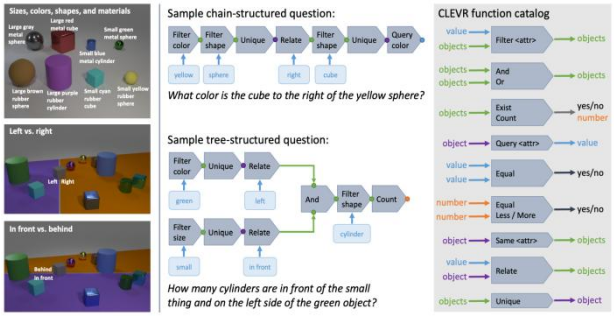

Figure 5: A field guide to the CLEVR universe. Left: Shapes, attributes, and spatial relationships. Center: Examples of questions and their associated functional programs. Right: Catalog of basic functions used to build questions[21].

The ground truth structures allow developers to analyze models based on various forms of relationships between objects. Below is the list of main components of CLEVR dataset:

Objects and relationships: Three object shapes (cube, sphere, and cylinder) that come in two absolute sizes (small and large), two materials (shiny "metal" and matte "rubber"), and eight colors are part of CLEVR. Objects are spatially related via four relationships: "left", "right", "behind", and "in front". The semantics of these prepositions are complex and depend not only on relative object positions but also on camera viewpoint and context.

Scene representation: Scenes are represented as collections of objects annotated with shape, size, color, material, and position on the ground-plane. A scene can also be represented by a scene graph [22, 23], where nodes are objects annotated with attributes and edges connect spatially related objects.

Image generation: CLEVR images are generated by randomly sampling a scene graph and rendering it using Blender [24]. Every scene contains between three and ten objects with random shapes, sizes, materials, colors, and positions.

Question representation: Every question is associated with a functional program that can be executed on an image's scene graph, yielding the answer to the question. Functional programs are built from simple basic functions that correspond to elementary operations of visual reasoning such as querying object attributes, counting sets of objects, or comparing values[21].

Auto generation of question: Based upon images, different combinations of questions have been generated. The process is fairly simple and fast and can be performed by choosing a question family, then selecting values for each of its template parameters, then executing the resulting program on the image's scene graph to find the answer, and finally, using one of the text templates from the question family to generate the final natural-language question.

The below diagram shows the number of images and questions present in the dataset with their spilt of train, val and test.

Split	Images	Questions	Unique questions	Overlap with train
Total	100,000	999,968	853,554	-
Train	70,000	699,989	608,607	-
Val	15,000	149,991	140,448	17,338
Test	15,000	149,988	140,352	17,335

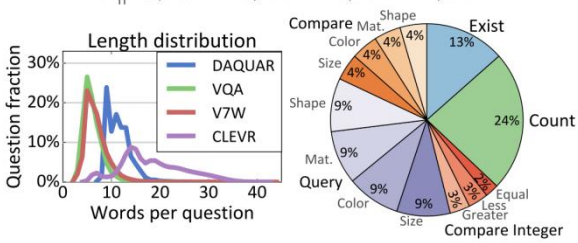

Figure 6: Top: Statistics for CLEVR; the majority of questions are unique and few questions from the val and test sets appear in the training set. Bottom left: Comparison of question lengths for different VQA datasets; CLEVR questions are generally much longer. Bottom right: Distribution of question types in CLEVR[21].

Another popular dataset is the Collision Events For Video REpresentation And Reasoning (CLEVRER) dataset. It is a diagnostic video dataset for systematic evaluation of computational models on a wide range of reasoning tasks. The CLEVRER dataset includes four types of question: descriptive (e.g., 'what color'), explanatory ('what's responsible for'), predictive ('what will happen next'), and counterfactual ('what if') is being evaluated for various state-of-the-art models for visual reasoning to generate improved benchmark results[25].

The CLEVRER includes 10,000 videos for training, 5,000 for validation, and 5,000 for testing. All videos last for 5 seconds. The videos are generated by a physics engine that simulates object motion plus a graphs engine that renders the frames[25]. The component of CLEVRER dataset is listed below:

Objects and events: Objects in CLEVRER videos adopt similar compositional intrinsic attributes as in CLEVR[21], including three shapes (cube, sphere, and cylinder), two materials (metal and rubber), and eight colors (gray, red, blue, green, brown, cyan, purple, and yellow).There are three types

of events: enter, exit and collision, each of which contains a fixed number of object participants: 2 for collision and 1 for enter and exit. The objects and events form an abstract representation of the video[25].

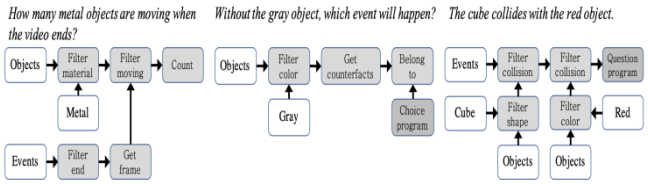

Figure 7: Sample questions and programs from CLEVRER. Left: Descriptive question. Middle and right: multiple-choice question and choice. Each choice can pair with the question to form a joint logic trace[25].

Causal structure: An event can be either caused by an object if the event is the first one participated by the object, or another event if the cause event happens right before the outcome event on the same object. Those objects and events in videos have causal structures[25].

Video generation: Using Bullet physics engine[26], for motion simulation, CLEVRER videos have been generated from the simulated motion traces, including each object's position and pose at each time step. Each simulation lasts for seven seconds. The motion traces are first down-sampled to fit the frame rate of the output video (25 frames per second). Then the motion of the first five seconds are sent to Blender[27] to render realistic video frames of object motion and collision[25].

Nature of Questions & Question generation:

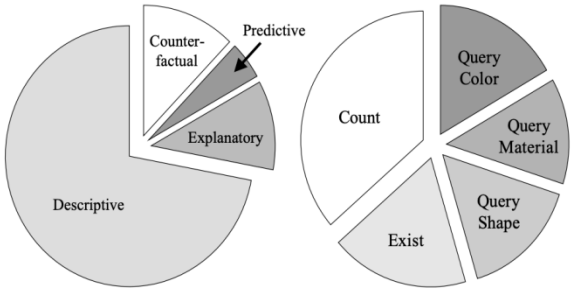

Figure 8: Distribution of CLEVRER question types. Left: distribution of four main question types. Right: distribution of descriptive sub-type[25].

All the videos in CLEVRER universe have been paired with machine-generated questions for descriptive, explanatory, predictive, and counterfactual reasoning. CLEVRER consists of 219,918 descriptive questions, 33,811 explanatory questions, 14,298 predictive questions and 37,253 counterfactual questions. Detailed distribution and split of the questions can be found in figure 8.

Comparison of datasets

The below figure 9 shows the comparison of different datasets popular for Neuro-Symbolic AI tasks for video and image.

Dataset	Video	Diagnostic Annotations	Temporal Relation	Explanation	Prediction	Counterfactual
VQA (Antol et al., 2015)	×	×	×	×	×	×
CLEVR (Johnson et al., 2017a)	×	✓	×	×	×	×
COG (Yang et al., 2018)	×	✓	✓	×	×	×
VCR (Zellers et al., 2019)	×	✓	×	✓	×	✓
GQA (Johnson et al., 2017a)	×	✓	×	×	×	×
TGIF-QA (Jang et al., 2017)	✓	×	✓	×	×	×
MovieQA (Tapaswi et al., 2016)	✓	×	✓	✓	×	×
MarioQA (Mun et al., 2017)	✓	×	✓	✓	×	×
TVQA (Lei et al., 2018)	✓	×	×	✓	×	×
Social-IQ (Zadeh et al., 2019)	✓	×	×	✓	×	×
CLEVRER (ours)	✓	✓	✓	✓	✓	✓

Figure 9: Comparison between CLEVRER and other visual reasoning benchmarks on images and videos. CLEVRER is a well-annotated video reasoning dataset created under a controlled environment. It introduces a wide range of reasoning tasks including description, explanation, prediction and counterfactuals[25].

3.2 Model Analysis

In this section we focus on different Artificial Intelligence(AI) models built for NSAI over time for VQA and its other applications. VQA is the task of answering a natural language question about a given image. NS approaches combine neural networks with a symbolic component (e.g., symbolic reasoning, symbolic representations, or exploiting pre-existing symbolic knowledge). NS approaches to VQA are of particular interest as they may promote compositional generalization[39][40][41] (i.e., the ability to understand new combinations of previously acquired concepts; for instance, "Buzz Aldrin wielded a glass hammer"). NS approaches also promise to be more interpretable and may allow for simpler skill transfer between tasks. Furthermore, areas of Neural Language Processing(NLP) are also benefited by NSAI approach, in particular, Natural Language Understanding (NLU), Natural Language Inference (NLI), and Natural Language Generation (NLG).

Natural Language Understanding (NLU) is a large subset of NLP containing topics particularly focused on semantics and meaning. The boundaries between NLP and NLU are not always clear and open to debate, and even when they are agreed upon, they're somewhat arbitrary, as it's a matter of convention and a reflection of history [43].

Natural Language Inference (NLI) enables tasks like semantic search, information retrieval, information extraction, machine translation, paraphrase acquisition, reading comprehension, and question answering. It is a problem of determining whether a natural language hypothesis h can reasonably be inferred from a given premise p [44]. For example, the premise "Hazel is an Australian Cattle Dog", entails the hypothesis "Hazel is a dog", and can be expressed in First Order Logic (FOL) by: p |= h.

Natural Language Generation (NLG) is the task of generating text or speech from non-linguistic (structured) input [45]. It can be seen as orthogonal to NLU, where the input is natural language. An end-to-end system can be made up of both NLU and NLG components. When that is the case, what happens in the middle is not always that clear-cut. A neural language model such as GPT3 [46] has no structured component; however, whether it performs "understanding" is subject to debate.

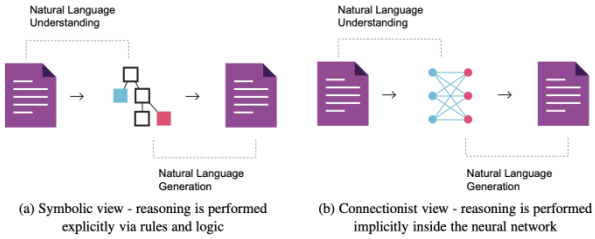

Figure 10: NLU takes as input unstructured text and produces output which can be reasoned over. NLG takes as input structured data and outputs a response in natural language[42].

NMN: The Neural Module Network (NMN) [47] was the first module network for VQA. The NMN DSL lists 5 modules types (e.g., attend), each with distinct architectures (see Fig 9). Most modules create or modify heatmaps of the image. Each module type has a unique instance per specific task. The sample execution in Fig 10 shows two such instances: attend[circle] and attend[red]. To produce the program, the NMN lemmatizes the question, applies the Stanford dependency parser [48], and applies fixed rules.

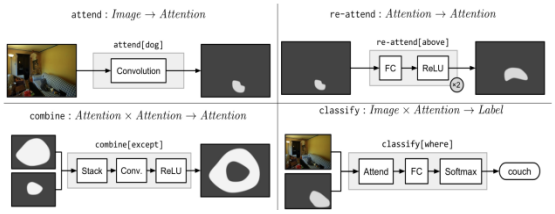

Figure 11: Graphic describing four of the five module types used by the NMN. Not shown is the measure module, which, given an attention map, returns a measurement (e.g., measure[exists], or measure[count]) [47].

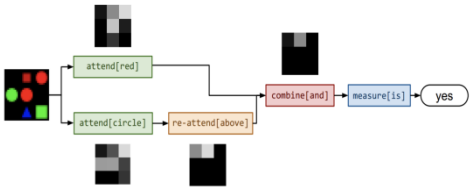

Figure 12: Visualization of NMN execution on a SHAPES problem. The question is "Is there a red shape above a circle?", and the corresponding program is measure[is](combine[and](attend[red], re-attend[above](attend[circle]))))[47].

N2NMN: Improving the NMN: This End-to-End Module Network (N2NMN) [49] is an improved version of NMN Network. Primarily, it removed the blind model and produced programs (in reverse Polish notation) with a seq2seq encoder-decoder LSTM architecture. Additionally, each module type (e.g., attend) has exactly one instance and is provided a text-context vector, xtxt, (an attention over input tokens), and image features, xvis, (extracted by VGG-16 [50]). The modules and their architectures are in figure 13 and 14.

Module name	Att-inputs	Features	Output	Implementation details
`find`	(none)	x_{vis}, x_{txt}	att	$a_{out} = \text{conv}_2(\text{conv}_1(x_{vis}) \odot W x_{txt})$
`relocate`	a	x_{vis}, x_{txt}	att	$a_{out} = \text{conv}_2(\text{conv}_1(x_{vis}) \odot W_1 \text{sum}(a \odot x_{vis}) \odot W_2 x_{txt})$
`and`	a_1, a_2	(none)	att	$a_{out} = \text{minimum}(a_1, a_2)$
`or`	a_1, a_2	(none)	att	$a_{out} = \text{maximum}(a_1, a_2)$
`filter`	a	x_{vis}, x_{txt}	att	$a_{out} = \textbf{and}(a, \textbf{find}[x_{vis}, x_{txt}]())$, *i.e.* reusing `find` and `and`
`[exist, count]`	a	(none)	ans	$y = W^T \text{vec}(a)$
`describe`	a	x_{vis}, x_{txt}	ans	$y = W_1^T (W_2 \text{sum}(a \odot x_{vis}) \odot W_3 x_{txt})$
`[eq_count, more, less]`	a_1, a_2	(none)	ans	$y = W_1^T \text{vec}(a_1) + W_2^T \text{vec}(a_2)$
`compare`	a_1, a_2	x_{vis}, x_{txt}	ans	$y = W_1^T (W_2 \text{sum}(a_1 \odot x_{vis}) \odot W_3 \text{sum}(a_2 \odot x_{vis}) \odot W_4 x_{txt})$

Figure 13: Table 1 of Hu et al. [49] describing the N2NMN's modules, and their implementations.

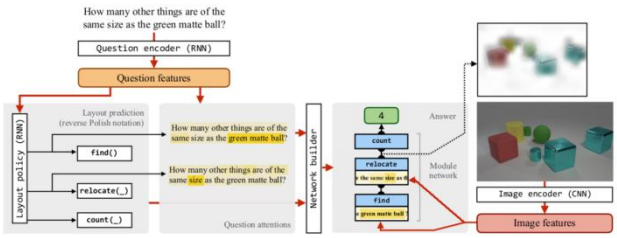

Figure 14: Illustration of N2NMN execution.

The above figure 12 shows an illustration of N2NMN execution. Note that, unlike the NMN which has separate instantiations for attend[red] and attend[circle], the equivalent module in N2NMN, find, has only one instantiation. find chooses what to search for based on an xtxt vector passed into the module; in this case xtxt is a weighted average of the input tokens with most of the weight on "green matte ball"[49].

Evaluation of the N2NMN on VQA 1.0 and SHAPES demonstrates increased performance over NMN. However, end-to-end training of the program generator through the modules via backpropagation is impossible. Executing the N2NMN requires arranging the modules; thus we must sample a discrete program from the program parser, and we cannot backpropagate through the sampling operation. REINFORCE can be used but causes a drop in performance (4% on SHAPES and 14% on CLEVR). Instead, the N2NMN is trained in two stages: initially the generator is encouraged to mimic a provided expert parser, afterwards it is fine-tuned with REINFORCE. Thus training requires an expert parser to mimic[51].

4. Results and Discussion

We have found that experiments done on one the CLEVR and the CLEVRER datasets, in recent years, produced better outcomes compared with other early systems. In this section, we focus on experimental results obtained on these datasets for VQA questions.

Accuracy of CLEVR models

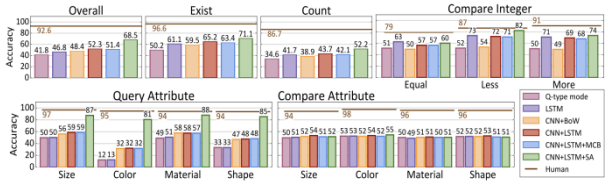

Figure 14: Accuracy per question type of the six VQA methods on the CLEVR dataset (higher is better). Figure best viewed in color[6].

For VQA tasks, CLAVR team reproduced a representative subset of methods: baselines that do not look at the image (Q-type mode, LSTM), a simple baseline (CNN+BoW) that performs near state-of-the-art , and more sophisticated methods using recurrent networks (CNN+LSTM), sophisticated feature pooling (CNN+LSTM+MCB), and spatial attention (CNN+LSTM+SA).

Q-type mode: Similar to the "per Q-type prior" method[52], this baseline predicts the most frequent training-set answer for each question's type.

LSTM: Similar to "LSTM Q"[52] , the question is processed with learned word embeddings followed by a word level LSTM . The final LSTM hidden state is passed to a multi-layer perceptron (MLP) that predicts a distribution over answers. This method uses no image information so it can only model question-conditional bias.

CNN+BoW: Following [53] , the question is encoded by averaging word vectors for each word in the question and the image is encoded using features from a convolutional network (CNN). The question and image features are concatenated and passed to a MLP which predicts a distribution over answers. The team uses word vectors trained on the GoogleNews corpus [54]; these are not fine-tuned during training.

CNN+LSTM: As above, images and questions are encoded using CNN features and final LSTM hidden states, respectively. These features are concatenated and passed to an MLP that predicts an answer distribution. CNN+LSTM+MCB: Images and questions are encoded as above, but instead of concatenation, their features are pooled using compact multimodal pooling (MCB) [55]. CNN+LSTM+SA: Again, the question and image are encoded using a CNN and LSTM, respectively. Following [56], these representations are combined using one or more rounds of soft spatial attention and the final answer distribution is predicted with an MLP.

Human: Mechanical Turk has been used to collect human responses for 5500 random questions from the test set, taking a majority vote among three workers for each question.

The CLAVR's CNNs are ResNet-101 models pre trained on ImageNet [57] that are not fine tuned; images are resized to 224×224 prior to feature extraction. CNN+LSTM+SA extracts features from the last layer of the conv4 stage, giving 14 × 14 × 1024-dimensional features. All other methods extract features from the final average pooling layer, giving 2048-dimensional features. LSTMs use one or two layers with 512 or 1024 units per layer. MLPs use ReLU functions and dropout ; they have one or two hidden layers with between 1024 and 8192 units per layer. All models are trained using Adam [58]. CLEVR is split into train, validation, and test sets (see Figure 6) and tuned hyperparameters (learning rate, dropout, word vector size, number and size of LSTM and MLP layers) independently per model based on the validation error. All experiments were designed on the validation set; after finalizing the design the team ran each model once on the test set. All experimental findings generalized from the validation set to the test set.

Performance analysis of different types of questions in CLEVR

Querying questions: Query questions ask about an attribute of a particular object (e.g. "What color is the thing right of the red sphere?"). The CLEVR world has two sizes, eight colors, two materials, and three shapes. On questions asking about these different attributes, Q-type mode and LSTM obtain accuracies close to 50%, 12.5%, 50%, and 33.3% respectively, showing that the dataset has minimal question-conditional bias for these questions. CNN+LSTM+SA substantially outperforms all other models on these questions; its attention mechanism may help it focus on the target object and identify its attributes.

Comparing attributes: Attribute comparison questions ask whether two objects have the same value for some attribute (e.g. "Is the cube the same size as the sphere?"). The only valid answers are "yes" and "no". Q-Type mode and LSTM achieve accuracies close to 50%, confirming there is no dataset bias for these questions.

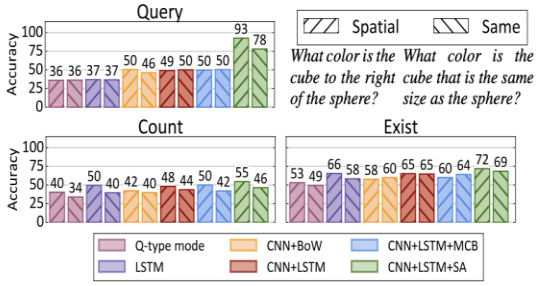

Figure 15: Accuracy on questions with a single spatial relationship vs. a single same-attribute relationship. For query and count questions, models generally perform worse on questions with same attribute relationships. Results on exist questions are mixed[6].

Existence questions: Existence questions ask whether a certain type of object is present (e.g., "Are there any cubes to the right of the red thing?"). The 50% accuracy of QType mode shows that both answers are a priori equally likely, but the LSTM result of 60% does suggest a question conditional bias.

Counting: Counting questions ask for the number of objects fulfilling some conditions (e.g. "How many red cubes are there?"); valid answers range from zero to ten. Images have three and ten objects and counting questions refer to subsets of objects, so ensuring a uniform answer distribution is very challenging; our rejection sampler, therefore, pushes towards a uniform distribution for these questions rather than enforcing it as a hard constraint. This results in a question-conditional bias, reflected in the 35% and 42% accuracies achieved by Q-type mode and LSTM. CNN+LSTM(+MCB) performs on par with LSTM, suggesting that CNN features contain little information relevant to counting. CNN+LSTM+SA performs slightly better, but at 52% its absolute performance is low.

Outcome of CLEVRER models

Methods	Descriptive	Explanatory		Predictive		Counterfactual	
		per opt.	per ques.	per opt.	per ques.	per opt.	per ques.
Q-type (random)	29.2	50.1	8.1	50.7	25.5	50.1	10.3
Q-type (frequent)	33.0	50.2	16.5	50.0	0.0	50.2	1.0
LSTM	34.7	59.7	13.6	50.6	23.2	53.8	3.1
CNN+MLP	48.4	54.9	18.3	50.5	13.2	55.2	9.0
CNN+LSTM	51.8	62.0	17.5	57.9	31.6	61.2	14.7
TVQA+	72.0	63.3	23.7	70.3	48.9	53.9	4.1
Memory	54.7	53.7	13.9	50.0	33.1	54.2	7.0
IEP (V)	52.8	52.6	14.5	50.0	9.7	53.4	3.8
TbD-net (V)	79.5	61.6	3.8	50.3	6.5	56.1	4.4
MAC (V)	85.6	59.5	12.5	51.0	16.5	54.6	13.7
MAC (V+)	86.4	70.5	22.3	59.7	42.9	63.5	25.1

Figure 16: Question-answering accuracy of visual reasoning baselines on CLEVRER. All models are trained on the full training set. The IEP (V) model and TbD-net (V) use 1000 programs to train the program generator[7].

The above figure 16 shows experimental results on CLEVRER database with different models. The fact that these models achieve different performances over the wide spectrum of tasks suggests that CLEVRER offers powerful assessment of the models' strengths and limitations in various domains. All models are trained on the training set until convergence, tuned on the validation set and evaluated on the test set. Baseline evaluations on CLEVRER have revealed two key elements that are essential to causal reasoning: an object-centric video representation that is aware of the temporal and causal relations between the objects and events; and a dynamics model able to predict the object dynamics under unobserved or counterfactual scenarios. The figure-16 below shows the performance of Neuro Symbolic Dynamic Reasoning.

On descriptive questions, our model achieves an 88.1% accuracy when the question parser is trained under 1,000 programs, outperforming other baseline methods. On explanatory, predictive, and counterfactual questions, our model achieves a more significant gain.

Methods	Descriptive	Explanatory		Predictive		Counterfactual	
		per opt.	per ques.	per opt.	per ques.	per opt.	per ques.
NS-DR	88.1	87.6	79.6	82.9	68.7	74.1	42.2
NS-DR (NE)	85.8	85.9	74.3	75.4	54.1	76.1	42.0

Figure 17: Quantitative results of NS-DR on CLEVRER. They evaluate models on all four question types. They also study a variation of our model NS-DR (NE) with "no events" but only motion traces from the dynamics predictor. The question parser is trained with 1000 programs[7].

Since NSAI is a proven breakthrough in the field of AI, it is being widely used in different sectors listed as following:

Healthcare

In the health domain, the challenge is to build a transparent artificial intelligence. Among the fields which use AI techniques, is Drug Repurposing (DR) which involves finding a new indication for an existing drug[59]. In this work [60], authors describe the use of neuro-symbolic algorithms in

order to explain the process of link prediction in a knowledge graph-based computational drug repurposing. Link prediction consists of generating hypotheses about the relationships between a known molecule and a given target.
In another article, in [61], author proposes a novel probabilistic programmed deep kernel learning approach that works well in predicting cognitive decline in Alzheimer's disease. The author shows that the proposed method is a powerful tool in neurodegenerative disease diagnosis and prognostic modeling and they suggest that the system will perform well in other disease areas.

Finance

For analysing the characteristics of applicants, authors in [62] present the design, implementation and evaluation of intelligent methods that assess bank loan applications where two separate intelligent systems have been developed and evaluated: a fuzzy expert system and a neuro-symbolic expert system. The former employs fuzzy rules based on knowledge elicited from experts. The later one is based on neurules, a type of neuro-symbolic rules that combines a symbolic and a connectionist representation. Neurules were produced from available patterns.

Recommender Systems

A system, called Neuro-Symbolic Interpretable Collaborative Filtering, has been proposed in article [63] that learns interpretable recommendation rules (consisting of user and item attributes) based on neural networks with two innovations: (1) a three-tower architecture tailored for the user and item sides in the RS domain; an improved architecture that is tailored for the user and item sides in recommendation; (2) fusing the powerful personalized representations of users and items to achieve adaptive rule weights and without sacrificing interpretability.

Cyber Security

A genetic algorithm is proposed to find combinatorial optimization of logic programmed constraints and deep learning from given components, which are rule-based and neural components. The genetic algorithm explores numerous searching spaces of combinations of rules with deep learning to get an optimal combination of the components[64].

NSAI is also being widely utilized in the development of Image processing, Business management, Brain modeling, Information Retrieval, Language generation, Question answering, Dialogue system, Education system Robotic, Smart city, and Safe machine learning[59].

5. Conclusion

In this paper, we have discussed the history of Neuro Symbolic Artificial Intelligence (NSAI) and its evolution. Although NSAI was invented in the 1950s, it got researchers' intentions recently. We conducted this survey to explore the Current and past breakthroughs in this field. The CLEVR and the CLEVRER datasets have worked as catalysts in the adoption of NSAI approach especially in Visual Questions Answering (VQA) tasks. Besides this, IBM is conducting a wide research With the collaboration of different researchers in different sub-area of NSAI including LNN, NLP via reasoning, KF, LwL. In this research, we have explored different datasets for NSAI VQA tasks and analyzed model performance on those datasets.
We have found Near Symbolic models achieved better accuracy on the recently developed CLEVR dataset compared with Other earlier models even when only 10% of the dataset has been used in the training. Further, models, developed for CLEVRER dataset containing 10,000 videos, Performed better in VQA tasks with greater performance. Hence, we deduce the existing problem of AI, called generalization, has been addressed for VQA tasks.
Now that researchers are able to combine neural and symbolic approaches together, there is a strong hope in the Future that the NSAI will be able to perform more complex tasks. By combining the best of two systems, a new AI System can be created which requires less data to train and demonstrate common sense, thereby accomplishing More complex tasks.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

ORCID iD (if any) : 0000-0002-7584-2224

References

[1] Marcus G. (2020) *The next decade in AI: Four steps towards robust artificial intelligence. CoRR, abs/1801.00631*
[2] Raghavan S. (2020) *AI predictions for IBM research. https://www.ibm.com/blogs/research/2019/12/2020-ai-predictions/*
[3] Mao J. (2019) "*The neurosymbolic concept learner: Interpreting scenes, words, and sentences from natural supervision,"*
[4] Mascharka D. (2018) *"Transparency by design: Closing the gap between performance and interpretability in visual reasoning," CoRR, vol. abs/1803.05268, [Online]. Available: http://arxiv.org/abs/1803.05268*
[5] Hudson D. A. (2018) *"Compositional attention networks for machine reasoning," in International Conference on Learning Representations, 2018. [Online]. Available: https://openreview.net/ forum?id=S1Euwz-Rb*
[6] Johnson J. (2016) *"CLEVR: A diagnostic dataset for compositional language and elementary visual reasoning," CoRR, vol. abs/1612.06890, [Online]. Available: http://arxiv.org/abs/1612.06890*
[7] Gan K. (2020) *"Clevrer: Collision events for video representation and reasoning," Available: arxiv.org/abs/1910.01442*
[8] Han C. (2020) *"Visual concept metaconcept learning,"*
[9] IBM, *Neuro Symbolic AI blog IBM. Available here: https://ibm.github.io/neuro-symbolic-ai*
[10] IBM, *Neuro-Symbolic AI toolkit. Available here: https://ibm.github.io/neuro-symbolic-ai/toolkit*
[11] IBM, *LNN Github. Available here:IBM/LNN: A `Neural = Symbolic` framework for sound and complete weighted real-value logic (github.com)*
[12] IBM, IBM LNN. *Available here: FOL-LNN – Neuro-Symbolic AI (ibm.github.io)*
[13] IBM, *Answer type prediction. Available here:Answer Type Prediction – Neuro-Symbolic AI (ibm.github.io)*
[14] IBM, *Answer type prediction github. Available here: IBM/answer-type-prediction: This system can predict the knowledge base types of the answer to a given natural language question. (github.com)*
[15] IBM, *Available here:IBM/expressive-reasoning-graph-store: Expressive Reasoning Graph Store (ERGS) is an OWL reasoner and an RDF triple store built on top of a Property Graph architecture. (github.com)*

[16] IBM, *OWL RL Reasoner. Available here: OWL RL Reasoner (ldf.fi)*
[17] IBM, *Learning with Less. Available here:Jan22: Day 2 Session 3 - Learning With Less (ibm.com)*
[18] Overko R. *RLAPSE. Available here: roman1e2f5p8s/rlapseingym: Reinforcement Learning with Algorithms from Probabilistic Structure Estimation (RLAPSE) (github.com)*
[19] IBM, *Learning Rates for Q-learning. Available here:evendar03a.dvi (jmlr.org)*
[20] Google HITL. *Available here: Human-in-the-Loop Overview | Document AI | Google Cloud*
[21] Johnson, J (2016) *CLEVR: A Diagnostic Dataset for Compositional Language and Elementary Visual Reasoning. Available here:https://arxiv.org/pdf/1612.06890.pdf*
[22] Johnson, J. (2015) *Image retrieval using scene graphs.*
[23] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L. Jia-Li, D. Shamma, M. Bernstein, and L. Fei-Fei. (2016) *Visual genome: Connecting language and vision using crowdsourced dense image annotations. IJCV*
[24] Blender Online Community.(2016) *Blender - a 3D modelling and rendering package. Blender Foundation, Blender Institute, Amsterdam*
[25] Yi, C. Gan, Y. Li, P. Kohli, J. Wu, A. Torralba, and J. B. Tenenbaum (2020) *"Clevrer: Collision events for video representation and reasoning," . Available here: 1910.01442.pdf (arxiv.org).*
[26] Erwin Coumans. (2010) *Bullet Physics Engine. Open Source Software: http://bulletphysics. org,*
[27] Blender Online Community. Blender - a 3D modelling and rendering package. Blender Foundation, Blender Institute, Amsterdam, 2016. URL http://www.blender.org
[28] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. (2015) *VQA: Visual Question Answering.*
[29] Guangyu R. Yang, Igor Ganichev, Xiao-Jing Wang, Jonathon Shlens, and David Sussillo.(2018) *A Dataset and Architecture for Visual Reasoning with a Working Memory.*
[30] Rowan Zellers, Yonatan Bisk, Ali Farhadi, and Yejin Choi (2019) *From recognition to cognition: Visual commonsense reasoning.*
[31] Chen Zhu, Yanpeng Zhao, Shuaiyi Huang, Kewei Tu, and Yi Ma. (2017) *Structured Attentions for Visual Question Answering.*
[32] Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Li Fei-Fei, C. Lawrence Zitnick, and Ross Girshick. (2017) *CLEVR: A Diagnostic Dataset for Compositional Language and Elementary Visual Reasoning.*
[33] Drew A Hudson and Christopher D Manning. (2019) *GQA: A new dataset for real-world visual reasoning and compositional question answering.*
[34] Jang Y.(2017) *GIF-QA: Toward Spatio-Temporal Reasoning in Visual Question Answering.*
[35] Makarand Tapaswi, Yukun Zhu, Rainer Stiefelhagen, Antonio Torralba, Raquel Urtasun, and Sanja Fidler.(2016) *MovieQA: Understanding Stories in Movies through Question-Answering.*
[36] Jonghwan Mun, Paul Hongsuck Seo, Ilchae Jung, and Bohyung Han.(2017) *MarioQA: Answering Questions by Watching Gameplay Videos.*
[37] Jie Lei, Licheng Yu, Mohit Bansal, and Tamara L Berg.(2018) *TVQA: Localized, compositional video question answering. In EMNLP,*
[38] Amir Zadeh, Michael Chan, Paul Pu Liang, Edmund Tong, and Louis-Philippe Morency. (2019) *Social-IQ: A question answering benchmark for artificial social intelligence.*
[39] Ramakrishna Vedantam et al. (2019) *"Probabilistic Neural Symbolic Models for Interpretable Visual Question Answering". In: Proceedings of the 36th International Conference on Machine Learning. Ed. by Kamalika Chaudhuri and Ruslan Salakhutdinov. Vol. 97. Proceedings of Machine Learning Research. PMLR, June 2019, pp. 6428–6437. Available here: http://proceedings.mlr.press/v97/vedantam19a.html.*
[40] Jacob Andreas et al.(2015) *Neural Module Networks*
[41] Tarek R. Besold et al. (2017) *"Neural-Symbolic Learning and Reasoning: A Survey and Interpretation". Available here: http://arxiv.org/abs/1711.03902.*
[42] Hamilton K. (2022) *Is Neuro-Symbolic AI Meeting its Promise in Natural Language Processing? A Structured Review.*
[43] B. MacCartney, *Understanding Natural Language Understanding, ACM SIGAI Bay Area Chapter Inaugural Meeting, San Mateo, CA. https://www.youtube.com/watch?v=vcPd0V4VSNU.*
[44] B. MacCartney and C.D. Manning, (2009) *An extended model of natural logic, in: Proceedings of the Eight International Conference on Computational Semantics, Association for Computational Linguistics, Tilburg, The Netherlands, pp. 140–156.*
[45] A. Gatt and E. Krahmer, (2018) *Survey of the State of the Art in Natural Language Generation: Core tasks, applications and evaluation, Journal of Artificial Intelligence Research 61 , 65–170. doi:10.1613/jair.5477.*
[46] B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D.M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever and D. Amodei (2020) *Language Models are Few-Shot Learners (2020). doi:10.48550/ARXIV.2005.14165.*
[47] Jacob Andreas et al.(2015) Neural Module Networks.
[48] Marie-Catherine de Marneffe and Christopher D. Manning.(2008) *"The Stanford Typed Dependencies Representation". In: Coling 2008: Proceedings of the Workshop on Cross-Framework and Cross-Domain Parser Evaluation. CrossParser '08. Manchester, United Kingdom: Association for Computational Linguistics, 2008, pp. 1–8. isbn: 9781905593507. Available here: https://dl.acm.org/doi/10.5555/1608858.1608859.*
[49] Ronghang Hu et al.(2017) *"Learning to Reason: End-To-End Module Networks for Visual Question Answering". In: Proceedings of the IEEE International Conference on Computer Vision (ICCV).*
[50] Karen Simonyan and Andrew Zisserman. (2015) *"Very Deep Convolutional Networks for Large-Scale Image Recognition". In: 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, ,Conference Track Proceedings. Ed. by Yoshua Bengio and Yann LeCun. 2015. url: http://arxiv.org/abs/1409.1556.*
[51] Iab Berlot A. (2021) *Neuro-Symbolic VQA: A review from the perspective of AGI desiderata. Available here:2104.06365.pdf (arxiv.org)*
[52] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. Zitnick, and D. Parikh. (2015) *VQA: Visual question answering.*
[53] B. Zhou, Y. Tian, S. Sukhbataar, A. Szlam, and R. Fergus.(2015) *Simple baseline for visual question answering. In arXiv:1512.02167*
[54] T. Mikolov, K. Chen, G. Corrado, and J. Dean. (2013) *Efficient estimation of word representations in vector space. In arXiv 1301.3781*
[55] A. Fukui, D. H. Park, D. Yang, A. Rohrbach, T. Darrell, and M. Rohrbach.(2016) *Multimodal compact bilinear pooling for visual question answering and visual grounding. In arXiv:1606.01847*
[56] Z. Yang, X. He, J. Gao, L. Deng, and A. Smola.(2016) *Stacked attention networks for image question answering*.
[57] K. He, X. Zhang, S. Ren, and J. Sun. (2016) *Deep residual learning for image recognition.*
[58] D. Kingma and J. Ba. Adam: A method for stochastic optimization. In ICLR, 2015.
[59] Djallel Bouneffouf, Charu C. Aggarwal, (2022) *Survey on Applications of Neurosymbolic Artificial Intelligence. Available here: arxiv.org/pdf/2209.12618.pdf*
[60] Dranc´e M., (2022) *"Neuro-symbolic xai: Application to drug repurposing for rare diseases," in International Conference on Database*

Systems for Advanced Applications, pp. 539–543, Springer,

[61] Lavin, A. (2021) *"Neuro-symbolic neurodegenerative disease modeling as probabilistic programmed deep kernels," in International Workshop on Health Intelligence, pp. 49–64. Springer.*

[62] Ioannis Hatzily, J. Prentzas, (2011) *"Fuzzy and neuro-symbolic approaches to assessment of bank loan applicants," in Artificial Intelligence Applications and Innovations, pp. 82–91, Springer.*

[63] Zhang W. (2022) *Neuro-Symbolic Interpretable Collaborative Filtering for Attribute-based Recommendation. Available here: dl.acm.org/doi/10.1145/3485447.3512042.*

[64] K. W. Park, S.-J. Bu, and S.-B. Cho,(2021) *"Evolutionary optimization of neuro-symbolic integration for phishing url detection," in International Conference on Hybrid Artificial Intelligence Systems, pp. 88–100, Springer.*