

Script-based Inferences in an Image Schema Story Understander

Jamie C. Macbeth^{1,*}, Boming Tony Zhang² and Sharmin Badhan³

¹Department of Computer Science, Smith College, 10 Elm Street, Northampton, Massachusetts, 01063, USA

²Manning College of Information and Computer Sciences, University of Massachusetts, Amherst,
140 Governors Drive, Amherst, Massachusetts, 01003, USA

³Independent Researcher, USA

Abstract

Recent studies of large language models (LLMs) have revealed that they lack human-like cognitive models of reasoning and understanding. An important thread of research merges image schemas into symbolic artificial intelligence systems where their use as conceptual building blocks and primitives shows promise for the study of human cognition through intelligent systems that perform neurosymbolically. The work presented in this paper demonstrates image schema primitives being used in structures of a representation system called *conceptual dependency* (CD) and in broader commonsense knowledge structures called *scripts*. We present a story understanding system that uses image schemas as primitives in scripts that encode stereotypical sequences of events within familiar contexts, such as dining at a restaurant or visiting a doctor. We explain the content and structure of an image schema script, and demonstrate the Image Schema Script Applier (ISSA) as it processes a story and performs anaphora resolution, inference, and summarization.

Keywords

Image Schemas, Scripts, Story Understanding

1. Introduction

Recent studies of large language models (LLMs) reveal the ways in which they lack human-like cognitive models of reasoning and understanding [1, 2, 3]. A thread of research which merges image schemas into symbolic artificial intelligence systems where they serve as conceptual building blocks and primitives shows promise for the study of human cognition through intelligent systems that perform neurosymbolically.

The work presented in this paper demonstrates image schema primitives being used in structures of a representation system called *conceptual dependency* (CD) [4, 5]. The CD framework supports inference and paraphrase by expressing meaning in a language-independent, structured form that reveals conceptual relationships between entities and events. CD has been used in broader commonsense knowledge structures called *scripts* [6], which encode stereotypical sequences of events within familiar contexts, such as dining at a restaurant or visiting a doctor. The Script Applier Mechanism (SAM) [7, 8] used CD-based scripts successfully for natural language understanding and story understanding to represent events and acts and the actors and objects involved in those acts.

In this paper, we explore structures of a representation system that combines image schemas and conceptual dependency, which we call IS-CD, and use them to create script structures. We present an example script structure composed of a sequence of IS-CD conceptualizations which have image schema primitives as their central events and conceptual dependency case frames to specify actors, objects, directions, and other aspects of the event. We also present the Image Schema Script Applier (ISSA), a rebuild of the original Script Applier Mechanism. ISSA uses image schema-based scripts to

Proceedings of the Joint Ontology Workshops (JOWO) - Episode XI: The Sicilian Summer under the Etna, co-located with the 15th International Conference on Formal Ontology in Information Systems (FOIS 2025), September 8-9, 2025, Catania, Italy

*Corresponding author.

✉ jmacbeth@smith.edu (J. C. Macbeth); bomingzhang@umass.edu (B. T. Zhang); sharminbadhan79@gmail.com (S. Badhan)

🌐 <https://jamiemacbeth.com> (J. C. Macbeth)

🆔 0000-0003-3474-203X (J. C. Macbeth)



© 2025 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

process a narrative posed in natural language and demonstrates its understanding and inferencing capabilities through summary generation. This work demonstrates image schemas playing a major part in artificial intelligence systems and structures for language understanding and reasoning.

The paper has the following structure: Section 2 presents background on image schemas, conceptual dependency, scripts, and script applier mechanism systems. Sections 3 and 4 introduce the Image Schema Script Applier and explain in detail its processing of a natural language story using an example image schema script. Section 5 discusses related work, and the paper concludes with Section 6, a discussion of future work.

2. Background

2.1. Image Schemas and Conceptual Dependency

Early work in artificial intelligence developed systems for in-depth understanding of natural language using the conceptual dependency theory of meaning representation structures. Originally developed by Schank [4, 5] and popularized by Minsky under the term “Trans-Frames” [9], conceptual dependency (CD) represents the meaning of natural language by abstracting away from surface syntax and focusing on underlying conceptual structures relying on a small set of abstract primitives. This thread of work evolved independently from the cognitive linguistics literature on image schemas, which reflect patterns for understanding and reasoning formed through bodily and sensorimotor experiences [10].

More recent work on conceptual modeling has explored mappings between image schemas and conceptual dependency. Macbeth, Gromann, and Hedblom [11] investigated how these systems relate, especially in representing spatial and physical concepts, and they find that several CD primitives that represent acts such as moving and ingesting correspond closely to image schemas like `CONTAINMENT` and `SOURCE_PATH_GOAL`. The comparison opens possibilities for refining CD by merging or simplifying its components based on image schema theories. Additionally, the connections between image schemas and CD opens possibilities for testing image schema theories by implementing them within artificial intelligence systems in place of CD primitives.

In the original CD, a conceptual structure can be composed from one of eleven primitive ACTs: `PROPEL`, `MOVE`, `INGEST`, `EXPEL`, `GRASP`, `PTRANS`, `ATRANS`, `SPEAK`, `ATTEND`, `MTRANS`, and `MBUILD` [5]. At the conceptual level, CD encodes meaning using networks of interrelated elements: Picture Producers (PPs) for entities, Action Primitives (ACTs) for basic actions, and modifiers called Picture Aiders (PAs) and Action Aiders (AAs), which describe attributes of objects and actions, respectively. These elements are linked through conceptual dependencies that specify how one concept contributes to the interpretation of another. The structure of a CD conceptualization is also governed by a set of conceptual rules. For instance, a central rule states that a conceptualization must involve an ACT and a PP in a two-way dependency, indicating that both elements are essential for the event to be meaningful. Other rules define how attributes can be predicated of concepts, how objects are related to actions, and how conceptual relations such as containment or possession are encoded. In this work, we create CD conceptualization structures conforming to the conceptual cases and rules of CD, but using image schemas as the ACT primitives.

2.2. Scripts and The Script Applier Mechanism

Scripts, as introduced by Schank and Abelson [6], are cognitive structures that encode stereotypical sequences of events within familiar contexts, with dining at a restaurant being the best known example. These structured scenarios consist of scenes, roles, and props linked by temporal and causal relationships. Scripts support efficient comprehension by enabling individuals to infer background events and disambiguate language based on prior experience. For instance, hearing “I left a tip” activates a restaurant script that implies a meal and payment occurred. This framework has been influential in cognitive science and AI, offering a model for how systems can interpret narratives by leveraging structured, experience-based knowledge. Scripts are one mechanism through which CD conceptualizations can

relate to each other through higher-level conceptual relations like causality, allowing complex events to be represented as sequences or chains of interdependent actions and states.

The Script Applier Mechanism (SAM) [7, 8] is an early computational model developed to simulate human story understanding by using scripts as structured, context-dependent knowledge. It processes natural language input by identifying relevant scripts and making inferences to fill narrative gaps through predefined causal chains and role expectations. The original SAM integrates components such as a conceptual analyzer (the English Language Interpreter, or ELI [12]), a memory module known as PP-Memory [8], and a script application system to convert text into meaning representations and generate outputs such as summaries and answers to questions [13]. Although limited in processing speed and domain coverage, SAM established foundational principles in natural language understanding by showing how structured world knowledge and inference strategies such as causal chain completion, role instantiation, and role merging support coherent interpretation of text. The original script applier mechanism could process stories presented in English, and generate summaries of stories in English, Spanish, and Chinese. SAM could also provide answers to questions about stories which were posed in English. In this paper we present a rebuild of SAM which utilizes image schemas in its knowledge structures.

3. The Image Schema Script Applier

In this section we describe the Image Schema Script Applier (ISSA), its components, and the script application process. We provide an example of a script composed of image schema CD (IS-CD) structures, and an example of the script being applied to process a brief natural language story. We also show how the script applier mechanism performs inferences and generates summaries and paraphrases of the story that it has processed.

3.1. An Image Schema Script

In the original conceptions of scripts [6], they are composed of “causal chains”, sequences of acts and events represented in a language-free conceptual representation. Scripts in the original SAM also had multiple “scenes” and “tracks” within scenes that represent the typical affairs in that situation or place. For example, the restaurant script had scenes for the different phases of activity: scenes for ordering, eating, and paying, with the scenes having multiple tracks or paths, often representing the different ways of accomplishing the activity. The original script applier mechanism was supplied with multiple scripts for understanding stories about a variety of subjects, from car accidents, train wrecks, and oil spills, to diplomatic meetings between ministers of foreign affairs. We based our implementation of SAM and our example of it on detailed logs from Cullingford’s Ph.D. dissertation which showed the inner workings of SAM as it read a brief story about a car crash [8].

Table 1 shows a sequence of acts from a script structure that represents knowledge about car accidents. The script is inspired by a script called \$VEHACCIDENT from Cullingford’s work [8]. The original \$VEHACCIDENT had three subscenes corresponding to the crash itself, the treatment of the crash victims, and (if relevant) the investigation of the accident. The subscenes in the crash scene of the vehicle accident script have multiple causal chains that represent the many ways and reasons that a vehicle can end up in a crash and the various objects that it can crash into. Other tracks or subscenes could handle variations in which, for instance, a vehicle hits another vehicle, or where a vehicle hits a pedestrian.

The example we present represents a partial path through subscenes of \$VEHACCIDENT in which a vehicle leaves a road and hits an obstacle on the side of the road. Table 1 shows the causal chain. The script consists of a special kind of conceptual dependency structure called a *pattern* which has variables in some locations of the structures. Each script pattern also has a list containing the pattern IDs of other patterns which are predicted to be mentioned soon afterward. These lists appear in the “Predicted” column.

Pattern ID	Image Schema	Pattern Cases	Predicted
CRA1	SOURCE_PATH_GOAL	ACTOR &VEHICLE OBJECT &VEHICLE	CRA2, CRA3
CRA2	SUPPORT	ACTOR &LINK OBJECT &VEHICLE	CRA3
CRA3	SOURCE_PATH_GOAL	ACTOR &VEHICLE OBJECT &VEHICLE FROM &LINK	CRA4
CRA4	FORCE	ACTOR &VEHICLE OBJECT &OBSTACLE	TRE1, TRE2
TRE1	CONTAINMENT	ACTOR &AMBVEHICLE OBJECT &HURTGRP	TRE2
TRE2	SOURCE_PATH_GOAL	ACTOR &AMBVEHICLE OBJECT &HURTGRP TO &HOSPOG	

Table 1

A script pattern chain for a subscene of the car accident script which matches stories that involve a vehicle leaving a road and striking an obstacle and an injured party being taken to the hospital. The script is composed of a sequence of patterns which can be matched against language-free story representations which are the outputs of a conceptual analysis of the input story language. Conceptual dependency primitives have been replaced by image schemas, but each pattern retains the conceptual cases which are common in CD structures. The patterns contain script variables which are assigned in the matching process.

The first pattern, CRA1, represents the car in motion. In a conceptual dependency-based script, this pattern would have been a PTRANS act, representing that a object or being changed its location. However, in IS-CD, based on an established mapping between CD primitives and image schemas [11], we represent CRA1 using a SOURCE_PATH_GOAL image schema. The conceptual structures that make up the script pattern retain the conceptual cases which are common in CD structures. In conceptual dependency, PTRANS acts usually have an ACTOR case representing the animate being that performed the act and an OBJECT case representing the thing that moved. In CRA1, both the ACTOR and the OBJECT are the script variable &VEHICLE to indicate that the vehicle is “moving itself.” In CD, the ACTOR and OBJECT case may be different beings or objects in situations where one animate being is responsible for changing the location of another thing or being. In CD, PTRANS additionally has TO and FROM conceptual cases to represent the direction of the movement. Since the TO and FROM conceptual cases are not part of the pattern and they do not have any script variables associated with them, the pattern will successfully match any TO and FROM cases in the input structure.

The second pattern, CRA2, represents a SUPPORT relationship between the vehicle and the road or other surface that it is traveling on. Here the ACTOR is a script variable named &LINK. Interestingly, while LINK is a known image schema [10], in earlier work on script understanding systems, LINK refers to an abstract class of picture producers that are objects that connect locations together. In Cullingford’s description, roads, train tracks, ship channels, and other paths are LINKs [8]. The script variable in the car crash script which is usually assigned to these kinds of picture producers (in English expressions such as “Route 9” or “Elm Street”) is called &LINK as well.

CRA3 is a second SOURCE_PATH_GOAL conceptualization which represents the car leaving the road, which could be matched to English verbs such as “veer” or “swerve”. As with the earlier SOURCE_PATH_GOAL, this appeared in CD scripts as a PTRANS act, and both the ACTOR and OBJECT are the &VEHICLE , again to indicate that no external force is causing the vehicle’s motion. There is also a FROM case which indicates that the motion is away from the &LINK.

In the original CD script conception, the next pattern, CRA4, would have been a PROPEL conceptualization referring to the car colliding with an obstacle. Earlier work [11] mapped the PROPEL CD primitive to the FORCE image schema. Therefore, here, in Table 1, CRA4 is a FORCE image schema con-

Story Input	Conceptual Analysis	Script Pattern Match	Variable Assignments
"A car swerved off the road."	SOURCE_PATH_GOAL ACTOR CAR OBJECT CAR FROM ROAD	CRA3	&VEHICLE → CAR &LINK → ROAD
"It hit a tree."	FORCE ACTOR PHYSOBJECT OBJECT TREE	CRA4	&OBSTACLE → TREE
"The driver went to the hospital ..."	SOURCE_PATH_GOAL ACTOR DRIVER OBJECT DRIVER TO HOSPITAL	TRE2	&HURTGRP → DRIVER &HOSPORG → HOSPITAL
"... in an ambulance."	CONTAINMENT ACTOR AMBULANCE OBJECT DRIVER	TRE1	&AMBVEHICLE → AMBULANCE

Table 2

Shows language-free story representations which are the outputs of a conceptual analysis of the input story language being matched against a script pattern chain for a subscene of the car accident script. The story involves a vehicle leaving a road and striking an obstacle and an injured party being taken to the hospital. The conceptual analysis typically would have conceptual dependency primitives, but here they have been replaced by image schemas. Each conceptual structure retains the conceptual cases which are common in CD structures. The "Variable Assignments" column shows how script variables are assigned in the matching process.

ceptualization which retains &VEHICLE as the ACTOR case and a script variable named &OBSTACLE as the OBJECT case. The &OBSTACLE variable will match to picture producers capable of damaging a vehicle in a collision, such as trees, walls, poles and the like.

The final two patterns in the script are part of a subscene in which one or more persons involved in the accident are treated for injuries. TRE1 represents one or more persons going into or being put into an ambulance. In CD there is a predicate called CONTAIN which is mapped to the CONTAINMENT image schema. While CONTAIN is not one of the eleven primitive acts of CD, it may be used to indicate containment relationships between picture producers. TRE1 has a conceptualization based on the CONTAINMENT image schema with the ACTOR case representing the containing object and the OBJECT case representing the contained object.

In this case, a script variable &AMBVEHICLE represents the vehicle that is transporting injured persons, and &HURTGRP is a variable that can represent one or more injured persons which are being transported. TRE2 represents the ambulance vehicle, &AMBVEHICLE, transporting the injured persons, &HURTGRP, to the hospital organization location, &HOSPORG. In the original SAM, the existence of these variables allows the script to handle sentences from newspaper stories which specified the ambulance and hospital organizations such as "[the driver] was taken to Milford Hospital by Flanagan Ambulance."

3.2. The Script Application Process

The original SAM system "read" various types of newspaper stories. Here we demonstrate ISSA's script application process as our car accident script is applied to a brief story about a car accident. The story below is reminiscent of a newspaper story from the *New Haven Register* which was processed by the original SAM [8, 7]. The story involves a vehicle going off a road and striking an obstacle, and then an injured party being taken to the hospital. The story is:

“A car swerved off the road. It hit a tree. The driver went to the hospital in an ambulance.”

All of the sentences in the story correspond directly to patterns in the car accident script, with “a car swerved off the road” corresponding to CRA3 and “it hit a tree” corresponding to CRA4 (see Table 1). The last sentence has parts that correspond to two different patterns in the script with “the driver went to the hospital” corresponding to TRE2 and “... in an ambulance” corresponding to TRE1. However there are patterns in the script which do not match with any particular statements in the story.

The Image Schema Script Applier performs the following steps in processing the story. First a sentence of the story is fed to the conceptual analyzer system, which analyzes the natural language input to produce an initial language-independent conceptual representation. The column labeled “Conceptual Analysis” in Table 2 shows conceptual dependency representations which are the outputs of a conceptual analysis of sentences and phrases from the input story.

In the original SAM, the output of a conceptual analysis would have been conceptual dependency representations which had primitive acts and conceptual “cases” which could indicate the actor or object of an act or the specification of a directionality of a movement. In CD structures, the conceptual cases are often filled by *picture producer* elements that represent objects, human story actors, and locations (also called *PPs*). As with the scripts, in the image schema version of conceptual analysis, the representations are a hybrid; they are CD structures which have, as primitive acts, image schemas in place of the CD primitives. ISSA uses a version of a conceptual analyzer called CA [14, 15] which originally created CD structures but has been modified to produce IS-CA structures with image schemas as primitives.

Next, in the most important part of the script applier mechanism understander’s process, a conceptual representation from the conceptual analysis is matched against conceptual representation patterns in the script. The original SAM [8] was furnished with multiple scripts and had a *script activation* process which simulated how a human understander with a large store of commonsense knowledge determines which script knowledge structures to call into memory for understanding the story. In contrast, in this simple demonstration of image schema script application, our simplified car accident script has already been activated previously to the first sentence being analyzed, and it has been pre-determined that the vehicle accident script patterns will be used for matches.

As stated above, in SAM, scripts are represented as sequences of conceptual dependency structures (called *patterns*) which have *script variables* in some locations of the structures where a picture producer would have been expected. In the basic mechanism, ISSA attempts to match the structure from the input against a pattern in the script that represents one of the events expected to come next in the story. The match takes into account the image schema which is in the typical position of the CD primitive act, and the actor, object, to, and from cases in the IS-CD structure. Some patterns do not have all of the possible conceptual “cases”. In that situation the pattern will match with anything that an IS-CD structure has for that particular case. When there is a successful pattern match, ISSA binds any script variables in the pattern to picture producers in the IS-CD structure. The “Variable Assignments” column in Table 2 shows how script variables are assigned in the matching process. We enhanced a script matcher from the Common Lisp version of Micro SAM [16] to perform the matching.

The script applier checks the story input CD structure against patterns in the script in the order that they appear in a *search list* maintained by the script applier [8]. At the start of the script matching process, the search list is initialized to contain the patterns in the script’s causal chain order. To reflect script expectations of what events are likely to appear next in the story, when a particular pattern is matched, the system reorders the search list to bring pattern IDs in the “predicted” list of the matched pattern to the beginning. This allows conceptual analyses on sentences and phrases in the story to match with patterns even if they appear slightly “out of order”. This turns out to be the case for the third and last sentence of the story. The conceptual analysis results in two different CD structures, the first representing “the driver went to the hospital ...”, and matching with TRE2, the second representing “... in an ambulance”, and matching with TRE1. The process also performs anaphora resolution. “It” in the second sentence appears as PHYSOBJECT in the conceptual analysis, and is “merged” with the &VEHICLE script variable in the script application process.

Event ID	Image Schema	CD Cases	Output Text
EVNT1	SOURCE_PATH_GOAL	ACTOR CAR OBJECT CAR FROM ? TO ?	“A car was moving ...”
EVNT2	SUPPORT	ACTOR ROAD OBJECT CAR	“ ... and a road supported the car”
EVNT3	SOURCE_PATH_GOAL	ACTOR CAR OBJECT CAR FROM ROAD TO ?	“The car left the road ...”
EVNT4	FORCE	ACTOR CAR OBJECT TREE	“ ... and ran into a tree.”
EVNT5	CONTAINMENT	ACTOR AMBULANCE OBJECT DRIVER	“The driver was in an ambulance ...”
EVNT6	SOURCE_PATH_GOAL	ACTOR DRIVER OBJECT DRIVER FROM ? TO HOSPITAL	“... and went to the hospital.”

Table 3

A story representation built by ISSA based on the script patterns and script variable assignments and summarized output text generated from the representation. ISSA makes inferences as interpolations or extrapolations by instantiating event structures corresponding to script patterns which were not matched to conceptualizations in the input story.

3.3. Story Representation and Summary Generation

Once the script application process is complete, the script applicer builds a full story representation. It does this by *instantiating* all of the patterns in the script and replacing script variables with their bindings. This includes patterns which matched the story as well as patterns that are present in the script but did not match any of the CD structures created in the conceptual analysis of the story input. Because the story representation contains conceptual structures which were part of the script but not part of the story, these structures comprise inferences of facts and events that the understander completes which were not explicitly stated in the story. We generated instantiations using code (the `instantiate` function) from a Common Lisp version of Micro SAM [16]. Table 3 shows the story representation built by SAM based on its processing of the story in Table 2 using the script in Table 1.

The system is also able to generate natural language summaries of stories based on the full story representation. For this, we used an enhanced version of Neil Goldman’s BABEL system [13], which generates natural language from non-linguistic conceptual structures. The Image Schema Script Applier system is able to generate a summary of the story which paraphrases the original and includes inferences that are the result of the script application process. Here is an example:

A car was moving and the road supported the car. The car left the road and ran into a tree.
The driver was in an ambulance and went to the hospital.

The “Output Text” column of Table 3 shows the sentences and phrases generated by BABEL based on the story representation. The story understanding structures labeled EVNT1 and EVNT2 corresponding to “a car was moving and the road supported the car,” are inferences of events and spatial relationships based on the script application.

4. Discussion

Combining image schemas with conceptual dependency primitives in a story understander provides new and interesting opportunities for juxtaposing the two systems. The patterns in scripts in the original SAM were mainly focused on conceptual dependency events and acts which brought about changes in the world, such as movement and state change. More recent work juxtaposing image schemas with conceptual dependency primitives [11, 17, 18] has raised the importance of spatial relationships in primitive decomposition representations. In image schema-based scripts, one point of interest is that we have an occurrence of a “static” spatial relationship being represented in a script pattern, which we believe would have been rare or nonexistent in the original SAM. For example, CRA2 (Table 1) has a SUPPORT image schema to represent the car being on the road and being supported by it, while TRE1 has a CONTAINMENT image schema to represent injured persons being in the ambulance. In the case of CRA2, this stretches considerably the convention that each pattern in a script sequence is causally linked to the next or happens temporally before the next, since the SUPPORT of the car by the road is conceived to be happening simultaneously with the SOURCE_PATH_GOAL event of CRA1.

During the development of the IS-CD script structures, we considered options for representing the road that the vehicle is traveling on. We may have been able to represent the road as an abstract space above the road surface so that we could use a CONTAINMENT image schema to represent the car being contained in that space. The original SAM also had representations of “settings” and “locales” where events occurred. For example, it would create a CD structure using a conceptual dependency LOC predicate to indicate the location where the car ran into the obstruction as being somewhere near the road that the car was traveling on. We considered representing the setting of the accident, the larger area around the road that contains the setting of the accident, and the hospital as larger spatial objects and using CONTAINMENT or LOCATION image schemas to indicate that other objects in the story were located in or at these spaces. This would align with recent work on primitive decompositions of spatial relationships [17], but we were discouraged to attempt this at such an early stage because of our concern that the BABEL generator might produce awkward-sounding texts such as “the ambulance went into the scene of the accident”.

A story about a car accident raises significant questions regarding the ACTOR case of the conceptualizations, since the engine which is propelling the vehicle is part of the vehicle, but, also, presumably, there is a person driving. The original SAM partially addressed this issue by having an embedded \$DRIVE script inside the \$VEHACCIDENT script [8]. Ideally there should be additional patterns in the script to represent driving. In both of the SOURCE_PATH_GOAL conceptual structures, the ACTOR and the OBJECT are the same, which is supposed to mean that the object was “moving itself”. In the case of the driver of the crashed vehicle going to the hospital in an ambulance this should not imply that the driver was driving the ambulance. One possible resolution would be to use an AGENCY image schema primitive [19]. Analyzing and representing ACTOR roles in these conceptualizations in consistent ways is a topic of ongoing research.

5. Related Work

An important genre of work maps, compares, and merges image schemas with conceptual dependency and uses the connections to evolve the set of CD primitives that CD-based AI systems use for in-depth understanding. In [11], Macbeth, Gromann, and Hedblom, investigate the relationship between image schemas and CD, two frameworks used to represent meaning in natural language understanding. Image schemas come from cognitive linguistics and reflect patterns formed through sensorimotor experiences, while conceptual dependency primitives are from artificial intelligence and aim to model human-like understanding through a limited set of abstract actions. The considerable overlap that they find suggests that CD can be grounded more firmly in cognitive theory and that image schemas may benefit from formal structuring provided by CD.

Other related work reveals that some CD primitives are potentially redundant, as they can be

expressed as combinations of other primitives. Macbeth and Gromann [20] investigate the potential of using a formal logic framework based on image schemas to represent the primitives of conceptual dependency. By applying a formal system called Image Schema Logic, which integrates elements of spatial and temporal logic, they show that complex CD primitives such as INGEST and EXPEL can be modeled using simpler, more general components like movement, containment, and direction. The study concludes that this approach could streamline the CD inventory and improve formal representations of language understanding in AI systems.

Macbeth et al. [18] take this work further. They explore removing the INGEST conceptual primitive from the set of CD primitives and replacing its use with combinations of other CD primitives, namely PTRANS and CONTAIN, which are the analogs of SOURCE_PATH_GOAL and CONTAINMENT image schemas. In this work, the BABEL system [13] proved effective in generating paraphrases that reveal how image schemas and CD primitives can be combined and contrasted, offering insight into improving CD's cognitive alignment. The results strongly support replacing INGEST with PTRANS and set the stage for similar future analyses, including the decomposition of EXPEL, INGEST's conceptual opposite. These threads of work seek to make the set of conceptual primitives more compact, increasing the richness and expressiveness of the primitive-decomposed meaning structures in ways that better corresponded with human cognition and its capability for complex mappings and manipulations of meaning structures.

The current paper combines image schemas with scripts in ways that allow for alternative forms of logical deduction and inference. Enhancements of Micro SAM [16] that combine multiple scripts have also been used for in artificial intelligence systems for story generation [21] and cyberbullying prevention [22]. In relation to the script applier's inference capability, Hedblom et al. [23] examine how image schemas can be used to represent complex events in a formal and meaningful way through Image Schema Logic, which enables the formalization of these schemas and their interactions over time and space. They examine and arrange image schemas sequentially to represent the progressions of events, a way to capture the dynamics of everyday actions in a way that relates to human reasoning. Other related work uses image schemas for logical inferences [24].

In the same way that scripts combine CD conceptualizations into more complex structures, there has been related work on combining image schemas. Hedblom et al. [25] demonstrate how image schema profiles can effectively represent the conceptualization of events. Drawing on research in event segmentation and cognitive linguistics, they show that clusters of image schemas can capture conceptualizations in particular linguistic contexts. The paper introduces three different characters by which image schemas can be combined: merge, collection and sequence. Again using Image Schema Logic (ISL), the authors illustrate how everyday actions like dropping an egg and cracking an egg into a bowl can be decomposed into schema-driven event segments such as SUPPORT, SOURCE_PATH_GOAL, CONTAINMENT, and SPLITTING. The resulting collection of formalized image schemas serves as a cognitively grounded repository of ontology design patterns for modeling event conceptualizations in intelligent systems. Besold, Hedblom, and Kutz [26] provide illustrations and a proof of concept for how the image schemas OBJECT, CONTACT, and PATH are combined in a temporal dimension to form more complex image schemas and simple events, specifically BLOCKAGE, BOUNCING, and CAUSED_MOVEMENT. The authors also present an outline of a proposed conceptual hierarchy of levels of modeling for image schemas and similar cognitive theories.

In related work on image schemas, scripts, and narrative, Ranta [27] explores how pictorial storytelling, especially in static images such as paintings, communicates narrative meaning through cognitive structures like schemas and scripts. Like script-based inferencing for natural language understanding, the work highlights how understanding a pictorial narrative often involves the viewer filling in missing information using prior experience, cultural context, and interpretive expectations. Wicke and Veale [28] present a novel approach to exploring image schemas in the realm of computational embodied storytelling and propose a framework that utilizes a storytelling system to explore the causal connections between image schemas. For this investigation, a system has been implemented on a Nao humanoid robot, which has a set of 9 prominent image schemas with more than 800 story actions (plot verbs) from the storytelling system. Kimmel [29] argues that story understanding involves a mental simulation of

interaction between image schemas and, in [30], demonstrates how image schemas are fundamental to understanding narratives, both in figurative and literal forms. Image schemas such as PATH, CONTAINER, FORCE, BALANCE, and PART-WHOLE structure the basic spatial, temporal, and causal-intentional logic of events, forming the foundation of story comprehension.

6. Conclusion and Future Work

In this paper we presented a script applier system based on image schemas and showed how it could process a brief natural language story and perform inferencing and understanding. This shows both how the system is able to perform script-based inferences based on image schemas, but also how it forms a language-independent representation of understanding. The findings also move toward unifying image schemas with CD, grounding the latter in cognitive linguistics and experimental psychology while supporting cognitive AI applications.

One important issue that remained unexplored in this work is the question of how to use ontological knowledge in the script application process. Conceptual dependency theory provides little in the way of characteristics and primitives to describe or represent the characteristics of objects when they are picture producers in conceptual structures. The original SAM performed a *rolefit* process as part of matching a script pattern to a story structure. Rolefit ascribed a “class” and “type” to all PPs (for example, making “car” a #STRUCTURE structured object of TYPE *CAR*) and compared these to sets of classes associated with the &VEHICLE script variable. Image schema research may provide building blocks for representing PPs in a richer way.

The original SAM also had many more scripts than the system presented in this paper. Scripts in the original SAM were larger and had multiple “tracks” and “subscenes”. Future work can explore image schemas in the script activation process and the simultaneous application of multiple scripts when understanding a story.

Having a script applier mechanism for both image schemas and conceptual dependency enriches the types of studies that can be performed with the two theories. For example, the issues with the CD ACTOR case in the context of the SOURCE_PATH_GOAL representing the movement of a vehicle being controlled by an intelligent being could be resolved with further studies of stories about driving. Further decompositions of the image schema CD structures’ ACTOR case using an AGENCY image schema could be studied with the inferences and paraphrases that are generated using scripts from different systems. Other possible topics of future work include turning ISSA on its head and making it a story generator, studies of uncertain or non-monotonic reasoning and question answering, and integrations with large language models.

Acknowledgments

We acknowledge Larry Birnbaum and Mallory Selfridge for their development of the original CA conceptual analyzer. We thank Mark Burstein for providing a version of the original CA code and transcoding it to Common Lisp, Neil Goldman for providing original code for BABEL, and Richard Cullingford for helpful discussions about SAM.

Declaration on Generative AI

The author(s) have not employed any Generative AI tools in writing this paper.

References

- [1] S. Greengard, Shining a light on AI hallucinations, *Commun. ACM* 68 (2025) 9–11. URL: <https://doi.org/10.1145/3715691>. doi:10.1145/3715691.

- [2] I. Mirzadeh, K. Alizadeh, H. Shahrokhi, O. Tuzel, S. Bengio, M. Farajtabar, GSM-symbolic: Understanding the limitations of mathematical reasoning in large language models, 2024. URL: <https://arxiv.org/abs/2410.05229>. arXiv:2410.05229.
- [3] L. Berglund, M. Tong, M. Kaufmann, M. Balesni, A. C. Stickland, T. Korbak, O. Evans, The reversal curse: LLMs trained on "A is B" fail to learn "B is A", in: The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024, OpenReview.net, 2024. URL: <https://openreview.net/forum?id=GPKTIktA0k>.
- [4] R. C. Schank, Conceptual dependency: A theory of natural language understanding, *Cognitive Psychology* 3 (1972) 552–631.
- [5] R. C. Schank, *Conceptual Information Processing*, Elsevier, New York, NY, 1975.
- [6] R. C. Schank, R. P. Abelson, *Scripts, Plans, Goals and Understanding: An Inquiry into Human Knowledge Structures*, Lawrence Erlbaum Associates, Mahwah, NJ, 1977.
- [7] R. E. Cullingford, Pattern-matching and inference in story understanding, *Discourse Processes* 2 (1979) 319–334.
- [8] R. E. Cullingford, *Script Application: Computer Understanding of Newspaper Stories*, Ph.D. thesis, Yale University, New Haven, CT, 1977.
- [9] M. Minsky, *Society of Mind*, Simon & Schuster, New York, 1988.
- [10] J. M. Mandler, C. Pagán Cánovas, On defining image schemas, *Language and Cognition* 6 (2014) 510–532.
- [11] J. C. Macbeth, D. Gromann, M. M. Hedblom, Image schemas and conceptual dependency primitives: A comparison, in: *Proceedings of The Joint Ontology Workshops, Episode 3: The Tyrolean Autumn of Ontology*, The International Association for Ontology and its Applications, Bolzano-Bozen, Italy, 2017.
- [12] C. K. Riesbeck, An expectation-driven production system for natural language understanding, in: D. A. Waterman, F. Hayes-Roth (Eds.), *Pattern-Directed Inference Systems*, Elsevier, New York, 1978, pp. 399–413.
- [13] N. M. Goldman, Sentence paraphrasing from a conceptual base, *Communications of the ACM* 18 (1975) 96–106.
- [14] L. Birnbaum, M. Selfridge, Conceptual analysis of natural language, in: R. C. Schank, C. K. Riesbeck (Eds.), *Inside Computer Understanding: Five Programs Plus Miniatures*, Lawrence Erlbaum Associates, Hillsdale, NJ, 1981, pp. 318–353.
- [15] L. Birnbaum, M. Selfridge, *Problems in Conceptual Analysis of Natural Language*, Research Report #168, Yale University, Department of Computer Science, New Haven, CT, 1979.
- [16] R. C. Schank, C. K. Riesbeck, Micro SAM, in: *Inside Computer Understanding: Five Programs Plus Miniatures*, Lawrence Erlbaum Associates, Hillsdale, NJ, 1981, pp. 120–135.
- [17] M. Zhou, B. Duah, J. C. Macbeth, Novel primitive decompositions for real-world physical reasoning, in: K. R. Thórisson (Ed.), *Proceedings of the Third International Workshop on Self-Supervised Learning*, volume 192 of *Proceedings of Machine Learning Research*, PMLR, 2022, pp. 22–34. URL: <https://proceedings.mlr.press/v192/zhou22a.html>.
- [18] J. C. Macbeth, A. Kilayko, Z. Zhao, S. Song, W. X. Zheng, Image schema decompositions of the conceptual dependency ingest primitive: A study of paraphrases, in: *Proceedings of The Seventh Image Schema Day (ISD7)*, The International Association for Ontology and its Applications, Rhodes, Greece, 2023.
- [19] J. M. Mandler, How to build a baby: Ii. conceptual primitives., *Psychological review* 99 (1992) 587.
- [20] J. C. Macbeth, D. Gromann, Towards modeling conceptual dependency primitives with image schema logic, in: *The Fourth Workshop on Cognition And Ontologies (CAOS IV) at The Fifth Joint Ontology Workshop (JOWO'19)*, The International Association for Ontology and its Applications, Graz, Austria, 2019.
- [21] M. McKenzie, A. Kilayko, J. C. Macbeth, S. Carter, K. Sieck, M. Klenk, Script combination for enhanced story understanding and story generation systems, in: *Proceedings of the Tenth Annual Conference on Advances in Cognitive Systems*, The Cognitive Systems Foundation, Arlington, VA, 2022.

- [22] J. Macbeth, H. Adeyema, H. Lieberman, C. Fry, Script-based story matching for cyberbullying prevention, in: CHI 2013 Extended Abstracts: ACM SIGCHI Conference on Human Factors in Computing Systems, Paris, France, 2013.
- [23] M. M. Hedblom, O. Kutz, R. Peñaloza, G. Guizzardi, Image schema combinations and complex events, *KI - Künstliche Intelligenz* 33 (2019) 279–291. doi:10.1007/s13218-019-00605-1.
- [24] M. M. Hedblom, O. Kutz, T. Mossakowski, F. Neuhaus, Between contact and support: Introducing a logic for image schemas and directed movement, in: F. Esposito, R. Basili, S. Ferilli, F. A. Lisi (Eds.), *AI*IA 2017 Advances in Artificial Intelligence*, Springer International Publishing, Cham, 2017, pp. 256–268.
- [25] M. M. Hedblom, O. Kutz, R. Peñaloza, G. Guizzardi, What's cracking? how image schema combinations can model conceptualisations of events, in: *Proceedings of The Fourth Image Schema Day (ISD4)*, Bolzano, Italy, 2018.
- [26] T. R. Besold, M. M. Hedblom, O. Kutz, A narrative in three acts: Using combinations of image schemas to model events, *Biologically Inspired Cognitive Architectures* 19 (2017) 10–20. doi:10.1016/j.bica.2016.11.001.
- [27] M. Ranta, The role of schemas and scripts in pictorial narration, *Semiotica* 201 (2021) 1–27. doi:10.1515/sem-2019-0071.
- [28] P. Wicke, T. Veale, Wheels within wheels: A causal treatment of image schemas in an embodied storytelling system, in: *ISD4: Image Schema Day IV*, Bozen-Bolzano, 2018. URL: <https://api.semanticscholar.org/CorpusID:123769744>.
- [29] M. Kimmel, From metaphor to the "mental sketchpad": Literary macrostructure and compound image schemas in heart of darkness, *Metaphor and Symbol* 20 (2005) 199–238. URL: https://doi.org/10.1207/s15327868ms2003_3. doi:10.1207/s15327868ms2003_3. arXiv:https://doi.org/10.1207/s15327868ms2003_3.
- [30] M. Kimmel, Analyzing image schemas in literature, *Cognitive Semiotics* 9 (2009) 159–188. doi:10.3726/81609_159.