The Tensor Brain: A Unified Theory of Perception, Memory, and Semantic Decoding

Volker Tresp
volker.tresp@lmu.de
LMU Munich and Siemens Munich, Germany

Sahand Sharifzadeh
sahand.sharifzadeh@gmail.com
LMU Munich, Germany

Hang Li
hang.li@campus.lmu.de
LMU Munich and Siemens Munich, Germany

Dario Konopatzki
dk@dkonopatzki.de
LMU Munich, Germany

Yunpu Ma
cognitive.yunpu@gmail.com
LMU Munich and Siemens Munich, Germany

We present a unified computational theory of an agent’s perception and memory. In our model, both perception and memory are realized by different operational modes of the oscillating interactions between a symbolic index layer and a subsymbolic representation layer. The two layers form a bilayer tensor network (BTN). The index layer encodes indices for concepts, predicates, and episodic instances. The representation layer broadcasts information and reflects the cognitive brain state; it is our model of what authors have called the “mental canvas” or the “global workspace.” As a bridge between perceptual input and the index layer, the representation layer enables the grounding of indices by their subsymbolic embeddings, which are implemented as connection weights linking both layers. The propagation of activation to earlier perceptual processing layers in the brain can lead to embodiments of indices. Perception and memories first create subsymbolic representations, which are subsequently decoded semantically to produce sequences of activated...
indices that form symbolic triple statements. The brain is a sampling engine: only activated indices are communicated to the remaining parts of the brain. Triple statements are dynamically embedded in the representation layer and embodied in earlier processing layers: the brain speaks to itself. Although memory appears to be about the past, its main purpose is to support the agent in the present and the future. Recent episodic memory provides the agent with a sense of the here and now. Remote episodic memory retrieves relevant past experiences to provide information about possible future scenarios. This aids the agent in decision making. “Future” episodic memory, based on expected future events, guides planning and action. Semantic memory retrieves specific information, which is not delivered by current perception, and defines priors for future observations. We argue that it is important for the agent to encode individual entities, not just classes and attributes. Perception is learning: episodic memories are constantly being formed, and we demonstrate that a form of self-supervised learning can acquire new concepts and refine existing ones. We test our model on a standard benchmark data set, which we expanded to contain richer representations for attributes, classes, and individuals. Our key hypothesis is that obtaining a better understanding of perception and memory is a crucial prerequisite to comprehending human-level intelligence.

1 Introduction

With an increase in higher animals’ abilities to move and act came a growing demand for high-performing perceptual systems, beyond simple labeling of entities with attributes and classes (Hommel, Müßeler, Aschersleben, & Prinz, 2001). This might have been a driving force to develop episodic and semantic memory. Episodic memory engrams permit the recall of recent and remote memories and provide guidance for acting right. Semantic memory engrams provide concept grounding and complement perceived information with background knowledge about concepts. The agent does not just acquire new skills but gains and memorizes knowledge: it learns about and remembers things in the world. Memories support the agent in the present and the future; without memory systems, the brain is literally memoryless.

Episodic memories retrieve previous experiences in an agent’s life. Recent episodic memory permits the agent to remember the immediate past since some state information cannot be directly derived from perceptual input. A recall is triggered by nearness in time and relevance. It has evolved such that the agent can remember where it has been before, why it is where it is, and what the general context is. Recent episodic memory is an episodic memory that is almost treated as a current observation. For instance, the agent needs to remember that even though perception does not give a clue, it is still in
the hideout because the bear had been chasing it and might still be lurking outside. *Remote episodic memory* can remind an agent about past situations—similar to the current one—and imminent danger and favorable actions associated with that situation. It provides estimates about possible future scenarios and aids the agent in decision making. Continuing the previous example, the agent might remember previous personal bear encounters and subsequent dangerous situations. Recall in remote episodic memory is triggered by closeness between episodic representation and scene representation. Finally, we define *future episodic memories* as events that are expected to be a memory in the future. Information on future events influences planning and action.

Semantic memory models the time-invariant statistics of statement probabilities and acts as their prior. It is a dictionary view of the agent’s life.\(^1\) It aggregates entity information, which might have been acquired at different episodes, and this information becomes local in the graph formed by semantic memory. Semantic memory enables multimodal integration. For example, if Sparky is discovered in a scene, semantic memory provides background information, for example, that Sparky is a young dog and is owned by Jack, and although dogs in general might be aggressive, Sparky is a friendly dog. Semantic memory support can be essential for survival. An agent simply knows that bears are dangerous, even when a bear looks cozy and sleepy and even if the agent did not yet have an unpleasant encounter with a bear. Semantic memory is the default semantic state estimator and is complemented by perception and recent episodic memories, when and where available. Our approach explains the great similarity between episodic and semantic memory: semantic memory is the expected episodic memory of a future instance.

An elementary symbolic statement says that an entity has a particular attribute and belongs to a specific class, or that an entity has a particular relationship with another entity. This motivates our assumption that basic facts are expressed as triple statements of the form \((\text{subject}, \text{predicate}, \text{object})\)\(^2\). In most languages, basic facts are expressed in this or a similar format. Thus, triple sentences are arguable of fundamental relevance for encoding and communicating perception, episodic memory, and semantic memory, all of which humans can easily describe by language. Communication is important to make the invisible visible; for example, an agent could inform peers that a bear is lurking outside the hideout even if it cannot be seen. Triple statements can represent relationships between entities, which enables, for example, rich scene descriptions and reasoning in social and spatial networks. An agent can understand not only that a bear and a deer are in a

---

\(^1\) Some authors distinguish between a formal dictionary and a grounded cognitive encyclopedia (Evans, 2012). We will not make this distinction.

\(^2\) In a relational Bayesian network or a Markov logic network, a triple statement is represented as a node; in a graph neural network, it would be a typical output node.
scene but that, luckily, the bear is not chasing the agent itself but the deer. Symbolic representations and reasoning, classical System-2 properties, depend on the representations of entity-to-entity relationships (Halford, Wilson, Andrews, & Phillips, 2014).

The starting point of our work is a mathematical model, the bilayer tensor network (BTN). The BTN implies a very simple architecture that contains two basic layers: the index layer and the representation layer. Context is provided by a third layer, the dynamic context layer, which interacts with the representation layer, provides context, and stores state information when the brain’s attention moves from one entity to another.

The symbolic index layer contains indices for concepts, predicates, and episodic instances known to the agent. The index layer labels the activation pattern in the representation layer and then feeds back the embedding of that label to the representation layer. The embedding vectors are implemented as connection weights linking both layers. An index is a focal point of activity and competes with other indices, but since it constantly interacts with the representation layer, it is never active in isolation. Embeddings have an integrative character; the embedding vector for a concept index integrates all that is known about that concept, and the embedding vector for an episodic index represents the world state at that instance. The subsymbolic representation layer is the main communication platform. In cognitive neuroscience, it would correspond to what authors call the “mental canvas” or the “global workspace” and reflects the cognitive brain state. In bottom-up mode, scene inputs activate the representation layer, which then activates the index layer. In top-down mode, an index activates the representation layer, which might subsequently activate even earlier processing layers. This last process is called the embodiment of a concept.

Perception and memories first produce subsymbolic representations. They represent the agent’s current and past cognitive state and are subsequently decoded semantically to produce sequences of activated indices that form symbolic triple statements. These sequences then give feedback to the representation layer and earlier processing layers and thus inform the brain as a whole about what has been decoded. The brain is a sampling engine: only activated indices are communicated to the remaining parts of the brain.

In our approach, perception and memory produce triple statements by stochastic sampling, which is a central bottleneck, as also discussed by Dehaene (2014). We introduce an attention approximation, which avoids intermediate sampling decisions on visual entities.

One can consider a graph where concepts are represented as nodes and triples become labeled directed links pointing from subject node to object node. In computer science, this would be called a knowledge graph (KG). Whereas symbolic reasoning would purely act on the graph, embedded reasoning, as realized by the BTN, involves concept embeddings and performs information propagation via the representation layer.
This article is organized as follows. In the next section, we cover related work. Section 3 introduces concepts, triple statements, and probabilistic memory models. In section 4, we present our proposed bilayer tensor network (BTN) and demonstrate how it realizes perception and the different memory functions. Section 5 presents and discusses the BTN’s algorithmic implementation, sampling, and attention approximations. The following sections present experimental results and discuss potential relationships to cognition and neuroscience. Section 6 introduces the data set and discusses perception, the representation layer, and the dynamic context layer. Section 7 discusses semantic memory engrams and their semantic decoding. Using social network data, we demonstrate multimodal integration in semantic memory. Section 8 compares embedded reasoning with symbolic reasoning and proposes that embedded symbolic reasoning performs symbolic reasoning using embedding vectors.

We introduce the one-brain hypothesis, which emphasizes, first, that the brain uses only one representation layer and, second, that perception, episodic memory, semantic memory, and embedded reasoning all rely on the same BTN architecture. We discuss relationships between our approach to cognitive linguistics and consciousness research. We argue that the internal triple-oriented fast speech is transformed into an external, sophisticated slow speech. Section 9 covers episodic memory and the way the agent estimates current and future world states. In section 10, we discuss that perception involves the storing of new episodic memories and show how a form of self-supervised learning can learn new concepts and refine existing ones. Section 11 contains our conclusions.

2 Related Work

2.1 Tensor Networks for Modeling Knowledge Graphs. The bilayer tensor network (BTN) is an example of a tensor network. RESCAL was the first tensor-based embedding model for triple prediction in relational data sets and knowledge graphs (Nickel, Tresp, & Kriegel, 2011, 2012). Embedding learning for knowledge graphs evolved into a sprawling research area (Bordes, Usunier, Garcia-Duran, Weston, & Yakhnenko, 2013; Socher, Chen, Manning, & Ng, 2013; Yang, Yih, He, Gao, & Deng, 2014; Nickel, Rosasco, & Poggio, 2015; Trouillon, Welbl, Riedel, Gaussier, & Bouchard, 2016; Dettmers, Minervini, Stenetorp, & Riedel, 2018). Nickel, Murphy, Tresp, and Gabrilovich (2015) provides an overview and PyKEEN (Ali et al., 2021) a comprehensive software library. In contrast to previous approaches, the BTN can be implemented as an interaction between an index layer and a representation layer, and thus it is more suitable for brainware implementations.

2.2 Cognitive Tensor Networks and Related Models. Tensors have been used previously as memory models where the main focus was on
simple associations (Hintzman, 1984; Kanerva, 1988); Humphreys, Bain, & Pike, 1989; Osth & Dennis, 2015) and compositional structures (Smolensky, 1990; Pollack, 1990; Plate, 1997; Halford, Wilson, & Phillips, 1998; Ma, Hildebrandt, Baier, & Tresp, 2018). In the tensor product approach (Smolensky, 1990), encoding or binding is realized by a tensor product (generalized outer product) and compositionality by tensor addition. In the structured tensor analogical reasoning (STAR) model (Halford et al., 1998), a working memory is constructed by a superposition of tensor products. Either standard basis (one-hot) or random vectors are used as representations for concepts and predicates. Some basic symbolic reasoning operations are implemented, such as analogical and transitive inference. The main difference to our work is that we approximate the data tensor by a tensor network, that is, the BTN. This permits generalization and realizes embedded reasoning, as well as the seamless integration of sensory input.

The application of embedding-based tensor models to neurocognitive models started with Tresp, Esteban, Yang, Baier, and Krompaß (2015), which introduced tensor networks with index embeddings for perception, as well as semantic and episodic memory. It did not explicitly consider scene bounding boxes and did not contain experimental results. In further work, the connection between temporal and semantic tensor networks was analyzed (Tresp & Ma, 2016; Tresp, Ma, & Baier, 2017; Tresp, Ma, Baier, & Yang, 2017; Ma, Tresp, & Daxberger, 2018). Semantic memory was derived from episodic memory by an integration step performed in latent space. In this article, we avoid explicit integration and extend the approach to include perception.

The neurocognitive tensor network theory evolved in the 1980s and followed the idea of geometrization of biology. It is a theory of brain function, particularly that of the cerebellum. Metric tensors transform sensory space-time coordinates into motor coordinates (Pellionisz & Llinás, 1980). There, tensor networks are used very differently from the work presented here.

2.3 Visual Relationship Detection and Scene Graphs. The Stanford Visual Relationship data set, published in 2016, contained images annotated with triple sentences (Lu, Krishna, Bernstein, & Fei-Fei, 2016; Krishna et al., 2017). The two papers made their annotated data available, which spawned an explosion of research activity in visual relationship detection (VRD). The background information in Lu et al. (2016) was extracted from a text corpus. Recent work in this direction is Luo, Zhang, Han, and Yang (2019).

VRD models for knowledge graphs were proposed by Baier, Ma, and Tresp (2017), Zhang, Kyaw, Chang, and Chua (2017), and Baier, Ma, and Tresp (2018). Baier et al. (2017) showed how a prior distribution derived from triple occurrences could significantly improve on pure vision-based approaches and on approaches that used prior distributions derived
from language models. Sharifzadeh, Berrendorf, and Tresp (2019) showed further improvements by including 3D image information. Tresp, Sharifzadeh, and Konopatzki (2019) and Tresp, Sharifzadeh, Konopatzki, and Ma (2020) describe more recent publications in this tradition. The presented work introduces more clearly the different operational modes and provides more extensive experimental results.

Triple statements generated from an image form a scene graph (Johnson et al., 2015). Work on scene graphs attempts to find a unique, globally optimal interpretation of an image. State-of-the-art scene graph models are described in Yang, Lu, Lee, Batra, and Parikh (2018), Zellers, Yatskar, Thomson, and Choi (2018), and Hudson and Manning (2019). Sharifzadeh, Baharlou, and Tresp (2020) captures the interplay between perception and semantic knowledge by introducing schema representations and implementing the classification as an attention layer between image-based representations and the schema.

2.4 Related Modern Technical Models for Memory. Hochreiter and Schmidhuber (1997) convincingly demonstrated the importance of memory systems in recurrent neural networks. Important later extensions are the neural Turing machine (NTMs; Graves, Wayne, & Danihelka, 2014) and memory networks (Weston, Chopra, & Bordes, 2014; Sukhbaatar, Weston, & Fergus, 2015). In those three papers, episodic memory acts as an instance buffer. Both use a recurrent neural network in combination with attention mechanisms. Large language models, like OpenAI’s GPT-3, are more recent developments (Brown et al., 2020).

In our BTN, a dynamic context layer is part of a (nonstandard) recurrent neural network. The attention mechanisms in our model, episodic attention and semantic attention, are quite different from the attention mechanisms in those papers. Also, our goal is to derive triple statements, whereas in those models, the tasks are query answering and improved token embeddings.

2.5 Dual Process Theory and Complementary Learning Systems (CLS). In psychology, dual process theory concerns the interplay in the mental processing of an implicit, automatic, unconscious process (shared with animals) and an explicit, controlled, conscious process (uniquely human). (See Evans, 2003, for a review.) In our model, the implicit side would be on the level of embeddings and representations, whereas the explicit side is on the level of the concept indices and the extracted triple sentences.

One instance of a dual process theory is Kahneman’s System-1/System-2 dichotomy (Kahneman, 2011). System-1 is fast, instinctive, and emotional and does not require mental effort. System-2 is slower, more deliberate, and more logical and requires mental effort.

CLARION is a dual-process model of implicit and explicit learning (Sun & Peterson, 1996). It is based on one-shot explicit rule learning (i.e., explicit learning) and gradual implicit tuning (i.e., implicit learning).
A different but related dichotomy can be found in the complementary learning systems (CLS) theory (McClelland, McNaughton, & O’Reilly, 1995; Kumaran, Hassabis, & McClelland, 2016), where the formation of time indices and their embeddings would be part of a nonparametric learning system centered on the hippocampus, which allows rapid learning of the specifics of individual items and experiences (Kumaran et al., 2016). Slow training would be part of a parametric learning system, which serves as the basis for the gradual acquisition of structured knowledge about the environment to neocortex (Kumaran et al., 2016). In this article, we introduce self-supervised learning for rapid learning and discuss a consolidation process of learned knowledge. McClelland, Hill, Rudolph, Baldridge, and Schütze (2020) analyzed the connection between memory, perception, and language, which is also a focus of our article.

### 2.6 The Bayesian Brain

Our approach can be related to the tradition of Bayesian approaches to brain modeling (Dayan, Hinton, Neal, & Zemel, 1995; Rao & Ballard, 1999; Knill & Pouget, 2004; Körding, Ku, & Wolpert, 2004; Tenenbaum, Griffiths, & Kemp, 2006; Griffiths, Kemp, & Tenenbaum, 2008; Friston, 2010). In our approach, all past experiences train the perceptual system and also contribute to perception via memory. In Baier et al. (2017) an explicit semantic prior distribution was used, describing a priori probabilities for triple sentences. For inference, Bayes’ formula is used. The remarkable improvement in performance after integrating the prior information indicates that triple representations might be a powerful abstraction level for formulating prior knowledge in general. Sharifzadeh et al. (2020) showed that the probabilistic knowledge graph acts as an inductive bias in perception. It discusses the role of a prior as an integrator of multimodal information and its role in filling in nonperceptual background information.

In the work presented here, we separately model and perform completion on the observed statements, statement priors, and statement posteriors. This is in the context of Bayesian probabilistic inference but not in the sense of Bayesian statistics.

In a Bayesian brain approach, top-down connections are often associated with predictive models; they provide predictions about future state probabilities. Here, we emphasize that in addition, top-down connections inform the representation layer and earlier processing layers—what the brain has detected.

### 3 Concepts, Triple Statements, and Probabilistic Memory Models

*You only see what you know.*

—Johann Wolfgang von Goethe, an Friedrich von Müller, April 24, 1819
3.1 Triple Statements. To understand what is perceived, an agent needs to have an understanding of the things in the world and their relationships. We assume that the agent’s mind is aware of \( N_C \) concepts \( C = \{c_1, \ldots, c_{N_C}\} \). A concept can, for example, represent an entity \( e \in E \subset C \) or a class \( k \in K \subset C \), which stands for a collection of entities, or an attribute \( b \in B \subset C \).

In addition to concepts, we also consider a set of predicates \( p \in P = \{p_1, \ldots, p_{N_P}\} \), where \( N_P \) is the number of predicates the agent is aware of. A triple statement has the form \((s, p, o)\), where \( s \in C \) assumes the role of the subject, \( o \in C \) assumes the role of the object, and predicate \( p \in P \). Examples of triple statements are \((\text{Munich}, \text{partOf}, \text{Bavaria})\), \((\text{Sparky}, \text{looksAt}, \text{Jack})\), \((\text{AkiraKurosawa}, \text{directorOf}, \text{SevenSamurai})\), and \((\text{Jack}, \text{knows}, \text{Mary})\).

Following the notion of the semiotic triangle of Ogden and Richards (1923), we can look at triples and their semantics from three different perspectives:

1. The agent-independent objective world (world semantics). Some triples have an interpretation in the real world, and they stand for propositions, like \((\text{Munich}, \text{partOf}, \text{Bavaria})\) and \((\text{Sparky}, \text{looksAt}, \text{Jack})\). This view is mostly taken in a formal analysis of cognition and linguistics (Montague, 1970; Fodor, 1975) and is sometimes referred to as truth-conditionalsemantics. An agent is only aware of some of the concepts in the world, sometimes called the “projected world” or the “construal world.”

2. An agent’s mind and brain (an agent’s neurocognitive semantics). A triple is a statement based on concepts in an agent’s mind. These might have an immediate meaning in the objective world but could also be untestable statements with unclear world semantics such as \((\text{Love}, \text{strongerThan}, \text{Hate})\). A hypothesis of this article is that triple statements correspond to processes in the brain, that is, a sequential firing of index representations. They form the inner fast speech.

3. An agent’s utterance (an agent’s linguistic semantics). A triple is a simple language clause involving symbols. Within the approach in this article, we assume that any triple statement in the mind can also be expressed linguistically. But what is spoken, in general, represents external slow speech and is obviously more complex, nuanced, and sophisticated than the inner fast speech, and is modulated, for example, by intent, social context, and cultural background. In general, the relationship between statements in the mind (with concepts) and in language (with symbols) is a matter of an open debate (Evans, 2012).

We are aware that the relationship of the three perspectives touches on fundamental issues in cognition, linguistics, neuroscience, philosophy, and many other scientific fields and academic disciplines. A more detailed discussion would be beyond the scope of this article. This article assumes the position of the agent. We are concerned only about how the agent views the
world. We focus on personal Bayesian probabilities, modeling the agent’s neurocognitive semantics, where we restrict ourselves to statements that, in the mind of the agent, can be true or false, for example, \((\text{Sparky}, \text{ownedBy}, \text{Jack})\), and facts that can be related to observations, for example, \((\text{Sparky}, \text{hasColor}, \text{Black})\). This is more of a practical than a principled constraint. On some dimensions, thought and reality agree simply to act right and to guarantee survival; on others, there is likely no agreement between agents, no clear relationship to an objective reality, and in determining the truth values on statements, personal emotions and judgments might play a significant role.

3.2 Knowledge Graphs. In a knowledge graph (KG), a triple is represented as a directed labeled link from concept \(s \in C\) to concept \(o \in C\), where the link is labeled by \(p\). Thus, a labeled link represents a triple statement of the form \((s, p, o)\) where \(s\) is called head or source node and \(o\) is called tail or target node. Knowledge graphs currently have a great impact in applications and, as our approach, are entity oriented.

3.3 Unary and Binary Statements. When \(s\) and \(o\) are entities, \((s, p, o)\) stands for the ground atom \(p(s, o)\). \((\text{Munich}, \text{partOf}, \text{Bavaria})\) would stand for \(\text{partOf(Munich, Bavaria)}\).

We assume a strong default predicate \(\text{hasAttribute}\) (abbreviated as \(hA\)), which, depending on the subject type and the object type, can stand for certain other predicates from the set \(P_B \subseteq P\). We write \((s, hA, c)\), \(c \in C\):

- If both subject and object are entities, the implicit predicate is \(\text{sameAs}\); thus, \((\text{Jack}, hA, \text{John})\) stands for \((\text{Jack}, \text{sameAs}, \text{John})\).
- If the subject is an entity and the object is an attribute, the implicit predicate is attribute specific but should be obvious; thus, \((\text{Jack}, hA, \text{Tall})\) stands for \((\text{Jack}, \text{height}, \text{Tall})\).
- If the subject is an entity and the object is a class, the implicit predicate is \(\text{type}\), as in \((\text{Sparky}, \text{type}, \text{Dog})\).
- If both subject and object are classes, the implicit predicate is \(\text{subClass}\), as in \((\text{Dog}, \text{subClass}, \text{Mammal})\); note that the transitive \(\text{subClass}\) predicate permits the modeling of deep ontologies.

Triple sentences involving the \(\text{hasAttribute}\) predicate (and its substitutes) we call unary statements. The remaining \(N_B\) predicates from \(P_B \subseteq P\) form binary statements. We will refer to the object in a unary statement also simply as the unary label of the subject and to the predicate in a binary statement as the binary label of a concept pair. Binary statements are required when the default unary interpretation is not applicable, as in \((\text{Jane}, \text{motherOf}, \text{Jack})\) (the default would be \((\text{Jane}, \text{sameAs}, \text{Jack})\)).

Higher-order relations can be reduced to a set of binary statements, for example, by using additional concepts (Noy, Rector, Hayes, & Welty, 2006).
For example, \textit{match(Player1, Player2, Location)} becomes \textit{(matchID, hasPlayer, Player1)}, \textit{(matchID, hasPlayer, Player2)}, and \textit{(matchID, hasLocation, Location)}, where \textit{(matchID)} is the additional concept.

3.4 Semantic-State Model. In the assumed lifetime and the mind of an agent, some statements are always true, as \textit{(Munich, partOf, Bavaria)}, but the truth values of other statements can change in time, as \textit{(Munich, weather, Sunny)}. Thus, with each triple sentence, at each episodic instance \(t\), the agent associates a \textit{semantic state variable} \(Y_{s,p,o,t}\). If the agent is certain that \((s, p, o)\) is true at episodic instance \(t\), then \(Y_{s,p,o,t} = 1\), and if \((s, p, o)\) is observed to be false at episodic instance \(t\), then \(Y_{s,p,o,t} = 0\). The agent’s \textit{semantic state} at episodic instance \(t\) is defined as the states of all semantic state variables. Here, we assume that the agent is concerned with \(N_T\) past episodic instance \(T = \{t_1, \ldots, t_{N_T}\}\).

We consider that data, that is, information on the states on triple statements, arrive in chunks, which we call episodes. McClelland et al. (2020) call them situations and in cognition they are called event frames. In knowledge graphs, they could form a namespace. We use the terms \textit{events} and \textit{episodes} almost interchangeably, although in a narrower sense, we reserve the term \textit{episode} for a sequence of events. (See also the discussion in section 9.) Each episode provides information about the truth values of a subset of all statements. In this article, we assume that episodic data are either provided by a human annotator (supervised learning) or generated in self-supervised learning (see section 10). We can form a data tensor of observed triples as a four-mode triple-observation tensor of the form

\[
T = \sum_{s,p,o,t} y_{s,p,o,t} \mathbf{e}^s \otimes \mathbf{e}^p \otimes \mathbf{e}^o \otimes \mathbf{e}^t.
\]

Here, \(\mathbf{e}^i\) is our notation for a standard basis vector, that is, a one-hot vector, with the one at position \(i\). Such data tensors have also been the starting point for the tensor memories of Halford et al. (1998) and Ma, Hildebrandt et al. (2018). A tensor entry \(y_{s,p,o,t}\) is simply equal to one if the statement \((s, p, o)\) was observed by the agent to be true at \(t\) and equal to zero if the corresponding statement was observed or concluded by the agent to be false at \(t\); otherwise, it is unknown. The temporal KG (tKG) is the graph representation of \(T\), where the labeled links between concepts are time dependent. We assume that data are acquired from different modalities, like vision or language and that at a given episodic instance \(t\), data from only some modalities are collected. We invoke a local closed-world assumption (LCWA): we assume that for entities for which data are acquired in a modality, the truth values of all modality-specific unary statements are available. For entity pairs in a modality, we assume that the truth values of all modality-specific binary statements are available.
In the next section, we introduce the bilayer tensor network (BTN), which performs data completion. We obtain the semantic-state model

$$E(Y_{s,p,o,t}) \approx y_{s,p,o,t}. \quad (3.1)$$

It is an agent’s estimate of the semantic state at any episodic instance $t$ and generalizes to unobserved statements.

### 3.5 Triple Retrieval Model and Episodic Memory.

We also define a second four-mode tensor as

$$\tilde{T} = \sum_{s,p,o,t} i_{s,p,o,t} e^s \otimes e^p \otimes e^o \otimes e^t. \quad (3.2)$$

A tensor entry here is simply equal to one if the corresponding statement was observed by the agent to be true and is zero otherwise. We obtain $\tilde{T}$ from $T$ by setting all unknown entries in the latter to zero. The triple retrieval model provides a normalized $\tilde{T}$ as

$$\mathbb{P}(s, p, o, t) \approx \frac{1}{N_{total}} i_{s,p,o,t}. \quad (3.2)$$

It models the distribution of the indices. $N_{total} > 0$ is the total number of observed facts (true triple statements). The triple retrieval model is our model for the symbolic aspect of episodic memory. It models the personal experiences of the agent, whereas the semantic-state model is concerned about what is true or false.

From the definitions, it follows that for observed triples and where the LCWA applies

$$\mathbb{P}(s, p, o, t) \approx \frac{E(Y_{s,p,o,t})}{N_{total}}. \quad (3.2)$$

The agent can thus assume that for a given $t$, a sample generated from the triple retrieval model is true.

### 3.6 Priors and Semantic Memory.

The prior tensor is

$$S = \sum_{s,p,o} \mu_{s,p,o} e^s \otimes e^p \otimes e^o \quad (3.3)$$

where $\mu_{s,p,o} = \frac{1}{N_{s,p,o}} \sum_t y_{s,p,o,t}$. Here, $N_{s,p,o} > 0$ counts the truth values of $(s, p, o)$ are known, either because they were measured to be true or as concluded by the LCWA to be false. In the sum, unknowns are counted
as zeros. \( \mu_{s,p,o} \) is the average truth value, where the average is over the true and false triples. The prior model is

\[
\mathbb{E}(Y_{s,p,o,t}) \approx \mu_{s,p,o}.
\] (3.4)

We are using here that the mean over observations is an estimate of the semantic state at a generic future episodic instance \( \bar{t} \) for which data are not yet available. If \( N_{s,p,o} = 0 \), we rely on the generalization ability of the model. The prior retrieval tensor is

\[
\mathbb{T} = \sum_{s,p,o} \bar{i}_{s,p,o} \mathbf{e}^s \otimes \mathbf{e}^p \otimes \mathbf{e}^o,
\]

where \( \bar{i}_{s,p,o} = 1/N_{\text{total}} \sum_i i_{s,p,o,i} \). The prior retrieval model is

\[
P(s, p, o, \bar{t}) \approx \bar{i}_{s,p,o}.
\] (3.5)

The prior retrieval model is typically triggered by a concept, for example, an entity \( s \) that occurred in perception or in episodic memory recall, and provides background information on that concept. As we will discuss later, the prior retrieval model mathematically describes the symbolic aspect of semantic memory.

We have

\[
P(s, p, o, \bar{t}) \approx N_{s,p,o} \mathbb{E}(Y_{s,p,o,t})/N_{\text{total}}.
\]

Thus, the samples from the prior retrieval model reflect the prior model weighted by the number of measurements on the triple statement. Furthermore,

\[
P(p|s, o, \bar{t}) \approx \mathbb{E}(Y_{s,p,o,t})/\sum_{p'} \mathbb{E}(Y_{s,p',o,t})
\]

where we have exploited that, due to the LCWA, \( N_{s,p,o} = N_{s,p',o} \). Thus sampled binary labels reflect the normalized triple probabilities.

### 3.7 Modeling Dependencies: Generalized Statements

Can classes or attributes be the subject in a triple statement? For example, whereas the semantics of \((\text{Sparky}, \text{hasColor}, \text{Black})\) is clear, the semantics of \((\text{Dog}, \text{hasColor}, \text{Black})\) is less well defined: it could mean that there is at least one dog that is black, or that some or most dogs are black, or that all dogs are black. In our work, we consider frequencies.
We define a triple \((c_1, hA, c_2)\) where
\[
\mathbb{E}(Y_{c_1, hA, c_2}, t) = \mathbb{E}(Y_{\bar{s}, hA, c_2}, t | Y_{\bar{s}, hA, c_1}, t).
\] (3.6)

This is the probability that a new entity \(s'\) at a new time instance \(t'\) that has unary label \(c_1\) will also have unary label \(c_2\). Generalized statements are useful for predictions at a new instance \(t'\) where only partial measurements are available. With a predicted unary label \(Dog\) and using the generalized statement \((Dog, hA, Mammal)\), the unary label \(Mammal\) can be predicted.

It corresponds to a probabilistic version of the rule \(\forall s : (s, hA, Mammal) \leftarrow (s, hA, Dog)\), where \(s\) is a variable that stands for an entity. We can generalize to more complex dependencies, such as \(E(Y_{\bar{s}', hA, c_3, t'} | Y_{\bar{s}', hA, c_1, t'}, Y_{\bar{s}', hA, c_2, t'})\), which corresponds to a rule with a conjunctions of literals in its body. Equation 17 in appendix A describes how the conditional probability in equation 6 can be estimated from data. Generalized statements are further discussed in section 5.2 and are the basis for embedded symbolic reasoning described in section 8.3.

4 A Bilayer Tensor Network

We now describe the bilayer tensor network (BTN), which approximates the tensors described in the last section. For episodic memory, the input is \(t\), and in the sampling mode, it generates samples for binary statements from
\[
P(s|t)P(o|s, t)P(p|s, o, t),
\] (4.1)

where \(p \in \mathcal{P}_B\) is the binary label. For unary statements where we use \(p = hA\) and generate samples \(P(s|t)P(c|s, t)\), where \(c\) is the unary label. The conditional probabilities are derived from the triple retrieval model of equation 3.2.

For semantic memory (i.e., the prior retrieval model), the input is \(t \leftarrow \bar{t}\), i.e., an unspecified future time index, and a concept \(s\). Using the conditional probabilities, the sampling schedule can easily be modified; for example, with a fixed sample \(t, s, o\), we can sample several unary labels \(c\) and binary \(p\) labels in one run.

4.1 BTN Architecture. The BTN is implemented by an interaction of two layers, which, as we argue later, might be related to functional brain operations. One layer is the representation layer \(q\) with preactivation vector \(q \in \mathbb{R}^r\); here, \(r \in \mathbb{N}\) is the embedding dimension, that is, the rank of the BTN approximation. As we propose later, \(q\) reflects the cognitive state of the brain (see Figure 1). The other one is the index layer \(n\) with preactivation
Figure 1: Perception and memory. Our model architecture consists of two main layers, the representation layer $q$ and the index layer $n$. $q$ and $n$ are the vectors of preactivations. In perception, the representation layer obtains inputs from the scene. Whereas the computational operations in these layers are practically instantaneous, the dynamic context layer $h$ represents internal brain activities and stores information when attention moves from one entity to another. The dotted line represents embodiment, the activation of earlier processing layers via the representation layer.

vector $n \in (0, 1)^d$, with $d = N_C + N_P + N_T$. The index layer contains one dimension or unit for each concept, predicate, and episodic instance. The connection matrix $A$ links both layers, and its columns are the embeddings of the indices. We also introduce a dynamic context layer $h$ with preactivation vector $h$. In the following, we describe the implementations of layers and operations using the unfolded view in Figure 2.

4.2 BTN Algorithm for Memory Recall. The BTN relies on embedding vectors $\{a_s, a_p, a_t\}$, which form the columns of $A$. Here, $s \in C$, $p \in P_B$, and $t \in T$. In this section, the embedding vectors contain free parameters optimized for good performance. In the following sections, we ground them in perception, in the spirit of an embodied approach.

We get for the episodic memory model for the subject

$$P(s|t) = \text{softmax}_s^\beta(n_S)$$

(4.2)

with preactivations $n_S = a_s^T \text{sig}(g(q_T))$ and $q_T = a_t$. For unary labels, we obtain

$$P(c|s, t) = \text{softmax}_c^\beta(n_C).$$

(4.3)
Figure 2: Unfolded representation of episodic and semantic memory. In perception, the structures in yellow at the bottom are added; they stand for inputs from the visual scene.

For the object, we obtain

\[ P(o | s, t) = \text{softmax}_o^\beta(n_O), \]  

(4.4)

where \( n_O = a_o^\top \text{sig}(g(q_S)) \) and \( q_S = a_s + g(q_T) \). For binary labels, we get

\[ P(p | s, o, t) = \text{softmax}_p^\beta(n_P), \]  

(4.5)

with \( n_P = a_p^\top \text{sig}(g(q_O)) \) and \( q_O = a_o + g(q_S) \).

The semantic-state model for the episodic memory model for unary statements becomes (cf. equation 1)

\[ E(Y_{s,hA,c,t}) = \text{sig}(n_C) \]  

(4.6)

and for binary statements,

\[ E(Y_{s,p,o,t}) = \text{sig}(n_P). \]  

(4.7)

Essentially, we replace the softmax activation function with the \text{sig} activation function. This renormalization is discussed in appendix B.
For the semantic memory model (i.e., the prior retrieval model), we use 
\( t \leftarrow \tilde{t} \) and \( \vec{a}_{t \leftarrow \tilde{t}} \leftarrow \vec{a} \). Thus, \( \vec{a} \) stands for the embedding of some future instance \( \tilde{t} \). Also, we input \( s \).

Here, \( g(\cdot) \) is a nonlinear function. In our approach, we use

\[
g(q) = W \text{sig}(B \text{sig}(V \text{sig}(q))).
\]

which performs computations in the hidden layer \( h \) of the recurrent network. \( W, B, V \) are learned matrices. Also, \( \text{sig}(x) = 1/(1 + \exp(-x)) \) is the logistic function, and

\[
\text{softmax}_i^\beta(x) = \frac{\exp \beta x_i}{\sum_i \exp \beta x_i}.
\]

Then the postactivation vector \( \text{softmax}_i^\beta(x) \) (without the subscript) is the column vector of all \( \{\text{softmax}_i^\beta(x)\}_i \). Here, \( \beta \geq 0 \) is an inverse temperature and can be used for making the response more or less focused. \( x \) is the vector of preactivations.

4.3 **Discussion on the BTN.** An important property of the BTN is that in the algorithmic implementation and at any iteration step, the representation vectors have the dimension of the embedding vector and, for example, cannot represent the concatenation of two embeddings. We can relate this to brain function. Since the representation layer might occupy a significant portion of the brain, leaving no space for a second concurrent representation, we call this property the “one-brain hypothesis” (discussed in more detail in section 8.6). Also, due to the nonlinearity of the function \( g(\cdot) \), one breaks the symmetry between subject and object embedding. It is clear which entity is the subject and which one is the object, and the agent can distinguish between \( \text{(Sparky, ownedBy, Jack)} \) and \( \text{(Jack, ownedBy, Sparky)} \). Thus, proper role labeling is performed. Halford et al. (2014) call this “structural alignment.” We did not select more standard models for several reasons. First, most tensor factorizations (Hackbusch, 2012), such as CPD, the Tucker decomposition, the tensor train, and RESCAL (Nickel et al., 2011), require an excessive multiplication of factors; multiplication is an operation that is not easily implemented in biological hardware. Second, standard tensor networks are functions of several embedding vectors that would need to be presented concurrently; this would violate our one-brain hypothesis. By representing functions of functions, our model is compositional (in the sense of Poggio, Banburski, & Liao, 2020), which is a property that is used to explain the superior performance of deep architectures. Other forms of compositionality are discussed in section 8.
4.4 BTN Algorithm for Perception. We consider the following setting:
At a new episodic instance $t'$, the agent encounters a new scene $t'$. Then a
bounding box $BB_{sub}$ is segmented, whose content describes a visual entity
$s'$. The visual entity $s'$ might be a known entity, or it might be a novel entity,
not yet known to the agent. The agent might also detect a second visual object
$o'$ with bounding box $BB_{obj}$ in the scene and might be interested in its
relationship to $s'$. The content of a third bounding box $BB_{pred}$, typically en-
ccompassing the subject and object bounding boxes, describes the predicate.
The goal is now to produce statements that are likely true, considering the
context of the scene.

We get for the episodic instance

$$P(t' = t | scene_{t'}) = \text{softmax}_t^\beta(n_T), \quad (4.8)$$

where $n_T = a_t^\top \text{sig}(f(scene_{t'}))$.

We obtain for the subject (see equation 4.2)

$$P(s' = s | t', BB_{sub}, scene_{t'}) = \text{softmax}_s^\beta(n_S)$$

with $n_S = a_s^\top \text{sig}(f(BB_{sub}) + g(q_T))$ and $q_T = a + f(scene_{t'})$.

For unary labels, we obtain

$$P(c' = c | s' = s, t' = t, BB_{sub}, scene_{t'}) = \text{softmax}_c^\beta(n_C)$$

with $n_C = a_c^\top \text{sig}(q_S)$.

We obtain for the object (see equation 4.4)

$$P(o' = o | s' = s, t' = t, BB_{sub}, BB_{obj}, scene_{t'}) = \text{softmax}_o^\beta(n_O)$$

with $n_O = a_o^\top \text{sig}(f(BB_{obj}) + g(q_O))$ and $q_O = a_o + f(BB_{sub}) + g(q_T)$.

For binary labels, we get

$$P(p' = p | o' = o, s' = s, t' = t, BB_{sub}, BB_{obj}, BB_{pred}, scene_{t'}) = \text{softmax}_p^\beta(n_P).$$

with $n_P = a_p^\top \text{sig}(f(BB_{pred}) + g(q_O))$ and $q_O = a_o + f(BB_{obj}) + g(q_S)$. The
semantic-state model equation for unary statements is

$$E(Y_{s',hA,c,t'} | s' = s, t' = t, BB_{sub}, scene_{t'}) = \text{sig}(n_C) \quad (4.9)$$

and for binary statements is

$$E(Y_{s',p,o',t'} | o' = o, s' = s, t' = t, BB_{sub}, BB_{obj}, BB_{pred}, scene_{t'}) = \text{sig}(n_P). \quad (4.10)$$
Algorithm 1: The BTN for Perception and Episodic Memory.

1 switch Perception do
   | Input: scene, BB\textsubscript{sub}, BB\textsubscript{obj}, BB\textsubscript{pred}; u = 1
2 end
3 switch Episodic Memory do
   | Input: t\textsuperscript{*}, u = 0
4 end
5 h = 0 \quad \triangleright \text{Alternatively, h is inherited from past decoding}
6 \tilde{q}_T \leftarrow u(f(\text{scene}) \quad \triangleright \text{Representation of overall scene}
7 switch Perception do
8 \forall t \in T : n_{T(t)} \leftarrow a_{t}^\top \text{sig}(\tilde{q}_T)
9 Sample \ t^* \sim \text{softmax}^\beta(\mu_T) \quad \triangleright \text{Sample a past episodic}
10 end
11 switch Episodic Memory do
12 \quad t^* = t^*_m \quad \triangleright \text{Input for episodic memory}
13 end
14 q_T \leftarrow \tilde{q}_T + a_{t}^*. \quad \triangleright \text{Sample a subject entity}
15 \forall c \in C : n_{C(c)} \leftarrow a_{c}^\top \text{sig}(q_T)
16 h \leftarrow B\text{sig}[(\text{sig}(h) + V\text{sig}(q_T)]
17 \tilde{q}_S \leftarrow u(f(BB\textsubscript{sub}) + W\text{sig}(h) \quad \triangleright \text{g(\cdot) = W\text{sig}(h)}
18 \forall s \in C : n_{S(s)} \leftarrow a_{s}^\top \text{sig}(q_S)
19 Sample s^* \sim \text{softmax}^\beta(\mu_{S}) \quad \triangleright \text{Sample a subject entity}
20 q_S \leftarrow \tilde{q}_S + a_{s}^*. \quad \triangleright \text{Sample a object entity}
21 \forall c \in C : n_{C(c)} \leftarrow a_{c}^\top \text{sig}(q_S)
22 Sample c^* \sim \text{softmax}^\beta(\mu_{C}) \quad \triangleright \text{Sample a an object entity}
23 h \leftarrow B\text{sig}[(\text{sig}(h) + V\text{sig}(q_S)]
24 \tilde{q}_O \leftarrow u(f(BB\textsubscript{obj}) + W\text{sig}(h)
25 \forall o \in C : n_{O(o)} \leftarrow a_{o}^\top \text{sig}(q_O)
26 Sample o^* \sim \text{softmax}^\beta(\mu_{O}) \quad \triangleright \text{Sample a object entity}
27 q_O \leftarrow \tilde{q}_O + a_{o}^*. \quad \triangleright \text{Sample a an object entity}
28 h \leftarrow B\text{sig}[(\text{sig}(h) + V\text{sig}(q_O)]
29 \tilde{q}_P \leftarrow u(f(BB\textsubscript{pred}) + W\text{sig}(h))
30 q_P \leftarrow \tilde{q}_P \quad \triangleright \text{Sample a binary label}
31 \forall p \in P^B : n_{P(p)} \leftarrow a_{p}^\top \text{sig}(q_P)
32 Sample p^* \sim \text{softmax}^\beta(\mu_{P})
33 return t^*, s^*, c^*, o^*, p^*

Here, f(\cdot) is a representation vector derived from visual inputs realized by a deep convolutional neural network (DCNN, see section 6).

5 Algorithmic Implementation and the Attention Approximations ____

5.1 Algorithmic Implementation by Stochastic Sampling. Figure 2 illustrates the unfolded processing steps of the architecture shown in Figure 1. Algorithm 1 describes the processing steps implementing the
Table 1: Generation of Activated Indices in Different Operational Modes.

| Sequential Index Pattern | Equivalent Triple Statements | Mode     |
|--------------------------|------------------------------|----------|
| Sparky, Dog, Friendly, Black | (Sparky, hA, (Dog, Friendly, Black)) | SM:U     |
| Sparky, Jack, looksAt, ownedBy | (Sparky, [looksAt, ownedBy], Jack) | SM:B     |
| t, Sparky, Dog, Friendly, Black | t: (Sparky, hA, (Dog, Friendly, Black)) | EM: U    |
| t, Sparky, Jack, looksAt, ownedBy | t: (Sparky, [looksAt, ownedBy], Jack) | EM: B    |
| Dog, Friendly, Black | t′, s′: (s′, hA, (Dog, Friendly, Black)) | P-Dir/SA: U |
| looksAt, ownedBy | t′, s′, o′: (s′, [looksAt, ownedBy], o′) | P-Dir/SA: B |
| Sparky, Dog, Friendly, Black | At t′: (Sparky, hA, (Dog, Friendly, Black)) | P-Samp: U |
| Sparky, Jack, looksAt, ownedBy | At t′: (Sparky, [looksAt, ownedBy], Jack) | P-Samp: B |

Notes: The first column shows the sequence of activated indices (“firing neurons”). The second column shows equivalent triples. The last column shows the operational model: SM (semantic memory), EM (episodic memory), P-Dir (direct perception), P-SA (semantic attention), and P-Samp (perception with sampling). We indicate that for given episodic instances, several unary labels (U) and binary labels (B) can be generated. If we extend the knowledge graph to also contain nodes for predicates and time instances, then, in sampling, exactly one node is active at a time.

Equations in the previous sections by generating samples from the conditional probabilities. The main difference between episodic memory and perception is that in the latter, visual inputs from the contents of the bounding boxes are integrated. The algorithm outputs \( t^*, s^*, c^*, o^*, p^* \) from which we can form triples of the form \( \langle s', hA, c^* \rangle \) and \( \langle s', p^*, o' \rangle \), but also \( \langle s^*, hA, c^* \rangle \) and \( \langle s^*, p^*, o^* \rangle \). Here, for example, \( s' \) is an entity in the scene, and \( s^* \) is an entity with a permanent representation in the index layer. In perception, the episodic index \( t^* \) would be the index of a similar episode in the past.

In Figure 2, we indicate that we can also sample a unary label for the complete scene or episode and for the predicate, which we do not actually do in the experiment. We also indicate additional processing steps for calculating embeddings. For example, \( \tilde{q}_S \leftarrow q_S + a_{c^*} \) is the embedding of the sentence \( \langle s^*, hA, c^* \rangle \) in the context of the scene. Whereas concepts, predicates, and episodic instances have static embeddings realized as connection weights linking the index layer and the representation layer, the embedding of a sentence is dynamic.

In algorithm 1, we explicitly introduce the latent vector \( h \) modeling dynamic context, supporting the implementation of the nonlinear function \( g(\cdot) \). Table 1 illustrates the generation of activated indices (“firing indices or neurons”) in different operational modes.

The algorithm for semantic memory is identical to episodic memory, only that the input is \( t^*_{in} \leftarrow \tilde{f} \) and \( a_t \leftarrow \tilde{a} \). One might also specify \( s^* \leftarrow s \) and not sample it.

Sampling is a unique, maybe temporary, commitment to a concept. In a biological interpretation, a single winning unit (neuron) fires. Even a single or a few spikes can be sufficient for communicating the winning concept.
to later processing. Since sampling commits unique indices $t^*$ and $s^*$, this allows the association of semantic and episodic memory experiences to the observation.

### 5.2 Embedded Symbolic Reasoning by Chaining

As indicated by the four groups of dots in Figure 2, unary decoding can continue during the time of operation of the dynamic context layer. For example, if $\tilde{q}_S$ represents the triples $(Sparky, hA, Dog)$ and the agent has learned the generalized statement $(Dog, hA, Mammal)$, then it is likely that the index for $Mammal$ fires next. Thus, the sequential index pattern $Sparky, Dog, Mammal$ is interpreted by the agent that Sparky is a dog and a mammal. This form of embedded symbolic reasoning, as further discussed in section 8.3, is based on generalized statements discussed in section 3.7.

### 5.3 Sampling and the Attention Approximation

For episodic and semantic memory, we need to generate triples of the form $(s^*, hA, c^*)$, whereas to label entities in perception, we only need $(s', hA, c^*)$; that is, $s'$, the subject in the scene, does not need to be identified as a stored entity $s^*$. For example, if the agent is attacked by a bear, it does not care what the name of the bear might be; it might be a bear not yet known. Our approach is motivated by the computational attention approach used in deep learning (Vaswani et al., 2017). With parallel hardware (e.g., brainware) the sampling process can be replaced by a computational attention approximation, which can be executed in parallel and does not commit to a particular $s^*$.

The semantic attention (SA) approximation for the subject replaces line 20 in algorithm 1 with

$$q_S \leftarrow \tilde{q}_S + A \operatorname{softmax}_\beta (A^\top \operatorname{sig}(\tilde{q}_S)).$$

In matrix $A$, we only consider the columns relating to entities. In terms of standard attention, as defined by Weston et al. (2014), Sukhbaatar et al. (2015), and Vaswani et al. (2017), the argument of the softmax calculates the score vector, the softmax output is the alignment vector, the multiplication with matrix $A$ calculates the context vector, the columns of $A$ are the key vectors and the value vectors, and $\operatorname{sig}(\tilde{q}_S)$ is the query. The main differences to the standard approaches to attention is that in our approach, key vectors and value vectors are stored semantic embeddings, and the first term on the right is the preactivation instead of the postactivation; processing several bounding boxes from the same image in parallel, as it is done in visual attention (e.g., Koner, Shit, & Tresp, 2020), would violate our one-brain hypothesis from section 8.6. In the sampling mode of operation, a symbolic

---

3 In our approach, sampling is balanced: we subsequently sample a member of each ontology—a Color, an Age class, and so on.
unit in the index layer competes with other units in the same layer to be activated. In contrast, in our attention approximation, the index layer acts as a standard neural network layer (without sampling) with softmax activation (i.e., the sum in the equation) and preactivation skip connections ($\tilde{q}_s$ in the equation).

In training the model and also in the attention approximation, we use $\beta = 1$. With $\beta = 1$, predictive samples reflect the predictive uncertainty in prediction. In the actual sampling experiments, we set the inverse-temperature parameter $\beta \to \infty$, effectively taking the concept with the highest probability. We call this winner-take-all sampling. If there were a dominating dimension or if we would set $\beta \to \infty$ in the attention approximation, the attention approximation would become identical to winner-take-all sampling.

Similarly for the object, we replace line 27 with

$$q_O \leftarrow \tilde{q}_O + A \text{softmax}^\beta(A^\top \text{sig}(\tilde{q}_O)).$$

In matrix $A$ we only consider the columns relating to entities. Attention can also be applied to episodes in perception. The episodic attention (EA) for the episodic index replaces line 14 with

$$q_T \leftarrow \tilde{q}_T + A \text{softmax}^\beta(A^\top \text{sig}(\tilde{q}_T)).$$

In matrix $A$ we only consider the columns relating to episodic instances. EA is the default in all experiments on perception. Sampling $t^*$ is important to initialize an associated episodic memory experience.

6 Perception, the Representation Layer, and the Dynamic Context Layer

In this and the following sections, we present experimental results. Here, we focus on perception, and in the following sections on engrams, semantic memory, reasoning, language, episodic memory, and self-supervised learning. Intertwined with the experiments, we make the connection to cognition and neuroscience.

There is a long tradition in cognition and neuroscience to distinguish between anatomical brain structure, for example, the actual anatomical structure of the biological neural network, and functional architectures of the cortex, where the latter might be dynamic and quickly adaptable (Friston, Tononi, Sporns, & Edelman, 1995; van den Heuvel & Sporns, 2013; Bassett & Sporns, 2017; Sporns, 2018; Leopold et al., 2019). We discuss the functional architectures primarily but occasionally also consider structure.

6.1 Data Set. We tested our approach experimentally using an augmented version of the VRD data set (Lu et al., 2016). In the past, this data
Figure 3: Ontologies. (Top) Class ontology. (Bottom) Attribute ontology.

set has been the basis for much research on visual relationship detection. Each visual entity is labeled as belonging to one out of 100 classes. There are 70 binary labels, with 37,993 binary statements total. We followed other work and assigned 4000 images to the training set and 1000 images to the test set. The training images contain, overall, 26,430 bounding boxes, thus on average, 6.60 per image.

We generated a first derived data set, VRD-E (for VRD-Entity), with additional labels for each visual entity. First, each entity in each image obtains an individual entity index (or name). The 26,430 bounding boxes in the training images describe as many entity indices. Second, we used concept hierarchies from WordNet (Fellbaum, 2010; see Figure 3). Each entity is assigned exactly one basis class (or B-Class) from VRD (e.g., Dog), one parent class (or P-Class, e.g., Mammal), and one grandparent class (or G-Class, e.g., LivingBeing). At any level, we use the default class Other for entities that cannot be assigned to a WordNet concept. We perform subclass reasoning in the training data and label an entity with its entity index, its B-Class, P-Class, and G-Class.

In addition, we used pretrained attribute classifiers (Anderson et al., 2018; Wu et al., 2019) to label visual entities using the attribute ontology shown in Figure 3. Each visual entity obtains exactly one color (including the color Other) and exactly one activity attribute, for example, a person can be standing or running. We also introduce the unary labels Young and Old, which are randomly assigned, such that these can only be predicted for test entities that already occurred in training, not for novel entities.

Furthermore, we introduce the nonvisual, or hidden, unary label Dangerous to all living things and Harmless to all nonliving things. We do not use these labels in perceptual training; we use them instead to demonstrate how semantic memory can supplement nonvisual labels.
In summary, every visual entity receives one entity index and, in addition, seven positive unary labels—for example, Entity = Sparky, B-Class = Dog, P-Class = Mammal, G-Class = LivingBeing, Age = Young, Color = Black, Activity = Standing, Risk = Harmless.

Based on the VRD-E data set, we generate the VRD-EX data set. Here, we distort each image in the training data set, which generates another 4000 training images.\(^4\) We used an open library to distort the image (Jung et al., 2020). To obtain a distorted image, we apply a sequence of transformations, including translation, rotation, shearing, and horizontal flipping of the original image (Bloice, Stocker, & Holzinger, 2017). Each transformation is associated with a probability of actually using it. When the operation has coefficients, such as the displacement of translation, a random value within a reasonable range is generated. Thus, VRD-EX has 8000 training images. Then we perform another distortion on each original image and create a set of another 4000 test images.\(^4\) Note that in these new 4000 test images, every visual entity has already occurred in the training set twice (see Figure 4).

In addition to the visual concepts, we introduce nonvisual entities, which do not occur in any image. In the experiments, we relate visual entities to those hidden entities, for example, by the predicate ownedBy or the predicate lovedBy. For example, each visual dog is owned by a person who is not in any scene. Table 2 shows the overall statistics.

Training was performed using the Adam optimizer (Kingma & Ba, 2014). For evaluation, we consider top-1 accuracy for unary labels and Hits@k for binary labels and sampling of entities. Hits@k is the fraction of correct entities that appear in the top-k-ranked entities.

6.2 DCNN. As discussed, we use a deep convolutional neural network (DCNN) for the mapping \(f(\cdot)\) from scene and bounding box content to the

\(^4\) Due to distortion, some objects and images are discarded, resulting in a reduced number of samples.
Table 2: Statistics of the data sets VRD-E and VRD-EX.

| Data Set        | Training Images | Test Images | #BB Train | #VisEnt Train | #BinStat Train | # Attr/Ent Train |
|-----------------|-----------------|-------------|-----------|---------------|----------------|-----------------|
| VRD (Lu et al., 2016) | 4000            | 1000        | 26,430    | 26,430        | 30,355         | 1               |
| VRD-Entity      | 4000            | 1000        | 26,430    | 26,430        | 30,355         | 8               |
| VRD-EXtended    | 7737            | 3753        | 50,910    | 26,430        | 59,095         | 8               |

Notes: Attr/Ent stands for the average number of unary labels per episodic instance. Overall, 15% of the labels are Others.

representation layer. Kriegeskorte and Douglas (2018) and others have discussed how DCNNs might represent functional modules in the brain. This is not the topic of this article. Our approach is object-centric. The specific DCNN architecture we use is the VGG-19 architecture (Simonyan & Zisserman, 2014), pretrained on ImageNet (Russakovsky et al., 2015) and fine-tuned to our data. The VGG-19 architecture constitutes the backbone layer of the Faster R-CNN (Ren, He, Girshick, & Sun, 2015). We employ Faster R-CNN to produce a set of bounding boxes as output, where each bounding box contains an object. (For more details, see appendix D.) In all of our experiments, we used rank $r = 4096$ as the dimension of the representation layer.

6.3 Direct Perception. In our approach, four bounding boxes are analyzed: first, a bounding box describing the complete scene; second, a bounding box describing the subject entity; third, a bounding box describing the object entity; and finally, a bounding box describing the predicate. In direct perception, we assume that labels are independently generated by perception, purely based on bounding box content and entirely in bottom-up mode. There are connections from the representation layer to the index layer but not in the opposite direction. Also, there are no connections between the representation layer and the dynamic context layer.

From a perceptual view, this approach exploits neither dependencies between labels for the same bounding box (that a detected Sparky is known to be a black dog), nor dependencies between labels for the different bounding boxes. Direct perception requires concept indices, together with their associated embedding vectors, which become the connection weights from the representation layer to the index layer. Without connections from the index layer back to the representation layer, no episodic memory or semantic memory can be formed. In direct perception, it is unclear how the overall brain is informed about winning or likely concept labels. This is one of the issues addressed in BTN perception, discussed next.

Table 4 shows results for unary labels on direct perception (rows labeled “P-Direct”). The results on VRD-E are pretty competitive. The performance is improved in VRD-EX, likely due to some form of memorization by
Table 3: Binary Label Prediction in Perception.

| Model  | Binary Labels | @10  | @1  |
|--------|---------------|------|-----|
| VRD-E  | P-Direct      | 85.45| 31.68|
|        | P-Samp        | 90.39| 45.09|
|        | P-SA          | 91.33| 46.84|
| VRD-EX | P-SA          | 99.58| 80.63|

Notes: As the inferior results for P-direct indicate, the dynamic context layer is essential to perform well in binary label prediction (rows 1–3). For known entities (VRD-EX) (fourth row), entity indices permit some memorization and improve performance. Best results in each column are in bold.

overfitting since different views on the same entities occur in the training set and the test set. The results on the prediction of binary labels (see Table 3) are not very good; as expected, and a well-known result from scene graph analysis, information on the subject and the object bounding boxes is required for binary label prediction.

6.4 BTN Perception. In perception with the BTN, we add connections from the index layer back to the representation layer. We also connect the representation layer with the dynamic context layer. This has several important benefits.

First, in a top-down mode of operation, the representation layer, and thus the whole brain, is informed about which concepts are detected in the scene. Thus, if an entity is labeled as being Dangerous or recognized as being Jack, this information is communicated to the whole brain. This pattern completion process is a property typically discussed in the context of associative memories. If this affects earlier processing layers, this process is referred to as embodiment. Some theories on embodiment assume, as the name indicates, that thought and language influence bodily states, or at least the brain regions mapping directly to bodily states (Lakoff, Johnson, & Sowa, 1999).

Second, unary labels would get biased: if Sparky is detected in the scene, it will bias the unary labeling toward the unary label Black if this is known from semantic memory to be true.

Third, by connecting the representation layer with the dynamic context layer, we obtain a memory that can transport information from the analysis of one bounding box to another. This change of attention from one bounding box to another might be associated with saccadic eye movement, where the dynamic context layer stores information during the transition phase. The dynamic context layer is crucial for evaluating triples involving binary labels.
Finally, the bidirectional connectivity between index layer and representation layer enables the formation of episodic and semantic memory and can enrich perception with information from either, as will be discussed in the following sections.

One can think of perception as a triple-generating language model in the context of visual inputs. Through the top-down information flow via the representation layer, the whole brain is kept up to date about scene content, about which concepts have been detected in the scene and which statements describe scene content.

The representation layer reflects the context. For example, if Sparky is detected in the subject bounding box, then \( q_s \) is an embedding of Sparky in the context of the scene. If Sparky in the scene is labeled as Friendly, then \( \tilde{q}_s \) would be the embedding of the statements (Sparky, ha, Friendly) in the context of the scene. If Sparky looks at Jack in the scene and this is detected, then \( q_P \) would be the embedding of the statement (Sparky, looksAt, Jack) in the scene’s context.

### 6.5 Experiments with BTN Perception

In all experiments, we use episodic attention (EA): a sampling of a past episodic index \( t^* \) is used only to evoke a related episodic memory, as discussed in section 9.

Experiments with semantic attention are labeled P-SA where there is no commitment to a specific \( s^* \) or a specific \( o^* \). In semantic attention, one is primarily interested in fast labeling rather than in associating memories.

The results in Table 4 show that on the VRD-E data set with novel entities in the test set, perception with semantic attention (P-SA) shows the best results. However, direct perception (P-Direct) is quite competitive.

On the VRD-EX data set with known entities in testing, perception with sampling, P-Samp, is best, where the algorithm could “remember” past encounters of the same entities. Here, P-Direct is not competitive. For P-Samp, we use winner-take-all sampling.

Table 3 shows results from binary label prediction. For the binary labeling with novel entities on the VRD-E data, P-SA shows the best performance. The inferior performance of P-Direct confirms that the dynamic context layer is important for achieving good performance. We see a large improvement for the VRD-EX data set.

In Table 5, we compare our model with other visual relationship detection methods. In the methods from the literature, only binary labels are tested, with one class label for each entity. In our framework, this would be the decoding of generalized statements. The table demonstrates that generalized statements (i.e., probabilistic rules) can be generalized themselves. In a zero-shot situation, where the triple \( (\text{subject-class}, \text{predicate}, \text{object-class}) \) never occurred in training, our approach achieved a recall score of 81.61%, which is much better than the result from the BFM (Baier et al., 2017) with 76.05%. In summary, our approach gives competitive results overall and superior results for zero-shot binary labeling.
Table 4: Unary Label Prediction in Perception.

| Model | Entity | B-Class | P-Class | G-Class | Y/O | Color | Activity | Average |
|-------|--------|---------|---------|---------|-----|-------|----------|---------|
| VRD-E | P-Direct | – | 81.30 | 88.44 | 94.67 | | | **50.60** | **69.06** | **83.73** | **77.97** |
| | P-Samp | – | 80.91 | 88.85 | 95.02 | | | 47.88 | 68.02 | 82.90 | 77.26 |
| | P-SA | – | **81.80** | **89.34** | **95.44** | | | 49.46 | 68.93 | 83.58 | **78.09** |
| VRD-EX | P-Direct | 92.20 | 90.87 | 93.54 | 96.47 | 76.54 | 84.49 | 92.77 | 89.56 |
| | P-Samp | **92.81** | **96.47** | **97.49** | **98.47** | **94.58** | **94.08** | **97.59** | **95.93** |
| | P-SA | 92.65 | 95.27 | 97.11 | 98.20 | 93.10 | 92.71 | 96.83 | 95.12 |

Notes: We evaluate \((s', hA, c)\) for the entities \(s'\) in the scene. The columns labeled “B-Class, P-Class, G-Class” are class labels, and the columns labeled “Y/O, Color, Activity” are unary labels. “Average” refers to the average over all columns. In P-Direct, there are no links from \(n\) to \(q\), and \(q\) and \(h\) are independent. P-Samp stands for the sampling approach using the index \(s^*\) with maximum activation. P-SA uses semantic attention. On the VRD-E data set (rows 1–3), where each entity is novel, P-SA shows the best results. Not surprisingly, P-Direct is also quite competitive. On the VRD-EX data set (rows 4–6), where each entity is known, P-Samp shows the best results since it can recognize specific entities. In particular, the unary label Y/O can only be learned for already known entities. Not surprisingly, P-Direct is significantly worse since it cannot benefit from memory, although overfitting leads to better results than with the VRD-E data. The column labeled “Entity” evaluates if the right entity is recognized. Thus, we evaluate \((s', \text{sameAs, s})\) for the entities \(s'\) in the scene. Best results in each block are in bold.

Figure 5 illustrates perception with unary labels and visual and nonvisual binary statements.

### 6.6 Representation Layer

In our model, \(q\), that is, the activation of the representation layer, represents the cognitive brain state. In cognitive neuroscience, the representation layer is referred to as “mental canvas” or the “global workspace,” enabling a global information exchange. It is also known as “theater of the brain,” “communication platform,” “communication bus,” or “blackboard.” It is assumed to have a distributed representation involving large parts of the brain. Binder and Desai (2011) state: “The neural systems specialized for storage and retrieval of semantic knowledge are widespread and occupy a large proportion of the cortex in the human brain.” During perception or memory recall, the representation layer integrates information and then makes this information available to the brain as a whole. In particular, the embedding vector of the episodic experience, that is, \(a_t\), represents a holistic, integrative view of the cognitive state at that instance.

Since the world of an agent mostly changes smoothly, one can develop models to forecast embedding vectors of future instances (Tresp et al., 2015; Han et al., 2020b). The brain’s cognitive state is to a large degree determined by perception, which might explain a personal feeling of personal instability and sensitivity to external influences, whereas other individuals,
Table 5: Binary Label Prediction in Perception.

| Model                        | ph   | z-s-ph | rl      | z-s-rl |
|------------------------------|------|--------|---------|--------|
| P-SA                         | 24.50| 8.55   | 93.99   | 81.61  |
| P-Direct                     | 13.54| 5.73   | 84.64   | 68.35  |
| BFM (Baier et al., 2017)     | 25.11| 7.96   | 93.81   | 76.05  |
| Approach in Tresp and Ma (2016) | 23.45| 10.95  | 93.32   | 78.79  |

Notes: We compare binary labeling with methods published in the literature using the original VRD data set. Phrase (ph) shows the recall of binary labels, where the extracted bounding box contents are evaluated. The binary (relationship) label (rl) shows the recall of the binary label given the ground-truth class of subject and object. z-s-ph and z-s-rl denote zero-shot performance for triples that did not occur in the training set. Our proposed model P-SA (first row) is superior in the last task. The much better performance of P-SA compared to P-Direct (second row) demonstrates the importance of the dynamic context layer. The third and fourth rows show results from approaches published in the literature. Best results in each block are in bold.

Figure 5: Illustration of perception with known entities (VRD-EX). The left bounding box is identified (sampled) as \( s' = s^* = 24470 = \text{Sparky} \). Highly ranked unary labels are Dog, Mammal, LivingBeing, Young, White, OtherActivity. Highly ranked unary labels for the second bounding box (24471) are Bench, Furniture, NonLivingBeing, Old, OtherColor, OtherActivity. Sampled binary statements are: (Dog, sitsOn, Bench), (LivingBeing, on, Furniture), (Mammal, sitsOn, Old), (White, sitsOn, Bench). We also indicate how semantic memory can support perception by adding binary statements to object entities, not in the scene: (Sparky, ownedBy, Jack(26675)), (Sparky, lovedBy, Mary(26676)), where Jack and Mary are not on the scene but in the agent’s semantic memory. In section 8.1 we call this “unobserved-modality generalization.” The semantic memory experience is further discussed in section 7.
represented by their embedding vectors, appear stable and slowly changing. In the view of the agent, the stable embedding of another entity $s$ is $a_s$. If data is acquired for entity $s$ at instance $t$, the entities embedding $q_S = a_s + g(a_t)$ become time dependent. In the view of the agent, the stable embedding of another entity $s$ is $a_s$, overwritten by $q_S = a_s + g(a_t)$ if data has been acquired for entity $s$ at instance $t$.

6.7 Dynamic Context Layer. To really capture the content of a scene, it is essential to understand the relationships between the scene entities; this requires an additional storage facility since, following our one-brain hypothesis, the brain possesses only exactly one global representation layer (see the discussion in section 8.6). As illustrated in Figures 1 and 2, we propose that this functionality is represented by the dynamic context layer $h$, which represents the state of some of the brain modules not explicitly discussed in this article. It receives input from the representation layer, processes that information as a recurrent neural network, and then returns the processed information to the representation layer, which starts processing visual input from a subsequent bounding box. Thus, the dynamic context layer stores and processes information between saccadic eye movements. The dynamic context layer might also involve higher brain regions, like working memory.

Figure 6 shows the Pearson correlation between activities of units in the dynamic context layer. It shows a certain clustering structure, as exhibited in
recent studies on functional and structural brain connectivity (e.g., Figure 2 in Pope, Fukushima, Betzel, & Sporns, 2021). In the brain, such a block structure is a signature of a small-world architecture. Since it is not clear what the dynamic context layer in our model actually represents in the brain, we refrain from a deeper discussion of this apparent similarity. The dynamic context layer also represents internal mental states that might not be driven directly by perception and memory.

In neuroscience, the default network is assumed to be active when a person is not focused on the outside world; instead, the brain is at wakeful rest, such as during daydreaming or mind-wandering (Gazzaniga et al., 2013). Perception, memory recall, the initiation of activity, and other causes can induce the brain to leave the default state. In the default state of our model, the brain’s representation layer interacts with the dynamic context layer, without being driven by perception.

7 Semantic Memory Engrams, Semantic Decoding, and Multimodality

7.1 Background on Semantic Memory. According to Tulving (1985), the semantic memory experience is independent of a particular episodic experience. It developed out of perception as an emerging property where semantic enrichment became independent of perceptual input. Thus, in the transition from episodic memory to semantic memory, provenance is lost. It is the longest-lasting and most durable memory since it models the stable statistics in the world. It aggregates information and is a dictionary view of the agent’s life experience. Our approach explains the remarkable similarity between episodic and semantic memory: semantic memory is the expected episodic memory of a future instance. Whereas an episodic memory experience requires the activation of the index for an episodic instance and its embedding vector in the representation layer, a semantic memory experience requires the activation of the semantic memory representation $\bar{a}$. The latter can be interpreted as the embedding of some future episodic index $\bar{t}$.

7.2 Neuroscience Perspective: Semantic Memory Engrams. Engrams are memory traces in the brain (Ralph, Jefferies, Patterson, & Rogers, 2017). In our approach, the combination of an index with its embedding vector would be an engram for that concept, although we will typically only refer to the latter part as the engram. An index is a symbol for a concept, whereas embeddings are part of an implicit concept memory and provide symbol grounding (Harnad, 1990; Barsalou, 2008).

Here, we are in agreement with several theories on semantic memory engrams from neuroscience. For example, Binder and Desai (2011) state that semantic memory engrams consist of both modal and amodal representations, supported by the “gradual convergence of information throughout large regions of temporal and inferior parietal association cortex.” The
amodal representations are described as a high-level convergence zone. In our approach, the embeddings would be the modal distributed representations and the indices the amodal, local, and symbolic representations.

The relationship between the index layer and the representation layer reminds one of the hub-and-spoke model (Ralph et al., 2017). The hub is supposed to be located in the anterior temporal lobes (ATLs), which might be where concept indices are consolidated. The hub is connected to several different areas (e.g., visual cortex, auditory cortex, orbitofrontal cortex), which might be part of the biological realization of the representation layer. Other hubs might be located in the parietal and the temporal lobe (Binder & Desai, 2011) and maybe in the frontal lobe (Tomasello, Garagnani, Wennekers, & Pulvermüller, 2017).

Some works propose that the anatomical distinction between the representation layer and the index layer might be blurred in the brain. One should instead assume an “interactive continuum of hierarchically ordered neural ensembles, supporting progressively more combinatorial and idealized representations” (Binder & Desai, 2011).

### 7.3 Neuroscience Perspective: Indices

Indices are explicit concept representations, that is, an index is a symbol for a concept and could be implemented functionally by a single unit. In the sampling mode of operation, a symbolic unit in the index layer competes with other units in the index layer. This is in contrast to the representation layer, where the activations of the ensemble of units contribute to the processing.

We are purposely imprecise about how exactly an index is represented anatomically (i.e., structurally), in brainware. In the one extreme, an index might be a single neuron, realizing localist codes, where neurons respond highly selectively to single entities (“grandmother cells”). The other extreme are densely distributed codes where items are encoded through the activity of many neurons (Kumaran et al., 2016). Most researchers favor a sparse population of cells realizing a population code. A distributed representation might facilitate the consolidation of new information (see section 10).

The debate about localized representations in the brain is ongoing. Specific concept cells have been found in the medial lobe (MTL) region of the brain. MTL includes the hippocampus, along with the surrounding hippocampal region consisting of the perirhinal (“what” path), parahippocampal (“where” path), and entorhinal neocortical regions. Researchers have reported on a remarkable subset of MTL neurons that are selectively activated by strikingly different pictures of given individuals, landmarks, or objects and, in some cases, even by letter strings with their names (Quiroga, 2012; Quiroga, Reddy, Kreiman, Koch, & Fried, 2005). Naturally, locality of representation is probably discovered only in well-designed experiments. In our model, an activated concept index activates many units in the representation layer, and a unit in the representation layer in turn activates
many indices. Since index activations might change rapidly, the general appearance might be that of a globally activated system, hiding the locality of representation. An index is a focal point of activity, but it is never active in isolation.

7.4 Neuroscience Perspective on Concept Grounding. Our discussion can be related to recent results from neuroscience where, in different brain regions, maps have been discovered that code (e.g., for visual appearance, sound, and function). For example, if the concept Hammer is activated in the index layer, it might excite brain areas indicating a typical hammer appearance, the sound of hammering, and the required motor movement of hammering, all represented in the biological equivalence of the representation layer (Rueschemeyer, van Rooij, Lindemann, Willems, & Bekkering, 2010). Another example is the concept Cat, which includes the information that a cat has four legs, is furry, meows, can move, or can be petted (Kiefer & Pulvermüller, 2012). In terms of Binder and Desai (2011), the embeddings are modality specific (here: visual), and the indices represent convergence zones. As Dehaene (2014) puts it: “Every cortical site holds its own specialized piece of knowledge.”

Evidence for distributed semantic activation has also been described by Huth, de Heer, Griffiths, Theunissen, and Gallant (2016) and de Heer, Huth, Griffiths, Gallant, and Theunissen (2017). Both papers developed a detailed atlas of semantic categories and their topographic organization through extensive fMRI studies, showing the involvement of the lateral temporal cortex, the ventral temporal cortex, the lateral parietal cortex, the medial parietal cortex, the medial prefrontal cortex, the superior prefrontal cortex, and the inferior prefrontal cortex (Huth et al., 2016).

Recently, it has been proposed that embeddings are context dependent (Popp, Trumpp, & Kiefer, 2019). Our model suggests that concept embedding is rather stable but that the induced activation of a concept in the representation layer is superimposed with context information, as discussed throughout the article. The representation layer is activated by sensory input and activated concept indices, so the model is informed about a concept in context, even with the connection weights between index and representation layer being fixed.

7.5 Semantic Memory Engram in the BTN. In perception, the embedding vectors determine which indices are activated by the representation layer in a bottom-up operation. However, the operation is bidirectional: an index can activate the representation layer by its embedding, which might even activate earlier perceptual layers. We call this top-down mode the grounding or the embodiment of that concept. Since an embedding vector is optimized for a concept’s role in perception and memory, it implicitly reflects all background that is known about it.
Consider equation 4.9, which labels entity $s$ with unary label $c$. With no visual inputs, $\mathbf{a}_c^T \text{sig}(\mathbf{a}_s)$ is the contribution from semantic memory; it acts as a prior for the unary statements. The visual input contributes the term $\mathbf{a}_c^T \text{sig}(\mathbf{f}(\mathbf{BB}_{sub}))$, which is the inner product of the embedding vector with high-level features generated from visual inputs. Thus, whereas semantic memory is dependent on the embedding vectors and does not care about the meaning of a dimension, this meaning is provided by perception—the inner product with high-level features. We call this the grounding of embedding vectors.

The embedding vector is a prototypical vector for that concept in a high-dimensional representation space; the assumption is that in this space, distances are meaningful, and the complex mapping performed by the DCNN performs a normalization, such that, for example, all kinds of dogs with different shapes and appearances form a connected subspace in the semantic embedding space. In cognitive neuroscience, the semantic embedding space is sometimes referred to as conceptual space (Gärdenfors, 2016) where points denote objects, and regions denote concepts.

Figure 7 shows an analysis of the entity embeddings, based on the t-SNE visualization (Van der Maaten & Hinton, 2008). The plot clearly shows a schema-like organization of concept embeddings in this two-dimensional projection, also known as a cognitive map (Tolman, 1948).

### 7.6 Structure of the Embedding Vector

All memories are implemented as embedding vectors forming the matrix $A$. In accordance with the standard neurocognitive view, the entries in $A$ correspond to synaptic coupling strengths (Gazzaniga et al., 2013). In our approach, we implemented symmetric connections between the units in the index layer and the representation layer. Thus, we have connection matrices $A$ and $A^T$ in Figures 1 and 2. Although backward connections are common in the brain, they are typically not symmetrical. We did extensive experiments where we removed that constraint. The result was that the performance dropped by about 1% in basically all experiments, so we stayed with symmetric connections in our work.

The representation layer is high-dimensional, although embeddings might be sparse, and a given index only activates a small number of components of the representation layer. Naturally, the index that represents the color Red is likely to mostly connect to components of the representation layer that are excited by red images. Another advantage of sparse distributed representations (Rolls, 2016) is that this might lead to increased memory capacity (Ma, Hildebrandt et al., 2018). Sparsity in the embedding vectors can be achieved in technical models (for example, by using appropriate regularization terms, like Lasso (Tibshirani, 1996). In our experiments, when we applied Lasso on all parameters, we obtained 70% sparsity.
Figure 7: t-SNE visualization of embeddings of 15 classes and randomly selected entities from these classes. Features for the analysis are respective embedding vectors. A dot stands for an entity, for example, a specific dog, *Sparky*. The color of a dot marks the basis class of that entity. A cross stands for a class, for example, *Dog*, which is labeled by the same color as entities belonging to the class. The embedding vector of a class concept lies within the cluster of the classes’ entities. Recall that we are not indicating cluster centers but the embeddings that happened to be learned for the class concepts in learning. In cognitive neuroscience, the semantic embedding space is sometimes referred to as conceptual space, where points denote objects, and regions denote concepts (Gärdenfors, 2016).

7.7 Semantic Memory Experience: Semantic Decoding and Embodiment. The engram of a concept is identical to its embedding. But its treasure is only discovered by the semantic decoding. A complete semantic memory experience is not only the activation of a concept index and the activation of the representation layer with its embedding (i.e., its
grounding) but also the subsequent decoding of the embedding into triples. For example, after \( s = \text{Sparky} \) activates \( q \leftarrow a_s \), the agent can classify, for example, if \( s \) is \( c = \text{Friendly} \), producing the triple statement (\text{Sparky, hA, Friendly}). This exactly is going on in semantic memory decoding when we generate the sequential index pattern \text{Sparky, Friendly}. The decoding produces triples from the prior retrieval model which reflects the prior probabilities for triple statements.

In the dual subsymbolic view, the semantic decoding results in an embedding of the decoded sentences: activating the index for Sparky, for example, embodies Sparky’s embedding vector in \( \tilde{q}_S \); activating the index for Friendly then leads to a superposition of both embeddings in \( \tilde{q}_S \) and represents Sparky but with an emphasis that Sparky is friendly.

In a semantic memory recall, a second entity \( o \) can be activated, say, \( o = \text{Jack} \), and then the binary label \text{ownedBy} obtains high activation, producing the triple (\text{Sparky, ownedBy, Jack}). The subsymbolic representation of that statement is activated in \( q_P \).

Both grounding and semantic decoding are local to the entity and might integrate information on that entity that was collected at different episodic instances and in different modalities. Table 7 shows that in our model, semantic memory can realize a very precise memory recall.

Table 6 illustrates the semantic memory experience, which is triggered by an entity, a class, or an attribute. The latter two correspond to a recall of generalized statements (see section 3.7). For example, it is shown that semantic memory can recall general information on Dogs (that they are mammals with 100%), Mammals (that they are dogs with 38% and living beings with 100%), and the color Black (which, with 24% probability, is a property of a person).

7.8 Multimodality and Social Networks. An agent can obtain data from different modalities. This could concern subsymbolic sensory data besides vision or symbolic information from conversations, books, movies, and other media. Multimodal data become part of episodic memory and semantic memory and enrich perception in different ways. First, a multimodal episode is represented by an episodic index, and by activating that index, the data from that modality in that episode can be retrieved. Second, semantic memory serves as a site of multimodal integration simply because semantic memory is trained on triples from all modalities. So multimodal data will become part of semantic memory. Third, consider perception. We can now distinguish between visual labels—labels, which were used in the training images like the color \text{Black}—and multimodal labels, which are the remaining labels, like \text{Dangerous} or \text{Rich}. Table 8 shows how multimodality directly enters in perception. We call this enriched perception, P-enriched. The unary label \text{Dangerous} was not trained in perception but just in semantic memory. The table shows that information from the semantic memory is integrated into perception and episodic memory, even without
Table 6: Semantic Memory Experience with Generalized Statements on VRD-EX Data.

| $s^*$ (ID) | Unary Attribute/Class Labels | Unary Entity Labels ($p.\ sameAs$) | Binary Statements ($s^*, p, o^*$) |
|------------|------------------------------|-----------------------------------|----------------------------------|
| 10830[Person] | Person, 0.96  
Mammal, 0.97  
LivingBeing, 0.98  
Old, 0.98  
Other, 0.98  
Walking, 0.96 | 10830[Person] | (10830, wears, Shirt)  
(10830, wears, 3495[Glasses]) |
| Dog | Dog, 0.99  
Mammal, 1.0  
LivingBeing, 1.0  
Young, 0.52  
Brown, 0.35  
Other, 0.99 | 3537[Dog]  
602[Dog]  
5976[Dog] | (Dog, on, Grass)  
(Dog, behind, Person) |
| Mammal | Dog, 0.38  
Mammal 1.0  
LivingBeing, 1.0  
Young, 0.6  
Brown, 0.31  
Other, 0.96 | 3901[Cat]  
9100[Horse] | (Mammal, on, Street) |
| Black | Person, 0.24  
Other, 0.43  
NonLivingBeing, 0.95  
Old, 0.59  
Black, 0.99  
Other, 0.99 | 9812[Bag]  
3634[Keyboard] | (Black, on, Person)  
(Black, under, Sky) |

Notes: The first column shows the queried $s^*$ (an entity, a class, or an attribute), the input to the algorithm. The second column shows highly rated attribute and class labels describing $s^*$. The third column shows highly ranked (sampled) entities for the $sameAs$ predicate. The fourth column shows binary statements. We see that person 10830 is a mammal and often wears shirts and glasses. However, we also see that the model “explains” what the class Dog stands for, what the class Mammal stands for, and what the attribute Black stands for: We see that a dog is a mammal with 100% probability, is brown with 35% probability, and is often on the grass or behind a person. We see that if something is black, it is often a person (24%), and black entities are often “on persons” and “under the sky.” See our discussion on generalized statements in section 3.7.

activating an explicit semantic memory experience. Activating an entity index in perception or episodic memory subsequently activates the unary labels of that entity in context, but also recalls background of that entity, for example, relating to other modalities. Here it relies on semantic memory (see section 8.1).

To further examine multimodality, we derived a social network involving all persons in the data set by linking entities representing persons with the predicate knows. Each person is linked with five other persons, whom
Table 7: Semantic Memory Experience for Models Trained on the VRD-E Data Set.

| Model               | Binary Label | Unary Labels: Attributes/Classes |
|---------------------|--------------|----------------------------------|
|                     | @10          | B-Class | P-Class | G-Class | Y/O | Color | Act. |
| SM-givenClass       | 92.67        | 49.74   |         |         |     |       |      |
| RESCAL-givenClass   | 89.95        | 26.06   |         |         |     |       |      |
| SM-givenEntity      | 100.0        | 90.42   | 100.0   | 100.0   | 100.0 | 100.0 | 100.0 |
| RESCAL-givenEntity  | 100.0        | 90.12   | 100.0   | 100.0   | 100.0 | 100.0 | 100.0 |

Notes: In the first two rows, we set the class labels of the subject and the object, for example, Dog or Person, and predict binary labels. Our results are better than a state-of-the-art model (RESCAL). Ruffinelli, Broscheit, and Gemulla (2019) showed in a recent study that RESCAL is competitive with state-of-the-art approaches. In the third and fourth rows, we set the entity indices (e.g., Sparky or Jack). In this memorization, the performance is boosted by the entity representations. In these experiments, we are testing memory retrieval and not generalization, which explains the high scores. Best results in each block are in bold.

Table 8: Semantic Memory Experience Integrated with Perception.

| Model       | Classification Accuracy |
|-------------|-------------------------|
|             | Dangerous               |
| P-SA        | 52.01                   |
| SM          | 100.0                   |
| P-enriched  | 98.24                   |

Notes: The task is to predict the unary label Dangerous with or without semantic memory. P-SA in the first row is the perceptual system, where the label Dangerous was not provided in training. It can only be predicted by chance if an entity is dangerous. We trained the label Dangerous as part of semantic memory. In SM, the semantic memory experience is activated, which supplements the information from semantic memory if a visual entity is dangerous. P-enriched shows that when the semantic memory is trained with the Dangerous label, this information is also automatically integrated with perception, without an extra activation of a semantic memory experience. Enrichment works well with not-perceptual information, like social network background, which is, in a way, orthogonal to the visual scene input and an indication that statements become dependent in training by the shared embeddings. The best result is in bold.

we also denote as friends. In addition, we define a social network’s episodic instance as an instance at which the agent learns about the social contacts of one person. Our social network data set has 4987 person entities (along with their unary labels), 24953 knows statements, and 4987 social episodes.
Table 9: Episodic and Semantic Memory Experience for the Social Network Data VRD-S.

| Model          | $s^*$ | $o^*$ | Unary Labels of $s^*$ |
|----------------|-------|-------|-----------------------|
|                | @10   | @1    | B-Class  | P-Class | G-Class | Y/O | Color | Activity |
| Episodic Memory| 100   | 51.47 | 100.0    | 100.0    | 100.0    | 100.0    | 100.0    | 100.0    |
| Semantic Memory| –     | –     | 97.39    | 100.0    | 100.0    | 100.0    | 100.0    | 100.0    |
| RESCAL         | –     | –     | 95.76    | 17.96    |          |          |          |          |

Notes: The first row shows the performance of an episodic recall, with $t^*$ given. Columns 2 and 3 evaluate the prediction of a subject entity. For columns 3 to 10, the subject $s^*$ is also given. The second row shows the semantic memory experience where the subject $s^*$ is given. The unary labels have perfect recall. Columns 3 and 4 show the performance of predicting a friend. Since each person has five friends, the recall@1 is upper-bounded by 20%. As a comparison, we show the performance of RESCAL in the third row, which shows slightly worse performance.

We refer to this data set as VRD-S. More details on the generation of the social network data are in appendix C.

Table 9 shows numerical results on VRD-S. Given a person of interest, the semantic memory can recover friends (object $o^*$) and predict unary labels. Table 10 illustrates episodic and semantic memory, including social network data recall.

8 Reasoning and Language

8.1 Embedded Reasoning and Generalization. Although triple statements are independent, given all entries in the embedding matrix $A$, the entries in $A$ change with new data, and they have more the characteristics of latent variables than of fixed parameters. For example, $a_s$ changes when new truth values of statements involving $s$ become available. Thus, if it becomes known that an entity is a “Dog,” the unary statement for the label “Mammal” will exhibit a high probability. Similarly, if at a new instance $t'$ new information becomes available, $a_{t'}$ is adapted, and with only a few new known triples for a time instance $t'$, the BTN can make predictions about other statements relating to $t'$: Perception, episodic and semantic memory perform inference on symbolic triple sentences by adapting the subsymbolic embedding vectors. Let’s consider three types of generalization for an instance $t'$, that is, LCWA, unobserved-modality, and unobserved-entity generalization.

LCWA generalization concerns the entities for which data was acquired at $t'$ and for the modality of $t'$. Embedding reasoning is executed as just described.
Table 10: Episodic and Semantic Memory Experience That Includes Multi-modal Data (from a Social Network).

| t^* | s^*          | Unary Labels                        | o^*          | Binary Labels |
|-----|--------------|-------------------------------------|--------------|---------------|
| 2177| 14518 Person | Person, Mammal, LivB               | 14511 Bus    | on            |
|     | [Person]     | Young, Other, Sitting               |              |               |
| 14515| Building, Other, NonLivB | Old, White, Other               | 14516 Roof   | has           |
|     | [Building]   |                                     |              |               |
| 14511| Bus, Vehicle, NonLivB | Old, Other, Other          | 14512 Road   | nextTo (-)    |
|     | [Bus]        |                                     |              |               |
| 6662| 14518 Person | Person, Mammal, LivB               | 10669        | knows         |
|     | [Person]     | Young, Other, Sitting               |              |               |
| 18010| Person, Mammal, LivB | Young, Other, Other          | 14518        | knows         |
| 8318| Person, Mammal, LivB | Old, Other, Other          | 14518        | knows         |
| 14518| Person, Mammal, LivB | Young, Other, Sitting          | 10669        | knows         |
|     |              |                                     | 25066        | knows         |
|     |              |                                     | 12825 (-)    | knows (+)     |

Notes: The figure shows the original visual scene, which is unavailable during episodic recall. The scene index of the image is t^* = 2177. The top segment shows sampled subject entities (s^*), highly ranked unary labels (c^*), a sampled object o^*, and the top predicted binary label (p^*). All labels are correct except for one binary label (indicated by (-)). The second segment shows an episodic recall of a social episode with index (t^* = 6662). At that episodic instance, social network information (binary label knows) was provided, which is recovered in the episodic recall. Shown are three correct triples that were recovered in the sampling of the episodic memory. The bottom segment shows knows statements recovered from semantic memory for s^* = 14518 (thus without a recall of a special episodic memory). In section 8.1 we call this “unobserved-modality generalization.” Two binary statements are correct, and one is incorrect (indicated by (-)).

Unobserved-modality generalization concerns other modalities of the entities for which data was acquired at t’. It demonstrates multimodal integration. For observed s and o at t’, the prior retrieval model is applicable. For
example, if Jack is in the scene, unobserved-modality generalization might add that he is a friendly person and that he is married to Jane, and he probably likes the new movie, which just came out. Unobserved-modality generalization is based on \( \tilde{a} \) instead of \( a_t \). Figure 5 and Table 10 illustrate LCWA generalization and unobserved-modality generalization.

Unobserved-entity generalization concerns entities for which no data was acquired at \( t' \). For example, it will predict that Sparky is black, even when Sparky is not in the scene. Triples would be generated from a counterfactual situation: assuming an entity that was not in the scene was actually in the scene (the counterfactual assumption), which triple statements would likely be true? Unobserved-entity generalization, by default, also follows the prior retrieval model. The semantic memory experience in Table 6 illustrates unobserved-entity generalization. The interplay between the different forms of memory will be further discussed in section 9, in particular in section 9.9.

8.2 Symbolic Reasoning. In contrast to embedded reasoning, symbolic reasoning solely depends on indices and their relationships, without involving the representation layer. It would be completely amodal, in contrast to the modal embedded reasoning. Examples would be the definitional structure of a bachelor (a male person who is not and has never been married) and reasoning with subclass relationships (any dog is also a mammal). Definitional examples, like the definition of a bachelor, enter the mind of the agent via language and are crisp. They describe how the world should be. Symbolic reasoning can also link singular events with state changes. For example, one might specify that of someone gets married (a singular event), the state of this person changes from unmarried to married. Another example is the triangle rule in social networks: if A and B are known to be friends and A is known to be a friend of C, then B is also likely a friend of C. This example shows that symbolic reasoning is not necessarily deterministic. Descriptive dependencies, like the triangle rule, are learned by means of observations, are often probabilistic, and describe how the world actually is. Halford et al. (2014) has explored how analogical reasoning can be performed in the data tensor (a deterministic version of our prior tensor in equation 3.3), loaded in working memory. The work assigns symbolic reasoning to System-2. One might call it “inference by contemplation.” Nickel, Jiang, and Tresp (2014) is an example where both embodied and symbolic reasoning are combined in an additive model.

8.3 Embedded Symbolic Reasoning. Triple sentences used in symbolic reasoning have an embedding and lead to another form of embedded reasoning. A simple symbolic rule would be that a dog is a mammal, that is, \( \forall s : (s, hA, Mammal) \leftarrow (s, hA, Dog) \). The corresponding triple statement is \( Y_{Dog,hA,Mammal} \) is embedded and \( \mathbb{E}(Y_{Dog,hA,Mammal}) \) can be learned from data. Section 5.2 describes how the BTN’s embedding approach can produce
the sequential index pattern Sparky, Dog, Mammal, which is interpreted as Sparky being a Dog and a Mammal. We call this embedded symbolic reasoning. It builds on the same BTN architecture as perception and memory. Embedded symbolic reasoning is integrated in sampling. Assuming that, based on scene input, the unary label Mammal is active with 60% for a given visual entity, so the agent is uncertain about this label. Then, if sampling “decodes” that the visual entity is a Dog, embedded symbolic reasoning lifts the probability for the label Mammal to 100% (due to the learned rule that every dog is a mammal) and the corresponding unary label is sampled as well. Embedded symbolic reasoning can capture complex dependencies and thus can go beyond embedded reasoning. Embedded symbolic reasoning can be executed fast since it does not depend on a tuning of embeddings for entities or instances.

Table 6 shows generalized statements, which are the basis for embedded symbolic reasoning. We see that (Dog, hA, Mammal) is true with 100% (every dog is a mammal), and (Mammal, hA, Dog) is true with 38% (not every mammal is a dog). Also, Table 3 shows results for generalized binary statements.

**8.4 A Foundation for Consciousness?** The dynamic context layer might play a role in working memory. In general, working memory is associated with decision making and cognitive control (Baddeley, 1992) and is necessary for keeping task-relevant information active, as well as manipulating that information to accomplish behavioral tasks. A modern view is that working memory is distributed across the cortex (Buschman & Miller, 2020). There is an emerging consensus that most working memory tasks recruit a network of the prefrontal cortex (PFC) (front-of-the-brain hypothesis) and/or parietal areas in the prefrontal parietal network (PPN) (back-of-the-brain hypothesis) (Seth & Bayne, 2022).

PPN activity is consistently reported in both attention and consciousness studies (Bor & Seth, 2012). Their publication proposes that the PPN can be viewed as a “core correlate” of consciousness. Dehaene (2014) defines consciousness as “global information sharing” where information has entered into a specific storage area that makes it available to the rest of the brain. Christof Koch and colleagues (2016) argue that the posterior hot zone (PHZ) is the minimal neural substrate essential for conscious perception. The PHZ includes cortical sensory areas in the parietal, temporal, and occipital lobes.

The interactions between the index layer, the representation layer, and the dynamic context layer (which might involve working memory), might be a foundation from which evolution eventually generated human consciousness. The representation layer, associated with the parietal area of the brain, plays a major role in our approach, as well as its interaction with the dynamic context layer, which might include the working memory. The activation of the representation layer encodes the cognitive brain state. It integrates information and then makes this information available to the brain as
a whole. In particular, the activation of the representation layer represents a holistic, integrative view of the agent’s cognitive state at that instance. See also our discussion in the remaining part of this section. The role of the memory system in consciousness is discussed in Budson et al. (2022).

### 8.5 Serialization of Parallel Computing and the Central Bottleneck.

Here are some examples of compositionality and serial processing in the BTN, and possibly in the mind. First, perception analyzes a scene that evolves in time, which is a serial process. Second, an episode is composed of several events. Third, an event consists of several bounding boxes, which are analyzed serially. Fourth, many triples are generated serially to decode a scene semantically. Finally, a triple itself is represented as a sequence of indices (see Table 1).

On the other hand, any computation involving the representation layer—in particular, the mappings involving the DCNN—and the mappings between the index layer and the representation layer are highly parallel.

It is this mixture of parallel and sequential processing exhibited in our model that might also make up the brain’s operation. Consider that in a first step, we map $\text{scene}_t$ to $f(\text{scene}_t)$, and this representation can be analyzed rapidly and might trigger action with a limited understanding of scene content. Only then is bounding box content analyzed in detail, which might be supported by saccadic eye movements, and this is a slower serial process, permitting a transition to language, as discussed later in this section. Maybe not surprisingly, language makes thoughts explicit but at the same time might slow down thinking.

Sequential processing is also a core concept in the theory of a global workspace (Baars, 1997; Bor & Seth, 2012; Dehaene, 2014; Goyal et al., 2021): Koch (2014) discusses that Dehaene’s workspace has extremely limited capacity (“the central bottleneck”) and that the mind can be conscious of and pay attention to only one or a few items or events, although these might be quickly varying. In cognition neuroscience, the general understanding is that parallel multitasking of cognitive tasks likely is an illusion. Even working memory is assumed to be able to store only three to four items at a time (Awh & Vogel, 2020). Sequential processing would also contribute to a potential solution of the binding problem (Singer, 2001) since the decoding focuses on concepts in a serial fashion and associations between activities in the representation layer and the index layer are well defined. The importance of sampling is also recognized in Dehaene (2014) in the context of conscious perception. For example, the author states that “consciousness is a slow sampler.”

Another interesting point is that both Dehaene (2014) and Koch et al. (2016) assume mental states, well delineated from all the other states. Switching between different interpretations is also a property of our sampling approach if one interprets a sample as a temporary decision or an
interpretation: “It’s a bird, or a plane, or it’s Superman, but not all of them at the same time” (Dehaene, 2014). Dehaene discusses a similar process of “collapsing all unconscious probabilities into a single conscious sample.” His model assumes a “winning neural coalition”; in our approach, the embedding of an episodic instance integrates all information available at that instance, and the winning neural coalition would be the samples and triple sentences associated with that embedding. The optimization of the embedding of an episode, that is, \( \mathbf{a}_t \), is a step that occurs after the decoding of a scene and thus, the conscious experience associated with it might be slightly delayed (see also section 10.3). A current hypothesis in cognitive neuroscience is that conscious awareness often follows after an observation or a decision is made and serves to explain and justify but not to trigger an action (Gazzaniga et al., 2013; Budson et al., 2022). Only some important decisions are actually made with the help of conscious slow thinking.

In the semantic decoding of our model, the representation layer is periodically activated, which might be reflected in neural signals and could be related to some of the neural oscillations found in the brain. A candidate is the beta rhythm (13–35 Hz), considered to be related to consciousness, perception, and motor behavior. Also of relevance might be the gamma wave (25–140 Hz), which is correlated with large-scale brain network activity and cognitive phenomena such as working memory, attention, and perceptual grouping.

8.6 One-Brain Hypothesis. We discuss the one-brain hypothesis in the context of two aspects. First, the representation layer can contain a superposition of embedding vectors and activation vectors, but it cannot represent a concatenation of two vectors. This would, in a way, require two separate representation layers. We propose that this is biologically not plausible, and we adhere to Dehaene’s concept of a central bottleneck. Applied to perception, this means that the mind can only process one bounding box content at a time.

Second, perception, episodic memory, semantic memory, and even embedded symbolic reasoning are realized by the different operational modes of a single architecture; thus, we do not assume different modules for these different functions. Instead, the brain repurposes the same architecture for different functions.

Of course, the brain does many things concurrently, and many of these activities are hidden from conscious awareness. This does not contradict our one-brain hypothesis, which is only concerned with the proposed operations of perception and memory, and in particular, when they involve the representation layer, that is, the cognitive state of the brain.

8.7 Cognitive Linguistics. Humans differ from other animals in their ability to express themselves through natural language. Human language is the basis for communication but also a means to argue and reason. Thus, an
agent can tell another agent not to leave the hideout since a bear is lurking outside, even when the bear is not visible. Perception, episodic memory, and semantic memory are all declarative (i.e., explicit), and humans can verbally report on either. We propose that the generated triples in our approach are a basis from which the rich human language might have evolved.

Leading approaches in cognition and linguistics are, first, the formal approach, second, the connectionist approach, and third, the embodied approach (Evans, 2012). Our work relates to all three.

Let’s consider first the formal approach. The language of thought hypothesis assumes that mental representation has a linguistic structure as well: thoughts are sentences in the mind. Fodor (1975) describes the nature of thought as possessing “language-like” or compositional structure (sometimes referred to as mentalese). In this view, simple concepts combine systematically (akin to the rules of grammar in language) to build thoughts. Also, in our approach, the brain talks to itself by producing triple sentences and their embeddings. We also agree that language offers a window into the operation of the brain. However, in our approach, we do not follow Fodor’s formal logic-based view: our brain is more of a chatterbox.

Second, we follow a connectionist approach (McClelland et al., 2020), since the BTN includes a DCNN and neural processing. However, our notions of discrete symbolic indices, embodiment, and compositionality go beyond a more standard connectionist approach.

Finally, we can relate to the concept of an embodied language. There is general agreement that in a form of bottom-up processing, perception, the state of mind, and the body influence language. The idea of an embodied language is that in a form of top-down processing, language can influence the perceptual path and the body (Evans, 2012). As discussed, we see our approach as being embodied, at least as far it concerns earlier processing in the brain. One might even consider that language takes over the mind, particularly in memory, with no immediate visual input to be decoded. From a brain, talking to itself, of course, there is a small step to a brain that speaks to others. Most generated triple sentences are not transferred into language; what is actually spoken is obviously more complex, nuanced, and sophisticated, and is modulated, for example, by intent, social context, and cultural background. Thus an internal triple-oriented fast speech is transferred into an external, sophisticated slow speech. When individuals learn a new language, this might involve the mappings between fast speech and slow speech, in addition to the direct mapping from one language to another language. The inner fast speech might already exist in some animals.

If the agent is the recipient of language, say, in a conversation, by reading, or by consuming media, the acquired information could be stored as an episodic memory instance, that is, as an episodic index with its associated embedding vector. This knowledge then can become part of semantic memory. In our simplistic triple-oriented conversation, the agent can learn
about facts, such as \((Sparky, hA, Dog)\), and generalized statements, such as Hearst patterns (Hearst, 1992) like \((Dog, hA, Mammal)\).

Language compresses information. Significantly different scenes might generate very similar descriptions, demonstrating great invariance in triple descriptions and language in general.

9 Episodic Memory

9.1 Background on Episodic Memory. Episodic memory documents the life of an agent. Tulving (1985) describes episodic memory as a memory that, in contrast to semantic memory, requires a recollection of a prior experience. It is considered to be the result of rapid associative learning in that a single episode and its context become associated and bound together and can be retrieved from memory after a single episode. Episodic memory stores information of general and personal events (Tulving, 1972, 1985, 2002; Gazzaniga, Ivry, & Mangun, 2013) and concerns information we “remember,” including the spatiotemporal context of events (Gluck, Mercado, & Myers, 2013).

Some theories emphasize the sequential nature of episodic memory and the memory process. Moscovitch, Cabeza, Winocur, and Nadel (2016) considers an episodic memory experience to be an active process that involves details of the event and its location. Sometimes the reconstruction is regarded as a Bayesian process of reconstructing the past as accurately as possible based on available engram information (Hemmer & Steyvers, 2009).

9.2 Event Memory and Episodic Memory. In this article, we use the terms event memory and episodic memory almost synonymously. In the narrower sense, we consider event memory to be related to a single instance in time and episodic memory to a sequence of events, that is, a story. This is in agreement with Mannila, Toivonen, and Inkeri Verkamo (1997), who define an episode as a collection of events that occur relatively close to each other in a given partial order. We propose that the distinction between event memory and episodic memory is blurred. First, we often remember rather static images of past episodes. Second, even an analysis of a static scene is a sequential process, where each bounding box is recovered in sequence. Third, during a single saccade, the scene might have already changed: Consider, for example, an agent riding a bicycle where the scenery is constantly changing. So the recovery of a single scene might already describe a dynamic event.

9.3 Recall of Episodic Memory Engrams. An activated past episodic index \(t\) restores its embedding \(a_t\), that is, its engram, in the representation layer. Figure 8 shows that the embedding vectors of episodic memories form meaningful maps and also are organized as a conceptual space.
Figure 8: t-SNE visualization of episodic instances based on their embeddings \( \{a_t\}_{t=1}^{N_t} \). One can see that similar scenes are often in proximity. For example, the circled areas show images of indoor scenes (bottom) and of buses on streets (right).

9.4 Episodic Memory Experience: Semantic Decoding and Embodiment. The embedding of an episodic index is all there is, but the treasures of an episodic memory are only deciphered by the semantic decoding of that embedding, the symbolic reconstruction of triples involving that episode (see Table 1). The recall of the episodic embedding, but especially the semantic decoding into triples, has the character of a simulation: Episodic memory is a reactivation of a possibly multimodal memory experience (Evans, 2012) and has been described as a reliving of past experience (Shapiro, 2010). Figure 9 illustrates an episodic memory experience. Table 11 provides numerical results.
Figure 9: Episodic memory experience using VRD-E data. The first row shows the visual input to perception and then extracted bounding boxes for entities in the scene. In the second row, we show an episodic recall of this scene ($t^* = t'$), which is now a memory. We show the bounding boxes of the scene entities, which were recalled. We see that in sampling, episodic recall recovers entities with correct bounding box content (e.g., fire hydrant, sky, car). Columns 6 to 8 show incorrect recalls. Note that these incorrect recalls would still be entirely plausible in the scene’s context.

Table 11: Experiments Using Episodic Memory.

|       | $s^*$ | Unary labels | Binary labels |
|-------|-------|--------------|---------------|
|       | &@10 & @1 | B-Class | P-Class | G-Class | Y/O | Color | Act. | &@10 | @1  |
| EM    | 99.84 37.13 | 100.0 | 100.0 | 100.0 | 100.0 | 100.0 | 100.0 | 100.0 | 90.40 |
| P-noI | 0.0    0.0 | 41.19 20.83 | 51.76 49.03 | 12.97 71.05 | 39.34 7.79 |

Notes: Top row: episodic memory experience using VRD-E data. For randomly selected past episodic instances of $t^*$ as input, we determine the highest-ranked entity indices (first two columns). For the other columns, we set the correct entity indices ($s^*$ and $o^*$) and predict unary labels and binary labels. The performance is, in many cases, 100%. Thus, an agent might recall seeing a black dog but might be less confident that it was Sparky. Considering the large number of entities in the data set, the performance on entity prediction (first two columns) is still impressive as well. Bottom row: In “P-noI” we removed all entity indices, using only class and unary labels. The bad performance demonstrates the relevance of representing entities for memory recall.

9.5 Episodic and Semantic Memory in Perception. We propose that in perception, an agent first uses both the episodic and semantic attention approximation, which can be executed fast and in parallel. Only in a second step are specific entities sampled. This permits the integration of specific multimodal background (see section 7.8), and permits associations with past episodic instances and known entities. Thus, if perception poses the hypothesis that an entity in the image is identical to Sparky, then the semantic memory experience would supplement specific background information on Sparky, which cannot be derived from the visual input. Semantic
Figure 10: Recent episodic memory experience. An illustration of the effect of a recent episodic memory experience using VRD-E data. The left image shows a harmless garden scene, but due to a recall of a recent episodic memory $t^* = 684$, the agent is aware of the lurking bear close by (right). Labels for visual entity recovered in the episodic recall (right) are Bear, Mammal, LivingBeing, Old, Black, OtherActivity, Dangerous. Note that episodic recall is not triggered by closeness in a scene but by recency and relevance.

memory reconstructs what is known about the concept (the prior information). Similarly, episodic memory would recall relevant past episodes.

9.6 Recent Episodic Memories for Context. Recent episodic memory can provide the agent with information on recent perceptual experiences. A recall is triggered by the recency of the episodic index and relevance, and it contributes to an agent’s sense of the world state. A recent episodic memory experience is an episodic memory that is almost treated as a current observation. An agent needs to know about the state of the world, even for parts that are not currently being perceived. An example is the “lurking-bear” situation (see Figure 10): “There was a bear strolling around outside the hideout. Remember: it might still be there, although it might not be visible from the hideout.” Figure 11 shows another example of a recent episodic memory recall. This emphasis on episodic closeness can be implemented as “time encoding” (Ma, Tresp et al., 2018), in a similar way as “position encoding” is used in the attention literature (Vaswani et al., 2017). Patients who are unable to form new episodic memory show great deficits in personal orientation and context understanding. These deficits are often associated with severe bilateral damage to MTL (Gluck et al., 2013).

9.7 Remote Episodic Memory for Decision Support. Remote episodic memory can provide the agent with information on remote perceptual experiences similar to the current perceptual experience, and this can contribute to decision making. An episodic recall is triggered by $t = t^*$ (a sample is in equation 4.8) which reflects the similarity between the current scene representation with the episodic embedding. It makes sense that the
The Tensor Brain

Figure 11: Recent episodic memory experience. (Top) The left image shows visual input to perception from VRD-EX data. Then a recent $t^*$ is sampled. The right image shows the image belonging to $t^*$. The table shows results from this episodic memory experience without the actual image for $t^*$ being available, only based on memory recall. The first column shows samples for $s^*$. The second column shows unary labels. The third column shows sampled objects $o^*$. The fourth column shows the most likely binary label.

| $s^*$ | Unary Labels                      | $o^*$ | Binary Label |
|------|-----------------------------------|-------|--------------|
| 346  | Person, 1.00, Mammal, 1.00, LivingBeing, 1.00 Young, 1.00, Other, 1.00, Other 1.00 | 345   | on           |
| 348  | Shirt, 1.00, Clothing, 1.00, NonLivingBeing, 1.00 Young, 1.00, Orange, 1.00, Other, 1.00 | 347   | on           |
| 347  | Person, 1.00, Mammal, 1.00, LivingBeing, 1.00 Young, 1.00, Other, 1.00, Playing, 1.00 | 348   | wear         |
| 345  | Grass, 1.00, Plant, 1.00 LivingBeing, 1.00 Old, 1.00, Green, 1.00, Other, 1.00 | 346   | under        |

current event should trigger the same action as in the retrieved episodic memory—if it led to a good outcome—or an alternative action if not. For example, if the agent finds the current situation very similar to a previous one, where it had walked toward a bear and almost got attacked, it would very likely not do this again! Remote episodic memory provides information about possible future scenarios and aids the agent in decision making.

Thus memory guides behavior. Duncan and Shohamy (2016) describe this process as integration across relational events by imaging possible rewards in the future. The value associated with a memory (e.g., reward, threat) might be an integral aspect of episodic memory. The paper also states that there is now extensive empirical data supporting the prevalent use of episodic memory across various decision-making tasks in humans.

Figure 12 shows that a perceptual scene indeed can activate memories of similar scenes in remote episodic memory.
Figure 12: (Top) Remote episodic memory experience using VRD-E data. The two images in the first column show the visual input to perception. Then we sample $t^*$, representing past episodic memories. The images of the scenes associated with the $t^*$ show that, indeed, recalled past episodic memories are related. (Bottom) The bounding boxes in the first column represent visual entities $s'$ in perception. The right bounding boxes display retrieved entities with high activation.

9.8 Future Episodic Memories. A future episodic memory is a forecast event in the future, which at some point in time is predicted to become a regular episodic memory. This helps plan future actions. Events might be predicted using, for example, temporal knowledge graph models (Han, Chen, Ma, & Tresp, 2020), assuming temporal smoothness and predictability of the embedding of time instances. Here we consider forecasts made entirely on a symbolic level, say, by verbal communication or reading. For example, if the agent has learned by verbal, symbolic communication that there is a football match in the stadium this evening and the weather will be bad, embedded reasoning, based on this assumed future information, predicts that there will be a traffic jam in the city and that driving will be difficult. Technically, the BTN for future episodic memories is identical to regular episodic memories, only that the episodic index embedding is calculated based on assumed events in the future, either without or with imagined and simulated future sensory input.

9.9 Memory Supports the Agent in the Present and the Future. It is of great interest for an agent to estimate the state of the world at the current instance $t'$ and predict how it will be in the future agent’s immediate
The Tensor Brain

Figure 13: The horizontal axis stands for time and the vertical for some abstract triple dimension. The current instance is $t'$ on the right. The light blue background box stands for the predictions of the semantic-state model of the event memory. The triple retrieval model of the episodic memory is represented by the horizontal gray bars; they indicate where, in the past, information was acquired. The left vertical gray bar represents the semantic memory. The dark blue bars represent the prior retrieval model and indicate where semantic memory is reliable. At episodic instance $t'$, the agent learns about some statements by perception (orange), and these become part of the event memory (recent event memory). There might be an association with another portion of the event memory, that is, the remote event memory (red dotted arrow). If the agent contemplates information unrelated to current perception, it relies on remote event memory and semantic memory (blue dotted lines). The right vertical bar indicates a future episodic memory.

environment. We propose that the mind estimates the world’s state using perception and memory: The memory systems provide information that makes the agent act right but is not communicated by current perceptual experience.

For example, to illustrate the relevance of memory in daily life, consider a typical day at the office of agent Mary (see Figure 13). When Mary arrives at the office in the morning, she expects that everything is as usual, as modeled by semantic memory. Semantic memory will produce statements describing the state of normality. This sets the stage.

Perception will produce triples describing the new situation, for example, the status of the coffee machine, which is broken. She immediately informs Jack in an office nearby: “Jack, can you believe this?! The coffee machine is broken, again!” (from triples to language). Remote episodic memory reminds her that this is not the first time that the coffee machine was broken.
Recent episodic memory will remind Mary all day that the coffee machine is broken, even when she is not in the same room as the coffee machine and thus does not have immediate perceptual information on its status. As discussed, recent episodic memory is an episodic memory that is almost treated as a current observation. Actively invoking the morning’s memory by replay enables Mary to integrate this recent event into her awareness. Since the coffee machine has mostly been working, semantic memory by itself would predict (incorrectly) that the coffee machine most likely is working. The state change of the coffee machine is only slowly integrated into semantic memory. Being a long-term average, semantic memory is a deficient state estimator and relies on recent episodic memory to integrate recent state changes.

Jack might ask Mary if the coffee machine had been working last Tuesday. If she does not have an episodic recall from last Tuesday about the status of the coffee machine, she might consult semantic memory, reminding her that the coffee machine mostly has been working. But if she has an episodic memory that last Monday, the coffee machine was broken, she might infer that it most likely was also broken on Tuesday. Here, she consults a memory that is recent to the past time instance of interest.

Another recent episodic memory might remind Mary that she had met Jane on her way to work and that she had said that she would drop by at the office (simulation of a recent episodic memory). Jane is a good friend (semantic memory).

Mary recognizes that Sparky is in the office (perception), and semantic memory adds triples describing background on Sparky, for example, that Sparky is Jack’s dog (unobserved-modality generalization). Mary might also recall that a while ago, Sparky was at the office and behaved well (remote episodic memory), although dogs in general can be quite a nuisance (as concluded by a generalized statement and by employing embedded symbolic reasoning).

When Mary relaxes, she suddenly recalls that people are repairing the heater today (episodic memory, not triggered by perception) and that she should call her roommate to check if they arrived in time.

Then Mary remembers that the football match will be in the stadium this evening and that the weather has been predicted to be bad (future episodic event): there will likely be a traffic jam (generalization of future episodic events), and due to expected bad weather conditions, she should drive carefully (an association stored in semantic memory). Future-state prediction uses both episodic memory and generalized future episodic memory.

The story illustrates that the default state estimate is provided by semantic memory that is supplemented by perception or other modalities. Recent episodic memory provides information that is not yet or will not be absorbed into semantic memory but contributes to the current state. Future episodic memory supports planning. Remote episodic memory guides behavior: the last time Sparky behaved well, and he might behave well again.
The whole story is on a symbolic level. But each index that fires is embedded and leads to embodiment. We propose that the storyline involves System-1-type associative reasoning, supported possibly by some background knowledge on how the world works. As our story suggests, to deal with daily life, agents might need little effortful System-2 reasoning.

10 To Perceive Is to Learn

There is nothing in the mind that was not first in the senses.

John Locke

10.1 Overview. In normal operations, a new perceptual event that becomes an episodic memory requires the establishment of a new episodic index and its associated embedding vector. This roughly corresponds to the MTL-based fast nonparametric learning system in the complementary learning systems (CLS) theory (McClelland et al., 1995; Kumaran et al., 2016). A new perceptual event might contain novel entities not yet known to the agent. For important ones, the agent needs to establish new indices and their embedding vectors as well. More rarely, new indices for attributes, classes, and predicates need to be established. Being quite stable, those might be represented in neocortex.

The second component of the CLS-theory is the parametric learning system, where the neocortex is trained by replay in a slow process from data from the nonparametric learning system. Slow training serves as the basis for the gradual acquisition of structured knowledge about the environment and its transfer to neocortex (Kumaran et al., 2016). An agent might get new information on existing entities, and this information might need to be integrated into their embeddings. In some applications involving online adaptation, interference of new knowledge with old ones can be a problem and might lead to catastrophic forgetting (Kumaran et al., 2016). Catastrophic forgetting does not show up in our preliminary experiments. The large dimensionality and modularity of the embedding vector lead to robustness and stability; for example, learning first about Jack’s size and hair color might not interfere with the learning of his social network later. And when he later dyes his hair, the changes in his embedding vector are local, that is, it affects only a few dimensions. Learning happens as an online process, so more recent episodes will have a greater effect than more distant ones.

10.2 Neuroscience Perspective on Establishing Episodic Memory Engrams. Episodic memories are first formed in the hippocampus, which is part of medial temporal lobe (MTL). The idea that episodic memory is index based is by now one of several accepted theories (Tonegawa,
Morrissey, & Kitamura, 2018). It goes back to the hippocampal memory indexing theory (Teyler & DiScenna, 1986; Teyler & Rudy, 2007), which was long controversial. The event indices have a relational memory function in that they bind together different pieces of experience. Recent research found evidence for the existence of time cells in the hippocampus (CA1) (Eichenbaum, Sauvage, Fortin, Komorowski, & Lipton, 2012; Eichenbaum, 2014; Kitamura, Sun et al., 2015; Kitamura, Macdonald, & Tonegawa, 2015).

There is some evidence that neurogenesis might be involved in forming new episodic memories. Neurogenesis by special stem cells has been discovered in the dentate gyrus (part of the hippocampal formation) and is active throughout adult life; these new neurons may be preferentially recruited in the formation of memories. In fact, it has been observed that the adult macaque monkey forms a few thousand new neurons daily (Gluck et al., 2013; Gould, Reeves, Graziano, & Gross, 1999), possibly to encode new information (Becker, 2005).

The establishment of new indices, together with their embedding vectors, are some of the most demanding learning tasks in the brain. Functionally, our model assumes that a new episodic memory engram is quickly stored by establishing a new index and its connections to the representation layer, copying the episodic memory trace. Although there exist several theories, little is known about how exactly new time indices are formed in the brain anatomically and how they quickly set up the bidirectional connection patterns with the representation layer, forming a hippocampal–cortical network (Frankland & Bontempi, 2005). Here, the brain might employ already existing networks, which might also explain how a single index can influence the distributed representation layer with potentially numerous neurons. A structural intra-hub connectivity might facilitate this process.

10.3 Experiments with Self-Supervised Learning. Self-supervised learning (SSL) concerns learning without teacher-provided explicit training labels. It is biologically highly relevant. During most of life, an individual has to learn without explicit supervision (lifelong learning). In our approach, self-supervised learning works exactly like supervised training; the difference is that the agent’s predicted winner-take-all unary and binary labels become the training labels. This form of self-supervised learning is a type of bootstrap learning; see the bootstrap Widrow-Hoff rule (Hinton & Nowlan, 1990) and learning with pseudo-labels (Lee, 2013). The generated data are then used to train our model using cross-entropy cost-function terms derived from perception, episodic, and semantic memory (see appendix D).

In our SSL experiments (see Table 12), we establish new indices and their embeddings for perceptional episodes (episodic indices) and for new entities in those episodes (concept indices); this would be part of the fast nonparametric learning system. The self-supervised training for a novel
Table 12: Self-Supervised Learning on the VRD-E Data.

| Training Regime | Test Set | Unary Labels (accuracy) | B-Class | P-Class | G-Class | Y/O | Color | Activity | Average |
|-----------------|---------|-------------------------|---------|---------|---------|-----|-------|----------|---------|
| SL SL-D         |         | 100.0 100.0 100.0 100.0 100.0 100.0 | 100.0 100.0 100.0 100.0 100.0 100.0 |
| SL SL-D         |         | 85.51 88.36 93.16 48.32 59.42 79.78 | 75.76 |
| SL G            |         | 77.43 85.79 92.63 49.08 62.35 81.39 | 74.78 |
| +SSL SL-D       |         | 77.54 86.15 92.83 48.58 62.62 82.78 | 75.08 |

Notes: SL stands for supervised learning and SSL for self-supervised learning. We first trained the model (SL) in a supervised way with human-provided labels (SL-D) with only 50% of the training images (2000 images). Columns 3 to 9 show perfect semantic memory performance for unary labels (first row). Then we continued to train the model with only the other 50% of the training images (SSL-D) in the SSL mode (+SSL). The second row shows that self-supervised learning does not lead to a deterioration of prediction performance on the entities in SL-D. The third row shows that the performance on SSL-D is quite good, but of course, not as good as on SL-D since predicted labels on these data are noisier than the human-provided training labels. Rows 4 and 5 show performance in perception on new entities (generalization, G). The performance of +SSL is better than SL, which shows that self-supervised learning also improves the detection of classes and attributes, helping to fine-tune their embeddings.

Episodic instance and a novel entity⁵ is modular and fast, both technically and most likely also biologically. The embedding vector of a new episodic instance \( a_t' \) is adapted to model the decoded triple statements. Decoded triple statements, which are more certain, will be decoded more often, will be more influential in memory formation, and will then be retrieved more likely in an episodic memory recall. Decoded triple statements, which are more certain, will be decoded more often, will be more influential in memory formation, and will then be retrieved more likely in an episodic memory recall. Row 3 in the table shows that the semantic recall on entities is reasonable, but not as good as on entities trained with annotator-labeled data (see row 1). This is understandable since predicted labels are noisier than human-annotator-provided training labels.

Now consider the slow parametric learning system. Row 2 shows that the embeddings of already established entities are not negatively affected by SSL. Rows 4 and 5 show perception performance on unseen entities. We see better performance after SSL has been applied. This shows that embeddings for classes and attributes are improved by SSL.

10.4 Replay for Episodic Memory Consolidation. As self-supervised learning adds information, the required memory capacity grows. In our

⁵The agent might decide that an entity is novel if the activation of all elements in \( \text{sig}(n_5) \) are below a threshold for all known entities.
approach, for each salient episode, a new episodic index with its embedding vector is established. Similarly, for each new entity, a new concept index with its embedding vector is added. The current thinking in neuroscience is that this capacity problem can be solved by a consolidation from MTL, with a limited capacity, to the neocortex, with an essentially unlimited capacity. This is called systems consolidation of memory (SCM).

We first consider the leading two theories about the consolidation of episodic memory. The standard theory assumes that at some point, episodic memory becomes independent of the hippocampus and MTL over a period of weeks to years (Squire & Alvarez, 1995; Frankland & Bontempi, 2005). In contrast, the multiple trace theory assumes that both the hippocampus and MTL remain involved (Nadel & Moscovitch, 1997; Jonides et al., 2008; Greenberg & Verfaellie, 2010); MTL remains the manager of complex spatial and relational memories (Whittington et al., 2020). In general, it is assumed that consolidation involves both the medial prefrontal cortex (mPFC) and the MTL. The temporal lobe might be where the transferred indices are established (Frankland & Bontempi, 2005; Tonegawa et al., 2018). After consolidation, episodic memories might be organized in temporal order or according to a similarity in representation (see Figure 8). It is assumed that consolidation might be a process executed entirely or partially during sleep (Stickgold, 2005).

We propose consolidation by replay, which, in our model, can be executed as follows. An episodic index $t$ in MTL is activated, which activates the representation layer with vector $\mathbf{a}_t$; this activation is then learned in the connection weights of a newly formed index in the neocortex, for example, by a form of Hebbian learning. Thus, these index duplicates in the neocortex would inherit the connection weights. If the index in the neocortex becomes more distributed, this will lead to greater robustness of memories after consolidation. For a while, both representations exist in parallel, and this facilitates the learning and consolidation of new memories; gradually, the index representation in the neocortex might become dominant. Consolidation by replay has the advantage that there is no need for direct interactions of indices in both storage sites, only indirect interactions by a shared activation of the representation layer. Replay might also be one way of how the brain implements large-scale structural changes in the brain, in general, for example, as a consequence of brain damage or as a consequence of a changing world with new statistics. Following the principle “use it or lose it,” replay might also be essential such that relevant information in consolidated memory is not forgotten. From our model’s perspective, the location of the index is irrelevant. It could be in MTL, in neocortex, or both at the same time. In either case, decoding relies on the same machinery (see Figure 2). Cognitive maps might be more pronounced after consolidation in the neocortex (Binder & Desai, 2011).
10.5 Replay for Semantic Memory Consolidation. Training by episodic memory replay might also be important for the gradual transition from episodic to semantic memory, in which episodic memory reduces its sensitivity and association to particular episodes so that the information can be generalized as semantic memory. It might enable the semantic memory to adapt more quickly to state changes, for example, to a friend’s status change from being single to being married. Some theories speculate that episodic memory may be the gateway to semantic memory (Baddeley & Hitch, 1974; Squire, 1987; Baddeley, 1988; Steyvers, Shiffrin, & Nelson, 2004; Socher et al., 2009; McClelland et al., 1995; Yee, Chrysikou, & Thompson-Schill, 2014; Kumar et al., 2015).

11 Summary, Conclusions, and Future Work

We have shown how perception, memory, reasoning, and foundations for language and consciousness can all be realized by different functional and operational modes of the oscillating interactions between an index layer and a representation layer in the BTN. Whereas the representation layer, as a global workspace, is prominent in the current discussion on consciousness, the introduced index layer is an original contribution. We have emphasized the role of episodic and semantic memory in perception. We have proposed an associative memory where recency is key to the recall of recent episodic memory and similarity of episodic representations with the scene representation for remote episodic memory. Semantic memory recall is triggered by the sampled concept. Our approach explains the great similarity between episodic and semantic memory: semantic memory is the expected episodic memory of a future instance.

We have developed a technical model in which perception and subsymbolic and symbolic processing are realized by one integrated neural network structure. Perception and memories first produce subsymbolic representations, which are subsequently decoded semantically to produce symbolic triple sentences. Our work suggests that perception and episodic and semantic memory all rely on the same brainware (one-brain hypothesis). Our paper contributed to the discussion (a recent example being Browning & LeCun, 2022) of how language, thought, and subsymbolic processing interact.

As part of future work, we will address more systematically subsumption hierarchies and how parts bind to form the whole, which is another form of compositionality addressed, for example, in capsule networks (Sabour, Frosst, & Hinton, 2017). The agent itself is also an important concept: it is part of each event. It might be quite relevant if the angry bear looks at a deer or at the agent itself. Other issues of the “I” are mood, state of mind, current mission, social contexts, and spatial location.

In this article, we did not focus on spatial representations. It is well known that MTL is not only instrumental for forming novel episodic
memories but also contains spatial representations, such as in the form of grid cells and place cells. It is generally assumed that MTL is central for reconstructing spatial memories, permitting complex spatial reasoning. A similar reconstruction of complex relational networks for episodic and semantic memories into a narrative might be realized in MTL as well (Whittington et al., 2020). The integration of spatial information and spatial reasoning in our approach is part of future work, including mental navigation in the connected graphs formed by the combined spatial and relational networks (Whittington et al., 2020).

The brain does many different things simultaneously and addresses a particular problem with several strategies. Our work emphasizes the role of perception and memory, and it might provide some insights into part of the amazing faculties of human intelligence.

Appendix A: Bernoulli versus Multinomial Models

Here we summarize the different BTN models for episodic memory, semantic memory, generalized statements, and perception. All models are coupled by shared embedding vectors. The semantic-state model is based on two-state Bernoulli variables, whereas the triple-retrieval model is based on a multinomial distribution. We train the BTN using cross-entropy cost function terms derived from the multinomial model. The softmax representation in the BTN implicitly implements the required normalizations. Training the multinomial model does not require the application of an LCWA. The interface between the multinomial and the two-state Bernoulli model is as follows: If, for a given instance \( t \), the multinomial model generates an \((s, p, o)\) or \((s, c)\) sample, then the random variable \(Y_{s,p,o,t}\) is assumed true.

A.1 Two-State Bernoulli Variables. The BTN’s semantic-state model has \(N_C N_P N_T\) two-state Bernoulli random variables. \(y_{s,p,o,t}\) is a measurement of the random variable \(Y_{s,p,o,t}\). Each random variable represents a propositional statement, that is, a ground atom. All discussed models share the same embeddings and are thus dependent, but given the embeddings, we obtain independence.

A.1.1 Semantic-State Model for Episodic Memory. \(E(Y_{s,p,o,t})\) is an estimate of \(y_{s,p,o,t}\) based on the BTN (see equation 3.1). The semantic-state model evaluates the truth values of triple statements in episodic memory. In modeling, the KL-divergence between the empirical distribution of the observed data and the model is minimized. This leads to the maximization of

\[
\prod_{(s,p,o,t) \in \text{LCWA}} P(y_{s,p,o,t} | a_s, a_p, a_o, a_t).
\]
This is not an i.i.d. model. Here $\mathbb{P}$ refers to the Bernoulli model. The product is over all measured triples and all triples as concluded by the LCWA to be false. Note that this is an approximating distribution where each triple statement is independent; these independences would not be present in the actual data.

A.1.2 Prior Model for Semantic Memory. For a new instance $\bar{t}$ with no observed data (cf. equation 4) we use the prior model

$$\mathbb{E}(Y_{s,p,o,t}).$$

The prior model evaluates the truth values of future triple statements as modeled in semantic memory. In modeling, the KL-divergence between observed data and the model is minimized. This leads to the maximization of

$$\prod_{(s,p,o,t) \in \text{LCWA}} \mathbb{P}(y_{s,p,o,t} | a_s, a_p, a_o).$$

This is an i.i.d. model: triple statement probabilities are independent of time.

A.1.3 Modeling Dependencies: Generalized Statements. Here we go beyond the semantic state model and model dependencies between the random variables $\{Y_{s,p,o,t}\}_{s,p,o,t}$. For example (see equation 3.6), (with LCWA),

$$\mathbb{E}(Y_{s',hA,c_2,t'} | Y_{s',hA,c_1,t'}) \approx \frac{1}{\sum_s \sum_t y_{s,hA,c_1,t} \sum_s \sum_t y_{s,hA,c_2,t}} \sum_s \sum_t y_{s,hA,c_1,t} \sum_t y_{s,hA,c_2,t}, \quad (A.1)$$

is the probability that at a new instance $t'$, $s'$, which has unary label $c_1$, will also have unary label $c_2$. We define this to be the interpretation of the expected value for the generalized triple statement $(c_1, hA, c_2)$. Generalized statements are useful for predictions at a new $t'$ where only partial measurements are available. With a predicted unary label “Dog” and using the generalized statement (Dog, $hA$, Mammal), the unary label “Mammal” can be predicted.

A.1.4 Perception. We condition on perceptual input (see equation 4.10),

$$\mathbb{E}(Y_{s',p,o',t'} | o' = o, s' = s, t' = t, BB_{sub}, BB_{obj}, BB_{pred}, scene).$$

A.2 Multinomial Sampling: Retrieval Probabilities. $\mathbb{P}(s, p, o, t)$ has four random variables. The number of states is $N_s^2 N_p N_T$, so there is one state for each two-state Bernoulli random variable.
A.2.1 Triple Retrieval Model for Episodic Memory. The triple retrieval model generates samples for episodic memory. For a given instance \( t \), the agent generates samples from \( P(s, p, o | t) \) (cf. equation 3.2). If \( i_{s, p, o, t} = 1 \), then the contribution to the cross-entropy cost function for binary labels is

\[-(\log P(p | s, o, t) + \log P(o | s, t) + \log P(s | t))\]

and for unary labels \(- \log P(c | s, t)\).

A.2.2 Prior Retrieval Model for Semantic Memory. The triple retrieval model generates samples for semantic memory. If \( i_{s, p, o, t} = 1 \), then the contribution to the cross-entropy cost function for binary labels is

\[-(\log P(p | s, o, \tilde{t}) + \log P(o | s, \tilde{t}))\]

and for unary labels \(- \log P(c | s, \tilde{t})\).

A.2.3 Modeling Dependencies: Generalized Statements. For the statement \((c_1, hA, c_2)\), the BTN directly trains \( P(c_2 | c_1) \), which is applicable for a new \( \tilde{s} \) and a new \( \tilde{t} \). If \( i_{s, hA, c_1, t} = 1 \) and \( i_{s, hA, c_2, t} = 1 \), then the contribution to the cross-entropy cost function is

\[-\log P(c_2 | c_1)\].

A.2.4 Perception. In perception (see section 4.3),

\[P(s' = s, p' = p, o' = o, t' = t | BB_{sub}, BB_{obj}, BB_{pred}, scene_t').\]

If \( i_{s, p, o, t} = 1 \), then the contribution to the cross-entropy cost function is

\[-\log P(s' = s, p' = p, o' = o, t' = t | BB_{sub}, BB_{obj}, BB_{pred}, scene_t').\]

Appendix B: Conditional Equivalence and Counterfactual Querying

As discussed, the interface between the multinomial and the two-state Bernoulli model is via sampling. If the multinomial model generates an \((s, p, o)\) or \((s, c)\) sample, then the random variable \( Y_{s, p, o, t} \) is assumed true. Here, we show a close mathematical relationship between the probabilistic scores. The key correspondence is that for entities that were observed in the episodic experience at \( t \),

\[P(s, p, o, t) \approx E(Y_{s, p, o, t})/N_{total}.\]
From this, one can derive

\[ P(p|s, o, t) \approx \frac{E(Y_s, p, o, t)}{\sum_{p'} E(Y_s, p', o, t)}, \]  
(B.1)

\[ P(c|s, t) \approx \frac{E(Y_s, hA, c, t)}{\sum_{c'} E(Y_s, hA, c', t)}, \]  
(B.2)

and for semantic memory,

\[ P(p|s, o, \bar{t}) \approx \frac{E(Y_s, p, o, \bar{t})}{\sum_{p'} E(Y_s, p', o, \bar{t})}, \]  
(B.3)

\[ P(c|s, \bar{t}) \approx \frac{E(Y_s, hA, c, \bar{t})}{\sum_{c'} E(Y_s, hA, c', \bar{t})}, \]  
(B.4)

In the last two equations, we have exploited that due to the LCWA, \( N_{s, p, o} = N_{s, p', o} \) and \( N_{s, c} = N_{s, c'} \). B.1 to B.5 generally hold. See also the discussion in section 8.1.

Appendix C: Social Network

We consider all 4987 persons in the data set and link a person \( s \) to persons \( s' \) if the score \( a_s^T a_{s'} \) is in the top five of all scores related to \( s \). Thus, links exist between persons with similar embeddings, simulating homophily. We then determine the link direction. Considering two entities \( s \) and \( s' \), \( \exp \beta \|a_s\|/(\|a_s + a_{s'}\|) \) is proportional to the probability that we determine that \((s, knows, s')\), otherwise, \((s', knows, s)\). At a social network episodic time step \( t \), all links to one person \( s \) are added. This defines 4987 episodes for the tKG. The pKG then aggregates the tKG. Overall, we have 24,953 \textit{knows} statements. In summary, our social network data set has 4987 person entities (along with their unary labels), 24,953 friendship statements, and 4987 episodes of social events.

Appendix D: Implementation Details

In this section, we focus on the implementation aspects and provide some details about the network architecture and training hyperparameters. Our program is written in Python and utilizes PyTorch.

D.1 Network Architecture. Our model mainly consists of two layers. The representation layer \( q \) has \( r = 4096 \) neurons. The index layer \( n \) has \( N_T + N_C + N_P \) neurons. We use VGG-19 backbone as the deep neural network \( f(\cdot) \), which takes as input a scene or a bounding box and outputs a 4096-dimensional feature vector. The VGG-19 network consists of a sequence of convolutional blocks followed by two fully connected hidden
layers. Each convolutional block is a sequence of two convolution layers with $3 \times 3$ filters, a max-pooling layer, and another two convolution layers with the same parameters. We use the activations before the nonlinear transformation from the last hidden layer and copy them over to $q$. Thus, the layer $q$ stores preactivation values in the range of $\mathbb{R}$. Next, we apply a nonlinear function (LeakyReLU) to the values in $q$ and feed them to the index layer via connection weights $A^\top$. At different decoding steps, $n$ covers different sets of indices: $n_T$ has $N_T$ units for episodic instances; $n_S$, $n_C$, $n_O$, all have $N_C$ concept units (entities, classes, attributes), and $n_P$ has $N_P$ units for predicates. The index layer in turn activates the representation layer via the same weights $A$. To calculate the enhanced representation $q_T$, $q_S$, and $q_O$, we add the preactivation of layer $q$ and corresponding activations from $n$. To obtain $n_S$ and $n_O$, we apply softmax on the output with an inverse temperature $\beta = 1$. For $n_C$ we split the concepts into eight sets and apply softmax on each set. Alternatively, we could also use the sigmoid function as stated in our algorithm. However, softmax fits here as our labels are mutually exclusive, and in practice, softmax leads to faster convergence and slightly better performance. The dynamic context layer $h$ contains 500 neurons with a self-connection via weight matrix $B$. The input to the dynamic context layer is postactivation of $q$, that is, values after a nonlinear transformation. There is a direct path for the hidden layer between different decoding steps (the dotted line between $h$ blocks in Figure 2), which stores the state of the working memory and leads to a slight improvement (1%) in relationship prediction. For $\bar{a}$, we use a learnable embedding vector of length 4096.

D.2 Training Scheme. We train a global set of parameters for our BTN with an objective of minimizing the multitask loss for perception, episodic, and semantic memory experience. We set the batch size to 128, the learning rate to 0.0001, and dropout $p = 0.5$. The BTN is optimized using an Adam optimizer for 60 epochs. During training, we freeze the VGG-19 layers except for its last fully connected layer so that the knowledge in the pretrained model will not be destroyed. For the last layer of VGG-19, we use a smaller learning rate of 0.00001 to allow it to adapt to the new task. Except for the VGG-19 backbone, we initialize our network using Kaiming uniform initialization proposed in He, Zhang, Ren, and Sun (2015). We minimize the summed cross-entropy loss on $n_T$, $n_S$, $n_C$, $n_O$, and $n_P$ for the perception and memory experience. For $n_S$ and $n_O$, we randomly activate an index from the set of entity indices and class/attribute labels for learning generalized statements (see section 3.7). For $n_C$, we apply cross-entropy loss on each subset. For the social network, we train the model for attribute prediction and relationship prediction on the social network data set. Concretely, we minimize the cross-entropy loss for $n_S$, $n_C$, $n_O$, and $n_P$. For SSL, we use a batch size of 128, a learning rate of 1e-5, and 20 training epochs. Except for the embeddings of time and entities, all weights are frozen. Regarding the
ablation studies, we train the RESCAL model using the implementation of PyKeen (Ali et al., 2021). The rank of entity embeddings and predicate embeddings are both set to 1000, which gives a comparable number of learnable parameters as our model. All experiments are conducted on an Nvidia GTX 1089 Ti GPU with a four core CPU of 16G memory.

Code

The source code related to our article is available at the following link: https://github.com/hangligit BTN

Acknowledgments

We thank the anonymous reviewers for their helpful comments on the manuscript.

References

Ali, M., Berrendorf, M., Hoyt, C. T., Vermue, L., Sharifzadeh, S., Tresp, V., & Lehmann, J. (2021). Pykeen 1.0: A Python library for training and evaluating knowledge graph embeddings. Journal of Machine Learning Research, 22(82), 1–6.

Anderson, P., He, X., Buehler, C., Teney, D., Johnson, M., Gould, S., & Zhang, L. (2018). Bottom-up and top-down attention for image captioning and visual question answering. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

Awh, E., & Vogel, E. (2020). Online and off-line memory states in the human brain. In D. Poeppel, G. Mangun, & M. Gazzinga (Eds.), The cognitive neurosciences. MIT Press.

Baars, B. J. (1997). In the theater of consciousness: The workspace of the mind. Oxford University Press.

Baddeley, A. (1988). Cognitive psychology and human memory. Trends in Neurosciences, 11(4), 176–181. 10.1016/0166-2236(88)90145-2

Baddeley, A. (1992). Working memory. Science, 255(5044), 556–559. 10.1126/science.1736359

Baddeley, A. D., & Hitch, G. (1974). Working memory. Psychology of Learning and Motivation, 8, 47–89. 10.1016/S0079-7421(08)60452-1

Baier, S., Ma, Y., & Tresp, V. (2017). Improving visual relationship detection using semantic modeling of scene descriptions. In ISWC, 1, 53–68.

Baier, S., Ma, Y., & Tresp, V. (2018). Improving information extraction from images with learned semantic models. In Proceedings of the 27th International Joint Conference on Artificial Intelligence (pp. 5214–5218).

Barsalou, L. W. (2008). Grounded cognition. Annual Review of Psychology, 59(1), 617–645. 10.1146/annurev.psych.59.103006.093639

Bassett, D. S., & Sporns, O. (2017). Network neuroscience. Nature Neuroscience, 20(3), 353. 10.1038/nn.4502

Becker, S. (2005). A computational principle for hippocampal learning and neurogenesis. Hippocampus, 15(6), 722–738. 10.1002/hipo.20095
Binder, J. R., & Desai, R. H. (2011). The neurobiology of semantic memory. Trends in Cognitive Sciences, 15(11), 527–536. 10.1016/j.tics.2011.10.001

Bloice, M. D., Stocker, C., & Holzinger, A. (2017). Augmentor: An image augmentation library for machine learning. arXiv:1708.04680.

Bor, D., & Seth, A. K. (2012). Consciousness and the prefrontal parietal network: Insights from attention, working memory, and chunking. Frontiers in Psychology, 3, 63. 10.3389/fpsyg.2012.00063

Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., & Yakhnenko, O. (2013). Translating embeddings for modeling multi-relational data. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, & K. Q. Weinberger (Eds.), Advances in neural information processing systems, 26. Curran.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., . . . Amodei, D. (2020). Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, & H. Lin (Eds.), Advances in neural information processing systems, 33, 1877–1901.

Browning, J., & LeCun, Y. (2022). AI and the limits of language. Noema Magazine, August 23.

Buschman, T. J., & Miller, E. K. (2020). How working memory works. In D. Poeppel, G. Mangun, & M. Gazzinga (Eds.), The cognitive neurosciences (6th ed., p. 357). MIT Press.

Dayan, P., Hinton, G. E., Neal, R. M., & Zemel, R. S. (1995). The Helmholtz machine. Neural Computation, 7(5), 889–904. 10.1162/neco.1995.7.5.889

de Heer, W. A., Huth, A. G., Griffiths, T. L., Gallant, J. L., & Theunissen, F. E. (2017). The hierarchical cortical organization of human speech processing. Journal of Neuroscience 37(27), 6539–6557. 10.1523/JNEUROSCI.3267-16.2017

Dehaene, S. (2014). Consciousness and the brain: Deciphering how the brain codes our thoughts. Penguin.

Dettmers, T., Minervini, P., Stenetorp, P., & Riedel, S. (2018). Convolutional 2D knowledge graph embeddings. In Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence. AAAI Press.

Duncan, K. D., & Shohamy, D. (2016). Memory states influence value-based decisions. Journal of Experimental Psychology: General, 145(11), 1420.

Eichenbaum, H. (2014). Time cells in the hippocampus. Nature Reviews Neuroscience, 15(11). 10.1038/nrr3827

Eichenbaum, H., Sauvage, M., Fortin, N., Komorowski, R., & Lipton, P. (2012). Towards a functional organization of episodic memory in the medial temporal lobe. Neuroscience and Biobehavioral Reviews, 36(7), 1597–1608. 10.1016/j.neubiorev.2011.07.006

Evans, J. S. B. (2003). In two minds: Dual-process accounts of reasoning. Trends in Cognitive Sciences, 7(10), 454–459. 10.1016/j.tics.2003.08.012

Evans, V. (2012). Cognitive linguistics. Cognitive Science, 3(2), 129–141.

Fellbaum, C. (2010). Wordnet. In R. Poli, M. Healy, & A. Kameas (Eds.), Theory and applications of ontology: Computer applications (pp. 231–243). Springer.

Fodor, J. A. (1975). The language of thought, vol. 5. Harvard University Press.

Frankland, P. W., & Bontempi, B. (2005). The organization of recent and remote memories. Nature Reviews Neuroscience, 6(2), 119–130. 10.1038/nrn1607

Friston, K. (2010). The free-energy principle: A unified brain theory? Nature Reviews Neuroscience, 11(2), 127–138. 10.1038/nrn2787
Friston, K. J., Tononi, G., Sporns, O., & Edelman, G. (1995). Characterising the complexity of neuronal interactions. *Human Brain Mapping, 3*(4), 302–314. 10.1002/hbm.460030405

Gärdenfors, P. (2016). *The geometry of meaning: Semantics based on conceptual spaces*. MIT Press.

Gazzaniga, M. S., Ivry, R. B., & Mangun, G. R. (2013). *Cognitive neuroscience: The biology of the mind* (4th ed.). Norton.

Gluck, M. A., Mercado, E., & Myers, C. E. (2013). *Learning and memory: From brain to behavior*. Palgrave.

Gould, E., Reeves, A. J., Graziano, M. S., & Gross, C. G. (1999). Neurogenesis in the neocortex of adult primates. *Science, 286*(5439), 548–552. 10.1126/science.286.5439.548

Goyal, A., Didolkar, A., Lamb, A., Badola, K., Ke, N. R., Rahaman, N., . . . Bengio, Y. (2021). Coordination among neural modules through a shared global workspace. arXiv:2103.01197.

Graves, A., Wayne, G., & Danihelka, I. (2014). *Neural Turing machines*. arXiv:1410.5401.

Greenberg, D. L., & Verfaellie, M. (2010). Interdependence of episodic and semantic memory: Evidence from neuropsychology. *Journal of the International Neuropsychological Society, 16*(5), 748–753. 10.1017/S1355617710000676

Griffiths, T. L., Kemp, C., & Tenenbaum, J. B. (2008). Bayesian models of cognition. In R. Sun (Ed.), *The Cambridge handbook of computational psychology*. Cambridge University Press.

Hackbusch, W. (2012). *Tensor spaces and numerical tensor calculus*. Springer.

Halford, G. S., Wilson, W. H., & Phillips, S. (1998). Processing capacity defined by relational complexity: Implications for comparative, developmental, and cognitive psychology. *Behavioral and Brain Sciences, 21*(6), 803–831. 10.1017/S0140525X98001769

Halford, G. S., Wilson, W. H., Andrews, G., & Phillips, S. (2014). *Categorizing cognition: Toward conceptual coherence in the foundations of psychology*. MIT Press.

Han, Z., Chen, P., Ma, Y., & Tresp, V. (2020). Explainable subgraph reasoning for forecasting on temporal knowledge graphs. In *Proceedings of the International Conference on Learning Representations*.

Harnad, S. (1990). The symbol grounding problem. *Physica D: Nonlinear Phenomena, 42*(1–3), 335–346. 10.1016/0167-2789(90)90087-6

He, K., Zhang, X., Ren, S., & Sun, J. (2015). Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification. In *Proceedings of the IEEE International Conference on Computer Vision* (pp. 1026–1034).

Hearst, M. A. (1992). Automatic acquisition of hyponyms from large text corpora. In *COLING 1992 Volume 2: The Proceedings of the 14th International Conference on Computational Linguistics*.

Hemmer, P., & Steyvers, M. (2009). A Bayesian account of reconstructive memory. *Topics in Cognitive Science, 1*(1), 189–202. 10.1111/j.1756-8765.2008.01010.x

Hinton, G. E., & Nowlan, S. J. (1990). The bootstrap Widrow-Hoff rule as a cluster formation algorithm. *Neural Computation, 2*(3), 355–362. 10.1162/neco.1990.2.3.355
Hintzman, D. L. (1984). Minerva 2: A simulation model of human memory. *Behavior Research Methods, Instruments, and Computers*, 16(2), 96–101. 10.3758/BF03202365

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8), 1735–1780. 10.1162/neco.1997.9.8.1735

Hommel, B., Müsseler, J., Aschersleben, G., & Prinz, W. (2001). The theory of event coding (TEC): A framework for perception and action planning. *Behavioral and Brain Sciences*, 24(5), 849–878. 10.1017/S0140525X0000103

Hudson, D., & Manning, C. D. (2019). Learning by abstraction: The neural state machine. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d’Alché-Buc, E. Fox, & R. Garnett (Eds.), *Advances in neural information processing systems*, 32 (pp. 5901–5914). Curran.

Humphreys, M. S., Bain, J. D., & Pike, R. (1989). Different ways to cue a coherent memory system: A theory for episodic, semantic, and procedural tasks. *Psychological Review*, 96(2), 208. 10.1037/0033-295X.96.2.208

Huth, A. G., de Heer, W. A., Griffiths, T. L., Theunissen, F. E., & Gallant, J. L. (2016). Natural speech reveals the semantic maps that tile human cerebral cortex. *Nature*, 532, 453–458. 10.1038/nature17637

Johnson, J., Krishna, R., Stark, M., Li, L.-J., Shamma, D., Bernstein, M., & FeiFei, L. (2015). Image retrieval using scene graphs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 3668–3678).

Jonides, J., Lewis, R. L., Nee, D. E., Lustig, C. A., Berman, M. G., & Moore, K. S. (2008). The mind and brain of short-term memory. *Annual Review of Psychology*, 59, 193. 10.1146/annurev.psych.59.103006.093615

Jung, A. B., Wada, K., Crall, J., Tanaka, S., Graving, J., Reinders, C., . . . Laporte, M. (2020). Augmentor: An image augmentation library for machine learning. https://github.com/aleju/imgaug

Kahneman, D. (2011). *Thinking, fast and slow*. Macmillan.

Kanerva, P. (1988). *Sparse distributed memory*. MIT Press.

Kiefer, M., & Pulvermüller, F. (2012). Conceptual representations in mind and brain: Theoretical developments, current evidence and future directions. *Cortex*, 48(7), 805–825. 10.1016/j.cortex.2011.04.006

Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. arXiv:1412.1698.

Kitamura, T., Macdonald, C. J., & Tonegawa, S. (2015). Entorhinal–hippocampal neuronal circuits bridge temporally discontinuous events. *Learning and Memory*, 22(9), 438–443. 10.1101/lm.038687.115

Kitamura, T., Sun, C., Martin, J., Kitch, L. J., Schnitzer, M. J., & Tonegawa, S. (2015). Entorhinal cortical ocean cells encode specific contexts and drive contextspecific fear memory. *Neuron*, 87(6), 1317–1331. 10.1016/j.neuron.2015.08.036

Knill, D. C., & Pouget, A. (2004). The Bayesian brain: The role of uncertainty in neural coding and computation. *Trends in Neurosciences*, 27(12), 712–719. 10.1016/j.tins.2004.10.007

Koch, C. (2014). Keep it in mind. *Scientific American Mind*, 25(3), 26–29.

Koch, C., Massimini, M., Boly, M., & Tononi, G. (2016). Neural correlates of consciousness: Progress and problems. *Nature Reviews Neuroscience*, 17(5), 307.

Koner, R., Shit, S., & Tresp, V. (2020). Relation transformer network. arXiv:2004.06193.

Körding, K. P., Ku, S.-p., & Wolpert, D. M. (2004). Bayesian integration in force estimation. *Journal of Neurophysiology*, 92(5), 3161–3165.
Kriegeskorte, N., & Douglas, P. K. (2018). Cognitive computational neuroscience. *Nature Neuroscience*, 21(9), 1148–1160. 10.1038/s41593-018-0210-5

Krishna, R., Zhu, Y., Groth, O., Johnson, J., Hata, K., Kravitz, J., . . . Shamma, D. A. (2017). Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123(1), 32–73. 10.1007/s11263-016-0981-7

Kumar, A., Irsoy, O., Su, J., Bradbury, J., English, R., Pierce, B., . . . Socher, R. (2015). *Ask me anything: Dynamic memory networks for natural language processing*. arXiv:1506.07285.

Kumaran, D., Hassabis, D., & McClelland, J. L. (2016). What learning systems do intelligent agents need? *Trends in Cognitive Sciences*, 20(7), 512–534. 10.1016/j.tics.2016.05.004

Lakoff, G., Johnson, M., & Sowa, J. F. (1999). Review of philosophy in the flesh: The embodied mind & its challenge to Western thought. *Computational Linguistics*, 25(4), 631–634.

Lee, D.-H. (2013). Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on Challenges in Representation Learning*. ICML.

Leopold, D. A., Strick, P. L., Bassett, D. S., Bruno, R. M., Cuntz, H., Harris, K. M., . . . Raichle, M. E. (2019). Functional architecture of the cerebral cortex. In W. Singer, T. Sejnowski, & P. Rakic (Eds.), *The neocortex* (pp. 141–164). MIT Press.

Lu, C., Krishna, R., Bernstein, M., & Fei-Fei, L. (2016). Visual relationship detection with language priors. In *Proceedings of the European Conference on Computer Vision*. Springer.

Luo, R., Zhang, N., Han, B., & Yang, L. (2019). Context-aware zero-shot recognition. arXiv:1904.09320.

Ma, Y., Hildebrandt, M., Baier, S., & Tresp, V. (2018). Holistic representations for memorization and inference. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*. AUAI Press.

Mannila, H., Toivonen, H., & Inkeri Verkamo, A. (1997). Discovery of frequent episodes in event sequences. *Data Mining and Knowledge Discovery*, 1(3), 259–289. 10.1023/A:1009748302351

McClelland, J. L., Hill, F., Rudolph, M., Baldridge, J., & Schütze, H. (2020). Placing language in an integrated understanding system: Next steps toward human level performance in neural language models. In *Proceedings of the National Academy of Sciences*, 117(42), 25966–25974. 10.1073/pnas.1910416117

McClelland, J. L., McNaughton, B. L., & O’Reilly, R. C. (1995). Why there are complementary learning systems in the hippocampus and neocortex: Insights from the successes and failures of connectionist models of learning and memory. *Psychological Review*, 102(3), 419. 10.1037/0033-295X.102.3.419

Montague, R. (1970). Universal grammar. *Theoria*, 36(3), 373–398. 10.1111/j.1755-2567.1970.tb00434.x

Moscovitch, M., Cabeza, R., Winocur, G., & Nadel, L. (2016). Episodic memory and beyond: The hippocampus and neocortex in transformation. *Annual Review of Psychology*, 67, 105–134. 10.1146/annurev-psych-113011-143733
Nadel, L., & Moscovitch, M. (1997). Memory consolidation, retrograde amnesia and the hippocampal complex. *Current Opinion in Neurobiology, 7*(2), 217–227. 10.1016/S0959-4388(97)80010-4

Nickel, M., Jiang, X., & Tresp, V. (2014). Reducing the rank in relational factorization models by including observable patterns. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, & K. Q. Weinberger, (Eds.), *Advances in neural information processing systems 27* (pp. 1179–1187). Curran.

Nickel, M., Murphy, K., Tresp, V., & Gabrilovich, E. (2015). A review of relational machine learning for knowledge graphs. In *Proceedings of the IEEE*, 104(1).

Nickel, M., Rosasco, L., & Poggio, T. (2015). *Holographic embeddings of knowledge graphs*. arXiv:1510.04935.

Nickel, M., Tresp, V., & Kriegel, H.-P. (2011). A three-way model for collective learning on multi-relational data. In *Proceedings of the 28th International Conference on International Conference on Machine Learning* (pp. 809–816).

Nickel, M., Tresp, V., & Kriegel, H.-P. (2012). Factorizing YAGO: Scalable machine learning for linked data. In *Proceedings of the 21st International Conference on World Wide Web* (pp. 271–280).

Noy, N., Rector, A., Hayes, P., & Welty, C. (2006). Defining n-ary relations on the semantic web. *W3C Working Group Note, 12*(4).

Ogden, C. K., & Richards, I. A. (1923). *The meaning of meaning: A study of the influence of thought and of the science of symbolism*. Harcourt.

Osth, A. F., & Dennis, S. (2015). Sources of interference in item and associative recognition memory. *Psychological Review, 122*(2), 260. 10.1037/a0038692

Pellionisz, A., & Llinás, R. (1980). Tensorial approach to the geometry of brain function: Cerebellar coordination via a metric tensor. *Neuroscience, 5*(7), 1125–1136. 10.1016/0306-4522(80)90191-8

Plate, T. (1997). A common framework for distributed representation schemes for compositional structure. In F. Maire, R. Hayward, & J. Diederich (Eds.), *Connectionist systems for knowledge representation and deduction* (pp. 15–34). Queensland University of Technology.

Poggio, T., Banburski, A., & Liao, Q. (2020). Theoretical issues in deep networks. In *Proceedings of the National Academy of Sciences*, 117(48), 30039–30045. 10.1073/pnas.1907369117

Pollack, J. B. (1990). Recursive distributed representations. *Artificial Intelligence, 46*(1), 77–105. 10.1016/0004-3702(90)90005-K

Pope, M., Fukushima, M., Betzel, R. F., & Sporns, O. (2021). Modular origins of high-amplitude cofluctuations in fine-scale functional connectivity dynamics. In *Proceedings of the National Academy of Sciences*, 118(46), e2109380118. 10.1073/pnas.2109380118

Popp, M., Trumpp, N. M., & Kiefer, M. (2019). Processing of action and sound verbs in context: An FMRI study. *Translational Neuroscience, 10*(1), 200–222. 10.1515/tnsci-2019-0035

Quiroga, R. Q. (2012). Concept cells: The building blocks of declarative memory functions. *Nature Reviews Neuroscience, 13*(8). 10.1038/nnrn3251

Quiroga, R. Q., Reddy, L., Kreiman, G., Koch, C., & Fried, I. (2005). Invariant visual representation by single neurons in the human brain. *Nature, 435*(7045), 1102–1107. 10.1038/nature03687
The Tensor Brain

Ralph, M. A. L., Jefferies, E., Patterson, K., & Rogers, T. T. (2017). The neural and computational bases of semantic cognition. *Nature Reviews Neuroscience*, 18(1), 42. 10.1038/nrn.2016.150

Rao, R. P., & Ballard, D. H. (1999). Predictive coding in the visual cortex: A functional interpretation of some extra-classical receptive-field effects. *Nature Neuroscience*, 2(1), 79–87. 10.1038/4580

Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster R-CNN: Towards real-time object detection with region proposal networks. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, & R. Garnett (Eds.), *Advances in neural information processing systems*, 28 (pp. 91–99). Curran.

Rolls, E. T. (2016). *Cerebral cortex: Principles of operation*. Oxford University Press.

Rueschemeyer, S.-A., van Rooij, D., Lindemann, O., Willems, R. M., & Bekkering, H. (2010). The function of words: Distinct neural correlates for words denoting differently manipulable objects. *Journal of Cognitive Neuroscience*, 22(8), 1844–1851. 10.1162/jocn.2009.21310

Ruffinelli, D., Broscheit, S., & Gemulla, R. (2019). You can teach an old dog new tricks! On training knowledge graph embeddings. In *Proceedings of the International Conference on Learning Representations*.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., . . . Bernstein, M. (2015). ImageNet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3), 211–252. 10.1007/s11263-015-0816-y

Sabour, S., Frosst, N., & Hinton, G. E. (2017). Dynamic routing between capsules. In I. Guyon, Y. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), *Advances in neural information processing systems*, 30. Curran.

Seth, A. K., & Bayne, T. (2022). Theories of consciousness. *Nature Reviews Neuroscience*, 23(7), 439–452. 10.1038/s41583-022-00587-4

Shapiro, L. (2010). *Embodied cognition*. Routledge.

Sharifzadeh, S., Baharlou, S. M., & Tresp, V. (2020). Classification by attention: Scene graph classification with prior knowledge. In *Proceedings of the National Conference on Artificial Intelligence*. AAAI.

Sharifzadeh, S., Berrendorf, M., & Tresp, V. (2019). Improving visual relation detection using depth maps. arXiv:1905.00966.

Simonyan, K., & Zisserman, A. (2014). *Very deep convolutional networks for largescale image recognition*. arXiv:1409.1556.

Singer, W. (2001). Consciousness and the binding problem. *Annals of the New York Academy of Sciences*, 929(1), 123–146. 10.1111/j.1749-6632.2001.tb05712.x

Smolensky, P. (1990). Tensor product variable binding and the representation of symbolic structures in connectionist systems. *Artificial Intelligence*, 46(1), 159–216. 10.1016/0004-3702(90)90007-M

Socher, R., Chen, D., Manning, C. D., & Ng, A. (2013). Reasoning with neural tensor networks for knowledge base completion. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, & K. Q. Weinberger (Eds.), *Advances in neural information processing systems*, 26. Curran.

Socher, R., Gershman, S., Sederberg, P., Norman, K., Perotte, A. J., & Blei, D. M. (2009). A Bayesian analysis of dynamics in free recall. In Y. Bengio, D. Schuurmans, J. Lafferty, C. Williams, & A. Culotta (Eds.), *Advances in neural information processing systems*, 22 (pp. 1714–1722). Curran.
Sporns, O. (2018). Graph theory methods: Applications in brain networks. *Dialogues in Clinical Neuroscience, 20*(2), 111. 10.31887/DCNS.2018.20.2/osporns
Squire, L. R. (1987). *Memory and brain*. Oxford University Press.
Squire, L. R., & Alvarez, P. (1995). Retrograde amnesia and memory consolidation: A neurobiological perspective. *Current Opinion in Neurobiology, 5*(2), 169–177. 10.1016/0959-4388(95)80023-9
Steyvers, M., Shiffrin, R. M., & Nelson, D. L. (2004). Word association spaces for predicting semantic similarity effects in episodic memory. In A. F. Healy (Ed.), *Experimental cognitive psychology and its applications: Festschrift in honor of Lyle Bourne, Walter Kintsch, and Thomas Landauer* (pp. 237–249). American Psychological Society.
Stickgold, R. (2005). Sleep-dependent memory consolidation., *Nature, 437*(7063), 1272–1278. 10.1038/nature04286
Sukhbaatar, S., Weston, J., & Fergus, R. (2015). End-to-end memory networks. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, & R. Garnett (Eds.), *Advances in neural information processing systems, 28* (pp. 2440–2448). Curran.
Sun, R., & Peterson, T. (1996). Learning in reactive sequential decision tasks: The clarification model. In *Proceedings of International Conference on Neural Networks* (2:1073–1078). IEEE.
Wu, Y., Kirillov, A., Massa, F., Lo, W.-Y., and Girshick, R. (2019). Detectron2. https://github.com/facebookresearch/detectron2
Tenenbaum, J. B., Griffiths, T. L., & Kemp, C. (2006). Theory-based Bayesian models of inductive learning and reasoning. *Trends in Cognitive Sciences, 10*(7), 309–318. 10.1016/j.tics.2006.05.009
Teyler, T. J., & DiScenna, P. (1986). The hippocampal memory indexing theory. *Behavioral Neuroscience, 100*(2), 147–15410.1037/0735-7044.100.2.147
Teyler, T. J., & Rudy, J. W. (2007). The hippocampal indexing theory and episodic memory. *Hippocampus, 17*(12), 1158–1169. 10.1002/hipo.20350
Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological), 58*(1), 267–288. 10.1111/j.2517-6161.1996.tb02080.x
Tolman, E. C. (1948). Cognitive maps in rats and men. *Psychological Review, 55*(4), 189.
Tomasello, R., Garagnani, M., Wennekers, T., & Pulvermüller, F. (2017). Brain connections of words, perceptions and actions: A neurobiological model of spatiotemporal semantic activation in the human cortex. *Neuropsychologia, 98*, 111–129. 10.1016/j.neuropsychologia.2016.07.004
Tonegawa, S., Morrissey, M. D., & Kitamura, T. (2018). The role of engram cells in the systems consolidation of memory. *Nature Reviews Neuroscience, 19*(8), 485. 10.1038/s41583-018-0031-2
Tresp, V., Esteban, C., Yang, Y., Baier, S., & Krompaß, D. (2015). *Learning with memory embeddings*. arXiv:1511.07972.
Tresp, V., & Ma, Y. (2016). The tensor memory hypothesis. In *NIPS Workshop on Representation Learning*.
Tresp, V., Ma, Y., & Baier, S. (2017). Tensor memories. In T. J. Teyler & J. W. Rudy (2007). The hippocampal indexing theory and episodic memory. In *Proceedings of the Hippcamp Conference on Cognitive Computational Neuroscience*. 
Tresp, V., Ma, Y., Baier, S., & Yang, Y. (2017). Embedding learning for declarative memories. In European Semantic Web Conference (pp. 202–216). Springer.

Tresp, V., Sharifzadeh, S., & Konopatzki, D. (2019). A model for perception and memory. In Proceedings of the Conference on Cognitive Computational Neuroscience.

Tresp, V., Sharifzadeh, S., Konopatzki, D., & Ma, Y. (2020). The tensor brain: Semantic decoding for perception and memory. arXiv:2001.10027.

Trouillon, T., Welbl, J., Riedel, S., Gaussier, E., & Bouchard, G. (2016). Complex embeddings for simple link prediction. In Proceedings of the International Conference on Machine Learning (pp. 2071–2080).

Tulving, E. (1972). Episodic and semantic memory 1. In E. Tulving & W. Donaldson (Eds.), Organization of memory Academic Press.

Tulving, E. (1985). Elements of episodic memory. Oxford University Press.

Tulving, E. (2002). Episodic memory: From mind to brain. Annual Review of Psychology, 53(1), 1–25. 10.1146/annurev.psych.53.100901.135114

van den Heuvel, M. P., & Sporns, O. (2013). Network hubs in the human brain. Trends in Cognitive Sciences, 17(12), 683–696. 10.1016/j.tics.2013.09.012

Van der Maaten, L., & Hinton, G. (2008). Visualizing data using t-SNE. Journal of Machine Learning Research, 9(11).

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, ... Polosukhin, I. (2017). Attention is all you need. In I. Guyon, Y. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), Advances in neural information processing systems, 30 (pp. 5998–6008). Curran.

Weston, J., Chopra, S., & Bordes, A. (2014). Memory networks. arXiv:1410.3916.

Whittington, J. C., Muller, T. H., Mark, S., Chen, G., Barry, C., Burgess, N., & Behrens, T. E. (2020). The Tolman-Eichenbaum machine: Unifying space and relational memory through generalization in the hippocampal formation. Cell, 183(5), 1249–1263.

Wu, Y., Kirillov, A., Massa, F., Lo, W-Y., & Girshick, R. (2019). Detectron2. https://github.com/facebookresearch/detectron2

Yang, B., Yih, W.-T., He, X., Gao, J., & Deng, L. (2014). Embedding entities and relations for learning and inference in knowledge bases. arXiv:1412.6575.

Yang, J., Lu, J., Lee, S., Batra, D., & Parikh, D. (2018). Graph R-CNN for scene graph generation. In Proceedings of the European Conference on Computer Vision (pp. 670–685). Springer.

Yee, E., Chrysikou, E. G., & Thompson-Schill, S. L. (2014). The cognitive neuroscience of semantic memory. Oxford University Press.

Zellers, R., Yatskar, M., Thomson, S., & Choi, Y. (2018). Neural motifs: Scene graph parsing with global context. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 5831–5840).

Zhang, H., Kyaw, Z., Chang, S.-F., & Chua, T.-S. (2017). Visual translation embedding network for visual relation detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 5532–5540).