Neuro-Symbolic Modeling: Overview

Technical challenges for neural-symbolic integration

This lecture

- The Two Systems of Thinking
- Learning & Reasoning
- History: Statistical relation learning
- Some examples of neural-symbolic integration
- Technical challenges for neural-symbolic integration
- A taxonomy of approaches

This lecture

- The Two Systems of Thinking
- Learning & Reasoning
- History: Statistical relation learning
- Some examples of neural-symbolic integration
- Technical challenges for neural-symbolic integration
- A taxonomy of approaches

Motivating question

If we have symbolic rules about a problem (or a symbolic solver related to a problem), can they help a neural network make better predictions?

Symbolic versus distributed representations

Where are the symbols stored in a neural network?

More generally, how is knowledge stored in a neural network?

Answering this is important to tie components of the network to components of symbolic knowledge

Symbolic versus distributed representations

Where are the symbols stored in a neural network?

More generally, how is knowledge stored in a neural network?

Answering this is important to tie components of the network to components of symbolic knowledge

Related question: Do symbols emerge as part of training the network? Or are they already present by design?

Symbolic versus distributed representations

Where are the symbols stored in a neural network?

More generally, how is knowledge stored in a neural network?

Answering this is important to tie components of the network to components of symbolic knowledge

Related question: Do symbols emerge as part of training the network? Or are they already present by design?

These are questions about the *interfaces* between symbolic and neural (i.e. distributed) representations

Symbols by themselves don't have meaning

Consider $\forall x, y, \text{Father}(x, y) \rightarrow \text{Parent}(x, y)$

Suppose we rename the predicates Father and Parent to Pred13 and Pred81 respectively. Will this change any inferences we make about the objects in this universe?

Symbols by themselves don't have meaning

Consider $\forall x, y, \text{Father}(x, y) \rightarrow \text{Parent}(x, y)$

Suppose we rename the predicates Father and Parent to Pred13 and Pred81 respectively. Will this change any inferences we make about the objects in this universe?

From $\forall x, y, \text{Father}(x, y) \rightarrow \text{Parent}(x, y)$ and $\text{Father}(\text{Bob}, \text{Rich})$,
we can conclude that $\text{Parent}(\text{Bob}, \text{Rich})$

Symbols by themselves don't have meaning

Consider $\forall x, y, \text{Father}(x, y) \rightarrow \text{Parent}(x, y)$

Suppose we rename the predicates Father and Parent to Pred13 and Pred81 respectively. Will this change any inferences we make about the objects in this universe?

From $\forall x, y, \text{Father}(x, y) \rightarrow \text{Parent}(x, y)$ and $\text{Father}(\text{Bob}, \text{Rich})$,
we can conclude that $\text{Parent}(\text{Bob}, \text{Rich})$

From $\forall x, y, \text{Pred13}(x, y) \rightarrow \text{Pred81}(x, y)$ and $\text{Pred13}(\text{Bob}, \text{Rich})$,
we can conclude that $\text{Pred81}(\text{Bob}, \text{Rich})$.

Symbols by themselves don't have meaning

Consider $\forall x, y, \text{Father}(x, y) \rightarrow \text{Parent}(x, y)$

Suppose we rename the predicates Father and Parent to Pred13 and Pred81 respectively. Will this change any inferences we make about the objects in this universe?

From $\forall x, y, \text{Father}(x, y) \rightarrow \text{Parent}(x, y)$ and $\text{Father}(\text{Bob}, \text{Rich})$,
we can conclude that $\text{Parent}(\text{Bob}, \text{Rich})$

From $\forall x, y, \text{Pred13}(x, y) \rightarrow \text{Pred81}(x, y)$ and $\text{Pred13}(\text{Bob}, \text{Rich})$,
we can conclude that $\text{Pred81}(\text{Bob}, \text{Rich})$.

But do the names like Father, Parent, Pred13 and Pred81 matter in the computational process of the inference? Do the names Bob and Rich matter or can they also be renamed?

Symbols by themselves don't have meaning

Consider $\forall x, y, \text{Father}(x, y) \rightarrow \text{Parent}(x, y)$

Suppose we rename the predicates Father and Parent to Pred13 and Pred81 respectively. Will this change any inferences we make about the objects in this universe?

From $\forall x, y, \text{Father}(x, y) \rightarrow \text{Parent}(x, y)$ and $\text{Father}(\text{Bob}, \text{Rich})$,
we can conclude that $\text{Parent}(\text{Bob}, \text{Rich})$

From $\forall x, y, \text{Pred13}(x, y) \rightarrow \text{Pred81}(x, y)$ and $\text{Pred13}(\text{Bob}, \text{Rich})$,
we can conclude that $\text{Pred81}(\text{Bob}, \text{Rich})$.

But do the names like Father, Parent, Pred13 and Pred81 matter in the computational process of the inference? Do the names Bob and Rich matter or can they also be renamed?

Think about refactoring variable names in your code. That shouldn't change the underlying behavior of the code

Insides of neural networks by themselves don’t have meaning either

A distributed representation is just a vector

A node in a computation graph is just a value

Insides of neural networks by themselves don't have meaning either

A distributed representation is just a vector

A node in a computation graph is just a value

For both symbols and neural network components, they derive meaning from

1. their association with objects in the world and facts about them, and
2. What computational processes are performed on them

But the interfaces between neural networks and symbols matter

Key challenge: Can we map symbols (both predicates and objects) in a symbolic expression to elements in a neural network?

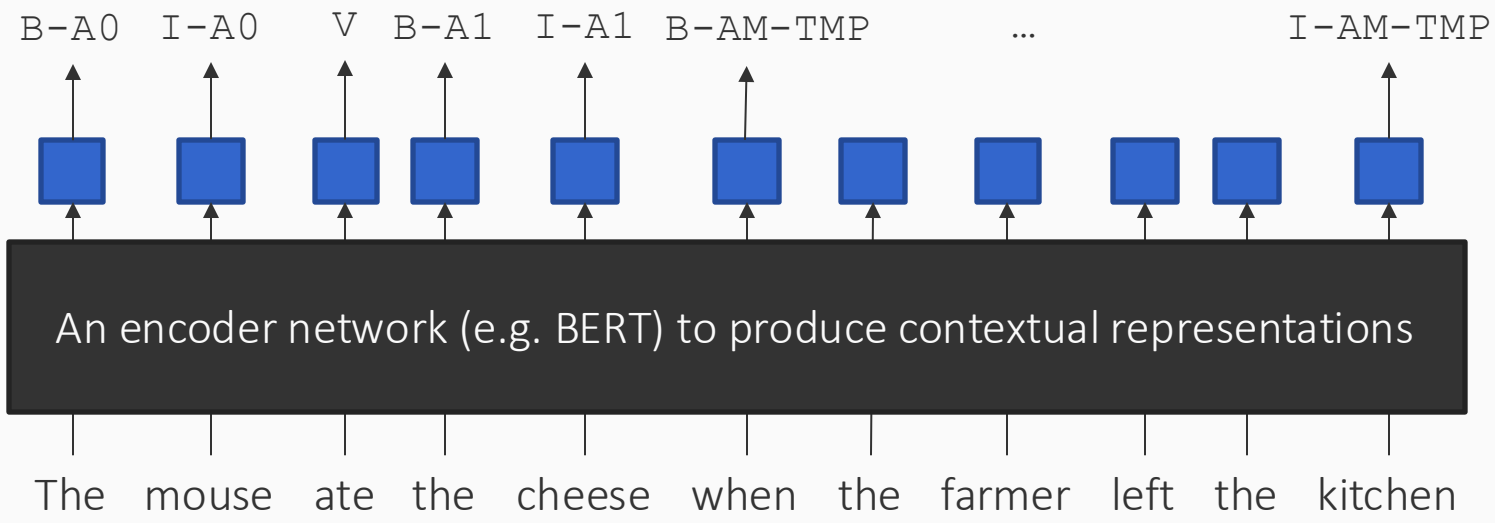
Claim: Every neural network exposes interfaces that have externally defined meaning.

Symbols in neural networks

All neural networks expose *interfaces* in the form of nodes that have externally defined meaning

Symbols in neural networks

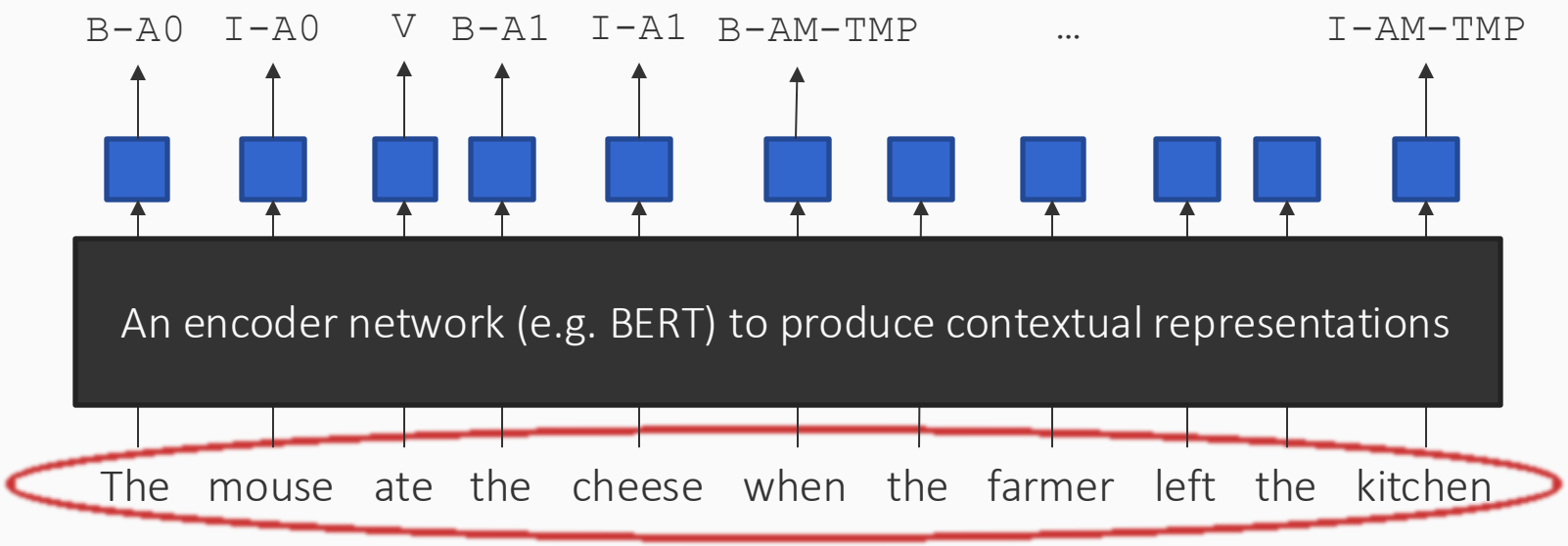
All neural networks expose *interfaces* in the form of nodes that have externally defined meaning



Consider this network to predict semantic roles for a verb

Symbols in neural networks

All neural networks expose *interfaces* in the form of nodes that have externally defined meaning

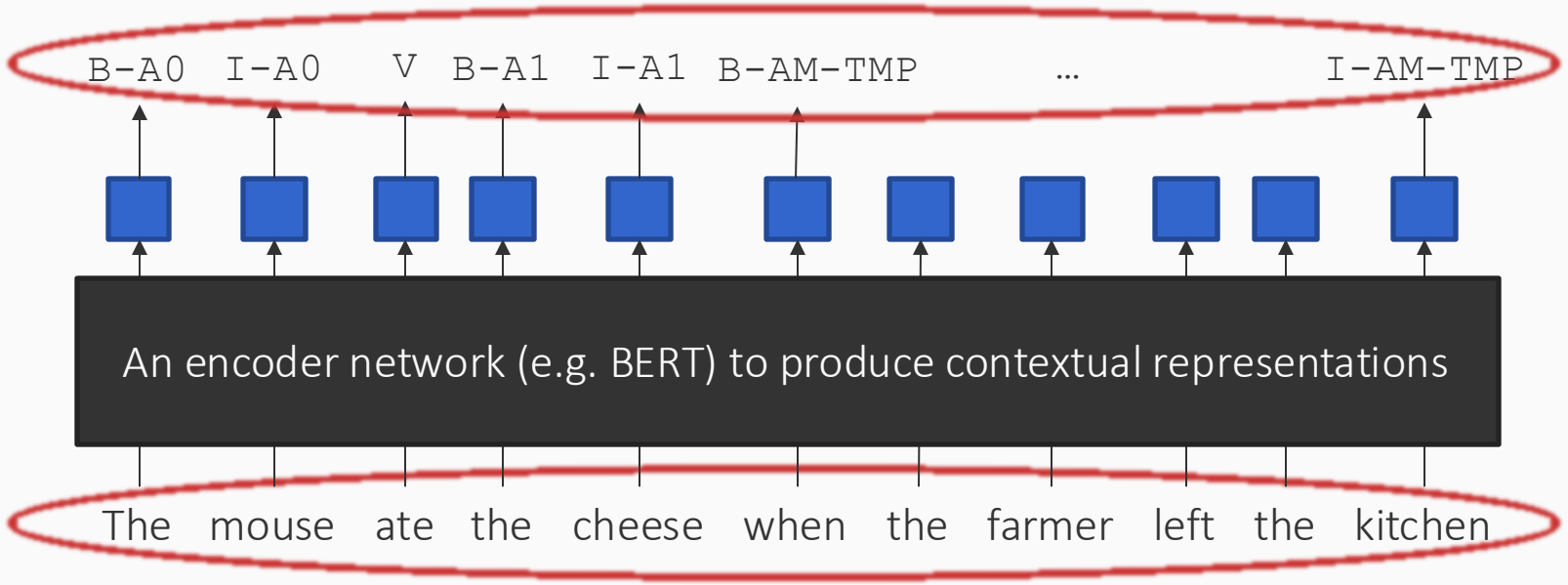


Consider this network to predict semantic roles for a verb

Inputs are objects that have meaning outside the network and can be mapped to symbolic expressions

Symbols in neural networks

All neural networks expose *interfaces* in the form of nodes that have externally defined meaning



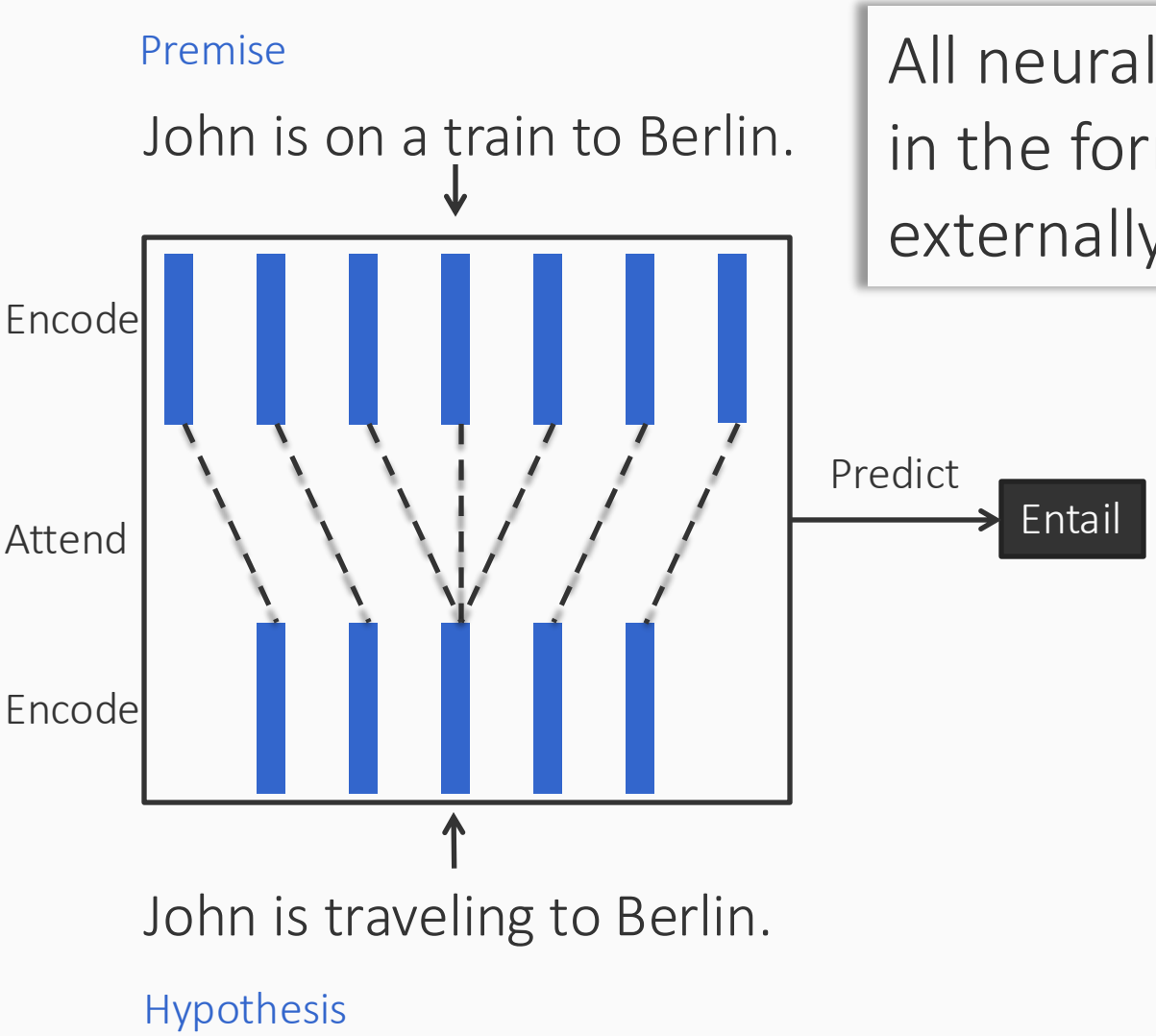
Consider this network to predict semantic roles for a verb

Outputs are predicates.

```
B-A0(ate, The)
I-A0(ate, mouse)
…
I-AM-TMP(ate, kitchen)
```

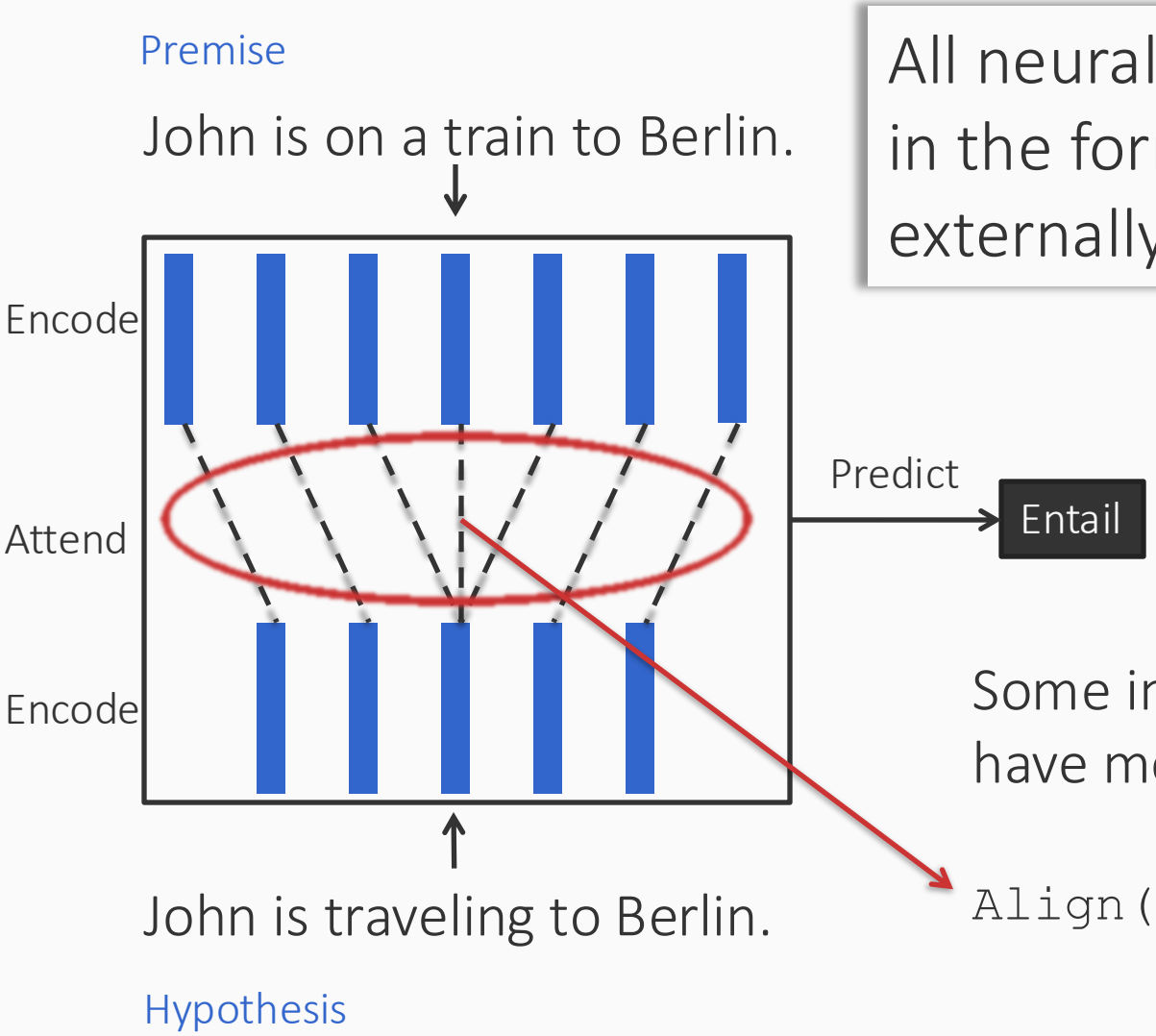
So are deterministic properties of inputs

Symbols in neural networks



All neural networks expose *interfaces* in the form of nodes that have externally defined meaning

Symbols in neural networks



All neural networks expose *interfaces* in the form of nodes that have externally defined meaning

Some internal nodes in the network have meaning by design

`Align(train, traveling)`

Named neurons

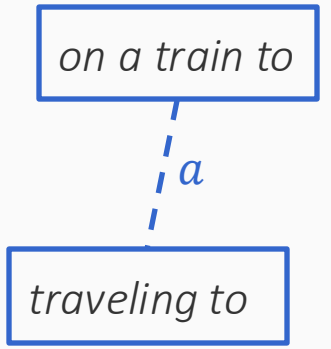
Nodes in a computation graph that have *externally* defined meaning

Named neurons give us the vocabulary for writing rules

Lower-case for neuron

Neuron a is associated with the predicate
Align(*on a train to, traveling to*).

Upper-case for predicate



Compositional statements

Logical statements, and programs that process them, are discrete, and have no concept of a derivative

But the rest of the neural network is differentiable

(Recall that gradient based learning is the most successful form of learning today)

Compositional statements

Logical statements, and programs that process them, are discrete, and have no concept of a derivative

But the rest of the neural network is differentiable

(Recall that gradient based learning is the most successful form of learning today)

Key technical issue: How do compositional concepts (e.g. logical statements) and neural network interface with each other?

Compositional statements

Logical statements, and programs that process them, are discrete, and have no concept of a derivative

But the rest of the neural network is differentiable

(Recall that gradient based learning is the most successful form of learning today)

Key technical issue: How do compositional concepts (e.g. logical statements) and neural network interface with each other?

There are different families of approaches to address this issue, which we will see next.