# Interactive Learning for Hierarchies of Concepts

Adwin Jahn, Ryan Maas, Harley Montgomery

## Background

- We are working with a crowdsourcing algorithm that builds hierarchies of concepts by asking users whether one entity is a subtype of another entity (i.e. "is orange a type of citrus fruit?")

- Hierarchies are represented by trees, where a node is considered a subclass of all of its ancestors.

- Questions must be simple since crowdsourcers don't know all possible entities, i.e. we can't ask about direct parent->child edges.

- Given the responses, a probabilistic model defines a posterior distribution over possible hierarchies and individual edges.

- At each iteration, the algorithm asks the question which is expected to maximize information gain.

## Prior Visualizations
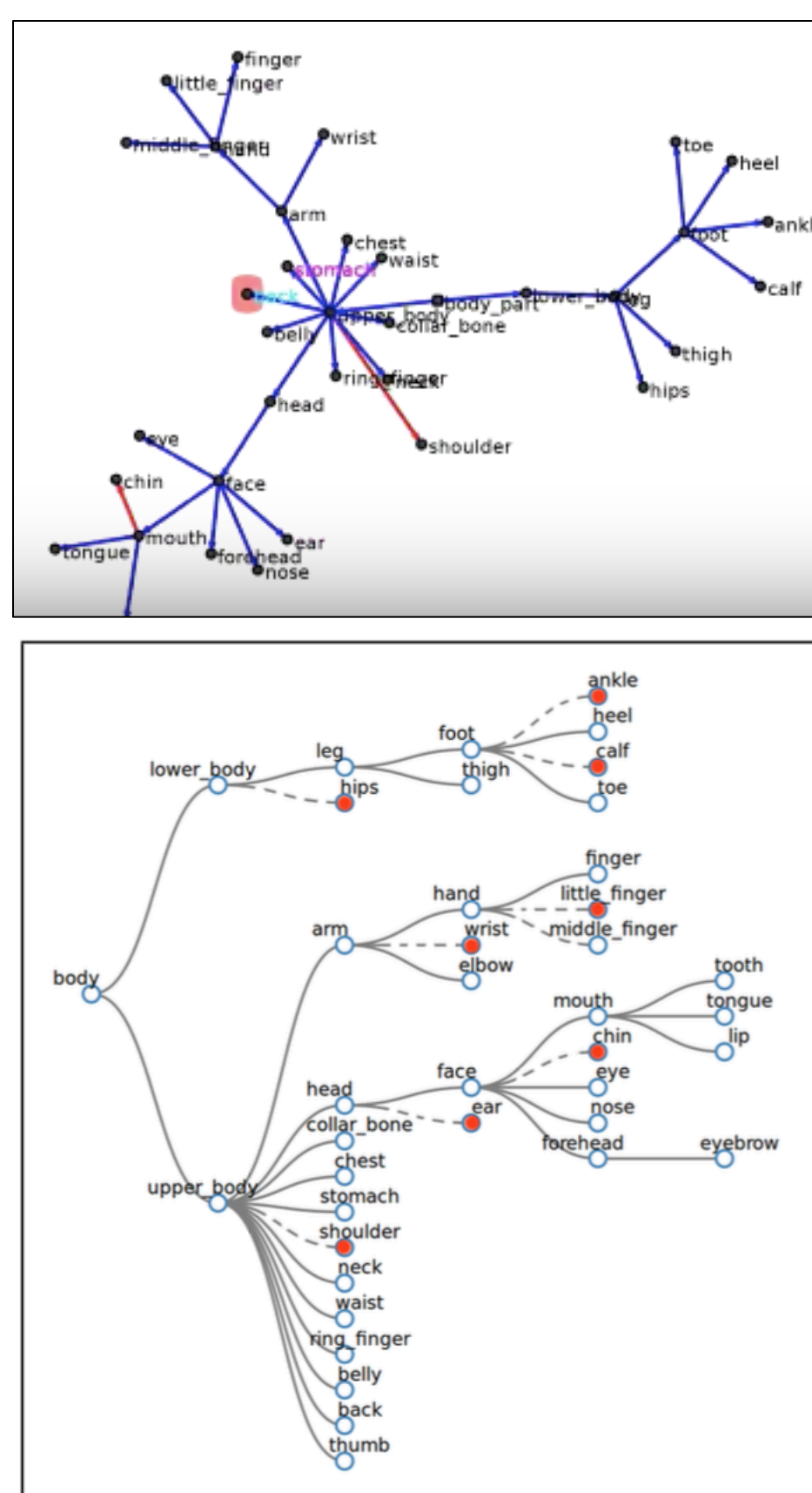




## Problem

- Algorithm returns the MAP tree at each iteration, but due to approximate inference and inherent uncertainty, this tree might not be representative of the entire distribution.

- One prior visualization tried to tackle this issue by highlighting nodes which whose parent edge has <75% marginal probability.

- However, it doesn't show other possible parent nodes, as this would be too cluttered in a static image.

- It can also be difficult to track individual changes to the tree during training.

- **How we can we enable the user to probe the uncertainty in the distribution over hierarchies, in order to better understand the algorithm and the resulting hierarchy?**

## Motivation

- Interacting with the distribution directly could allow the user to see the faults with the current MAP tree, and get a sense for the likelihood of other hierarchies.

- For experienced users, this could serve as a debugging tool, for when the MAP tree looks incorrect.

- For new users, this could serve as an educational tool, to understand how the distribution over hierarchies changes as the user responds to questions.

- Visualizing uncertainty can be challenging, but is often overlooked when learning distributions.

- Concepts from this project could be applied to other probabilistic models, in which uncertainty is often a factor.

## Approach

- We deal with the uncertainty in two main ways:

  ➢ Use color/width/opacity to triply edge strength, so that weaker links look more "unstable", while stronger ones look more "stable"

  ➢ "Edge highlighting": Hovering over a node shows all parent edges that are above the threshold, with "stable/unstable" edge encodings
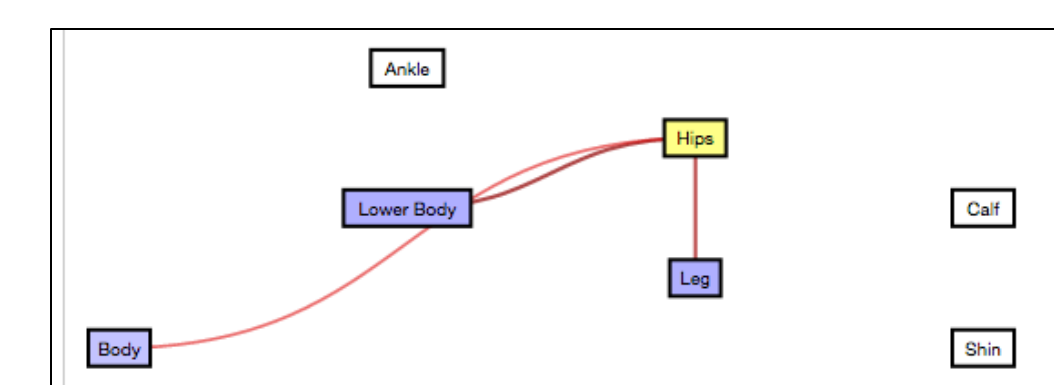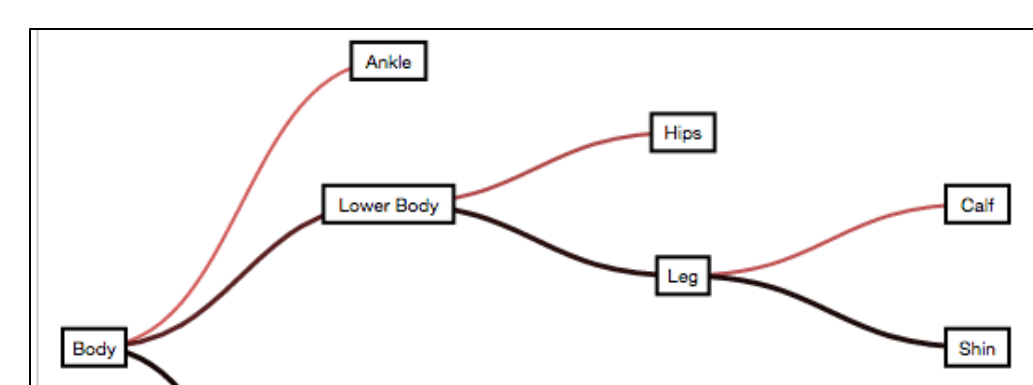


- We also replace the crowdsourcing portion with an interactive app, which allows the user to see how the tree is built during training.

- To help the user track changes, we highlight the nodes in question, and use smooth transitions after the question is answered.

## Results

- Below we show the MAP tree after 300 answers from Amazon Mechanical Turk users.

- Edge encodings make it clear which edges have high probability, and which do not.

- Then user can then probe nodes with unstable edges, giving a sense of other places the node could have been placed.
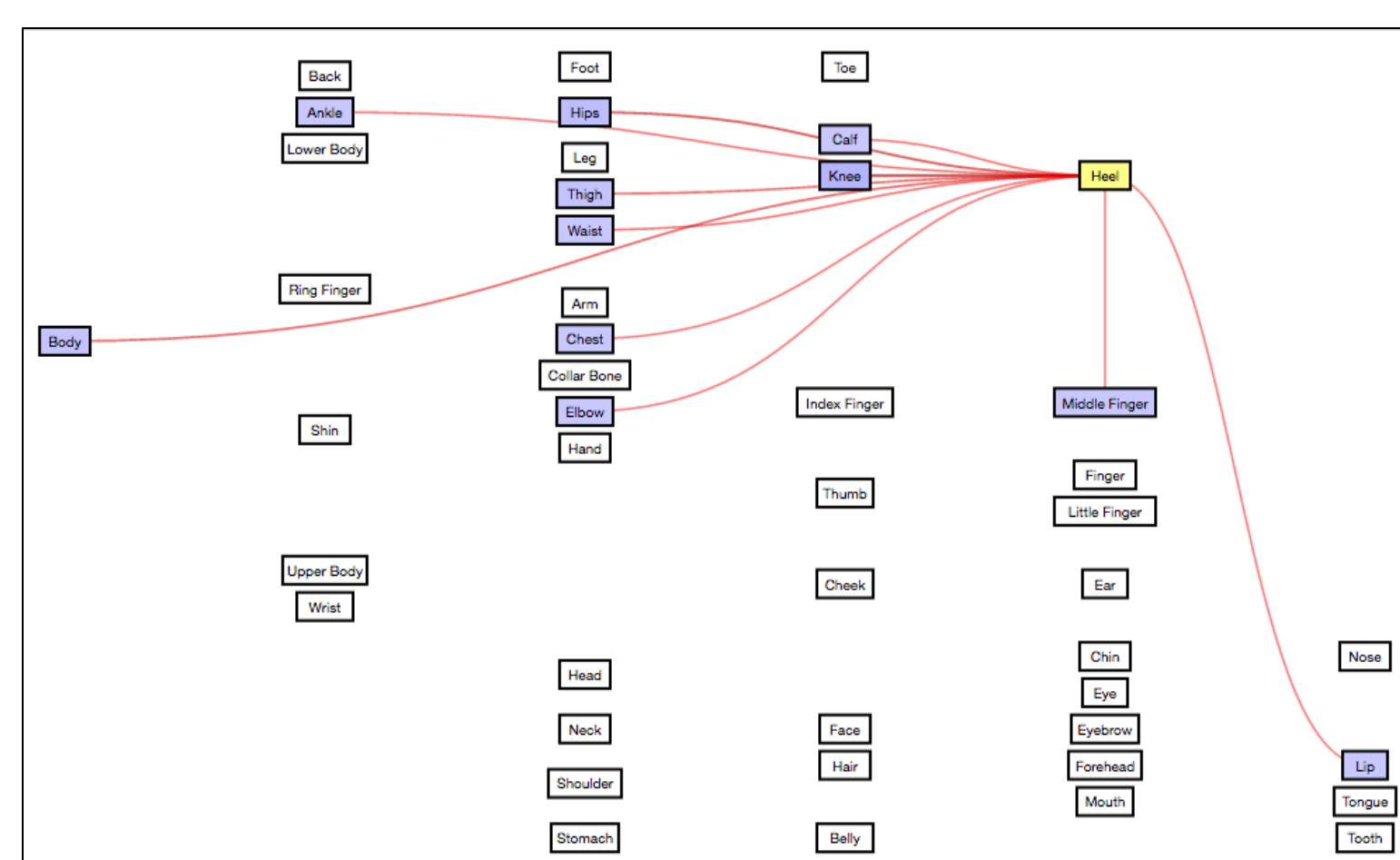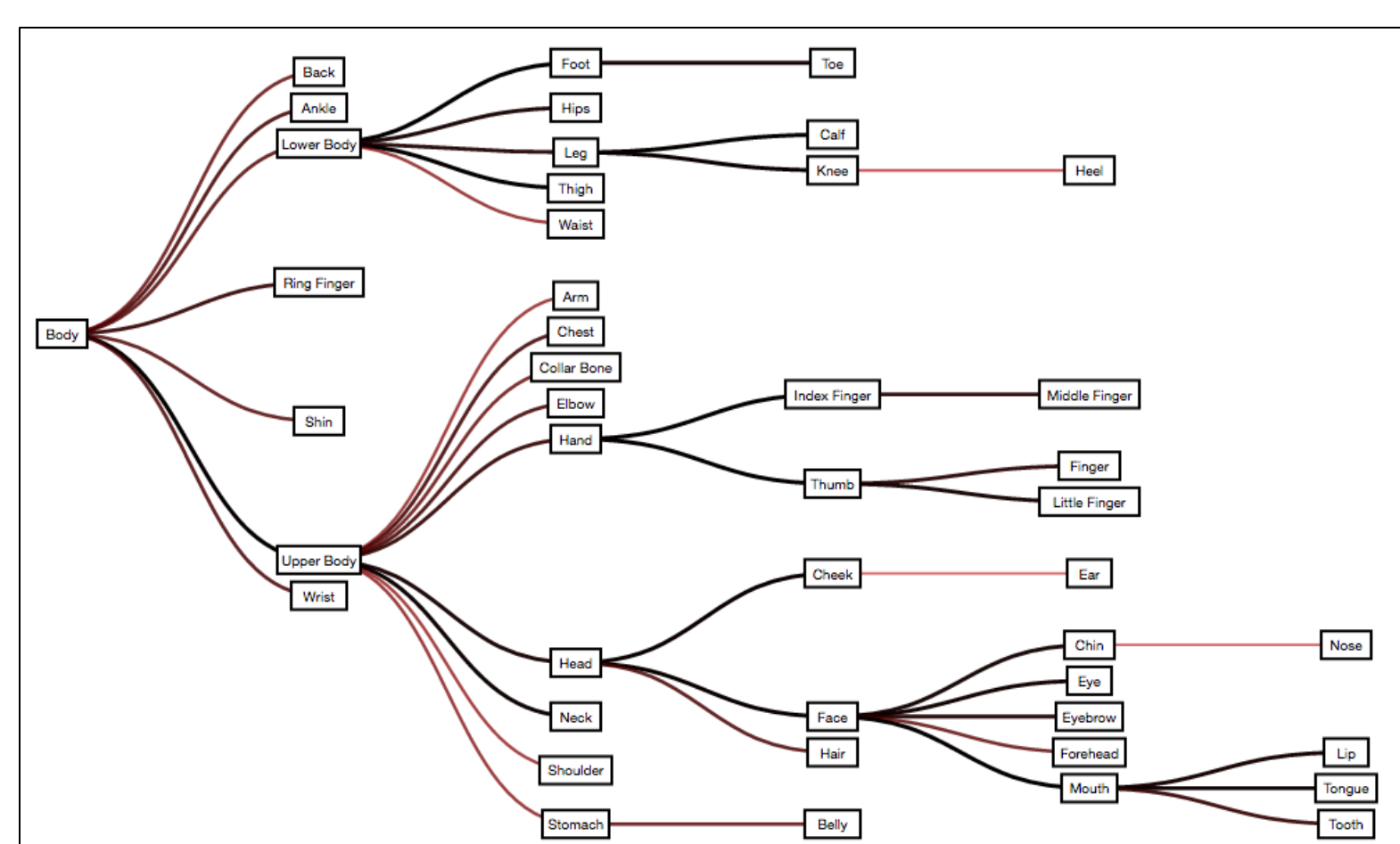
- We also built an interactive app which allows the user to step through the algorithm to understand it better.

- Users can see how the answer changes the relationship between nodes, and probe the trees generated during training.
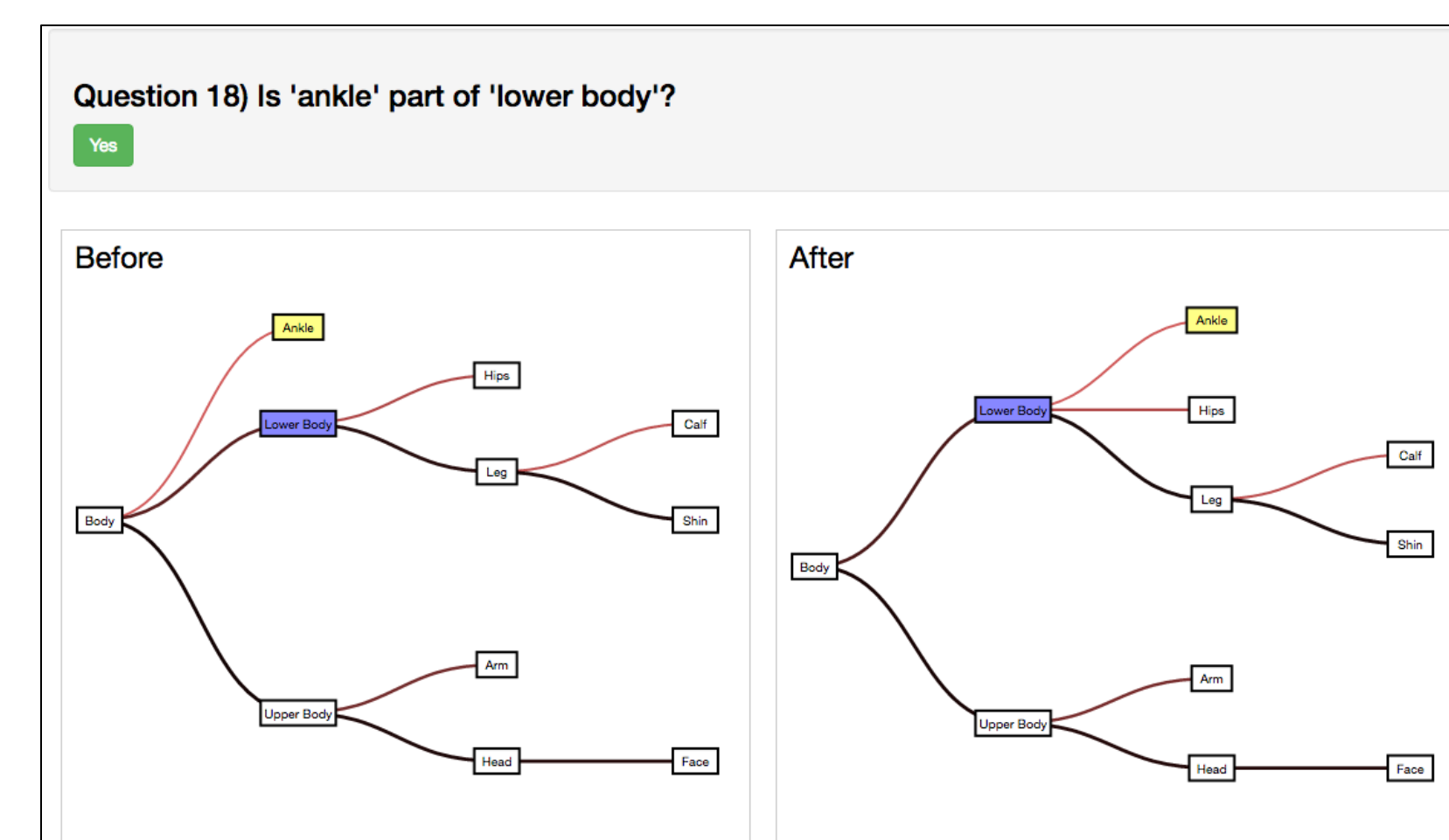






## Future Work

- Due to time constraints, our collaborator was unable to get us an interactive version of the algorithm, so the interactive app relies on AMT answers and just assumes that the user gave the same answer. This works well enough for our demo, but true interaction would improve the user experience.

- Our current work allows the user to see the uncertainty of the parent edge of a given entity, but it would be interesting to find efficient ways of visualizing the entire distribution at once (We made some attempts at edge highlighting for multiple nodes, but found that this was too cluttered).

- For example, given more access to the algorithm internals we could also sample from the posterior distribution, allowing the user to probe the distribution in a more direct manner.