Nick Liccini
CS4635 Section A – Project 3 Reflection
11.25.2019

Introduction

The objective of this project is to develop a human-like learning agent that understands the relationships between inputs so that it can classify new ones. There are many machine learning techniques that can be used (neural networks, decision trees), but this project will focus on an AI approach which begins with the fundamental question: *"How do humans learn?"*

Agent Architecture

The agent consists of four steps: **Induction**, **Learning**, **Reflection**, and **Classification**. This section describes the functionality of each step along with some connections to KBAI techniques and human cognition using the below example and nomenclature.

<"What are the learning goals for this class?", learninggoal>

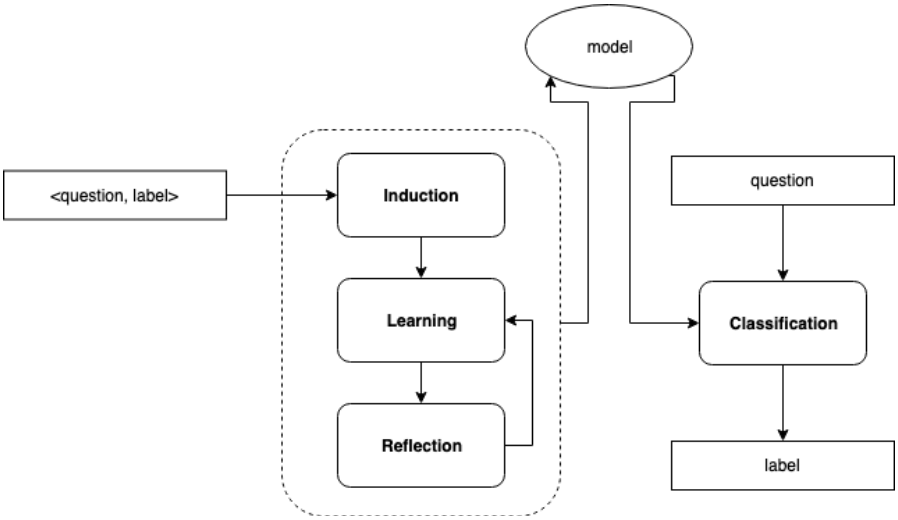

Figure 1. High level learning agent architecture.

- **Perception**: a natural language question.
- **Training input**: a perception and corresponding label.
- **Test input**: a perception to be classified.
- **Feature**: attributes of a perception that provide information.
- **Weight**: an importance value for a feature.
- **Rule**: a grouping for similar features and weights.
- **Concept**: the set of rules that describe a label.
- **Model**: the set of concepts that describe the agent's knowledge.

Figure 2. Nomenclature for this agent architecture.

Induction

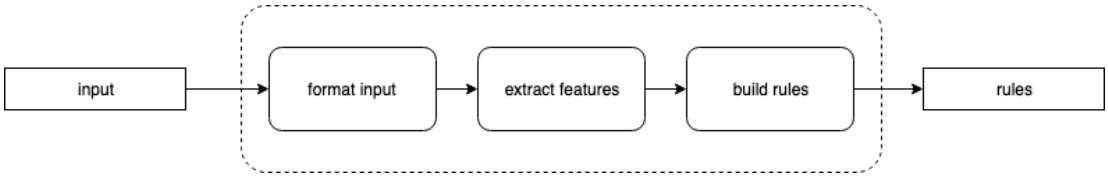

Figure 3. Induction process detailed.

When a new training input is received, it is formatted by the natural language parser where auxiliary features are stripped away from the input such as articles, pronouns, conjunctions, and numerals. Then each word in the stripped sentence is reduced to its base following grammar rules (e.g. 'teaches' reduced to 'teach', 'learning' to 'learn'). At this point, the input looks like the following:

['what', 'learn', 'goal', 'for', 'class']

These words represent useful features which get sorted based on their roles. Prepositions are sorted as "constraints" and all other words are sorted as "keywords." These two roles are the basis for the rules that will be returned by the induction step. All features are put into their corresponding rule with a count of how many times that feature appeared in the sentence.

This step represents how a human might read a training input and quickly pick out the most important characteristics. Many words in a sentence don't add to the meaning of the sentence, so humans implicitly "format" the sentence to pick out what's important. Then reducing all the words to their base ensures that variations of a word all maintain the same intent. A human might also implicitly categorize different words for different levels of importance that provide meaning, for example a preposition might not define the meaning, but might give some context to deconflict between two meanings. This step makes use of **Constraint Propagation** when applying grammar constraints to define useful features, **Diagnosis** when using a cause (sentence) and effect (label) to determine the rules governing that relationship, and **Understanding** when applying thematic roles to isolate different roles of features.

Learning

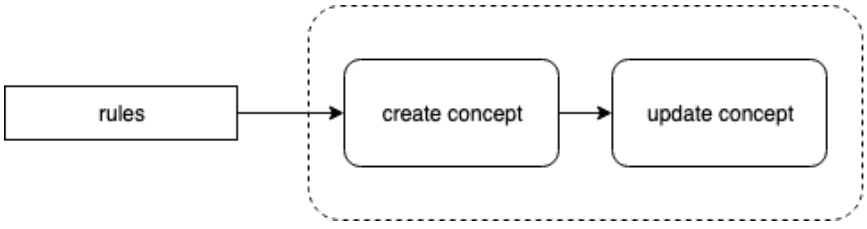

Figure 4. Learning process detailed.

Using the rules produced by the induction step, the agent can update its concept of this label. For each new label the agent encounters, it creates a concept which contains the rules that describe the label, a partial concept may look like Fig.5:

```
Concept
    label: 'learninggoal'
    rules['keyword']:
              ['what' : 2.0,
               'learn': 4.0,
               'goal' : 4.0,
               'know' : 1.0]
    rules['constraint']:
               ['for'  : 0.2,
               'in'    : 0.1]
```

Figure 5. Representation of a concept.

After creating a concept for this label, the agent specializes or generalizes its concept to fit the new rules. To specialize, the agent checks if the features in the new rules are also in its current concept; if so, it will increase the weight of those features, increasing the impact they have on this label. To generalize, the agent checks if the features in the new rules are unknown; if so, it appends them to the concept, increasing the possible features that describe this label.

This step represents how a human might internalize what the underlying meaning of a sentence is. A human might specialize a meaning by thinking that “learn” and “goal” together indicate that a sentence is referring to *learninggoal*, and a human might generalize by thinking that “know” is sometimes associated with questions regarding *learninggoal*. This step makes use of **Incremental Concept Learning** when generalizing and specializing features to create an updated version of the concept for each label.

Reflection

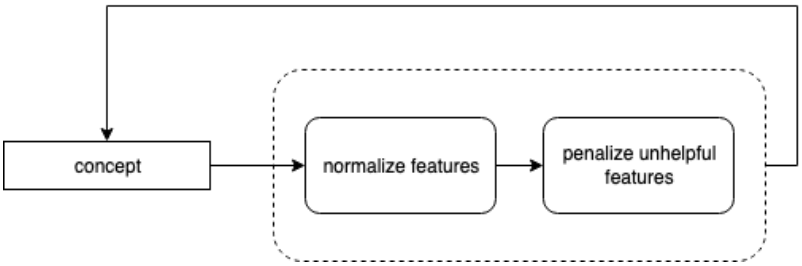

Figure 6. Reflection process detailed.

After all of the training inputs have been evaluated and the agent has built up various concepts based on features, the agent reflects on its model. The agent accounts for unevenly weighted and unhelpful features. Some concepts may have had repeated or overlapping features. To account for this, the agent considers all concepts and normalizes each feature's weight for each concept by the max weight of that feature across all concepts. The agent then examines features that are the same across multiple concepts and reduces the weight of them to ensure there is less bias unhelpful features. Consider the revised two concepts in Fig.7:

```
Concept
   label: 'learninggoal'
   rules['keyword']:
            ['what' :  1.0,
             'learn':  1.0 -(0.1),
             'goal' :  1.0,
             'know' :  1.0,
             'class':  0.8 -(0.08)]
   rules['constraint']:
              ['for'  :  0.2,
             'in'   :  0.1]
```

```
Concept
   label: 'learningstrategy'
   rules['keyword']:
            ['what' :  0.8,
             'learn':  0.8 -(0.08),
             'strategy' :  1.0,
             'know' :  1.0,
             'class':  1.0 -(0.1)]
   rules['constraint']:
              ['for'  :  0.2]
```

Figure 7. A revised concept.

This step represents how a human might review what they know to rationally remove biases and gain a holistic understanding of the world. Since concepts are interrelated, it is important for humans to reflect on how these relationships affect features and rules between concepts. This step makes use of **Metacognition** when reviewing each concept's interrelationships and revising how each one is affected by its rules, and **Learning by Correcting Mistakes** by adjusting the weights of unevenly weighted, overlapping, and unhelpful features.

Classification

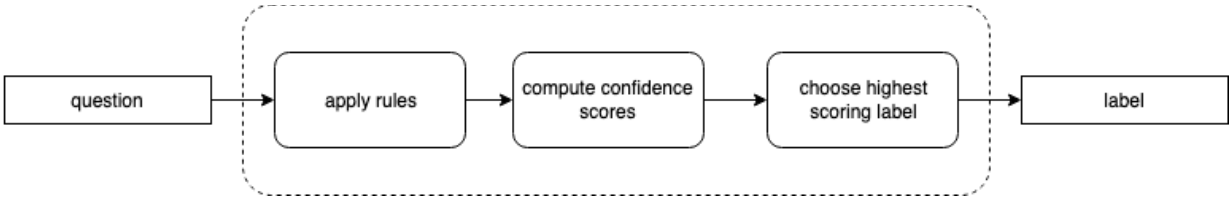

Figure 8. Classification process detailed.

Now the agent can apply the rules from each concept to score how well a test input matches a label. For each label, the rules within the concept are applied to the perception and the same feature extraction occurs. Next, the rules are applied and a score is computed by adding the weights corresponding to each feature. The label with the highest score is returned.

This step represents how a human interacts with the world using the knowledge they have learned. A human might try to match important features of the perception with the rules that they know and rank how well those match different concepts, then the best fitting label is chosen. This step uses **Classification** when applying a scoring function to an input to choose the best label, and **Production Systems** when applying episodic, procedural, and semantic knowledge to build a model that applies rules, experience, and concepts to identify a label.

Use of Knowledge-Based AI Techniques

KBAI techniques are extensively used in this system as described above and they were very important considerations when designing the agent. They provide an outline for the

subprocesses involved in a learning agent where the production system model below provides a high-level outline of its inspiration.

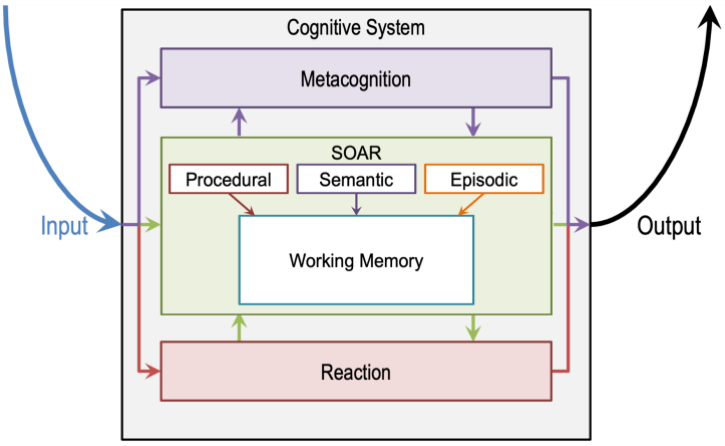

Figure 9. Production system architecture using the Soar model (Goel, 2019).

Most of the agent was designed to use KBAI techniques but the use of feature extraction and weighting don't align with any particular technique, rather these aspects of design were used to mimic less formal human processes such as bias.

Since humans are more complex than the agent, it is unable to completely mimic human thought processes for learning and classification. Regarding learning, the agent is unable to extract more abstract features such as relationships between words within a sentence and thematic roles of verbs and nouns. The agent is also unable to properly deconflict between similar concepts using features alone, which could be handled by a more powerful reflection step. Regarding classification, the agent must compute a score for each label, whereas a human might be able to filter out irrelevant labels almost instantly and only consider labels that could be candidates. Thus, for large data sets, this agent might perform slowly when a human might perform just as fast as for small data sets.

Human Cognition

The agent architecture presented above is designed to mimic the mental process of a human, but some key differences remain. For me, the reflection step occurs simultaneously to the learning step where I gain new knowledge from an input and update all of my concepts right after. At the end of training, another reflection step occurs for me where I take the concepts and use the relationships between features to form a final model as opposed to just using features. This is because the human mind deeply learns by understanding relationships between features and a simple computational agent is unable to understand how words (represented only as strings) are related. Because of this, the agent was limited to using scores for words and normalizing them in an attempt to mimic the reflection step, but these simple heuristics do not capture the deep interrelationships between features and concepts. In an attempt to simulate bias and preferences, the scores were weighted based on repetition, but the agent cannot accurately represent bias using such a simple method.

Design Review

A robust agent could learn the main distinguishing features between any set of labels and could build a general model that fits most cases where the features and their relationships are present. A brittle agent would be too limited on specific features such as specific words in the sentence, and the model would be based too much on values rather than relationships. This agent falls in the middle of that spectrum since it is able to produce a concept with weighted features representing importance and it is able to recognize features that provide unhelpful information. However, it is unable to acknowledge the relationships between features and concepts such as positioning of words in a sentence, synonyms, verbs versus nouns, or relevance to one category versus another.

Because of these reasons, the agent is able to answer questions that use distinct features and don't have features that fall under multiple categories. This indicates that with an infinitely scaled vocabulary, the agent's performance would decrease since multiple words could be used to describe the same categories and the agent would be unable to differentiate between labels because every concept could converge to the same set of rules.

This agent mimics human-level intelligence to some degree by learning, creating rules from feature extraction, and reflecting to holistically update its concepts, but it is not close to human intelligence without the ability to identify deep relationships between features and concepts.