“The Meaning of Meaning”

An overview of the SHRDLU System

Artificial Intelligence: Chp. 6
Patrick Winston

CS 395T Spring 2019
1/30/2019
Neha Srikanth
Problem Statement/Motivation
Problem Statement

- How can we make computers **understand** natural language?
  - This is **HARD**!
  - How can we tell whether a computer has really understood something?

- Can we build a system that actively interprets language to interact with an artificial environment?

- **Limiting domain of discourse**
- **Tackle a specific task**
Blocks World: Limited Domain

(a) "Pick up a big red block."

(b) "Find a block which is taller than the one you are holding and put it into the box."

(c) "Will you please stack up both of the red blocks and either a green cube or a pyramid?"

"PICK UP A BIG RED BLOCK."

"OK."

Photo from hci.stanford.edu, biocote.com, www.wiley.com
Interaction with Blocks World

● Useful because number of things to ask about is very limited
  ○ Objects in the world + how they’re related

● Need to handle two types of sentences
  ○ State of the world (*Is there a red pyramid above the green block?*)
  ○ Change the state of the world (*Put the red pyramid behind the it.*)
Methodology
Understanding a Block World Command

1. Divide the sentence into **word groups** (noun and verb groups)
2. Translate the noun group into a **search program** that generates a unique object name
3. Translate the verb group to an **action program**
4. Object name + action program = **function/argument pair** to execute

“Pick up a red object which supports a pyramid”
Breaking Down Natural Language Utterances

**Syntactic Specialist**
- **Parse** sentence
- Break into **word groups** (noun and verb groups)

**Sentence Specialist**
- Determine **relations** between objects and actions in different word groups **within** a sentence

**Scenario Specialist**
- Determine **relations** between sentences
- **Resolve** context references
Word Groups

- **Noun Groups (Noun + modifiers)**
  - Use them to talk about specific objects in the world
  - Need to resolve them to specific object name
    $$\text{“Big red pyramid” } \rightarrow \text{Object Identifier}$$
  - Almost all the information that the system knows about an object is accessed through the object identifier

- **Verb Groups (Verb + arguments + adverbs)**
  - Use them to talk about actions/interactions with the objects
  - Simpler than noun groups to resolve $$\rightarrow$$ clear action
    $$\text{“Pick up”}$$
Noun Group Resolution

- SHRDLU ("Robbie") can figure out object identifiers by using natural language noun groups to write search programs to execute and retrieve names.
- Analyze the structure of noun groups:
  - Constraint grammar for natural language
    - Augmented Transition Network (ATN) Grammars
    - Traversing network = valid noun group

Basic Noun Group Network

```
Basic Noun Group Network
```

```
S1
```

```
S2
```

```
S3
```

```
“the big red pyramid”
```

```
adj
```

```
det
```

```
noun
```

Success state
Transition Networks

- Capture syntax structure $\rightarrow$ a valid noun group is a **set of instructions for moving through** the transition network
- Effective for representing various noun groups (prepositional groups modifying noun group, recursive noun groups, etc.)
  - “A red pyramid on the big block near the empty box”
- Traversing these nets allows us to **use** the constraints and facts that are captured effectively $\rightarrow$ we are free to write/read memory as we move through the network
When we successfully traverse a transition network, we figure out **word and group features** (helping us determine the words’ grammatical role, etc)

- Type, singular vs. plural (nouns), present vs. past (verbs), etc

- “Augmented” because they allow us to read from/write to memory (indicating actions) as we discover word and group features

- Ex: Group Definite vs. Indefinite when we hit a determiner
Putting it All Together

- Mechanisms for translating noun groups into **programs**
- **Group’s features** help shape its corresponding program
- SHRDLU Dictionary
  - Contains **program fragments** to construct search programs that return object identifiers

Noun Group

\[
\text{A red object which supports 3 pyramids}
\]

Program

\[
\text{FIND > 0 X SUCH THAT}
\text{ ?X HAS-COLOR RED}
\text{ THERE ARE >2 Y S.T}
\text{ ?X SUPPORTS ?Y}
\text{ ?Y IS A PYRAMID}
\]
Putting it All Together

- Now able to translate natural language descriptions to unique internal object identifiers!
  - makes commands and world-state questions easier!
      - Can be resolved into noun group programs
    - Commands: Providing object identifiers to verb group programs
- Combine the object names with action programs → function/argument pair executed by block manipulation module
Conclusion + Reflection

- Limiting domain allows us to begin tackling complex problems
- Allowed for the development of a system that can interact with a specific type of environment
  - But this rule-based system requires program reformulation for a new domain
  - Only some types of questions make sense within the domain
- Employed classical NLP techniques to parse commands and questions (hard-coding grammar, looking at parse trees), but nowadays DL systems can use lots of training data to figure out the grammar, etc.