Natural Language Processing

- Attempt to process language like humans
- Natural language processing: receives and analyzes
- Natural language generating: performs function on given text

NLP levels of language

- **Phonology**: audible sounds
- **Morphology**: prefixes, root, suffixes etc.
- **Lexical**: lexicon, arguments and classes
- **Syntactic**: grammar, parsing
- **Semantic**: use of sentence to solve disambiguation
- **Discourse**: sentences as functions
- **Pragmatic**: outside of text, real world knowledge

NLP approaches

- **Statistical**: probability, collections of language data and statistical data
- **Symbolic**: rules and logic of language, facts, rule-base, semantic networks, decision trees
- **Connectionist**: statistical models + representation
- all approaches are viable

Uses for NLP

- Information retrieval
- Information extraction
- Summarization
- Dialogue systems
Parallel cost efficiency

- Parallel systems don’t always achieve linear speedup
- Costup won’t likely reach linear (parallelizing a job rarely requires \( m \times p \))

Parallel cost efficiency

\[
\begin{align*}
\text{speedup}(p) &= \frac{\text{time}(1)}{\text{time}(p)} \\
\text{costup}(p) &= \frac{\text{cost}(p)}{\text{cost}(1)} \\
\text{cost performance} &= \frac{\text{cost}(p)}{1 + \text{time}(p)} \\
\text{speedup}(p, m, m') &= \frac{\text{time}(1, m)}{\text{time}(p, m')} \\
\text{costup}(p, m, m') &= \frac{\text{cost}(p, m, m')}{\text{cost}(1, m)}
\end{align*}
\]

Parallel cost efficiency

\[
\begin{align*}
\text{cost}(1, m) &= f(1) + g(m), \\
\text{cost}(p, m') &= f(p) + g(m') \\
\text{costup}(p, m, m') &= \frac{f(p) + g(m')}{1 + g(m)} \\
\text{costup}(p, m, m') &= \frac{f(p) + 1}{1 + 1} = \text{costup}(p, m, m') = \frac{f(p) + 1}{2}
\end{align*}
\]

Poetry Wizard

Rule-based translation

- Uses existing language rules
- Tree structures break down sentences
- Logic forms
  - Example: \(< \text{PAST HAPPY}> \ (\text{NAME} \ j1 \ "Joe")\)
Anaphora

- Anaphora resolution couples anaphor and antecedant
- Example: “Joe is not yet here but he is expected to arrive in the next one hour.”
- Requires knowledgebase

Parallel decision trees

- Can use numerous lightweight posix threads
- Thread scheduler to control threads
- Two levels of parallelization
  - Parallelize building of tree
  - Quicksort data into nodes
- High level for scheduler

Conclusion

- Parallel systems more efficient with large memory – NLP is good candidate
- NLP requires processing large amounts of information, various functions, need to think about speed