Machine Translation

- Introduction
- Why is MT hard?
- Approaches to MT
  - Direct Translation
  - Syntactic Transfer
  - Interlingua
- Parallel Texts
- Statistical Machine Translation
- Computer Aided Translation

Introduction

- Goal: Automate of some or all of the task of translation.
  - Fully-Automated Translation
  - Computer Aided Translation
- What is "translation"?
  - Transformation of utterances from one language to another that preserves "meaning".
- What is "meaning"?
  - Depends on how we intend to use the text.

Machine Translation Uses

- Fully automated translation
  - Informal translation
    - babelfish
    - e-mail
  - Translating technical writing
    - Manuals
    - Proceedings
  - Translating literary writing
- Computer aided translation
  - Deciding what to translate "properly"

Why is MT hard?

- Languages differ from each other in many ways.
  - Lexical Differences
  - Syntactic Differences
  - Semantic Differences
  - Pragmatic Differences
- Ambiguity in the source language
  - Need to resolve ambiguity before we can translate
Why is MT hard: Lexical Difficulties

• One word can have multiple translations
  - e.g., "know" in English: "savoir" or "connaitre" in French

• Complex word overlap

• Lexical gap: word with no (simple) translation

• Idioms

Why is MT hard: Syntactic Difficulties

• Different languages use different syntactic structures.
  - SVO vs SOV vs VSO
  - Free word order languages

• To translate, we need to find the correct syntactic structure:
  - Resolve ambiguities

• Some syntactic forms are not possible in some languages
  - Center embedding

Why is MT hard: Semantic and Pragmatic Difficulties

• Literal translation does not produce fluent speech:
  - Ich esse gern: I eat readily.
  - La botella entro a la cueva flotando: The bottle entered the cave floating.

• Literal translation does not preserve semantic information
  - e.g., "I am full" translates to "I am pregnant" in French.

• Literal translation does not preserve pragmatic information
  - e.g., focus, sarcasm

Approaches to MT

• Machine translation makes use many NLP technologies.
  - Word sense disambiguation
  - Tagging
  - Parsing
  - Collocations
  - Document classification
Approaches to MT

- Interlingua (knowledge representation)
  - Knowledge-based Transfer
  - Semantic Transfer
- English (semantic representation)
  - Syntactic Transfer
- Direct Translation
- French (semantic representation)
  - Syntactic Transfer
- French (syntactic parse)
  - Direct Translation
- English (word string)
  - French (word string)

Direct Translation

- Series of processing stages
  - Each focused on a single problem (e.g., morphological analysis)
- Stages manipulate strings of tokens
  - No parsing or syntactic structures.
- Each stage performs a uni-directional transformation on the input.

Direct Translation: Example

- Input
  - "watashihatsukuenouenopenwojonniageta"
- Morphological Analysis
  - "watashi ha tsuke no ue no pen wo jon ni ageru PAST"
- Lexical transfer of content words
  - "I ha desk no ue no pen wo John ni give PAST."
- Preposition re-arrangement
  - "I ha pen on desk wo John to give PAST"
- SVO rearrangements & determiners
  - "I give PAST the pen on the desk to John"
- Morphological Generation
  - "I gave the pen on the desk to John"

Syntactic Transfer

- Parse
- Transform
- Generate

Three steps:
- Parse the source text.
- Transform the source language syntax tree into the target language.
- Use the target language syntax tree to generate a sentence.
Syntactic Transfer

- Define transformational rules on syntax trees

\[
S \quad NP \quad VP \quad S \quad VP \quad NP
\]

- Context-free rules
- Context-sensitive rules

- Apply rules to the source language syntax tree.
  - Top-down or bottom-up

Interlingua

- Two steps:
  - Translate source text into a universal knowledge representation.
  - Use the knowledge representation to generate a target text.

- Advantages:
  - For \( n \) languages, we need \( n \) components (not \( n^2 \))
  - Other programs can use the interlingua

Interlingua: Difficulties

- Universal lexicon
  - How do we construct a universal lexicon?
  - Must include all distinctions made by any language.
  - How to differentiate similar terms?
    - e.g., "shake" vs "vibrate"

- Universal knowledge format
  - How do we encode "knowledge"
  - What to include? (e.g., pragmatic information?)

- Unnecessary disambiguation
- Preserving ambiguity

Robustness Issues

- Machine translation should usually be robust
  - Always produce a sensible output

- Ways to achieve robustness:
  - Use robust components (robust parsers, etc.)
  - Use fallback mechanisms (e.g., to word-for-word translation)
  - Use statistical techniques to find the translation that is most likely to be correct.
Text Alignment

- Statistical techniques need training data.
- *Parallel texts (or bitexts):* one text in multiple languages.
  - Produced by human translation
  - Readily available
- The alignment problem:
  - Which sentences in one language correspond with which sentences in another?
  - One-to-one alignment doesn't work: translators don't translate each sentence separately.

Text Alignment

- Types of alignment
  - "n:m" → n sentences are translated into m sentences.
  - Common types of alignment
    - 1:1 (90%), 1:2, 2:1, 1:3, 3:1
- Algorithms:
  - Dictionary-based methods
  - Length-based methods
  - Arrival vectors
  - Lexical algorithms

Statistical MT

- Noisy Channel Model
  - Assume that we started with an English sentence.
  - The sentence was then translated to French.
  - We want to translate it back.
- Use bayes rule:
  \[
  \text{arg max}_e P(e|f) = \text{arg max}_e \frac{P(e)P(f|e)}{P(f)}
  \]

Statistical MT (Continued)

- Two components:
  - \(P(e):\) Language Model
  - \(P(f|e):\) Translation Model
- Task:
  - \(P(f|e):\) translates words
  - \(P(e):\) helps puts them in the correct order
- Estimate \(P(f|e)\) using a parallel corpus.
**Problems with Statistical ML**

- No notion of syntactic phrases
  - Words often get scrambled
- Difficulties with idioms
- Non-local dependencies
  - N-gram models cannot encode non-local dependencies.
  - Transform sentences to remove non-local dependencies (e.g., un-do movement)
- Sparse data problems

**Computer Assisted Translation**

- Machine translation performs tedious work for human translators.
  - Provide correct translation for "easy" sentences
  - Provide noisy translation for "difficult" sentences
- Post-editing: human cleans up the output of the machine translator.
  - Often required for human translation, as well.

**CAT (continued)**

We can make the translation task easier:

- **Sublanguages:**
  - If we can identify the genre of the text precisely, MT can use more specialized algorithms.
- **Pre-editing:**
  - Edit source text to use constrained vocabulary and constrained syntactic forms.
- **Interactive Systems:**
  - The computer can ask a human to help it make better choices.
  - Translation Memory