Language and Computers

Machine Translation

Based on Dickinson, Brew, & Meurers (2013)
Indiana University
Fall 2013

What is Machine Translation?

Translation is the process of:

- moving texts from one (human) language (source language) to another (target language),
- in a way that preserves meaning.

Machine translation (MT) automates (part of) the process:

- Fully automatic translation
- Computer-aided (human) translation

Is MT needed?

- Translation is of immediate importance for multilingual countries (Canada, India, Switzerland, . . . ), international institutions (United Nations, International Monetary Fund, World Trade Organization, . . . ), multinational or exporting companies.
- The European Union has 23 official languages. All federal laws and other documents have to be translated into all languages.

Example translations

The simple case

- It will help to look at a few examples of real translation before talking about how a machine does it.
- Take the simple Spanish sentence and its English translation below:

  (1) Yo hablo español.
  I speak Spanish.

  Words in this example pretty much translate one-for-one
  But we have to make sure hablo matches with Yo, i.e., that the subject agrees with the form of the verb.

Example translations

A slightly more complex case

The order and number of words can differ:

(2) a. Tu hablas español?
    You speak Spanish
    ‘Do you speak Spanish?’

   b. Hablas español?
    Speak español
    ‘Do you speak Spanish?’

Example translations

The order and number of words can differ:

(2) a. Tu hablas español?
    You speak Spanish
    ‘Do you speak Spanish?’

   b. Hablas español?
    Speak español
    ‘Do you speak Spanish?’

Example translations

The order and number of words can differ:

(2) a. Tu hablas español?
    You speak Spanish
    ‘Do you speak Spanish?’

   b. Hablas español?
    Speak español
    ‘Do you speak Spanish?’
What goes into a translation

Some things to note about these examples and thus what we might need to know to translate:
- Words have to be translated → dictionaries
- Words are grouped into meaningful units → syntax
- Word order can differ from language to language
- The forms of words within a sentence are systematic, e.g., verbs have to be conjugated, etc.

Dictionaries

An MT dictionary differs from a “paper” dictionary:
- must be computer usable (electronic form, indexed)
- needs to be able to handle various word inflections
- can contain (syntactic and semantic) restrictions that a word places on other words
  - e.g., subcategorization information: give needs a giver, a person given to, and an object that is given
  - e.g., selectional restrictions: if X eats, X must be animate
- contains frequency information
  - for SMT, may be the only piece of additional information

Steps in a transfer system

1. source language grammar analyzes the input and puts it into an underlying representation (UR).
   *Londres plait à Sam* → *Londres plaire Sam* (source UR)
2. The transfer component relates this source language UR (French UR) to a target language UR (English UR).
   *Londres plaire Sam* (source UR) → *Sam like London* (target UR)
3. target language grammar translates the target language UR into an actual target language sentence.
   *Sam like London* → *Sam likes London*

Different approaches to MT

We’ll look at some basic approaches to MT:
- Systems based on linguistic knowledge (Rule-Based MT (RBMT))
  - Direct transfer systems
  - Machine learning approaches, i.e., statistical machine translation (SMT)
  - SMT is the most popular form of MT right now

Direct transfer systems

A direct transfer systems consists of:
- A source language grammar
- A target language grammar
- Rules relating source language underlying representation (UR) to target language UR
  - A direct transfer system has a transfer component which relates a source language representation with a target language representation.
  - This can also be called a comparative grammar.

We’ll walk through the following French to English example:

(3) Londres plait à Sam.
London is pleasing to Sam
*‘Sam likes London.’*

Notes on transfer systems

- The transfer mechanism is in theory reversible; e.g., the *plaire* rule works in both directions
  - Not clear if this is desirable: e.g., Dutch *aanvangen* should be translated into English as begin, but begin should be translated as beginnen.
  - Because we have a separate target language grammar, we are able to ensure that the rules of English apply; *like* → *likes.*
- RBMT systems are still in use today, especially for more exotic language pairs
There are differing levels of abstraction at which transfer can take place. So far we have looked at URs that represent only word information.

We can do a full syntactic analysis, which helps us to know how the words in a sentence relate.

Or we can do only a partial syntactic analysis, such as representing the dependencies between words.

This method works best for structurally-similar languages. Ideally, we could use an interlingua = a language-independent representation of meaning.

Benefit: To add new languages to your MT system, you merely have to provide mapping rules between your language and the interlingua, and then you can translate into any other language in your system.

Ideally, we could use an interlingua = a language-independent representation of meaning.

Benefit: To add new languages to your MT system, you merely have to provide mapping rules between your language and the interlingua, and then you can translate into any other language in your system.

Interlingual problems

What exactly should be represented in the interlingua?

- e.g., English corner = Spanish rincón = ‘inside corner’ or esquina = ‘outside corner’

A fine-grained interlingua can require extra (unnecessary) work:

- e.g., Japanese distinguishes older brother from younger brother, so we have to disambiguate English brother to put it into the interlingua.

Then, if we translate into French, we have to ignore the disambiguation and simply translate it as frère, which simply means ‘brother’.
**Machine learning**

Instead of trying to tell the MT system how we’re going to translate, we might try a **machine learning** approach.

- We can look at how often a source language word is translated as a target language word, i.e., the **frequency** of a given translation, and choose the most frequent translation.
- But how can we tell what a word is being translated as? There are two different cases:
  - We are told what each word is translated as: **text alignment**
  - We are not told what each word is translated as: use a **bag of words**

We can also attempt to learn alignments, as a part of the process, as we will see.

**Word alignment**

- **word alignment** = determine which source language words align with which target language ones
  - Much harder than sentence alignment to do automatically.
  - But if it has already been done for us, it gives us good information about a word’s translation equivalent.

**Sentence alignment**

- **sentence alignment** = determine which source language sentences align with which target language ones (what we assumed in the bag of words example).
- Intuitively easy, but can be difficult in practice since different languages have different punctuation conventions.

**Calculating probabilities**

- With word alignments, it is relatively easy to calculate probabilities.
  - e.g., What is the probability that *run* translates as *correr* in Spanish?
    1. Count up how many times *run* appears in the English part of your bi-text. e.g., 500 times
    2. Out of all those times, count up how many times it was translated as (i.e., aligns with) *correr*. e.g., 275 (out of 500) times.
    3. Divide to get a probability: 275/500 = 0.55, or 55%
- Word alignment gives us some frequency numbers, which we can use to align new cases, using other information, too (e.g., contextual information)

**The bag of words method**

- What if we’re not given word alignments?
  - How can we tell which English words are translated as which German words if we are only given an English text and a corresponding German text?
    - We can treat each sentence as a **bag of words** = unordered collection of words.
    - If word A appears in a sentence, then we will record all of the words in the corresponding sentence in the other language as appearing with it.
Example for bag of words method
Calculating probabilities: sentence 1

So, for He in He speaks Russian well/On khorosho govorit po-russki, we do the following:

1. Count up the number of Russian words: 4.
2. Assign each word equal probability of translation: $1/4 = 0.25$.

Example for bag of words method
Calculating probabilities: sentence 2

If we also have He is nice/On simpatich'nyi, then for He, we do the following:

1. Count up the number of possible translation words: 4 from the first sentence, 2 from the second = 6 total.
   - Note that we are NOT counting the number of English words: we count the number of possible translations.
2. Count up the number of times On is the translation = 2 times out of 6 = $1/3 = 0.33$, or 33%.

All other words have the probability $1/6 = 0.17$, or 17%, so On is the best translation for He.

Probabilities used in IBM models

Probabilistic models are generally more sophisticated, treating the problem as the source language generating the target and taking into account probabilities such as:

- $n(\text{word}) =$ probability of the number of words in the target language that the source word generates
- $p-$null = probability of a null word appearing
- $t(\text{word};\text{word}) =$ probability of a target word, given the source word (i.e., what we’ve seen so far)
- $d(\text{position};\text{position}) =$ probability of a target word appearing in position $\text{position}$, given the source position $\text{position}$

But we need alignments to estimate these parameters.

A Generative Story (IBM Models)


Beyond Bags of Words

A chicken-and-egg problem

- If we had the word alignments, we could estimate the parameters of our generative story.
- If we had the parameters, we could estimate the alignments.
**Expectation Maximization Algorithm**

The Expectation Maximization (EM) algorithm works forwards and backwards to estimate the probabilities:

**EM in a nutshell**

1. initialize model parameters (e.g. uniform)
2. (re-)assign probabilities to the missing data
3. (re-)estimate model parameters from completed data (weighted counts)
4. iterate, i.e., repeat steps 2&3 until you hit some stopping point

**After 1st Iteration**

... la maison ... la maison blue ... la fleur ...
... the house ... the blue house ... the flower ...

● Connections between e.g. la and the are more likely.

**After Another Iteration**

... la maison ... la maison bleu ... la fleur ...
... the house ... the blue house ... the flower ...

● Connections between e.g. fleur and flower are more likely (pigeon hole principle).

**Convergence**

... la maison ... la maison bleu ... la fleur ...
... the house ... the blue house ... the flower ...

\[
p(\text{la | the}) = 0.453 \\
p(\text{le | the}) = 0.334 \\
p(\text{maison | house}) = 0.876 \\
p(\text{bleu | blue}) = 0.563 \\
\]

**Phrase-Based Translation Overview**

But this word-based translation doesn’t account for many-to-many mappings between languages

- Foreign “phrases” are translated into English.
- Phrases may be reordered.

Current models allow for many-to-one mappings → we can use those to induce many-to-many mappings

- tomorrow I will fly to the conference in Canada

**Initial Step**

... la maison ... la maison blue ... la fleur ...
... the house ... the blue house ... the flower ...

● All connections equally likely.
Flaws of Word-Based MT

Semantically: In the way meanings work

Lexical ambiguity

Words can be **lexically ambiguous** = have multiple meanings.

- **bank** can be a financial institution or a place along a river.
- **can** can be a cylindrical object, as well as the act of putting something into that cylinder (e.g., John cans tuna.), as well as being a word like **must**, **might**, or **should**.

Many-to-many translation can handle non-compositional phrases.

Use of local context.

The more data, the longer the phrases that can be learned.

We have seen how MT systems can work, but MT is a very difficult task because languages are vastly different.

Lexical ambiguity

Languages differ:

- Lexically: In the words they use
- Syntactically: In the constructions they allow
- Semantically: In the way meanings work
- Pragmatically: In what readers take from a sentence.

In addition, there is a good deal of real-world knowledge that goes into a translation.
Semantic relationships

Often we find (roughly) **synonyms** between two languages:

- **English** book = **Russian** kniga
- **English** music = **Spanish** música

But words don’t always line up exactly between languages.

- **English** hypernyms = words that are more general in English than in their counterparts in other languages
  - **English** know is rendered by the French savoir (‘to know a fact’) and connaître (‘to know a thing’)
  - **English** library is German Bücherei if it is open to the public, but Bibliothek if it is intended for scholarly work.
- **English** hyponyms = words that are more specific in English than in their foreign language counterparts.
  - The German word berg can mean either hill or mountain in English.
  - The Russian word ruka can mean either hand or arm.

Venn diagram of semantic overlap

- **Etape** = **Spanish** etapa (journey), **French** jambe (human), **Dutch** pied (chair), **English** paw (animal)
- **Foot** = **Human** pied (human), **Paw** = patte (animal)

Semantic non-compositionality

Some verbs carry little meaning, so-called **light verbs**

- French faire une promenade is literally ‘make a walk,’ but it has the meaning of the English **take a walk**
- Dutch een poging doen ‘do an attempt’ means the same as the English **make an attempt**

And we often face **idioms** = expressions whose meaning is not made up of the meanings of the individual words.

- e.g., **English** kick the bucket
  - approximately equivalent to the French casser sa pipe (‘break his/her pipe’)
  - but we might want to translate it as mourir (‘die’)
  - and we want to treat it differently than kick the table

Idiosyncratic differences

Some words do not exist in a language and have to be translated with a more complex phrase: **lexical gap** or **lexical hole**.

- **French** gratiner means something like ‘to cook with a cheese coating’
- **Hebrew** stam means something like ‘I’m just kidding’ or ‘Nothing special’.

There are also idiosyncratic **collocations** among languages, e.g.:

- **English** heavy smoker
- **French** grand fumeur (‘large smoker’)
- **German** starker Raucher (‘strong smoker’)

Evaluating quality

Two main components in evaluating quality:

- **Intelligibility** = how understandable the output is
- **Accuracy** = how faithful the output is to the input

A common (though problematic) evaluation metric is the BLEU metric, based on n-gram comparisons.
Further reading

Some of the examples are adapted from the following books:
