6.864 (Fall 2007)

Machine Translation Part I

Overview

- Challenges in machine translation
- Classical machine translation
- A brief introduction to statistical MT
- Evaluation of MT systems
- The sentence alignment problem
- IBM Model 1

Lexical Ambiguity

Example 1:

book the flight ⇒ reservar

read the book ⇒ libro

Example 2:

the box was in the pen

the pen was on the table

Example 3:

kill a man ⇒ matar

kill a process ⇒ acabar

Differing Word Orders

- English word order is *subject – verb – object*
- Japanese word order is *subject – object – verb*

English:	IBM bought Lotus
Japanese:	*IBM Lotus bought*
English:	Sources said that IBM bought Lotus yesterday
Japanese:	*Sources yesterday IBM Lotus bought that said*

Syntactic Structure is not Preserved Across Translations

The bottle floated into the cave

⇓

La botella entro a la cuerva flotando
(the bottle entered the cave floating)

Syntactic Ambiguity Causes Problems

John hit the dog with the stick

⇓

John golpeo el perro con el palo/que tenia el palo

Pronoun Resolution

The computer outputs the data; it is fast.

⇓

La computadora imprime los datos; es rapida

The computer outputs the data; it is stored in ascii.

⇓

La computadora imprime los datos; estan almacendos en ascii

Differing Treatments of Tense

From Dorr et. al 1998:

Mary went to Mexico. During her stay she learned Spanish.

Went ⇒ iba (simple past/preterit)

Mary went to Mexico. When she returned she started to speak Spanish.

Went ⇒ fue (ongoing past/imperfect)

The Best Translation May not be 1-1

(From Manning and Schuetze):

According to our survey, 1988 sales of mineral water and soft drinks were much higher than in 1987, reflecting the growing popularity of these products. Cola drink manufacturers in particular achieved above average growth rates.

⇒

Quant aux eaux minerales et aux limonades, elles recontrent toujours plus d'adeptes. En effet notre sondage fait ressortir des ventes nettement superieures a celles de 1987, pour les boissons a base de cola notamment.

With regard to the mineral waters and the lemonades (soft drinks) they encounter still more users. Indeed our survey makes stand out the sales clearly superior to those in 1987 for cola-based drinks especially.

From Babel Fish:

Aznar ha premiado a Rodrigo Rato (vicepresidente primero), Javier Arenas (vicepresidente segundo y ministro de la Presidencia) y Eduardo Zaplana (ministro portavoz y titular de Trabajo) en la septima remodelacion de Gobierno en sus dos legislaturas. Las caras nuevas del Ejecutivo son las de Juan Costa, al frente del Ministerio de Ciencia y Tecnologia, y la de Julia Garcia Valdecasas, que ocupara la cartera de Administraciones Publicas.

⇓

Aznar has awarded to Rodrigo Short while (vice-president first), Javier Sands (vice-president second and minister of the Presidency) and Eduardo Zaplana (minister spokesman and holder of Work) in the seventh remodeling of Government in its two legislatures. The new faces of the Executive are those of Juan Coast, to the front of the Ministry of Science and Technology, and the one of Julia Garci'a Valdecasas, who will occupy the portfolio of Public Administrations.

An Example: Google Translation from Arabic

Stock prices retreated in the stock markets again with increasing concern about the circumstances surrounding the credit markets in the world, due mostly to the problems it faces American mortgage lending market, which raised concern among investors.
The index retreated Vuciji / 100 on the London Stock Exchange at the beginning of a percentage point in the dealings of up to 6082 points, while the Nikkei index retreated / 225 Japanese rate of 2.2% to close at the lowest level in eight months.
The American Jones index has lost about 1.6 points Tuesday to reach 13029 points, the Nasdaq index had lost 1.7 of its value.
These declines came despite statements by the American Federal Reserve Bank (Central Bank), in which he said that the process of pumping more funds into capital markets when necessary.
The American Federal Reserve Board, for the purposes of relaxation of tension in global financial markets, resulting in the Gaza backtrackings American real estate lending, have pumped billions of dollars of emergency funds allocation to the banking sector during the past few days, on Friday and Monday. As the European Central Bank did the same.

Overview

- Challenges in machine translation
- Classical machine translation
- A brief introduction to statistical MT
- Evaluation of MT systems
- The sentence alignment problem
- IBM Model 1

Direct Machine Translation

- Translation is word-by-word
- Very little analysis of the source text (e.g., no syntactic or semantic analysis)
- Relies on a large bilingual directionary. For each word in the source language, the dictionary specifies a set of rules for translating that word
- After the words are translated, simple reordering rules are applied (e.g., move adjectives after nouns when translating from English to French)

An Example of a set of Direct Translation Rules

(From Jurafsky and Martin, edition 2, chapter 25. Originally from a system from Panov 1960)

Rules for translating *much* or *many* into Russian:

if preceding word is *how* **return** *skol'ko*
else if preceding word is *as* **return** *stol'ko zhe*
else if word is *much*
    if preceding word is *very* **return** nil
    else if following word is a noun **return** *mnogo*
else (word is many)
    if preceding word is a preposition and following word is noun **return** *mnogii*
    else return *mnogo*

Some Problems with Direct Machine Translation

- Lack of any analysis of the source language causes several problems, for example:
  - Difficult or impossible to capture long-range reorderings

English:	Sources said that IBM bought Lotus yesterday
Japanese:	*Sources yesterday IBM Lotus bought that said*

  - Words are translated without disambiguation of their syntactic role
    e.g., *that* can be a complementizer or determiner, and will often be translated differently for these two cases

    They said *that* ...

    They like *that* ice-cream

Transfer-Based Approaches

- Three phases in translation:

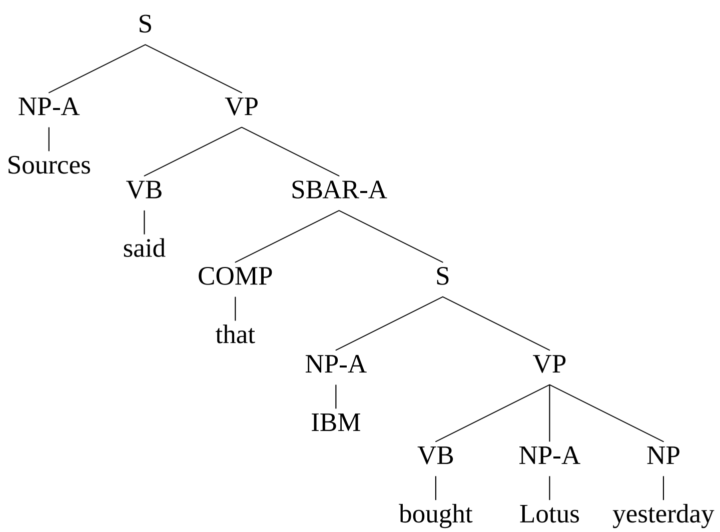
  Analysis: Analyze the source language sentence; for example, build a syntactic analysis of the source language sentence.

  Transfer: Convert the source-language parse tree to a target-language parse tree.

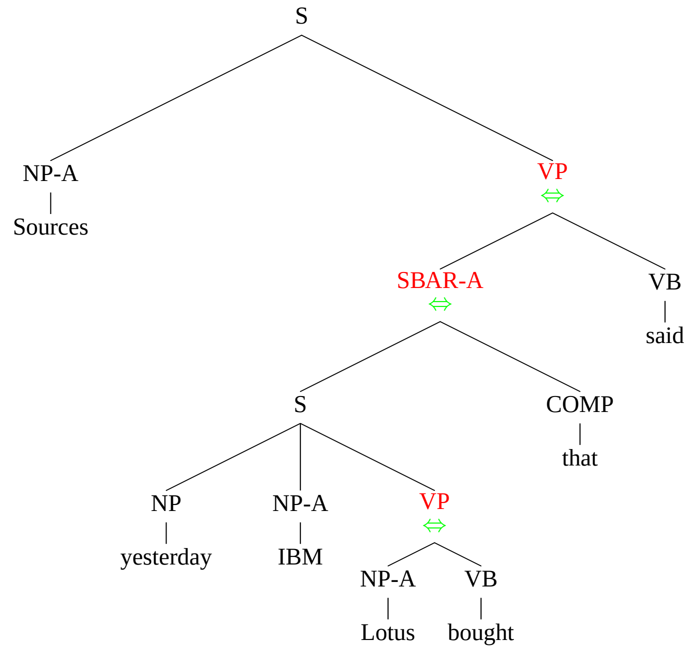
  Generation: Convert the target-language parse tree to an output sentence.

Transfer-Based Approaches

- The “parse trees” involved can vary from shallow analyses to much deeper analyses (even semantic representations).
- The transfer rules might look quite similar to the rules for direct translation systems. But they can now operate on syntactic structures.
- It’s easier with these approaches to handle long-distance reorderings
- The *Systran* systems are a classic example of this approach



⇒ Japanese: *Sources yesterday IBM Lotus bought that said*



Interlingua-Based Translation

- Two phases in translation:

  Analysis: Analyze the source language sentence into a (language-independent) representation of its meaning.

  Generation: Convert the meaning representation into an output sentence.

Interlingua-Based Translation

One Advantage: If we want to build a translation system that translates between n languages, we need to develop n analysis and generation systems. With a transfer based system, we'd need to develop $O(n^2)$ sets of translation rules.

Disadvantage: What would a language-independent representation look like?

Interlingua-Based Translation

- How to represent different concepts in an interlingua?
- Different languages break down concepts in quite different ways:

  German has two words for *wall*: one for an internal wall, one for a wall that is outside

  Japanese has two words for *brother*: one for an elder brother, one for a younger brother

  Spanish has two words for *leg*: *pierna* for a human's leg, *pata* for an animal's leg, or the leg of a table
- An interlingua might end up simple being an intersection of these different ways of breaking down concepts, but that doesn't seem very satisfactory...

Overview

- Challenges in machine translation
- Classical machine translation
- A brief introduction to statistical MT
- Evaluation of MT systems
- The sentence alignment problem
- IBM Model 1

A Brief Introduction to Statistical MT

- Parallel corpora are available in several language pairs
- Basic idea: use a parallel corpus as a training set of translation examples
- Classic example: IBM work on French-English translation, using the Canadian Hansards. (1.7 million sentences of 30 words or less in length).
- Idea goes back to Warren Weaver (1949): suggested applying statistical and cryptanalytic techniques to translation.

The Noisy Channel Model

- Goal: translation system from French to English
- Have a model $P(e \mid f)$ which estimates conditional probability of any English sentence e given the French sentence f. Use the training corpus to set the parameters.
- A Noisy Channel Model has two components:

$P(e)$ **the language model**

$P(f \mid e)$ **the translation model**

- Giving:

$$P(e \mid f) = \frac{P(e,f)}{P(f)} = \frac{P(e)P(f \mid e)}{\sum_e P(e)P(f \mid e)}$$

and

$$\text{argmax}_e P(e \mid f) = \text{argmax}_e P(e)P(f \mid e)$$

More About the Noisy Channel Model

- The **language model** $P(e)$ could be a trigram model, estimated from any data (parallel corpus not needed to estimate the parameters)
- The **translation model** $P(f \mid e)$ is trained from a parallel corpus of French/English pairs.
- Note:
  - The translation model is backwards!
  - The language model can make up for deficiencies of the translation model.
  - Later we'll talk about how to build $P(f \mid e)$
  - Decoding, i.e., finding

$$\text{argmax}_e P(e)P(f \mid e)$$

is also a challenging problem.

Example from Koehn and Knight tutorial

Translation from Spanish to English, candidate translations based on $P(Spanish \mid English)$ alone:

Que hambre tengo yo

$\rightarrow$

What hunger have $P(S|E)$ = 0.000014
Hungry I am so $P(S|E)$ = 0.000001
I am so hungry $P(S|E)$ = 0.0000015
Have i that hunger $P(S|E)$ = 0.000020

...

With $P(Spanish \mid English) \times P(English)$:

Que hambre tengo yo

$\rightarrow$

What hunger have $P(S|E)P(E)$ = 0.000014×0.000001
Hungry I am so $P(S|E)P(E)$ = $0.000001 \times 0.0000014$
I am so hungry $P(S|E)P(E)$ = 0.0000015×0.0001

Have i that hunger $P(S|E)P(E)$ = $0.000020 \times 0.00000098$

...

Overview

- Challenges in machine translation
- Classical machine translation
- A brief introduction to statistical MT
- Evaluation of MT systems
- The sentence alignment problem
- IBM Model 1

Evaluation of Machine Translation Systems

- Method 1: human evaluations
  accurate, **but** expensive, slow
- "Cheap" and fast evaluation is essential
- We'll discuss one prominent method:
  Bleu (Papineni, Roukos, Ward and Zhu, 2002)

Evaluation of Machine Translation Systems

Bleu (Papineni, Roukos, Ward and Zhu, 2002):

Candidate 1: It is a guide to action which ensures that the military always obeys the commands of the party.

Candidate 2: It is to insure the troops forever hearing the activity guidebook that party direct.

Reference 1: It is a guide to action that ensures that the military will forever heed Party commands.

Reference 2: It is the guiding principle which guarantees the military forces always being under the command of the Party.

Reference 3: It is the practical guide for the army always to heed the directions of the party.

Unigram Precision

- **Unigram Precision** of a candidate translation:

$$\frac{C}{N}$$

where N is number of words in the candidate, C is the number of words in the candidate which are in at least one reference translation.

e.g.,

Candidate 1: It is a guide to action which ensures that the military always obeys the commands of the party.

$$Precision = \frac{17}{18}$$

(only *obeys* is missing from all reference translations)

Modified Unigram Precision

- Problem with unigram precision:

  Candidate: the the the the the the the

  Reference 1: the cat sat on the mat

  Reference 2: there is a cat on the mat

  precision = 7/7 = 1???

- **Modified unigram precision:** “Clipping”
  - Each word has a “cap”. e.g., *cap(the) = 2*
  - A candidate word w can only be correct a maximum of $cap(w)$ times. e.g., in candidate above, $cap(the) = 2$, and *the* is correct twice in the candidate $\Rightarrow$

$$Precision = \frac{2}{7}$$

Modified N-gram Precision

- Can generalize modifi ed unigram precision to other n-grams.

- For example, for candidates 1 and 2 above:

$$Precision_1(bigram) = \frac{10}{17}$$

$$Precision_2(bigram) = \frac{1}{13}$$

Precision Alone Isn’t Enough

Candidate 1: of the

Reference 1: It is a guide to action that ensures that the military will forever heed Party commands.

Reference 2: It is the guiding principle which guarantees the military forces always being under the command of the Party.

Reference 3: It is the practical guide for the army always to heed the directions of the party.

$$Precision(unigram) = 1$$

$$Precision(bigram) = 1$$

But Recall isn’t Useful in this Case

- Standard measure used in addition to precision is **recall**:

$$Recall = \frac{C}{N}$$

where C is number of n-grams in candidate that are correct, N is number of words in the references.

Candidate 1: I always invariably perpetually do.

Candidate 2: I always do

Reference 1: I always do

Reference 1: I invariably do

Reference 1: I perpetually do

Sentence Brevity Penalty

- Step 1: for each candidate, compute closest matching reference (in terms of length)
  e.g., our candidate is length 12, references are length $12, 15, 17$. Best match is of length 12.
- Step 2: Say l_i is the length of the i'th candidate, r_i is length of best match for the i'th candidate, then compute

$$brevity = \frac{\sum_i r_i}{\sum_i l_i}$$

  (I think! from the Papineni paper, although $brevity = \frac{\sum_i r_i}{\sum_i min(l_i, r_i)}$ might make more sense?)
- Step 3: compute brevity penalty

$$BP = \begin{cases} 1 & \text{If } brevity < 1 \\ e^{1-brevity} & \text{If } brevity \geq 1 \end{cases}$$

  e.g., if $r_i = 1.1 \times l_i$ for all i (candidates are always 10% too short) then $BP = e^{-0.1} = 0.905$

The Final Score

- Corpus precision for any n-gram is

$$p_n = \frac{\sum_{C \in \{Candidate\}} \sum_{ngram \in C} \quad Count_{clip}(ngram)}{\sum_{C \in \{Candidate\}} \sum_{ngram \in C} \quad Count(ngram)}$$

  i.e. number of correct ngrams in the candidates (after "clipping") divided by total number of ngrams in the candidates
- Final score is then

$$Bleu = BP \times (p_1 p_2 p_3 p_4)^{1/4}$$

  i.e., BP multiplied by the geometric mean of the unigram, bigram, trigram, and four-gram precisions

Overview

- Challenges in machine translation
- Classical machine translation
- A brief introduction to statistical MT
- Evaluation of MT systems
- The sentence alignment problem
- IBM Model 1

The Sentence Alignment Problem

- Might have 1003 sentences (in sequence) of English, 987 sentences (in sequence) of French: **but which English sentence(s) corresponds to which French sentence(s)?**

e_1 f_1
e_2 f_2
e_3 f_3
e_4 f_4
e_5 f_5
e_6 f_6
e_7 f_7
$\ldots$

$\Rightarrow$

e_1 f_1
e_2
----
e_3 f_2
----
e_4 f_3
----
e_5 f_4
f_5
----
e_6 f_6
e_7 f_7

$\ldots$

- Might have 1-1 alignments, 1-2, 2-1, 2-2 etc.

The Sentence Alignment Problem

- Clearly needed before we can train a translation model
- Also useful for other multi-lingual problems
- Two broad classes of methods we'll cover:
  - Methods based on sentence lengths alone.
  - Methods based on lexical matches, or "cognates".

Sentence Length Methods

(Gale and Church, 1993):

- Method assumes paragraph alignment is known, sentence alignment is not known.
- Define:
  - l_e = length of English sentence, in characters
  - l_f = length of French sentence, in characters
- Assumption: given length l_e, length l_f has a gaussian/normal distribution with mean $c \times l_e$, and variance $s^2 \times l_e$ for some constants c and s.
- Result: we have a cost

$$Cost(l_e, l_f)$$

for any pairs of lengths l_e and l_f.

Each Possible Alignment Has a Cost

e_1 f_1
e_2
----
e_3 f_2
----
e_4 f_3
----
e_5 f_4
f_5
----
e_6 f_6
e_7 f_7

...

In this case, if length of e_i is l_i, and length of f_i is m_i, total cost is

$$\begin{aligned} Cost = {} & Cost(l_1 + l_2, m_1) + Cost_{21} + \\ & Cost(l_3, m_2) + Cost_{11} + \\ & Cost(l_4, m_3) + Cost_{11} + \\ & Cost(l_4, m_4 + m_5) + Cost_{12} + \\ & Cost(l_6 + l_7, m_6 + m_7) + Cost_{22} \end{aligned}$$

where $Cost_{ij}$ terms correspond to costs for 1-1, 1-2, 2-1 and 2-2 alignments.

- Dynamic programming can be used to search for the lowest cost alignment

Methods Based on Cognates

- Intuition: related words in different languages often have similar spellings e.g., government and gouvernement
- Cognate matches can "anchor" sentence-sentence correspondences
- A method from (Church 1993): track all 4-grams of characters which are identical in the two texts.
- A method from (Melamed 1993), measures similarity of words A and B:

$$LCSR(A, B) = \frac{length(LCS(A, B))}{max(length(A), length(B))}$$

where LCS is the longest common subsequence (not necessarily contiguous) in A and B. e.g.,

$$LCSR(\text{government,gouvernement}) = \frac{10}{13}$$

More on Melamed’s Definition of Cognates

- Various refinements (for example, excluding common/stop words such as “the”, “a”)
- Melamed uses a cut-off of 0.58 for LCSR to identify cognates: 25% of words in Hansards are then part of a cognate
- Represent an English/French parallel text e/f as a “bitext”: graph where we have a point at position (x, y) if and only if $word_x$ in e is a cognate of $word_y$ in f.
- Melamed then uses a greedy method to identify a diagonal chain of cognates through the parallel text.

Overview

- Challenges in machine translation
- Classical machine translation
- A brief introduction to statistical MT
- Evaluation of MT systems
- The sentence alignment problem
- IBM Model 1
  - How do we model $P(f \mid e)$?

IBM Model 1: Alignments

- How do we model $P(f \mid e)$?
- English sentence e has l words $e_1 \dots e_l$,
  French sentence f has m words $f_1 \dots f_m$.
- An **alignment** a identifies which English word each French word originated from
- Formally, an **alignment** a is $\{a_1, \dots a_m\}$, where each $a_j \in \{0 \dots l\}$.
- There are $(l + 1)^m$ possible alignments.

IBM Model 1: Alignments

- e.g., $l = 6$, $m = 7$

$e =$ And the program has been implemented

$f =$ Le programme a ete mis en application

- One alignment is

$$\{2, 3, 4, 5, 6, 6, 6\}$$

- Another (bad!) alignment is

$$\{1, 1, 1, 1, 1, 1, 1\}$$

IBM Model 1: Alignments

- In IBM model 1 all allignments a are equally likely:

$$P(a \mid e) = C \times \frac{1}{(l+1)^m}$$

where $C = prob(length(f) = m)$ is a constant.

- This is a **major** simplifying assumption, but it gets things started...

IBM Model 1: Translation Probabilities

- Next step: come up with an estimate for

$$P(f \mid a, e)$$

- In model 1, this is:

$$P(f \mid a, e) = \prod_{j=1}^{m} P(f_j \mid e_{a_j})$$

- e.g., $l = 6$, $m = 7$

$$e = \text{And the program has been implemented}$$
$$f = \text{Le programme a ete mis en application}$$

- $a = \{2, 3, 4, 5, 6, 6, 6\}$

$$\begin{aligned} P(f \mid a, e) \;\; = \;\; & P(Le \mid the) \times \\ & P(programme \mid program) \times \\ & P(a \mid has) \times \\ & P(ete \mid been) \times \\ & P(mis \mid implemented) \times \\ & P(en \mid implemented) \times \\ & P(application \mid implemented) \end{aligned}$$

IBM Model 1: The Generative Process

To generate a French string f from an English string e:

- **Step 1:** Pick the length of f (all lengths equally probable, probability C)
- **Step 2:** Pick an alignment a with probability $\frac{1}{(l+1)^m}$
- **Step 3:** Pick the French words with probability

$$P(f \mid a, e) = \prod_{j=1}^{m} P(f_j \mid e_{a_j})$$

The final result:

$$P(f, a \mid e) = P(a \mid e) \times P(f \mid a, e) = \frac{C}{(l+1)^m} \prod_{j=1}^{m} P(f_j \mid e_{a_j})$$

A Hidden Variable Problem

- **We have:**

$$P(f, a \mid e) = \frac{C}{(l+1)^m} \prod_{j=1}^{m} P(f_j \mid e_{a_j})$$

- **And:**

$$P(f \mid e) = \sum_{a \in \mathcal{A}} \frac{C}{(l+1)^m} \prod_{j=1}^{m} P(f_j \mid e_{a_j})$$

  where $\mathcal{A}$ is the set of all possible alignments.

A Hidden Variable Problem

- Training data is a set of (f_i, e_i) pairs, likelihood is

$$\sum_i \log P(f \mid e) = \sum_i \log \sum_{a \in \mathcal{A}} P(a \mid e_i) P(f_i \mid a, e_i)$$

  where $\mathcal{A}$ is the set of all possible alignments.

- We need to maximize this function w.r.t. the translation parameters $P(f_j \mid e_{a_j})$.

- EM can be used for this problem: initialize translation parameters randomly, and at each iteration choose

$$\Theta_t = \text{argmax}_{\Theta} \sum_i \sum_{a \in \mathcal{A}} P(a \mid e_i, f_i, \Theta^{t-1}) \log P(f_i \mid a, e_i, \Theta)$$

  where Θ^t are the parameter values at the t'th iteration.

An Example

- I have the following training examples

  the dog ⇒ le chien
  the cat ⇒ le chat

- Need to find estimates for:

$$P(le \mid the) \quad P(chien \mid the) \quad P(chat \mid the)$$
$$P(le \mid dog) \quad P(chien \mid dog) \quad P(chat \mid dog)$$
$$P(le \mid cat) \quad P(chien \mid cat) \quad P(chat \mid cat)$$

- As a result, each (e_i, f_i) pair will have a most likely alignment.