

Intro to SMT

Sara Stymne

2020-09-07

Partly based on slides by Jörg Tiedemann and Fabienne Cap

The revolution of the empiricists

Classical approaches require lots of manual work!

- long development times
- low coverage, not robust
- disambiguation at various levels → slow!

Learn from translation data:

- example databases for CAT and MT
- bilingual lexicon/terminology extraction
- **statistical translation models**

Motivation for Data-Driven MT

How do we learn to translate?

- grammar vs. examples
- teacher vs. practice
- intuition vs. experience

Is it possible to create an MT engine without any human effort?

- no writing of grammar rules
- no bilingual lexicography
- no writing of preference & disambiguation rules

Motivating example

Imagine a spaceship with aliens coming to earth, telling you:

pe- li kaj meni

Translation? Anyone?

Motivating example

Imagine a spaceship with aliens coming to earth, telling you:

pe- kaj meni

Translation? Anyone?

Problem:

- Human translators may not be available
- Human translators are expensive

Motivating example

Imagine a spaceship with aliens coming to earth, telling you:

pe- kaj meni

Translation? Anyone?

Problem:

- Human translators may not be available
- Human translators are expensive

Possible solution:

We found a collection of translated text!

Practical exercise

15–20 minutes

Try to learn to translate the alien language!

What can we learn from this exercise?

- We can learn to translate from translated texts
- 1-to-1 translations are easier to identify than 1-to-n n-to-1 or n-to-m
- unseen words cannot be translated
- ambiguity: some words have more than one correct translation → the context helps determine which one
- sometimes words need to be reordered

Motivation for Data-Driven MT

Learning to translate:

- there is a bunch of translated stuff (collect all)
- learn common word/phrase translations from this collection
- look at typical sentences in the target language
- learn how to write a sentence in the target language

Motivation for Data-Driven MT

Learning to translate:

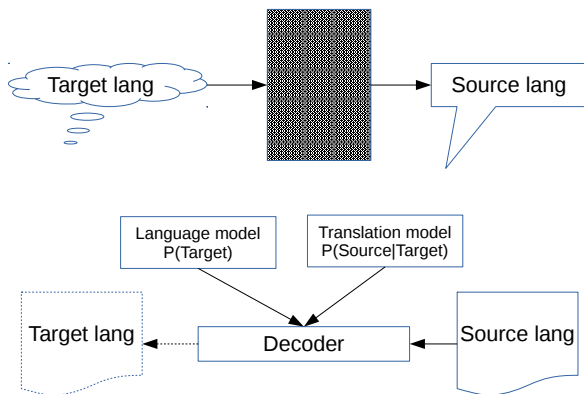
- there is a bunch of translated stuff (collect all)
- learn common word/phrase translations from this collection
- look at typical sentences in the target language
- learn how to write a sentence in the target language

Translation:

- try various translations of words/phrases in given sentence
- put them together, shuffle them around
- check which translation candidate looks best

Statistical Machine Translation

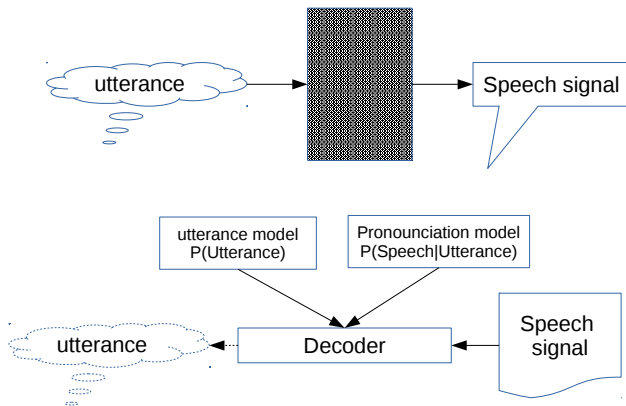
Noisy channel for MT: “What could have been the sentence that has generated the observed source language sentence?”



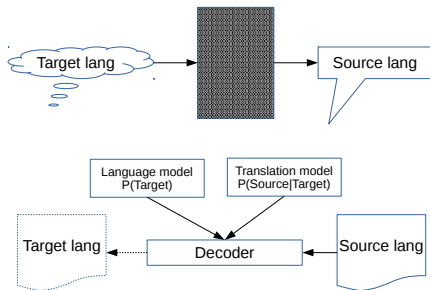
... what a strange idea!

Statistical Machine Translation

Ideas borrowed from Speech Recognition:



Statistical Machine Translation



Probabilistic view on MT (T = target language, S = source language):

$$\begin{aligned}\hat{T} &= \operatorname{argmax}_T P(T|S) \\ &= \operatorname{argmax}_T P(S|T)P(T)\end{aligned}$$

Statistical Machine Translation Modeling

- model translation as an optimization (search) problem
- look for the **most likely translation T** for a given input S
- use a probabilistic model that assigns these conditional likelihoods
- use Bayes theorem to split the model into 2 parts:
 - a language model (for the target language)
 - a translation model (source language given target language)

Statistical Machine Translation

- Learn statistical models automatically from bilingual corpora
- Bilingual corpora: collections of texts translated by humans
- Use the models to translate unseen texts

Statistical Machine Translation

- Learn statistical models automatically from bilingual corpora
- Bilingual corpora: collections of texts translated by humans
- Use the models to translate unseen texts
- Models can be have different granularity
 - Word-based
 - Phrase-based – sequences of words
 - Hierarchical – tree structures
 - Syntactical – linguistically motivated tree structures

Some (very) basic concepts of probability theory

- probability $P(X)$ maps event X to number between 0 and 1
- $P(X)$ represents the likelihood of observing event X in some kind of experiment (trial)
- discrete probability distribution: $\sum_i P(X = x_i) = 1$

Some (very) basic concepts of probability theory

- probability $P(X)$ maps event X to number between 0 and 1
- $P(X)$ represents the likelihood of observing event X in some kind of experiment (trial)
- discrete probability distribution: $\sum_i P(X = x_i) = 1$
- $P(X|Y) =$ **conditional probability** (likelihood of event X given that event Y has been observed before)

Some (very) basic concepts of probability theory

- probability $P(X)$ maps event X to number between 0 and 1
- $P(X)$ represents the likelihood of observing event X in some kind of experiment (trial)
- discrete probability distribution: $\sum_i P(X = x_i) = 1$
- $P(X|Y) =$ **conditional probability** (likelihood of event X given that event Y has been observed before)
- **joint probability**: $P(X, Y)$ (likelihood of seeing both events)
- $P(X, Y) = P(X) * P(Y|X) = P(Y) * P(X|Y)$

Some (very) basic concepts of probability theory

- probability $P(X)$ maps event X to number between 0 and 1
- $P(X)$ represents the likelihood of observing event X in some kind of experiment (trial)
- discrete probability distribution: $\sum_i P(X = x_i) = 1$
- $P(X|Y) =$ **conditional probability** (likelihood of event X given that event Y has been observed before)
- **joint probability**: $P(X, Y)$ (likelihood of seeing both events)
- $P(X, Y) = P(X) * P(Y|X) = P(Y) * P(X|Y)$, therefore:

$$\text{Bayes Theorem: } P(X|Y) = \frac{P(X) * P(Y|X)}{P(Y)}$$

Some quick words on probability theory & Statistics

Where do the probabilities come from? → Experience!

Use experiments (and repeat them often)

Maximum Likelihood Estimation (rely on N experiments only):

$$P(X) \approx \frac{\text{count}(X)}{N}$$

Some quick words on probability theory & Statistics

Where do the probabilities come from? → Experience!

Use experiments (and repeat them often)

Maximum Likelihood Estimation (rely on N experiments only):

$$P(X) \approx \frac{\text{count}(X)}{N}$$

For conditional probabilities:

$$P(X|Y) = \frac{P(X, Y)}{P(Y)} \approx \frac{\text{count}(X, Y) * N}{\text{count}(Y) * N} = \frac{\text{count}(X, Y)}{\text{count}(Y)}$$

Translation Model Parameters

Lexical translations:

- das → the
- haus → house, home, building, household, shell
- ist → is
- klein → small, low

Multiple translation options:

- learn translation probabilities from data
- use the most common one in that context

Context-independent models

Count translation statistics:

- How often is *Haus* translated into:

Translation of <i>Haus</i>	Count
house	8,000
building	1,600
home	200
household	150
shell	50
	10,000

Context-independent models

- Maximum likelihood estimation (MLE)

$$t(s|t) = \frac{\text{count}(s,t)}{\text{count}(t)} \quad (1)$$

- for $s = \textit{Haus}$:
 - $t(s|t) = 0.8$ if $t = \textit{house}$
 - $t(s|t) = 0.16$ if $t = \textit{building}$
 - $t(s|t) = 0.2$ if $t = \textit{home}$
 - $t(s|t) = 0.015$ if $t = \textit{household}$
 - $t(s|t) = 0.005$ if $t = \textit{shell}$

(Classical) Statistical Machine Translation

$$\begin{aligned}\hat{T} &= \mathit{argmax}_T P(T|S) \\ &= \mathit{argmax}_T \frac{P(S|T)P(T)}{P(S)} \\ &= \mathit{argmax}_T P(S|T)P(T)\end{aligned}$$

(Classical) Statistical Machine Translation

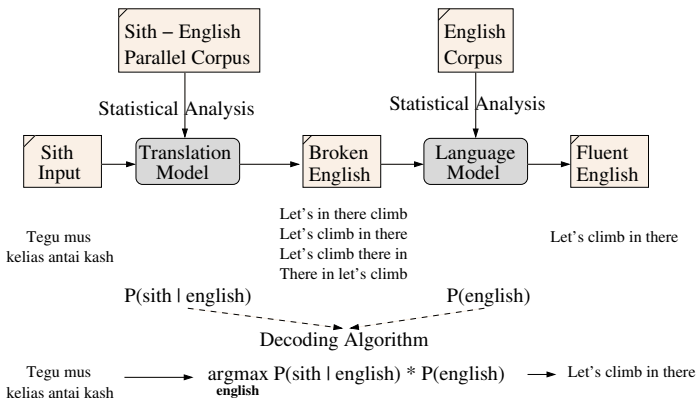
$$\begin{aligned}\hat{T} &= \operatorname{argmax}_T P(T|S) \\ &= \operatorname{argmax}_T \frac{P(S|T)P(T)}{P(S)} \\ &= \operatorname{argmax}_T P(S|T)P(T)\end{aligned}$$

Translation model: $P(S|T)$, estimated from (big) parallel corpora, takes care of **adequacy**

Language model: $P(T)$, estimated from (huge) monolingual target language corpora, takes care of **fluency**

Decoder: global search for $\operatorname{argmax}_T P(S|T)P(T)$ for a given sentence S

Modelling Statistical Machine Translation



The role of the translation and language model

- Translation model: prefer **adequate** translations
 - $P(\text{Das Haus ist klein} \text{---} \text{The house is small}) >$
 - $P(\text{Das Haus ist klein} \text{---} \text{The } \mathbf{building} \text{ is small}) >$
 - $P(\text{Das Haus ist klein} \text{---} \text{The } \mathbf{shell} \text{ is } \mathbf{low})$
- Language model: prefer **fluent** translations:
 - $P(\text{The house is small}) >$
 - $P(\text{The is house small})$

Word-based SMT models

Why do we need word alignment?

- Cannot directly estimate $P(S|T)$... Why not?

Word-based SMT models

Why do we need word alignment?

- Cannot directly estimate $P(S|T)$... Why not?

- almost all sentences are unique

- sparse counts! → no good estimations

→ decompose into smaller chunks!

Word-based SMT models

Why do we need word alignment?

- Cannot directly estimate $P(S|T)$... Why not?

- almost all sentences are unique

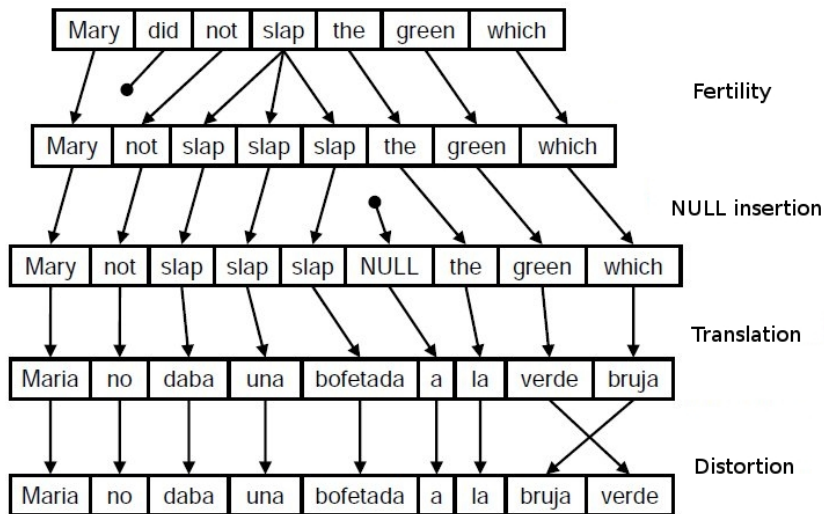
- sparse counts! → no good estimations

→ decompose into smaller chunks!

Word-based model: Assume that words in one language have been generated by words in another!

→ a (hidden) word alignment explains this process

Word-based Translation Models



Word-based Translation Models

What do we need to estimate model parameters?

- lexical translation
- distortion/re-ordering
- fertility
- NULL insertion

→ We need a word-aligned parallel corpus!

Word alignment

What is word alignment? A simple example:

1	2	3	4
das	Haus	ist	klein
↑	↑	↑	↑
the	house	is	small
1	2	3	4

Word alignment

Another visualization:

	the	house	is	small
das	■	□	□	□
Haus	□	■	□	□
ist	□	□	■	□
klein	□	□	□	■

Word alignment

Natural languages are not that easy ...

- not always 1:1 relation between words
- some words may be dropped
- word order can be quite different

Word alignment example

	Nyss	hade	jag	precis	tappat	bort	glassen
A	■						
moment	■						
ago	■						
I			■				
had		■					
just				■			
lost					■	■	
my							
ice							■
cream							■

Statistical word alignment models

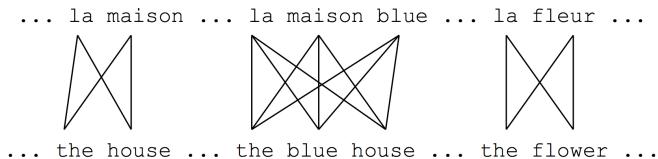
Standard word-based translation models:

- IBM 1: lexical translation probabilities
- IBM 2: add absolute reordering
- IBM 3: add fertility
- IBM 4: relative reordering
- IBM 5: fix deficiency
- HMM model (2): relative distortion

Training word alignment models

- Learning with incomplete data
 - word alignment is hidden
 - need to fill in the gaps in the data
- Expectation Maximization (EM) algorithm
 - 1 Initialize model parameters (e.g. uniform)
 - 2 Assign probabilities to the missing data
 - 3 Estimate model parameters from completed data
 - 4 iterate steps 2–3 (to convergence, or a set number of times)

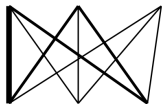
EM algorithm



- Initialization: all alignments are equally likely
- Model learns that **la**, for example, is often aligned with **the**

EM algorithm

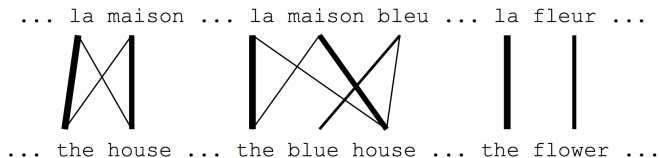
... la maison ... la maison blue ... la fleur ...



... the house ... the blue house ... the flower ...

- After one iteration
- Certain alignments, for example between **la** and **the**, are now more likely

EM algorithm



- After another iteration
- It becomes apparent the other alignments, such as **fleur** and **flower**, are more likely

EM algorithm

... la maison ... la maison bleu ... la fleur ...
/ | | X | |
... the house ... the blue house ... the flower ...

- Convergence
- Inherent hidden structure revealed by EM

EM algorithm

... la maison ... la maison bleu ... la fleur ...
/ | | X | |
... the house ... the blue house ... the flower ...



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

IBM Model 1 and EM

- EM Algorithm consists of two steps
- Expectation-Step: Apply model to the data
 - parts of the model are hidden (here: alignments)
 - using the model, assign probabilities to possible alignments
- Maximization-Step: Estimate model from data
 - take assigned values as fractional counts
 - collect counts (weighted by probabilities)
 - estimate model from counts
- Iterate these steps until convergence

EM and the IBM models

IBM Model 1	lexical translation
IBM Model 2	adds absolute reordering model
IBM Model 3	adds fertility model
IBM Model 4	relative reordering model
IBM Model 5	fixes deficiency

- EM algorithm can be applied to all IBM models
- With lower IBM models we can apply certain mathematical tricks to simplify calculations (see course textbook)
- Only with IBM Model 1 are we guaranteed to reach a global maximum

EM and the IBM models

IBM Model 1	lexical translation
IBM Model 2	adds absolute reordering model
IBM Model 3	adds fertility model
IBM Model 4	relative reordering model
IBM Model 5	fixes deficiency

- From IBM Model 3 computation becomes more expensive and sampling over high probability alignments is employed
- Typical training scheme use all IBM models sequentially, using results from one to initialize the next
- Popular implementation: GIZA++

Typical Training Scheme

- Iterations over alignment models of increasing complexity:
 - 1 n EM iterations of IBM Model 1 with uniform initialization
 - 2 n EM iterations of IBM Model 2 or HMM initialized with Model 1
 - 3 parameter transfer from IBM Model 2 / HMM to IBM Model 3
 - 4 n hill-climbing iterations of IBM Model 3 based on best alignment
 - 5 parameter transfer from IBM Model 3 to IBM Model 4
 - 6 n hill-climbing iterations of IBM Model 4 based on best alignment
- Typical number of iterations: 5
- Popular implementation: GIZA++

Statistical Machine Translation

Remember:

$$\hat{T} = \mathit{argmax}_T P(S|T)P(T)$$

- aligned parallel corpora \rightarrow translation model

What is missing?

Statistical Machine Translation

Remember:

$$\hat{T} = \operatorname{argmax}_T P(S|T)P(T)$$

- aligned parallel corpora \rightarrow translation model

What is missing?

- aligned parallel corpora \rightarrow translation model $P(S|T)$
- we still need the **language model** $P(T)$

\rightarrow Standard N-gram language models

Statistical Machine Translation: Language Modeling

Language modeling:

- (probabilistic) LM = predict likelihood of any given string
- What is the likelihood $P(T)$ to observe sentence T ?

$P_{LM}(\text{the house is small}) > P_{LM}(\text{small the is house})$

$P_{LM}(\text{small step}) > P_{LM}(\text{little step})$

N-gram language models

- Markov chain

- $p(w_1, w_2, \dots, w_n) = p(w_1)p(w_2|w_1) \dots p(W_n|w_1, w_2, \dots, w_{n-1})$

- Markov assumption

- $p(w_1, w_2, \dots, w_n) \simeq p(w_n|w_{n-m}, \dots, w_{n-2}, w_{n-1})$

- Maximum likelihood estimation

- $p(w_3|w_1, w_2) = \frac{\text{count}(w_1, w_2, w_3)}{\sum_w \text{count}(w_1, w_2, w)}$

N-gram language models

- Markov chain

- $p(w_1, w_2, \dots, w_n) = p(w_1)p(w_2|w_1) \dots p(W_n|w_1, w_2, \dots, w_{n-1})$

- Markov assumption

- $p(w_1, w_2, \dots, w_n) \simeq p(w_n|w_{n-m}, \dots, w_{n-2}, w_{n-1})$

- Maximum likelihood estimation

- $p(w_3|w_1, w_2) = \frac{\text{count}(w_1, w_2, w_3)}{\sum_w \text{count}(w_1, w_2, w)}$

- unigram model: $P(T) = P(t_1) * P(t_2) \dots P(t_n)$

- bigram model: $P(T) = P(t_1) * P(t_2|t_1) * P(t_3|t_2) \dots P(t_n|t_{n-1})$

- trigram model:

- $P(T) = P(t_1) * P(t_2|t_1) * P(t_3|t_1, t_2) \dots P(t_n|t_{n-2}t_{n-1},)$

A note on word-based SMT

Today, word-based translation models are **outdated**, but they introduce some **important concepts** which are still relevant for state-of-the-art SMT models:

- generative modelling
- noisy-channel model
- word alignment and IBM models 1–5
- expectation-maximisation algorithm

Next lecture we will focus on phrase-based SMT!

Summary

- MT can be put into a probabilistic framework
- **translation models**: estimated from parallel corpora
- **language models**: estimated from monolingual corpora
- global search = **decoding** = translating

→ fully automatic

→ various simplifications / assumptions necessary

→ probabilistic variant of direct translation

Coming up

- This week:
 - Lecture on PBSMT
 - Assignment 2: Moses
- Coming weeks:
 - Lectures on sequence models and NMT
 - Assignments: on LMs and NMT
 - Lab: sequence to sequence models and attention
 - Project groups

Project groups

- Low-resource languages
 - Sinhala
 - Galician
 - Inuktitut
 - Kazakh
 - Gujarati
 - Xhosa
- Note that you cannot work on a language which you know or if you know a closely related language, i.e. do not choose Galician if you know Portuguese
- You will be able to sign up for a project group in Studentportalen soon
- Registration will open at a set date (I'm holding off a bit, so that everyone will have the chance to become registered first)