INF5820/INF9820

LANGUAGE TECHNOLOGICAL APPLICATIONS

Jan Tore Lønning, Lecture 8, 12 Oct. 2016

jtl@ifi.uio.no

Today

- Preparing bitext
- Parameter tuning
- Reranking
- Some linguistic issues

STMT so far

- We have seen all the main elements of a phrase-based STMT-system:
  - The model
  - How to train it
  - Decoding
- For actually building a system, we will use some tools implementing this model, e.g. Moses
- Remains some details at both ends
  - Start: Preprocessing
  - End: Tuning

Preprocessing

- (Exercise 4.1) Which preparations must be done to a training corpus to make it suitable for training an SMT system?
- Clean up text
  - Character encoding
  - Mark-up (XML, html)
- Tokenization
  - Why?

Preprocessing: casing

- Case folding (downcasing) or not?
  - The city is large. → the city is large .
  - Washington is large.→ washington is large .
- Why not?
  - Mary met Smith and Browne → mary met smith and browne
  - Adding ambiguity
  - Translating proper names wrongly

Alternative: true casing

- What
  - The city is large. → the city is large .
  - Washington is large.→ Washington is large .
- How?
  - Read through corpus, count occurrences which are not sentence initial → model
  - Use model on corpus
- Remaining problem: beginning of sentence
  - Browne met Smith → ?
  - Green men entered the room →
  - Cannot always do right. But true casing is best alternative.

Sentence align the source and target

- Assume two parallel texts which are split into sentences
  - E: e1, e2, …, en
  - F: f1, f2, …, fm
  - Not necessarily equally many
- Task: organize E and F into equally many segments
  - E: E1, E2, …, Ek
  - F: F1, F2, …, Fk
  - Where Ei corresponds to Fi
  - Ei consists of some consecutive sentences: e_j, e_{j+1}, …, e_{j+s}
  - Similarly for Fi: f_k, fk_{+1}, …, f_{k+u}
  - Ei and Fi are minimal:
    - There is no sub segment Ei' of Ei and Fi' of Fi such that Ei' and Fi' correspond and Ei'=/=Ei or Fi'=/=Fi

Sentence alignment continued

- Sentence pairs:
  - *x* many sentences in Ei, *y* many sentences in Fi
  - Then call Ei-Fi an *x*-*y* pair
  - 1-1pair: Good for training
  - 1- 0 pair or 0 – 1 pair: Ignore for training
  - 1 – 2 or 2 – 1 pairs. Might be considered for training.
- How to identify sentences?
  - Sentence length
  - Word pairs from dictionary

Today

- Preparing bitext
- Parameter tuning
- Reranking
- Some linguistic issues

The generative SMT-model

- Adding weights:
  - Koehn, lecture 5, Slide 17-21

How to tune weights?

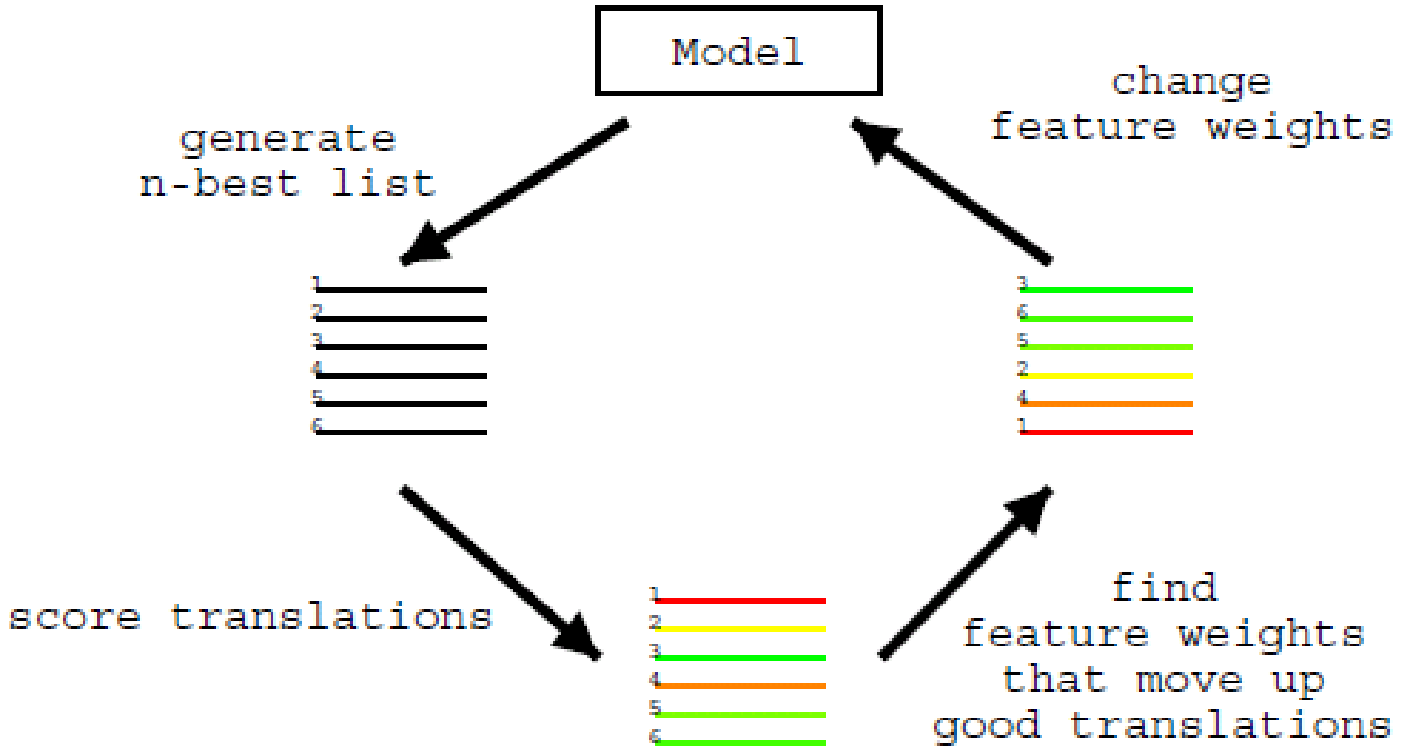
1. Make an original system, S0, using a parallel corpus, C1, for the phrase table.
2. Use a distinct small parallel corpus, C2. (dev set)
3. Produce several S0-translations for each f-sentence in C2.
   - n-best list (n=100, 1000, 10000)
4. Use a method for scoring the candidate translations in C2.
   - (typically modified BLEU-score).
5. Try to adjust the weights to bring the best candidates in (4) towards top of list.
6. Make new system with adjusted weights.
7. Repeat from 3 towards convergence.

Learning task

- Task: *find weights*, so that feature vector of the correct translations *ranked first*

rank	translation	LM	TM	WP	BBR
1	Mary not give slap witch green .	-17.2	-5.2	-7	1
2	Mary not slap the witch green .	-16.3	-5.7	-7	1
3	Mary not give slap of the green witch .	-18.1	-4.9	-9	1
4	Mary not give of green witch .	-16.5	-5.1	-8	1
5	Mary did not slap the witch green .	-20.1	-4.7	-8	1
6	Mary did not slap green witch .	-15.5	-3.2	-7	1
7	Mary not slap of the witch green .	-19.2	-5.3	-8	1
8	Mary did not give slap of witch green .	-23.2	-5.0	-9	1
9	Mary did not give slap of the green witch .	-21.8	-4.4	-10	1
10	Mary did slap the witch green .	-15.5	-6.9	-7	1
11	**Mary did not slap the green witch .**	**-17.4**	**-5.3**	**-8**	**0**
12	Mary did slap witch green .	-16.9	-6.9	-6	1
13	Mary did slap the green witch .	-14.3	-7.1	-7	1
14	Mary did not slap the of green witch .	-24.2	-5.3	-9	1
15	Mary did not give slap the witch green .	-25.2	-5.5	-9	1
rank	translation	feature vector			

Discriminative training



How to? (sec. 9.3)

5. Try to adjust the weights to bring the best candidates in (4) towards top of list.

- No analytic solution
  - We can't differentiate a function and find zero values
- Take 1: try systematically, say
  - λ_{LM}= .1, .2, .3, …, .9
  - λ_{φ}= .1, .2, …, .9- λ_{LM}= λ_LM
  - λ_{D}= = .1, .2, …, 1- (λ_{LM}+ λ_{φ})
  - Too many values to try out
  - Small changes in λs, large effect on result:
    - The steps are too large

Take 2: Powell search

- Optimize one λ, say λ_{LM}, keeping the other fixed.
- With this value for λ_{LM}, optimize the next λ, etc.
- A method for searching for the best value for each λ

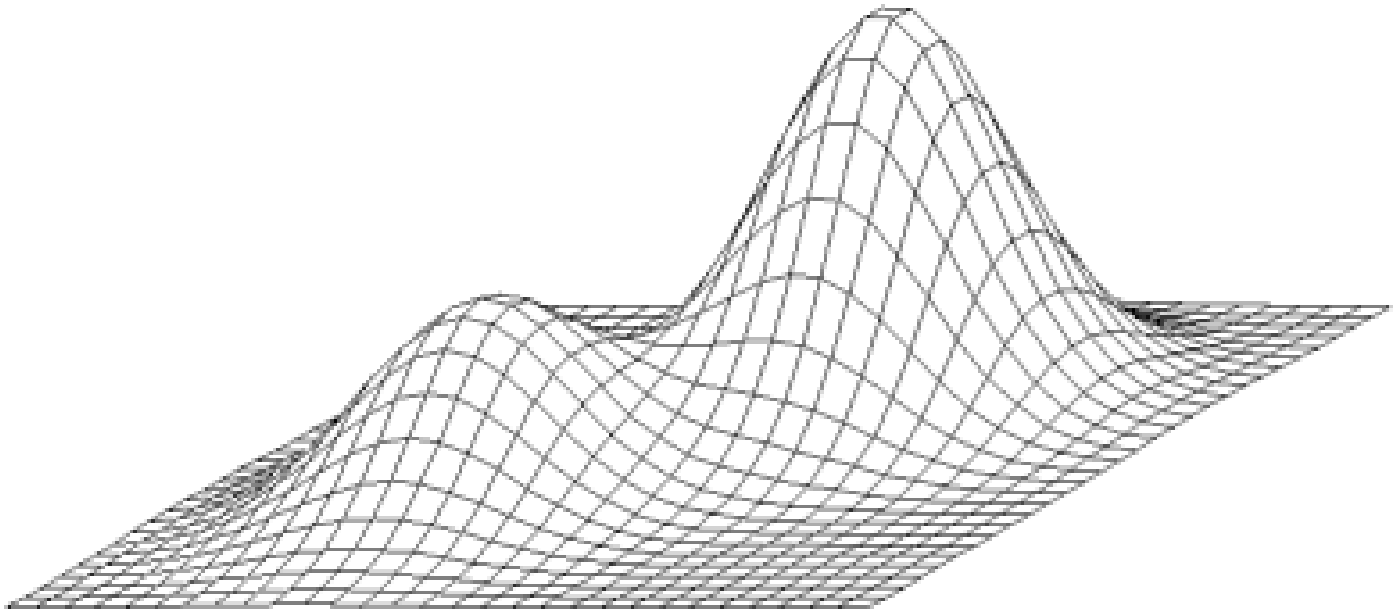
Take 3:

- (alternative) Simplex algorithm
- Variants of “hill climbing”

- Read sec 9.3
  - Not the details of
    - Finding threshold points
    - Combining threshold points

    in sec 9.3.2
  - Not 9.3.3 Simplex

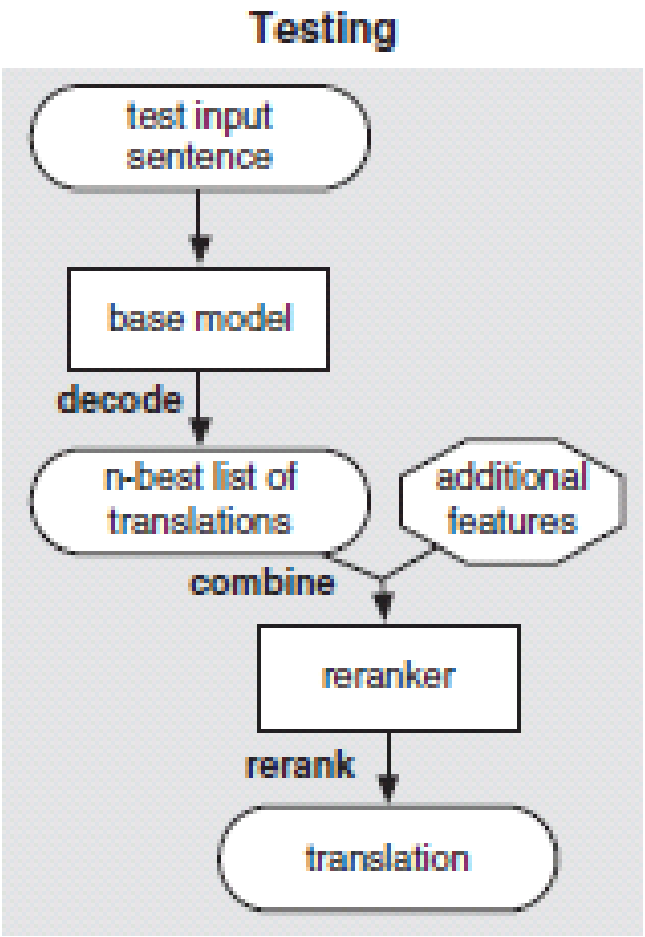
Will the solutions be global?



Today

- Preparing bitext
- Parameter tuning
- Reranking
- Some linguistic issues

Reranking model for SMT



Translation in two steps

1. Use a STMT-system as we have seen so far
   - Output: A set of translation candidates
2. Use the reranker to rank the outputs and select between them
   - Discriminative model

Statistical models

Generative model

- Construct solutions and assign them probabilities
- Examples
  - PCFG:
    - Assign trees
    - Probabilities to the trees
  - HMM-tagger
    - Produce tag-sequences w/probabilities
  - The translation models, both IBM and phrase-based

Discriminative model

- Starts with a set of solutions
- Select between them on the basis of a statistical score
- Example:
  - Malt parser

The reranker

- Consider it as a classification problem
- Supervised learning
- Training material:
  - A set of sentences in source language
  - One or more reference translations of these
  - Output of the STMT-system for these sentences
- Choose learning goal:
  - (A way to evaluate the output sentences)
  - Typically modified BLEU (or NIST) score

The reranker - learning

- Choose features
- Choose a learning strategy:
  - Naïve Bayes
  - Maximum entropy
    - (INF5830)
    - Skip here 9.2.4
  - Etc.
- The result is a ranker which - if we have succeeded -will return a reordered list where the top element has a better score than the top element before reranking
- Observe:
  - This is machine learning
  - The resulting reranker will not always improve the results

Reranking vs Tuning

- What is the difference between
  - Tuning and
  - Reranking?

Reranking vs Tuning

- Tuning is part of the training of the original STMT-system.
- Tuning is applied to make an optimal translation system

- Reranking is part of a full MT-pipeline..
- It is part of the decoding.
- It is applied to each sentence after the beam decoder has made an *n*-best list.

Today

- Preparing bitext
- Parameter tuning
- Reranking
- Some linguistic issues

SMT + Linguistic Information

- Transliteration
- Compounds
- Names
- Morphology
- Word order and syntax

Translating numbers

- How should a STMT system translate 12356?
- Most likely: It hasn't seen the number before.
- But since it translates into itself:
  - one possible solution:
    - Remove the number before translation, replace it by some dummy (say NNNN)
    - Insert the number after translation
  - (A default of not translating unseen tokens would give the same result)

Transliteration

Norwegian, German	English
12,3	12.3
12 500 or 12.500	12 500 or 12,500

- But the numbers are not exactly the same even in closely related languages

- Solution: Specific modules
  - Translate specific phenomena
  - Taken out of the regular SMT

Transliteration - names

- How should names in Japanese or Russian be spelled in English?
  - The book describes recipes for doing this statistically(sec. 10.1)
  - We will not consider this

Compounds

- German, Norwegian
  - samhandlingsreform
  - snøskredfare
- English
  - word segmentation
  - cruising speed
- Phrase-based SMT better than word-based
- New compounds/sparse data still a challenge
- Compound splitting in source language may help
- But how to put Humpty Dumpty together again?
  - (when going to German or Norwegian)

Compounds in LOGON

- A set of possible templates:
  - N_1 N_2 → Ad_1 N_2
  - N_1 N_2 → N_2 of N_1
  - etc.
- Generate all possible transfer rules
  - from basic transfer rules for N_1 and N_2
  - from dictionary
  - (order of n*n)
- Filter against monolingual target corpus
- Possible improvement: Turn into a probabilistic model

Translating names

- Which names should be translated and which not?
- How?
  - Oslofjorden
  - Rondane
  - Statens lånekasse for utdanning
  - Sognefjellesveien

- Sognefjellsveien
- the Sognefjellsveien
- Sognefjell Road
- the Sognefjell Road
- Sogn Mountain Road
- the Sogn Mountain Road
- the Parish Mountain Road

Translating names

- Difficult to specify a recipe that fits all.