SMT/NMT: OSM and OOV

Alexander Fraser

Lecture Today

- I will present two pieces of further work by the group today
  - **Operation Sequence Model (OSM)**
    - **Competitor to PBSMT**
    - **Widely combined with PBSMT (as a set of feature functions)**
  - **Using Bilingual Word Embeddings for Domain Adaptation of NMT**
    - **This involves modeling Out-Of-Vocabulary (OOV) words**
    - **OOV words are words we want to translate that do not occur in the parallel training data**
- Next Tuesday, I will present the first of four exercises (which you will do offline)
- The week of ACL (July 5th to July 10th) everything will be cancelled
  - BTW: ACL is online and less expensive this year!
- End of the semester: review, mini practice exam, online exam

A Short Introduction to the Operation Sequence Model

Alexander Fraser
(slides mostly from Nadir Durrani)

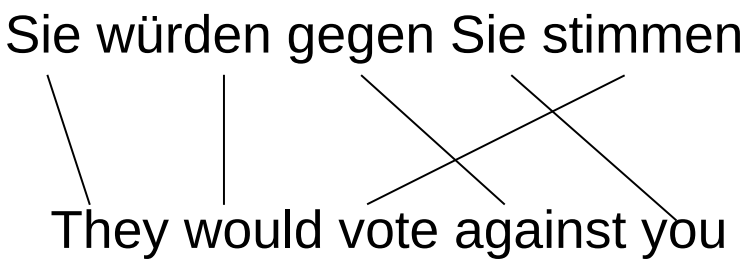
Improving the modeling of syntax in SMT

- Novel model: **Operation Sequence Model**
- New model overcoming problems with phrase-based model
- Joint work with Durrani and Schmid
  - Durrani's 2013 PhD thesis won GSCL prize for best CL/NLP thesis in Germany from 2011-2013
  - Numerous papers at *ACL conferences

Motivation: Long Distance Reordering in German-to-English SMT

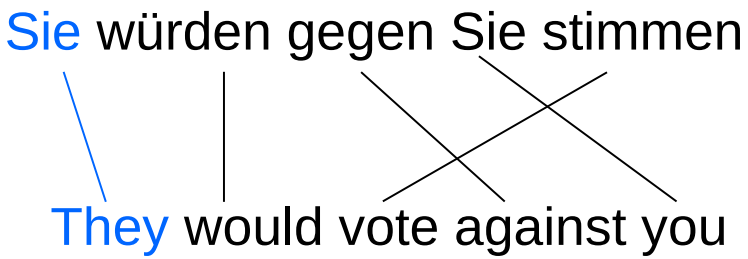
- Er hat ein Buch gelesen → He read a book
- Er hat gestern Nachmittag mit seiner kleinen Tochter, die aufmerksam zugehört hat, und seinem Sohn, der lieber am Computer ein Videogame gespielt hätte, ein spannendes Buch gelesen
- We want a model that
  - captures ”hat … gelesen = read”
  - captures the generalization that an arbitrary amount of stuff can occur between hat and gelesen (in the so-called "mittelfeld")
  - is a simple left-to-right model

Example



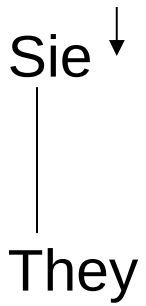
- Rules:
  - Simultaneous generation of bilingual sentence pair through a sequence of operations
  - Generation is done in order of the target (English) sentence
  - Idea behind operations: either Translate or Reorder

Example

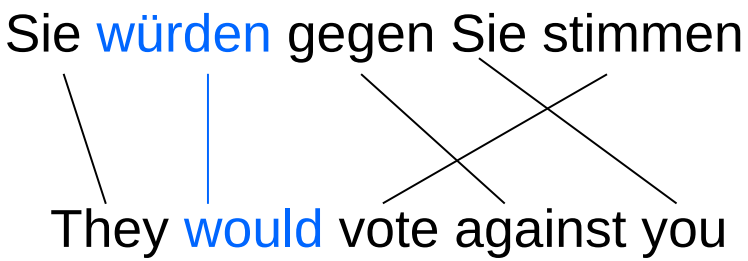


Operations

o_1: Generate (Sie – They)

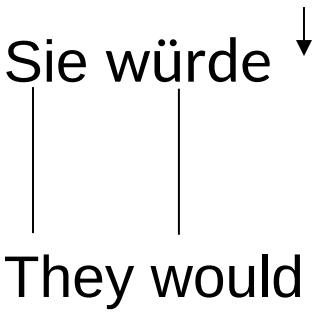


Example

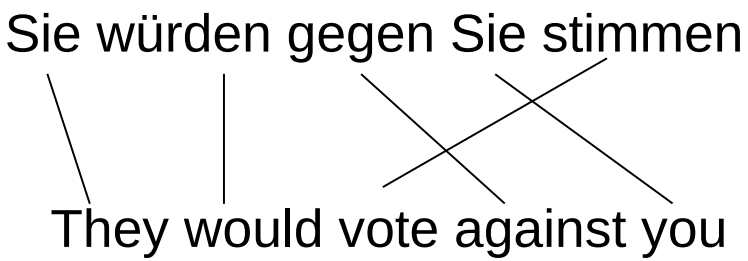


Operations

- o_1 Generate (Sie, They)
- o_2 Generate (würden, would)

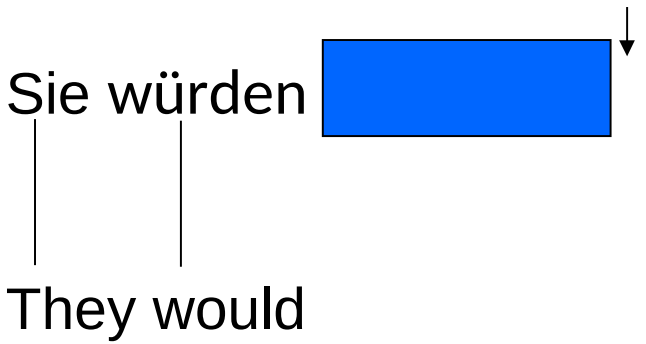


Example

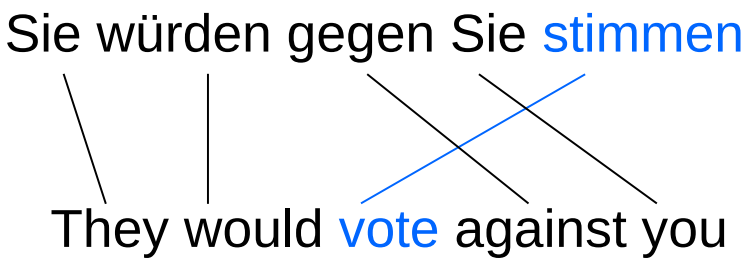


Operations

- o_1 Generate (Sie, They)
- o_2 Generate (würden, would)
- o_3 Insert Gap

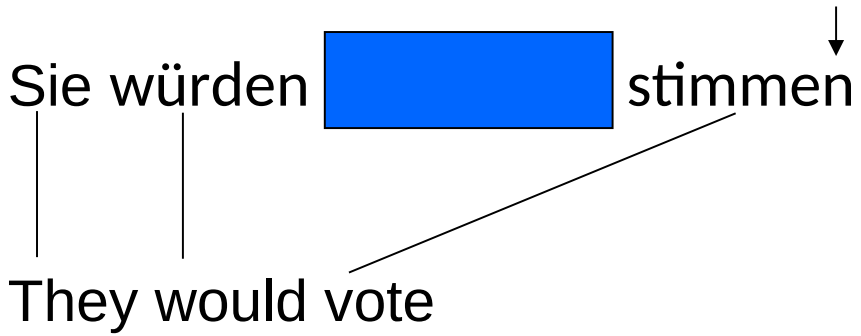


Example

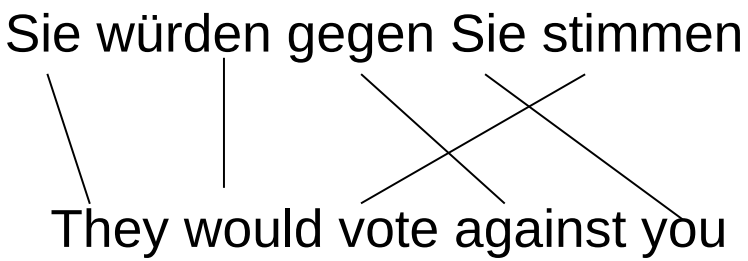


Operations

- o_1 Generate (Sie, They)
- o_2 Generate (würden, would)
- o_3 Insert Gap
- o_4 Generate (stimmen, vote)

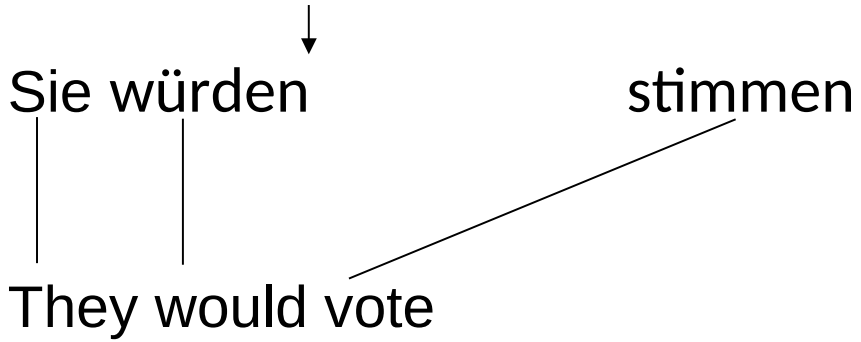


Example

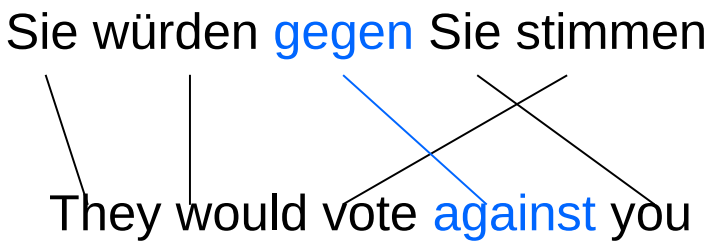


Operations

- o_1 Generate (Sie, They)
- o_2 Generate (würden, would)
- o_3 Insert Gap
- o_4 Generate (stimmen, vote)
- o_5 Jump Back (1)

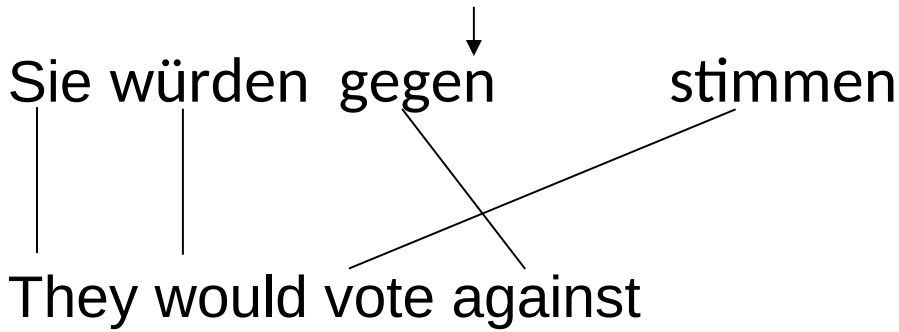


Example

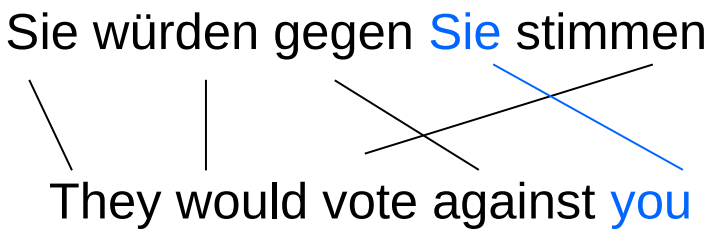


Operations

- o_1 Generate (Sie, They)
- o_2 Generate (würden, would)
- o_3 Insert Gap
- o_4 Generate (stimmen, vote)
- o_5 Jump Back (1)
- o_6 Generate (gegen, against)

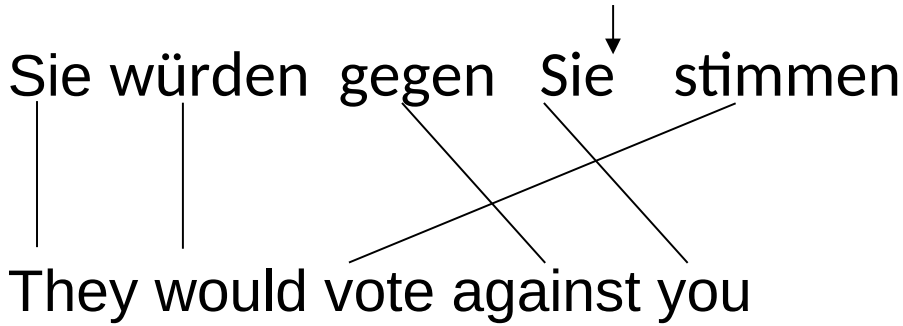


Example



Operations

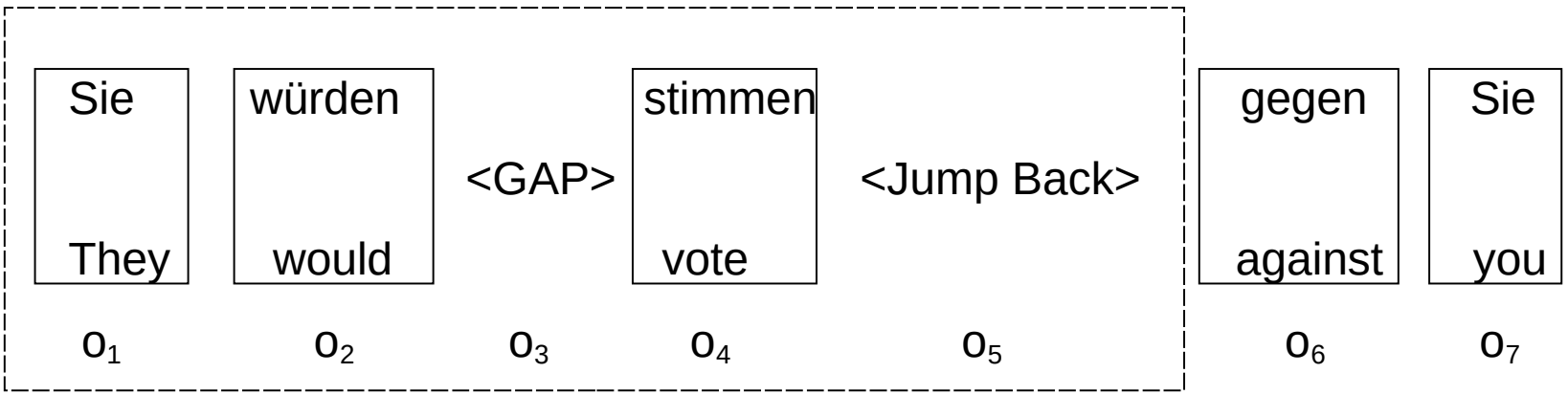
- o_1 Generate (Sie, He)
- o_2 Generate (würde, would)
- o_3 Insert Gap
- o_4 Generate (stimmen, vote)
- o_5 Jump Back (1)
- o_6 Generate (gegen, against)
- o_7 Generate (Sie, you)



Model

- Joint probability model over operation sequences

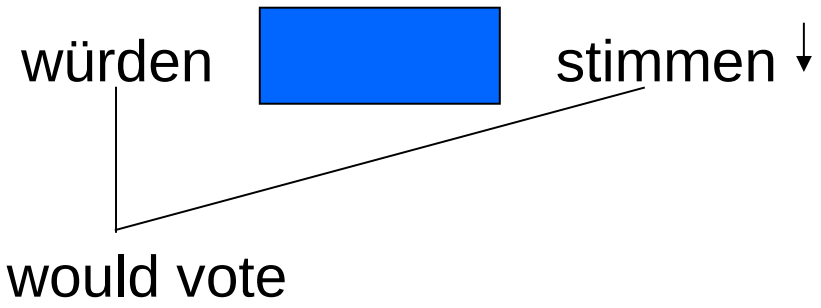
$$p_{osm}(F, E, A) = p(o_1^J) = \prod_{j=1}^{J} p(o_j | o_{j-n+1}, \ldots, o_{j-1})$$



Context window: 9-gram model

Example of a learned pattern

- Operations
  - Generate (würden, would)
  - Insert Gap
  - Generate (stimmen, vote)



- Can generalize to
  - Die Menschen würden dafür stimmen
  - Die Menschen würden gegen meine Außenpolitik stimmen
  - Die Menschen würden für die Legalisierung der Abtreibung in Kanada stimmen

- Equivalent to hierarchical phrase “würden X stimmen – would vote X”

- Gaps can be created recursively
  - Multiple gaps can occur simultaneously

Results and outlook

- Operation sequence model overcomes problems with the phrase-based model
  - Models minimal translation units well that are highly dependent on one another but not contiguous, unlike phrase-based
  - Reordering is integrated with lexical generation
- Operation sequence model is available as a feature function in the latest version of Moses (open-source statistical machine translation toolkit)
- The model is widely acknowledged to lead to actual improvements in systems in large scale evaluation campaigns such as WMT and IWSLT
  - Standardly used in all competitive PBSMT systems
- What I didn't talk about: our related work on synchronous grammars, particularly Synchronous Context-Free Grammars (SCFG), Synchronous Tree Substitution Grammars (STSG)

- Thank you!