Word Alignment

Hasso Plattner Institut
IT Systems Engineering | Universität Potsdam

Dr. Mariana Neves
(adapted from the original slides
of Prof. Philipp Koehn)

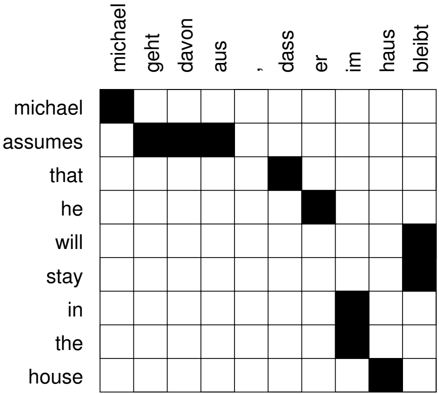
November 14th, 2016

Overview

- Further discussion on word alignment, such as problems and quality measurement
- Present a method on word alignment based on the IBM models

Word Alignment

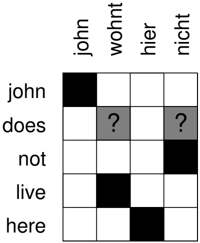
Given a sentence pair, which words correspond to each other?



Word alignment

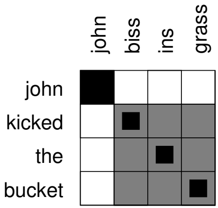
- It does not need to be one-by-one.
- Words can have multiple or no alignment points.

Word Alignment?



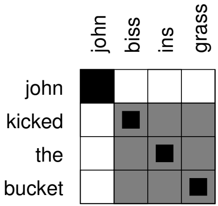
Is the English word does aligned to
the German wohnt (verb) or nicht (negation) or neither?

Word Alignment?



How do the idioms kicked the bucket and biss ins grass match up?
Outside this exceptional context, bucket is never a good translation for grass

Word Alignment?



The better solution here is a phrasal alignment!

Word alignment

- Sure alignments:
  - John to John
- Possible alignments:
  - kicked to biss
  - the to im
  - bucket to Grass

Measuring Word Alignment Quality

- Manually align corpus with *sure* (S) and *possible* (P) alignment points ($S \subseteq P$)
- Alignment Error Rate (AER): common metric for evaluation word alignments

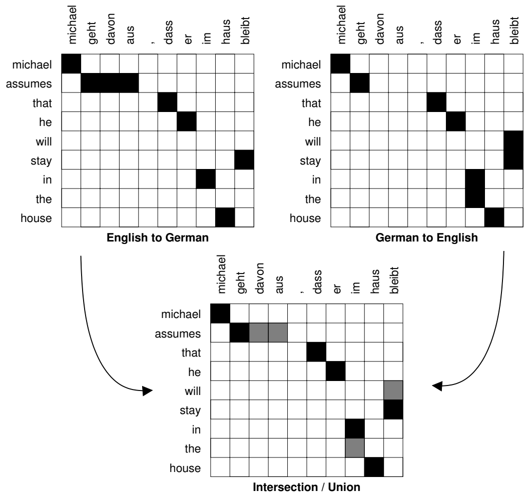
$$\text{AER}(S, P; A) = 1 - \frac{|A \cap S| + |A \cap P|}{|A| + |S|}$$

- AER $= 0$: alignment A matches all sure, any number of possible alignment points

Word Alignment with IBM Models

- IBM Models create a **many-to-one** mapping
  - words are aligned using an alignment function
  - a function may return the same value for different input (one-to-many mapping)
  - a function cannot return multiple values for one input (no many-to-one mapping)
- Real word alignments have **many-to-many** mappings

Symmetrizing Word Alignments



- Intersection of GIZA++ bidirectional alignments

Symmetrizing Word Alignments

- The **intersection** usually contains good alignment points (high precision), but not all of them.
- The **union** usually contains most of the desired align points (high recall), but also faulty points.

- We want to explore the space between the two extremes:
  - Take the all alignment points in the intersection (reliable).
  - Add some of the points from the union (neighboring candidates), incrementally.

Growing heuristic

grow-diag-final(e2f,f2e)

neighboring = {(-1,0),(0,-1),(1,0),(0,1),(-1,-1),(-1,1),(1,-1),(1,1)}
alignment A = intersect(e2f,f2e); grow-diag(); final(e2f); final(f2e);

grow-diag()

while new points added **do**
    for all English word $e \in [1...e_n]$, foreign word $f \in [1...f_n]$, $(e, f) \in A$ **do**
        for all neighboring alignment points $(e_{\text{new}}, f_{\text{new}})$ **do**
            if (e_{new} unaligned OR f_{new} unaligned) AND $(e_{\text{new}}, f_{\text{new}}) \in$ union(e2f,f2e) **then**
                add $(e_{\text{new}}, f_{\text{new}})$ to A
            end if
        end for
    end for
end while

final()

for all English word $e_{\text{new}} \in [1...e_n]$, foreign word $f_{\text{new}} \in [1...f_n]$ **do**
    if (e_{new} unaligned OR f_{new} unaligned) AND $(e_{\text{new}}, f_{\text{new}}) \in$ union(e2f,f2e) **then**
        add $(e_{\text{new}}, f_{\text{new}})$ to A
    end if
end for

Suggested reading

- Statistical Machine Translation, Philipp Koehn (section 4.5).