

# Evaluating Word Order Recursively over Permutation-Forests

Miloš Stanojević and Khalil Sima'an

October 25, 2014

## What is *wrong* with existing metrics related to word order

- ▶ BLEU and many others have **local view** of word order (n-gram window) which is not good for long distance reordering.

## What is *wrong* with existing metrics related to word order

- ▶ BLEU and many others have **local view** of word order (n-gram window) which is not good for long distance reordering.
- ▶ We need **better representation** that would allow a **global view** – permutations (LRscore, RIBES, FuzzyScore)

# What is *wrong* with existing metrics related to word order

- ▶ BLEU and many others have **local view** of word order (n-gram window) which is not good for long distance reordering.
- ▶ We need **better representation** that would allow a **global view** – permutations (LRscore, RIBES, FuzzyScore)
- ▶ Problem: **not hierarchical and not flexible**

# What is *wrong* with existing metrics related to word order

- ▶ BLEU and many others have **local view** of word order (n-gram window) which is not good for long distance reordering.
- ▶ We need **better representation** that would allow a **global view** – permutations (LRscore, RIBES, FuzzyScore)
- ▶ Problem: **not hierarchical and not flexible**
- ▶ **Permutation Trees (PETs)** might come handy

# What is *wrong* with existing metrics related to word order

- ▶ BLEU and many others have **local view** of word order (n-gram window) which is not good for long distance reordering.
- ▶ We need **better representation** that would allow a **global view** – permutations (LRscore, RIBES, FuzzyScore)
- ▶ Problem: **not hierarchical and not flexible**
- ▶ **Permutation Trees (PETs)** might come handy
- ▶ Our metric computes its score in a way **similar to PCFG** on these hierarchical structures

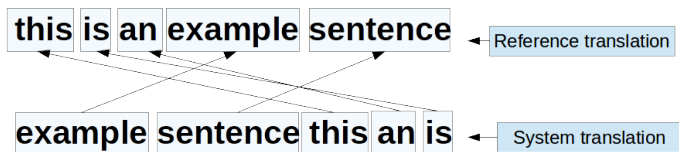
# Recursive metrics

this is an example sentence ← Reference translation

example sentence this an is ← System translation

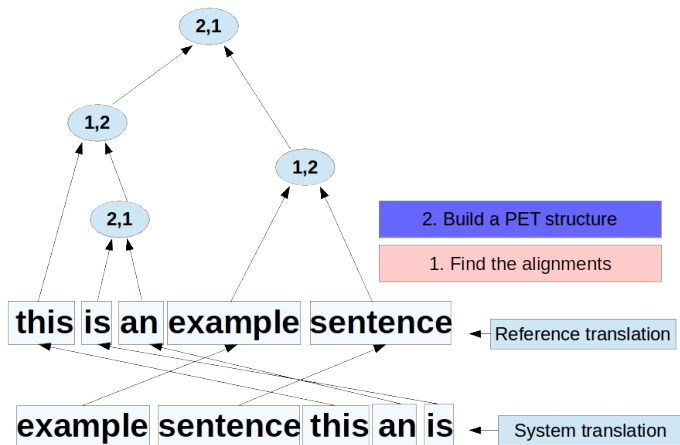
# Recursive metrics

1. Find the alignments

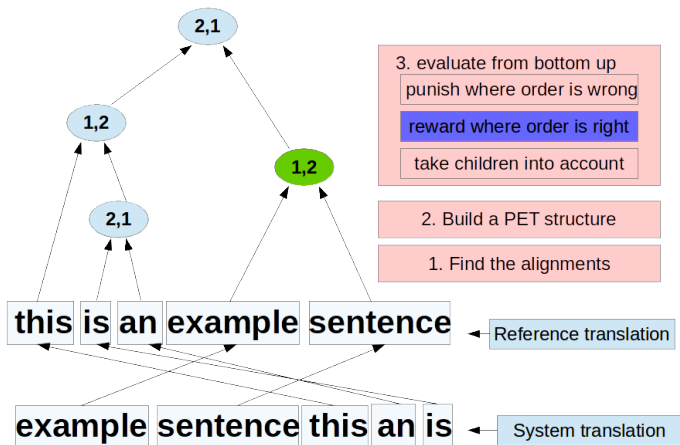




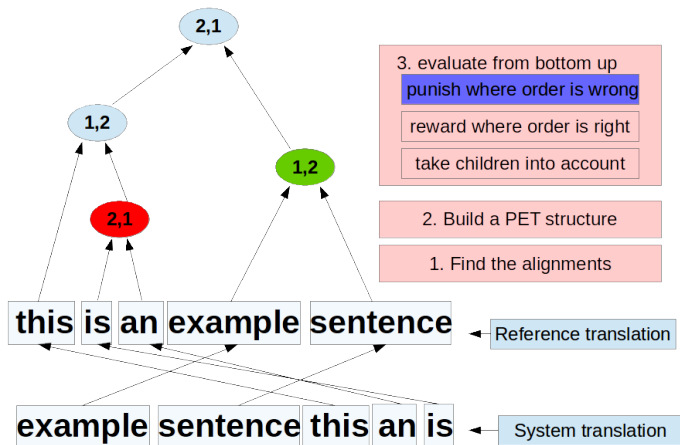
# Recursive metrics



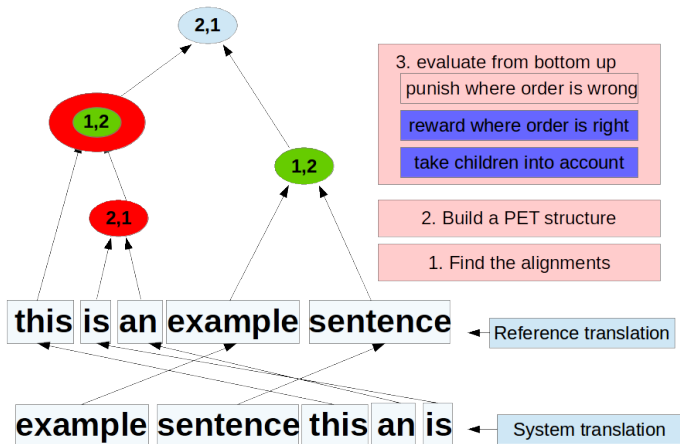
# Recursive metrics



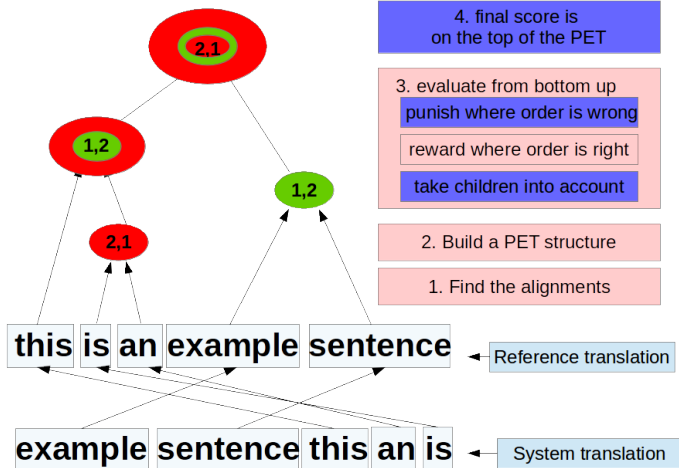
# Recursive metrics



# Recursive metrics



# Recursive metrics



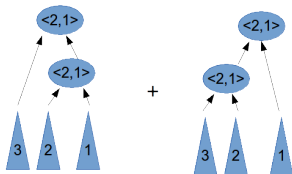
## $PETscore(\cdot)$ and $PEFscore(\cdot)$

$$PETscore(node) = \beta \text{ opScore}(node.op) + (1 - \beta) \sum_{c \in node.children} PETscore(c)$$

## $PETscore(\cdot)$ and $PEFscore(\cdot)$

$$PETscore(node) = \beta \text{ opScore}(node.op) + (1 - \beta) \sum_{c \in node.children} PETscore(c)$$

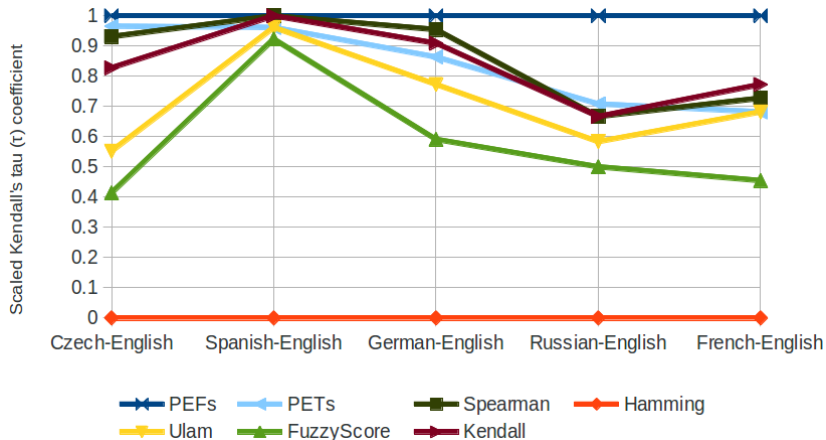
But, there might be (exponentially) many PETs for a single permutation!



$$PEFscore(\pi) = \frac{\sum_{t \in PEF(\pi)} PETscore(t)}{\#PETs}$$

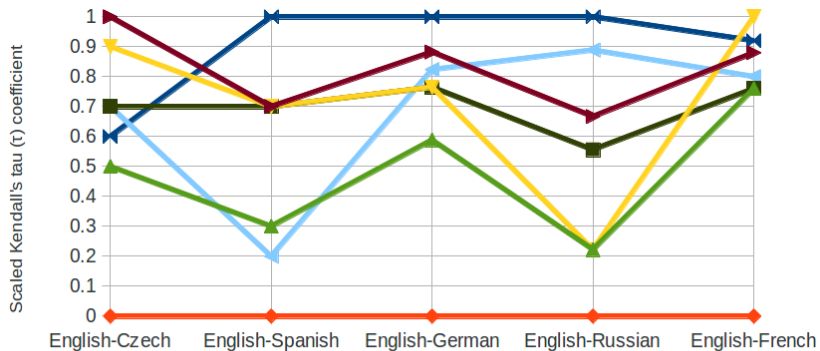
Can be efficiently computed with a version of Inside algorithm.

# Results into English (scaled Kendall sent level)





# Results out of English (scaled Kendall sent level)



PEFs    PETs    Spearman    Hamming  
Ulam    FuzzyScore    Kendall

# Conclusion

- ▶ Consider all factorizations (PEF)

# Conclusion

- ▶ Consider all factorizations (PEF)
- ▶ Do it hierarchically

# Conclusion

- ▶ Consider all factorizations (PEF)
- ▶ Do it hierarchically
- ▶ Metric is available online together with BEER

# Conclusion

- ▶ Consider all factorizations (PEF)
- ▶ Do it hierarchically
- ▶ Metric is available online together with BEER
- ▶ Come to see the poster

# Conclusion

- ▶ Consider all factorizations (PEF)
- ▶ Do it hierarchically
- ▶ Metric is available online together with BEER
- ▶ Come to see the poster
- ▶ Thank you