

Weighted Alignment Matrices for Statistical Machine Translation

Yang Liu, Tian Xia, Xinyan Xiao, and Qun Liu

Institute of Computing Technology,

Chinese Academy of Sciences

{yliu, xiatian, xiaoxinyan, liuqun}@ict.ac.cn



中国科学院
INSTITUTE OF COMPUTING TECHNOLOGY

Word Alignment

economy			●	
's		●		
China	●			
of				
development				●
the				
	zhongguo	de	jingji	fazhan

Alignment Consistency

economy			●	
's		●		
China	●			
of				
development				●
the				
	zhongguo	de	jingji	fazhan

Och and Ney (2004)

consistent



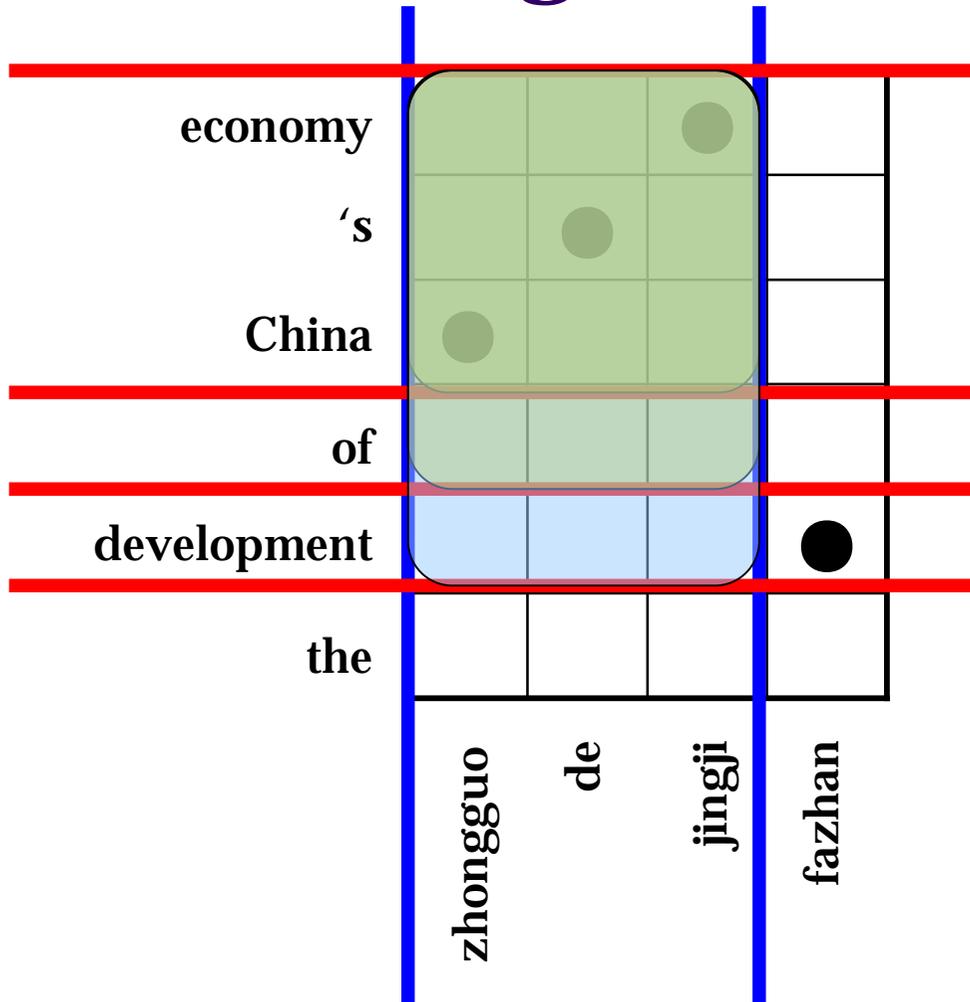
Alignment Consistency

economy			●	
's		●		
China	●			
of				
development				●
the				
	zhongguo	de	jingji	fazhan

Och and Ney (2004)

inconsistent

Extracting Phrase Pairs from 1-best Alignments



Och and Ney (2004)

Zhongguo de jingji ||| China 's economy

Zhongguo de jingji ||| of China 's economy

Extracting Phrase Pairs from N-best Alignments

Venugopal et al., (2008)

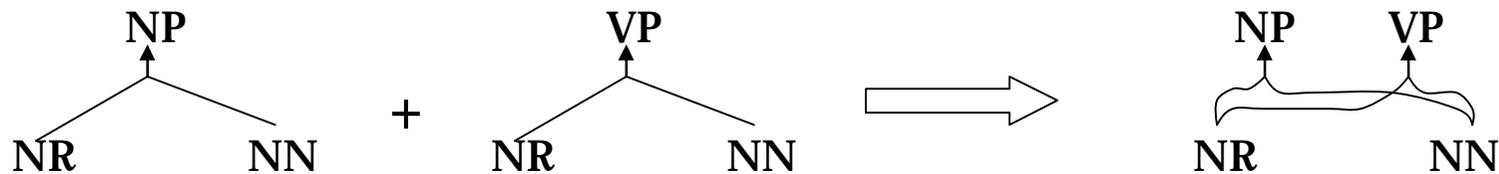
economy			●	
's				
China	●			
of		●		
development				●
the				
	zhongguo	de	jingji	fazhan

economy			●	
's		●	●	
China	●			
of				
development				●
the				●
	zhongguo	de	jingji	fazhan

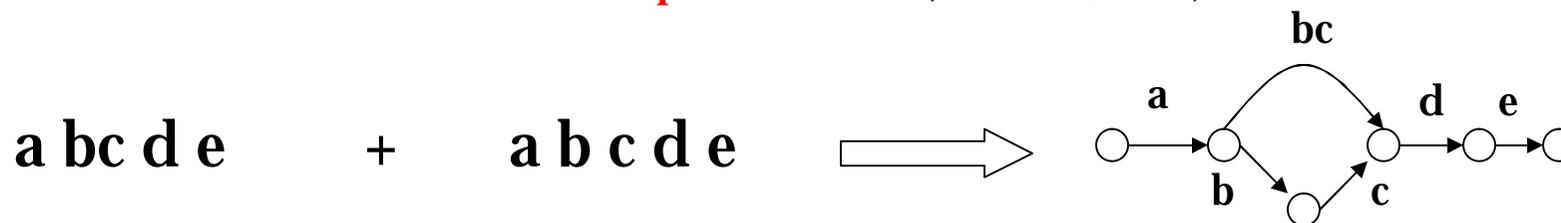
This Work

- | We propose ***weighted alignment matrix***, a compact representation of multiple word alignments.
- | We design a new method for extracting phrase pairs from weighted alignment matrices.

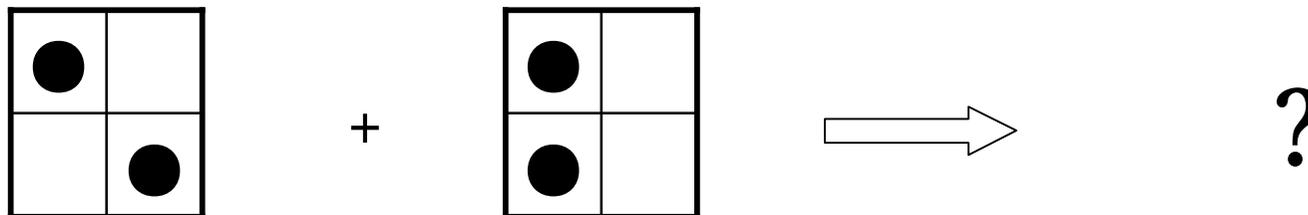
Compact Representation



from 1-best tree to **packed forest** (Mi et al., 2008)



from 1-best segmentation to **lattice** (Dyer et al., 2008)



from 1-best alignment to **what?**



Weighted Alignment Matrix



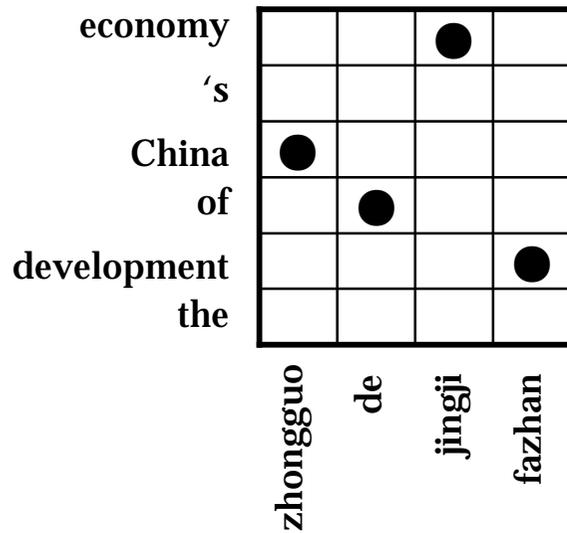
Neo: What is the Matrix?

Morpheus: The Matrix is officially called weighted alignment matrix, which stores link probabilities to indicate how well two words are aligned.

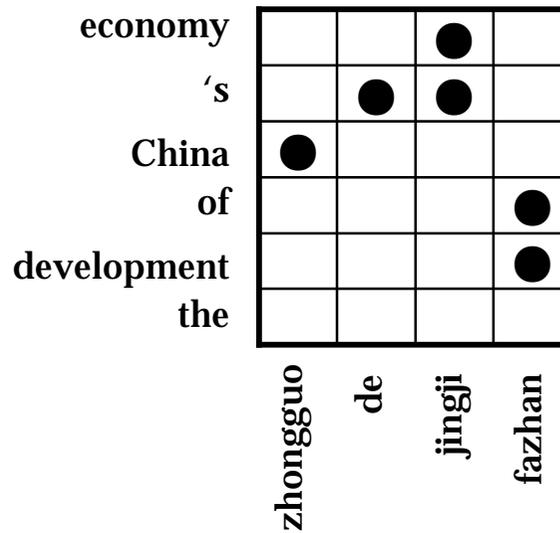


economy	0.0	0.0	1.0	0.0
's	0.0	0.4	0.4	0.0
China	1.0	0.0	0.0	0.0
of	0.0	0.6	0.0	0.4
development	0.0	0.0	0.0	1.0
the	0.0	0.0	0.0	0.0
	zhongguo	de	jingji	fazhan

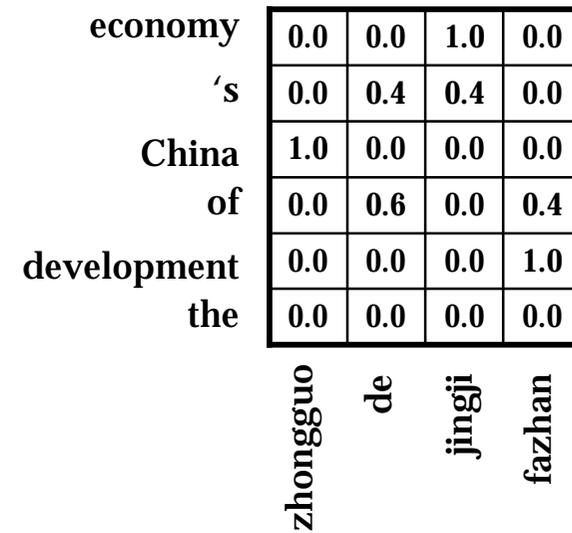
Estimating Link Probabilities from N-best List



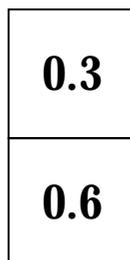
0.6



0.4



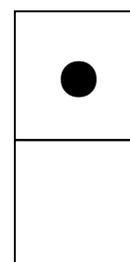
Alignment Probability



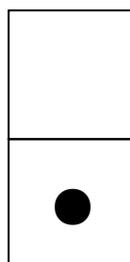
There are $2^{(2*1)}=4$ possible alignments



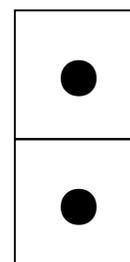
$$(1-0.3) \times (1-0.6)=0.28$$



$$0.3 \times (1-0.6)=0.12$$



$$(1-0.3) \times 0.6=0.42$$



$$0.3 \times 0.6=0.18$$



Collecting Fractional Count

economy	0.0	0.0	1.0	0.0
's	0.0	0.4	0.4	0.0
China	1.0	0.0	0.0	0.0
of	0.0	0.6	0.0	0.4
the	0.0	0.0	0.0	1.0
	zhongguo	de	jingji	fazhan

- | Suppose we extract a phrase pair “zhongguo de ||| of China” from the example;
- | There are $2^{(4*6)}=16,777,216$ possible alignments;
- | How many alignments from which this phrase pair can be extracted?
- | What is **the sum of probabilities** of such alignments?

Inside Probability

economy	0.0	0.0	1.0	0.0
's	0.0	0.4	0.4	0.0
China	1.0	0.0	0.0	0.0
of	0.0	0.6	0.0	0.4
	0.0	0.0	0.0	1.0
the	0.0	0.0	0.0	0.0
	zhongguo	de	jingji	fazhan

1. There must be at least one word inside one phase aligned to a word inside the other phrase

$$1.0 - (1.0-1.0) \times (1.0-0) \times (1.0-0) \times (1.0-0.6) = 1.0$$

no words inside are aligned

Outside Probability

economy	0.0	0.0	1.0	0.0
's	0.0	0.4	0.4	0.0
China	1.0	0.0	0.0	0.0
of	0.0	0.6	0.0	0.4
the	0.0	0.0	0.0	1.0
	zhongguo	de	jingji	fazhan

2. No words inside one phrase can be aligned to a word outside the other phrase

$$(1.0-0.4) \times (1.0-0.4)=0.36$$

no words outside are aligned

Fractional Count

economy	0.0	0.0	1.0	0.0
's	0.0	0.4	0.4	0.0
China	1.0	0.0	0.0	0.0
of	0.0	0.6	0.0	0.4
	0.0	0.0	0.0	1.0
the	0.0	0.0	0.0	0.0
	zhongguo	de	jingji	fazhan

How well the phrase pair is consistent with word alignment distribution?

$$1.0 \times 0.36 = 0.36$$

Extracting Phrase Pairs

economy	0.0	0.0	1.0	0.0
's	0.0	0.4	0.4	0.0
China	1.0	0.0	0.0	0.0
of	0.0	0.6	0.0	0.4
development	0.0	0.0	0.0	1.0
the	0.0	0.0	0.0	0.0
	zhongguo	de	jingji	fazhan

Extracting Phrase Pairs

length=1

economy	0.0	0.0	1.0	0.0
's	0.0	0.4	0.4	0.0
China	1.0	0.0	0.0	0.0
of	0.0	0.6	0.0	0.4
development	0.0	0.0	0.0	1.0
the	0.0	0.0	0.0	0.0
	zhongguo	de	jingji	fazhan

$$0.6 \times 0 = 0$$

Extracting Phrase Pairs

length=1

economy	0.0	0.0	1.0	0.0
's	0.0	0.4	0.4	0.0
China	1.0	0.0	0.0	0.0
of	0.0	0.6	0.0	0.4
development	0.0	0.0	0.0	1.0
the	0.0	0.0	0.0	0.0
	zhongguo	de	jingji	fazhan

$$1.0 \times 0.24 = 0.24$$

Extracting Phrase Pairs

length=1

economy	0.0	0.0	1.0	0.0
's	0.0	0.4	0.4	0.0
China	1.0	0.0	0.0	0.0
of	0.0	0.6	0.0	0.4
development	0.0	0.0	0.0	1.0
the	0.0	0.0	0.0	0.0
	zhongguo	de	jingji	fazhan

$$0.4 \times 0 = 0$$

Extracting Phrase Pairs

length=2

economy	0.0	0.0	1.0	0.0
's	0.0	0.4	0.4	0.0
China	1.0	0.0	0.0	0.0
of	0.0	0.6	0.0	0.4
development	0.0	0.0	0.0	1.0
the	0.0	0.0	0.0	0.0
	zhongguo	de	jingji	fazhan

$$0.6 \times 0 = 0$$

Extracting Phrase Pairs

length=2

economy	0.0	0.0	1.0	0.0
's	0.0	0.4	0.4	0.0
China	1.0	0.0	0.0	0.0
of	0.0	0.6	0.0	0.4
development	0.0	0.0	0.0	1.0
the	0.0	0.0	0.0	0.0
	zhongguo	de	jingji	fazhan

$$1.0 \times 0.36 = 0.36$$

Extracting Phrase Pairs

length=2

economy	0.0	0.0	1.0	0.0
's	0.0	0.4	0.4	0.0
China	1.0	0.0	0.0	0.0
of	0.0	0.6	0.0	0.4
development	0.0	0.0	0.0	1.0
the	0.0	0.0	0.0	0.0
	zhongguo	de	jingji	fazhan

$$1.0 \times 0.24 = 0.24$$

Extracting Phrase Pairs

length=2

economy	0.0	0.0	1.0	0.0
's	0.0	0.4	0.4	0.0
China	1.0	0.0	0.0	0.0
of	0.0	0.6	0.0	0.4
development	0.0	0.0	0.0	1.0
the	0.0	0.0	0.0	0.0
	zhongguo	de	jingji	fazhan

$$0.4 \times 0 = 0$$

Extracting Phrase Pairs

length=3

economy	0.0	0.0	1.0	0.0
's	0.0	0.4	0.4	0.0
China	1.0	0.0	0.0	0.0
of	0.0	0.6	0.0	0.4
development	0.0	0.0	0.0	1.0
the	0.0	0.0	0.0	0.0
	zhongguo	de	jingji	fazhan

$$0.6 \times 0 = 0$$

Extracting Phrase Pairs

length=3

economy	0.0	0.0	1.0	0.0
's	0.0	0.4	0.4	0.0
China	1.0	0.0	0.0	0.0
of	0.0	0.6	0.0	0.4
development	0.0	0.0	0.0	1.0
the	0.0	0.0	0.0	0.0
	zhongguo	de	jingji	fazhan

$$1.0 \times 0 = 0$$

Extracting Phrase Pairs

length=3

economy	0.0	0.0	1.0	0.0
's	0.0	0.4	0.4	0.0
China	1.0	0.0	0.0	0.0
of	0.0	0.6	0.0	0.4
development	0.0	0.0	0.0	1.0
the	0.0	0.0	0.0	0.0
	zhongguo	de	jingji	fazhan

$$1.0 \times 0.36 = 0.36$$

Extracting Phrase Pairs

length=3

economy	0.0	0.0	1.0	0.0
's	0.0	0.4	0.4	0.0
China	1.0	0.0	0.0	0.0
of	0.0	0.6	0.0	0.4
development	0.0	0.0	0.0	1.0
the	0.0	0.0	0.0	0.0
	zhongguo	de	jingji	fazhan

$$1.0 \times 0 = 0$$

Absolute Threshold Pruning

economy	0.0	0.0	1.0	0.0
's	0.0	0.4	0.4	0.0
China	1.0	0.0	0.0	0.0
of	0.0	0.6	0.0	0.4
development	0.0	0.0	0.0	1.0
the	0.0	0.0	0.0	0.0
	zhongguo	de	jingji	fazhan

candidate translations of “zhongguo de”:

of China	0.36	t=0.3
of China 's	0.36	
China 's	0.24	
China	0.24	



Estimating a Lexicon

economy	0.0	0.0	1.0	0.0
's	0.0	0.4	0.4	0.0
China	1.0	0.0	0.0	0.0
of	0.0	0.6	0.0	0.4
development	0.0	0.0	0.0	1.0
the	0.0	0.0	0.0	0.0
	zhongguo	de	jingji	fazhan

We need first to learn a lexicon by relative frequencies from the training corpus:

count(zhongguo, China)	1.0
count(de, of)	0.6
count(de, NULL)	0.24
count(NULL, of)	0.24

Calculating Lexical Weights

economy	0.0	0.0	1.0	0.0
's	0.0	0.4	0.4	0.0
China	1.0	0.0	0.0	0.0
of	0.0	0.6	0.0	0.4
development	0.0	0.0	0.0	1.0
the	0.0	0.0	0.0	0.0
	zhongguo	de	jingji	fazhan

the probability that “of” and
“de” is aligned

$$(0.6 \times w(\text{of}|\text{de}) + 0.4 \times w(\text{of}|\text{NULL})) \times$$

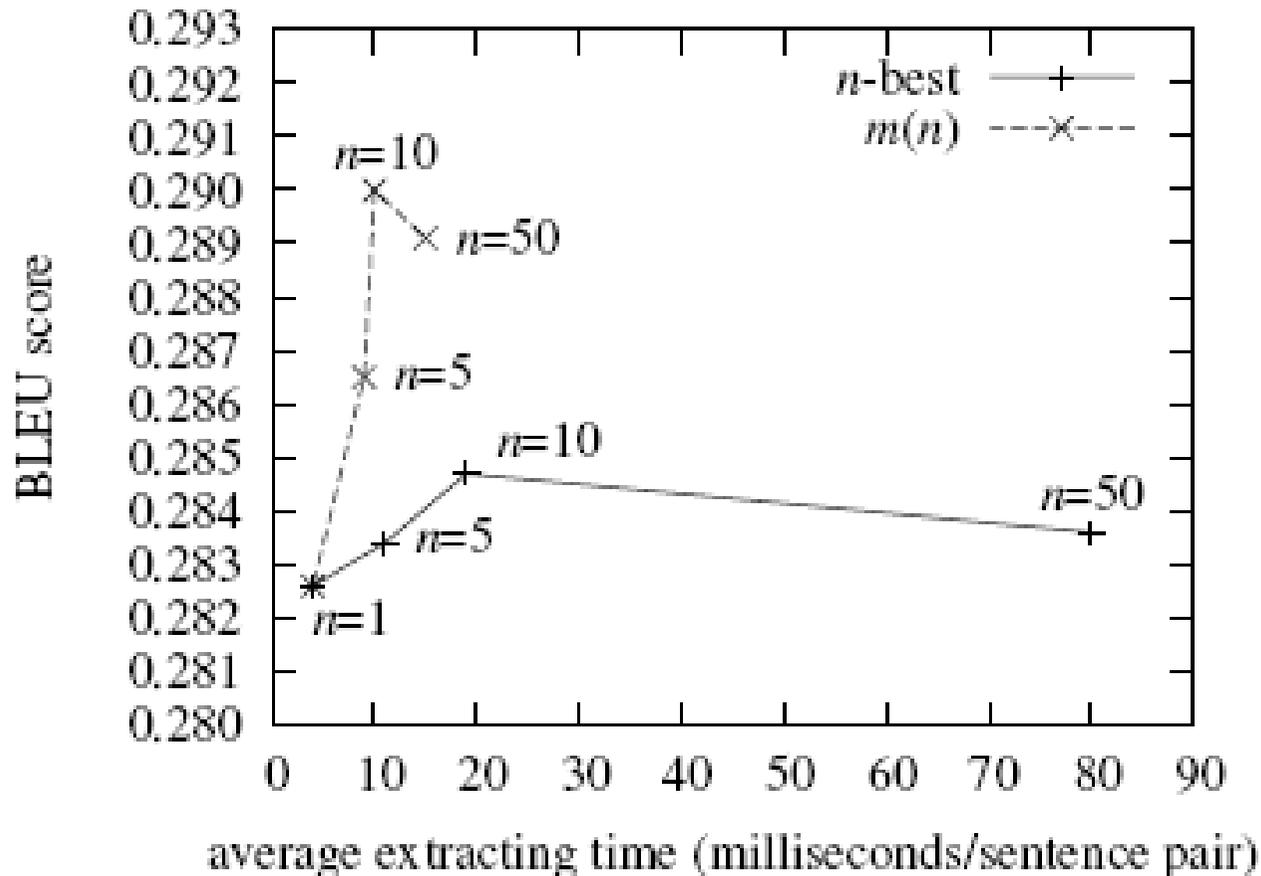
$$(1.0 \times w(\text{China}|\text{zhongguo}) + 0 \times w(\text{China}|\text{NULL}))$$

the probability
that “China” is
not aligned

Setup

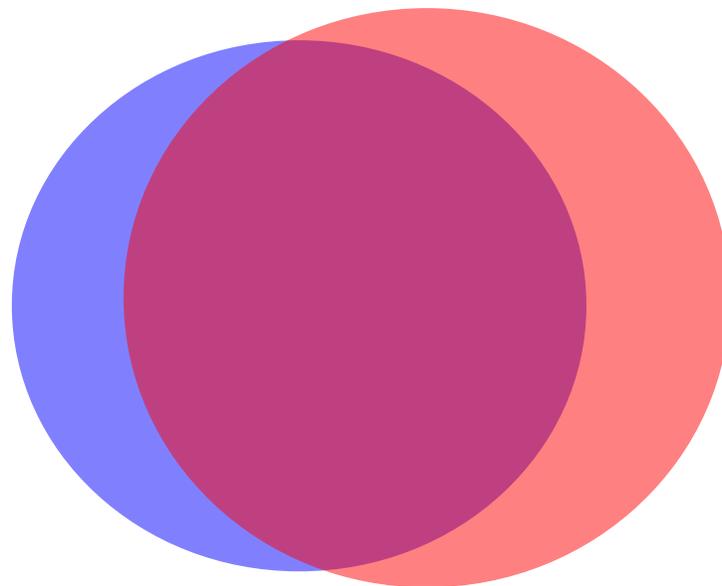
- | **Training set: FBIS (6.9M+8.9M)**
- | **Language model: 4-gram model trained on GIGAWORD Xinhua portion**
- | **Development set: NIST 2002 Chinese-to-English**
- | **Test set: NIST 2005 Chinese-to-English**
- | **Metric: case-insensitive BLEU4**
- | **Decoder: Moses (simple distance-based reordering)**

N-best Lists Vs. Weighted Matrices



Comparison of Probability Estimation

10-best list



matrix

method	intersection		complement		all	
	phrases	BLEU	phrases	BLEU	phrases	BLEU
10-best	4.58M	28.35	1.55M	12.32	6.13M	28.47
m(10)	4.58M	28.90	1.76M	13.21	6.34M	29.01

Conclusion and Future Work

- | We propose a compact representation of multiple word alignments and show promising results on phrase-based systems.
- | Future directions
 - | Direct modeling of weighted matrices
 - | Algorithms for extracting rules with hierarchical structures



Please protect the
Matrix, thanks!