Machine Translation without Words through Substring Alignment

Graham Neubig[1,2,3], Taro Watanabe[2], Shinsuke Mori[1], Tatsuya Kawahara[1]

1

2

3 now at

Machine Translation

- Translate a source sentence **F** into a target sentence **E**
- F and E are strings of **words**

F = これ は ペン です
f1 f2 f3 f4

E = this is a pen
e1 e2 e3 e4

Sparsity Problems

- **Transliteration:** For proper names, change from one writing system to another

Sparsity Problems

- **Inflected** or **compound** words cause large vocabularies and sparsity

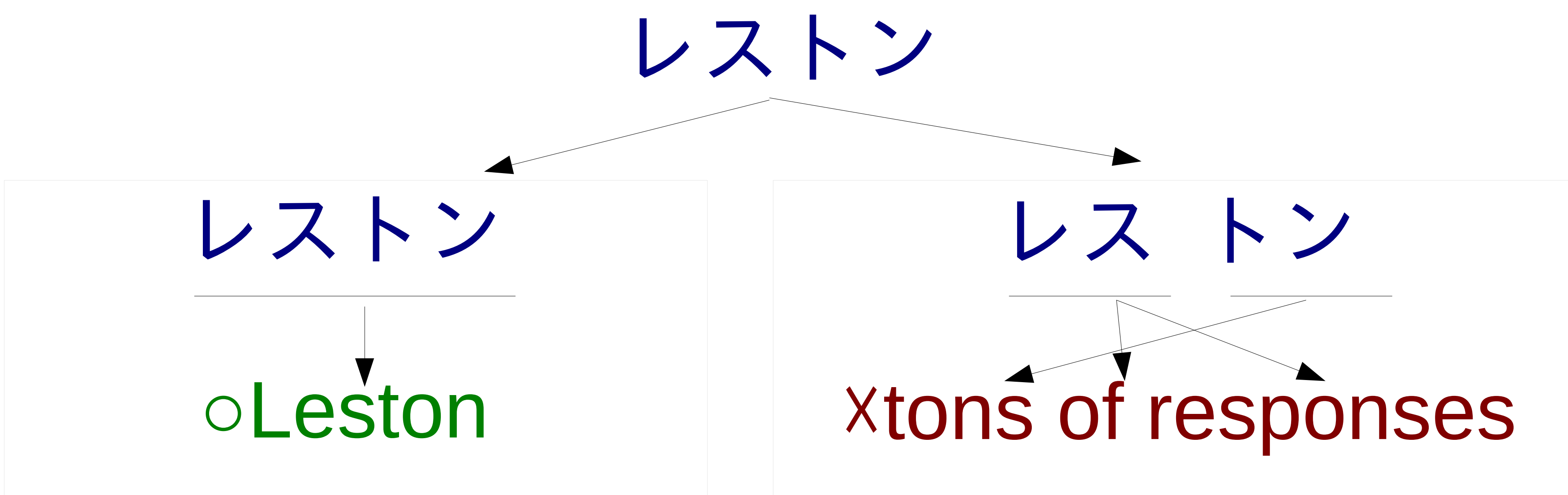
huolestumista

○concerned [elative]

☓huolestumista

Sparsity Problems

- Chinese and Japanese have no spaces, must be **segmented** into words



(Lots of!) Previous Research

- Transliteration: [Knight&Graehl 98, Al-Onaizan&Knight 02, Kondrak+ 03, Finch&Sumita 07]
- Compounds/Morphology: [Niessen&Ney 00, Brown 02, Lee 04, Goldwater&McClosky 05, Talbot&Osborne 06, Bojar 07, Macherey+ 11, Subotin 11]
- Segmentation: [Bai 08, Chang 08, Zhang 08]
- All focus on solving one of these particular problems

Can We Translate Letters?
[Vilar+ 07]

- Problems because we are translating words!

F = こ れ は ペ ン で す

↓

E = t h i s _ i s _ a _ p e n

- Previously: “Yes, but only for similar languages”
  - Spanish-Catalan [Vilar+ 07]
    Thai-Lao [Sornlertlamvanich+ 08]
    Swedish-Norwegian [Tiedemann 09]

Yes, We Can!

- We show **character-based MT can match word-based MT for distant languages**.
- Key: Many-to-many alignment through Bayesian Phrasal ITG [Neubig+ 11]
    - Improved speed and accuracy for character-based alignment
- Competitive automatic and human evaluation
- Handles many sparsity phenomena

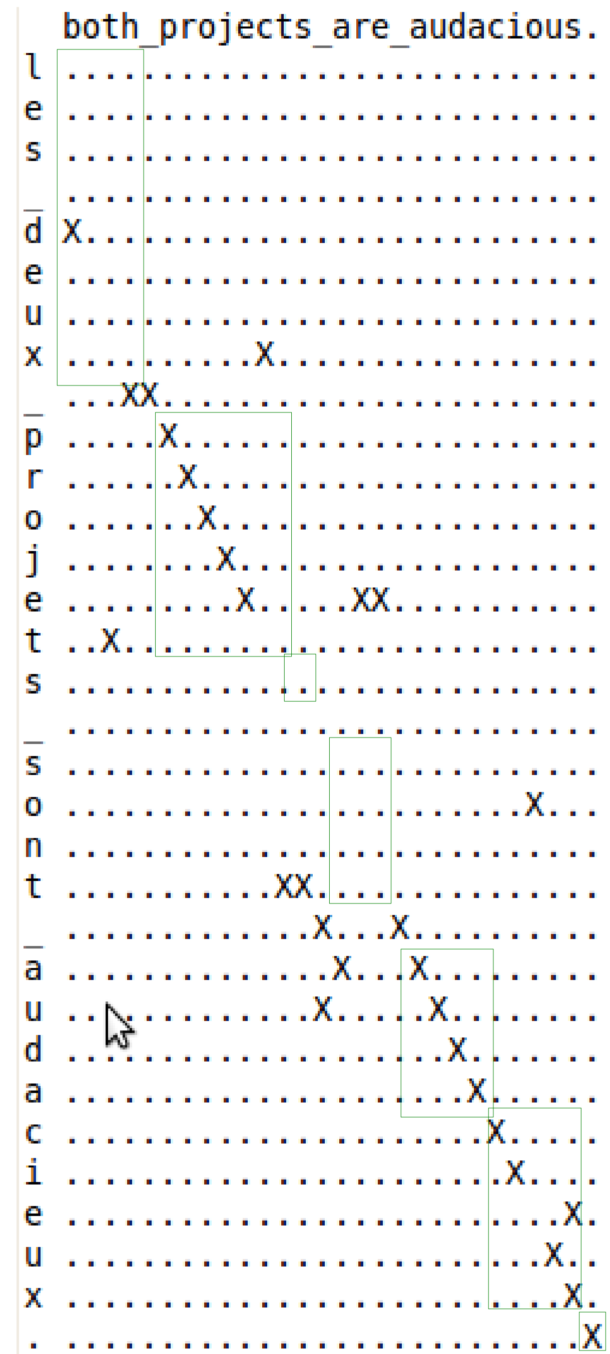
Word/Character Alignment

One-to-Many Alignment (IBM Models, GIZA++)

- Each source word must align to at most one target word [Brown 93, Och 05]

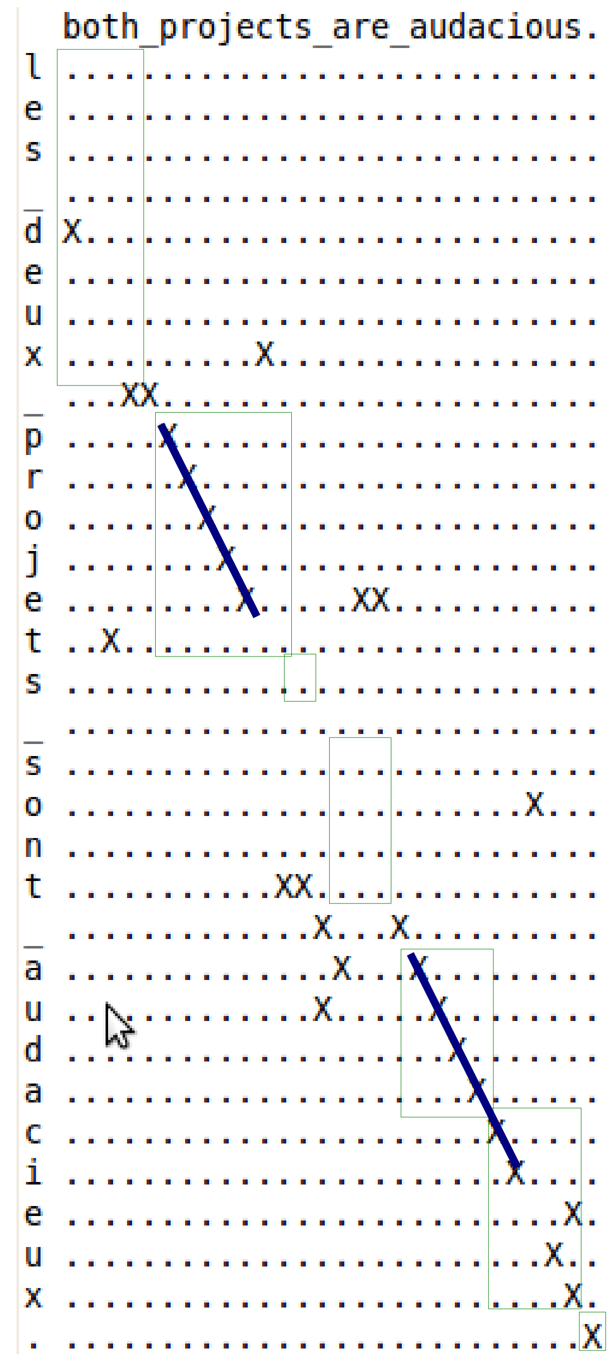
One-to-Many Alignment of Character Strings

- There is not enough information in single characters to align well



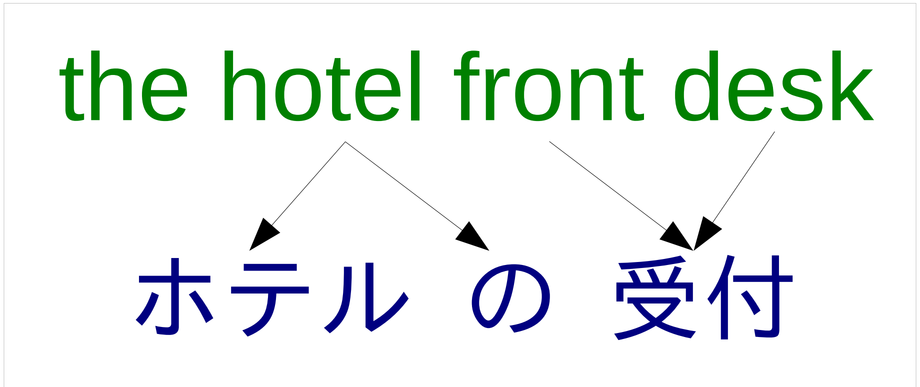
One-to-Many Alignment of Character Strings

- There is not enough information in single characters to align well
- Words with the same spelling do work:
  - “proje” ⇔ “proje”
  - “audaci” ⇔ “audaci”



Many-to-Many Alignment

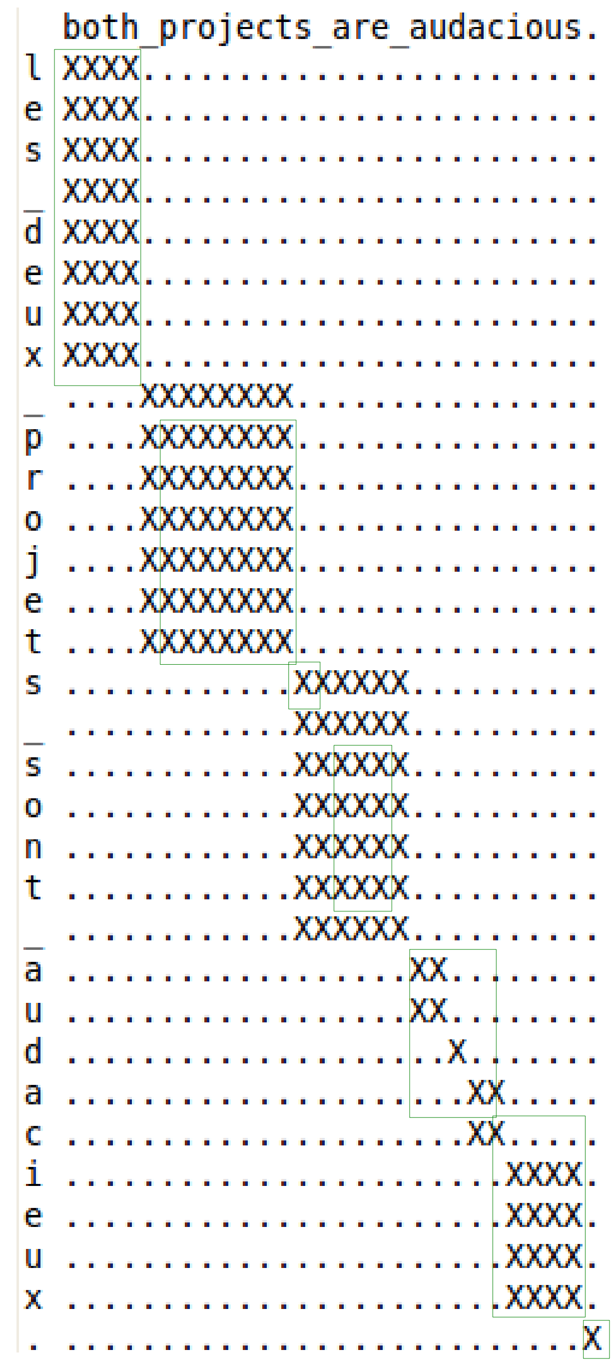
- Can directly generate phrasal alignments



- Often use the Inversion Transduction Grammar framework [Zhang 08, DeNero 08, Blunsom 09, Neubig 11]

Many-to-Many Alignment of Character Strings

- Example of [Neubig+ 11] applied to characters
- Recover many types of alignments:
  - Words: “project” ⇔ “projet”
  - Phrases: “both” ⇔ “les deux”
  - Subwords: “~cious” ⇔ “~cieux”
  - Even agreement!: “~s are” ⇔ “~s sont”



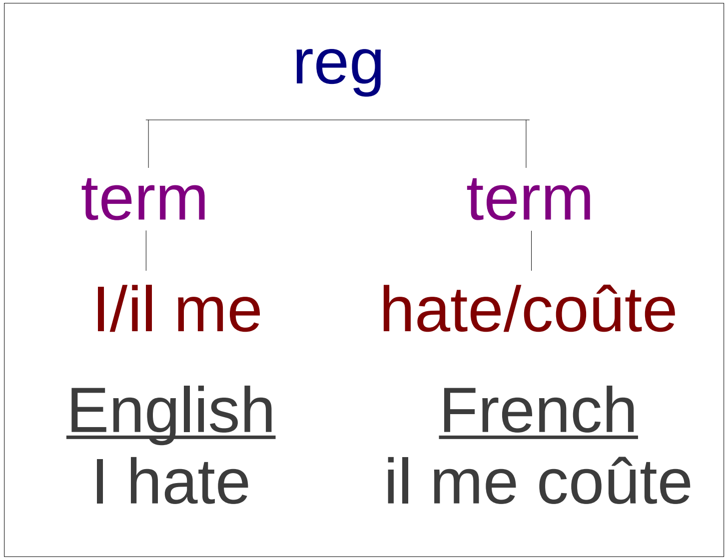
Two Problems

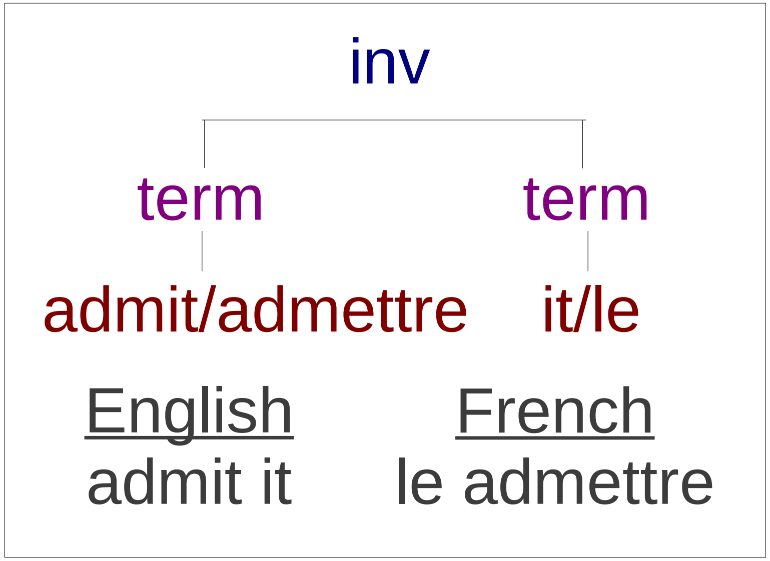
1) Alignment algorithm is too slow
    - We introduce a more effective beam pruning method using look-ahead probabilities (similar to A*)
2) Prior probability is still single-unit based
    - We introduce prior based on sub-string co-occurrence

Look-Ahead Parsing for ITGs

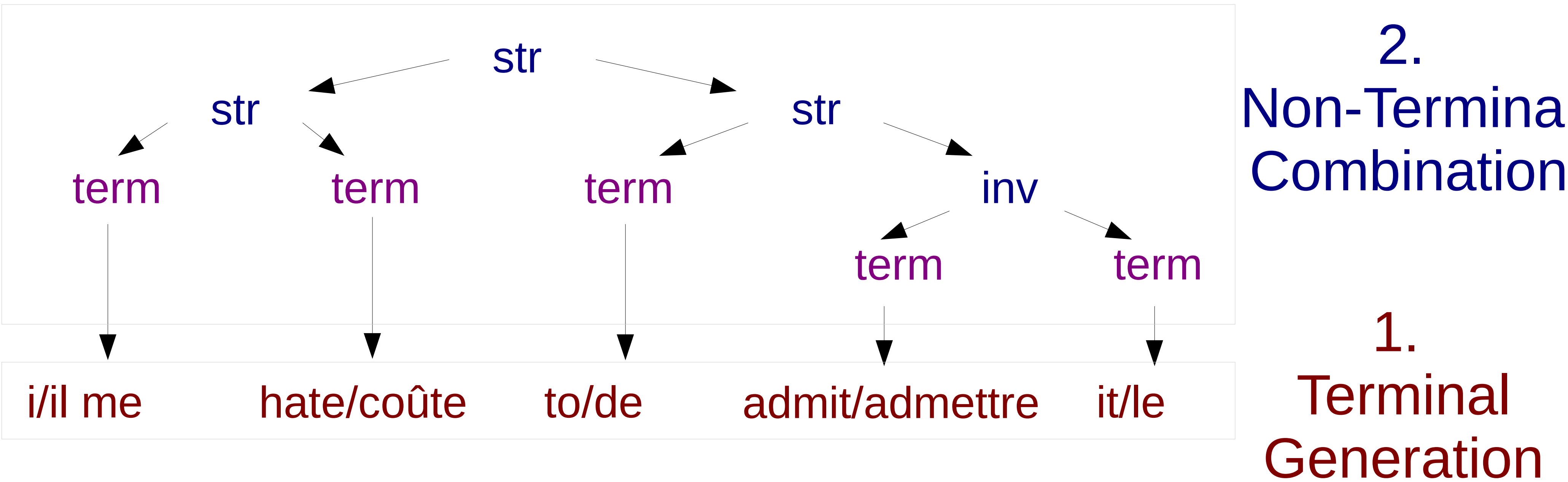
Inversion Transduction Grammar (ITG)

- Like a CFG over two languages
  - Have non-terminals for regular and inverted productions
  - One pre-terminal
  - Terminals specifying phrase pairs





Two Steps in ITG Parsing



- Step 1 is calculated by looking up all phrase pairs
- Step 2 is calculated by combining neighboring pairs
  - Takes most of the time

Beam Search for ITGs [Saers+ 09]

- Stacks of elements with same number of words

Size 1		Size 2		Size 3	
i/ε	P=1e-6	i/me	P=1e-3	i/il me	P=1e-5
ε/il	P=1e-6	to/de	P=5e-4	hate/me coûte	P=1e-6
ε/le	P=7e-7	it/le	P=4e-4	i hate/coûte	P=8e-7
to/ε	P=6e-7	i/il	P=2e-4	to/il me	P=4e-7
ε/me	P=3e-7	hate/coûte	P=1e-4	admit/le admettre	P=8e-8
ε/de	P=2e-7	ε/il me	P=4e-5	admit it/admettre	P=5e-8
it/ε	P=2e-7	admit/admettre	P=2e-5	i/me coûte	P=2e-8
hate/ε	P=5e-8	to/me	P=5e-6	to/me	P=1e-8
…		…		…	

- Do not expand elements outside of fixed beam (1e-1)

Problem with Simple Beam Search

- Does not consider competing alignments!

		f_1 les	f_2 années	f_3 60
e_1	the	-2	-12	-5
e_2	1960s	-8	-8	-8

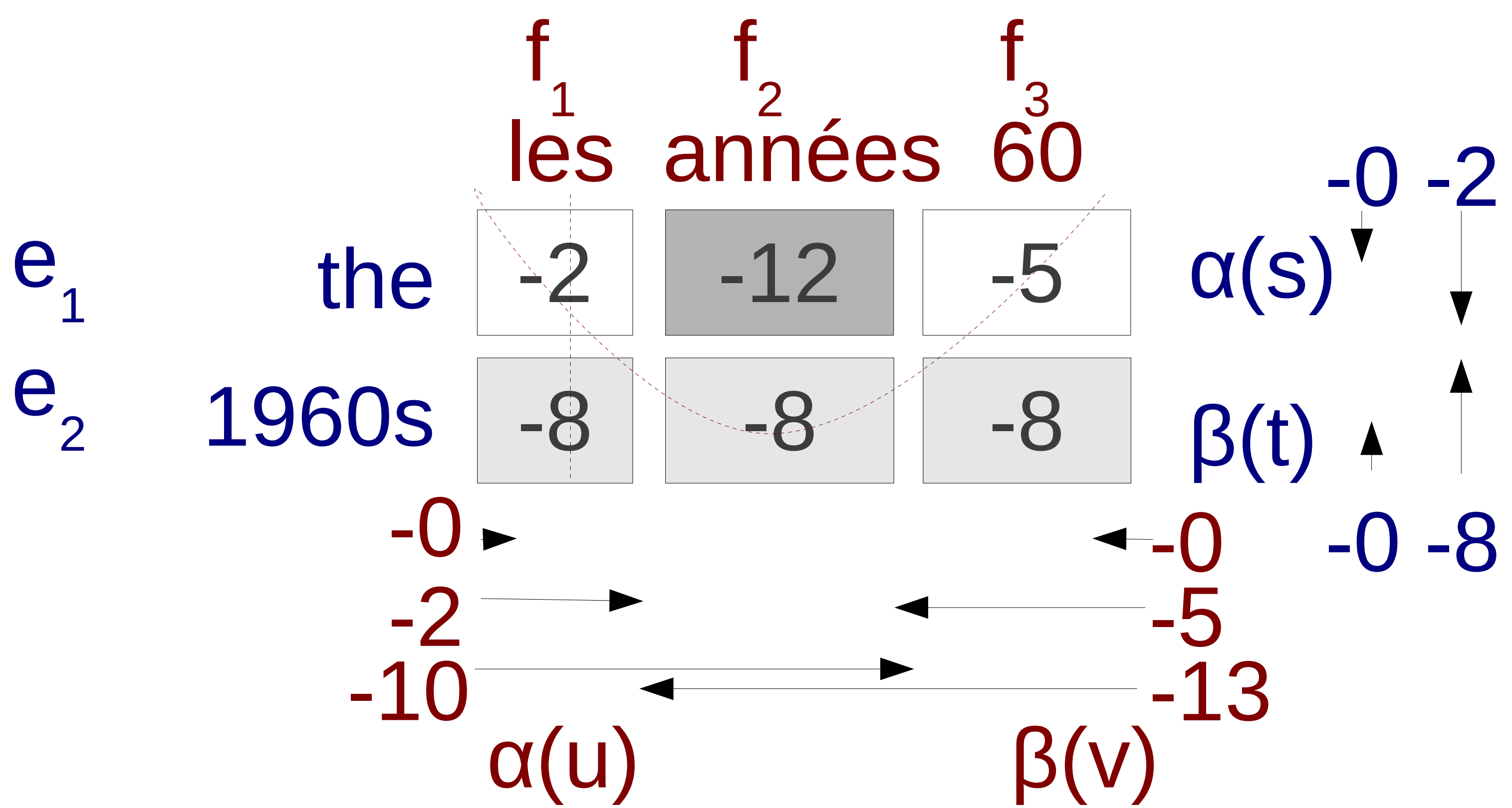
Has competitor “les/the”
= can be pruned

Has no good competitor
= should not be pruned

* scores are log probabilities

Proposed Solution: Look Ahead Probabilities and A* Search

- Minimum probability to translate monolingual span



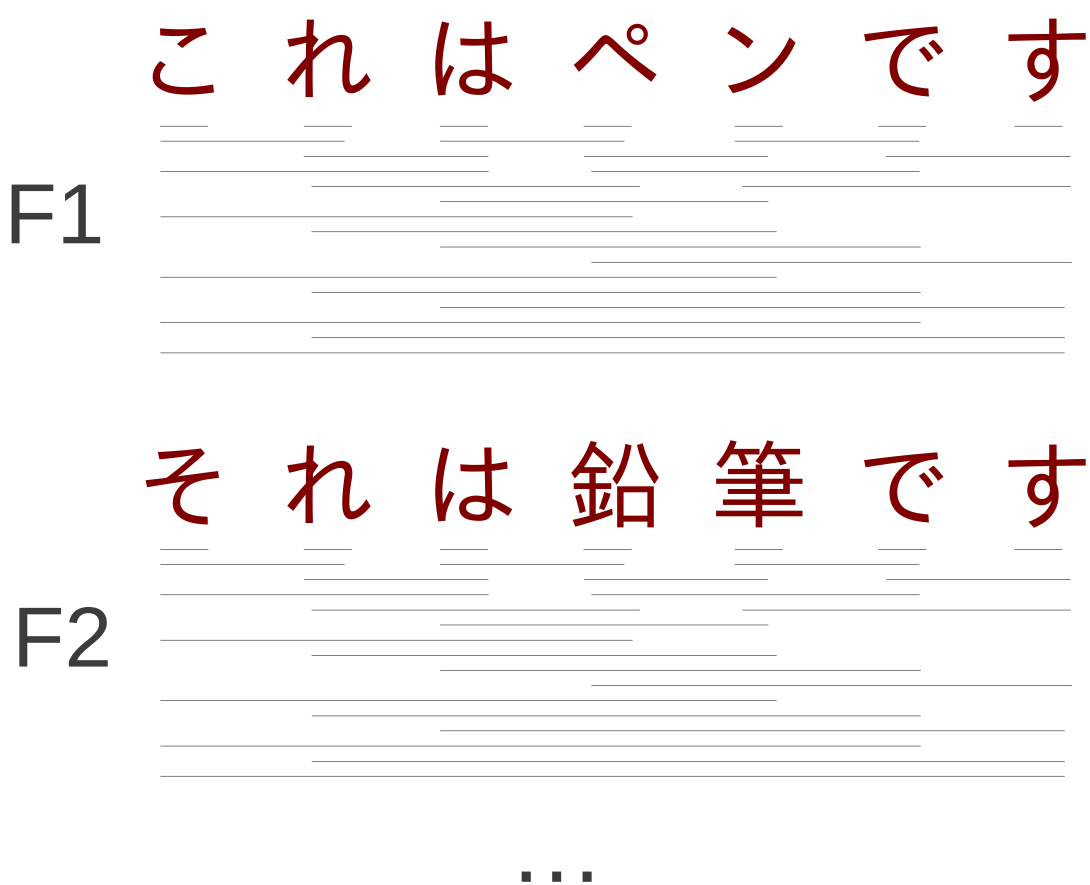
- Beam score: inside prob * outside probabilities

$$\min\left(\begin{matrix}\alpha(u)+\log(P(s,t,u,v))+\beta(v), \\ \alpha(s)+\log(P(s,t,u,v))+\beta(t)\end{matrix}\right)$$

Substring Co-occurrence Prior Probability

Substring Occurrence Statistics

- For each input sentence count every substring
  - Use enhanced suffix array for efficiency (esaxx library)
- Make a matrix



	F1	F2
こ	1	0
れ	1	1
これ	1	0
は	1	1
れは	1	1
これは	1	0
ペ	1	0
はペ	1	0
れはペ	1	0
これはペ	1	0

…

Substring Co-occurrence Statistics

- Take the product of two matrices to get co-occurrence

	F1	F2
こ	1	0
れ	1	1
これ	1	0
は	1	1
れは	1	1
これは	1	0
ぺ	1	0
はぺ	1	0
れはぺ	1	0
これはぺ	1	0

…

*

	t	h	th	i	hi	thi	s	is	his	this
E1	1	1	1	1	1	1	1	1	1	1
E2	1	1	1	1	0	0	1	1	0	0

…

c(f,e)

Making Probabilities and Discount

- Convert counts to probabilities by taking geometric mean of conditional probabilities (best results)
- In addition, discount counts by fixed d (=5)
  - Reduces memory usage (do not store $c_{e,f}$ <= 5)
  - Helps prevent over-fitting of the training data

$$P(e,f)=\left(\frac{c_{e,f}-d}{c_e-d}\frac{c_{e,f}-d}{c_f-d}\right)/Z$$

Experiments

Experimental Setup

- 4 languages with varying characteristics ⇔ English:
  - German: Some compounding
  - Finnish: Very morphologically rich
  - French: Mostly word-word correspondence
  - Japanese: Requires segmentation, transliteration

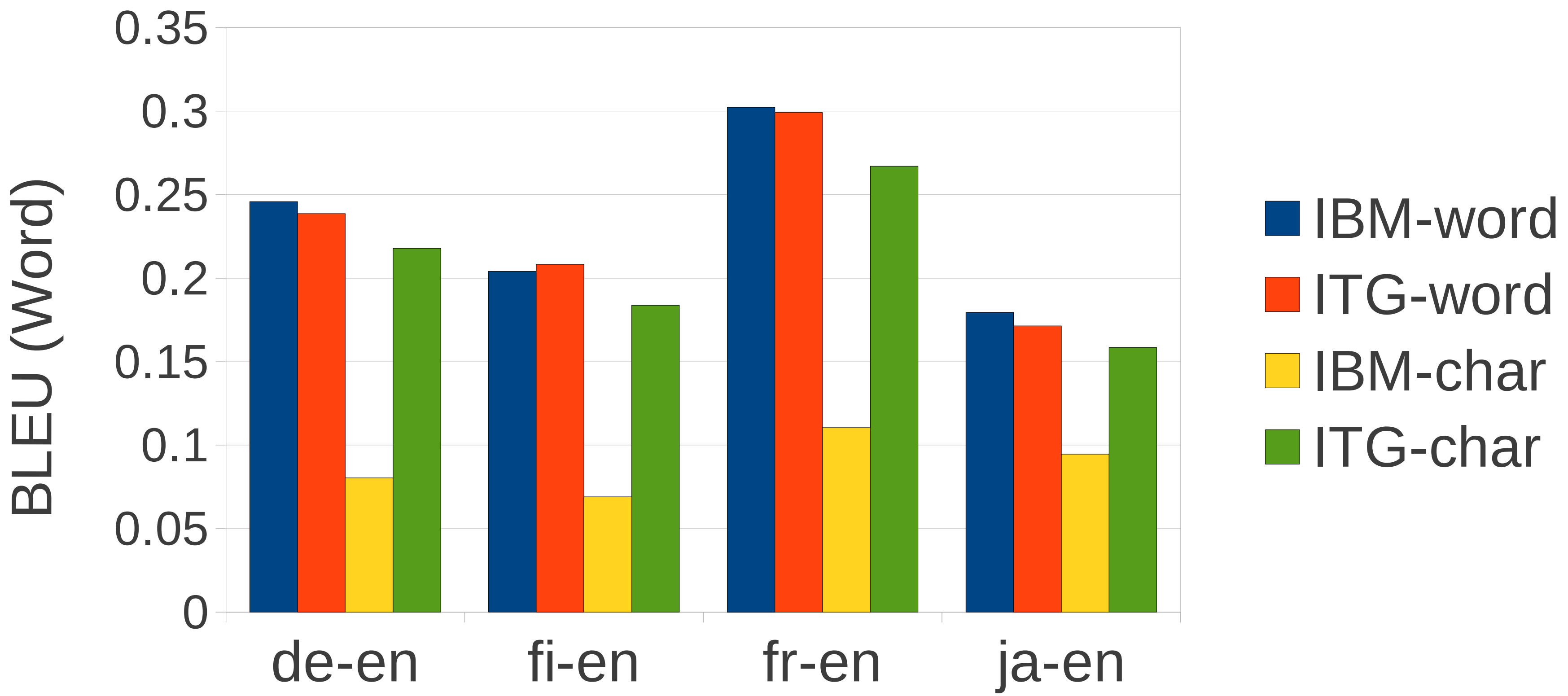
	EuroParl				KFTT	
	de	fi	fr	en	ja	en
TM	2.56M	2.23M	3.05M	2.80M/3.10M/2.77M	2.34M	2.13M
LM	15.3M	11.3M	15.6M	16.0M/15.5M/13.8M	11.9M	11.5M
Tune	55.1k	42.0k	67.3k	58.7k	34.4k	30.8k
Test	54.3k	41.4k	66.2k	58.0k	28.5k	26.6k

- Sentences of 100 characters and under were used
- Evaluate with word/character BLEU, METEOR

Systems

- Which unit?
  - Word-Based: Align and translate words
  - Char-Based: Align and translate characters
- Which alignment method?
  - One-to-many: IBM Model 4 for words, HMM model for characters
  - Many-to-many: ITG-based model with proposed improvements

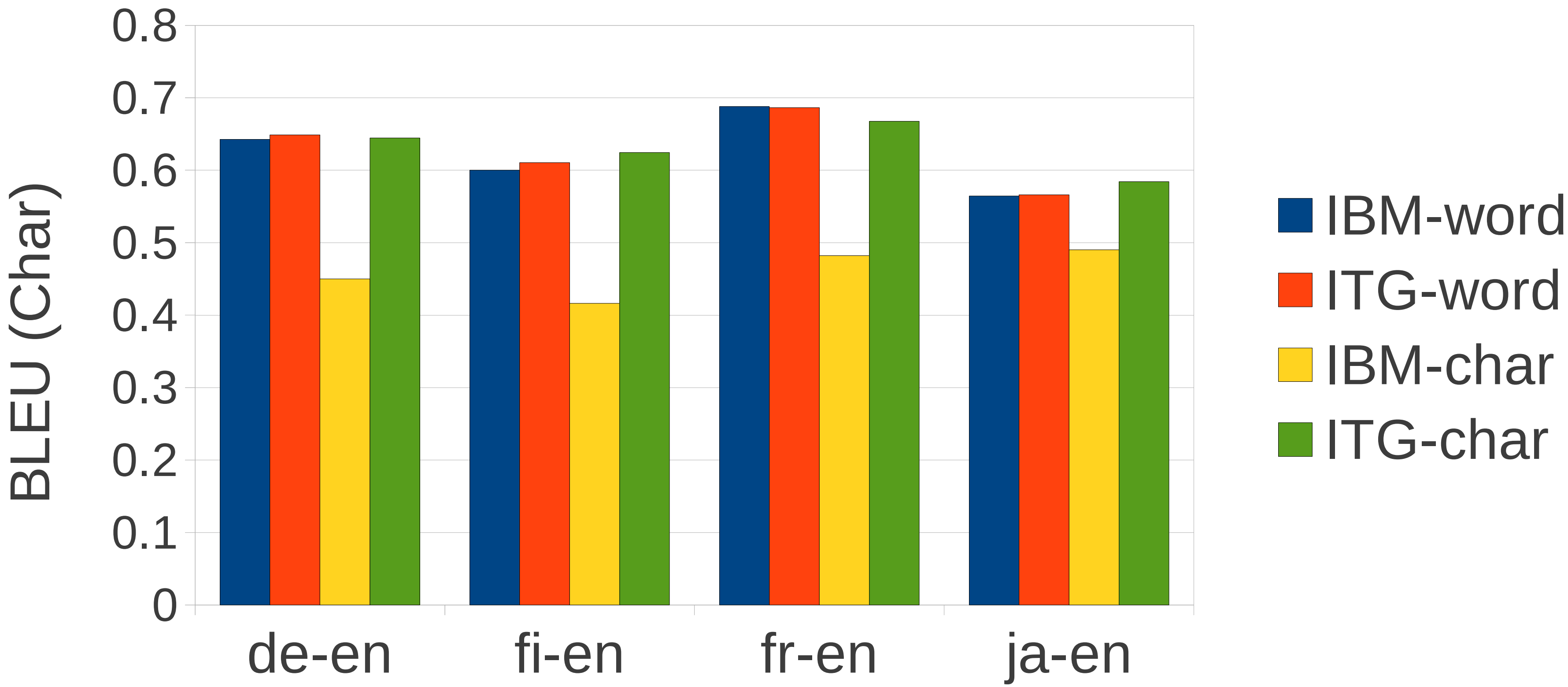
BLEU Score (Word)



ITG-Char vs.

	de-en	fi-en	fr-en	ja-en
IBM-Char:	+0.1374	+0.1147	+0.1565	+0.0638
ITG-Word:	-0.0208	-0.0245	-0.0322	-0.0130

BLEU Score (Char)



<u>ITG-Char vs.</u>

IBM-Char: +0.1946 +0.2082 +0.1853 +0.0939

ITG-Word: -0.0042 +0.0140 -0.0188 +0.0181

Human Adequacy Evaluation

- 200 sentences in each language, 0-5 rating

	ITG-Word	ITG-Char
ja-en	2.085	2.154
fi-en	2.851	2.826

- Systems are comparable (no difference at P<0.05)

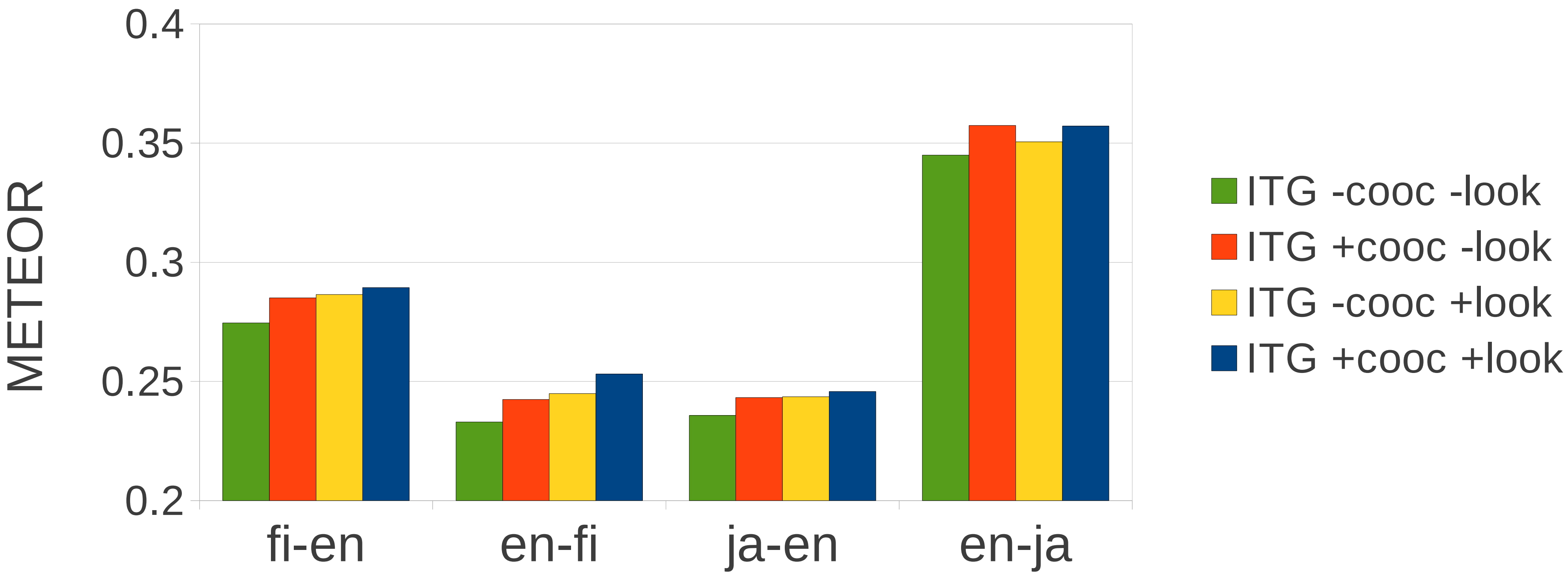
Notable Improvements

- Examples where ITG-Char was 2+ points better

Unknown (13/26)	Ref	directive on equality
	Word	tasa-arvodirektiivi
	Char	equality directive
Target Unknown (5/26)	Ref	yoshiwara-juku station
	Word	yoshiwara no eki
	Char	yoshiwara-juku station
Uncommon (5/26)	Ref	world health organisation
	Word	world health
	Char	world health organisation

- ITG-Word was often better at reordering

Effect of Proposed Improvements over [Neubig+ 11] (METEOR)



- Look-ahead also allowed for a 2x improvement in speed

Conclusion

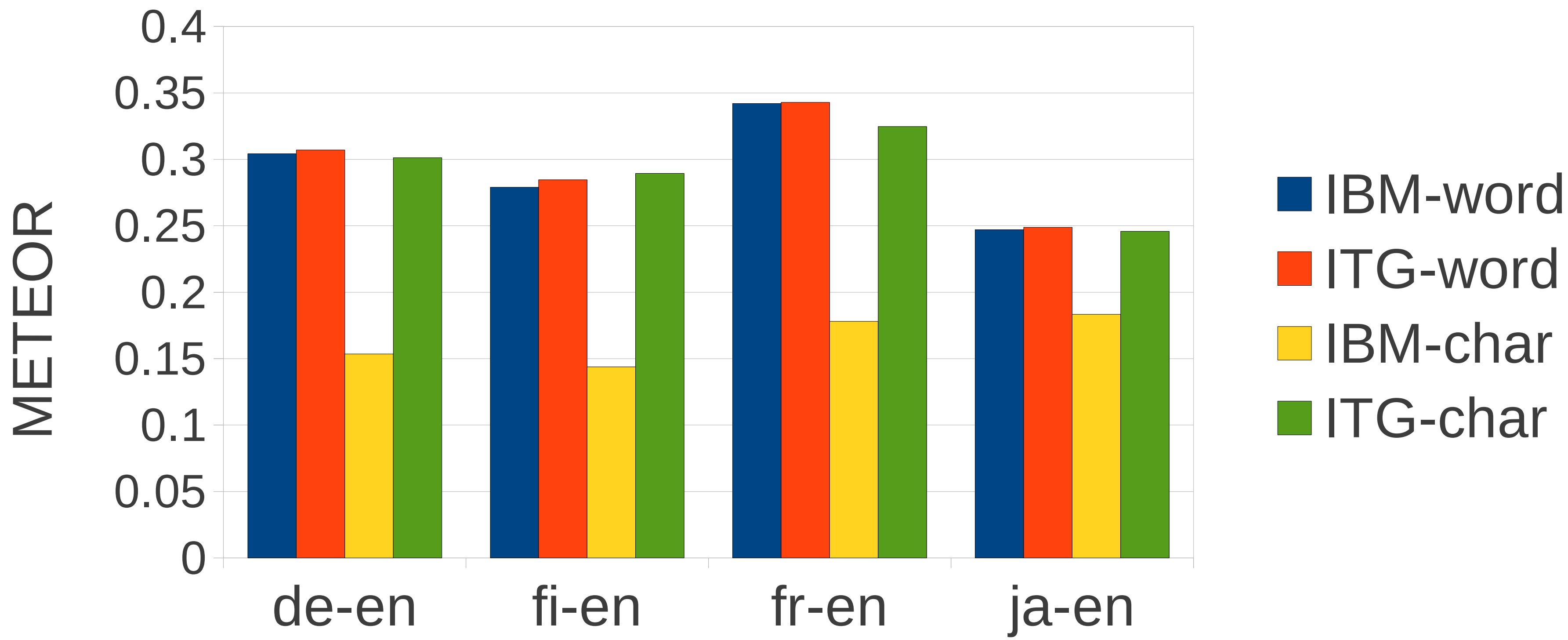
- Character-based SMT can achieve comparable results to word-based SMT
  - + Is able to handle sparsity issues
- Remaining challenges:
  - Improve decoding to allow for better reordering
  - Treat spaces differently than other characters to improve alignment/decoding

Available Open Source:
http://www.phontron.com/pialign

Thank You!

METEOR Score (Word)

- Counts reordering, matches using lemmatization, synonyms

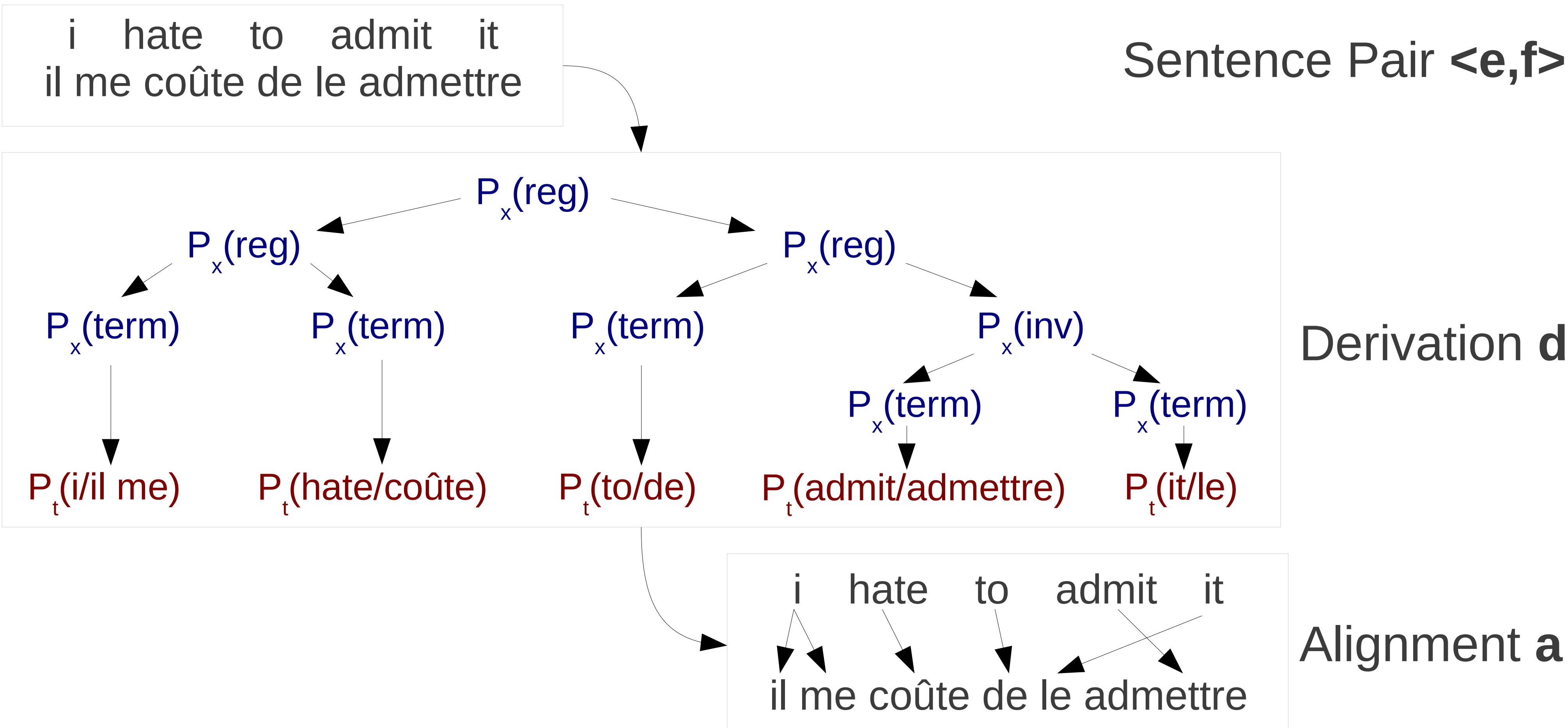


ITG-Char vs.

IBM-Char: +0.1477 +0.1455 +0.1467 +0.0624
ITG-Word: -0.0059 +0.0048 -0.0182 -0.0031

Biparsing-based Alignment with ITGs

- Non/pre-terminal distribution P_x, and phrase distribution P_t



- Viterbi parsing and sampling both possible in $O(n^6)$