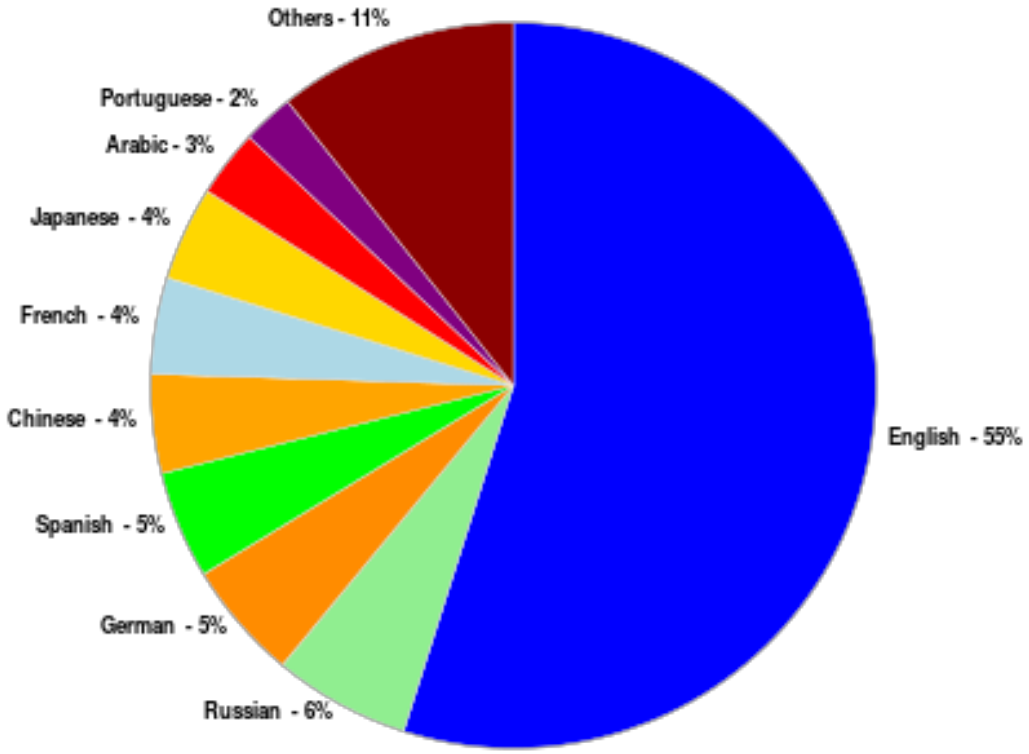
Introduction to NLP

451.

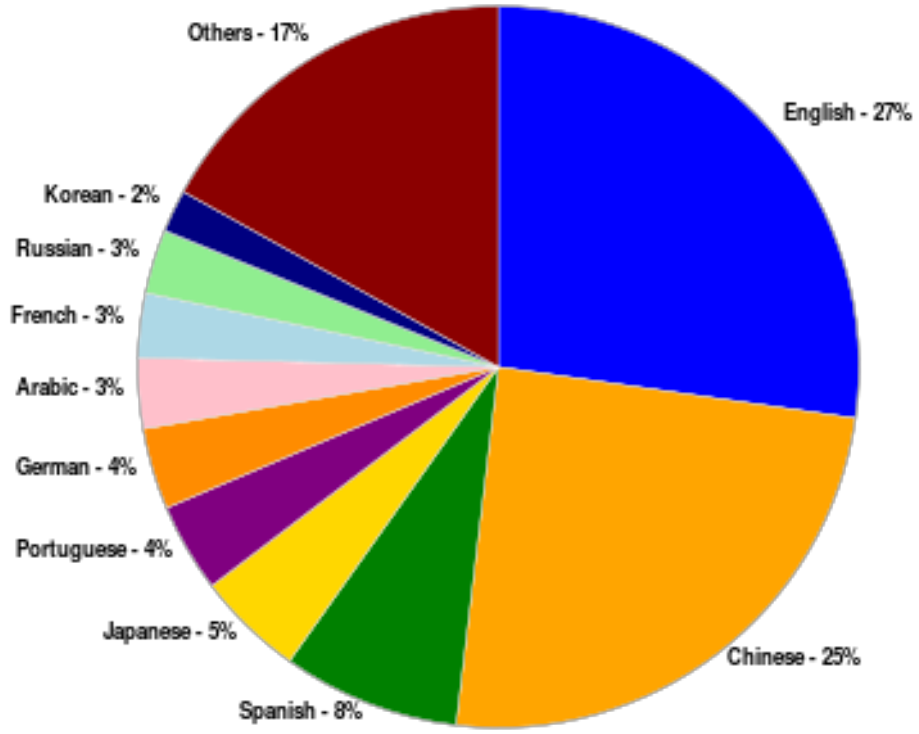
Machine Translation

Multilingual Users

- Content languages for websites



Percentage of Internet users by language



April 2013

http://en.wikipedia.org/wiki/Global_Internet_usage

Genesis 11:1-9

[The Tower of Babel, by Pieter Bruegel the Elder, 1563]

The Rosetta Stone

Carved in 196 BC in Egypt
Deciphered by Champollion in 1822
Mixture of Egyptian (hieroglyphs and Demotic) and Greek

http://www.ancientegypt.co.uk/writing/rosetta.html

English-Cebuano Bible Example

In the beginning God created the heaven and the earth.

Sa sinugdan gibuhat sa Dios ang mga langit ug ang yuta.

And God called the firmament Heaven.

Ug gihinganlan sa Dios ang hawan nga Langit.

And God called the dry land Earth

Ug ang mamala nga dapit gihinganlan sa Dios nga Yuta

- use: co-occurrence, word order, cognates
- corpora are needed
- sentence alignment needs to be done first

http://en.wikipedia.org/wiki/Bible_translations_by_language

NACLO Problem

- http://nacloweb.org/resources/problems/2012/N2012-C.pdf
- http://nacloweb.org/resources/problems/2012/N2012-CS.pdf
- Problem by Simon Zwarts, based on work by Kevin Knight

Arcturan Problem – 1/4

It's hard enough to translate between languages when you understand both languages. It's harder still when you only understand one. But what do computers do? They don't truly understand either language. To illustrate the challenge that computers face, Kevin Knight posed this classic puzzle (Knight 1997): given two equivalent texts in two unknown alien languages, how would you go about translating one to another?

It is the year 2354 AD. Our scientists have been eavesdropping on messages between two alien civilizations for a very long time, but we have never met either. The closest aliens, the Centauri, have finally begun to communicate with us. Their first message was a message of peace, "Farok crrok hihok yorok clok kantok ok -yurp."

Now, we know that the Centauri have been in contact for some time with the Arcturan race, who live in another solar system. We have never had contact with the Arcturans, but newly developed technology makes it possible for us to send them a message. We would like to send them, first, a message of peace, but because we do not understand their language, this is not an easy task.

Luckily, we have intercepted communications from the Centauri that include both languages. Here are 12 sentences in Centauri and their 12 translations in Arcturan. Unfortunately, because we have only been eavesdropping, their meaning is unknown. However, we do know that the sentence pairs on each line are translations of each other. We want to use this information to translate the original peace message from the Centauri and then send this to the Arcturans. Your assignment will be to do this translation.

Arcturan Problem – 2/4

CENTAURI	ARCTURAN
ok-voon ororok sprok.	at-voon bichat dat.
ok-drubel ok-voon anok plok sprok.	at-drubel at-voon pippat rrat dat.
erok sprok izok hihok ghirok.	totat dat arrat vat hilat.
ok-voon anok drok brok jok.	at-voon krat pippat sat lat.
wiwok farok izok stok.	totat jjat quat cat.
lalok sprok izok jok stok.	wat dat krat quat cat.
lalok farok ororok lalok sprok izok enemok.	wat jjat bichat wat dat vat eneat.
lalok brok anok plok nok.	iat lat pippat rrat nnat.
wiwok nok izok kantok ok-yurp.	totat nnat quat oloat at-yurp.
lalok mok nok yorok ghirok clok.	wat nnat gat mat bat hilat.
lalok nok crrok hihok yorok zanzanok.	wat nnat arrat mat zanzanat.
lalok rarok nok izok hihok mok.	wat nnat forat arrat vat gat.

Arcturan Problem – 3/4

C-1 Let's start with the first Centauri word: "farok". This word occurs in two of our Centauri sentences. Given that these sentences' Arcturan translations only have one word in common with each other, we can assume that this word is the translation for "farok". Which word it is?

farok

C-2 Do the same thing for "hihok" and "yorok". For "yorok" you will need to make some assumptions about word ordering.

hihok

yorok

C-3 The Centauri word "clok" only occurs once. However, you can figure out its Arcturan translation in another way.

clok

C-4 Try to use the processes from the previous assignments to complete as much as possible of the following table.

crrok

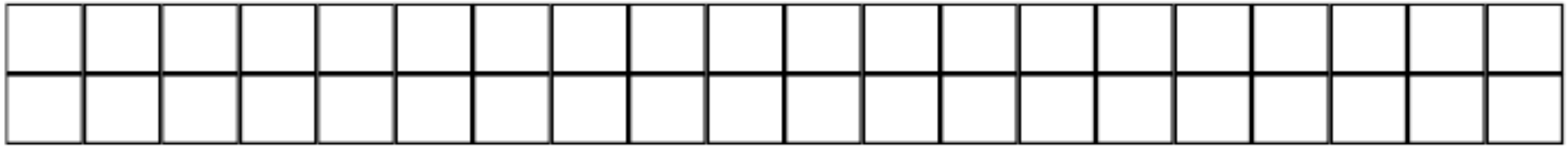
kantok

ok-yurp

Arcturan Problem – 4/4

C-5 Complete the translation of "farok crrok hihok yorok clok kantok ok-yurp." Keep in mind that Centauri and Arcturan sentences can have a different word order. There may be more than one correct reply.

C-6 After some years a reply message is received in Arcturan. It reads, "Totat nnat forat arrat mat bat." Translate this message into Centauri. There may be more than one correct reply.



Parallel Corpora

- The Rosetta Stone
- The Hansards Corpus
- The Bible and other religious texts
- Europarl

42nd PARLIAMENT, 1st SESSION

EDITED HANSARD • NUMBER 095

CONTENTS

Friday, October 21, 2016

HOUSE OF COMMONS
CHAMBRE DES COMMUNES
CANADA

House of Commons Debates

VOLUME 148 | NUMBER 095 | 1st SESSION | 42nd PARLIAMENT

OFFICIAL REPORT (HANSARD)

Friday, October 21, 2016

Speaker: The Honourable Geoff Regan

The House met at 10 a.m.

Prayer

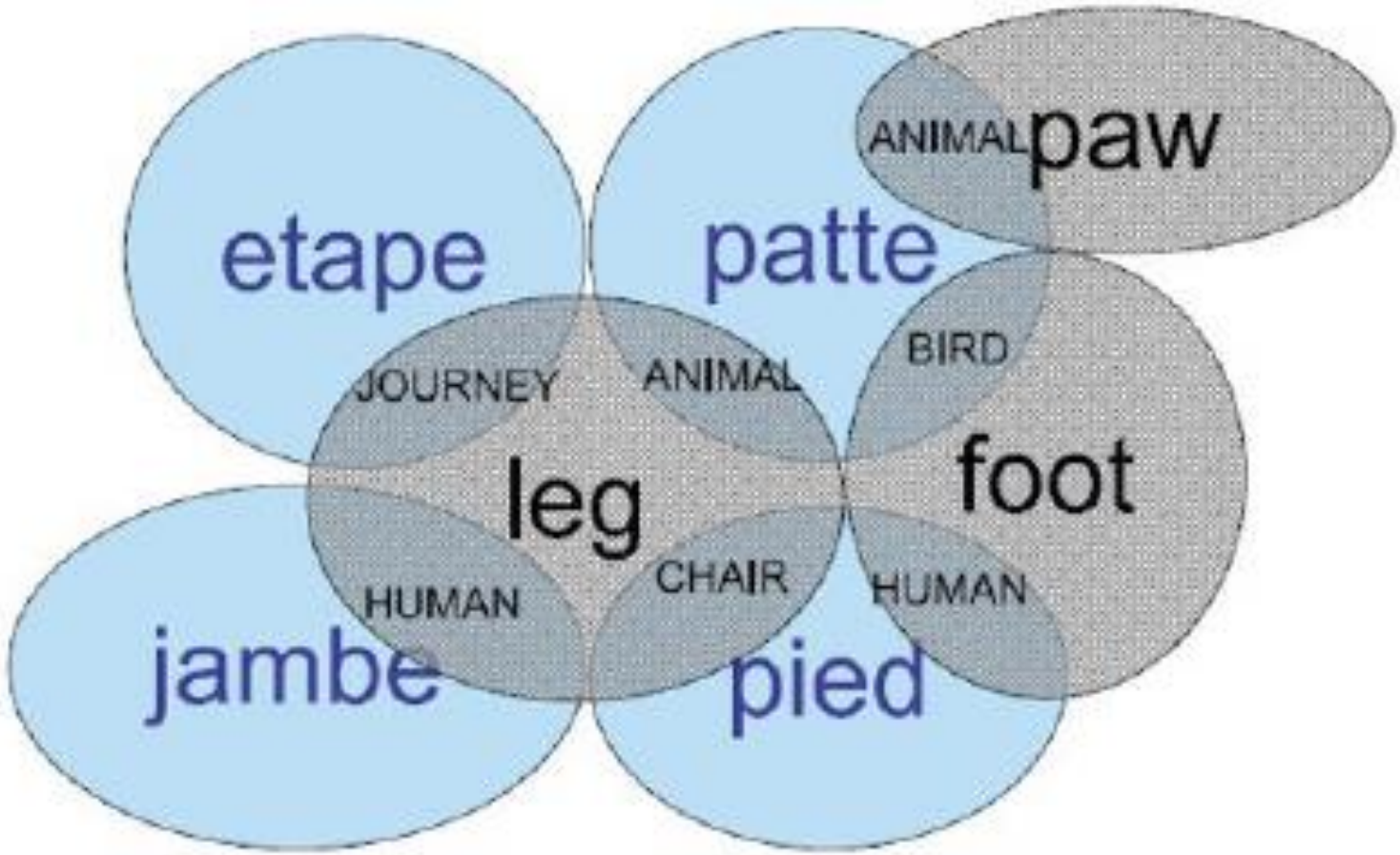
GOVERNMENT ORDERS

[Government Orders]

Hansards Example

- English
  - <s id=960001> I would like the government and the Postmaster General to agree that we place the union and the Postmaster General under trusteeship so that we can look at his books and records, including those of his management people and all the memos he has received from them, some of which must have shocked him rigid.
  - <s id=960002> If the minister would like to propose that, I for one would be prepared to support him.
- French
  - <s id=960001> Je voudrais que le gouvernement et le ministre des Postes conviennent de placer le syndicat et le ministre des Postes sous tutelle afin que nous puissions examiner ses livres et ses dossiers, y compris ceux de ses collaborateurs, et tous les mémoires qu'il a reçus d'eux, dont certains l'ont sidéré.
  - <s id=960002> Si le ministre voulait proposer cela, je serais pour ma part disposé à l'appuyer.

Language Differences (1/6)



[Example from Jurafsky and Martin]

Language Differences (2/6)

- Word order in phrases (Fr.)
  - la maison bleue, the blue house
- Word order in sentences (Jap.)
  - I like to drink coffee
  - watashi wa kohii o nomu no ga suki desu
  - I-subj coffee-obj drink-dat-rheme like
- vocabulary (Sp.)
  - wall
  - pared, muro
- phrases (Fr.)
  - play
  - pièce de théâtre

Language Differences (3/6)

- Prepositions (Jap.)
  - to Mariko, Mariko-ni
- Inflection (Sp.)
  - have: tengo, tienes, tenemos, tienen, tener
- Lexical distinctions (Sp.):
  - the bottle floated out - la botella salió flotando
- Brother (Jap.)
  - otooto (younger), oniisan (older)
- They (Fr.)
  - elles (feminine), ils (masculine)

Language Differences (4/6)

- Das Vorhaben verwarf die Kommission
- The plan rejected the commission
  - OSV reading is more plausible

- I saw the movie and it is good
  - How do we translate “it”?

[Examples from Philipp Koehn]

Language Differences (5/6)

- Color Names
  - Russian:
    - light blue (голубой, *goluboy*)
    - dark blue (синий, *siniy*)
  - Japanese:
    - **青** (ao) – both blue and green historically
    - **緑** (midori) – recent addition

Language Differences (6/6)

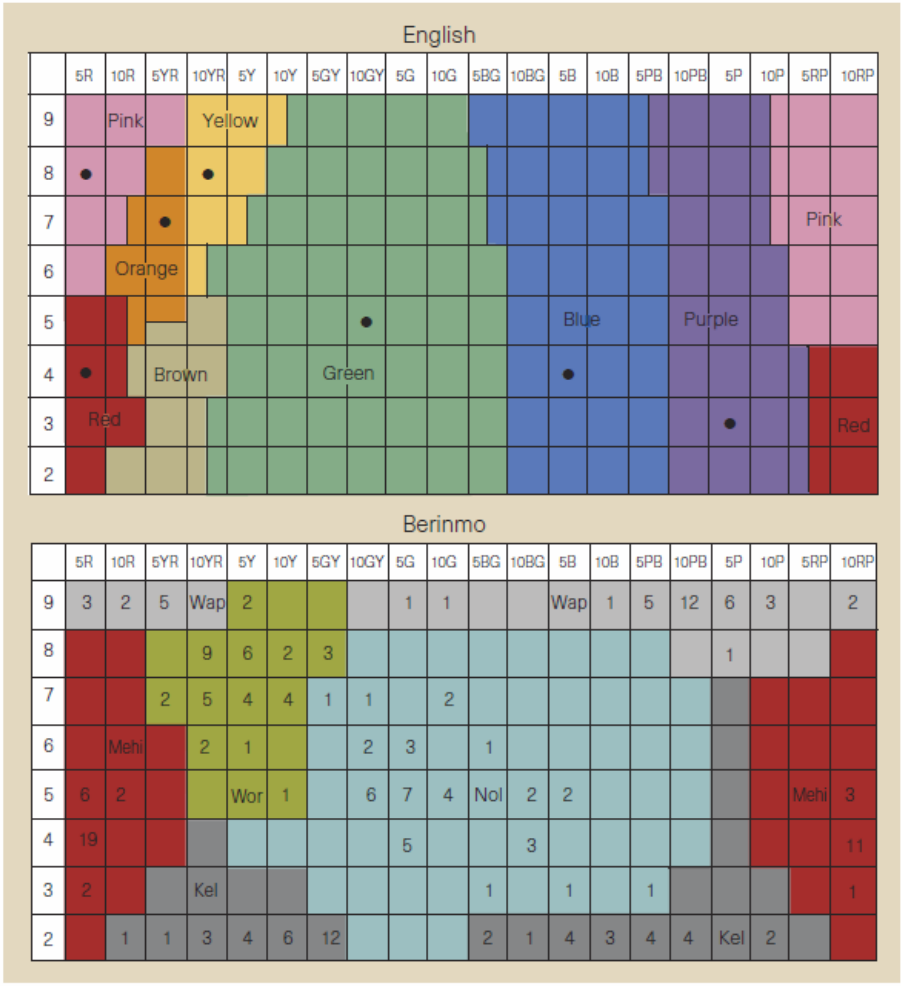


Figure 1 Distribution of English and Berinmo colour names. The Munsell system provides equally spaced samples in three dimensions, but is shown here as a Mercator projection of hue (horizontal axis) against lightness (vertical axis). The colours used to denote colour categories on these Mercator projections are for illustration only. Eight colour terms for English and five for Berinmo are shown. Dots on English naming data represent the position of focal colours[2]. Numbers on the Berinmo naming data represent the number of subjects who designated that colour as best example of the category. R, red; Y, yellow; G, green; B, blue; P, pink.

Davidoff et al. 99

Machine Translation

452.

Basic Techniques

Automatic Translation

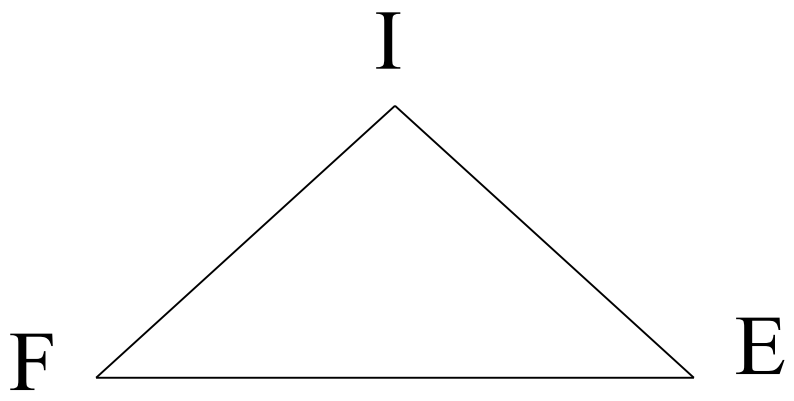
- Google Translate
  - Statistical system: 100 languages, 200M users daily
    - https://research.googleblog.com/2006/04/statistical-machine-translation-live.html
  - Newer (neural) system
    - https://research.googleblog.com/2016/09/a-neural-network-for-machine.html
- Facebook, Amazon, Microsoft, Baidu, etc.

Translation as Decoding

- “One naturally wonders if the problem of translation could conceivably be treated as a problem in cryptography. When I look at an article in Russian, I say: 'This is really written in English, but it has been coded in some strange symbols. I will now proceed to decode.' “
  - Warren Weaver, “Translation (1955)”

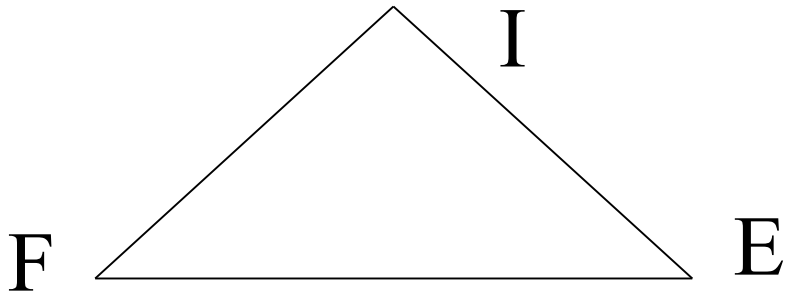
Vauquois's Triangle

- (F) oreign
- (I) nterlingua
- (E) nglish



Basic Strategies of MT

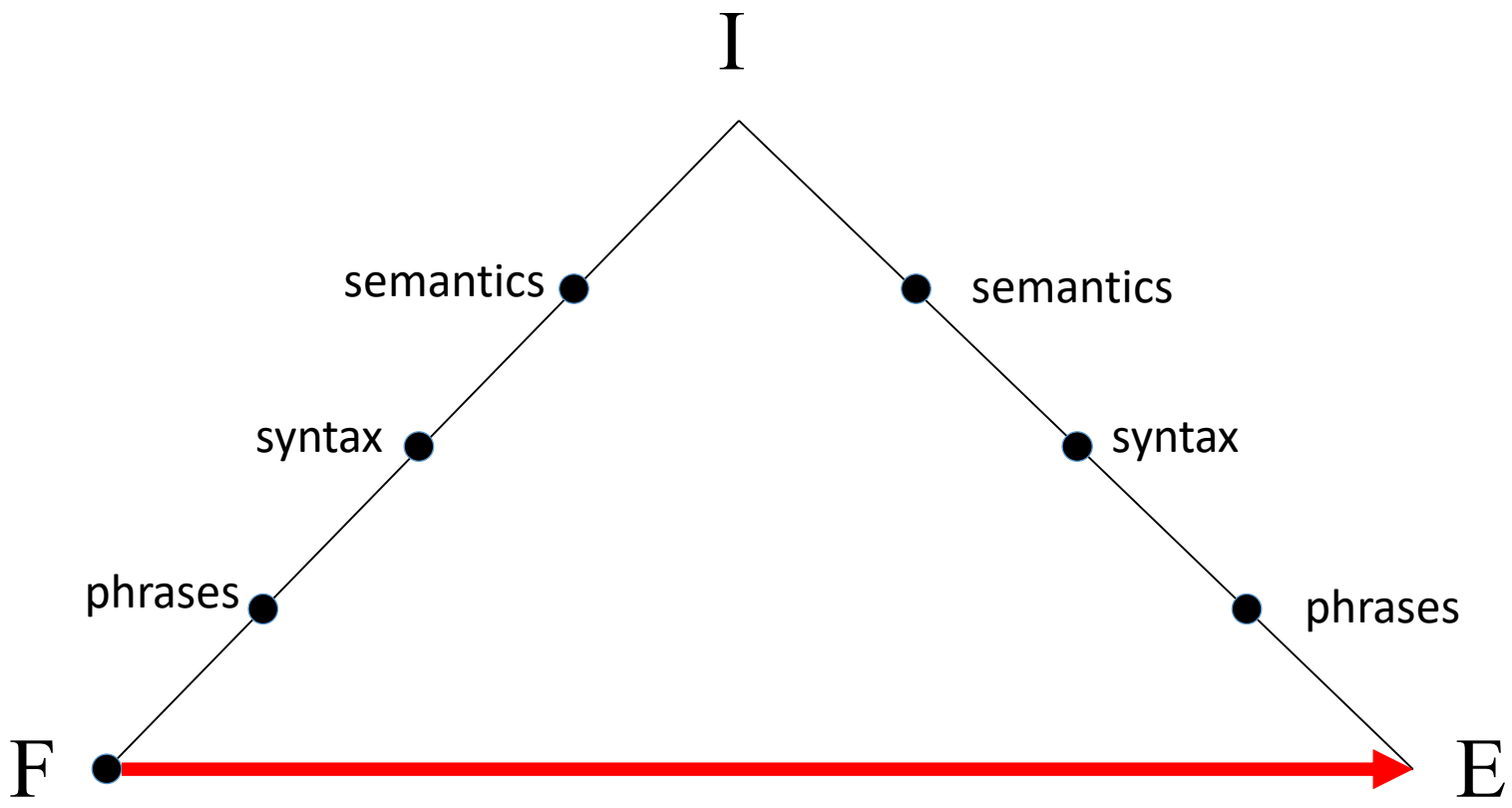
- Direct Approach
  - 50’s,60’s
  - naïve
  - the flesh is weak, but the spirit is strong
  - out of sight, out of mind
- Indirect: Transfer
- Indirect: Interlingua
  - No looking back
  - Language-neutral
  - No influence on the target language



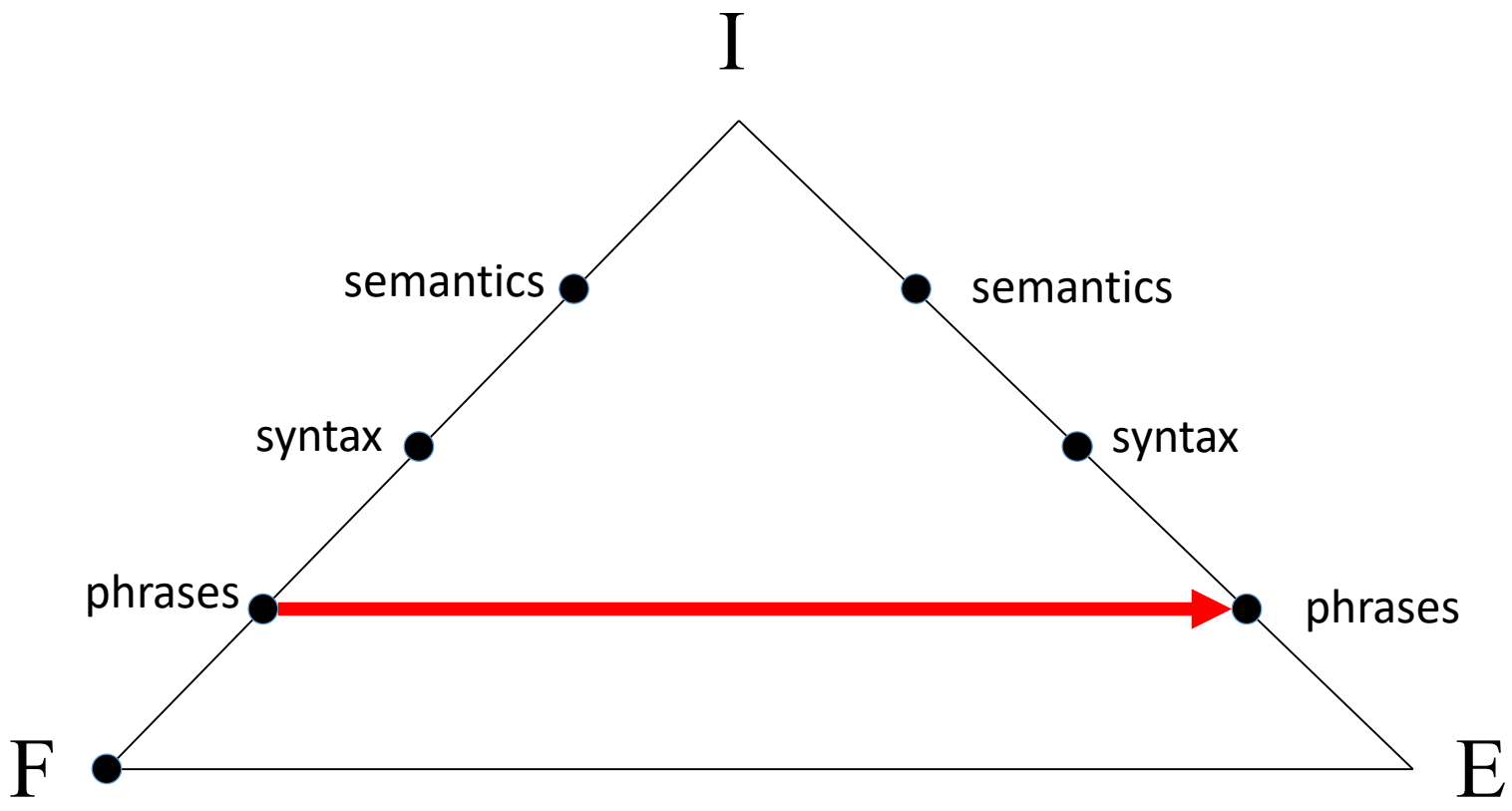
Basic Strategies of MT

- Example:
  - This is a blue house
- Direct Approach
  - translate each word separately
  - doesn’t work well across word orders
- Syntactic Transfer
  - Eng (adj noun) → Fr (noun adj)
- Interlingua
  - $\exists h$: House(h) $\wedge$ Blue(h)

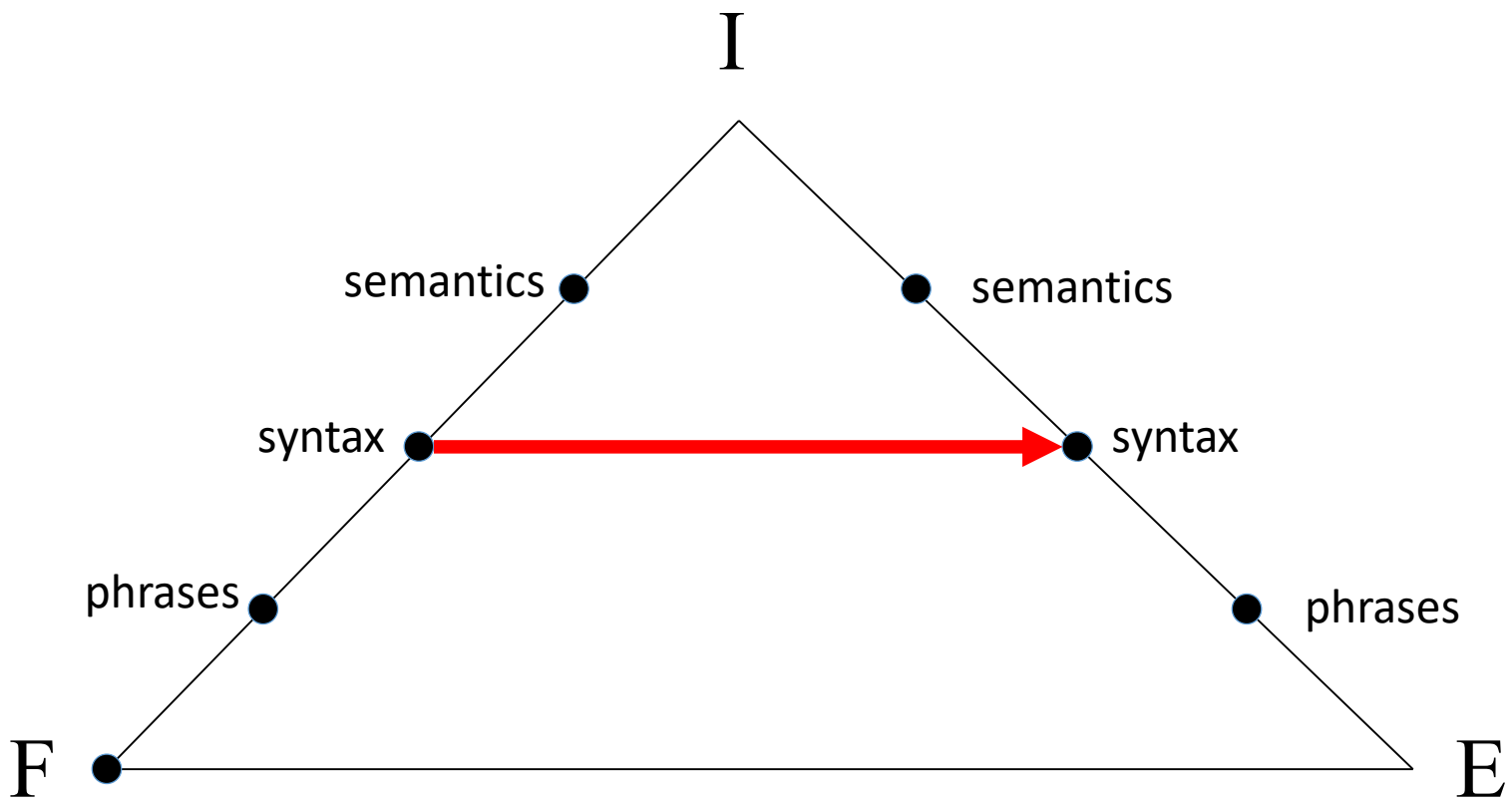
String-to-String Translation



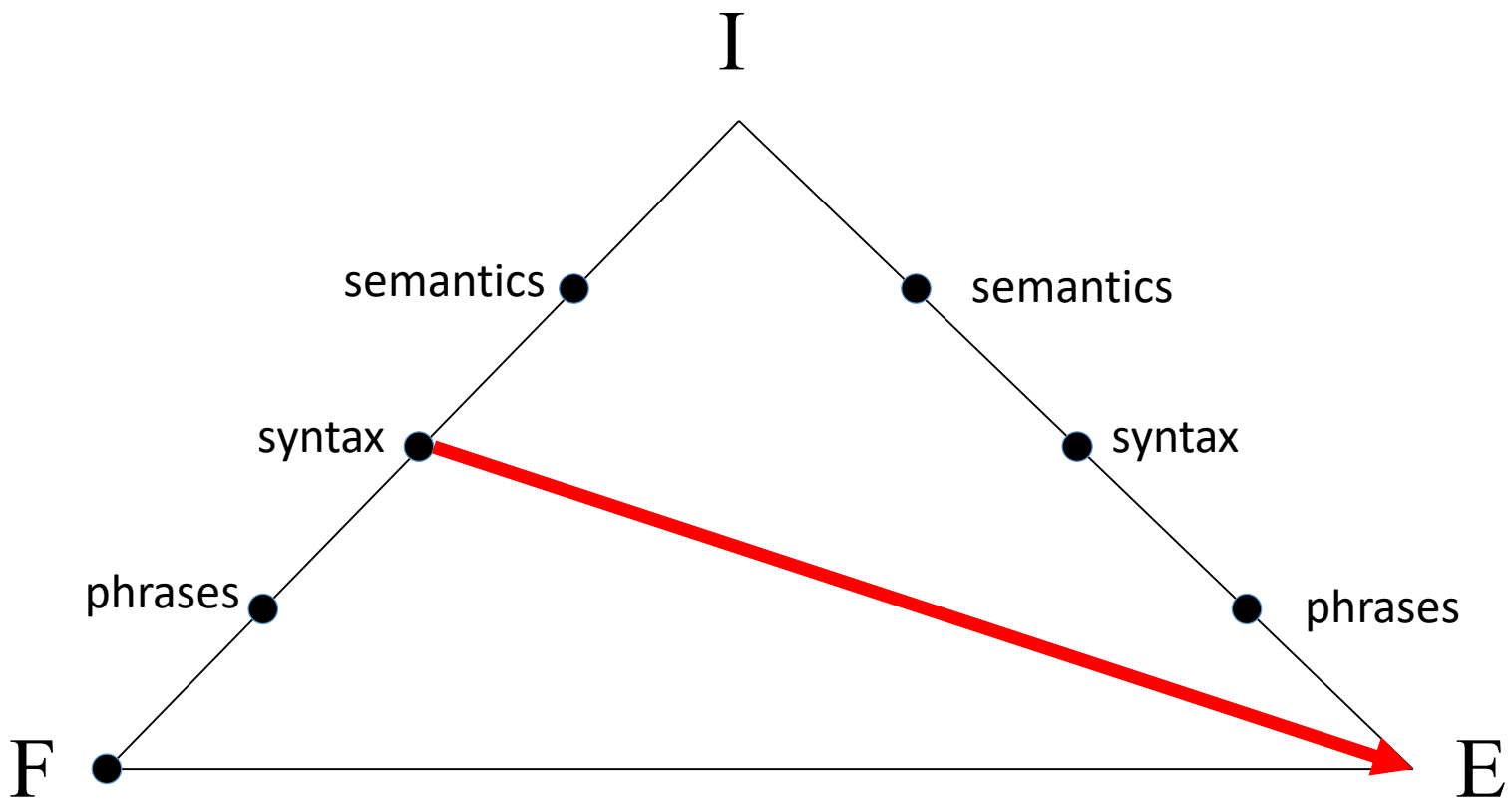
Phrase-Based Translation



Tree-to-Tree Translation



Tree-to-String Translation



Machine Translation

453.

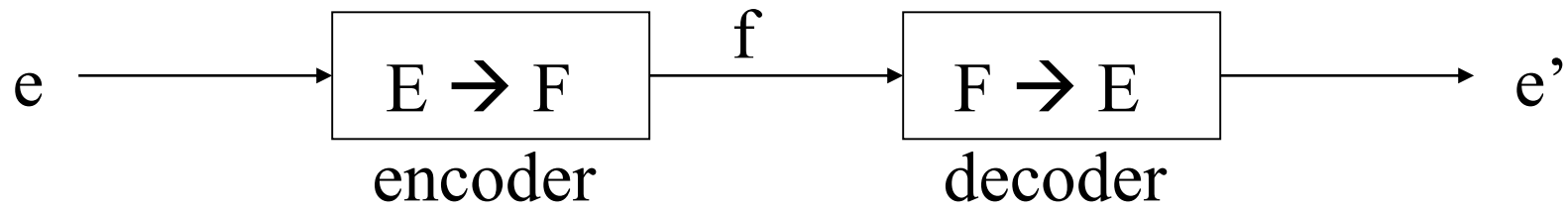
The Noisy Channel Model

The Noisy Channel Model

- Source-channel model of communication
- Parametric probabilistic models of language and translation

Statistics

- Given f, guess e



$$e' = \text{argmax}_e P(e|f) = \text{argmax}_e P(f|e)\ P(e)$$

translation model

language model

Statistical MT

Translate from French: “une fleur rouge”?

	p(e)	**p(f|e)**	**p(e)*p(f|e)**
a flower red	Low	High	Low
red flower a	Low	High	Low
flower red a	Low	High	Low
a red dog	High	Low	Low
dog cat mouse	Low	Low	Low
a red flower	High	High	High

Machine Translation

454.

The IBM Models

Motivation

- Difficult to model $p(f_1,\ldots,f_n)|(e_1,\ldots,e_m)$ directly

Questions

- If the word order is fixed
  - Align strings using the Levenshtein method
- What about the following:
  - How to deal with word reorderings?
  - How to deal with phrases?
- We need a systematic (and feasible) approach

Generative Story (almost IBM)

- I watched an interesting play
- I watched watched an interesting play play play
- I watched watched an play play play interesting
- J’ ai vu une pièce de théâtre intéressante

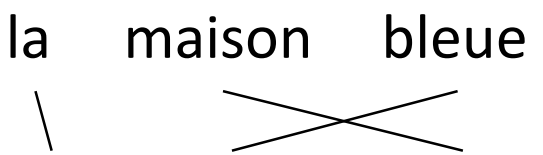
IBM’s EM trained models (1-5)

- Word translation
- Local alignment
- Fertilities
- Class-based alignment
- Non-deficient algorithm (avoid overlaps, overflow)

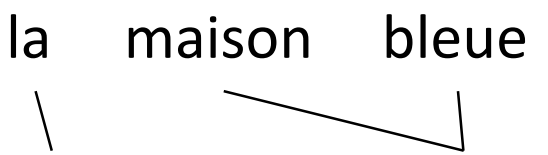
Alignments

uniform alignment 27=3x3x3 combinations

[1,2,3]

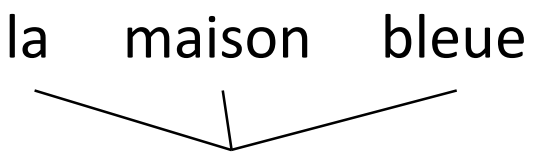


[1,3,2]

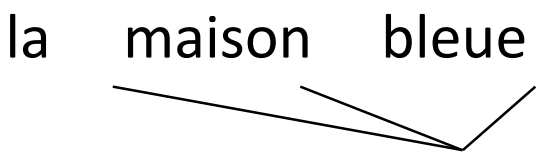


[1,3,3]

[1,1,1]



[2,2,2]

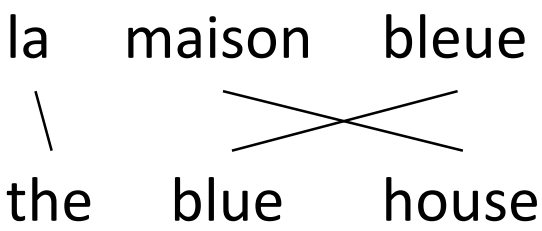


[3,3,3]

Model 1

- Alignments
  - La maison bleue
  - The blue house
  - Alignments: [1,2,3], [1,3,2], [1,3,3], [1,1,1], etc.
  - A priori, all are equally likely
- Conditional probabilities
  - $P(f|A,e) = ?$

Computing p(f|a,e)



[1,3,2]

P(f|a,e)=t(la|the) x t(bleu|blue) x t(maison|house)

Model 1 (cont'd)

- Algorithm
  - Pick length of translation (uniform probability)
  - Choose an alignment (uniform probability)
  - Translate the foreign words (only depends on the word)
  - That gives you P(f,A|e)
  - We need P(f|A,e)
  - Use EM (expectation-maximization) to find the hidden variables

Model 1 (cont'd)

- Length probability

$$p(m|e)=c$$

- Alignment probability (uniform)

$$p(a_i|e)=\frac{1}{(n+1)}$$

- Translation probability

$$p(f_i|e_{a_i})$$

$$p(f,a|e)=p(a|e)*p(f|a,e)=\frac{c}{(n+1)^m}\prod_{j=1}^{m}p(f_j|e_{a_j})$$

Finding the Optimal Alignment

$$\hat{a} = \arg\max_{a} p(f, a \mid e)$$

$$\hat{a} = \arg\max_{a} \frac{c}{(n+1)^m} \prod_{j=1}^{m} p(f_j \mid e_{a_j})$$

$$\hat{a} = \arg\max_{a} \prod_{j=1}^{m} p(f_j \mid e_{a_j})$$

$$\hat{a}_j = \arg\max_{a_j} p(f_j \mid e_{a_j})$$

Training Model 1

- Goal:
  - Learn the translation probabilities p(f|e)
- EM Algorithm
  - Used to estimate the translation probabilities from a training corpus
  - Guess p(f|e) (could be uniform)
  - Repeat until convergence:
    - E-step: compute counts
    - M-step: recompute p(f|e)

Example

Corpus:

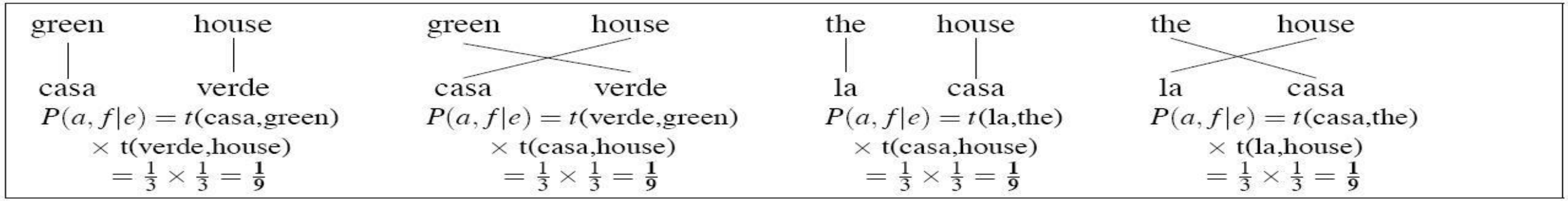
green house	the house
casa verde	la casa

Uniform translation model:

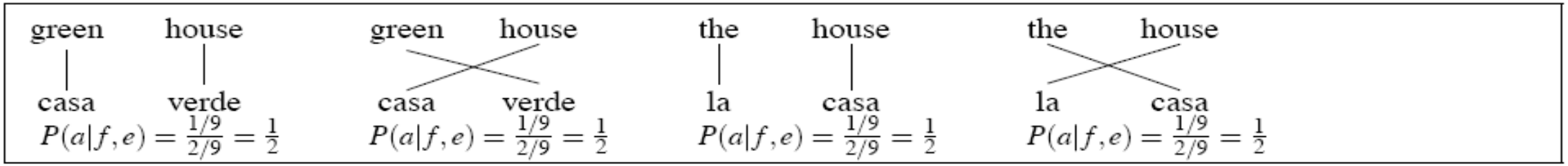
$t(\text{casa}\mid\text{green}) = \frac{1}{3}$		$t(\text{verde}\mid\text{green}) = \frac{1}{3}$		$t(\text{la}\mid\text{green}) = \frac{1}{3}$	
$t(\text{casa}\mid\text{house}) = \frac{1}{3}$		$t(\text{verde}\mid\text{house}) = \frac{1}{3}$		$t(\text{la}\mid\text{house}) = \frac{1}{3}$	
$t(\text{casa}\mid\text{the}) = \frac{1}{3}$		$t(\text{verde}\mid\text{the}) = \frac{1}{3}$		$t(\text{la}\mid\text{the}) = \frac{1}{3}$	

E-step 1: compute the expected counts E[count(t(f|e))] for all word pairs (f_j,e_{aj})

E-step 1a: compute P(a,f|e) by multiplying all *t* probabilities using $P(A,F|E) = \prod_{j=1}^{J} t(f_j|e_{a_j})$



E-step 1b: normalize P(a,f|e) to get P(a|e,f) using $P(a|e,f) = \frac{P(a,f|e)}{\sum_a P(a,f|e)}$



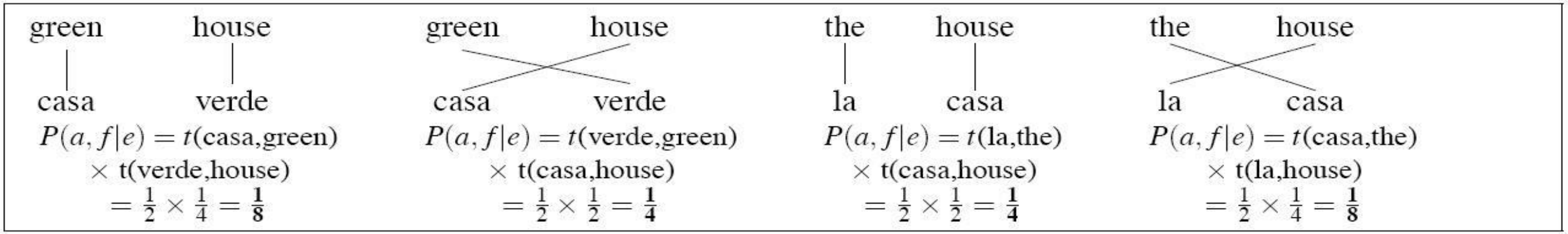
E-step 1c: compute expected fractional counts, by weighting each count by P(a|e,f)

tcount(casa|green) = $\frac{1}{2}$	tcount(verde|green) = $\frac{1}{2}$	tcount(la|green) = 0 total(green) = 1
tcount(casa|house) = $\frac{1}{2} + \frac{1}{2}$	tcount(verde|house) = $\frac{1}{2}$	tcount(la|house) = $\frac{1}{2}$ total(house) = 2
tcount(casa|the) = $\frac{1}{2}$	tcount(verde|the) = 0	tcount(la|the) = $\frac{1}{2}$ total(the) = 1

M-step 1: Compute the MLE probability params by normalizing the tcounts to sum to 1.

$t(\text{casa}\mid\text{green}) = \frac{1/2}{1} = \frac{1}{2}$	$t(\text{verde}\mid\text{green}) = \frac{1/2}{1} = \frac{1}{2}$	$t(\text{la}\mid\text{green}) = \frac{0}{1} = 0$
$t(\text{casa}\mid\text{house}) = \frac{1}{2} = \frac{1}{2}$	$t(\text{verde}\mid\text{house}) = \frac{1/2}{2} = \frac{1}{4}$	$t(\text{la}\mid\text{house}) = \frac{1/2}{2} = \frac{1}{4}$
$t(\text{casa}\mid\text{the}) = \frac{1/2}{1} = \frac{1}{2}$	$t(\text{verde}\mid\text{the}) = \frac{0}{1} = 0$	$t(\text{la}\mid\text{the}) = \frac{1/2}{1} = \frac{1}{2}$

E-step 2a: Recompute P(a,f|e) again by multiplying the *t* probabilities



More iterations are needed (until convergence)

```
import itertools

corpus = [('green house','casa verde'),('the house','la casa')]
# Print corpus:
vocab1 = []
vocab2 = []
print "Sentence pairs"
for i in range(len(corpus)):
    tup = corpus[i]
    print i,
    print '%s\t%s' % tup
    vocab1 += tup[0].split()
    vocab2 += tup[1].split()

# Print Vocabulary
vocab1 = list(set(vocab1))
vocab2 = list(set(vocab2))
print
print "Vocabulary"
print "Source Language:",
print vocab1
print "Target Language:",
print vocab2

print
print "EM initialization"
prob = {}
for w in vocab1:
    for v in vocab2:
        prob[(w,v)] = 1. / len(vocab2)
        print "P(%s|%s) = %.2f\t" % (v,w,prob[(w,v)]),
    print
```

Code by Rui Zhang

```
def E_step(prob):
    print "E_step"
    def compute_align(a,sent_pair):
        print "\t Alignment:",
        p = 1.
        s = sent_pair[0].split()
        t = sent_pair[1].split()
        for i in range(len(a)):
            w = s[i]
            v = t[a[i]]
            print (w,v),
            p = p * prob[(w,v)]
        print
        print "\t p(a,f|e): %.2f" % p
        return p

    new_prob = {}
    for w in vocab1:
        for v in vocab2:
            new_prob[(w,v)] = 0.

    for i in range(len(corpus)):
        print "Sentence Pair",i
        sent_pair = corpus[i]
        sent_l = len(sent_pair)
        total_i = []
        for a in itertools.permutations(range(sent_l)):
            total_i.append(compute_align(a, sent_pair))
        #normalize
        #print "\tp(a,f|e):",total_i
        total_i_sum = sum(total_i)
        total_i = [t / total_i_sum for t in total_i]
        print "\n\t Normalizing"
        print "\t p(a|e,f):",total_i
        print
        s = sent_pair[0].split()
        t = sent_pair[1].split()
        cnt = 0
        for a in itertools.permutations(range(sent_l)):
            for j in range(len(a)):
                w = s[j]
                v = t[a[j]]
                new_prob[(w,v)] += total_i[cnt]
            cnt += 1

    for w in vocab1:
        total_w = 0.
        for v in vocab2:
            total_w += new_prob[(w,v)]
            print "P(%s|%s) = %.2f\t" % (v,w,new_prob[(w,v)]),
        print "total(%s) = %2.f" % (w,total_w)

    return new_prob
```

```
def M_step(prob):
    print "M_step"
    for w in vocab1:
        total_w = sum([prob[w,v] for v in vocab2])
        for v in vocab2:
            prob[(w,v)] = prob[(w,v)] / total_w
            print "P(%s|%s) = %.2f\t" % (v,w,prob[(w,v)]),
        print
    return prob

for i in range(0,10):
    print "step: ",i
    prob = E_step(prob)
    prob = M_step(prob)
```

```
Sentence pairs
0 green house   casa verde
1 the house     la casa

Vocabulary
Source Language: ['house', 'the', 'green']
Target Language: ['verde', 'casa', 'la']
```

EM initialization

```
P(verde|house) = 0.33   P(casa|house) = 0.33    P(la|house) = 0.33
P(verde|the) = 0.33     P(casa|the) = 0.33      P(la|the) = 0.33
P(verde|green) = 0.33   P(casa|green) = 0.33    P(la|green) = 0.33
```

```
step:  0

E_step

Sentence Pair 0
        Alignment: ('green', 'casa') ('house', 'verde')
        p(a,f|e): 0.11
        Alignment: ('green', 'verde') ('house', 'casa')
        p(a,f|e): 0.11

        Normalizing
        p(a|e,f): [0.5, 0.5]

Sentence Pair 1
        Alignment: ('the', 'la') ('house', 'casa')
        p(a,f|e): 0.11
        Alignment: ('the', 'casa') ('house', 'la')
        p(a,f|e): 0.11

        Normalizing
        p(a|e,f): [0.5, 0.5]

P(verde|house) = 0.50   P(casa|house) = 1.00    P(la|house) = 0.50      total(house) =  2
P(verde|the) = 0.00     P(casa|the) = 0.50      P(la|the) = 0.50        total(the) =  1
P(verde|green) = 0.50   P(casa|green) = 0.50    P(la|green) = 0.00      total(green) =  1

M_step

P(verde|house) = 0.25   P(casa|house) = 0.50    P(la|house) = 0.25
P(verde|the) = 0.00     P(casa|the) = 0.50      P(la|the) = 0.50
P(verde|green) = 0.50   P(casa|green) = 0.50    P(la|green) = 0.00
```

```
step:  1

E_step

Sentence Pair 0
        Alignment: ('green', 'casa') ('house', 'verde')
        p(a,f|e): 0.12
        Alignment: ('green', 'verde') ('house', 'casa')
        p(a,f|e): 0.25

        Normalizing
        p(a|e,f): [0.3333333333333333, 0.6666666666666666]

Sentence Pair 1
        Alignment: ('the', 'la') ('house', 'casa')
        p(a,f|e): 0.25
        Alignment: ('the', 'casa') ('house', 'la')
        p(a,f|e): 0.12

        Normalizing
        p(a|e,f): [0.6666666666666666, 0.3333333333333333]

P(verde|house) = 0.33   P(casa|house) = 1.33    P(la|house) = 0.33      total(house) =  2
P(verde|the) = 0.00     P(casa|the) = 0.33      P(la|the) = 0.67        total(the) =  1
P(verde|green) = 0.67   P(casa|green) = 0.33    P(la|green) = 0.00      total(green) =  1

M_step

P(verde|house) = 0.17   P(casa|house) = 0.67    P(la|house) = 0.17
P(verde|the) = 0.00     P(casa|the) = 0.33      P(la|the) = 0.67
P(verde|green) = 0.67   P(casa|green) = 0.33    P(la|green) = 0.00
```

```
step:  2

E_step

Sentence Pair 0
        Alignment: ('green', 'casa') ('house', 'verde')
        p(a,f|e): 0.06
        Alignment: ('green', 'verde') ('house', 'casa')
        p(a,f|e): 0.44

        Normalizing
        p(a|e,f): [0.11111111111111112, 0.888888888888889]

Sentence Pair 1
        Alignment: ('the', 'la') ('house', 'casa')
        p(a,f|e): 0.44
        Alignment: ('the', 'casa') ('house', 'la')
        p(a,f|e): 0.06

        Normalizing
        p(a|e,f): [0.888888888888889, 0.11111111111111112]

P(verde|house) = 0.11   P(casa|house) = 1.78    P(la|house) = 0.11      total(house) =  2
P(verde|the) = 0.00     P(casa|the) = 0.11      P(la|the) = 0.89        total(the) =  1
P(verde|green) = 0.89   P(casa|green) = 0.11    P(la|green) = 0.00      total(green) =  1

M_step

P(verde|house) = 0.06   P(casa|house) = 0.89    P(la|house) = 0.06
P(verde|the) = 0.00     P(casa|the) = 0.11      P(la|the) = 0.89
P(verde|green) = 0.89   P(casa|green) = 0.11    P(la|green) = 0.00
```

```
step:  3

E_step

Sentence Pair 0
        Alignment: ('green', 'casa') ('house', 'verde')
        p(a,f|e): 0.01
        Alignment: ('green', 'verde') ('house', 'casa')
        p(a,f|e): 0.79

        Normalizing
        p(a|e,f): [0.007751937984496124, 0.9922480620155039]

Sentence Pair 1
        Alignment: ('the', 'la') ('house', 'casa')
        p(a,f|e): 0.79
        Alignment: ('the', 'casa') ('house', 'la')
        p(a,f|e): 0.01

        Normalizing
        p(a|e,f): [0.9922480620155039, 0.007751937984496124]

P(verde|house) = 0.01   P(casa|house) = 1.98    P(la|house) = 0.01      total(house) =  2
P(verde|the) = 0.00     P(casa|the) = 0.01      P(la|the) = 0.99        total(the) =  1
P(verde|green) = 0.99   P(casa|green) = 0.01    P(la|green) = 0.00      total(green) =  1

M_step

P(verde|house) = 0.00   P(casa|house) = 0.99    P(la|house) = 0.00
P(verde|the) = 0.00     P(casa|the) = 0.01      P(la|the) = 0.99
P(verde|green) = 0.99   P(casa|green) = 0.01    P(la|green) = 0.00
```

```
corpus = [('green house','casa verde'),('the house','la casa'),('my house','mi casa')]
Sentence pairs
0 green house   casa verde
1 the house     la casa
2 my house      mi casa

Vocabulary
Source Language: ['house', 'the', 'green', 'my']
Target Language: ['mi', 'verde', 'casa', 'la']

EM initialization

P(mi|house) = 0.25      P(verde|house) = 0.25   P(casa|house) = 0.25    P(la|house) = 0.25
P(mi|the) = 0.25        P(verde|the) = 0.25     P(casa|the) = 0.25      P(la|the) = 0.25
P(mi|green) = 0.25      P(verde|green) = 0.25   P(casa|green) = 0.25    P(la|green) = 0.25
P(mi|my) = 0.25         P(verde|my) = 0.25      P(casa|my) = 0.25       P(la|my) = 0.25

step:  0

E_step

Sentence Pair 0
        Alignment: ('green', 'casa') ('house', 'verde')
        p(a,f|e): 0.06
        Alignment: ('green', 'verde') ('house', 'casa')
        p(a,f|e): 0.06
        Normalizing
        p(a|e,f): [0.5, 0.5]

Sentence Pair 1
        Alignment: ('the', 'la') ('house', 'casa')
        p(a,f|e): 0.06
        Alignment: ('the', 'casa') ('house', 'la')
        p(a,f|e): 0.06
        Normalizing
        p(a|e,f): [0.5, 0.5]
```

```
Sentence Pair 2
        Alignment: ('my', 'mi') ('house', 'casa')
        p(a,f|e): 0.06
        Alignment: ('my', 'casa') ('house', 'mi')
        p(a,f|e): 0.06
        Normalizing
        p(a|e,f): [0.5, 0.5]

P(mi|house) = 0.50      P(verde|house) = 0.50   P(casa|house) = 1.50    P(la|house) = 0.50      total(house) =  3
P(mi|the) = 0.00        P(verde|the) = 0.00     P(casa|the) = 0.50      P(la|the) = 0.50        total(the) =  1
P(mi|green) = 0.00      P(verde|green) = 0.50   P(casa|green) = 0.50    P(la|green) = 0.00      total(green) =  1
P(mi|my) = 0.50         P(verde|my) = 0.00      P(casa|my) = 0.50       P(la|my) = 0.00         total(my) =  1

M_step

P(mi|house) = 0.17      P(verde|house) = 0.17   P(casa|house) = 0.50    P(la|house) = 0.17
P(mi|the) = 0.00        P(verde|the) = 0.00     P(casa|the) = 0.50      P(la|the) = 0.50
P(mi|green) = 0.00      P(verde|green) = 0.50   P(casa|green) = 0.50    P(la|green) = 0.00
P(mi|my) = 0.50         P(verde|my) = 0.00      P(casa|my) = 0.50       P(la|my) = 0.00

...

step:  3
...

M_step

P(mi|house) = 0.00      P(verde|house) = 0.00   P(casa|house) = 1.00    P(la|house) = 0.00
P(mi|the) = 0.00        P(verde|the) = 0.00     P(casa|the) = 0.00      P(la|the) = 1.00
P(mi|green) = 0.00      P(verde|green) = 1.00   P(casa|green) = 0.00    P(la|green) = 0.00
P(mi|my) = 1.00         P(verde|my) = 0.00      P(casa|my) = 0.00       P(la|my) = 0.00
```

```
corpus = [('green house','casa verde'),('the house','la casa'),('my house','mi casa'),('my houses','mis casas')]
Sentence pairs
0 green house   casa verde
1 the house     la casa
2 my house      mi casa
3 my houses     mis casas

Vocabulary
Source Language: ['house', 'the', 'green', 'my', 'houses']
Target Language: ['casa', 'la', 'mi', 'verde', 'casas', 'mis']

EM initialization
P(casa|house) = 0.17    P(la|house) = 0.17      P(mi|house) = 0.17      P(verde|house) = 0.17   P(casas|house) = 0.17   P(mis|house) = 0.17
P(casa|the) = 0.17      P(la|the) = 0.17        P(mi|the) = 0.17        P(verde|the) = 0.17     P(casas|the) = 0.17     P(mis|the) = 0.17
P(casa|green) = 0.17    P(la|green) = 0.17      P(mi|green) = 0.17      P(verde|green) = 0.17   P(casas|green) = 0.17   P(mis|green) = 0.17
P(casa|my) = 0.17       P(la|my) = 0.17         P(mi|my) = 0.17         P(verde|my) = 0.17      P(casas|my) = 0.17      P(mis|my) = 0.17
P(casa|houses) = 0.17   P(la|houses) = 0.17     P(mi|houses) = 0.17     P(verde|houses) = 0.17  P(casas|houses) = 0.17  P(mis|houses) =  0.17

step:  0

E_step
Sentence Pair 0
        Alignment: ('green', 'casa') ('house', 'verde')
        p(a,f|e): 0.03
        Alignment: ('green', 'verde') ('house', 'casa')
        p(a,f|e): 0.03

        Normalizing
        p(a|e,f): [0.5, 0.5]

Sentence Pair 1
        Alignment: ('the', 'la') ('house', 'casa')
        p(a,f|e): 0.03
        Alignment: ('the', 'casa') ('house', 'la')
        p(a,f|e): 0.03

        Normalizing
        p(a|e,f): [0.5, 0.5]
```

```
Sentence Pair 2
        Alignment: ('my', 'mi') ('house', 'casa')
        p(a,f|e): 0.03
        Alignment: ('my', 'casa') ('house', 'mi')
        p(a,f|e): 0.03

        Normalizing
        p(a|e,f): [0.5, 0.5]

Sentence Pair 3
        Alignment: ('my', 'mis') ('houses', 'casas')
        p(a,f|e): 0.03
        Alignment: ('my', 'casas') ('houses', 'mis')
        p(a,f|e): 0.03

        Normalizing
        p(a|e,f): [0.5, 0.5]

P(casa|house) = 1.50    P(la|house) = 0.50      P(mi|house) = 0.50      P(verde|house) = 0.50   P(casas|house) = 0.00   P(mis|house) = 0.00    total(house) =  3
P(casa|the) = 0.50      P(la|the) = 0.50        P(mi|the) = 0.00        P(verde|the) = 0.00     P(casas|the) = 0.00     P(mis|the) = 0.00      total(the) =  1
P(casa|green) = 0.50    P(la|green) = 0.00      P(mi|green) = 0.00      P(verde|green) = 0.50   P(casas|green) = 0.00   P(mis|green) = 0.00    total(green) =  1
P(casa|my) = 0.50       P(la|my) = 0.00         P(mi|my) = 0.50         P(verde|my) = 0.00      P(casas|my) = 0.50      P(mis|my) = 0.50       total(my) =  2
P(casa|houses) = 0.00   P(la|houses) = 0.00     P(mi|houses) = 0.00     P(verde|houses) = 0.00  P(casas|houses) = 0.50  P(mis|houses) = 0.50   total(houses) =  1

M_step

P(casa|house) = 0.50    P(la|house) = 0.17      P(mi|house) = 0.17      P(verde|house) = 0.17   P(casas|house) = 0.00   P(mis|house) = 0.00
P(casa|the) = 0.50      P(la|the) = 0.50        P(mi|the) = 0.00        P(verde|the) = 0.00     P(casas|the) = 0.00     P(mis|the) = 0.00
P(casa|green) = 0.50    P(la|green) = 0.00      P(mi|green) = 0.00      P(verde|green) = 0.50   P(casas|green) = 0.00   P(mis|green) = 0.00
P(casa|my) = 0.25       P(la|my) = 0.00         P(mi|my) = 0.25         P(verde|my) = 0.00      P(casas|my) = 0.25      P(mis|my) = 0.25
P(casa|houses) = 0.00   P(la|houses) = 0.00     P(mi|houses) = 0.00     P(verde|houses) = 0.00  P(casas|houses) = 0.50  P(mis|houses) = 0.50
```

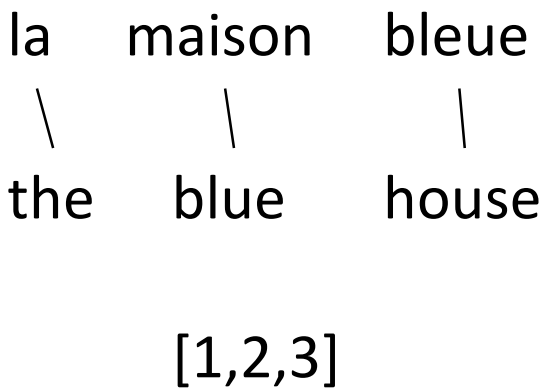
```
step 3:

M_step
P(casa|house) = 1.00    P(la|house) = 0.00      P(mi|house) = 0.00      P(verde|house) = 0.00   P(casas|house) = 0.00   P(mis|house) = 0.00
P(casa|the) = 0.00      P(la|the) = 1.00        P(mi|the) = 0.00        P(verde|the) = 0.00     P(casas|the) = 0.00     P(mis|the) = 0.00
P(casa|green) = 0.00    P(la|green) = 0.00      P(mi|green) = 0.00      P(verde|green) = 1.00   P(casas|green) = 0.00   P(mis|green) = 0.00
P(casa|my) = 0.00       P(la|my) = 0.00         P(mi|my) = 0.50         P(verde|my) = 0.00      P(casas|my) = 0.25      P(mis|my) = 0.25
P(casa|houses) = 0.00   P(la|houses) = 0.00     P(mi|houses) = 0.00     P(verde|houses) = 0.00  P(casas|houses) = 0.50  P(mis|houses) = 0.50
```

Model 2

- Distortion parameters d(i|j,l,m)
  - i and j are word positions in the two sentences
  - l and m are the lengths of these sentences
- Example
  - q(3|2,5,6)

Alignments



$$p(a|e) = q(2|1,3,3).\, q(1|2,3,3).\, q(3|3,3,3)$$

Model 2

- The distortion parameters are also learned by EM

$$p(f,a|e) = p(a|e)p(f|a,e) = \frac{c}{(n+1)^m}\prod_{j=1}^{m}[q(i|j,l,m)t\left(f_j\middle|e_{a_j}\right)]$$

$$p(f|e) = \sum_a p(f,a|e) = \sum_a p(a|e)p(f|a,e) = \sum_a \prod_{j=1}^{m}[q(i|j,l,m)t\left(f_j\middle|e_{a_j}\right)]$$

Model 3

- Fertility

  f(ϕ_i|e)

  f_0 is an extra parameter that defines ϕ_0

- Examples
  - program = programme $f(1|program) \approx 1$
  - NOUN play = pièce de théâtre $f(3|play\_N) \approx 1$
  - VERB place = mettre en place $f(3|place\_V) \approx 1$

should

f	$t(f \mid e)$	ϕ	$n(\phi \mid e)$
devrait	0.330	1	0.649
devraient	0.123	0	0.336
devrions	0.109	2	0.014
faudrait	0.073		
faut	0.058		
doit	0.058		
aurait	0.041		
doivent	0.024		
devons	0.017		
devrais	0.013		

Figure 7
Translation and fertility probabilities for *should*.

national

f	$t(f \mid e)$	ϕ	$n(\phi \mid e)$
nationale	0.469	1	0.905
national	0.418	0	0.094
nationaux	0.054		
nationales	0.029		

Figure 8
Translation and fertility probabilities for *national*.

the

f	$t(f \mid e)$	ϕ	$n(\phi \mid e)$
le	0.497	1	0.746
la	0.207	0	0.254
les	0.155		
l′	0.086		
ce	0.018		
cette	0.011		

Figure 9
Translation and fertility probabilities for *the*.

farmers

f	$t(f \mid e)$	ϕ	$n(\phi \mid e)$
agriculteurs	0.442	2	0.731
les	0.418	1	0.228
cultivateurs	0.046	0	0.039
producteurs	0.021		

Figure 10
Translation and fertility probabilities for *farmers*.

not

f	$t(f \mid e)$	ϕ	$n(\phi \mid e)$
ne	0.497	2	0.735
pas	0.442	0	0.154
non	0.029	1	0.107
rien	0.011		

Figure 15
Translation and fertility probabilities for *not*.

[Brown et al. 1993]

References

- http://www.isi.edu/natural-language/mt/wkbk.rtf

  (an awesome tutorial by Kevin Knight)

- http://www.statmt.org/

  (a comprehensive site, including references to the old IBM papers, pointers to Moses, etc.)

Machine Translation

455.

Syntax in Machine Translation

Notes

- Bilingual CKY parsing

	e1	e2	e3	e5	e6	e7	e8
f1					N/N		
f2				V/V			
f3			N/N				
f4							

Notes

- Bilingual CKY parsing

	e1	e2	e3	e5	e6	e7	e8
f1					NP/NP		
f2				V/V			
f3	NP/NP						
f4							

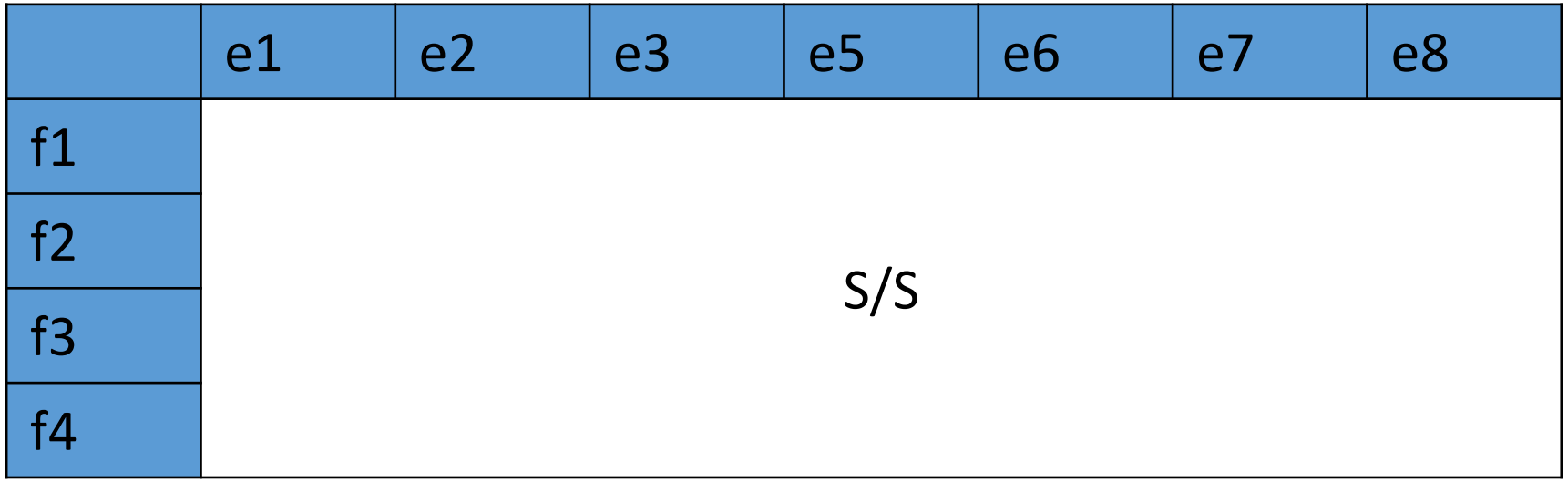
Notes

- Bilingual CKY parsing

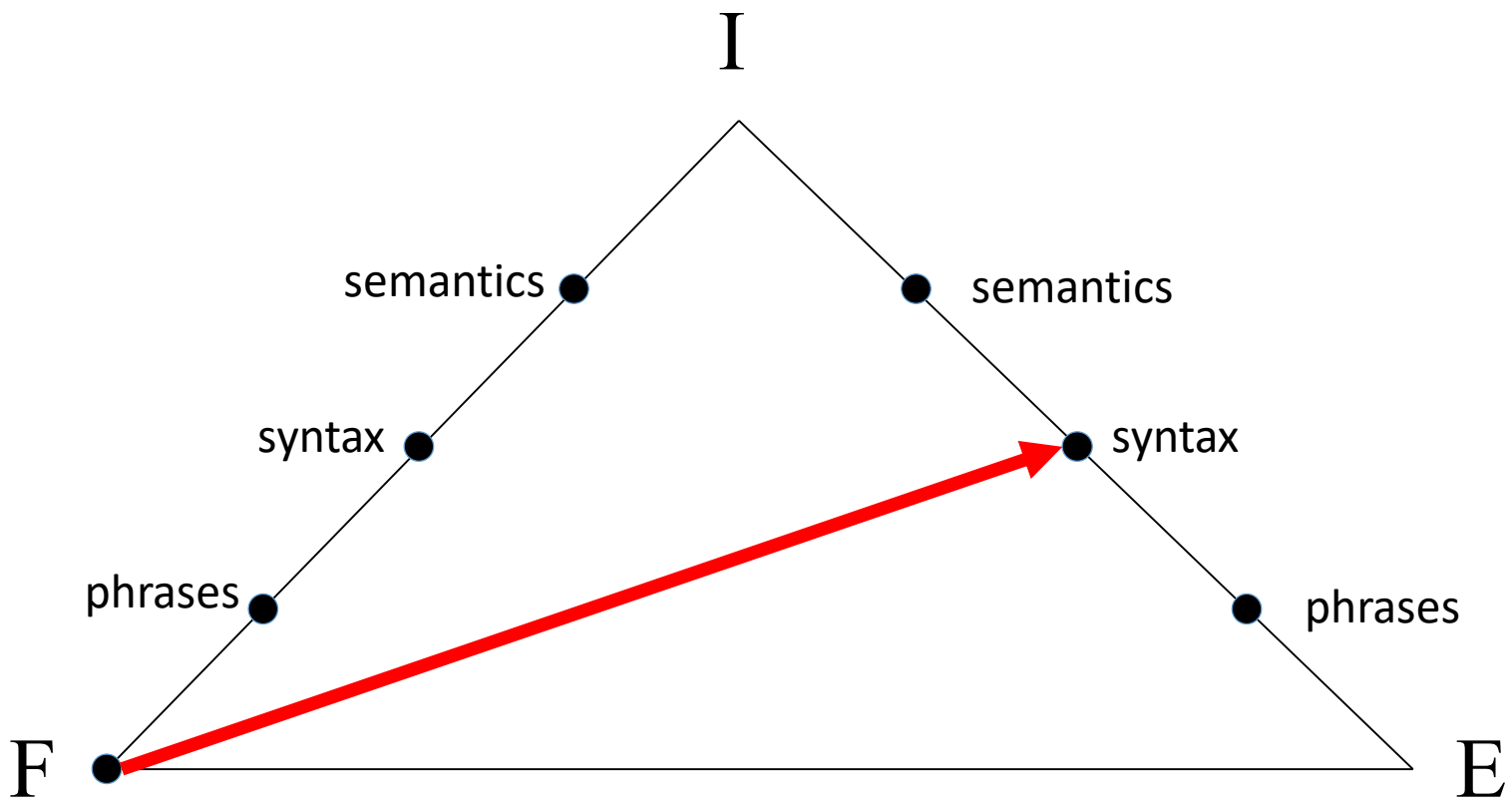
<table>
<tr><th></th><th>e1</th><th>e2</th><th>e3</th><th>e5</th><th>e6</th><th>e7</th><th>e8</th></tr>
<tr><td>f1</td><td></td><td></td><td></td><td colspan="4" rowspan="2">VP/VP</td></tr>
<tr><td>f2</td><td></td><td></td><td></td></tr>
<tr><td>f3</td><td colspan="3" rowspan="2">NP/NP</td><td></td><td></td><td></td><td></td></tr>
<tr><td>f4</td><td></td><td></td><td></td><td></td></tr>
</table>

Notes

- Bilingual CKY parsing



String to Tree Translation



(Yamada and Knight 2001)

String to Tree Translation

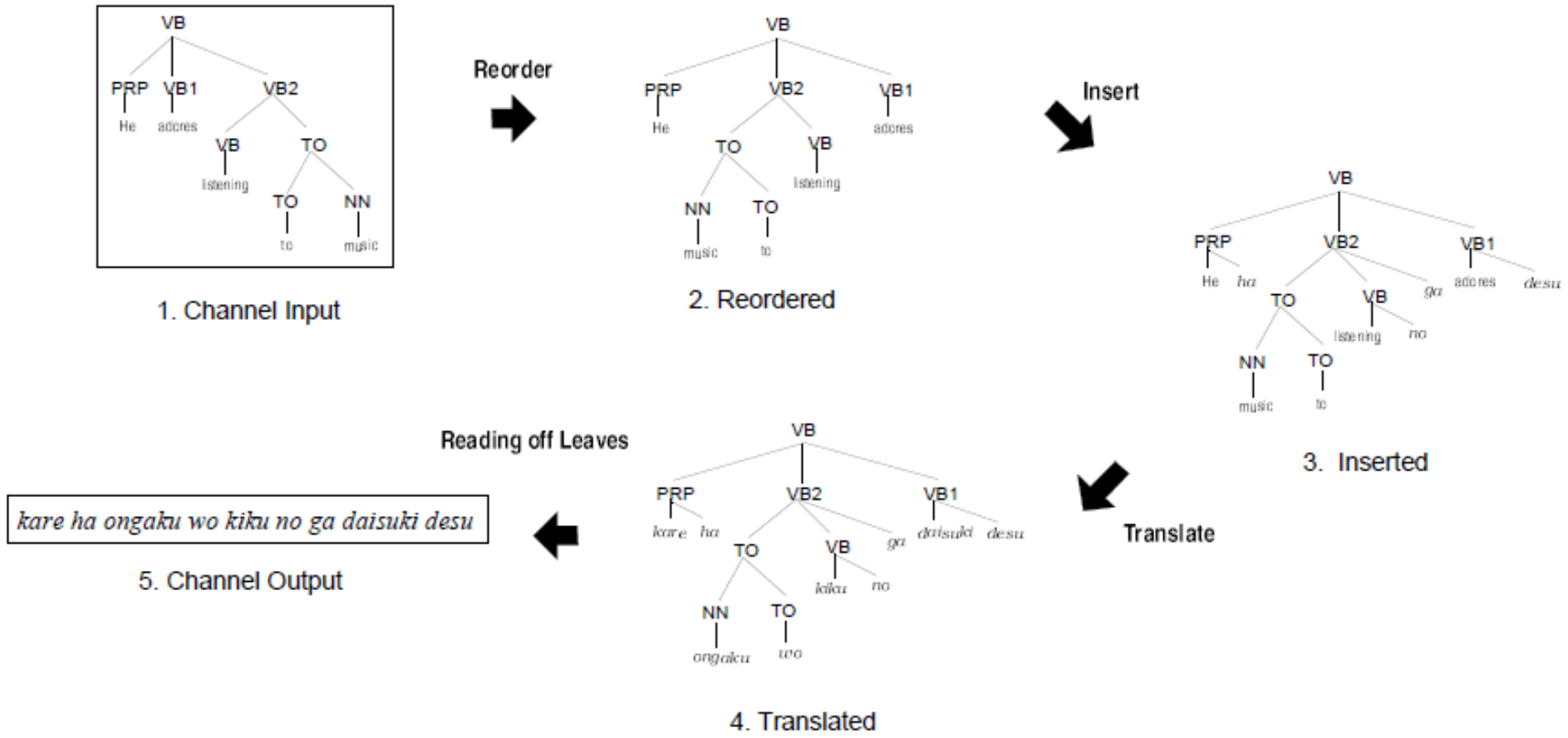


Figure 1: Channel Operations: Reorder, Insert, and Translate

(Yamada and Knight 2001)

Synchronous Grammars

- Generate parse trees in parallel in two languages using different rules
- E.g.,
  - NP -> ADJ N (in English)
  - NP -> N ADJ (in Spanish)
- ITG (Inversion Transduction Grammar) [Wu 1995]
  - Don’t allow all permutations in derivations
  - Only <> and [ ] are allowed

Machine Translation

456.

Evaluation of Machine Translation

Evaluation

- Human judgments
  - adequacy
  - grammaticality
  - [expensive]
- Automatic methods
  - Edit cost (at the word, character, or minute level)
  - BLEU (Papineni et al. 2002)

BLEU

- Simple n-gram precision

  log BLEU = min (0,1-reflen/candlen) + mean of log precisions

- Multiple human references
- Brevity penalty
- Correlates with human assessments of automatic systems
- Doesn’t correlate well when comparing human and automatic translations

Example from MTC

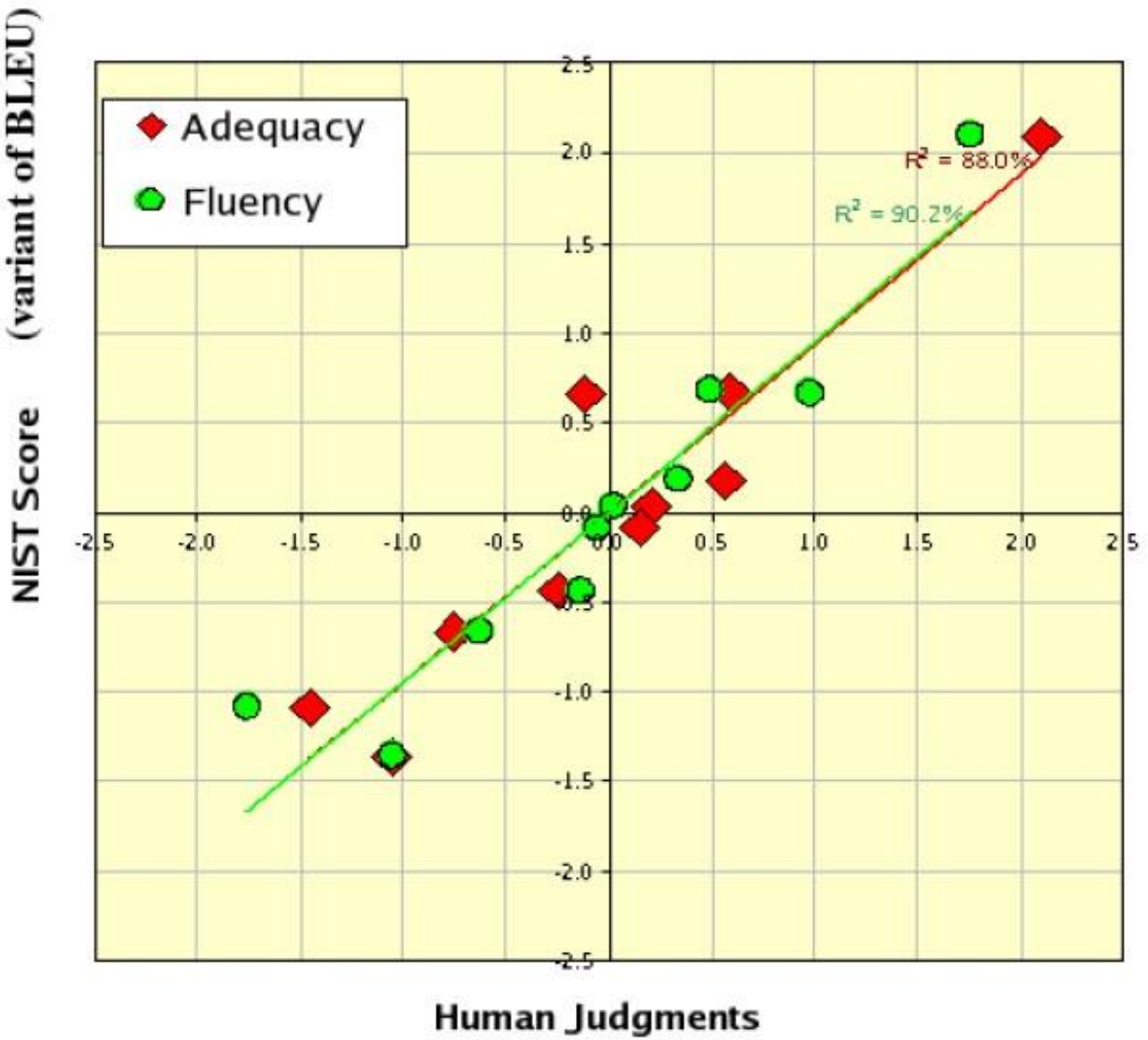
- Chinese:
  - Napster执行长希尔柏斯辞职
- English
  - Napster CEO Hilbers Resigns
  - Napster CEO Hilbers resigned
  - Napster Chief Executive Hilbers Resigns
  - Napster CEO Konrad Hilbers resigns
- Full text
  - http://clair.si.umich.edu/~radev/nlp/mtc/

“Good” Compared to What?

- Idea #1: a human translation.
  - OK, but
    - Good translations can be very dissimilar
    - We’d need to find hidden features (e.g. alignments)
- Idea #2: other top *n* translations (the “n-best list”).
  - Better in practice, but
    - Many entries in n-best list are the same apart from hidden links
- Compare with a **loss function** *L*
  - 0/1: wrong or right; equal to reference or not
  - Task-specific metrics (word error rate, BLEU, …)

[Example from Schafer&Smith 2006]

Correlation: BLEU and Humans



[Example from Doddington 2002]

Tools for Machine Translation

- Language modeling toolkits
  - SRILM, CMULM
- Translation systems
  - Giza++, Moses
- Decoders
  - Pharaoh

Deep Learning

753.

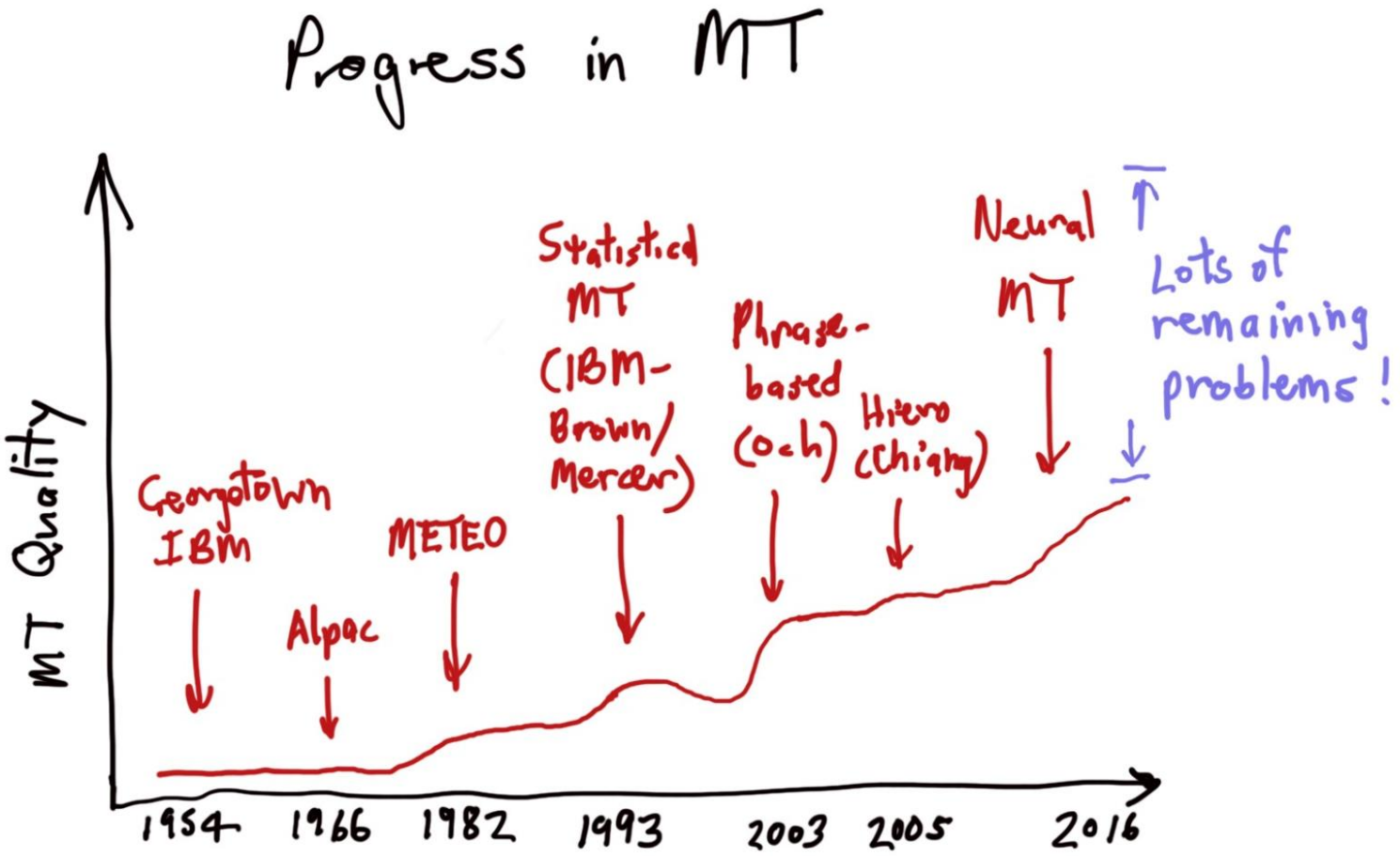
Sequence-to-sequence Methods and

Neural Machine Translation

Machine Translation (MT)

- Goal: Translate text from one language to the other
  - Natural Language Understanding
  - Natural Language Generation

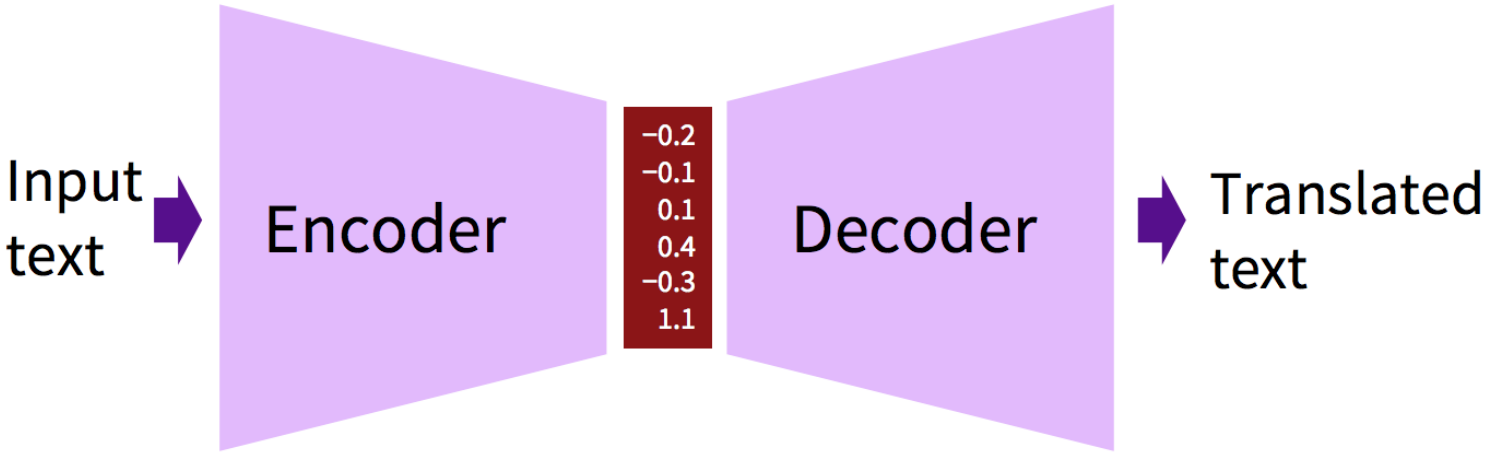
Progress in Machine Translation



https://nlp.stanford.edu/projects/nmt/Luong-Cho-Manning-NMT-ACL2016-v4.pdf

Neural Machine Translation

- Modeling the machine translation using neural networks
  - Encoder for language understanding in source language
  - Decoder for language generation in target language

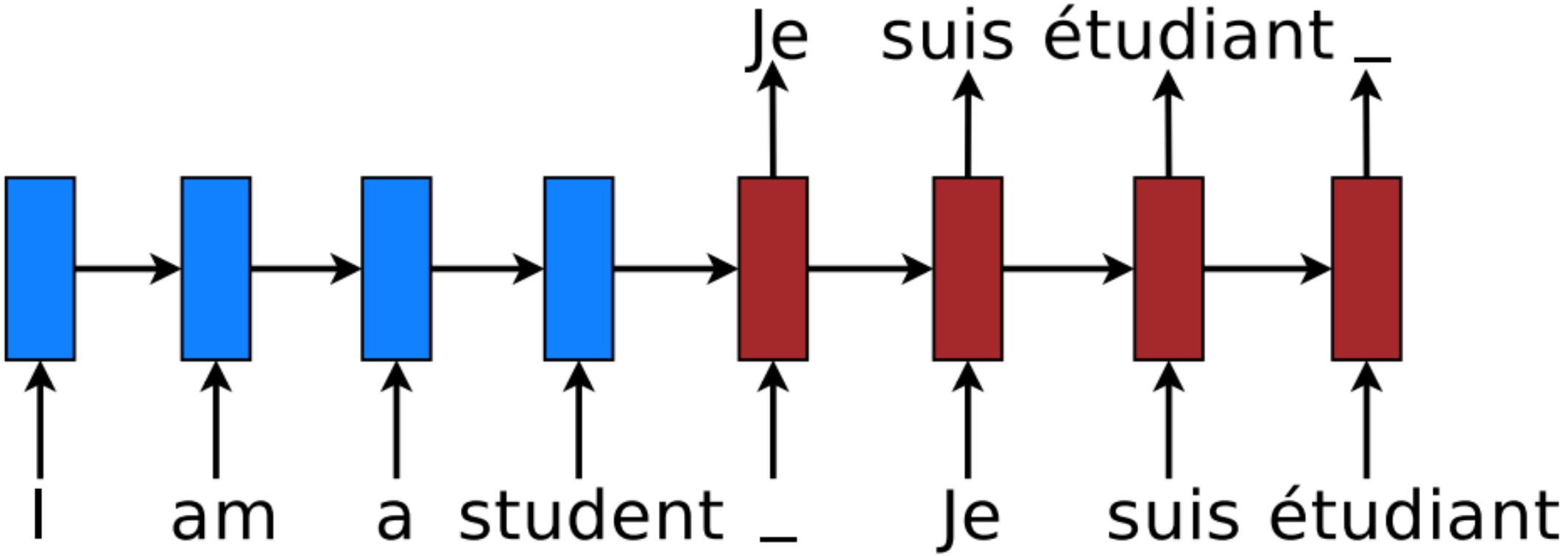


https://nlp.stanford.edu/projects/nmt/Luong-Cho-Manning-NMT-ACL2016-v4.pdf

RNN as Encoder and Decoder

- MT is essentially a task in which some structured input gets converted to a structured output
- A Recurrent Neural Network, or RNN, is a network that operates on a sequence and uses its own output as input for subsequent steps.
- A sequence-to-sequence network, or Encoder Decoder network, is a model consisting of two RNNs called the encoder and the decoder.
- The encoder reads an input sequence and outputs vectors, and the decoder reads encoder outputs as input to produce an output sequence.

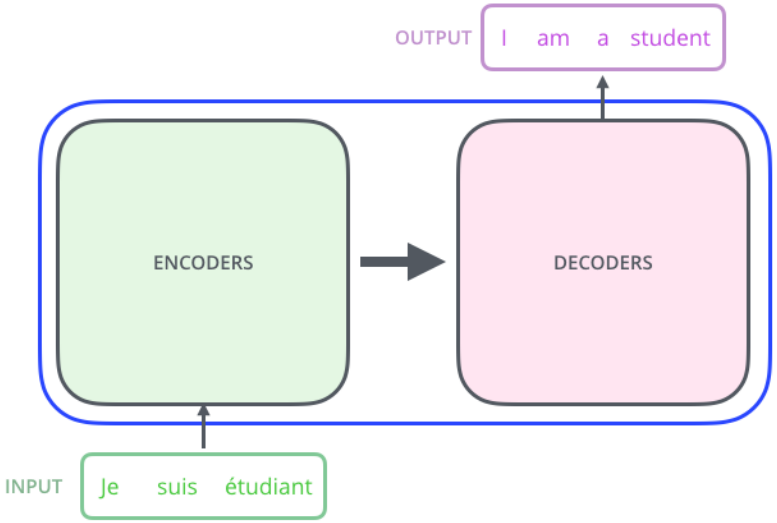
Use RNNs as Encoder and Decoder



https://nlp.stanford.edu/pubs/luong2016iclr_multi.pdf

Encoder-Decoder Architectures

- Encoder-decoder structure - used by most competitive neural sequence transduction models
- Encoder maps an input sequence of symbol representations $(x_1, ..., x_n)$ to a sequence of continuous representations $(y_1, ..., y_m)$
- Given **z**, the decoder then generates an output sequence $\mathbf{z} = (z_1, ..., z_n)$ of symbols one element at a time
- Auto-regressive at each step



WMT Dataset

- The academic benchmark for machine translation
- Most benchmark NMT papers use the WMT 2014 training dataset on En-Fr and/or En-De translation tasks
- WMT 2015 benchmark does exist but less frequently used
- ~4 million sentence pairs for En-De, ~12M for En-Fr

The Idea of Seq2Seq

- Use one LSTM to read the input sequence one time-step at a time -> obtain large fixed-dimensional vector representation
- Use another LSTM to extract the output sequence from that vector
  - This is essentially an RNN language model but conditioned on the input sequence
- Why LSTM?
  - Ability to learn on data with long range temporal dependencies

Miscellaneous Details

- Use simple left-to-right beam search to find most likely translation
- Source sentences are fed into the encoder in reverse

Results

Method	test BLEU score (ntst14)
Bahdanau et al. [2]	28.45
Baseline System [29]	33.30
Single forward LSTM, beam size 12	26.17
Single reversed LSTM, beam size 12	30.59
Ensemble of 5 reversed LSTMs, beam size 1	33.00
Ensemble of 2 reversed LSTMs, beam size 12	33.27
Ensemble of 5 reversed LSTMs, beam size 2	34.50
Ensemble of 5 reversed LSTMs, beam size 12	**34.81**

Table 1: The performance of the LSTM on WMT'14 English to French test set (ntst14). Note that an ensemble of 5 LSTMs with a beam of size 2 is cheaper than of a single LSTM with a beam of size 12.

Image: Table 1, Sequence to Sequence Learning with Neural Networks

Results

Method	test BLEU score (ntst14)
Baseline System [29]	33.30
Cho et al. [5]	34.54
Best WMT'14 result [9]	**37.0**
Rescoring the baseline 1000-best with a single forward LSTM	35.61
Rescoring the baseline 1000-best with a single reversed LSTM	35.85
Rescoring the baseline 1000-best with an ensemble of 5 reversed LSTMs	**36.5**
Oracle Rescoring of the Baseline 1000-best lists	~45

Table 2: Methods that use neural networks together with an SMT system on the WMT'14 English to French test set (ntst14).

Image: Table 2, Sequence to Sequence Learning with Neural Networks

Sequence to Sequence Model

Type	Sentence
Our model	Ulrich UNK , membre du conseil d' administration du constructeur automobile Audi , affirme qu' il s' agit d' une pratique courante depuis des années pour que les téléphones portables puissent être collectés avant les réunions du conseil d' administration afin qu' ils ne soient pas utilisés comme appareils d' écoute à distance .
Truth	Ulrich Hackenberg , membre du conseil d' administration du constructeur automobile Audi , déclare que la collecte des téléphones portables avant les réunions du conseil , afin qu' ils ne puissent pas être utilisés comme appareils d' écoute à distance , est une pratique courante depuis des années .
Our model	" Les téléphones cellulaires , qui sont vraiment une question , non seulement parce qu' ils pourraient potentiellement causer des interférences avec les appareils de navigation , mais nous savons , selon la FCC , qu' ils pourraient interférer avec les tours de téléphone cellulaire lorsqu' ils sont dans l' air " , dit UNK .
Truth	" Les téléphones portables sont véritablement un problème , non seulement parce qu' ils pourraient éventuellement créer des interférences avec les instruments de navigation , mais parce que nous savons , d' après la FCC , qu' ils pourraient perturber les antennes-relais de téléphonie mobile s' ils sont utilisés à bord " , a déclaré Rosenker .
Our model	Avec la crémation , il y a un " sentiment de violence contre le corps d' un être cher " , qui sera " réduit à une pile de cendres " en très peu de temps au lieu d' un processus de décomposition " qui accompagnera les étapes du deuil " .
Truth	Il y a , avec la crémation , " une violence faite au corps aimé " , qui va être " réduit à un tas de cendres " en très peu de temps , et non après un processus de décomposition , qui " accompagnerait les phases du deuil " .

Table 3: A few examples of long translations produced by the LSTM alongside the ground truth translations. The reader can verify that the translations are sensible using Google translate.

Sutskever, Le, and Vinyals 2014

Multilingual Embeddings

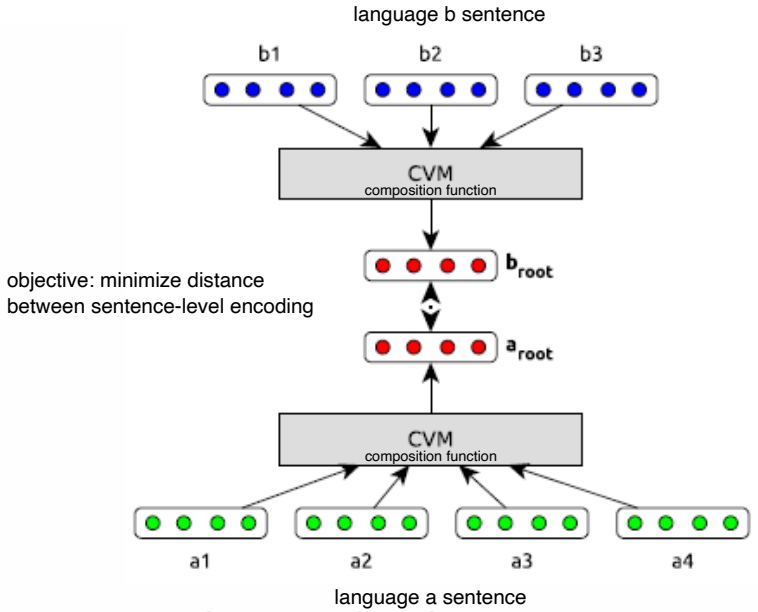


Figure 1: Description of a bilingual model with parallel input sentences a and b. The objective function of this model is to minimize the distance between the sentence level encoding of the bi-text. Principally any composition function can be used to generate the compositional sentence level representations. The composition function is represented by the *CVM* boxes in the diagram above.

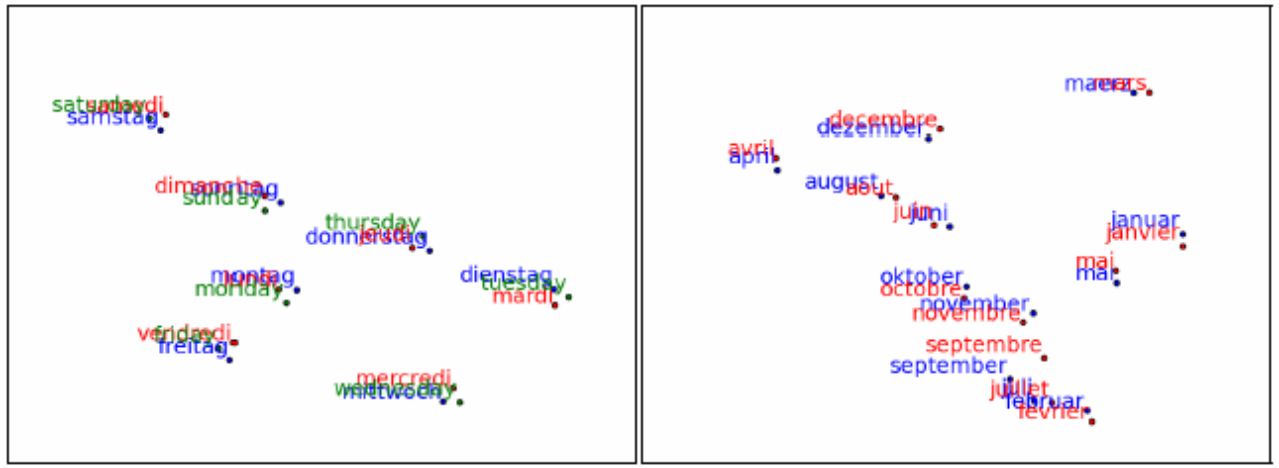
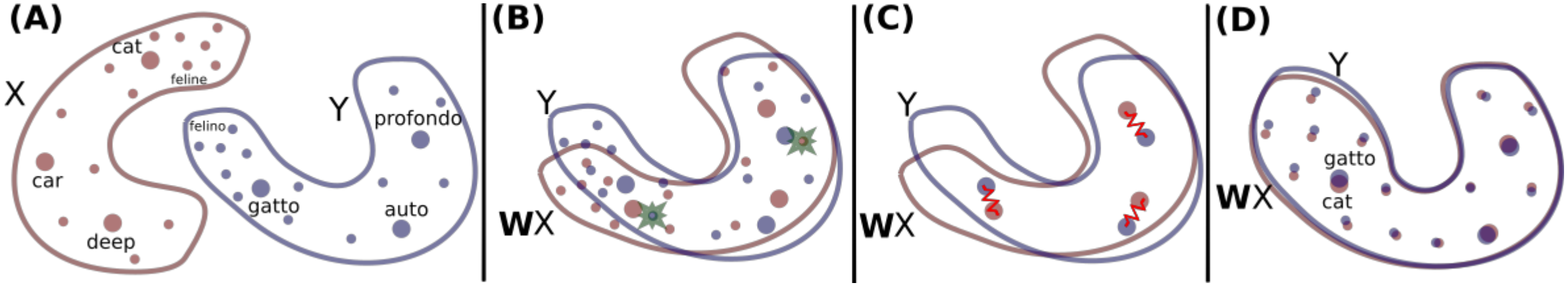


Figure 3: The left scatter plot shows t-SNE projections for a weekdays in all three languages using the representations learned in the BICVM+ model. Even though the model did not use any parallel French-German data during training, it still learns semantic similarity between these two languages using English as a pivot. To highlight this, the right plot shows another set of words (months of the year) using only the German and French words.

[Hermann and Blunsom 2013: https://arxiv.org/abs/1312.6173]

Multilingual Embeddings



https://engineering.fb.com/ml-applications/under-the-hood-multilingual-embeddings/ (demo)

https://github.com/facebookresearch/MUSE

https://github.com/facebookresearch/MUSE/blob/master/demo.ipynb

Beam Decoding

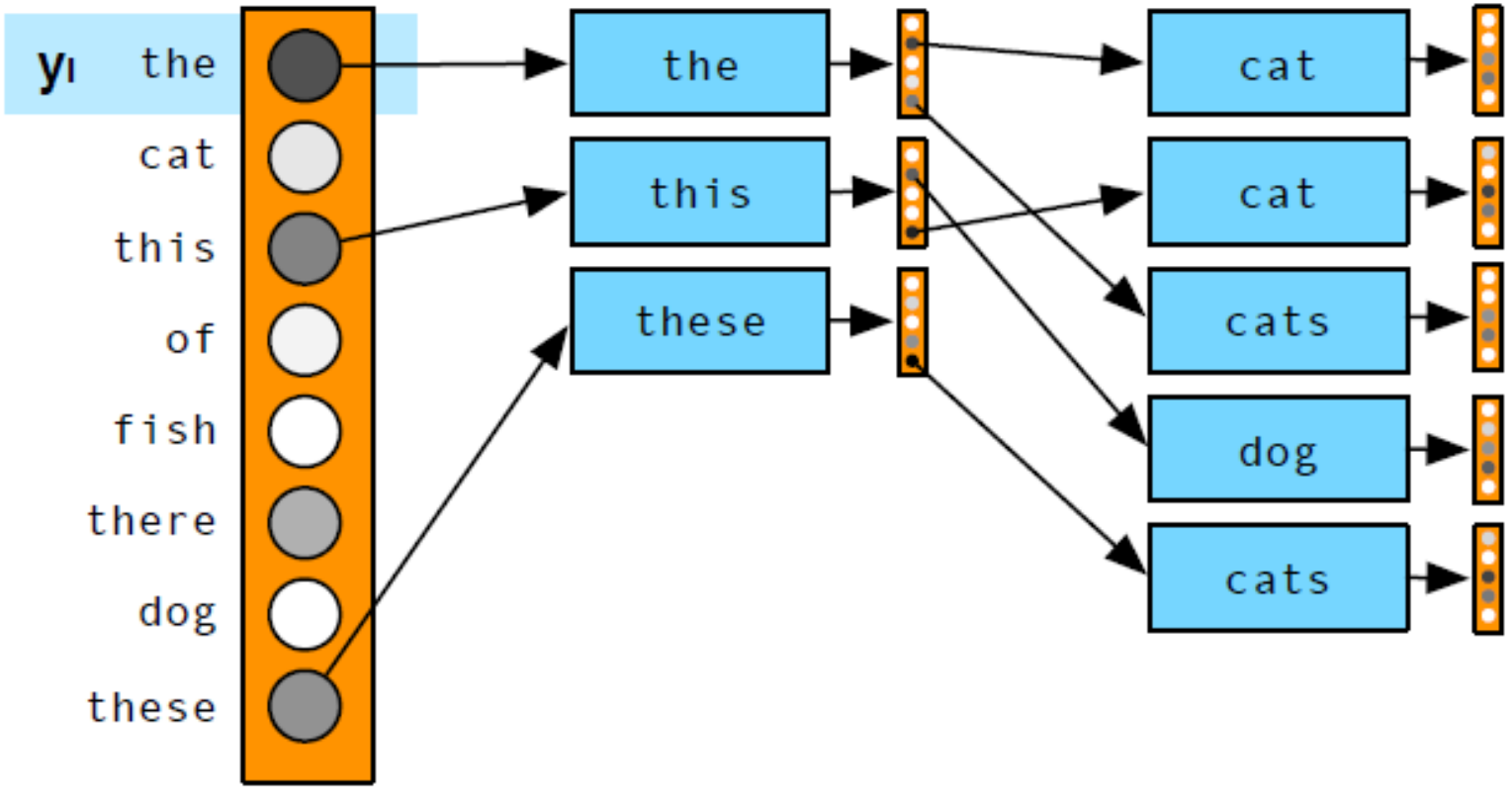


Figure 13.26: Beam search in neural machine translation. After committing to a short list of specific output words (the **beam**), new word predictions are made for each. These differ since the committed output word is part of the conditioning context to make predictions.

[Koehn 2017]

Beam Decoding

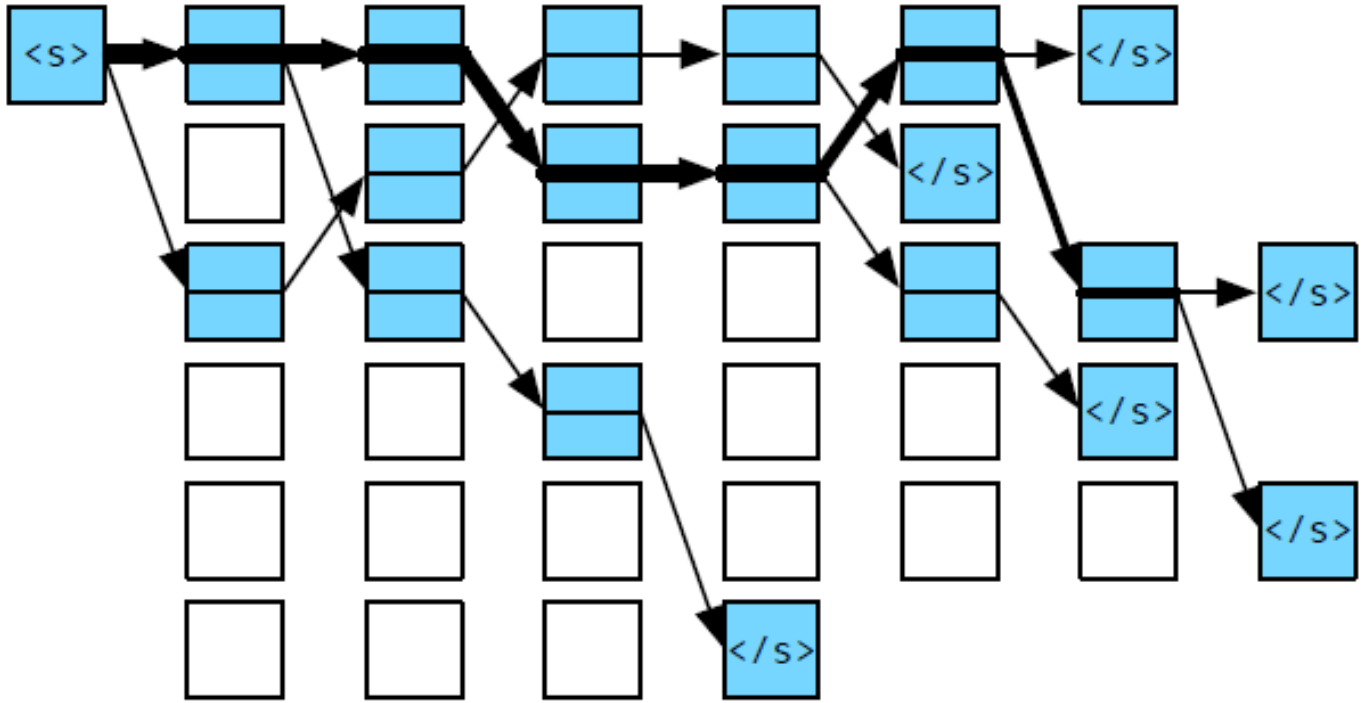


Figure 13.27: Search graph for beam search decoding in neural translation models. At each time step, the $n = 6$ best partial translations (called hypotheses) are selected. An output sentence is complete when the end of sentence token </s> is predicted. We reduce the beam after that and terminate when n full sentence translations are completed. Following the back-pointers from the end of sentence tokens allows us to read them off. Empty boxes represent hypotheses that are not part of any complete path.

[Koehn 2017]

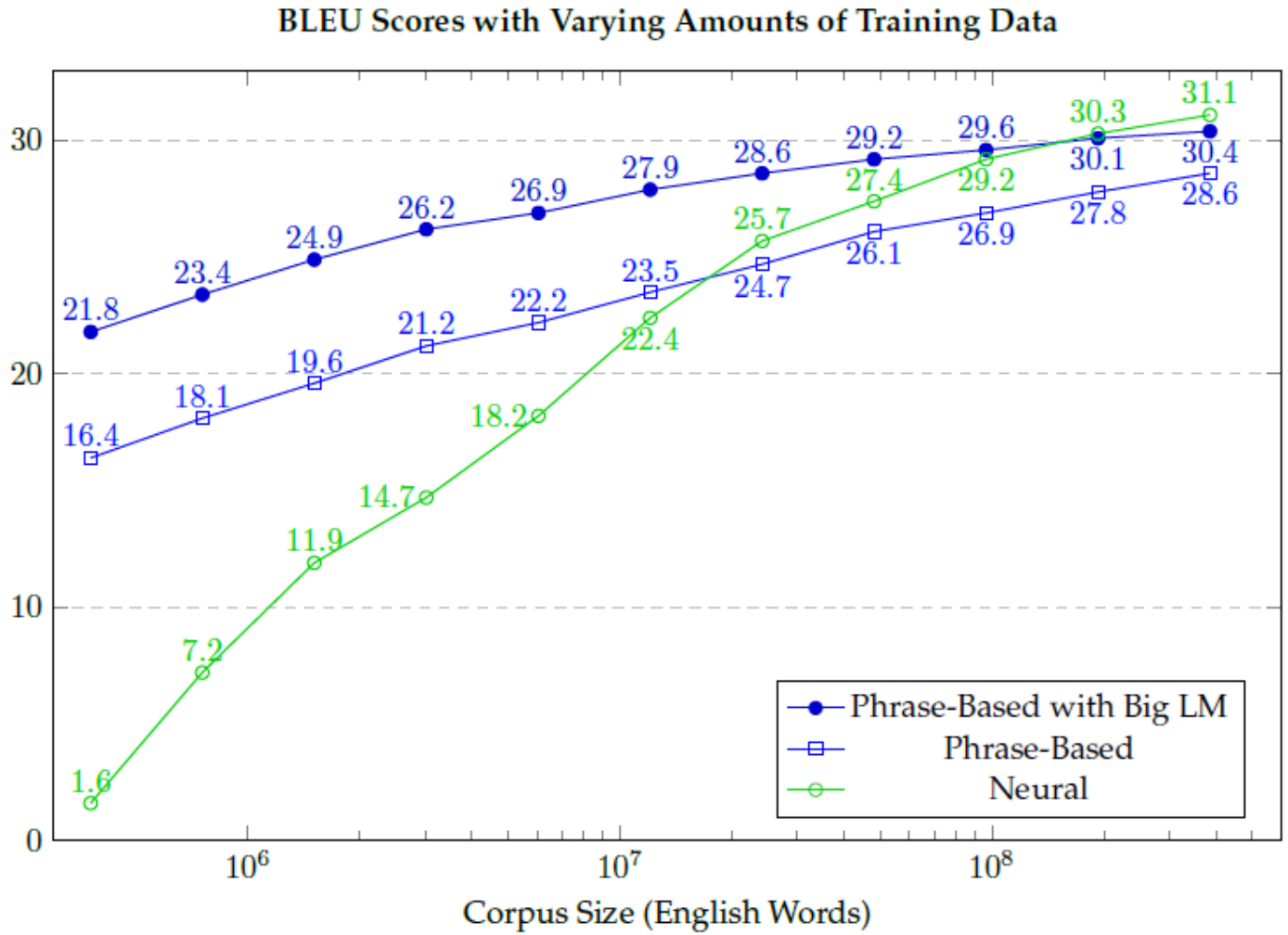


Figure 13.48: BLEU scores for English-Spanish systems trained on 0.4 million to 385.7 million words of parallel data. Quality for neural machine translation starts much lower, outperforms statistical machine translation at about 15 million words, and even beats a statistical machine translation system with a big 2 billion word in-domain language model under high-resource conditions.

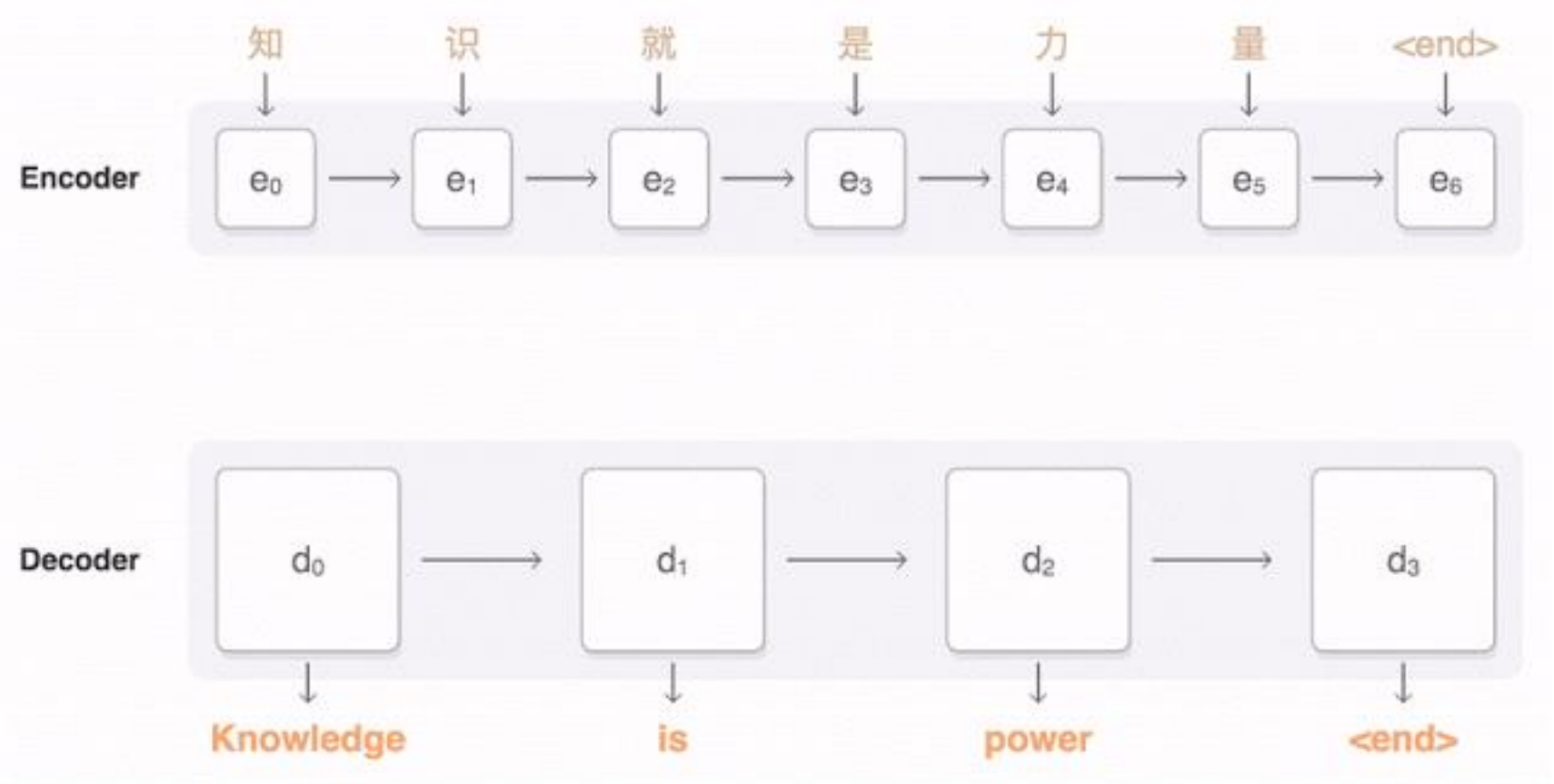
[Koehn 2017]

Ratio	Words	Source: *A Republican strategy to counter the re-election of Obama*
$\frac{1}{1024}$	0.4 million	*Un órgano de coordinación para el anuncio de libre determinación*
$\frac{1}{512}$	0.8 million	*Lista de una estrategia para luchar contra la elección de hojas de Ohio*
$\frac{1}{256}$	1.5 million	*Explosión realiza una estrategia divisiva de luchar contra las elecciones de autor*
$\frac{1}{128}$	3.0 million	*Una estrategia republicana para la eliminación de la reelección de Obama*
$\frac{1}{64}$	6.0 million	*Estrategia siria para contrarrestar la reelección del Obama .*
$\frac{1}{32}+$	12.0 million	*Una estrategia republicana para contrarrestar la reelección de Obama*

Figure 13.49: Translations of the first sentence of the test set using neural machine translation system trained on varying amounts of training data. Under low resource conditions, neural machine translation produces fluent output unrelated to the input.

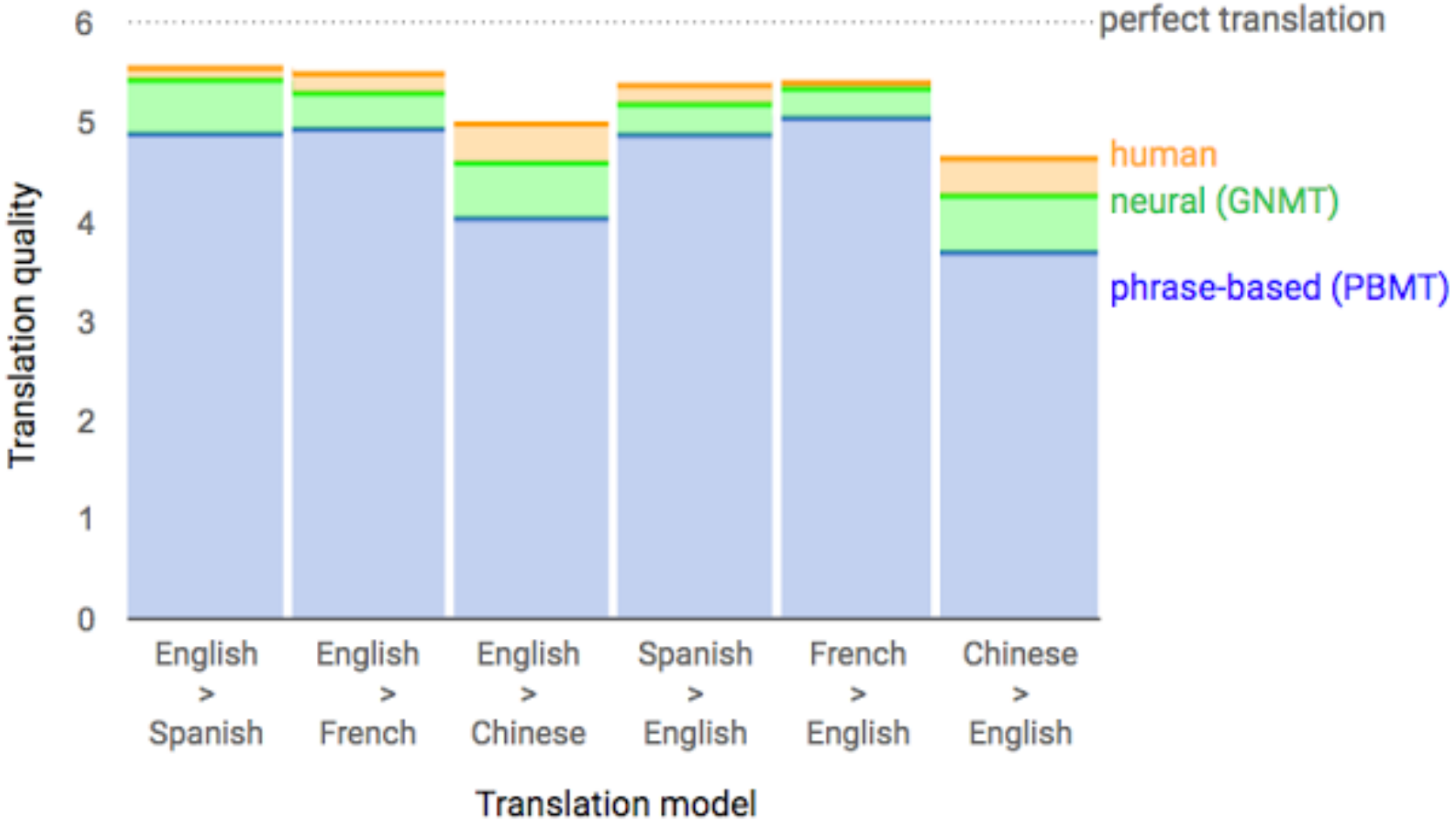
[Koehn 2017]

Google Neural Translation System



https://research.googleblog.com/2016/09/a-neural-network-for-machine.html

Google Neural Translation System



https://research.googleblog.com/2016/09/a-neural-network-for-machine.html

Google Neural Translation System

Input sentence:	*Translation (PBMT):*	*Translation (GNMT):*	*Translation (human):*
李克強此行將啟動中加總理年度對話機制，與加拿大總理杜魯多舉行兩國總理首次年度對話。	Li Keqiang premier added this line to start the annual dialogue mechanism with the Canadian Prime Minister Trudeau two prime ministers held its first annual session.	Li Keqiang will start the annual dialogue mechanism with Prime Minister Trudeau of Canada and hold the first annual dialogue between the two premiers.	Li Keqiang will initiate the annual dialogue mechanism between premiers of China and Canada during this visit, and hold the first annual dialogue with Premier Trudeau of Canada.

Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation, *Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Łukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, Jeffrey Dean. Technical Report, 2016.*

Remaining Issues in NMT

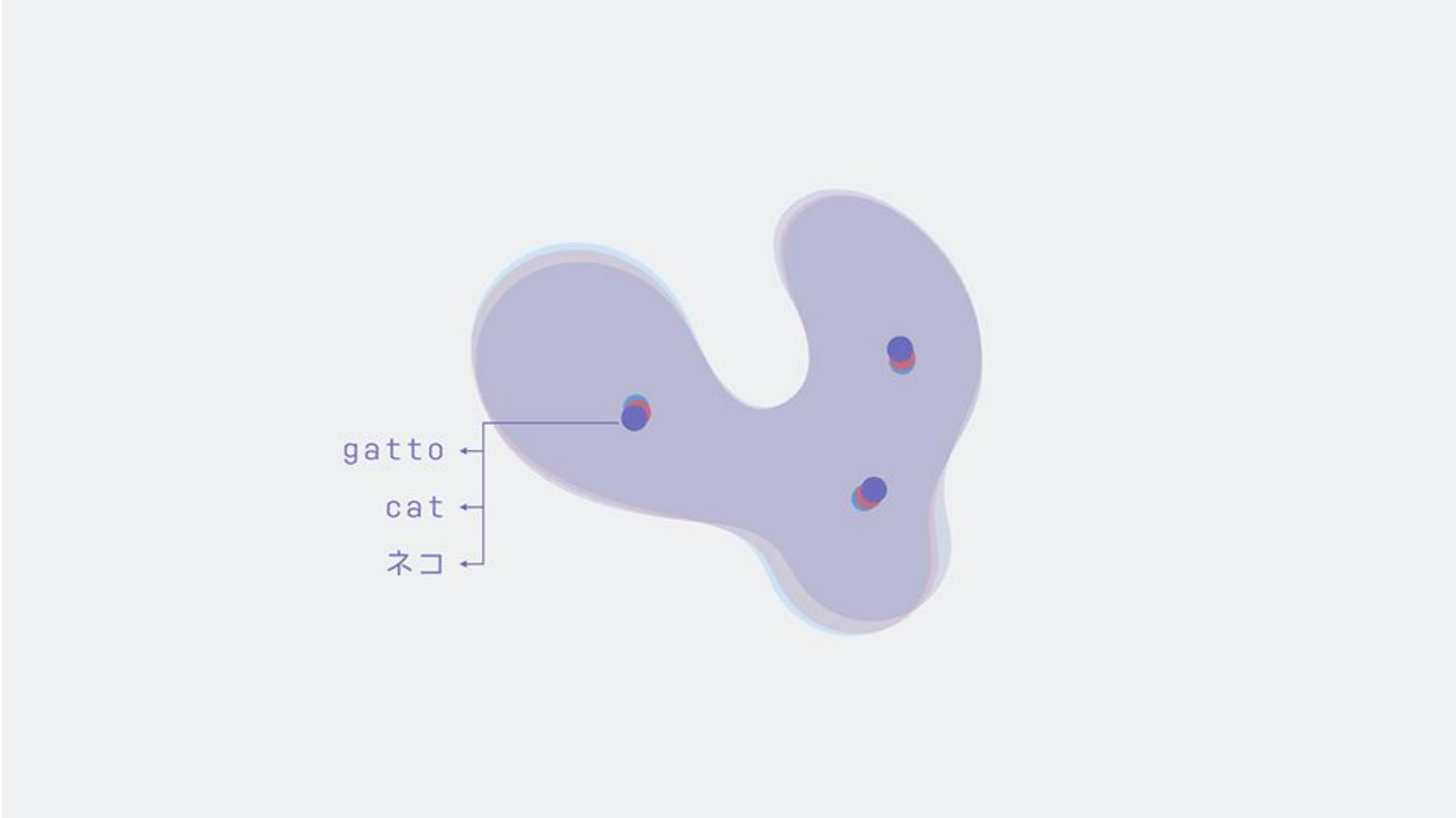
- Out of vocabulary terms
- Domain adaptation
- Long-distance context
- Low-resource languages
- Common sense
- Debiasing
- Interpretability

Seq2Seq Uses

- Not just for MT
- Also:
  - Syntactic parsing
  - Semantic parsing
  - Text generation
  - Dialogue systems
  - Summarization
  - Speech processing

Deep Learning

Unsupervised Neural Machine Translation



https://code.fb.com/ai-research/unsupervised-machine-translation-a-novel-approach-to-provide-fast-accurate-translations-for-more-languages/

Language pairs, even distant ones, share some fundamental linguistic structure

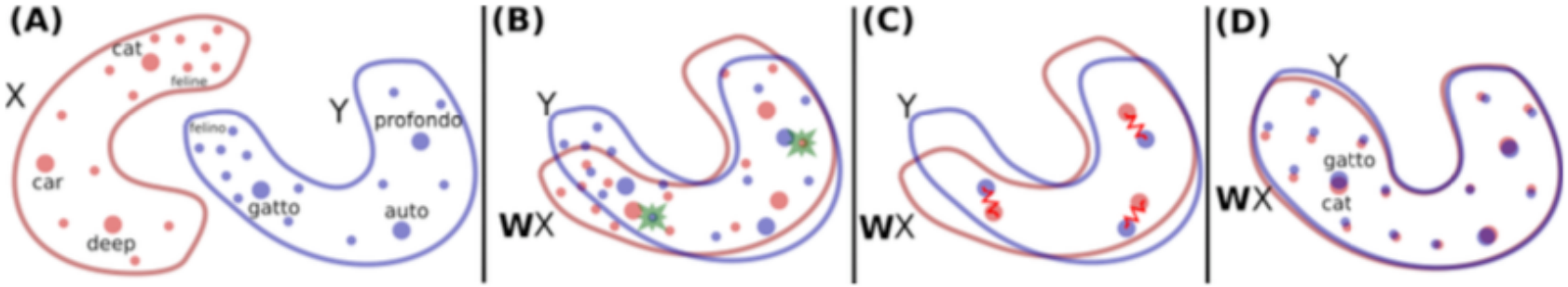


Figure 1: Toy illustration of the method. **(A)** There are two distributions of word embeddings, English words in red denoted by X and Italian words in blue denoted by Y, which we want to align/translate. Each dot represents a word in that space. The size of the dot is proportional to the frequency of the words in the training corpus of that language. **(B)** Using adversarial learning, we learn a rotation matrix W which roughly aligns the two distributions. The green stars are randomly selected words that are fed to the discriminator to determine whether the two word embeddings come from the same distribution. **(C)** The mapping W is further refined via Procrustes. This method uses frequent words aligned by the previous step as anchor points, and minimizes an energy function that corresponds to a spring system between anchor points. The refined mapping is then used to map all words in the dictionary. **(D)** Finally, we translate by using the mapping W and a distance metric, dubbed CSLS, that expands the space where there is high density of points (like the area around the word "cat"), so that "hubs" (like the word "cat") become less close to other word vectors than they would otherwise (compare to the same region in panel (A)).

Alexis Conneau, Guillaume Lample, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou.
Word Translation Without Parallel Data. arXiv: 1710.04087v3, 2017.