An Introduction to Machine Translation

Anoop Kunchukuttan

Microsoft AI and Research

Center for Indian Language Technology
Indian Institute of Technology Bombay

Ninth IIIT-H Advanced Summer School on NLP, 27th June 2018

Agenda

- What is Machine Translation & why is it interesting?
- Machine Translation Paradigms
- Word Alignment
- Phrase-based SMT
- Extensions to Phrase-based SMT
    - Addressing Word-order Divergence
    - Addressing Morphological Divergence
    - Handling Named Entities
- Syntax-based SMT
- Machine Translation Evaluation
- Summary

Statistical Machine Translation

Agenda

- <u>What is Machine Translation & why is it interesting?</u>
- Machine Translation Paradigms
- Word Alignment
- Phrase-based SMT
- Extensions to Phrase-based SMT
  - Addressing Word-order Divergence
  - Addressing Morphological Divergence
  - Handling Named Entities
- Syntax-based SMT
- Machine Translation Evaluation
- Summary

What is Machine Translation?

Automatic conversion of text/speech from one natural language to another

Be the change you want to see in the world

वह परिवर्तन बनो जो संसार में देखना चाहते हो

Machine Translation Usecases

Government

- Administrative requirements
- Education
- Security

Enterprise

- Product manuals
- Customer support

Social

- Travel (signboards, food)
- Entertainment (books, movies, videos)

Translation under the hood

- Cross-lingual Search
- Cross-lingual Summarization
- Building multilingual dictionaries

Any multilingual NLP system will involve some kind of machine translation at some level

Why should you study Machine Translation?

- One of the most challenging problems in Natural Language Processing
- Pushes the boundaries of NLP
- Involves analysis as well as synthesis
- Involves all layers of NLP: morphology, syntax, semantics, pragmatics, discourse
- Theory and techniques in MT are applicable to a wide range of other problems like transliteration, speech recognition and synthesis

Why is Machine Translation interesting?

Language Divergence ➔ *the great diversity among languages of the world*

The central problem of MT is to bridge this language divergence

Language Divergence

<u>Word order: SOV (Hindi), SVO (English), VSO, OSV</u>

E: $\overbrace{\text{Germany}}^{S}$ $\overbrace{\text{won}}^{V}$ the last $\overbrace{\text{World Cup}}^{O}$

H: $\underbrace{\text{जर्मनी}}_{S}$ ने पिछला $\underbrace{\text{विश्व कप}}_{O}$ $\underbrace{\text{जीता था}}_{V}$

<u>Free (Hindi) vs rigid (English) word order</u>

पिछला विश्व कप जर्मनी ने जीता था *(correct)*

The last World Cup Germany won *(grammatically incorrect)*

The last World Cup won Germany *(meaning changes)*

Language Divergence

Analytic vs Polysynthetic languages

Analytic (Chinese) → very few morphemes per word, no inflections

Polysynthetic (Finnish)→ many morphemes per word, no inflections

English: *Even if it does not rain*

Malayalam: മഴ പെയ്യുതിലെങ്ങിലും

(rain_noun shower_verb+not+even_if+then_also)

Inflectional systems [infixing (Arabic), fusional (Hindi), agglutinative (Marathi)]

Arabic

k-t-b: root word
katabtu: I wrote
kattabtu: I had (something) written
kitaab: book
kotub: books

Hindi

Jaaunga (1st per, singular, masculine)
Jaaoge (2nd per)
Jaayega (3rd per, singular, masculine)
Jaayenge (3rd per, plural)

Marathi

कपाटावरील: कपाट + वर + ईल
(the one over the cupboard)
दारावरील: दार + वर + ईल
(the one over the door)
दारामागील: दार + मागे + ईल
(the one behind the door)

Language Divergence

Different ways of expressing same concept

water → पानी, जल, नीर

Language registers

Formal: आप बैठिये

Informal: तू बैठ

Standard : मुझे डोसा चाहिए

Dakhini: मेरे को डोसा होना

Language Divergence

- Case marking systems
- Categorical divergence
- Null Subject Divergence
- Pleonastic Divergence

… and much more

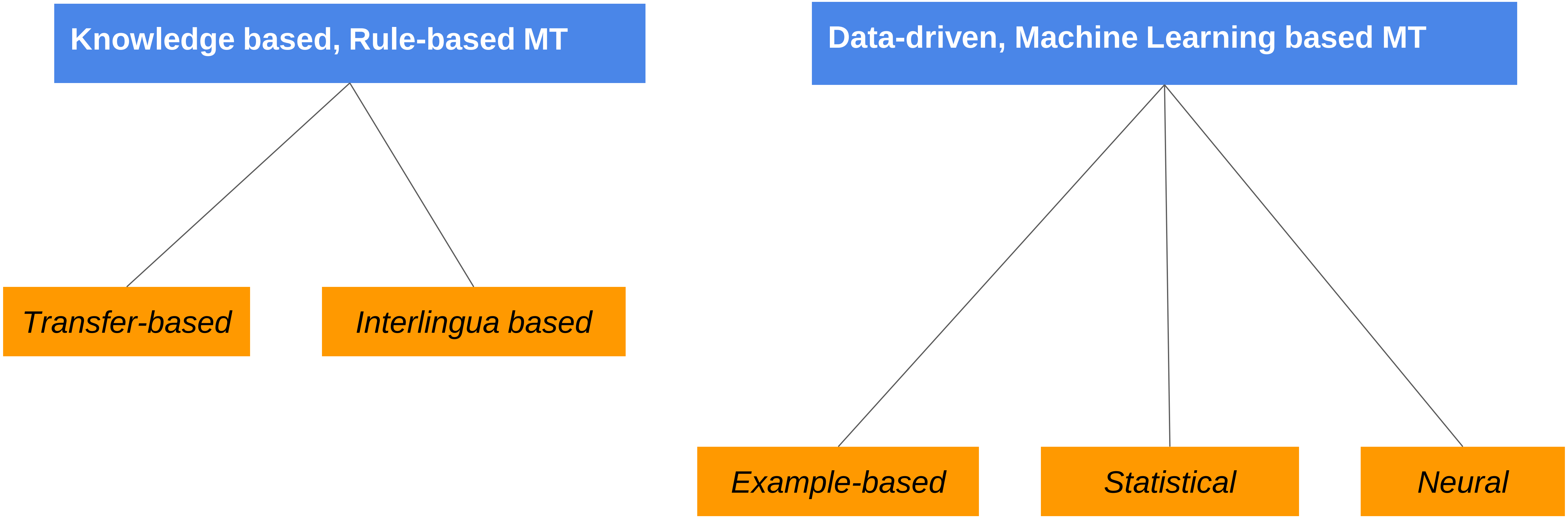
Why is Machine Translation difficult?

- **Ambiguity**
  - Same word, multiple meanings: मंत्री (minister or chess piece)
  - Same meaning, multiple words: जल, पानी, नीर (water)
- **Word Order**
  - Underlying deeper syntactic structure
  - Phrase structure grammar?
  - Computationally intensive
- **Morphological Richness**
  - Identifying basic units of words

Agenda

- What is Machine Translation & why is it interesting?
- <u>Machine Translation Paradigms</u>
- Word Alignment
- Phrase-based SMT
- Extensions to Phrase-based SMT
  - Addressing Word-order Divergence
  - Addressing Morphological Divergence
  - Handling Named Entities
- Syntax-based SMT
- Machine Translation Evaluation
- Summary

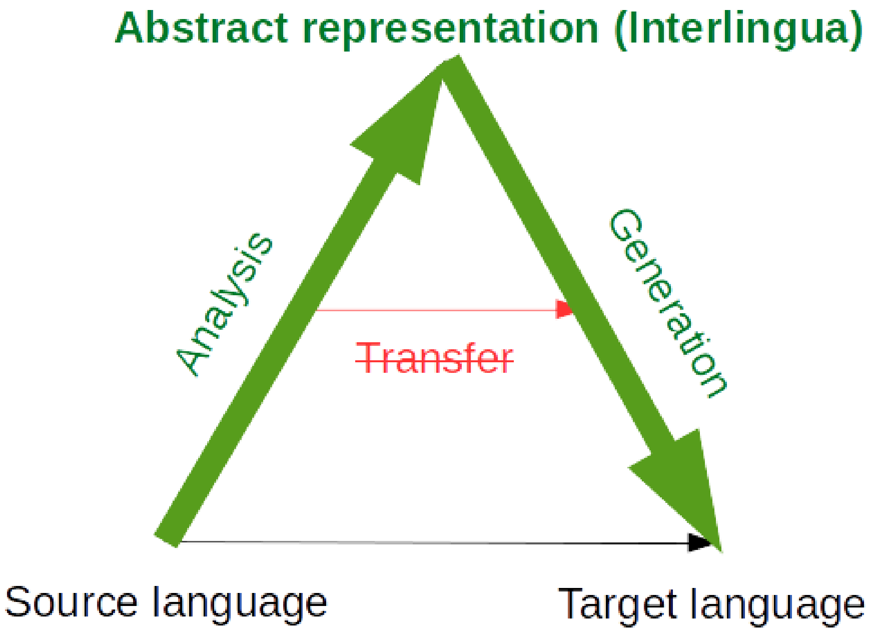
Approaches to build MT systems



Rule-based MT

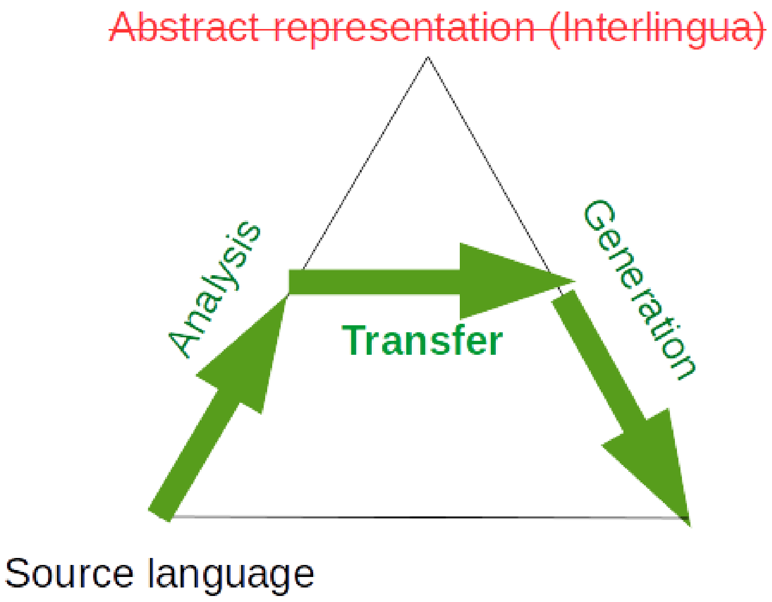
- Rules are written by ***linguistic experts*** to analyze the source, generate an intermediate representation, and generate the target sentence
- Depending on the depth of analysis: interlingua or transfer-based MT

Interlingua based MT



Deep analysis, complete disambiguation and language independent representation

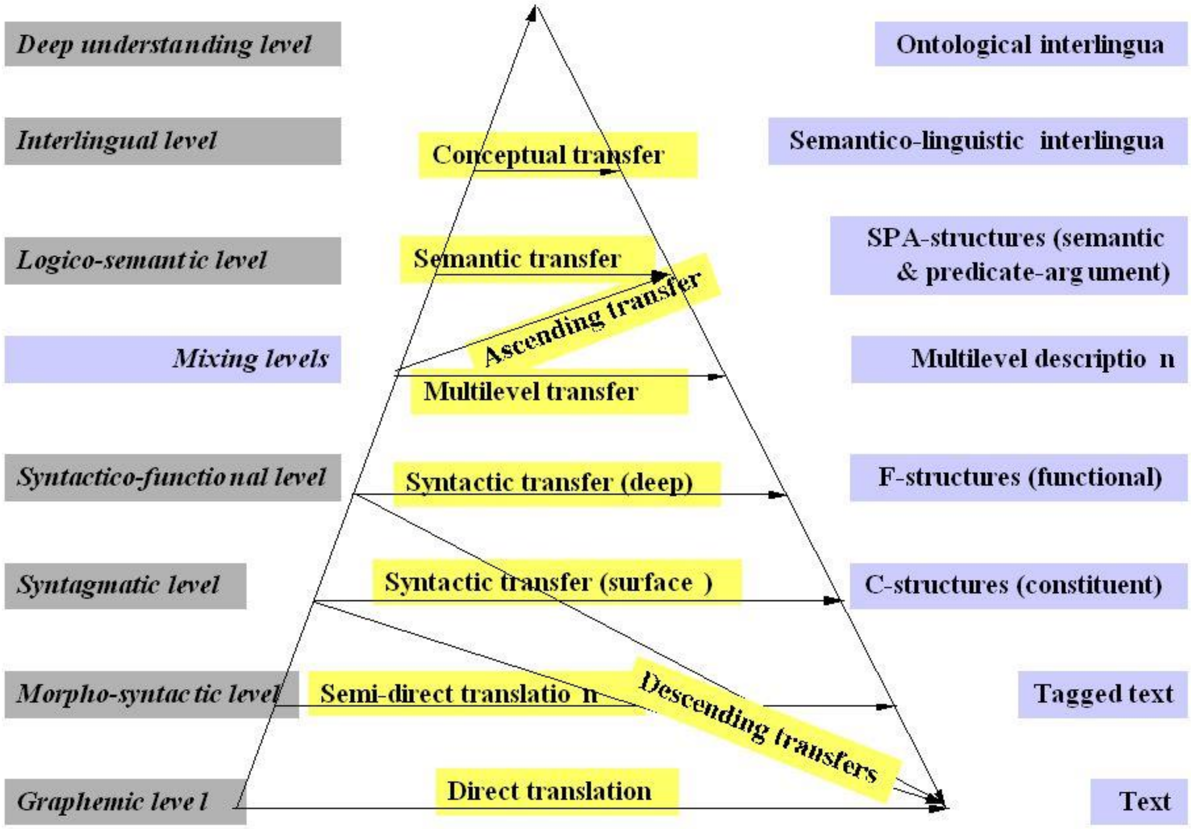
Transfer based MT



Partial analysis, partial disambiguation and a bridge intermediate representation

Vauquois Triangle

Translation approaches can be classified by the depth of linguistic analysis they perform

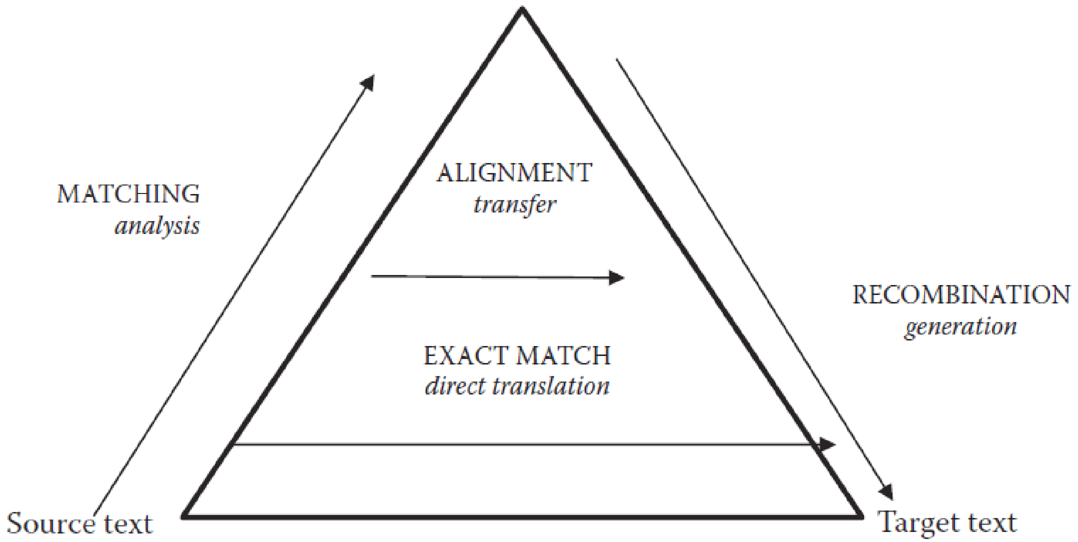


Problems with rule-based MT

- Required linguistic expertise to develop systems
- Maintenance of system is difficult
- Difficult to handle ambiguity
- Scaling to a large number of language pairs is not easy

Example-based MT

Translation by analogy ⇒ *match parts of sentences to known translations and then combine*



Input: *He buys a book on international politics*

1. **Phrase fragment matching: (*data-driven*)**

   he buys
   a book
   international politics

2. **Translation of segments: (*data-driven*)**

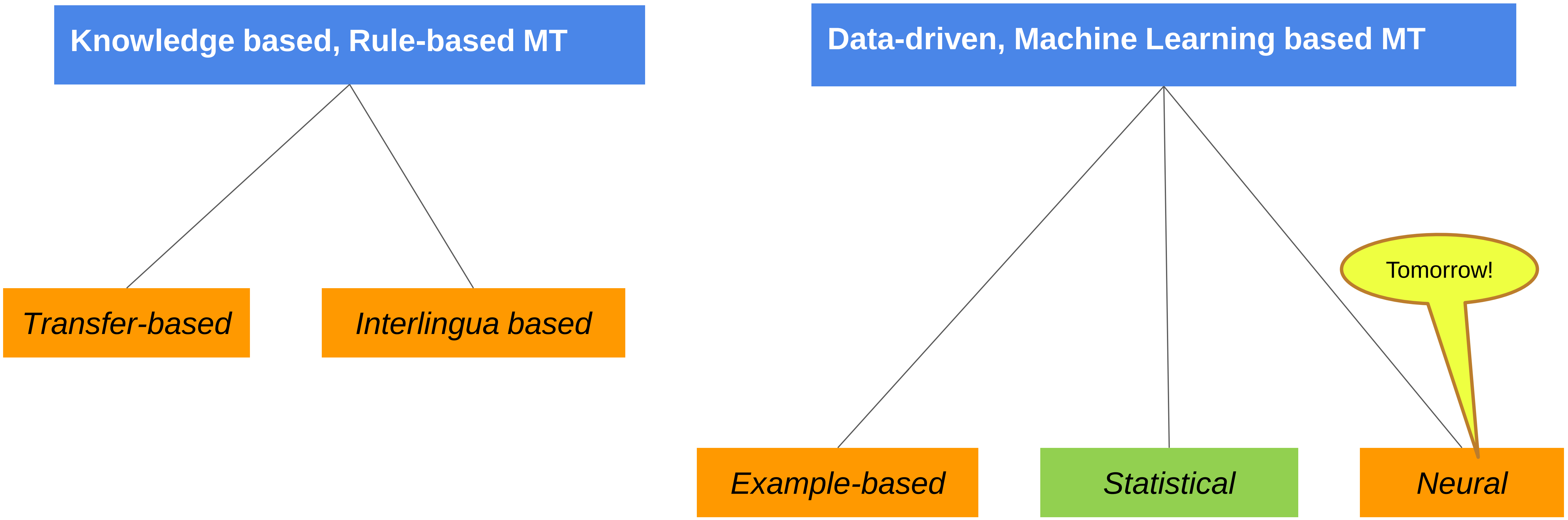
   वह खरीदता है
   एक किताब
   अंतर राष्ट्रीय राजनीति

3. **Recombination:** *(human crafted rules/templates)*

   वह अंतर राष्ट्रीय राजनीति पर एक किताब खरीदता है

- *Partly rule-based, partly data-driven.*
- *Good methods for matching and large corpora did not exist when proposed*

Approaches to build MT systems



Statistical Machine Translation

A Probabilistic Formalism

Let's formalize the translation process

We will model translation using a ***probabilistic model****. Why?*

- *We would like to have a measure of confidence for the translations we learn*
- *We would like to model uncertainty in translation*

E: target language
F: source language

e: source language sentence
f : target language sentence

Best translation

$$\bar{e} = \arg\max_{e} P(e|f)$$

How do we **model** this quantity?

Model*: a simplified and idealized understanding of a physical process*

We must first explain the process of translation

We explain translation using the Noisy Channel Model

A very general framework for many NLP problems

Generate target sentence

Channel corrupts the target

Source sentence is a corruption of the target sentence

Language Model (LM)
P(e)

Captures fluency

Translation Model (TM)
P(f|e)

Captures fidelity

$$P(e|f) = P(e) \times P(f|e)$$

Translation is the process of recovering the original signal given the corrupted signal

Why use this counter-intuitive way of explaining translation?

- Makes it easier to mathematically represent translation and learn probabilities
- **Fidelity** and **Fluency** can be modelled separately

We have already seen how to learn n-gram language models

Let's see how to learn the translation model → $P(\boldsymbol{f}|\boldsymbol{e})$

To learn sentence translation probabilities,

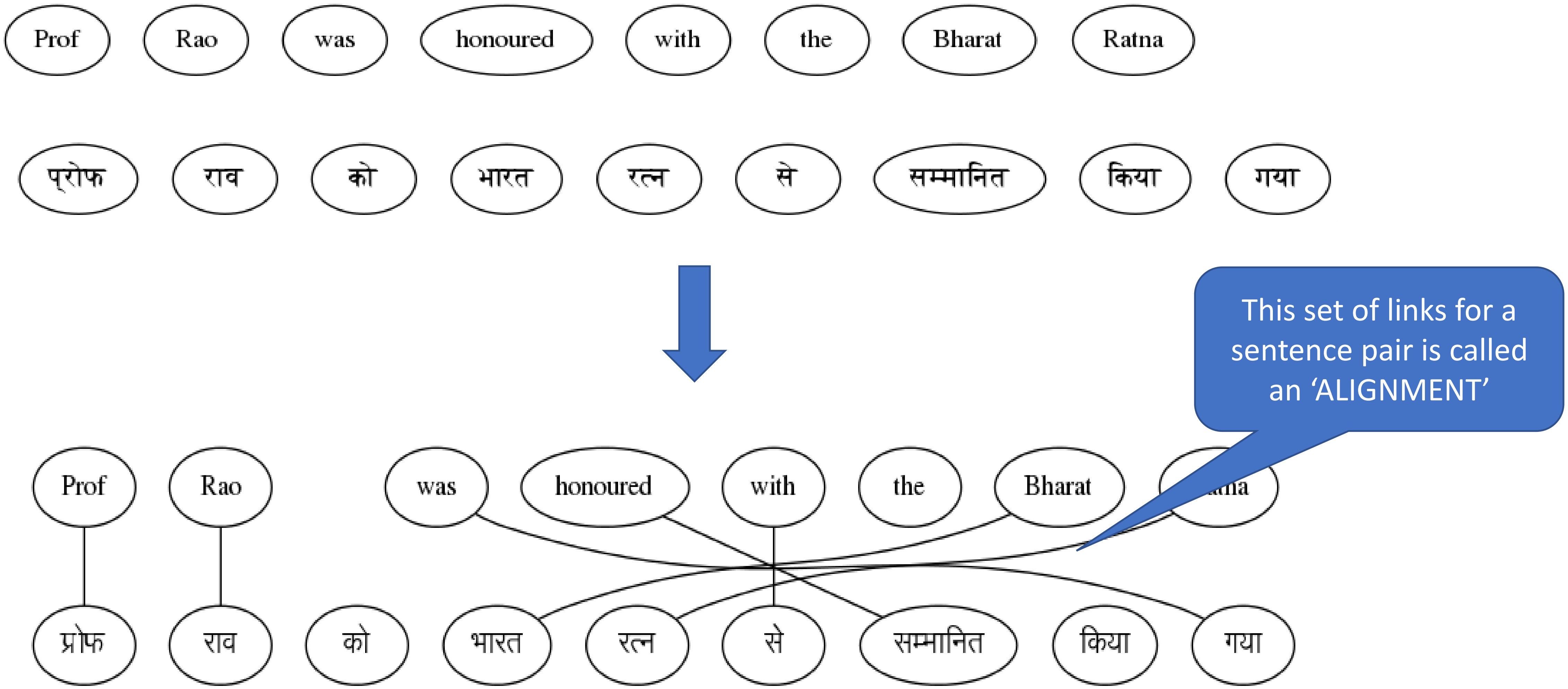
→ we first need to learn word-level translation probabilities

That is the task of word alignment

Agenda

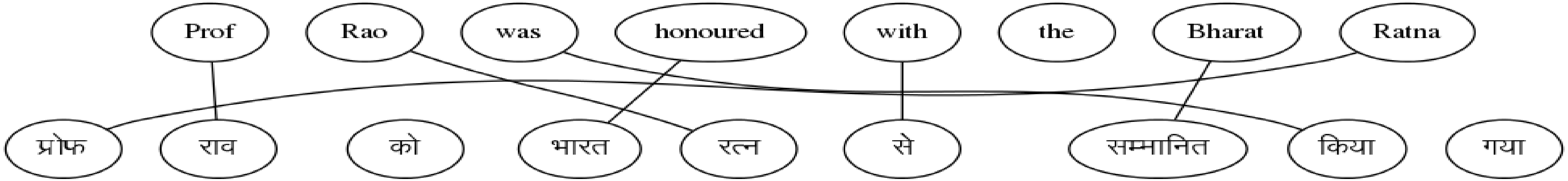
- What is Machine Translation & why is it interesting?
- Machine Translation Paradigms
- **Word Alignment**
- Phrase-based SMT
- Extensions to Phrase-based SMT
  - Addressing Word-order Divergence
  - Addressing Morphological Divergence
  - Handling Named Entities
- Syntax-based SMT
- Machine Translation Evaluation
- Summary

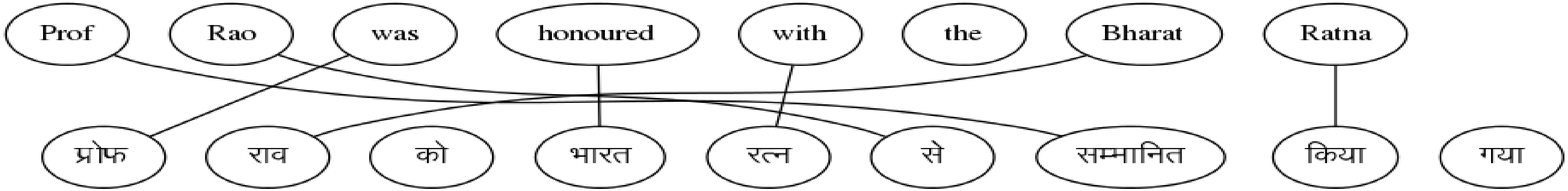
Given a parallel sentence pair, find word level correspondences



But there are multiple possible alignments

Sentence 1

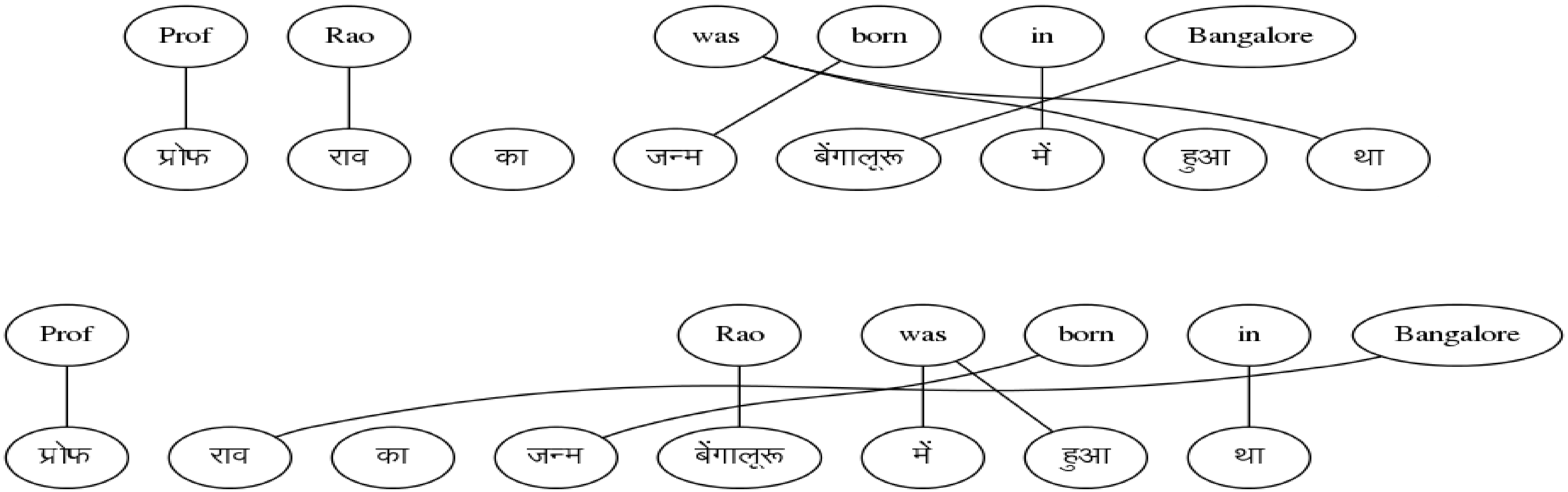




With one sentence pair, we cannot find the correct alignment

Can we find alignments if we have multiple sentence pairs?

Sentence 2



Yes, let's see how to do that …

Parallel Corpus	
A boy is sitting in the kitchen	एक लडका रसोई मे बैठा है
A boy is playing tennis	एक लडका टेनिस खेल रहा है
A boy is sitting on a round table	एक लडका एक गोल मेज पर बैठा है
Some men are watching tennis	कुछ आदमी टेनिस देख रहे है
A girl is holding a black book	एक लडकी ने एक काली किताब पकडी है
Two men are watching a movie	दो आदमी चलचित्र देख रहे है
A woman is reading a book	एक औरत एक किताब पढ रही है
A woman is sitting in a red car	एक औरत एक काले कार मे बैठी है

Parallel Corpus	
A boy is **sitting** in the kitchen	एक लडका रसोई मे **बैठा** है
A boy is playing **tennis**	एक लडका **टेनिस** खेल रहा है
A boy is **sitting** on a round table	एक लडका एक गोल मेज पर **बैठा** है
Some men **are watching tennis**	कुछ आदमी **टेनिस देख रहे है**
A girl is holding a black book	एक लडकी ने एक काली किताब पकडी है
Two men **are watching** a movie	दो आदमी चलचित्र **देख रहे है**
A woman is reading a book	एक औरत एक किताब पढ रही है
A woman is **sitting** in a red car	एक औरत एक काले कार मे **बैठा** है

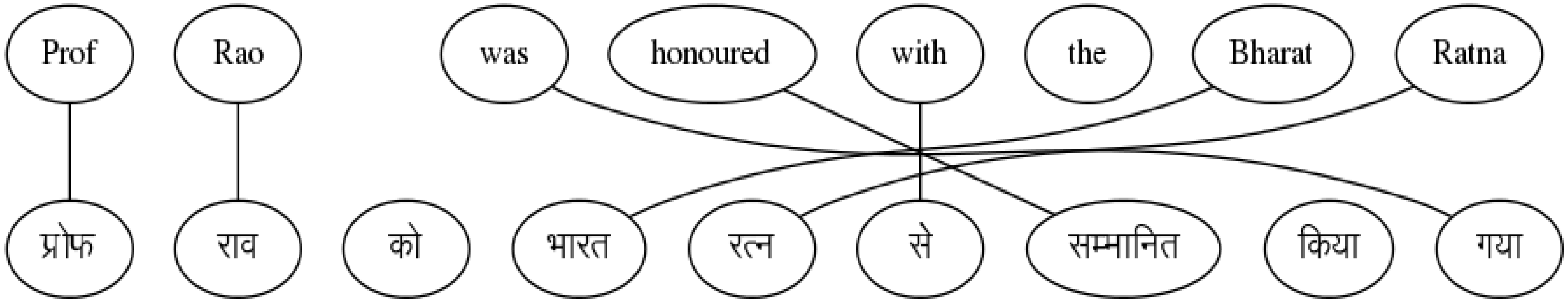
Key Idea

Co-occurrence of translated words

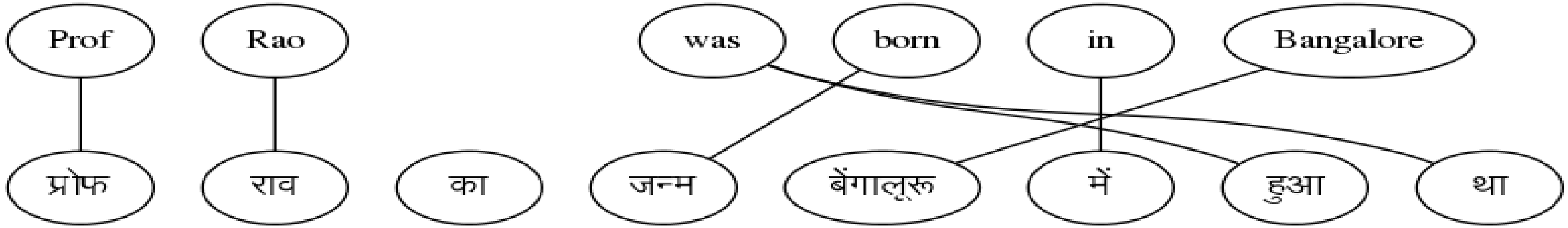
Words which occur together in the parallel sentence are likely to be translations (higher P(f|e))

If we knew the alignments, we could compute P(f|e)

Sentence 1



Sentence 2



$$P(f|e) = \frac{\#(f,e)}{\#(*,e)}$$

$$P(Prof|\text{प्रोफ}) = \frac{2}{2}$$

$\#(a,b)$: number of times word a is aligned to word b

But, we can find the best alignment only if we know the word translation probabilities

The best alignment is the one that maximizes the sentence translation probability

$P(\boldsymbol{f}, \boldsymbol{a} | \boldsymbol{e}) = P(a) \prod_{i=1}^{i=m} P\left(f_i \middle| e_{a_i}\right)$ $\Rightarrow$ $\boldsymbol{a}^* = \underset{\boldsymbol{a}}{\operatorname{argmax}} \prod_{i=1}^{i=m} P(f_i | e_{a_i})$

This is a chicken and egg problem! How do we solve this?

We can solve this problem using a two-step, iterative process

Start with random values for word translation probabilities

Step 1: Estimate alignment probabilities using word translation probabilities

Step 2: Re-estimate word translation probabilities

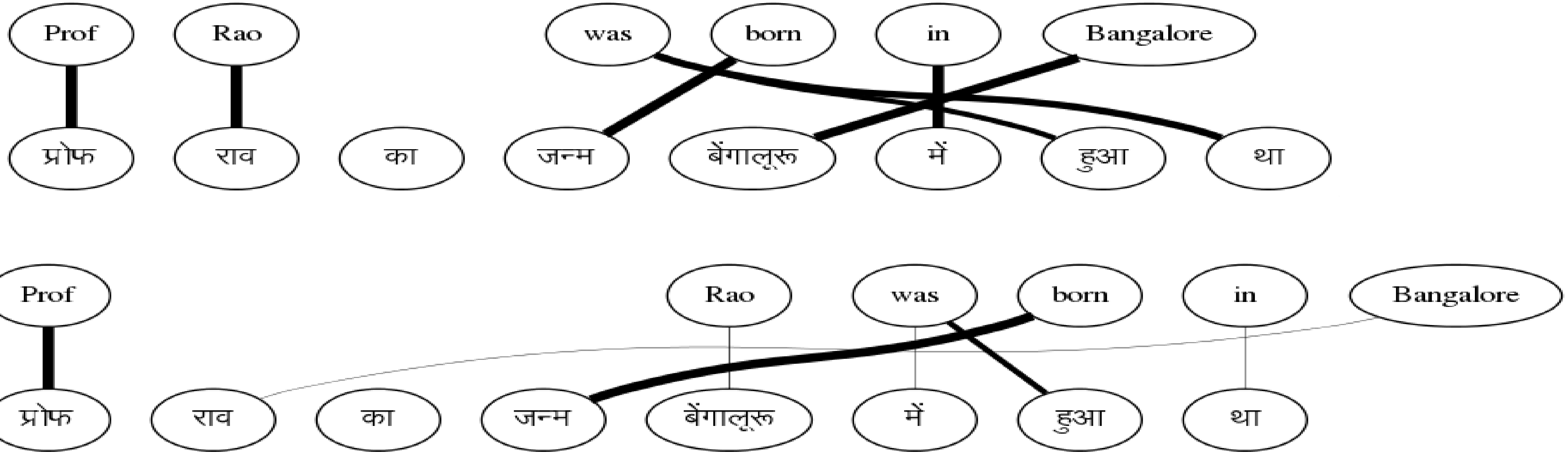
- *We don't know the best alignment*
- *So, we consider all alignments while estimating word translation probabilities*
- *Instead of taking only the best alignment, we consider all alignments and weigh the word alignments with the alignment probabilities*

$$P(f|e) = \frac{expected\ \#(f,e)}{expected\ \#(*,e)}$$

Repeat Steps (1) and (2) till the parameters converge

At the end of the process …

Sentence 2



Is the algorithm guaranteed to converge?

That’s the nice part → it is guaranteed to converge

This is an example of the well known Expectation-Maximization Algorithm

However, the problem is highly non-convex

Will lead to local minima

Good modelling assumptions necessary to ensure a good solution

IBM Models

- IBM came up with a series of increasingly complex models
- Called Models 1 to 5
- Differed in assumptions about alignment probability distributions
- Simper models are used to initialize the more complex models
- This pipelined training helped ensure better solutions

IBM Model 1

Assumption: All alignments are equally likely

E-step *computes expected counts*

$$c(f|e;\mathbf{f},\mathbf{e}) = \frac{t(f|e)}{t(f|e_0)+\cdots+t(f|e_l)} \underbrace{\sum_{j=1}^{m}\delta(f,f_j)}_{\text{count of } f \text{ in } \mathbf{f}} \overbrace{\sum_{i=0}^{l}\delta(e,e_i)}^{\text{count of } e \text{ in } \mathbf{e}}$$

M-step *uses expected counts to compute translation probabilities*

$$t(f|e) = \lambda_e^{-1}\sum_{s=1}^{S} c(f|e;\mathbf{f}^{(s)},\mathbf{e}^{(s)}).$$

$$\lambda_e = \sum_{s=1}^{S}\sum_{f \text{ in Vocab}(F)} c(f|e;\mathbf{f}^{(s)},\mathbf{e}^{(s)})$$ *(normalization factor)*

Summary

- EM provides a semi-supervised method for learning word alignments and word translation probabilities
- Word translation probabilities can be used to extract a bilingual dictionary
- Avoids the new for word-aligned corpora
- If a few word-aligned sentences are available, discriminative alignment methods can improve upon the EM-based solution
  - Arbitrary features can be incorporated
  - Morphological information
  - Character level edit distance

Agenda

- What is Machine Translation & why is it interesting?
- Machine Translation Paradigms
- Word Alignment
- **<u>Phrase-based SMT</u>**
- Extensions to Phrase-based SMT
  - Addressing Word-order Divergence
  - Addressing Morphological Divergence
  - Handling Named Entities
- Syntax-based SMT
- Machine Translation Evaluation
- Summary

What is PB-SMT?

Why stop at learning word correspondences?

KEY IDEA → Use “Phrase” (Sequence of Words) as the basic translation unit

Note: the term ‘phrase’ is not used in a linguistic sense

The Prime Minister of India	भारत के प्रधान मंत्री bhArata ke pradhAna maMtrI India of Prime Minister
is running fast	तेज भाग रहा है teja bhAg rahA hai fast run -continuous is
honoured with	से सम्मानित किया se sammanita kiyA with honoured did
Rahul lost the match	राहुल मुकाबला हार गया rAhula mukAbalA hAra gayA Rahul match lost

Benefits of PB-SMT

Local Reordering → Intra-phrase re-ordering can be memorized

The Prime Minister of India	भारत के प्रधान मंत्री bhaarat ke pradhaan maMtrI India of Prime Minister

Sense disambiguation based on local context → Neighbouring words help make the choice

heads towards Pune	पुणे की ओर जा रहे है pune ki or jaa rahe hai Pune towards go –continuous is
heads the committee	समिति की अध्यक्षता करते है Samiti kii adhyakshata karte hai committee of leading -verbalizer is

Benefits of PB-SMT (2)

Handling institutionalized expressions

- Institutionalized expressions, idioms can be learnt as a single unit

hung assembly	त्रिशंकु विधानसभा trishanku vidhaansabha
Home Minister	गृह मंत्री gruh mantrii
Exit poll	चुनाव बाद सर्वेक्षण chunav baad sarvekshana

- Improved Fluency
  - The phrases can be arbitrarily long (even entire sentences)

Mathematical Model

Let's revisit the decision rule for SMT model

$$\mathbf{e}_{\text{best}} = \text{argmax}_{\mathbf{e}}\, p(\mathbf{e}|\mathbf{f}) = \text{argmax}_{\mathbf{e}}\, p(\mathbf{f}|\mathbf{e})\, p_{\text{LM}}(\mathbf{e})$$

Let's revisit the translation model *p*(**f**|**e**)

- Source sentence can be segmented in **I** phrases
- Then, *p*(**f**|**e**) can be decomposed as:

$$p(\bar{f}_1^I|\bar{e}_1^I) = \prod_{i=1}^{I} \underbrace{\phi(\bar{f}_i|\bar{e}_i)}_{\text{Phrase Translation Probability}}\, \underbrace{d(\text{start}_i - \text{end}_{i-1} - 1)}_{\text{Distortion probability}}$$

start_i :start position in **f** of i^{th} phrase of **e**
end_i :end position in **f** of i^{th} phrase of **e**

Learning The Phrase Translation Model

Involves Structure + Parameter Learning:

- Learn the **Phrase Table**: the central data structure in PB-SMT

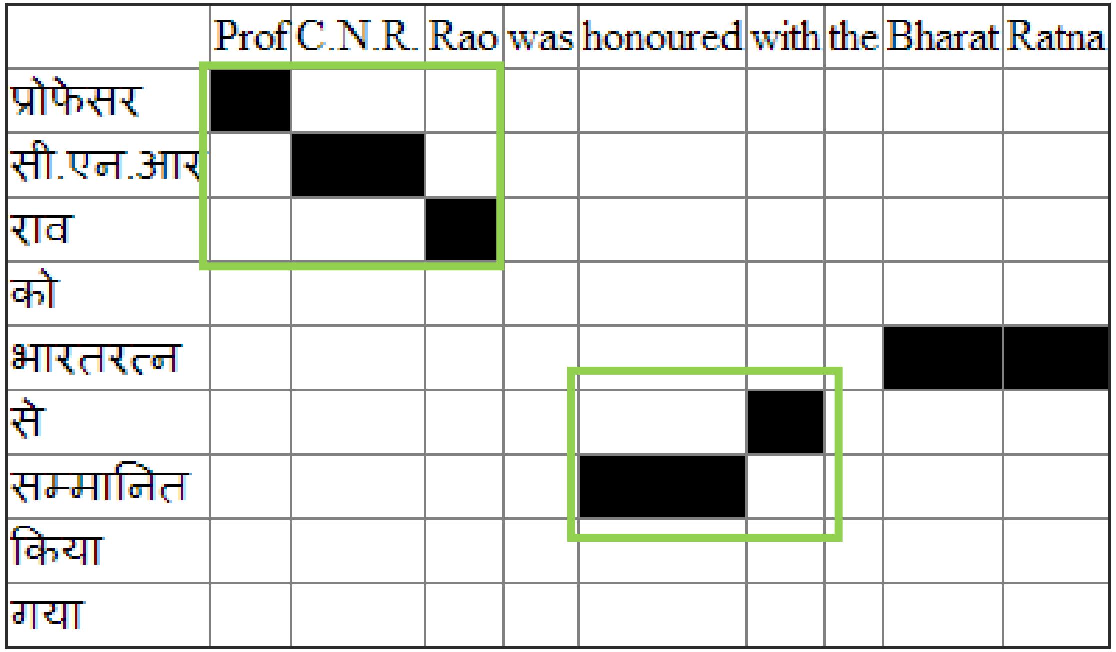
The Prime Minister of India	भारत के प्रधान मंत्री
is running fast	तेज भाग रहा है
the boy with the telescope	दूरबीन से लड़के को
Rahul lost the match	राहुल मुकाबला हार गया

- Learn the **Phrase Translation Probabilities**

Prime Minister of India	भारत के प्रधान मंत्री India of Prime Minister	0.75
Prime Minister of India	भारत के भूतपूर्व प्रधान मंत्री India of former Prime Minister	0.02
Prime Minister of India	प्रधान मंत्री Prime Minister	0.23

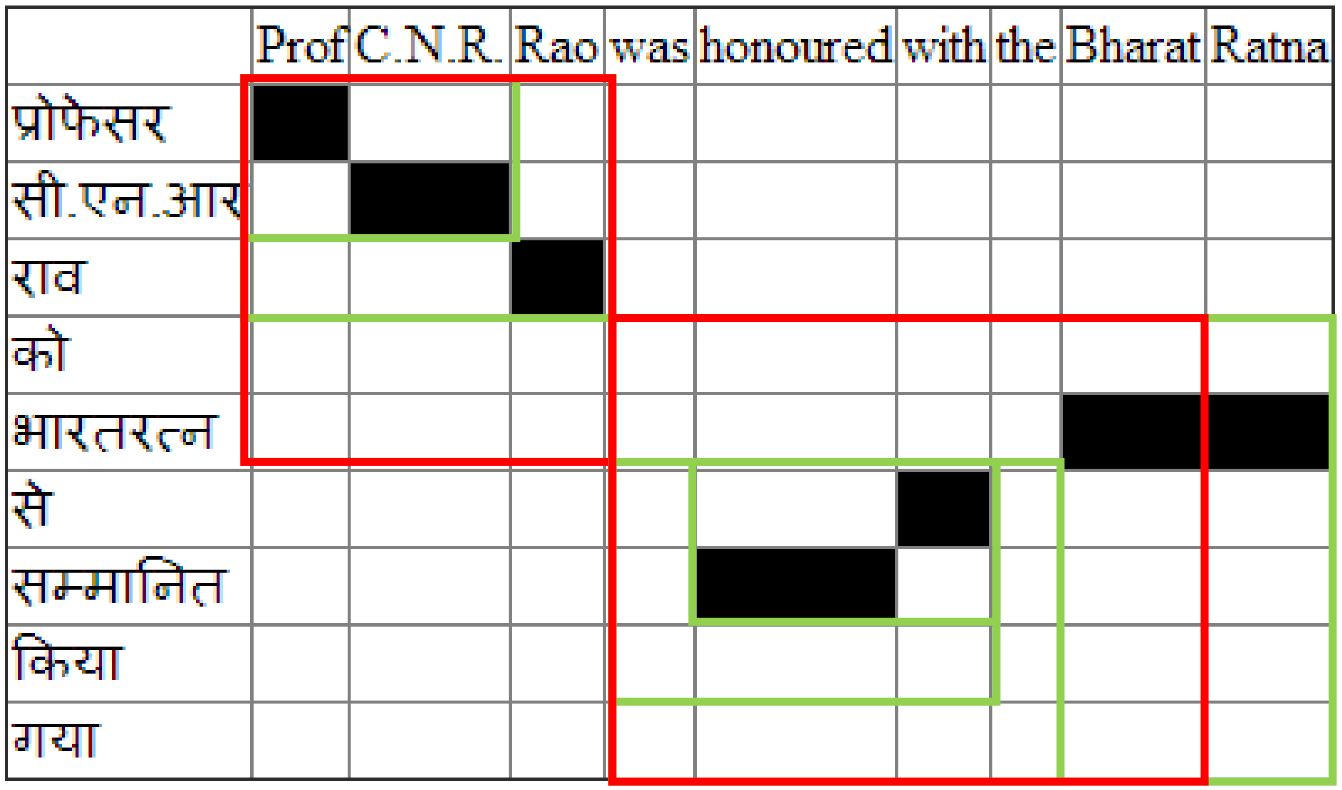
Learning Phrase Tables from Word Alignments

- Start with word alignments
- Word Alignment : reliable input for phrase table learning
    - high accuracy reported for many language pairs
- Central Idea: A consecutive sequence of aligned words constitutes a “phrase pair”

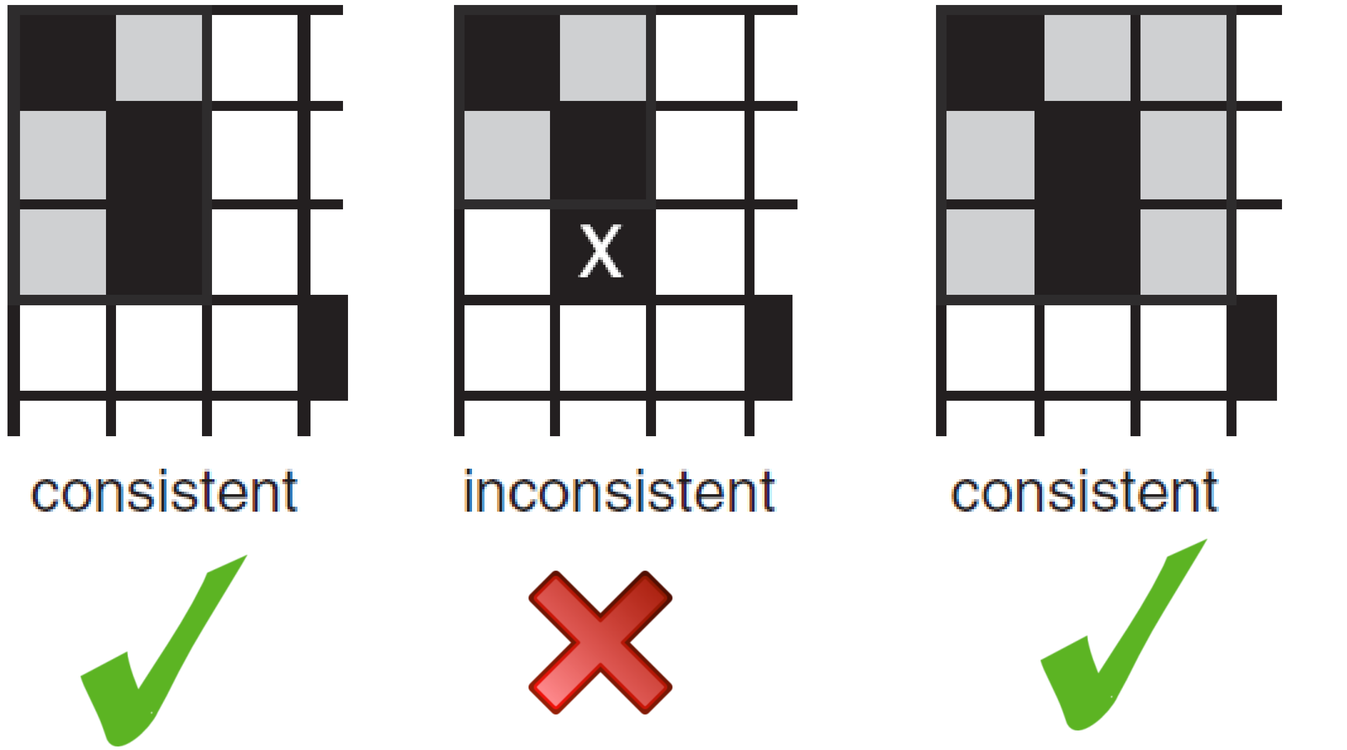


Which phrase pairs to include in the phrase table?

Extracting Phrase Pairs

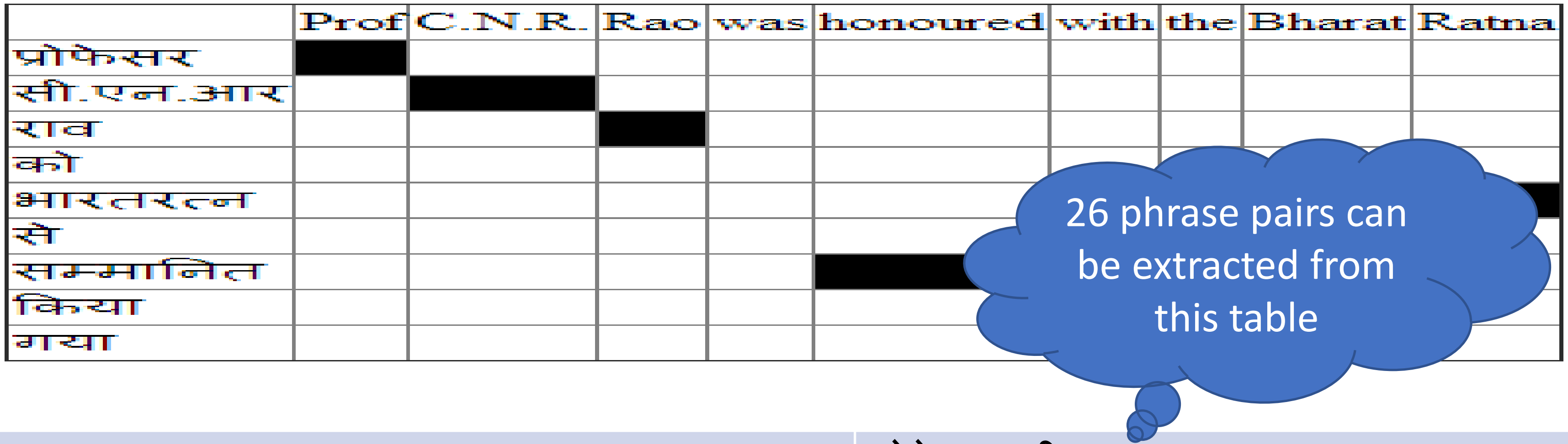


Phrase Pairs “consistent” with word alignment



Source: SMT, Phillip Koehn

Examples



Professor CNR	प्रोफेसर सी.एन.आर
Professor CNR Rao	प्रोफेसर सी.एन.आर राव
Professor CNR Rao was	प्रोफेसर सी.एन.आर राव
Professor CNR Rao was	प्रोफेसर सी.एन.आर राव को
honoured with the Bharat Ratna	भारतरत्न से सम्मानित
honoured with the Bharat Ratna	भारतरत्न से सम्मानित किया
honoured with the Bharat Ratna	भारतरत्न से सम्मानित किया गया
honoured with the Bharat Ratna	को भारतरत्न से सम्मानित किया गया

Computing Phrase Translation Probabilities

- Estimated from the relative frequency:

$$\boxed{\phi(\bar{f}|\bar{e}) = \frac{\text{count}(\bar{e},\bar{f})}{\sum_{\bar{f_i}} \text{count}(\bar{e},\bar{f_i})}}$$

Prime Minister of India	भारत के प्रधान मंत्री India of Prime Minister	***0.75***
Prime Minister of India	भारत के भूतपूर्व प्रधान मंत्री India of former Prime Minister	***0.02***
Prime Minister of India	प्रधान मंत्री Prime Minister	***0.23***

Discriminative Training of PB-SMT

- Directly model the posterior probability p**(e|f)**
- Use the Maximum Entropy framework

$$P(\mathbf{e}|\mathbf{f}) = \exp\left(\sum_i \lambda_i h_i(f_1^I, e_1^J)\right)$$

$$e^* = \arg\max_{e_i} \sum_i \lambda_i h_i(f_1^I, e_1^J)$$

  - h_i**(f,e)** are feature functions , λ_i's are feature weights
- Benefits:
  - Can add arbitrary features to score the translations
  - Can assign different weight for each features
  - Assumptions of generative model may be incorrect

More features for PB-SMT

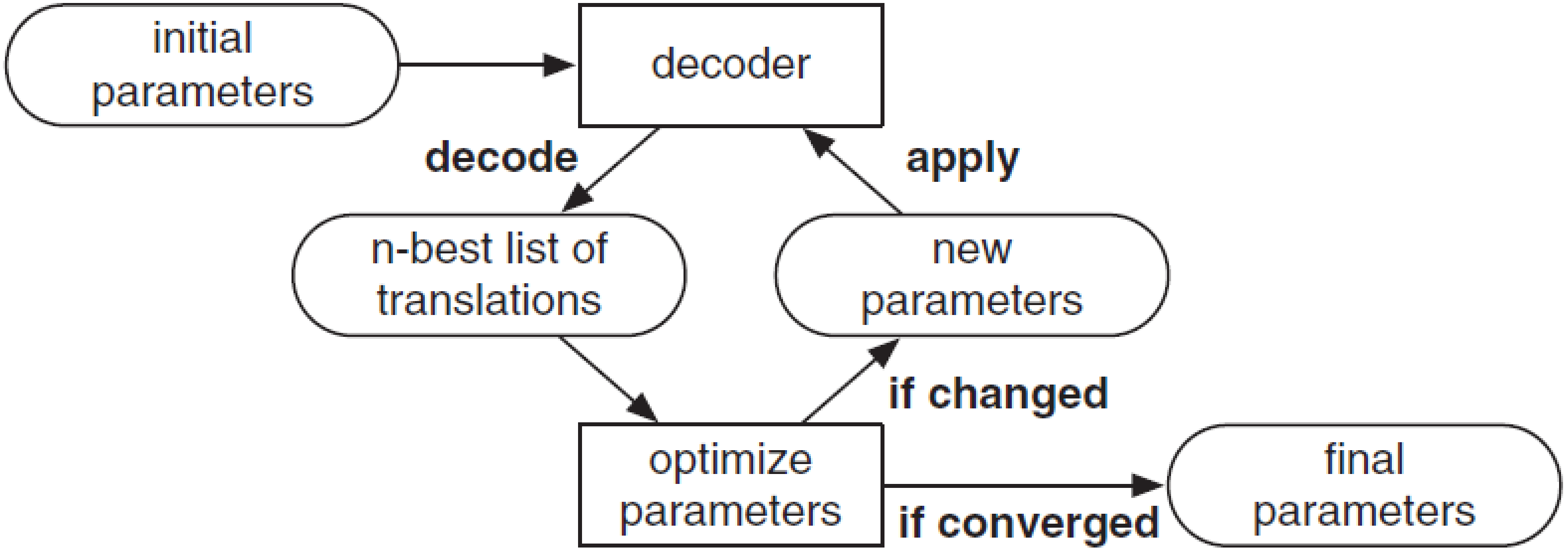
- Inverse phrase translation probability ($\phi(\bar{f}|\bar{e})$)
- Lexical Weighting

$$\boxed{\text{lex}(\bar{e}|\bar{f}, a) = \prod_{i=1}^{\text{length}(\bar{e})} \frac{1}{|\{j|(i,j) \in a\}|} \sum_{\forall (i,j) \in a} w(e_i|f_j)}$$

  - *a:* alignment between words in phrase pair (ē, f)
  - *w(x|y):* word translation probability
- Inverse Lexical Weighting
  - Same as above, in the other direction

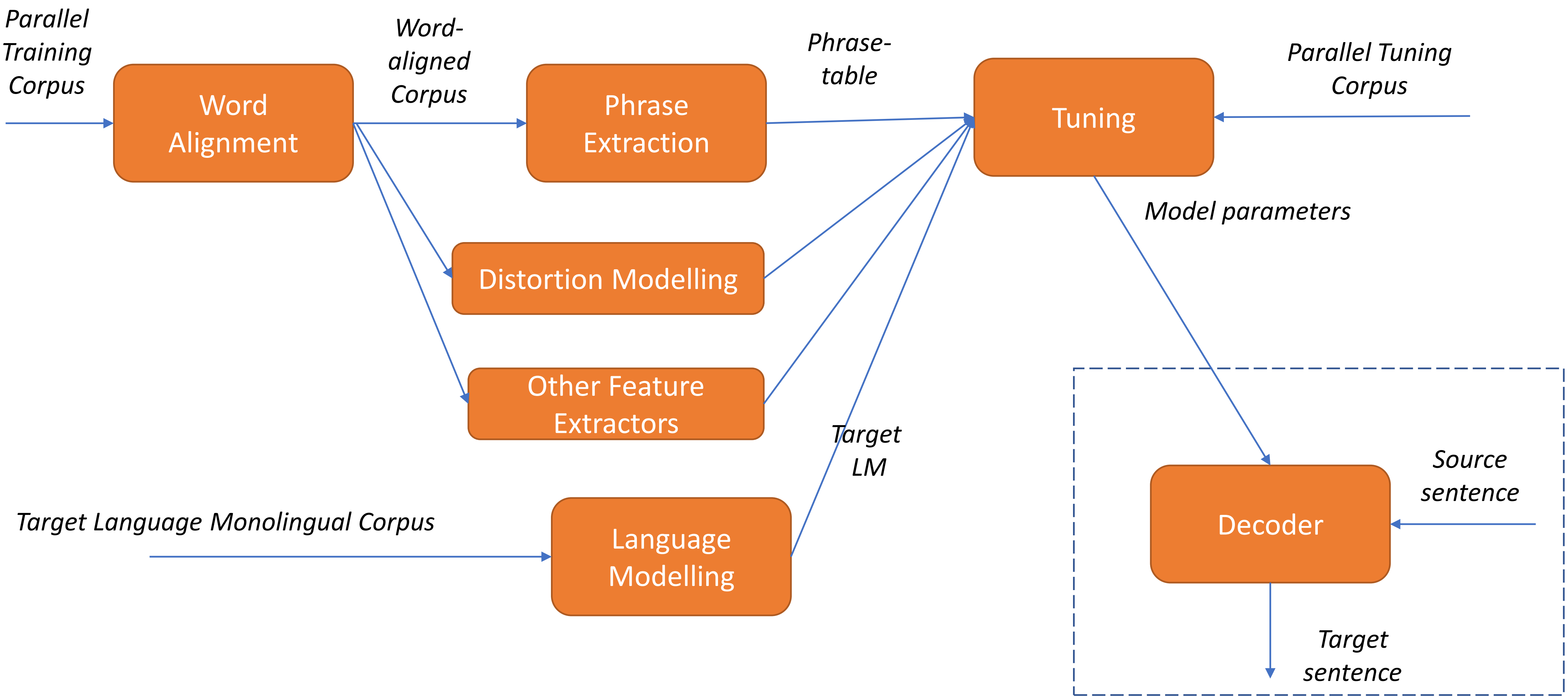
Tuning

- Learning feature weights from data – λ_i
- Minimum Error Rate Training (MERT)
- Search for weights which minimize the translation error on a held-out set (tuning set)
  - Translation error metric : (1 – *BLEU)*



Source: SMT, Phillip Koehn

Typical SMT Pipeline

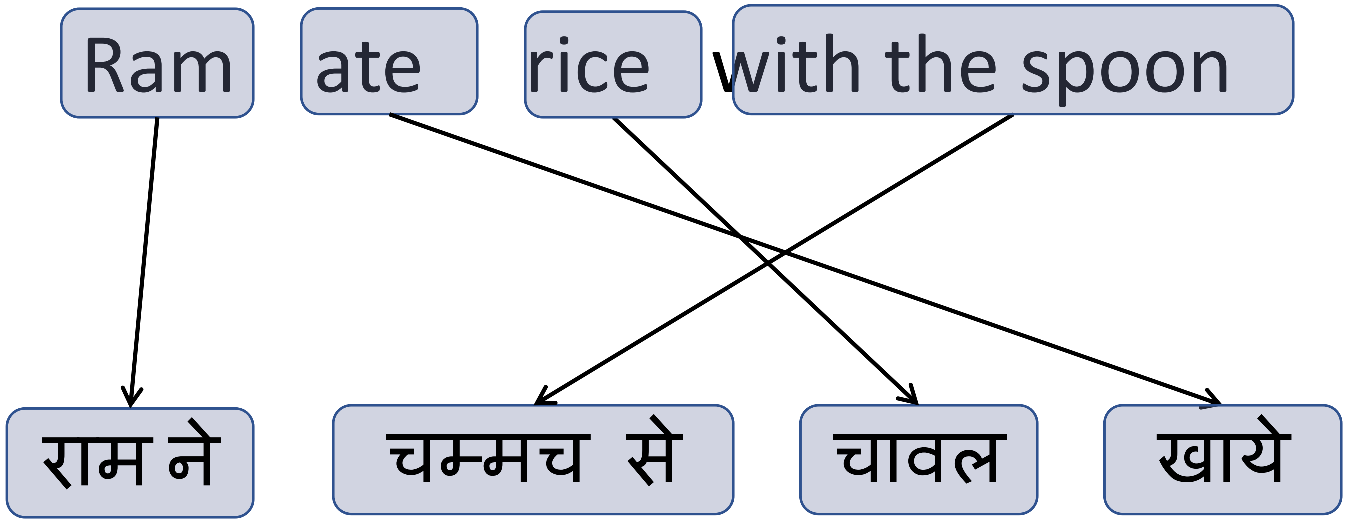


Decoding

Searching for the best translations in the space of all translations

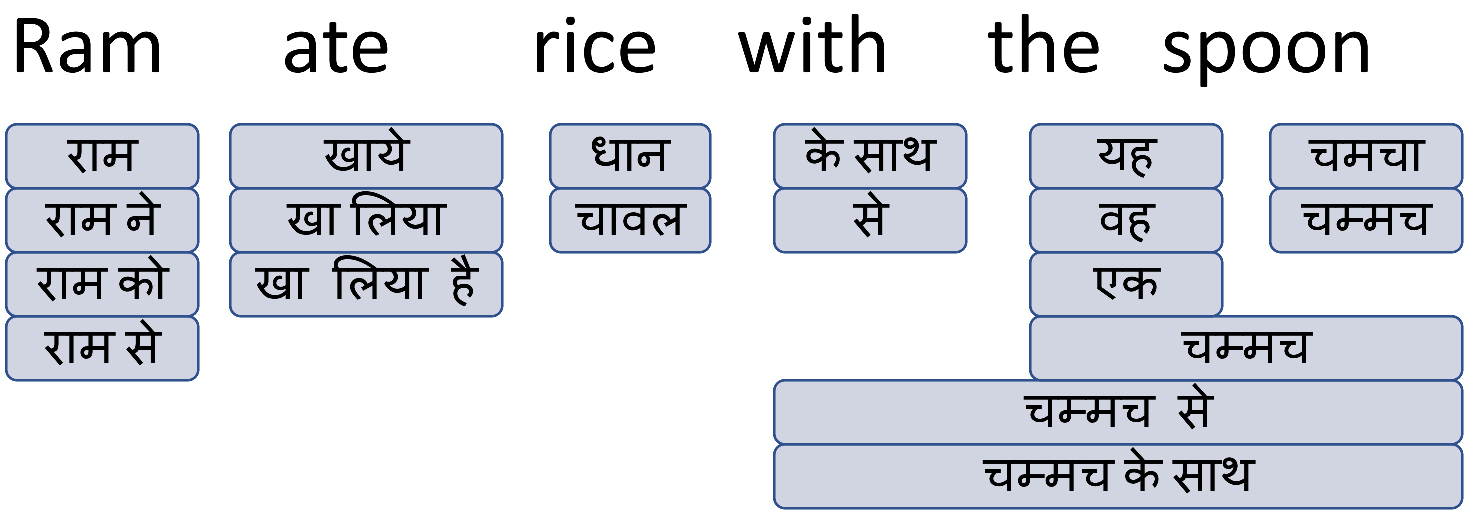
$$\boxed{e^* = \arg\max_{e_i} \sum_i \lambda_i h_i(f_1^I, e_1^J)}$$

An Example of Translation



Reality

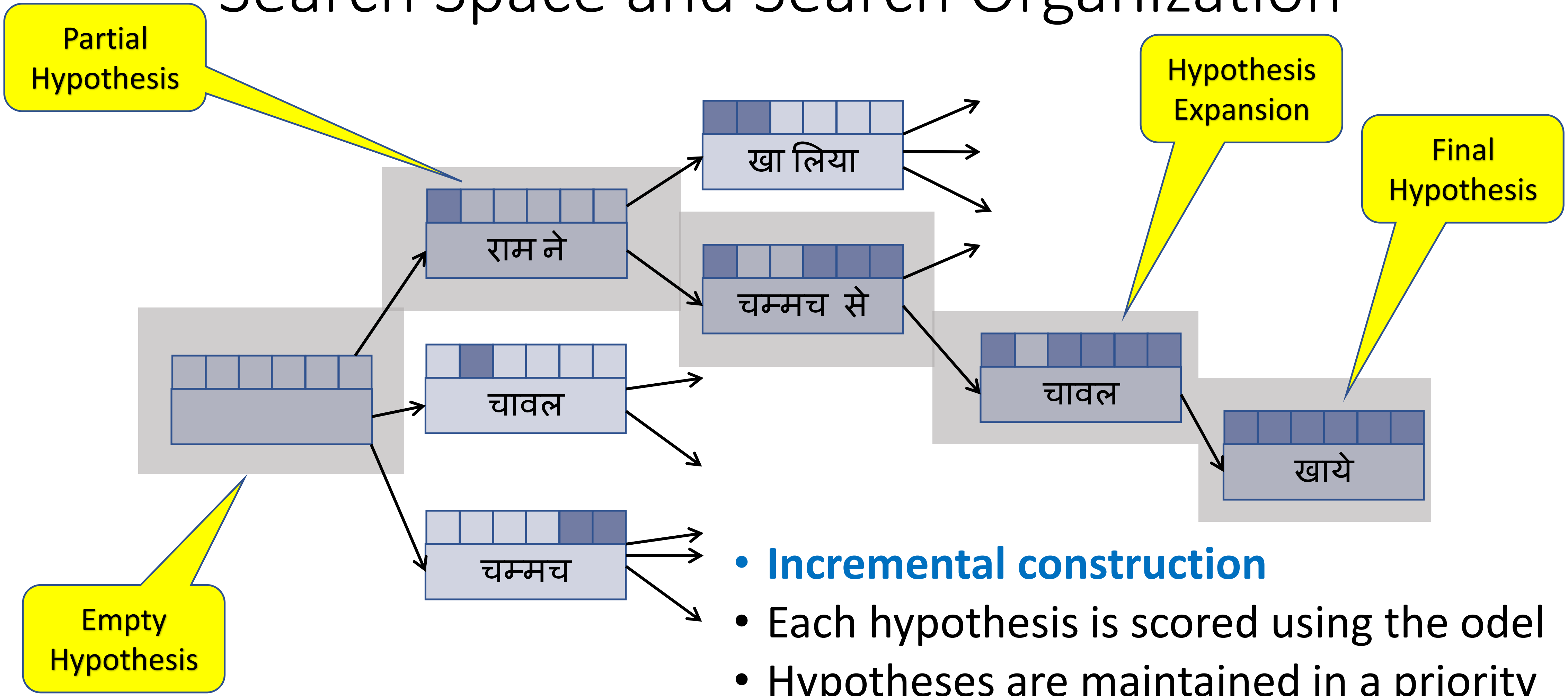
- We picked the phrase translation that made sense to us
- The computer has less intuition
- Phrase table may give many options to translate the input sentence



What is the challenge in decoding?

- The task of decoding in machine translation is to find the best scoring translation according to translation models
- Hard problem, since there is a exponential number of choices, given a specific input sentence
- Shown as an NP complete problem
- Need to come up with heuristic search methods
- No guarantee of finding the best translation

Search Space and Search Organization



- **Incremental construction**
- Each hypothesis is scored using the odel
- Hypotheses are maintained in a priority queue
- Limit to the reordering window for efficiency

Agenda

- What is Machine Translation & why is it interesting?
- Machine Translation Paradigms
- Word Alignment
- Phrase-based SMT
- **Extensions to Phrase-based SMT**
    - Addressing Word-order Divergence
    - Addressing Morphological Divergence
    - Handling Named Entities
- Syntax-based SMT
- Machine Translation Evaluation
- Summary

We have looked at a basic phrase-based SMT system

This system can learn word and phrase translations from parallel corpora

But many important linguistic phenomena need to be handled

- **Divergent Word Order**
- Rich morphology
- Named Entities and Out-of-Vocabulary words

Getting word order right

Phrase based MT is not good at learning word ordering

Solution: Let's help PB-SMT with some preprocessing of the input

Change order of words in input sentence to match order of the words in the target language

Let's take an example

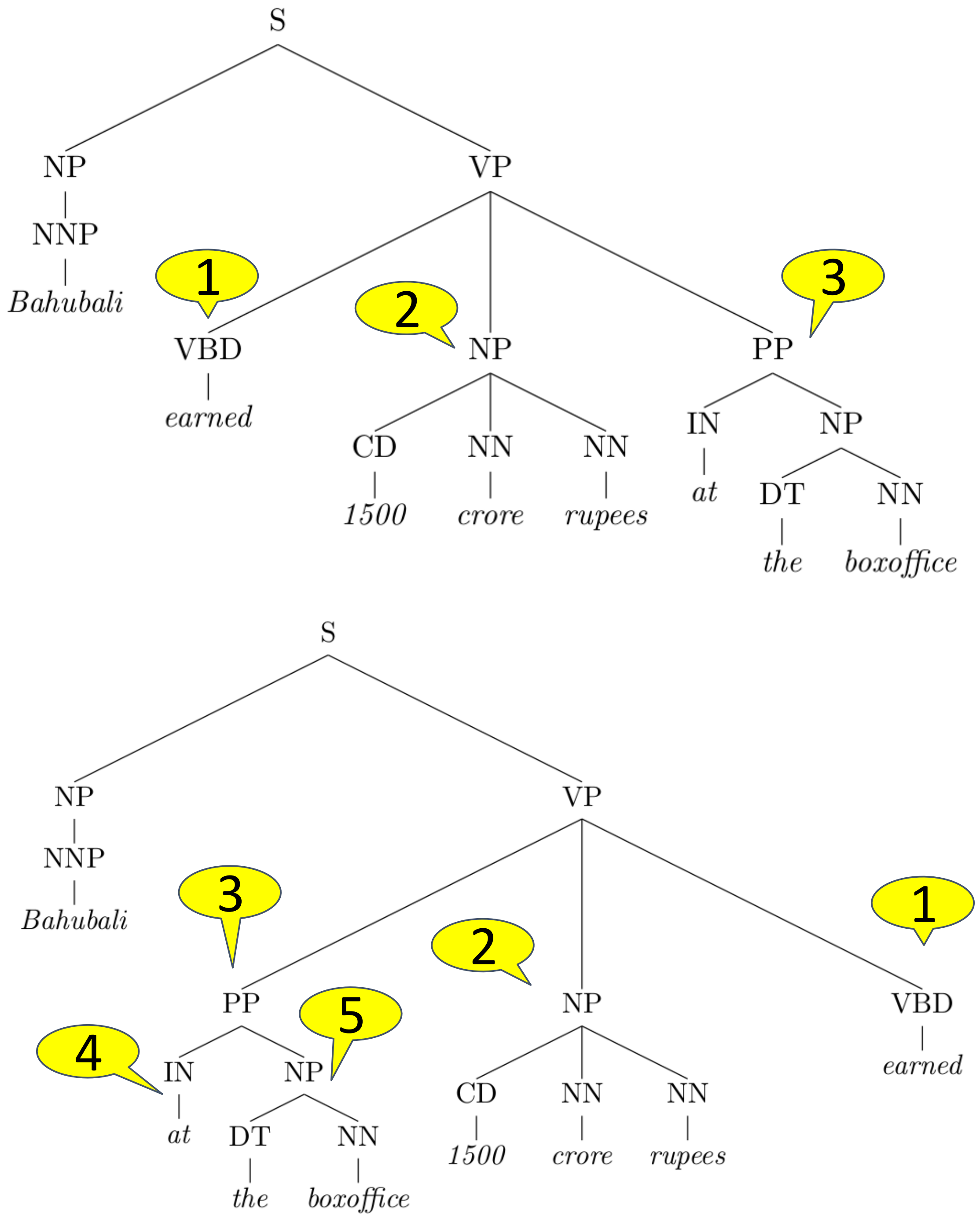
Bahubali earned more than 1500 crore rupee sat the boxoffice

Parse the sentence to understand its syntactic structure

Apply rules to transform the tree

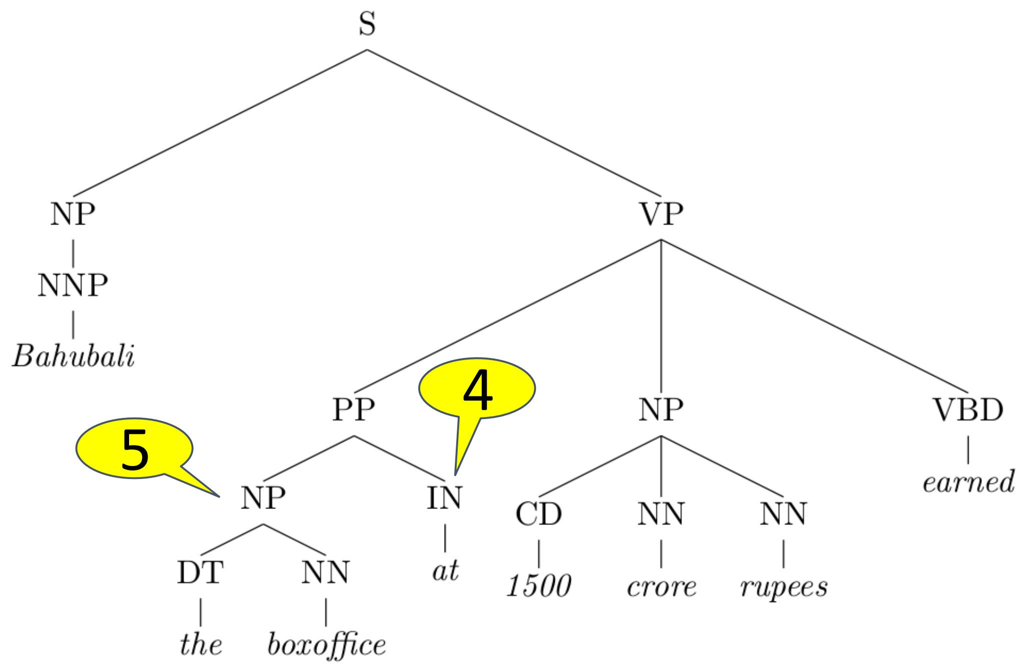
VP → VBD NP PP ⇒ VP → PP NP VBD

This rule captures Subject-Verb-Object to Subject-Object-Verb divergence



Prepositions in English become postpositions in Hindi

PP → IN NP ⇒ PP → NP IN



The new input to the machine translation system is

Bahubali the boxoffice at 1500 crore rupees earned

Now we can translate with little reordering

बाहुबली ने बॉक्सओफिस पर 1500 करोड रुपए कमाए

These rules can be written manually or learnt from parse trees

Better methods exist for generating the correct word order

Incorporate learning of reordering is built into the SMT system

Hierarchical PBSMT ⇒ Provision in the phrase table for limited & simple reordering rules

Syntax-based SMT ⇒ Another SMT paradigm, where the system learns mappings of “treelets” instead of mappings of phrases

We have looked at a basic phrase-based SMT system

This system can learn word and phrase translations from parallel corpora

But many important linguistic phenomena need to be handled

- Divergent Word Order
- **Rich morphology**
- Named Entities and Out-of-Vocabulary words

Language is very productive, you can combine words to generate new words

Inflectional forms of the Marathi word घर

घर	house
घरात	in the house
घरावरती	on the house
घराखाली	below the house
घरामध्ये	in the house
घरामागे	behind the house
घराचा	of the house
घरामागचा	that which is behind the house
घरासमोर	in front of the house
घरासमोरचा	that which is in front of the house
घरांसमोर	in front of the houses

Hindi words with the suffix वाद

साम्यवाद	communism
समाजवाद	socialism
पूंजीवाद	capitalism
जातीवाद	casteism
साम्राज्यवाद	imperialism

The corpus should contains all variants to learn translations

This is infeasible!

Language is very productive, you can combine words to generate new words

Inflectional forms of the Marathi word घर

घर	house
घर ◌ा त	in the house
घर ◌ा वरती	on the house
घर ◌ा खाली	below the house
घर ◌ा मध्ये	in the house
घर ◌ा मागे	behind the house
घर ◌ा चा	of the house
घर ◌ा माग चा	that which is behind the house
घर ◌ा समोर	in front of the house
घर ◌ा समोर चा	that which is in front of the house
घर ◌ा ◌ं समोर	in front of the houses

Hindi words with the suffix वाद

साम्य वाद	communism
समाज वाद	socialism
पूंजी वाद	capitalism
जाती वाद	casteism
साम्राज्य वाद	imperialism

- *Break the words into its component morphemes*
- *Learn translations for the morphemes*
- *Far more likely to find morphemes in the corpus*

We have looked at a basic phrase-based SMT system

This system can learn word and phrase translations from parallel corpora

But many important linguistic phenomena need to be handled

- Divergent Word Order
- Rich morphology
- **Named Entities and Out-of-Vocabulary words**

Some words not seen during train will be seen at test time

These are out-of-vocabulary (OOV) words

***Names** are one of the most important category of OOVs*

⇒ There will always be names not seen during training

How do we translate names like Sachin Tendulkar to Hindi?

What we want to do is map the Roman characters to Devanagari to they sound the same when read → सचिन तेंदुलकर

➔ *We call this process **'transliteration'***

How do we transliterate?

Convert a sequence of characters in one script to another script

s a c h i n →स च ि न

Isn't that a translation problem → at the character level?

Albeit a simpler one,

- *Smaller vocabulary*
- *No reordering*
- *Shorter segments*

Translation between Related Languages

Related Languages

Related by Genealogy

<u>Language Families</u>

Dravidian, Indo-European, Turkic

(Jones, Rasmus, Verner, 18th & 19th centuries, Raymond ed. (2005))

Related by Contact

<u>Linguistic Areas</u>

Indian Subcontinent,
Standard Average European

(Trubetzkoy, 1923)

Related languages may not belong to the same language family!

Key Similarities between related languages

भारताच्या स्वातंत्र्यदिनानिमित्त अमेरिकेतील लॉस एन्जल्स शहरात कार्यक्रम आयोजित करण्यात आला
bhAratAcyA svAta.ntryadinAnimitta ameriketIla lOsa enjalsa shaharAta kAryakrama Ayojita karaNyAta AlA

Marathi

भारता च्या स्वातंत्र्य दिना निमित्त अमेरिके तील लॉस एन्जल्स शहरा त कार्यक्रम आयोजित करण्यात आला
bhAratA cyA svAta.ntrya dinA nimitta amerike tIla lOsa enjalsa shaharA ta kAryakrama Ayojita karaNyAta AlA

Marathi segmented

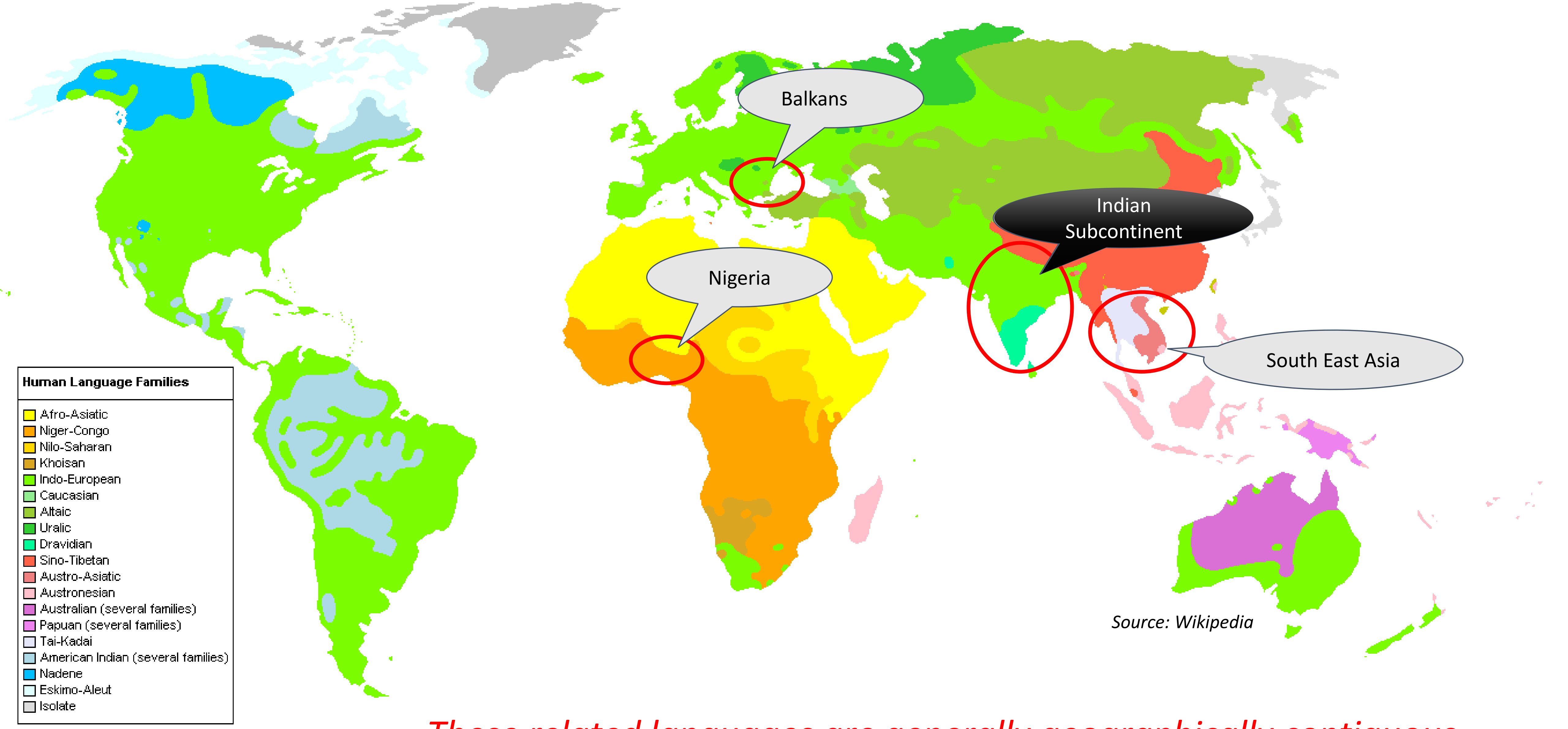
भारत के स्वतंत्रता दिवस के अवसर पर अमरीका के लॉस एन्जल्स शहर में कार्यक्रम आयोजित किया गया
bhArata ke svata.ntratA divasa ke avasara para amarIkA ke losa enjalsa shahara me.n kAryakrama Ayojita kiyA gayA

Hindi

Lexical: share significant vocabulary (cognates & loanwords)

Morphological: correspondence between suffixes/post-positions

Syntactic: share the same basic word order



These related languages are generally geographically contiguous

Naturally, lot of communication between such languages
(government, social, business needs)

Most translation requirements also involves related languages

Between related languages

Hindi-Malayalam
Marathi-Bengali
Czech-Slovak

Related languages ⇐⇒ Link languages

Kannada,Gujarati ⇒ English
English ⇒ Tamil,Telugu

We want to be able to handle a large number of such languages

e.g. 30+ languages with a speaker population of 1 million + in the Indian subcontinent

Lexically Similar Languages

*(Many words having similar **form** and **meaning**)*

- *Cognates*

a common etymological origin

roTI (hi)	*roTIA (pa)*	*bread*
bhai (hi)	*bhAU (mr)*	*brother*

- *Named Entities*

do not change across languages

mu.mbaI (hi)	mu.mbaI (pa)	mu.mbaI (pa)
keral (hi)	k.eraLA (ml)	keraL (mr)

- *Loan Words*

borrowed without translation

matsya (sa)	*matsyalu (te)*	*fish*
pazha.m (ta)	*phala (hi)*	*fruit*

- *Fixed Expressions/Idioms*

MWE with non-compositional semantics

dAla me.n kuCha kAlA honA	*(hi)*	*Something fishy*
dALa mA kAlka kALu hovu	(gu)	

Translation at subword level which exploits lexical similarity

What is a good unit of representation?

Let's take the word EDUCATION as an example

<u>Character</u>: E D U C A T I O N

ambiguity in character mappings

<u>Character n-gram</u>: ED UC AT IO N

Vocabulary size explodes for n>2

<u>Orthographic Syllable</u>

- Break at vowel boundaries
- Approximate syllable

E DU CA TIO N

Variable length

Small Vocabulary

More relevant units

<u>Byte Pair Encoded Unit</u>

- Identify most frequent character substrings as vocabulary
- Motivated from compression theory

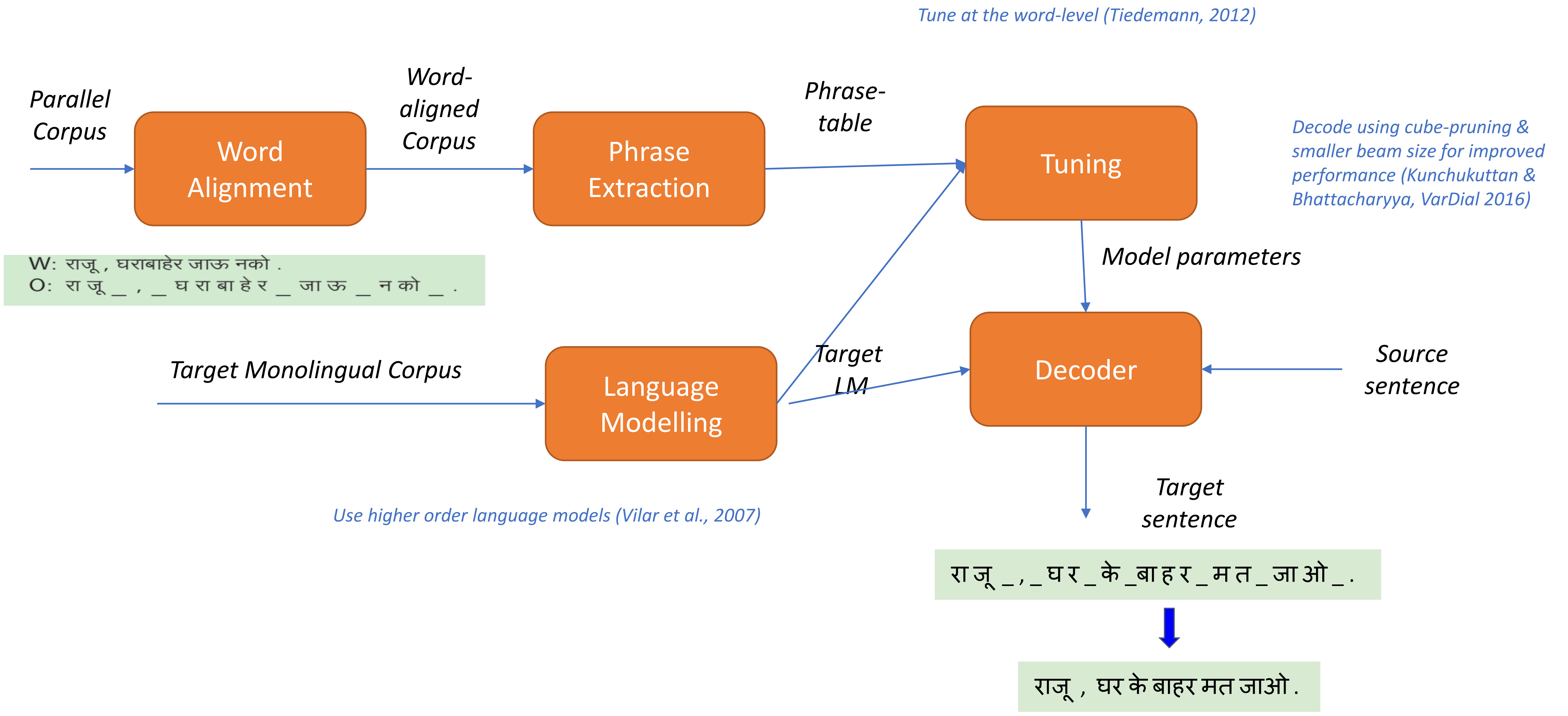
EDU CA TION

Training objective?

What about sentence length?

<u>Sentence Representation</u> → मु म्ब ई _ म हा रा ष्ट्र _ की _ रा ज धा नी _ है _ ।

Adapting SMT for subword-level translation



Agenda

- What is Machine Translation & why is it interesting?
- Machine Translation Paradigms
- Word Alignment
- Phrase-based SMT
- Extensions to Phrase-based SMT
    - Addressing Word-order Divergence
    - Addressing Morphological Divergence
    - Handling Named Entities
- <u>Syntax-based SMT</u>
- Machine Translation Evaluation
- Summary

Problems with Phrase Based models

- Heavy reliance on lexicalization
  - Direct Translation method
  - No generalization
  - Lot of data is required

For similar sentences, sometimes reordering occurs, sometimes it does not

Correct reordering
Oracle bought Sun Microsystems in 2010
ओरेकल 2010 में सन माइक्रोसिस्टम्स को खरीदा

Incorrect Reordering
IBM approached Sun Microsystems in 2008
आईबीएम दरवाजा खटखटाया 2008 में सन माइक्रोसिस्टम्स का

Problems with Phrase Based models (2)

- Learning is very local in nature
  - Local reordering, sense disambiguation learnt
  - Phenomena like word order divergence, recursive structure are non-local

Word order divergence (SVO-SOV) is not learnt

[The USA] [is not engaging] [in war] [with Iran]
[अमरीका] [संलग्न नहीं है] [युद्ध में] [ईरान के साथ]

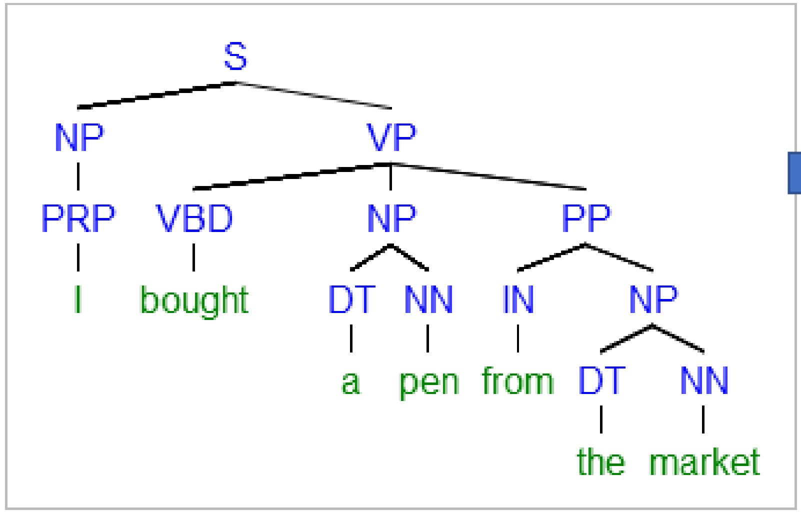
Recursive structure: phrase boundaries are not maintained

[[It is necessary [that the person [who is travelling for the conference]] should get approval prior to his departure]]
यह सम्मेलन के लिए यात्रा कर रहा है, जो व्यक्ति पहले अपने प्रस्थान से अनुमोदन प्राप्त करना चाहिए कि आवश्यक है

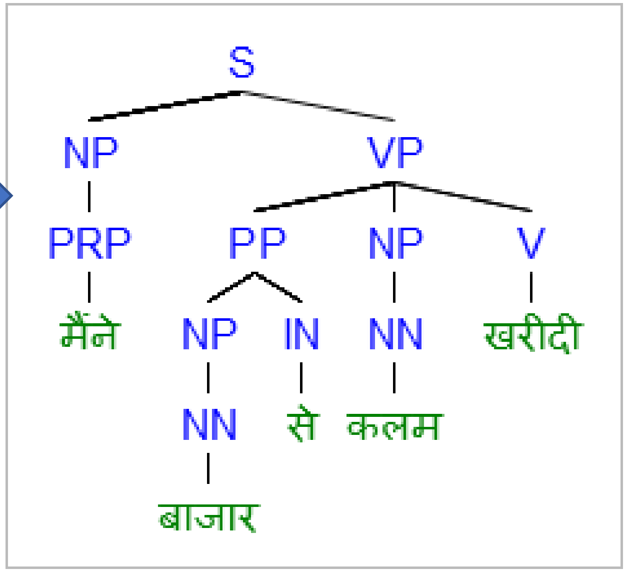
Tree based models

- Source and/or Target sentences are represented as trees
- Translation as Tree-to-Tree Transduction
  - As opposed to string-to-string transduction in PB-SMT
- Parsing as Decoding
  - Parsing of the source language sentence produces the target language sentences

Example



Source Tree

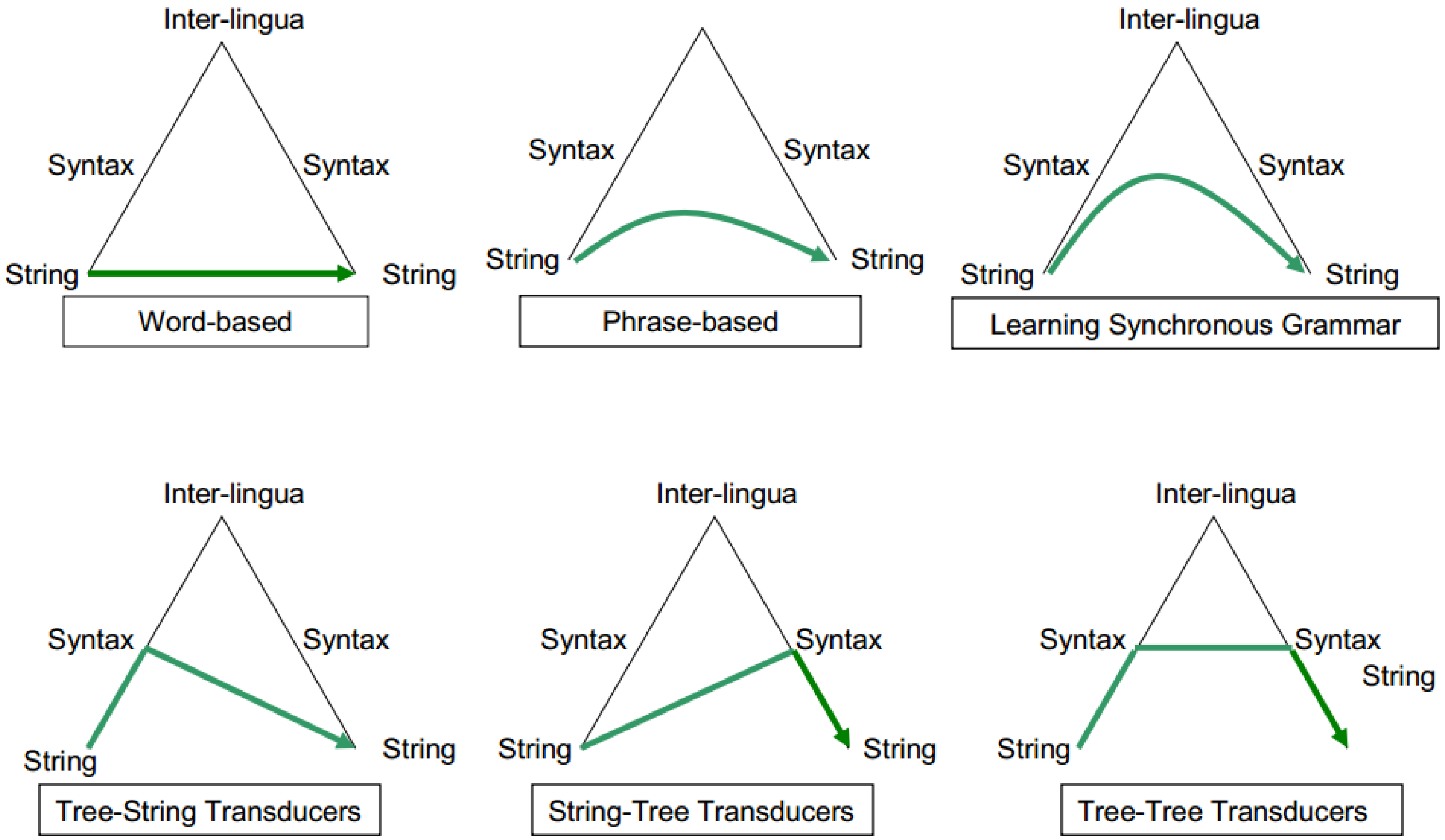


Target Tree

Why tree based model?

- Natural language sentences have a tree-like structure
- Syntax based Reordering
- **Source side tree**: guides decoding by constraining the possible rules that can be applied
- **Target side tree** ensures grammatically correct output

Different flavours of tree-based models



[Slide from Amr Ahmed]

Synchronous Context Free Grammar

- Fundamental formal tool for Tree-based translation models
- An enhanced Context Free Grammar for generating two related strings instead of one
- Alternatively, SCFG defines a tree transducer

Definition

$S \rightarrow NP\ VP$
$VP \rightarrow V$
$VP \rightarrow V\ NP$
$VP \rightarrow VP\ NP\ PP$
$NP \rightarrow NN$
$NN \rightarrow$ market

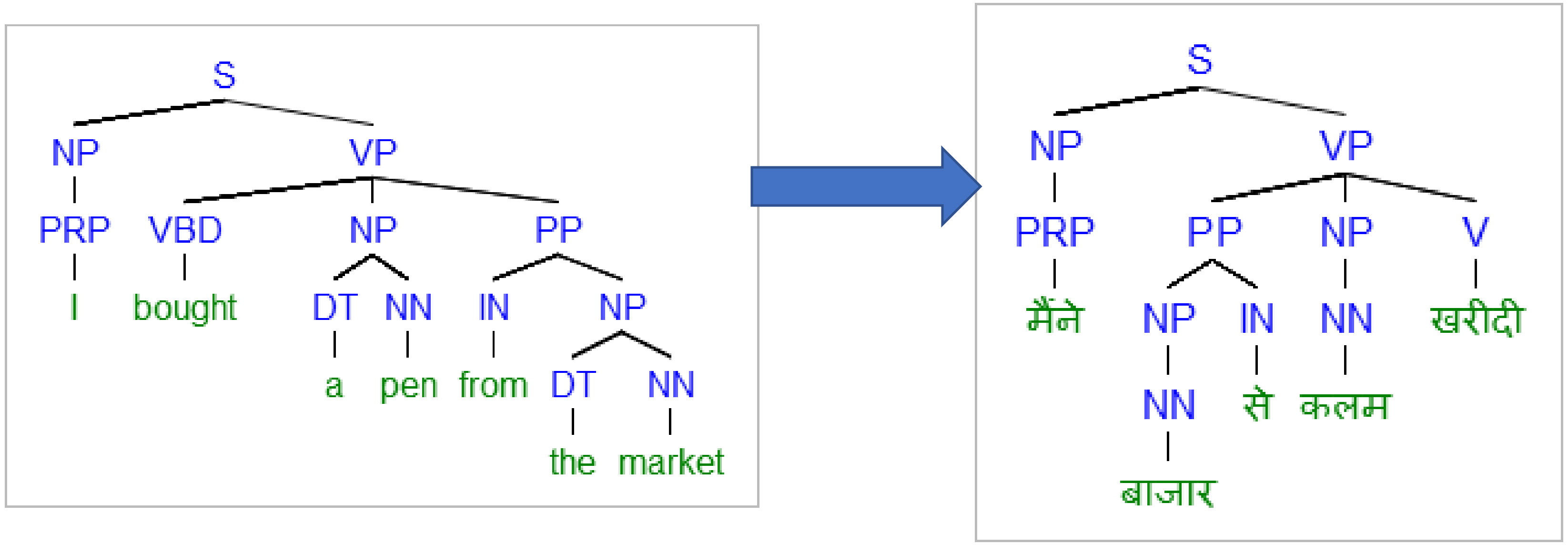
CFG

$S \rightarrow \langle NP_1\ VP_2 ,\ NP_1\ VP_2 \rangle$
$VP \rightarrow \langle V_1 ,\ V_1 \rangle$
$VP \rightarrow \langle V_1\ NP_2 ,\ NP_2\ V_1 \rangle$
$VP \rightarrow \langle V_1\ NP_2\ PP_3 ,\ PP_3\ NP_2\ V_1 \rangle$
$NP \rightarrow \langle NN_1 ,\ NN_1 \rangle$
$NN \rightarrow \langle$ market , बाजार $\rangle$

SCFG

- **Differences of SCFG from CFG:**
  - 2 components on the RHS of production rule
  - Same number of non-terminals
  - Non-terminals have one-one correspondence (index-linked)

Example



Source Tree

Target Tree

Example SCFG for English-Hindi

1. $S \rightarrow < NP_1\ VP_2\ ,\ NP_1\ VP_2 >$
2. $VP \rightarrow < V_1\ ,\ V_1 >$
3. $VP \rightarrow < V_1\ NP_2,\ NP_2\ V_1 >$
4. $VP \rightarrow < V_1\ NP_2\ PP_3\ ,\ PP_3\ NP_2\ V_1 >$
5. $NP \rightarrow < NN_1\ ,\ NN_1 >$
6. $NP \rightarrow < PRP_1\ ,\ PRP_1 >$
7. $PP \rightarrow < IN_1\ NP_2\ ,\ NP_2\ IN_1 >$
8. NN $\rightarrow <$ market , बाजार $>$
9. NN $\rightarrow <$ pen , कलम $>$
10. PRP $\rightarrow <$ I , मैंने $>$
11. V $\rightarrow <$ bought, खरीदी $>$
12. IN $\rightarrow <$ from , से $>$
13. DT $\rightarrow <$ the , $\varepsilon >$
14. DT $\rightarrow <$ a ,$\varepsilon >$

Derivation

Parsing as Decoding!

- S
- < NP_1 VP_2 , NP_1 VP_2 >
- < NP_1 VP_2 , NP_1 VP_2 >
- < PRP_3 VP_2 , PRP_3 VP_2 >
- < I VP_2 , मैंने VP_2 >
- < I V_3 NP_4 PP_5 , मैंने PP_5 NP_4 V_3 >
- < I bought NP_4 PP_5 , मैंने PP_5 NP_4 खरीदी >
- < I bought DT_6 NN_7 PP_5 , मैंने PP_5 DT_6 NN_7 खरीदी >
- < I bought a NN_7 PP_5 , मैंने PP_5 NN_7 खरीदी >
- < I bought a pen PP_5 , मैंने PP_5 कलम खरीदी >
- < I bought a pen IN_8 NP_9 , मैंने NP_9 IN_8 कलम खरीदी >
- < I bought a pen from NP_9 , मैंने NP_9 से कलम खरीदी >
- < I bought a pen from DT_{10} NN_{11} , मैंने DT_{10} NN_{11} से कलम खरीदी >
- < I bought a pen from the NN_{11} , मैंने NN_{11} से कलम खरीदी >
- < I bought a pen from the market, मैंने बाजार से कलम खरीदी >

Reordering and Relabeling among Child Nodes

- The only operations a SCFG allows is:
  - reordering among child nodes

$$VP \rightarrow < V_1\ NP_2\ PP_3\ ,\ PP_3\ NP_2\ V_1 >$$

  - Re-labelling of nodes

$$VP \rightarrow < V_1\ NP_2\ PP_3\ ,\ PREPP_3\ NP_2\ V_1 >$$
$$PP/PREPP \rightarrow < IN_1\ NP_2\ ,\ NP_2\ IN_2 >$$

- The condition is overly restrictive, hardly any pair of languages would follow such a grammar
  - Useful for representing non-linguistic formalisms like hierarchical model, Inverse Transduction Grammar
- Other tree-based models like Tree Substitution Grammars are more powerful

Hierarchical Phrase Based Models

- Learns a SCFG purely from data
  - no source, target side parsers used
- Learns an undifferentiated grammar
  - Grammar does not have notion of different types of non-terminals (eg. NP, VP, etc.)
  - Only one type of non-terminal, called X
- Production rules are of the form

$$X \rightarrow <\alpha\, X_1\, \beta\, X_2\, \gamma\, X_3\ ,\ X_2\, \alpha'\, \beta'\, X_3\, X_1>$$

- Useful in generalizing learning of reordering among phrases

The SCFG for the Hierarchical Model

- A rule is of the form:

$$X \rightarrow \langle \gamma, \alpha, \sim \rangle$$

where, ~ is one-one correspondence between non-terminals

$X \rightarrow$ < with X_1 , X_1 के साथ >

- In addition, there are “glue” rules for the initial state

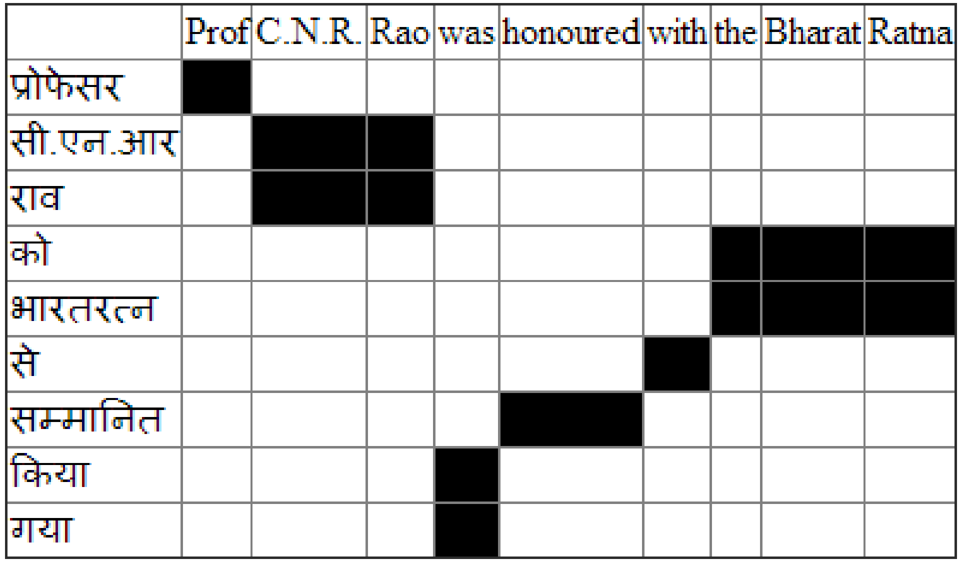
$$S \rightarrow \langle S_{\boxed{1}} X_{\boxed{2}}, S_{\boxed{1}} X_{\boxed{2}} \rangle$$
$$S \rightarrow \langle X_{\boxed{1}}, X_{\boxed{1}} \rangle$$

$$S \rightarrow \langle S_{\boxed{1}} X_{\boxed{2}}, S_{\boxed{1}} X_{\boxed{2}} \rangle$$
$$S \rightarrow \langle X_{\boxed{1}}, X_{\boxed{1}} \rangle$$

Formal, Not Linguistic

- "Formal", but not linguistic
  - The SCFG grammar does not correspond to any natural language
  - "non-linguistic" phrases (not words) as basic units
- The HPBSMT model defines a formal SCFG model for reordering of these "phrases“ in PBMST
- A custom designed engineering solution for a purpose

Example of rule generation



Extracted Phrase alignments

Extracted Rules

Phrase Pair	Extracted Rule
(was honoured, सम्मानित किया गया)	X → < was X_1 , X_1 किया गया >
(with the Bharat Ratna, भारतरत्न से)	X → < with X_1 , X_1 से >
(was honoured with the Bharat Ratna, भारतरत्न से सम्मानित किया गया)	X → < was X_1 with X_2, X_2 से X_1 किया गया >

Summary

- Tree based models can better handle syntactic phenomena like reordering, recursion
- Basic formalism: Synchronous Context Free Grammar
- Decoding: Parsing on the source side
  - CYK Parsing
  - Integration of the language model presents challenge
- Parsers required for learning syntax transfer
- Without parsers, some weak learning is possible with hierarchical PBSMT

Agenda

- What is Machine Translation & why is it interesting?
- Machine Translation Paradigms
- Word Alignment
- Phrase-based SMT
- Extensions to Phrase-based SMT
    - Addressing Word-order Divergence
    - Addressing Morphological Divergence
    - Handling Named Entities
- Syntax-based SMT
- <u>Machine Translation Evaluation</u>
- Summary

Motivation

- How do we judge a good translation?
- Can a machine do this?
- Why should a machine do this?
  - Because human evaluation is time-consuming and expensive!
  - Not suitable for rapid iteration of feature improvements

What is a good translation?

Evaluate the quality with respect to:

- **Adequacy**: How good the output is in terms of preserving content of the source text
- **Fluency**: How good the output is as a well-formed target language entity

For example, I am attending a lecture

मैं एक व्याख्यान बैठा हूँ
Main ek vyaakhyan baitha hoon
I a lecture sit (Present-first person)
I sit a lecture : Adequate but not fluent

मैं व्याख्यान हूँ
Main vyakhyan hoon
I lecture am
I am lecture: Fluent but not adequate.

Human Evaluation

Common techniques:

1. Assigning fluency and adequacy scores *(Direct Assessment)*
2. Ranking translated sentences relative to each other *(Relative Ranking)*

Direct Assessment

> **How do you rate your Olympic experience?**
>
> — Reference

> **How do you value the Olympic experience?**
>
> — Candidate translation

- Average score over the entire test set
- Gives a sense of the absolute translation quality
- Evaluators use their own perception to rate
- Often adequacy/fluency scores correlate: undesirable

Adequacy:

is the meaning translated correctly?

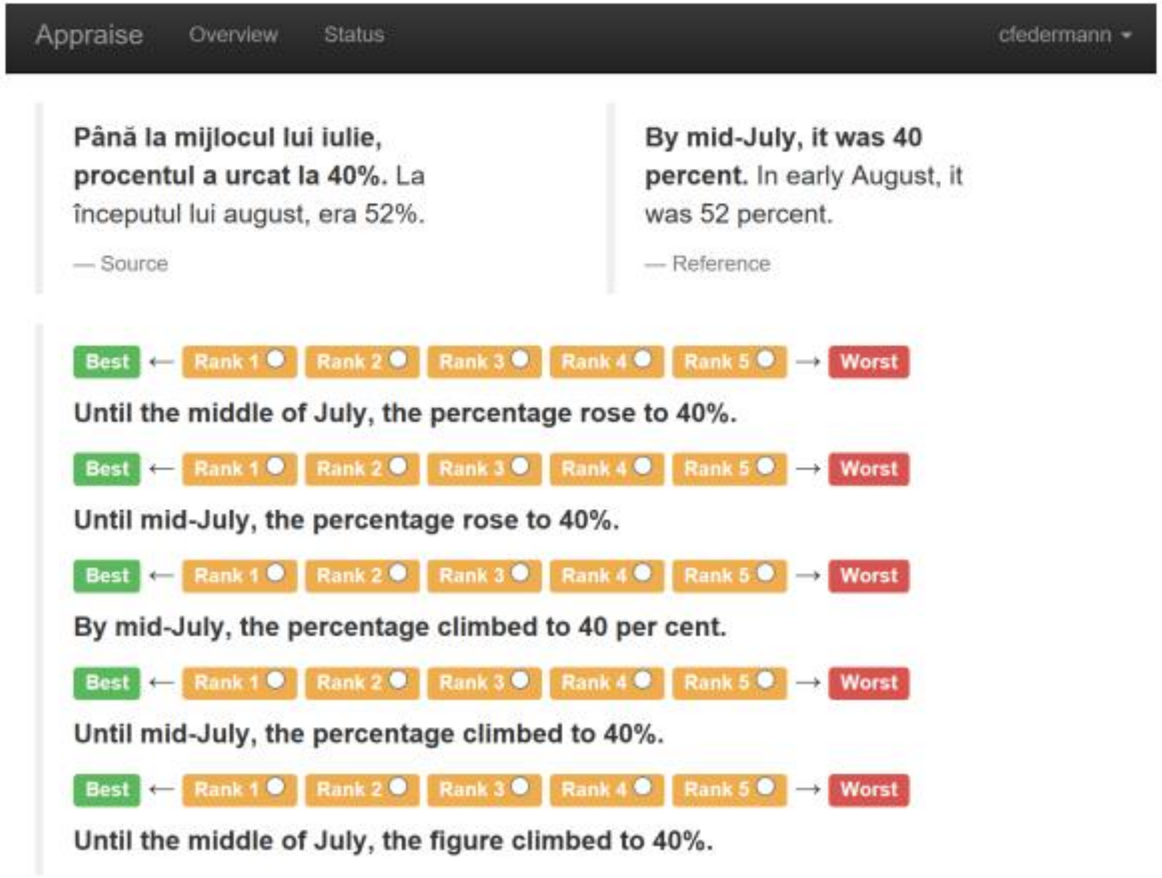
5 = All
4 = Most
3 = Much
2 = Little
1 = None

Fluency:

Is the sentence grammatically valid?

5 = Flawless
4 = Good
3 = Non-native
2 = Disfluent
1 = Incomprehensible

Ranking Translations



	1	2	3	4	5
F				●	
A				●	
B		●			
J					●
H			●		

$$A > B, A = F, A > H, A < J$$
$$B < F, B < H, B < J$$
$$F > H, F < J$$
$$H < J$$

Wins(S_i,S_j) = number of times system S_i is ranked better than system S_j

$$\text{score}(S_i) = \frac{1}{|\{S\}|} \sum_{S_j \neq S_i} \frac{\text{wins}(S_i, S_j)}{\text{wins}(S_i, S_j) + \text{wins}(S_j, S_i)}$$

- Can provide only relative quality judgments
- Faster to collect data than judgments than Direct Assessment
- Correlates well with direct assessment
- Another popular adaptive algorithm: Trueskill

Automatic Evaluation

The closer a machine translation is to a professional human translation, the better it is.

- Given: A corpus of good quality human reference translations
- Output: A numerical “translation closeness” metric
- Given (ref,sys) pair, score = f(ref,sys) ➔ $\mathbb{R}$

where,

sys (candidate Translation): Translation returned by an MT system

ref (reference Translation): ‘Perfect’ translation by humans

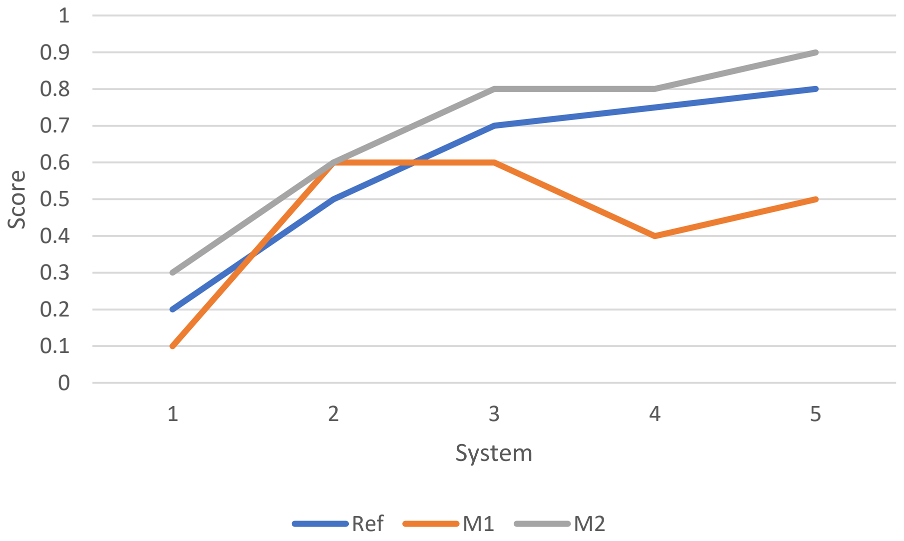
Multiple references are better

Some popular automatic evaluation metrics

- BLEU (Bilingual Evaluation Understudy)
- TER (Translation Edit Rate)
- METEOR (Metric for Evaluation of Translation with Explicit Ordering)

How good is an automatic metric?

How well does it correlate with human judgment?



BLEU

- Most popular MT evaluation metric
- Requires only reference translations
  - No additional resources required
- Precision-oriented measure
- Difficult to interpret absolute values
- Useful to compare two systems

Formulating BLEU (Step 1): Precision

I had lunch now.

Reference 1: मैने अभी खाना खाया
maine abhi khana khaya
I now food ate
I ate food now.

Reference 2 : मैने अभी भोजन किया
maine abhi bhojan kiyaa
I now meal did
I did meal now

Candidate 1: **मैने** अब **खाना खाया**
maine ab khana khaya
I *now food ate*
I ate food now

matching unigrams: 3,
matching bigrams: 1

Candidate 2: **मैने अभी** लंच एट
maine abhi lunch ate.
I now lunch ate
I ate lunch(OOV) now(OOV)

matching unigrams: 2,
matching bigrams: 1

Unigram precision: Candidate 1: 3/4 =0.75, Candidate 2: 2/4 = 0.5
Bigram precision: Candidate 1: 0.33, Candidate 2 = 0.33

Precision: Not good enough

Reference: मुझ पर तेरा सुरूर छाया

mujh-par tera suroor chhaaya

me-on your spell cast

Your spell was cast on me

Candidate 1: मेरे **तेरा सुरूर छाया**   matching unigram: 3

mere tera suroor chhaaya

my your spell cast

Your spell cast my

Candidate 2: **तेरा तेरा तेरा सुरूर**   matching unigrams: 4

tera tera tera suroor

your your your spell

Unigram precision: Candidate 1: 3/4 = 0.75, Candidate 2: 4/4 = 1

Formulating BLEU (Step 2): Modified Precision

- Clip the total count of each candidate word with its maximum reference count
- $\text{Count}_{\text{clip}}$(n-gram) = min (count, max_ref_count)

Reference: मुझ पर तेरा सुरूर छाया

mujh-par tera suroor chhaaya

me-on your spell cast

Your spell was cast on me

Candidate 2: **तेरा तेरा तेरा सुरूर**

tera tera tera suroor

your your your spell

- matching unigrams:

  (**तेरा : **min(3, 1) = 1) (सुरूर : min (1, 1) = 1)

Modified unigram precision: 2/4 = 0.5

Recall for MT (1/2)

- Candidates shorter than references
- Reference: क्या ब्लू लंबे वाक्य की गुणवत्ता को समझ पाएगा?

kya blue lambe vaakya ki guNvatta ko samajh paaega?

Will blue long sentence-of quality (case-marker) understandable(III-person-male-singular)?

Will blue be able to understand quality of long sentence?

Candidate: लंबे वाक्य

lambe vaakya

long sentence

long sentence

modified unigram precision: 2/2 = 1
modified bigram precision: 1/1 = 1

Recall for MT (2/2)

- Candidates longer than references

Reference 1: मैने भोजन किया

maine bhojan kiyaa

I meal did

I had meal

Candidate 1: मैने खाना भोजन किया

maine khaana bhojan kiya

I food meal did

I had food meal

Modified unigram precision: 1

Reference 2: मैने खाना खाया

maine khaana khaaya

I food ate

I ate food

Candidate 2: मैने खाना खाया

maine khaana khaaya

I food ate

I ate food

Modified unigram precision: 1

Formulating BLEU (Step 3): Incorporating recall

- Sentence length indicates ‘best match’
- Brevity penalty (BP):
  - Multiplicative factor
  - Candidate translations that match reference translations in length must be ranked higher

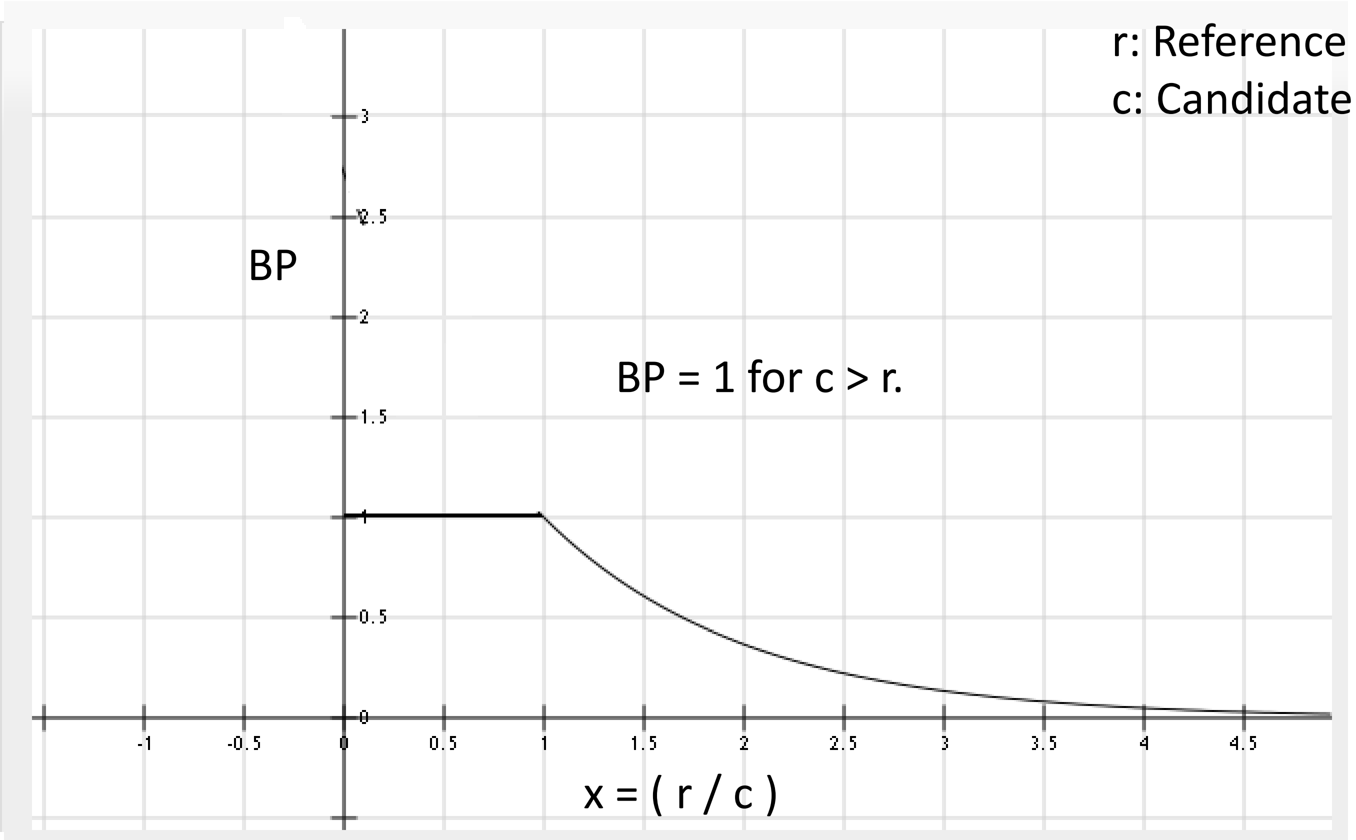
Candidate 1: लंबे वाक्य

Candidate 2: क्या ब्लू लंबे वाक्य की गुणवत्ता समझ पाएगा?

Formulating BLEU (Step 3): Brevity Penalty

$$\mathrm{BP} = \begin{cases} 1 & \text{if } c > r \\ e^{(1-r/c)} & \text{if } c \leq r \end{cases}$$

r: Reference sentence length
c: Candidate sentence length



Formula from [2]
Graph drawn using www.fooplot.com

BP leaves out longer translations

Why?

Translations longer than reference are already penalized by modified precision

$$p_n = \frac{\sum_{C \in \{Candidates\}} \sum_{n\text{-}gram \in C} Count_{clip}(n\text{-}gram)}{\sum_{C' \in \{Candidates\}} \sum_{n\text{-}gram' \in C'} Count(n\text{-}gram')}$$

Formula from [2]

BLEU score

Recall -> Brevity Penalty

$$\mathrm{BP} = \begin{cases} 1 & \text{if } c > r \\ e^{(1-r/c)} & \text{if } c \leq r \end{cases}$$

Precision -> Modified n-gram precision

$$p_n = \frac{\sum_{C \in \{Candidates\}} \sum_{n\text{-}gram \in C} Count_{clip}(n\text{-}gram)}{\sum_{C' \in \{Candidates\}} \sum_{n\text{-}gram' \in C'} Count(n\text{-}gram')}$$

$$\text{BLEU} = \mathrm{BP} \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right)$$

Formula from [2]

METEOR: Criticisms of BLEU

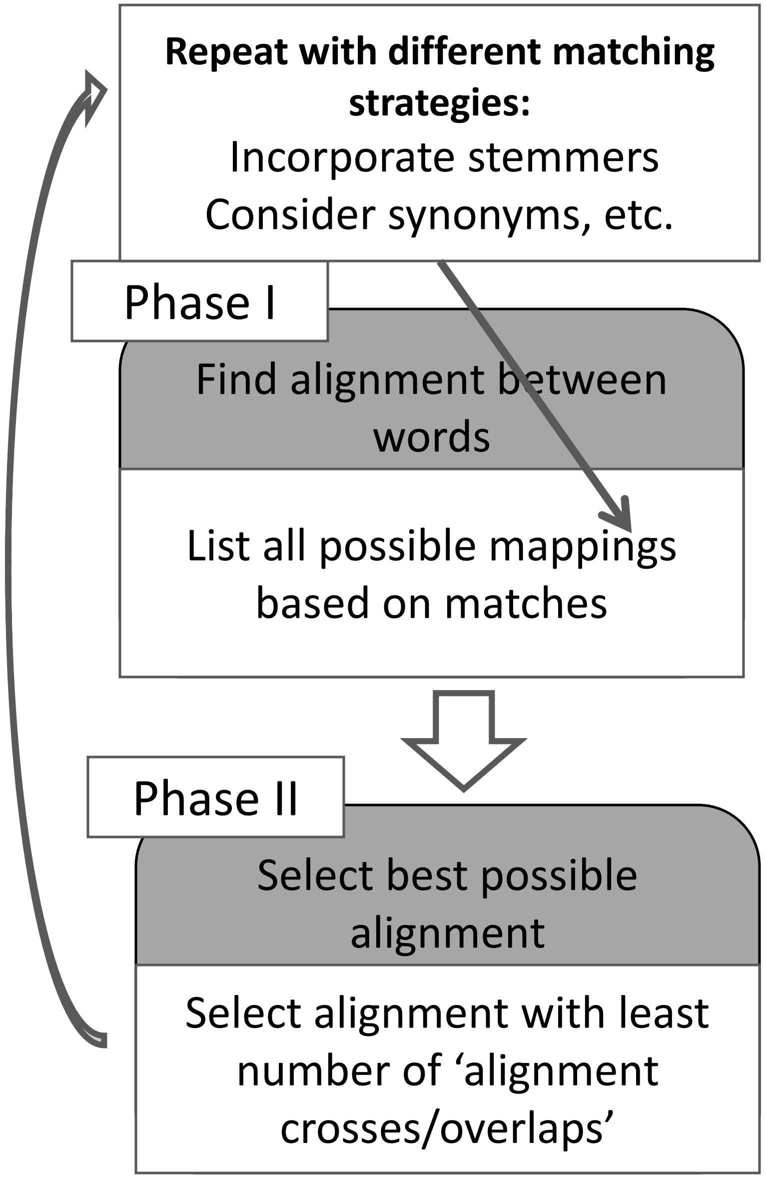
- Brevity penalty is not a good measure of recall
- Higher order n-grams may not indicate grammatical correctness of a sentence
- BLEU is often zero. Should a score be zero?

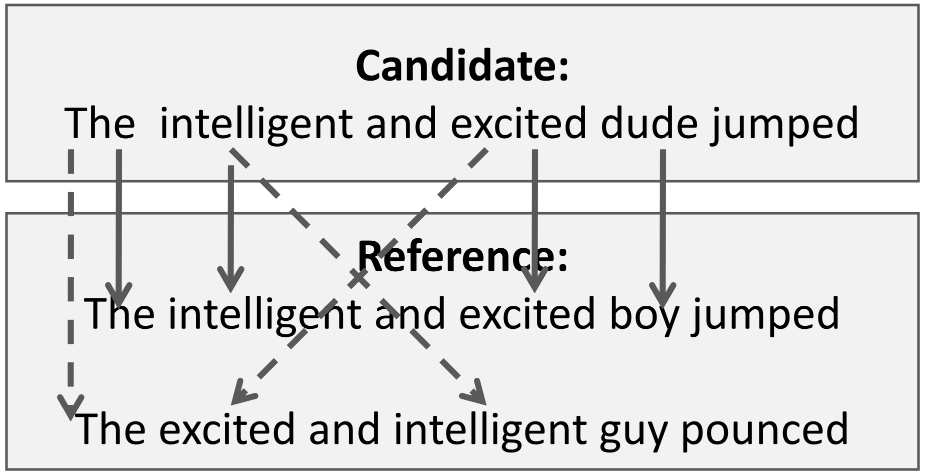
METEOR

Aims to do better than BLEU

Central idea: Have a good unigram matching strategy

METEOR: Process





METEOR: The score

- Using unigram mappings, precision and recall are calculated. Then, harmonic mean:

$$\boxed{Fmean = \frac{10PR}{R + 9P}}$$

$$\boxed{Score = Fmean * (1 - Penalty)}$$

$$\boxed{Penalty = 0.5 * \left(\frac{\#chunks}{\#unigrams\_matched} \right)}$$

Penalty: Find 'as many chunks' that match

The bright boy sits on the black bench

$\boxed{\text{The intelligent guy}}$ sat $\boxed{\text{on the dark bench}}$

More accurate -> Less #chunks, Less penalty
Less accurate -> More #chunks, more penalty

METEOR v/s BLEU

	METEOR	BLEU
Handling incorrect words	Alignment chunks. Matching can be done using different techniques: Adaptable	N-gram mismatch
Handling incorrect word order	Chunks may be ordered in any manner. METEOR does not capture this.	N-gram mismatch
Handling recall	Idea of alignment incorporates missing word handling	Precision cannot detect 'missing' words. Hence, brevity penalty!

$$Score = Fmean * (1 - Penalty)$$

$$\text{BLEU} = \text{BP} \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right)$$

Agenda

- What is Machine Translation & why is it interesting?
- Machine Translation Paradigms
- Word Alignment
- Phrase-based SMT
- Extensions to Phrase-based SMT
  - Addressing Word-order Divergence
  - Addressing Morphological Divergence
  - Handling Named Entities
- Syntax-based SMT
- Machine Translation Evaluation
- <u>Summary</u>

Summary

- Machine Translation is a challenging and exciting NLP problem
- Machine Translation is important to build multilingual NLP systems
- Rule-based systems provide principled linguistic approaches to build translation systems
- Statistical MT systems provide ways to handling uncertaintly
- Incorporating Neural Networks in SMT
    - Distributed Representations are a strength of neural network
    - Use NN-based LM and TM instead of discrete counterparts
- SMT and NMT
    - SMT is useful when corpora available is limited
    - SMT is useful for translation of rare words

Thank you!

anoopk@cse.iitb.ac.in

https://www.cse.iitb.ac.in/~anoopk

The material in the presentation draws from an earlier tutorial I was part of. For a more comprehensive treatment of the material please refer to the tutorial on 'Machine learning for Machine Translation' at ICON 2013 conducted by Prof. Pushpak Bhattacharyya, Piyush Dungarwal, Shubham Gautam and me. You can find the tutorial slides here: https://www.cse.iitb.ac.in/~anoopk/publications/presentations/icon_2013_smt_tutorial_slides.pdf

Acknowledgments: Thanks to Prof. Pushpak Bhattacharyya, Aditya Joshi, Shubham Gautam and Kashyap Popat for some of the slides and diagrams