CS460/626 : Natural Language Processing/Speech, NLP and the Web

Lecture 33: Transliteration

Pushpak Bhattacharyya
CSE Dept.,
IIT Bombay

8th Nov, 2012

Transliteration

ट्रॅन्स्लीटरेशन

Credit: lot of material from seminar of Maoj (PhD student)
Purva, Mugdha, Aditya, Manasi (M.Tech students)

What is transliteration?__

- Task of converting a word from one alphabetic script to another

  Used for:

- **Named entities**
- गांधीजी : Gandhiji
- **Out of vocabulary words**
- बॅंक : Bank

Transliteration for OOV words

- Name searching (people, places, organizations) constitutes a large proportion of search
- Words of foreign origin in a language - *Loan Words*
    - ❖Example: बस (bus), स्कूल (school)
- Such words not found in the dictionary are called *“Out Of Vocabulary (OOV) words”* in CLIR/MT

Machine Transliteration – The Problem

- Graphemes – Basic units of written language (English – 26 letters, Devanagari – 92 including matraas)
- Definition

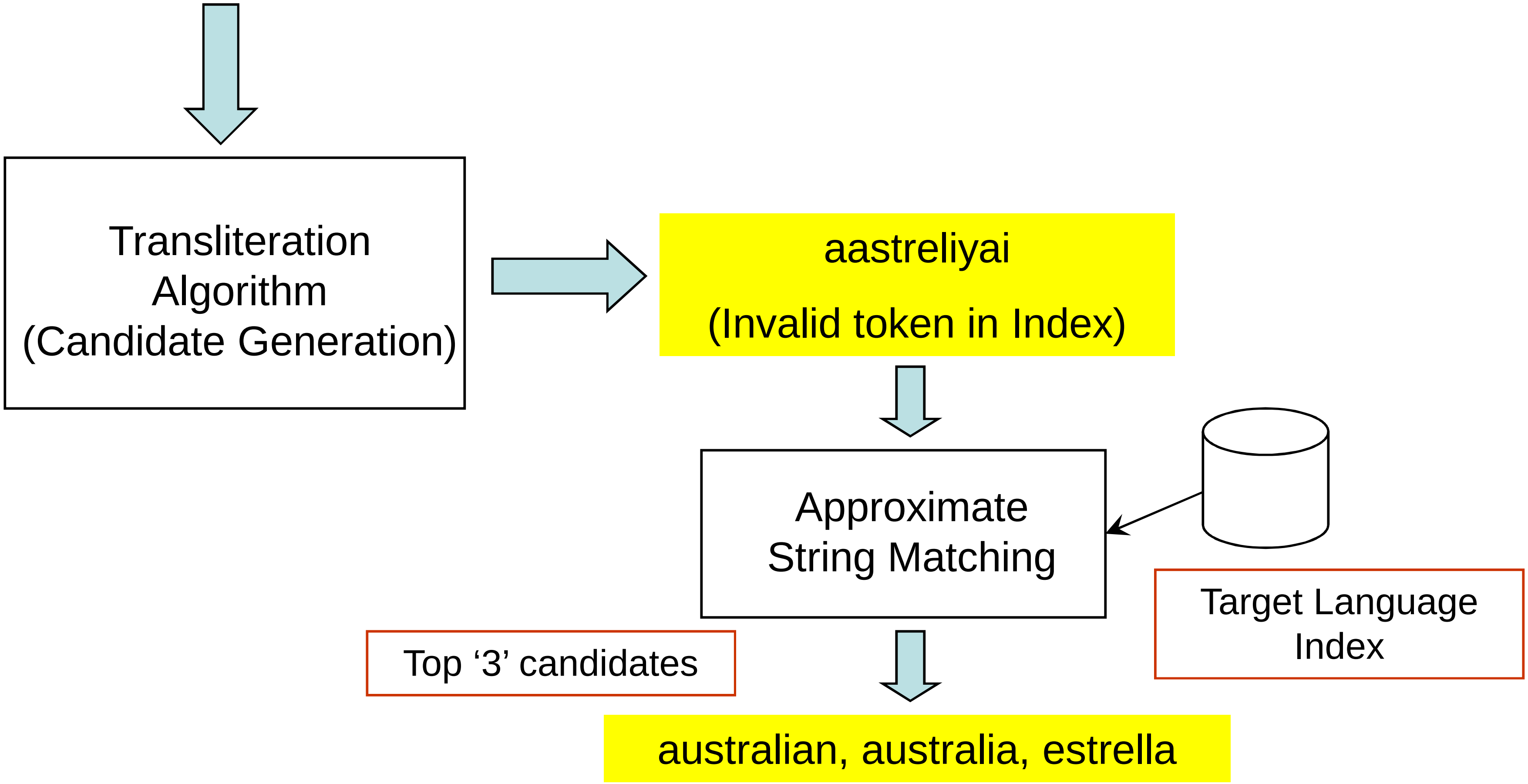
“The process of automatically mapping an given grapheme sequence in source language to a valid grapheme sequence in the target language such that it preserves the pronunciation of the original source word”

Challenges in Machine Transliteration

- Lot of ambiguities at the grapheme level *esp.* while dealing with non-phonetic languages
  - ❖ Example: Devanagari letter *क* has multiple grapheme mappings in English *{ca, ka, qa, c, k, q, ck}*
- Presence of silent letters
  - ❖ Pneumonia – नूमोनिया
- Difference of scripts causes spelling variations *esp.* for loan words
  - ❖ रिलीस, रिलीज, जार्ज, जॉर्ज, बैंक, बॅंक

An Example from CLEF 2007

आस्ट्रेलियाई प्रधानमंत्री



Transliteration Algorithm (Candidate Generation)

aastreliyai

(Invalid token in Index)

Approximate String Matching

Target Language Index

Top ‘3’ candidates

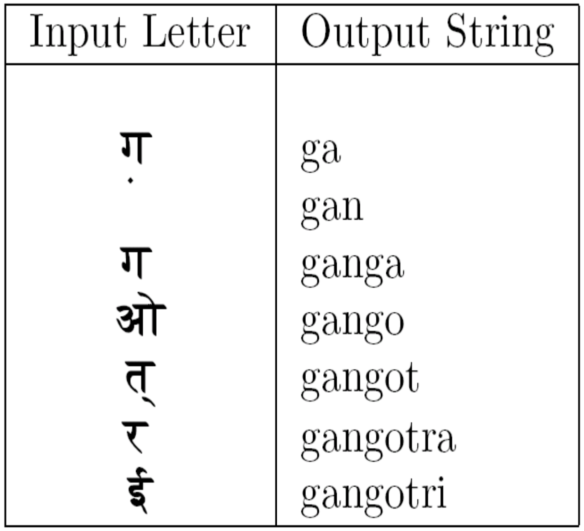
australian, australia, estrella

Candidate Generation Schemes

- Takes an input Devanagari word and generates most likely transliteration candidates in English
- Any standard transliteration scheme could be used for candidate generation
- In our current work, we have experimented with
  - ❖Rule Based Schemes
    - o Single Mapping
    - o Multiple Mapping
- Pre-Storing Hindi Transliterations in Index

Rule Based Transliteration

- Manually defined mapping for each Devanagari grapheme to English grapheme(s)
- Devanagari being a phonetic script, easy to come up with such rules
- Single Mapping
  - ❖ Each Devanagari grapheme has only a single mapping to English grapheme(s)
  - ❖ Example: न – {na}
- A given Devanagari word is transliterated from left-right



Input Letter	Output String
ग	ga
ं	gan
ग	ganga
ओ	gango
त्	gangot
र्	gangotra
ई	gangotri

Rule Based Transliteration (Contd..)

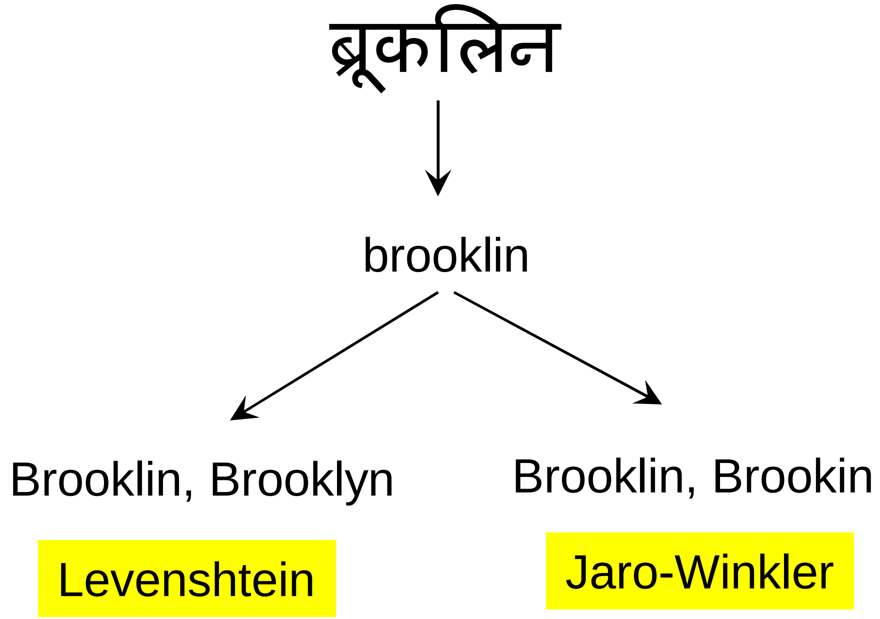
- Multiple Mapping
  - Each Devanagari grapheme has multiple mappings to target English grapheme(s) *Example:* न – {na,kn,n}
  - May lead to very large number of possible candidates
  - Not possible to efficiently rank and perform approximate matching
- Pruning Candidates
  - At each stage rank and retain only top *‘n’* desirable candidates
  - Desirability based on probability of forming a valid spelling in English language
  - Bigram letter model trained on words of English language

Evaluation Metrics

- Transliteration engine outputs ranked list of English transliterations
- Following metrics used to evaluate various transliteration techniques
    - ❖ Accuracy – Percentage of words where right transliteration was retrieved as one of the candidates in list
    - ❖ Mean Reciprocal Rank (MRR) – Used for capturing efficiency of ranking

$$MRR = \sum_{i=1}^{N} \frac{1}{Rank(i)}$$

Example result



ब्रूकलिन

brooklin

Brooklin, Brooklyn

Levenshtein

Brooklin, Brookin

Jaro-Winkler

Overview__

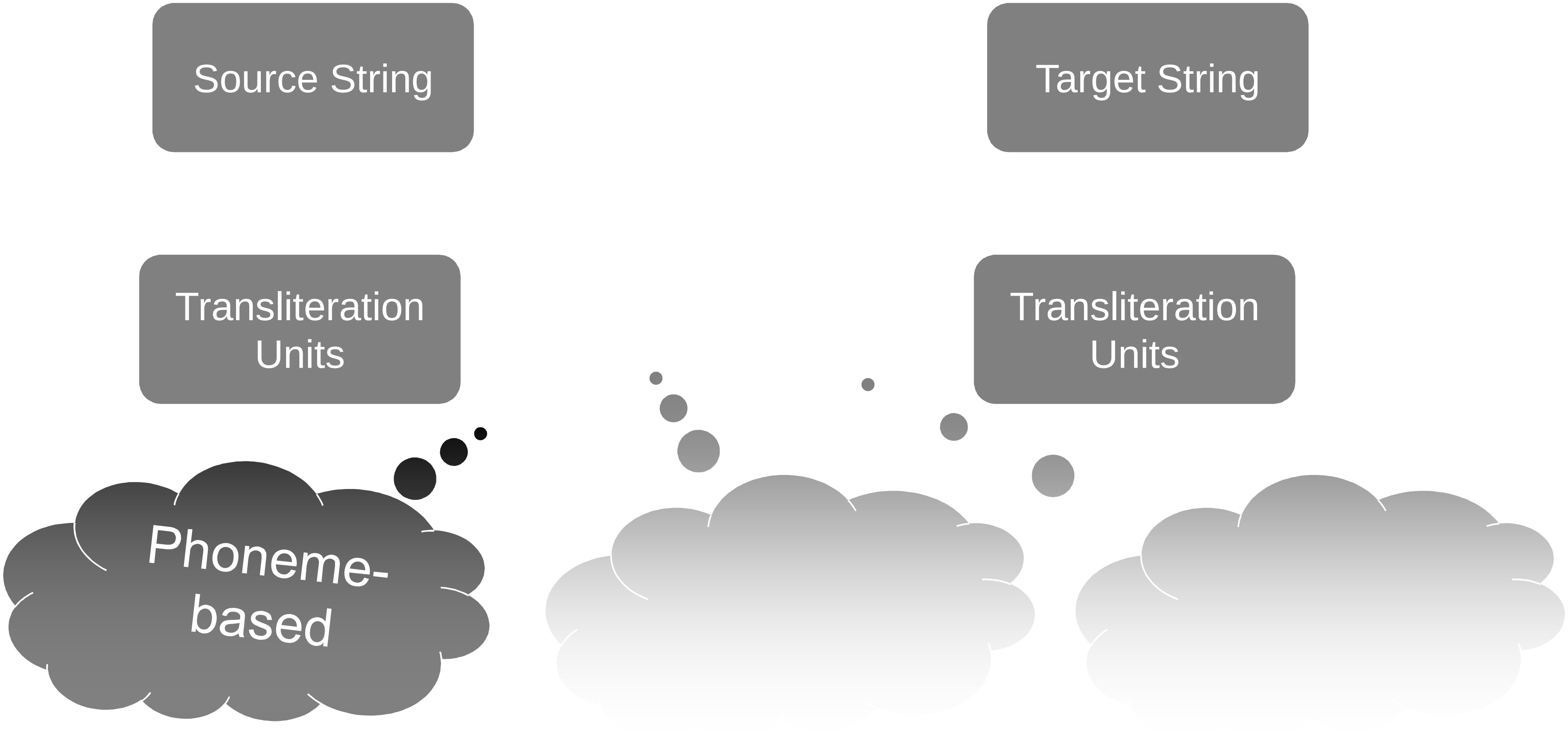
Source String

Target String

Transliteration Units

Transliteration Units

Contents__



Source String

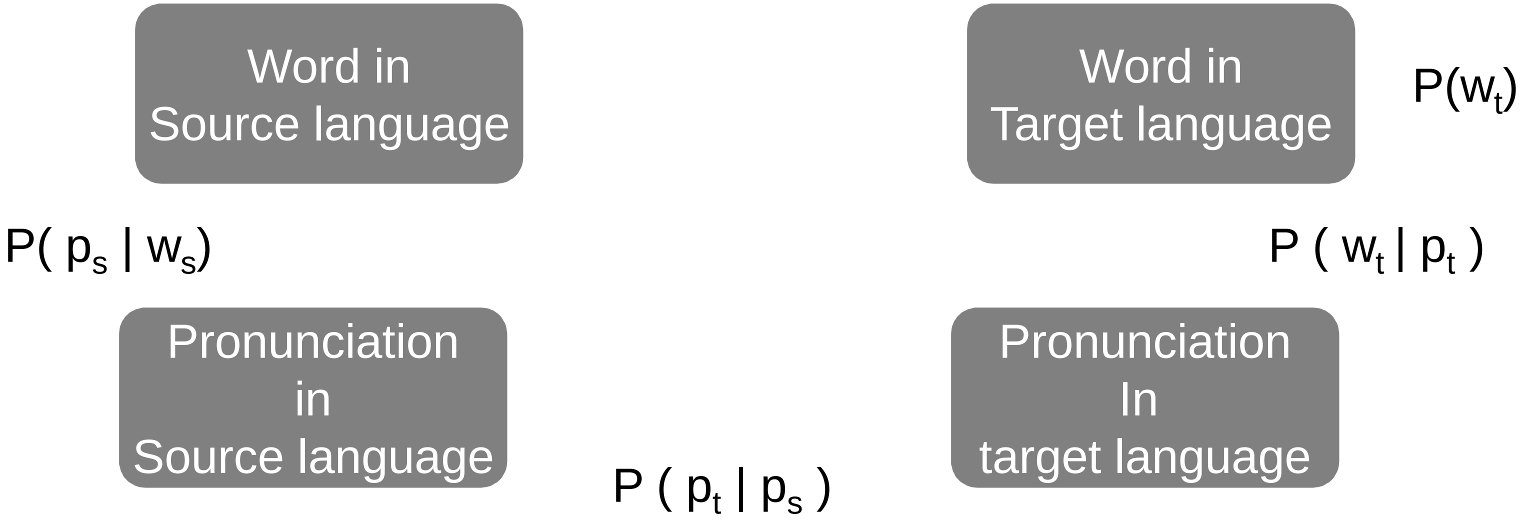
Target String

Transliteration Units

Transliteration Units

Phoneme-based

Phoneme-based approach__



Word in Source language

Word in Target language

$P(w_t)$

$P(p_s | w_s)$

$P (w_t | p_t)$

Pronunciation in Source language

Pronunciation In target language

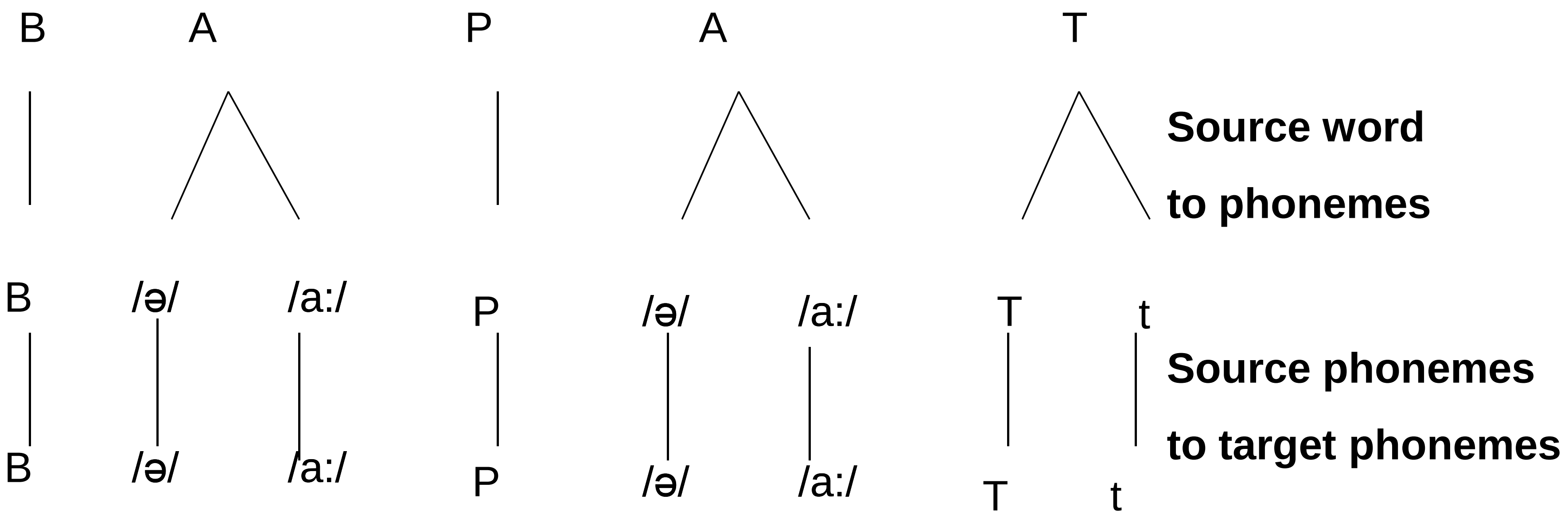
$P (p_t | p_s)$

$$W_t^* = \text{argmax} (P (w_t). P (w_t | p_t) . P (p_t | p_s) . P (p_s | w_s))$$

Note: **Phoneme** is the smallest linguistically distinctive unit of sound.

Phoneme-based approach__

Transliterating 'BAPAT'



B A P A T

B /ə/ /a:/ P /ə/ /a:/ T t

B /ə/ /a:/ P /ə/ /a:/ T t

Source word

to phonemes

Source phonemes

to target phonemes

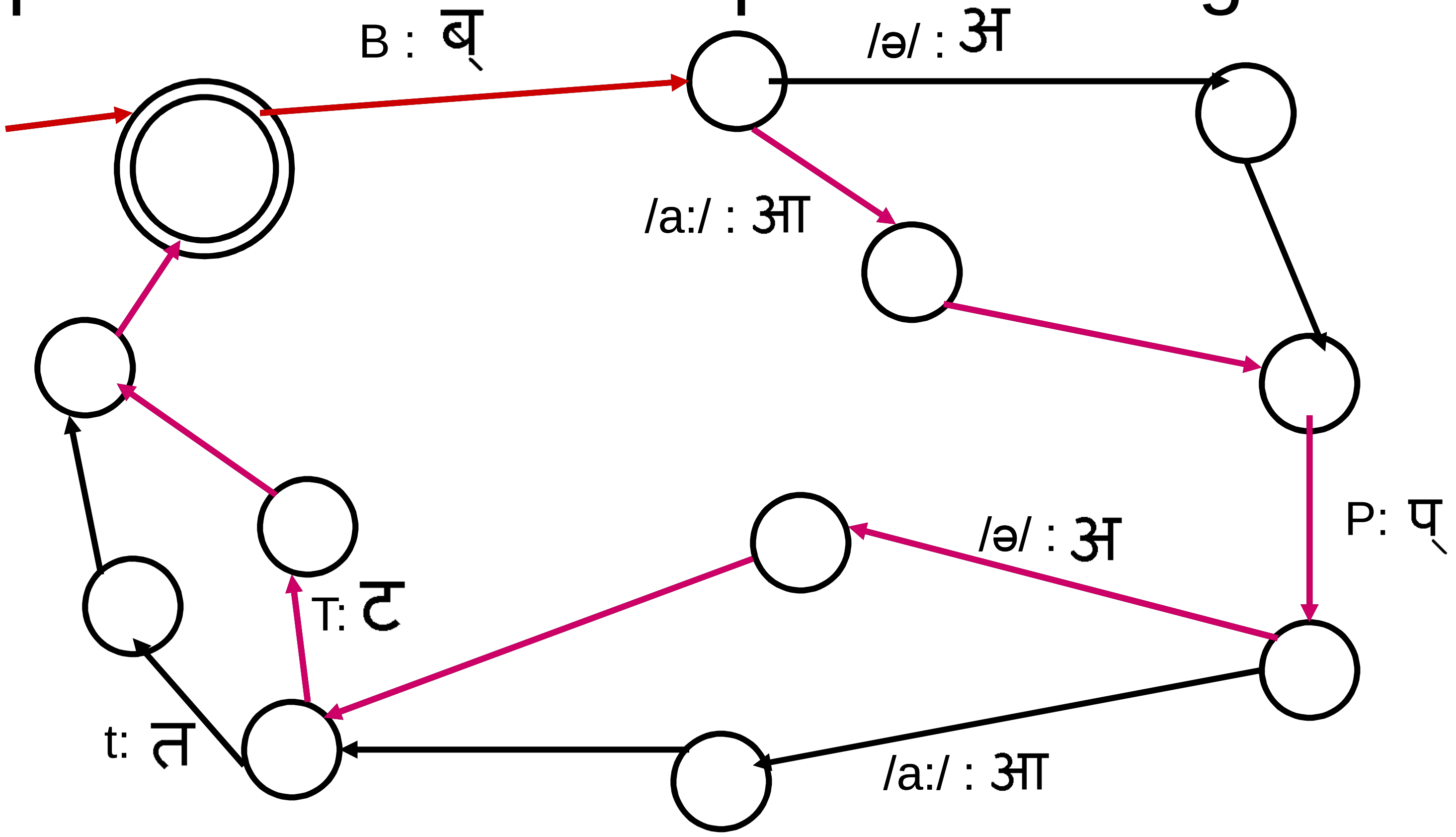
Step I :

Consider each character of the word

Step II :
Converting to phoneme seq.
Step III :
Converting to target phoneme seq.

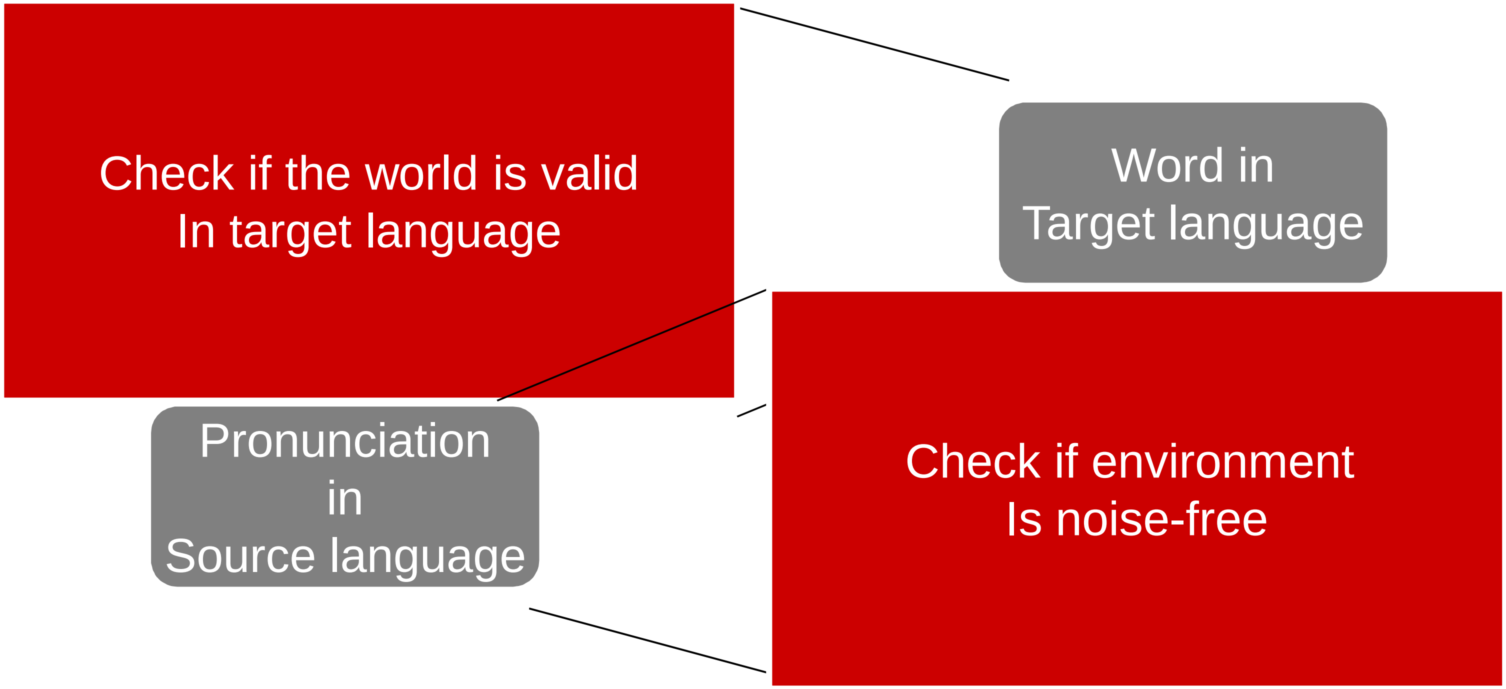
Phoneme-based approach__

Step IV : Phoneme sequence to target string



Output :

Concerns__



Check if the world is valid
In target language

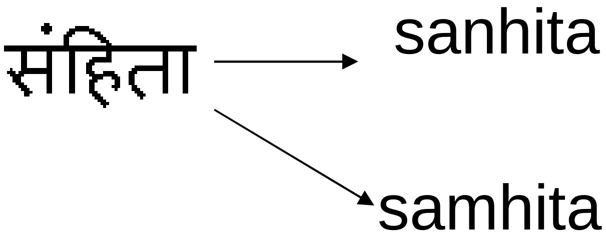
Word in
Target language

Pronunciation
in
Source language

Check if environment
Is noise-free

Issues in phonetic model

- Unknown pronunciations



संहिता

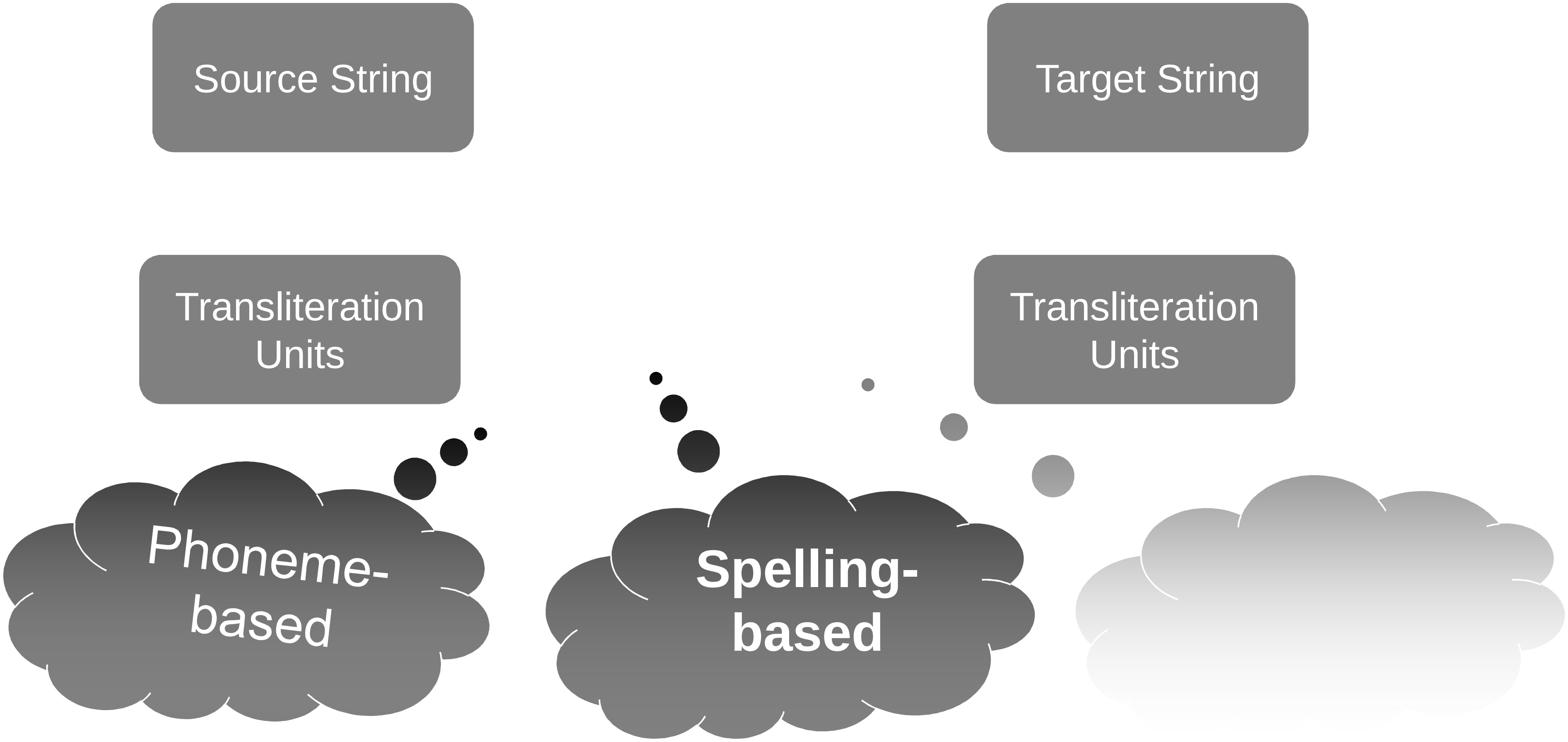
sanhita

samhita

- Back-transliteration can be a problem

Johnson → जॉनसन → Jonson

Contents



Source String

Target String

Transliteration Units

Transliteration Units

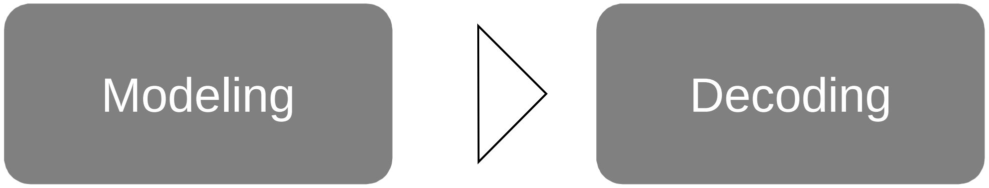
Phoneme-based

Spelling-based

LM based method

- Particularly developed for Chinese
- **Chinese** : Highly ideographic
- Example :

- Two main steps:



Modeling

Decoding

Image courtesy: wikimedia-commons

Modeling step

- A bilingual dictionary in the source and target language

John	जॉन
Georgia	जॉर्जिया
Geology	जियोलॉजी

- From this dictionary, the character mapping between the source and target language is learnt

Geo	जॉ
Geo	जियो

The word “Geo” has two possible mappings, the “context” in which it occurs is important

Modeling step

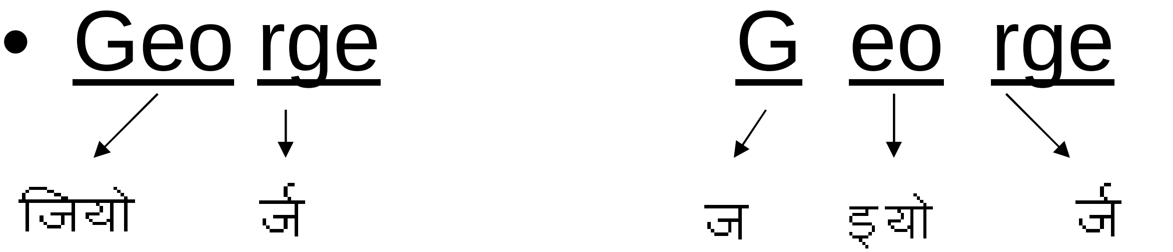
- N-gram Mapping :
- < Geo, जॉ > < rge, र्ज >
- < Geo, जियो > < lo, लॉ >

$$P(E,C) = P(\alpha,\beta,\gamma) = \prod_{k=1}^{K} P(<e,c>_k \mid <e,c>_{k-n+1}^{k-1})$$

- This concludes the modeling step

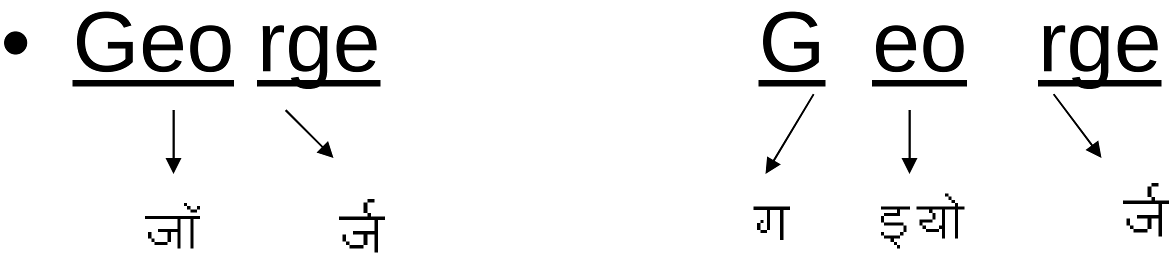
Decoding step__

- Consider the transliteration of the word “George”.
- Alignments of George:



- Geo rge G eo rge

जियो र्ज ज इयो र्ज

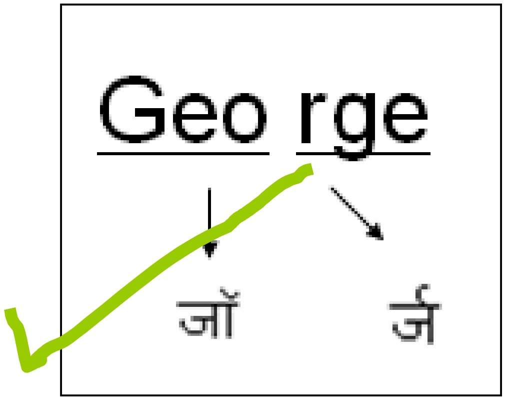


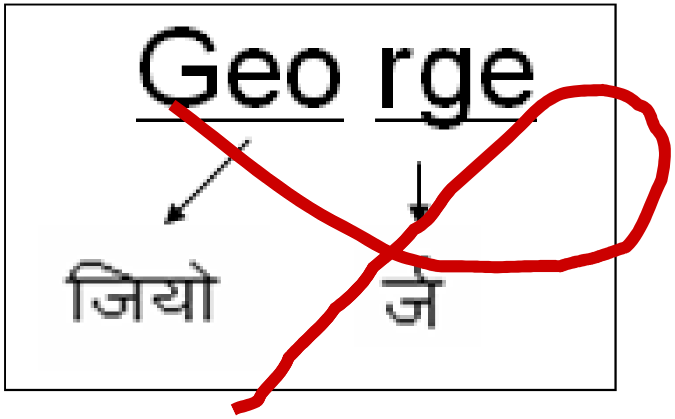
- Geo rge G eo rge

जॉ र्ज ग इयो र्ज

Decoding step …_

Decision to be made between….





- The context mapping <Geo, जॉ > <rge. र्ज > is present in the map-dictionary
- Using $\overline{\beta} = \arg\max_{\beta,\gamma} P(\alpha,\beta,\gamma)$…..

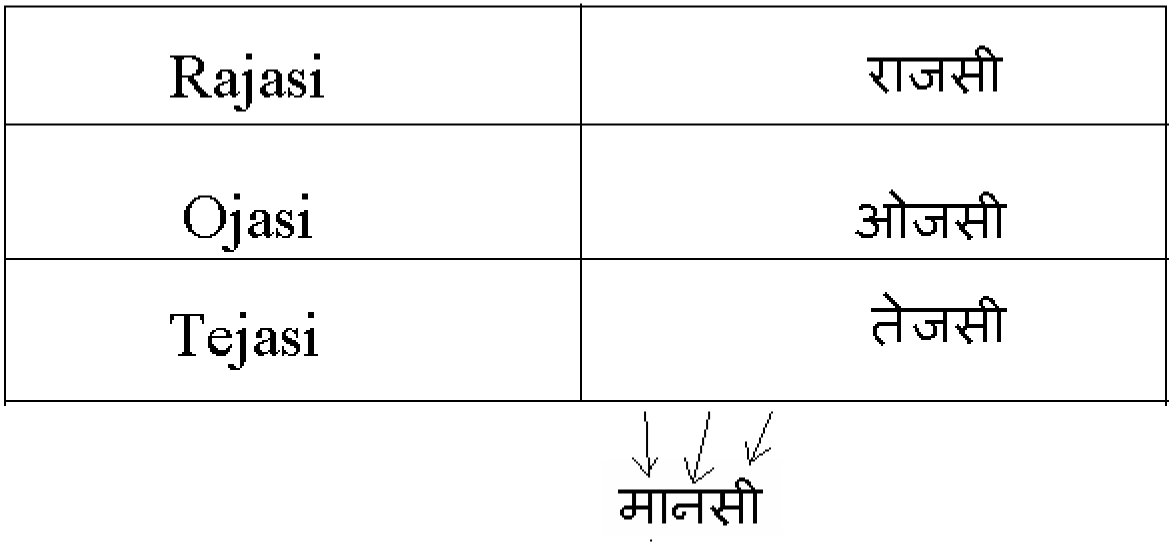
Transliteration Alignment

- Where do the n-gram statistics come from?

Ans.: Automatic analysis of the bilingual dictionary

- How to align this dictionary?

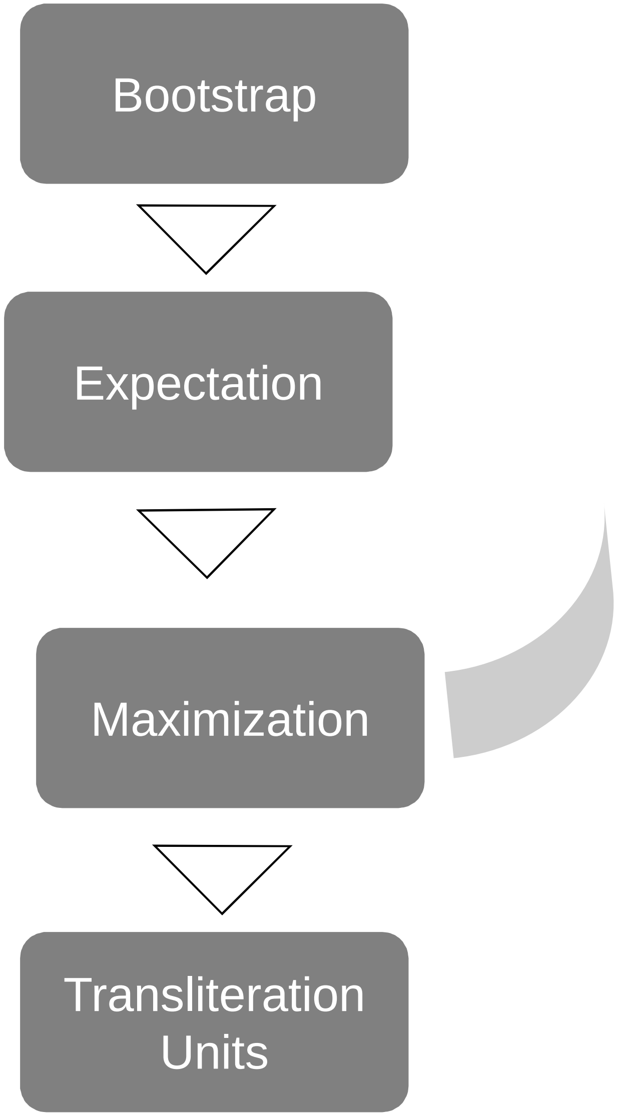
Ans. : Using EM-algorithm



Rajasi	राजसी
Ojasi	ओजसी
Tejasi	तेजसी

मानसी

EM Algorithm



Bootstrap initial random alignment

Update n-gram statistics to estimate probability distribution

Apply the n-gram TM to obtain n

Ra ja si | रा ज सी

Calculating...

P(< ja , ज	0.552
P(< a , ज > | < si , सी >)	0.004

$$\gamma = \arg\max_{\gamma} P(\alpha, \beta, \gamma)$$

P (Ojasi , ओजसी ,<Oj ,ओ > < a , ज >< si , सी >)

P (Ojasi , , सी >)

Oj a si | ओ ज सी

↓

O ja si | ओ ज सी

“Parallel” Corpus

Phoneme	Example	Translation
-------	-------	-----------
AA	odd	AA D
AE	at	AE T
AH	hut	HH AH T
AO	ought	AO T
AW	cow	K AW
AY	hide	HH AY D
B	be	B IY

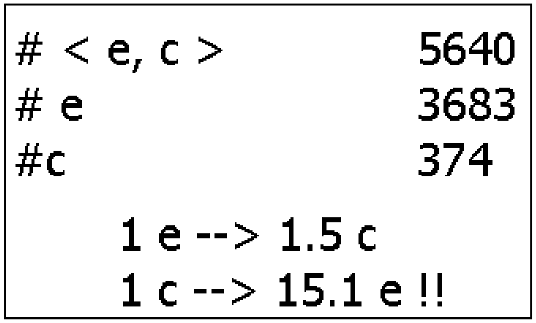
“Parallel” Corpus cntd

Phoneme	Example	Translation
CH	cheese	CH IY Z
D	dee	D IY
DH	thee	DH IY EH Ed EH D
ER	hurt	HH ER T
EY	ate	EY T
F	fee	F IY
G	green	G R IY N
HH	he	HH IY
IH	it	IH T
IY	eat	IY T
JH	gee	JH IY

A Statistical Machine Translation like task

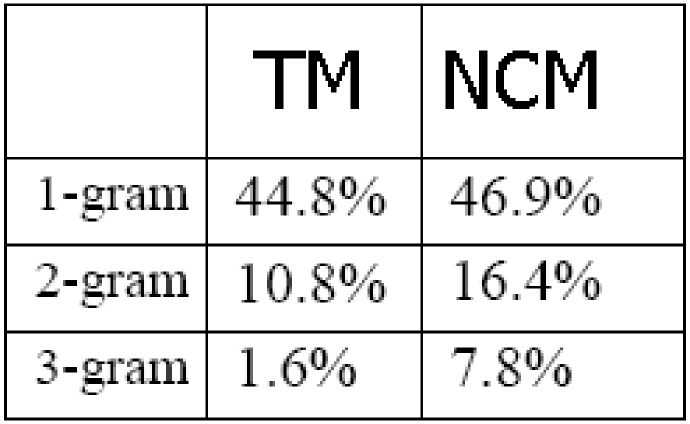
- First obtain the Carnegie Mellon University's Pronouncing Dictionary
- Train and Test the following Statistical Machine Learning Algorithms
- HMM - For HMM we can use either Natural Language Toolkit or you can use GIZA++ with MOSES

Evaluation



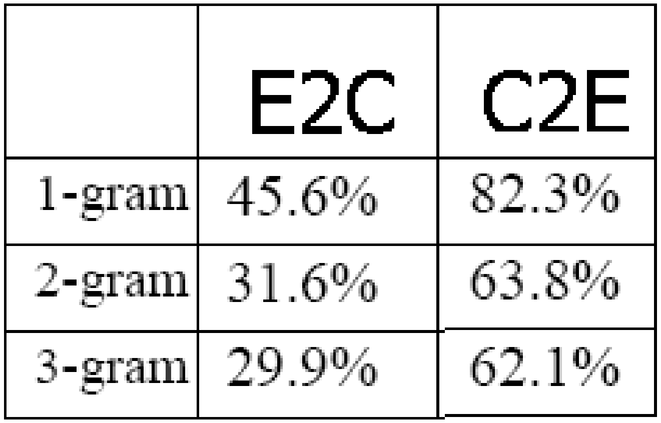
```
# < e, c >           5640
# e                  3683
#c                   374

     1 e --> 1.5 c
     1 c --> 15.1 e !!
```



	TM	NCM
1-gram	44.8%	46.9%
2-gram	10.8%	16.4%
3-gram	1.6%	7.8%

E2C Error rates for n-gram tests



	E2C	C2E
1-gram	45.6%	82.3%
2-gram	31.6%	63.8%
3-gram	29.9%	62.1%

E2C v/s C2E for TM Tests

Read up/look up/ study

- Google transliterator (routinely used; supervised by Anupama Dutt, ex-MTP student of CFILT)
- For all Devnagari transliterations, www.quillpad.in/hindi/

- **Phoneme and spelling-based models**

K. Knight and J. Graehl. 1998. Machine transliteration. *Computational Linguistics*, 24(4):599–612.

N. AbdulJaleel and L. S. Larkey. 2003. Statistical transliteration for English-Arabic cross language information retrieval. In *CIKM*, pages 139–146.

Y. Al-Onaizan and K. Knight. 2002. Machine transliteration of names in Arabic text. In *ACL Workshop on Comp. Approaches to Semitic Languages.*

- **Joint source-channel model**

H. Li,M. Zhang, and J. Su. 2004. A joint source-channel model for machine transliteration. In *ACL*, pages 159–166.

www.wikipedia.org