An Introduction to Machine Translation & Transliteration

Anoop Kunchukuttan

Research Scholar
Center for Indian Language Technology
www.cfilt.iitb.ac.in

Department of Computer Science & Engineering
IIT Bombay

anoopk@cse.iitb.ac.in

www.cse.iitb.ac.in/~anoopk
Machine Translation
What is Machine Translation?

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

e.g.
- Be the change you want to see in the world
- वह परिवर्तन बनो जो संसार में देखना चाहते हो
Why do we need machine translation?

- 4 language families
- 22 scheduled languages
- 11 languages with more than 25 million speakers
- 30 languages with more than 1 million speakers
- Only India has 2 languages in the world’s 10 most spoken languages
- 7-8 Indian languages in the top 20 most spoken languages
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 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 difficult?

Language Divergence: the great diversity among languages of the world
- Word order: SOV (Hindi), SVO (English), VSO, OSV,
- Free (Sanskrit) vs rigid (English) word order
- Analytic (Chinese) vs Polysynthetic (Finnish) languages
- Different ways of expressing same concept
- Case marking systems
- Language registers
- Inflectional systems [infixing (Arabic), fusional (Sanskrit), agglutinative (Marathi)]
... and much more
Why is machine translation difficult?

• Ambiguity
  – Same word, multiple meanings:
  – 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
Taxonomy of MT systems

MT Approaches

Knowledge Based; Rule Based MT
- Interlingua Based
- Transfer Based

Data driven; Machine Learning Based
- Example Based MT (EBMT)
- Statistical MT
Vauquois Triangle

Kinds of MT Systems
(point of entry from source to the target text)

- Deep understanding level
- Interlingual level
- Logico-semantic level
- Mixing levels
- Syntactico-functional level
- Syntagmatic level
- Morpho-syntactic level
- Graphemic level

- Ontological interlingua
- Semantico-linguistic interlingua
- SPA-structures (semantic & predicate-argument)
- Multilevel description
- F-structures (functional)
- C-structures (constituent)
- Tagged text
- Text

Conceptual transfer
Semantic transfer
Ascending transfer
Multilevel transfer
Syntactic transfer (deep)
Syntactic transfer (surface)
Semi-direct translation
Descending transfers
Direct translation
Statistical Machine Translation

Data driven translation
Parallel Corpus

<table>
<thead>
<tr>
<th>English</th>
<th>Hindi</th>
</tr>
</thead>
<tbody>
<tr>
<td>So far there is no evidence that there is a limit to the Universe.</td>
<td>ब्रम्हांड की कोई सीमा होने का अब तक कोई सबूत नहीं है।</td>
</tr>
<tr>
<td>The limit is rather on what we can see and how much we can understand.</td>
<td>सीमा बल्कि यही है कि हम क्या देख सकते हैं और हम कितना समझ पाते हैं।</td>
</tr>
</tbody>
</table>

The Noisy Channel Model

A very general framework for many NLP problems

Language Model (LM)
$P(e)$
Captures fluency

Translation Model (TM)
$P(ffe)$
Captures fidelity

Source $\rightarrow E \rightarrow$ Channel $\rightarrow F \rightarrow$ Destination
The SMT Process

**Training**

- Given: Parallel Corpus
- Output: $P(e)$, $P(f|e)$
  - This is model learning
- Learning Objective: Maximize Likelihood
- Offline, one-time process
- Different translation models from different choice of $P(f|e)$

$$P^*(f|e) = \arg \max \text{Likelihood}(data; P(f|e))$$

**Decoding**

- Given:
  - Sentence $f$ in language $F$
  - $P(e)$ and $P(f|e)$
- Output: Translation $e$ for $f$
- Online process, should be fast
- TM & LM are used for scoring translation candidates

$$e^* = \arg \max_e P(f|e)P(e)$$
Phrase-based Translation Model

- One of the most successful models
- Widely used in commercial systems like Google Translate
- Basic unit of translation is a phrase
- A *phrase* is just a sequence of words

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

<table>
<thead>
<tr>
<th>English</th>
<th>Hindi</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Prime Minister of India</td>
<td>भारत के प्रधान मंत्री</td>
</tr>
<tr>
<td></td>
<td>bharat ke pradhaan maMtrl</td>
</tr>
<tr>
<td></td>
<td>India of Prime Minister</td>
</tr>
</tbody>
</table>

- Sense disambiguation based on local context
  - Neighbouring words help do the right translation

<table>
<thead>
<tr>
<th>English</th>
<th>Hindi</th>
</tr>
</thead>
<tbody>
<tr>
<td>heads towards Pune</td>
<td>पुणे की ओर जा रहे है पुणे ki or jaa rahe hai</td>
</tr>
<tr>
<td></td>
<td>Pune towards go – continuous is</td>
</tr>
<tr>
<td>heads the committee</td>
<td>समिति की अध्यक्षता करते है समिति kii adhyakshata karte hai</td>
</tr>
<tr>
<td></td>
<td>committee of leading -verbalizer is</td>
</tr>
</tbody>
</table>
So how the model look now?

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

$$p(f_1^I|e_1^I) = \prod_{i=1}^{I} \phi(f_i|e_i) d(\text{start}_i - \text{end}_{i-1} - 1)$$

$\text{start}_i$: start position in $f$ of $i^{th}$ phrase of $e$

$\text{end}_i$: end position in $f$ of $i^{th}$ phrase of $e$
Training a Phrase-based SMT system

- Building the Language Model
- Building the Translation Model
  - Word Alignment (find word-level correspondences)
  - Phrase Extraction (extract phrase pairs)
- Tuning
Building the Language Model

- Probability of a sentence $e$
  - $P(e) = P(e_1, e_2, ..., e_k)$
  - $= \prod_{i=1..k} P(e_i | e_{i-1..i-n+1})$
  - Apply Chain Rule of probability
  - Markov Assumption: $i^{th}$ words depends only previous $n-1$ words ($n^{th}$ order Markov model)

- Estimate $P(e_i | e_{i-1..i-n+1})$ from a monolingual corpus
  e.g. of a bigram (2-gram) language model
  - $P(\text{book}|\text{the}) = \frac{c(\text{the,book})}{c(\text{the})}$
  - A little complication: what happens if book never comes in the training corpus
  - That's the complicated part of language modelling, let's skip it for now!
Word Alignment

- Central Task in Statistical Machine Translation
- Given a parallel sentence pair, find word level correspondences (*alignment, let's say a*)
But there are multiple possible alignments
But there are multiple possible alignments

How do we find the correct alignment?
Key ideas

- **Co-occurrence of words**
  - Words which occur together in the parallel sentence are likely to be translations \((higher P(f|e))\)
  - Alignments which have more likely word-translation pairs are more likely \((higher P(a))\)
  - Its a chicken-and-egg problem!
  - How to actually find the best alignment?

- **Expectation-Maximization Algorithm**
  - Find the best *hidden* alignment
  - A key algorithm for various machine learning problems
    - Start with a random alignment
    - Find \(P(f|e)\) given the alignments
    - Now compute alignment probabilities \(P(a)\) with these new translation probabilities
    - Do this repeatedly till \(P(f|e)\) does not change
At the end of the process
Learning Phrase Tables from Word Alignments

- Leverages word alignments learnt from IBM models
- 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

<table>
<thead>
<tr>
<th>प्रोफेसर</th>
<th>C.N.R. Rao</th>
<th>राव</th>
<th>को</th>
<th>भारतरत्न</th>
<th>से</th>
<th>सम्मानित</th>
<th>किया</th>
<th>गया</th>
</tr>
</thead>
<tbody>
<tr>
<td>सी.एन.आर</td>
<td>was honoured</td>
<td>with the</td>
<td>Bharat</td>
<td>Ratna</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Phrase Pairs “consistent” with word alignment

consistent

inconsistent

consistent

Source: SMT, Phillip Koehn
## Examples

<table>
<thead>
<tr>
<th>English</th>
<th>Hindi</th>
</tr>
</thead>
<tbody>
<tr>
<td>Professor CNR</td>
<td>प्रोफेसर सी.एन.आर</td>
</tr>
<tr>
<td>Professor CNR Rao</td>
<td>प्रोफेसर सी.एन.आर राव</td>
</tr>
<tr>
<td>Professor CNR Rao was</td>
<td>प्रोफेसर सी.एन.आर राव को</td>
</tr>
<tr>
<td>honoured with the Bharat Ratna</td>
<td>भारतरत्न से सम्मानित</td>
</tr>
<tr>
<td>honoured with the Bharat Ratna</td>
<td>भारतरत्न से सम्मानित किया</td>
</tr>
<tr>
<td>honoured with the Bharat Ratna</td>
<td>भारतरत्न से सम्मानित किया गया</td>
</tr>
<tr>
<td>honoured with the Bharat Ratna</td>
<td>को भारतरत्न से सम्मानित किया गया</td>
</tr>
</tbody>
</table>

26 phrase pairs can be extracted from this table.
Computing Phrase Translation Probabilities

- Estimated from the relative frequency:

\[ \phi(f|\bar{e}) = \frac{\text{count}(\bar{e}, f)}{\sum \text{count}(\bar{e}, f_i)} \]

<table>
<thead>
<tr>
<th>Phrase</th>
<th>Translation</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>Prime Minister of India</td>
<td>भारत के प्रधान मंत्री</td>
<td>0.75</td>
</tr>
<tr>
<td></td>
<td>India of Prime Minister</td>
<td></td>
</tr>
<tr>
<td>Prime Minister of India</td>
<td>भारत के भूतपूर्व प्रधान मंत्री</td>
<td>0.02</td>
</tr>
<tr>
<td></td>
<td>India of former Prime Minister</td>
<td></td>
</tr>
<tr>
<td>Prime Minister of India</td>
<td>प्रधान मंत्री</td>
<td>0.23</td>
</tr>
<tr>
<td></td>
<td>Prime Minister</td>
<td></td>
</tr>
</tbody>
</table>
Tuning

- Learning feature weights from data – $\lambda_i$
- 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
Overall Training Process for PB-SMT

Parallel training corpus → Word aligner E.g. GIZA++ → Word alignments → Parallel tuning corpus

Monolingual target corpus → Phrase pair extraction → Phrase tables → Distortion model learning → Distortion Model

Language model learner E.g. SRI, IRST → Distortion model learning → Language Model

Other Feature Extractors → Feature values → MERT Tuning

Parameter weights → Decoder
Decoding

- Find best translation among a very large number of possible translations
- NP-hard problem: 10-word sentence, 5 translations per word: $10^5 \cdot 10! \approx 362$ billion possible translations
- Look for approximate solutions
  - Restrict search space: *some word orders are not possible*
  - Incremental construction and scoring
  - Remove candidates that are unlikely to eventually generate good translations
Search Space and Search Organization

- Each hypothesis is scored using the SMT model
- Hypotheses are maintained in a priority queue (called stack decoding historically)
- Limit to the reordering window for efficiency
Richer Translation Models

- Syntax based SMT
- Factor based SMT
- Whole Document Translation
<table>
<thead>
<tr>
<th>English</th>
<th>Hindi</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jagdish Tytler is accused of leading a mob during the 1984 riots.</td>
<td>दिल्ली की एक अदालत ने हक़म दिया है कि कांग्रेस नेता और पूर्व मंत्री जगदीश टाइटलर के खिलाफ 1984 सिख विरोधी दंगा मामले में फिर से जांच शुरू की जाए.</td>
</tr>
<tr>
<td>The court has ordered the reopening of a case against this Congress Party leader for his involvement in anti-Sikh riots in 1984.</td>
<td>केंद्रीय जांच एजेंसी सीबीआई की सिफारिश पर दिल्ली की एक कोर्ट ने पहले जगदीश टाइटलर के खिलाफ मामले को बंद करने की इजाज़त दे दी थी.</td>
</tr>
<tr>
<td>Jagdish Tytler was originally cleared by the Central Bureau of Investigation (CBI).</td>
<td>दिल्ली से सांसद रह चुके जगदीश टाइटलर पर आरोप लगाते रहे हैं कि उन्होंने 1984 में लोगों को सिख विरोधी दंगों के दौड़ान भड़काया था.</td>
</tr>
</tbody>
</table>

The 1984 riots began following the assassination of Mrs Gandhi.

जगदीश टाइटलर कांग्रेस के तीन अहम नेताओं में से एक हैं जिनके खिलाफ सिख विरोधी दंगों को लेकर आरोप लगाते रहे हैं.
Speech-to-Speech Translation
Image Text to Image Text Translation

Translation on smaller devices
Some more Interesting Problems

• Translation among Related Languages

• Scaling to larger corpora

• Deep learning and Machine Translation
References

• Introductory textbooks

• Other introductory material
  - *Kevin Knight's MT workbook*
    www.isi.edu/natural-language/mt/wkbk.pdf
  - *ICON 2013 tutorial on Statistical Machine Translation*
References (2)

• Getting hands on
  – Moses
    http://www.statmt.org/moses/
  – Google Translate API
    https://cloud.google.com/translate/
  – Indic NLP Library
    https://github.com/anoopkunchukuttan/indic_nlp_library
  – IITB SMT tools for Indian languages
    http://www.cfilt.iitb.ac.in/static/download.html
Transliteration
You are in Kerala … waiting to travel by bus

Not a hypothetical situation …. Read this:
How do you translate Xi Jinping?

Xi Jinping is the President of China

शी चिनफिंग चीन के राष्ट्रपति है

Ok, we got lucky here … but there are so many names you will not find in any corpus
Transliteration can simplify Translation

<table>
<thead>
<tr>
<th>Hindi</th>
<th>Punjabi Translation</th>
</tr>
</thead>
<tbody>
<tr>
<td>यदि श्वास प्रणालिका में सूजन आ जाये तब भी रक्त मुँह के रास्ते बाहर आने लगता है।</td>
<td>निवल मात्र पूर्णस्व नीच में भा सच्चे उठ दी हुई मृत्यु दे उठमे चारव भारूढ़ कलात्मक ने। जेकर साह प्रणाली विच सोज आ जावे तद वी खून मूः ह दे रासते बाहर आउण लगदा है।</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Hindi-Punjabi Transliteration</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>असि मात्र पूर्णस्री में मृत्यु भा सच्चे उठ दी हुई मृत्यु दे उठमे चारव भारूढ़ कलात्मक ने। आदि साह प्रणाली में सूजन आ जावे तद वी रक्त मूः ह दे रासते बाहर आउण लगदा है।</td>
<td></td>
</tr>
</tbody>
</table>
Some Concepts

**Natural Language**: A system of communication among humans with sound

**Script**: A system of symbols for representing language in writing

- *A language can have multiple scripts:*
  - Sanskrit is written in many scripts (*Devanagari, Malayalam, Tamil, Telugu, Roman, etc.*)
- *A script can be used for multiple languages*
  - *Devanagari is used to write Sanskrit, Hindi, Marathi, Konkani, Nepali*

**Phoneme**: basic unit of sound in a language that is meaningful

**Grapheme**: basic distinct unit of a script

- *A phoneme can be represented by multiple graphemes*
  - *cut, dirt*
- *A grapheme can be used to represent multiple sounds*
  - *cut, put*
What is transliteration?

Transliteration is the conversion of a given name in the source language (from source script) to a name in the target language (target script), such that the target language name is:

- phonemically equivalent to the source name
  
  मुम्बई → Mumbai

- conforms to the phonology of the target language
  
  ਨਰੰਦਰ → ਰੰਦਰ (ਰੰਦਰ)

- matches the user intuition of the equivalent of the source language name in the target language, considering the culture and orthographic character usage in the target language
  
  ആലപ്പുഴ (aalappuzha) → Alappuzha
Isn't it easy to just map characters from one script to another?

- Local spelling conventions
  लता in Roman: Latha (South India) vs Lata (North India)
  Laxmi → लक्ष्मी

- Missing sounds
  കൊഴിക്കോട് (kozhikkoT) → कोषिक्कोड (koShikkod)

- Transliterate or translate
  കൊഴിക്കോട് (kozhikkoT) → Calicut

- Transliteration variants
  मुंबई, मुम्बई
Why English spellings caused trouble in school ...

Ambiguity in character to sound mapping

ionize vs nation

*fish* can be pronounced as *ghoti*

*gh* as in *tough*
*o* as in *women*
*ti* as in *nation*
... and Hindi spellings didn't

Unambiguous mapping from character to sound

Rememember the **varnamala**? – organized according to scientific principles

<table>
<thead>
<tr>
<th>Voicing →</th>
<th>sparśa (Plosive)</th>
<th>anunāsika (Nasal)</th>
<th>antastha (Approximant)</th>
<th>ūṣma/samghaśhri (Fricative)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>aghoṣa</td>
<td>ghōṣa</td>
<td>aghoṣa</td>
<td>ghōṣa</td>
</tr>
<tr>
<td>Aspiration →</td>
<td>alpaprāṇa</td>
<td>mahāprāṇa</td>
<td>alpaprāṇa</td>
<td>mahāprāṇa</td>
</tr>
<tr>
<td>kānthya (Guttural)</td>
<td>ka /k/</td>
<td>kha /kʰ/</td>
<td>ga /g/</td>
<td>gha /ɡʰ/</td>
</tr>
<tr>
<td>tālavya (Palatal)</td>
<td>ca /c, ɟ/</td>
<td>cha /cʰ, ɟʰ/</td>
<td>ja /j/</td>
<td>jha /ɟʰ/</td>
</tr>
<tr>
<td>mūrdhanya (Retroflex)</td>
<td>tā /t/</td>
<td>tha /tʰ/</td>
<td>da /d/</td>
<td>dha /dʰ/</td>
</tr>
<tr>
<td>dantya (Dental)</td>
<td>tā /t/</td>
<td>tha /tʰ/</td>
<td>da /d/</td>
<td>dha /dʰ/</td>
</tr>
<tr>
<td>oṣṭhya (Labial)</td>
<td>pa /p/</td>
<td>pha /pʰ/</td>
<td>ba /b/</td>
<td>bha /bʰ/</td>
</tr>
</tbody>
</table>
The extent of Devanagari-like scripts
How do we solve the transliteration problem?

- Transliteration is very similar to translation
- Instead of words, we have characters
- However, it is much simpler
  - No reordering
  - Small vocabulary (except Chinese and Japanese Kanji)
  - Regular grammar
- Similar to Vouquois triangle, you can transliterate at different levels:
  - Phoneme (like transfer based MT)
  - Grapheme (like direct MT)
References

• About Scripts
  – Omniglot: http://www.omniglot.com/
  – Wikipedia pages on Devanagari & Brahmi script

• About Transliteration

• Hands on
  – Google Transliterate
    http://www.google.com/inputtools/
  – Brahmi-Net: IITB's transliteration system
    http://www.cfilt.iitb.ac.in/brahminet/
Thank You!
Acknowledgments

• Some of slides & images have been borrowed from
  – Wikipedia