

# Hindi↔English Parallel Corpus Generation and Neural Machine Translation Ekansh Gupta and Rohit Gupta

CS671: Natural Language Processing, IIT Kanpur

# **Introduction**

Large collections of parallel texts are called **parallel corpora**. Alignments of parallel corpora at sentence level are prerequisite for many areas of linguistic research. During translation, sentences can be split, merged, deleted, inserted or reordered by the translator. This makes alignment a non-trivial task.

A comparable corpus is built from non-sentence-aligned and untranslated bilingual documents, but the documents are topic-aligned.

Quite a few machine translators use SMT (Statistical Machine Translation). Neural machine translation is a new approach to machine translation, where we train a single, large neural network to maximize the translation performance. This is a radical departure from existing (phrase-based) machine translation approaches, where a translation system consists of many subcomponents which are optimized separately

Here we use RNN to train a weak translator which then is used to generate parallel corpus from comparable corpus

## **Motivation**

Advantages of Neural Machine Translation

- Require only a fraction of the memory needed by traditional statistical machine translation (SMT) models
- Deep Neural Nets out-perform previous state of the art methods assuming availability of large parallel corpora Can be combined with word-alignment approach to address the rare-word problem

### **Pre-Processing**

The words in each sentence were encoded using one-hot encoding for top-N words chosen from large monolingual corpora for each language (N=5000). Out of Vocabulary (OOV) words represented by *<unk>*. Monolingual corpora used for choosing vocabulary range was obtained from HelioHost.

#### Sentence Alginment

We built an heuristic for aligning sentences between native English and weakly translated English, it used the token sort ratio metric (number of similar words).

#### RNN Encoder ↔ Decoder Model

The Keras Library was used to implement the RNN encoder-decoder translation model. The model encoded each input sentence (variable length) into a fixed-sized vector of length 512. The vocabulary size used was 5000 for both Hindi and English.

The translator was trained using 25000 sentence pairs on a GPU system with 4GB memory. The train-test split was 10%-90%.

## Results



The steadily increasing training accuracy but low validation accuracy is a result of over-fitting due to a relatively small set of sentence pairs which was limited by the GPU memory and time. As a result, to generate parallel corpus, a python library, TextBlob, which is a weak alternative for translators was used.

Very large parallel English-Hindi corpora are unavailable

However comparable corpora available

## Theory

**Recurrent Neural Networks** 

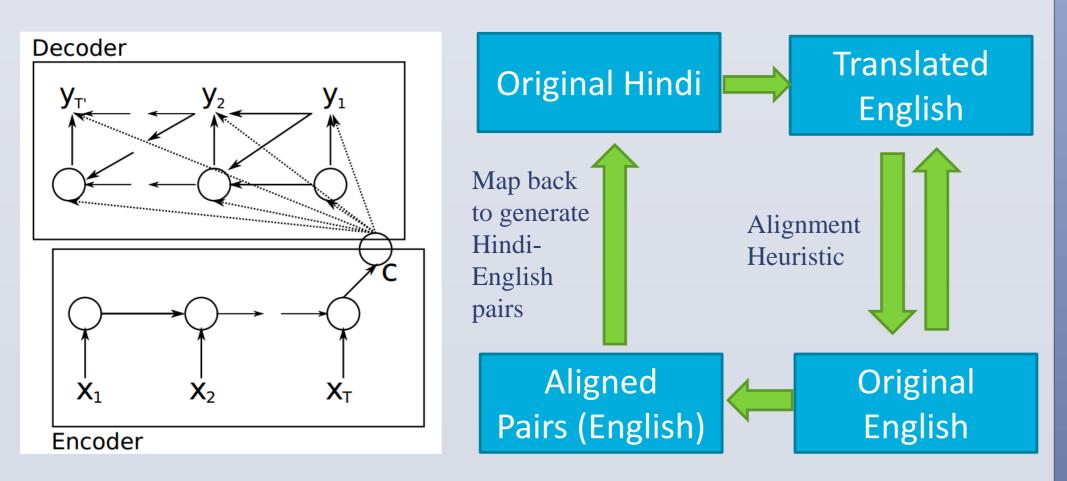
A standard RNN maps a sequence of inputs to outputs by iterating the following equations:

- $h_t = \sigma(W^{hx}x_t + W^{hh}h_{t-1})$
- $y_t = W^{yh}h_t$

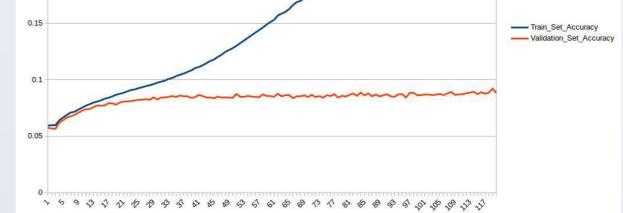
LSTM:

- $p(y_1, \dots, y_{T'} | x_1, \dots, x_t) = \prod_{t=1}^{T'} p(y_t | v, y_1, \dots, y_{t-1})$ 
  - Distribution is represented with a softmax over all the words in the vocabulary

We use a recurrent neural network architecture that learns to encode a variable-length sequence into a fixed-length vector representation and to decode a given fixed-length vector representation back into a variable-length sequence. From a probabilistic perspective, this new model is a general method to learn the conditional distribution over a variable-length sequence conditioned on yet another variable-length sequence



Left (Representation of the RNN Encoder-Decoder model Right (The Parallel corpus alignment schematic)



Some examples from the generated parallel corpus:

Good:

• अधिकांश लोग समय के साथ लगभग 90% कंधे की गति पुन प्राप्त करते हैं

Most people regain about 90% of shoulder motion over time

- कंपनी ऑफ हीरोज रेलिक का पहला शीर्षक था जिसमें एसेंस इंजन का उपयोग किया गया Company of Heroes is Relics first title to make use of the Essence Engine Average:
- सिल्वीकल्चर गार्डन में रबड़ और बांस के पेड़ हैं

The silviculture garden has rubber trees and human thick bamboo plants Bad:

+ 7.9%

<u>+</u> 6.7%

5.09%

• इनमें शामिल हैं प्राचीन ग्रीस की एंटिकिथेरा प्रक्रिया और एस्ट्रॉलैब जिन्हें आम तौर पर सबसे प्रारंभिक ज्ञात यांत्रिक एनालॉग कंप्यूटर माना जाता है

The Antikythera mechanism is believed to be the earliest mechanical analog computer according to Derek J

To test the quality of generated corpus, we sampled 100 generated pairs and got them rated by 4 different human evaluators. The results (within 95% confidence interval) are as follows:

Perfect:	77% _
Acceptable:	15% -
Rejected:	8% <u>+</u>

# References

- Kalchbrenner and Blunsom, Recurrent Continuous Translation Models, 2013 1.
- Sutskever et al, Sequence to Sequence Learning with Neural Networks, 2014 2.
- Cho et al, On the Properties of Neural Machine Translation: Encoder–Decoder 3. Approaches, 2014
- 4. Hermann and Blunsom, Multilingual Distributed Representations without Word Alignment, 2014

#### Summary:

- Training weak translator using limited parallel corpus
- Weak translator and aligning heuristic (ex: *Hunalign*) used to create additional parallel corpus
- Neural translator re-trained on generated bigger parallel corpus •



A schematic of the summary

## **Implementation Details**

#### Scraping

Wikipedia was used as a resource for collection of comparable corpora. Scraping was done using the MediaWiki API to fetch Hindi and English articles and a python library, BeautifulSoup4 was used to extract meaningful content from the HTTP response

- 5. Cho et al, Neural Machine Translation by jointly learning to align and translate, **ICLR 2015**
- Wolk and Marasek, Building subject aligned comparable corpora and mining it 6. for truly parallel sentence pairs, 2014
- Heliohost: http://corpora.heliohost.org/

MediaWiki API: <u>https://www.mediawiki.org/wiki/API:Main\_page</u>

BeautifulSoup4: http://www.crummy.com/software/BeautifulSoup/

Hindi-English Parallel Corpus: https://lindat.mff.cuni.cz/repository/xmlui/handle/11858/00-097C-0000-0001-BD17-1

Keras Deep Learning Library: http://keras.io/

### **Acknowledgements**

We would like to thank Prof.Amitabha Mukherjee and other fellow students pursuing CS671 for fruitful discussions and guidance towards completing the project.

**RESEARCH POSTER PRESENTATION DESIGN © 201** www.PosterPresentations.con