Inferring Translation Candidates for Multilingual Dictionary Generation with Multi-Way Neural Machine Translation

Mihael Arcan, Daniel Torregrosa*, Sina Ahmadi* and John P. McCrae
Introduction

Neural machine translation

Results

Dictionary data

Conclusion
Motivation

• Knowledge bases are useful for many applications, but available in few languages

• The creation and curation of knowledge bases is expensive

• Hence, few or no knowledge bases in most languages

• Can we use machine translation to translate knowledge?
Overview

- Multi-way neural machine translation without the targeted direction
- Continuous training with a small curated dictionary
- Discovery of new bilingual dictionary entries
Targeted languages

- PT
- GL
- RO
- IT
- CA
- ES
- EU
- EN
- EO

Languages:
- ES
- RO
- CA
- FR
- IT
- GL
- PT
- EO
- EU
- EN
Introduction

Neural machine translation

Results

Dictionary data

Conclusion
Machine translation before 2014

- Rule-based machine translation
  - Humans write rules
  - Highly customisable
  - High maintenance cost

- Phrase-based statistical machine translation
  - Learns from parallel corpus
  - Less control on the translations
Word embeddings

• Fixed size numerical representation for words

• From one-hot space (one dimension per different word) to embedding space

• The embedding vector represents the context where the word appears
Long-short term memory

Based on tex.stackexchange.com/questions/332747/how-to-draw-a-diagram-of-long-short-term-memory
Bi-directional LSTM

Based on github.com/PetarV-/TikZ
Neural machine translation

Encoder

Attention

Decoder

My

house

is

red
Neural machine translation

Encoder  Attention  Decoder

My
house
is
red
Neural machine translation

Encoder

My

Attention

house

Decoder

is

red
Neural machine translation
Neural machine translation
Neural machine translation
Neural machine translation

Encoder

Attention

Decoder

My

house

is

red

Mi

casa
Neural machine translation
Neural machine translation
Subword units

- One-hot vocabulary space has to be limited due to performance issues
- This generates a lot of out-of-vocabulary entries
- To minimize the effect, we use subword units instead of words
Byte pair encoding

- BPE is a compression technique
- It starts with all the different characters in the corpus
- The most frequent character combination is selected as a BPE operation
- This is repeated until the desired number of BPE is reached
- The final size of the vocabulary is the number of BPE operations + the alphabet
Byte pair encoding example

low lower big bigger
Byte pair encoding example

low_lower_big_bigger
Byte pair encoding example

low_lower_big_bigger
Byte pair encoding example

low lower big bigger
Byte pair encoding example

low_lower_bigger
# Byte pair encoding II

<table>
<thead>
<tr>
<th>Tense</th>
<th>Present</th>
<th>Conditional</th>
<th>Future</th>
<th>Preterit</th>
<th>Imperfect</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>bebo</td>
<td>bebemos</td>
<td>bebería</td>
<td>bebí</td>
<td>bebía</td>
</tr>
<tr>
<td></td>
<td>bebes</td>
<td>bebéís</td>
<td>beberías</td>
<td>bebiste</td>
<td>bebías</td>
</tr>
<tr>
<td></td>
<td>bebe</td>
<td>beben</td>
<td>bebería</td>
<td>bebió</td>
<td>bebía</td>
</tr>
<tr>
<td></td>
<td>bebéís</td>
<td>bebíamos</td>
<td>beberéis</td>
<td>bebisteis</td>
<td>bebíasis</td>
</tr>
<tr>
<td></td>
<td>bebíamos</td>
<td>beberán</td>
<td>beberán</td>
<td>bebieron</td>
<td>bebían</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tense</td>
<td>Singular</td>
<td>Plural</td>
<td>Conditional</td>
<td>Future</td>
<td></td>
</tr>
<tr>
<td>--------------</td>
<td>----------------</td>
<td>--------------</td>
<td>-------------</td>
<td>----------------</td>
<td></td>
</tr>
<tr>
<td><strong>Present</strong></td>
<td>bebo</td>
<td>bebemos</td>
<td>bebéis</td>
<td>bebería</td>
<td></td>
</tr>
<tr>
<td></td>
<td>bebes</td>
<td>bebéis</td>
<td>beberías</td>
<td>beberías</td>
<td></td>
</tr>
<tr>
<td></td>
<td>bebe</td>
<td>beben</td>
<td>bebería</td>
<td>bebería</td>
<td></td>
</tr>
<tr>
<td><strong>Preterit</strong></td>
<td>bebí</td>
<td>bebimos</td>
<td>bebistéis</td>
<td>beberé</td>
<td></td>
</tr>
<tr>
<td></td>
<td>bebiste</td>
<td>bebistéis</td>
<td>beberás</td>
<td>beberás</td>
<td></td>
</tr>
<tr>
<td></td>
<td>bebió</td>
<td>bebieron</td>
<td>beberá</td>
<td>beberá</td>
<td></td>
</tr>
<tr>
<td><strong>Imperfect</strong></td>
<td>bebía</td>
<td>bebíamos</td>
<td>bebíais</td>
<td>beberemos</td>
<td></td>
</tr>
<tr>
<td></td>
<td>bebías</td>
<td>bebíais</td>
<td>beberéis</td>
<td>beberemos</td>
<td></td>
</tr>
<tr>
<td></td>
<td>bebía</td>
<td>bebían</td>
<td>beberán</td>
<td>beberán</td>
<td></td>
</tr>
</tbody>
</table>
Multi-way model

• The model receives corpus in several different languages both for source and target sentences

• Each input sentence is annotated with the source language and the requested target language

• In our case, Spanish-English, French-Romanian and Italian-Portuguese
Continuous training

• After training, the network is seldom able to produce text in the requested language other than the training one

• For example, if requested to translate Spanish to French, it will generate English

• We continue the training with a small corpus of sentences
Dictionary data

We used three different dictionaries to continue training the system

• Spanish to French Apertium dictionary (paper)

• Spanish-French, Spanish-Portuguese and French-Portuguese dictionaries generated from Apertium data (task)
  • By following a cycle-based approach
  • By following a path-based approach
Part of speech

• The NMT models were trained without part of speech (POS) data

• To assign POS, we use monolingual dictionaries automatically extracted from Wiktionary

• If
  > the source word is in the source-language dictionary; and
  > the target word is in the target-language dictionary; and
  > they have one or more POS tags in common,

• generate one entry per shared POS
Introduction

Neural machine translation

Results

Dictionary data

Conclusion
Evaluation

• We used a dictionary automatically extracted from Wiktionary as gold standard

• For those systems that have confidence intervals, we calculate the precision and recall for all possible thresholds
Results (paper)

The diagram shows the precision of correct entries for translation between Spanish to French and French to Spanish, comparing different methods:

- **Apertium**
- **NMT+Apertium**
- **NMT+Apertium$_1$**
- **NMT+Apertium$_{10}$**

The x-axis represents the number of correct entries, and the y-axis represents precision.

The plot indicates that the NMT+Apertium$_{10}$ method generally has the highest precision for both translation directions, while Apertium tends to have the lowest precision.
Introduction

Neural machine translation

Results

Dictionary data

Conclusion
Graph-based approaches

**Basic idea:** Retrieve translations based on the graph of languages

Two definitions:

- Language graph refers to the Apertium dictionary graph
- Translation graph refers to a graph where vertices represent a word and edges represent the translations in other languages.
Cycle-based approach

Apertium translations (black lines) in English (EN), French (FR), Basque (EU) and Esperanto (EO), and discovered possible translations (gray lines) and synonyms (red lines).
Path-based approach

Traverse all simple paths using **pivot-oriented inference**

(Task) Weight translations w.r.t. frequency and path length
Results (task, Wiktionary reference)

- **English→French**
- **French→English**
- **English→Portuguese**
- **Portuguese→English**
- **Portuguese→French**
- **French→Portuguese**

Each graph shows the precision as the number of correct entries increases across different cycles and paths. The graphs compare the performance of NMT-Cycle and NMT-Path.
Introduction

Neural machine translation

Results

Dictionary data

Conclusion
Conclusion

• Using neural machine translation with
  • Existing bilingual knowledge (Paper)
  
  • Discovered bilingual knowledge (Task)

• to generate new dictionaries.