The Low Resource NLP Toolbox, 2020 Version

Graham Neubig
@ AfricaNLP 4/26/2020

(collaborators highlighted throughout)
How do We Build NLP Systems?

- **Rule-based systems**: Work OK, but require lots of human effort for each language for where they're developed

- **Machine learning based systems**: Work really well when lots of data available, not at all in low-data scenarios
The Long Tail of Data

Articles in Wikipedia vs. Language Rank
Machine Learning Models

- Formally, map an input $X$ into an output $Y$. Examples:

<table>
<thead>
<tr>
<th>Input X</th>
<th>Output Y</th>
<th>Task</th>
</tr>
</thead>
<tbody>
<tr>
<td>Text</td>
<td>Text in Other Language</td>
<td>Translation</td>
</tr>
<tr>
<td>Text</td>
<td>Response</td>
<td>Dialog</td>
</tr>
<tr>
<td>Speech</td>
<td>Transcript</td>
<td>Speech Recognition</td>
</tr>
<tr>
<td>Text</td>
<td>Linguistic Structure</td>
<td>Language Analysis</td>
</tr>
</tbody>
</table>

- To learn, we can use
  - Paired data $<X, Y>$, source data $X$, target data $Y$
  - Paired/source/target data in *similar* languages
Method of Choice for Modeling: Sequence-to-sequence with Attention

- **Various tasks**: Translation, speech recognition, dialog, summarization, language analysis
- **Various models**: LSTM, transformer
- Generally trained using **supervised learning**: maximize likelihood of \( <X,Y> \)

The Low-resource NLP Toolbox

- In cases when we have lots of paired data \( <X, Y> \)
  -> **supervised learning**

- But what if we don't?!

- Lots of source or target data \( X \) or \( Y \)
  -> **monolingual pre-training, back-translation**

- Paired data in another, similar language \( <X', Y> \) or \( <X, Y'> \)
  -> **multilingual training, transfer**

- Can ask speakers to do a little work to generate data
  -> **active learning**
Learning from Monolingual Data
Language-model Pre-training

- Given source or target data $X$ or $Y$, train just the encoder or decoder as a language model first

- Many different methods: simple language model, BERT, etc.

Sequence-to-sequence Pre-training

- Given just source, or just target data $X$ or $Y$, train the encoder and decoder together


Back Translation

- Translate target data $Y$ into $X$ using a target-to-source translation system, then use translated data to train source-to-target system

Iterative back-translation: train src-to-trg, trg-to-src, src-to-trg etc

Semi-supervised translation: many iterations of iterative translation, weighting confident instances

Multilingual Learning, Cross-lingual Transfer
Multilingual Training [Johnson+17, Ha+17]

- Train a large multi-lingual NLP system


Massively Multilingual Systems

- Can train on 100, or even 1000 languages (e.g. Multilingual BERT, XLM-R)
- Hard to balance multilingual performance, careful data sampling necessary

**Multi-DDS:** Data sampling can be *learned automatically* to maximize accuracy on all languages

XTREME: Benchmark for Multilingual Learning
[Hu, Ruder+ 2020]

- Difficult to examine performance of systems on many different languages
- XTREME benchmark makes it easy to evaluate on existing datasets over 40 languages
- Some coverage of African languages -- Afrikaans, Swahili, Yoruba

Cross-lingual Transfer

• Train on one language, transfer to another

• Train on many languages, transfer to another

Challenges in Multilingual Transfer
Problem: Transfer Fails for Distant Languages

How can We Transfer Across Languages Effectively?

• Select similar languages, add to training data.
• Model lexical/script differences
• Model syntactic differences
Which Languages to Use for Transfer?

• Similar languages are better for transfer when possible!
• But when want to transfer, what language do we transfer from? (various factors: language similarity, available data, etc.)
• **LangRank**: Automatically choose transfer languages data, language similarity features

Problems w/ Word Sharing in Cross-lingual Learning

- Spelling variations (esp. in subword models)
- Script differences / morphology (conjugation) differences

<table>
<thead>
<tr>
<th>Units</th>
<th>Turkish</th>
<th>Uyghur</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Graphemes</strong></td>
<td>&lt;yetmiyor&gt; it is not enough</td>
<td>دەگەر کە نەدێد</td>
</tr>
<tr>
<td><strong>Phonemes</strong></td>
<td>/qarijalmajdou/</td>
<td>/jetmijor/</td>
</tr>
<tr>
<td><strong>Morphemes</strong></td>
<td>/qari-jal-ma-jdu/</td>
<td>/jet-mi-jor/</td>
</tr>
<tr>
<td><strong>Conjugations</strong></td>
<td>qari + Verb + Pot + Neg + Pres + A3sg</td>
<td>jet + Verb + Neg + Prog1 + A3sg</td>
</tr>
</tbody>
</table>
Better Cross-lingual Models of Words

[Wang+19]

- A method for word encoding particularly suited for cross-lingual transfer

![Diagram showing word encoding process]

- Handles spelling similarity
- Handles consistent variations b/t languages
- Attempts to capture latent "concepts"

On MT for four low-resource languages, we find that:
- SDE is better than other options such as character n-grams
- SDE improves significantly over subword-based methods (e.g. used in multilingual BERT)

A skilled linguist can create a "reasonable" morphological analyzer and transliterator for a new language in short order.

Our method: represent words by bag of
- phoneme n-grams
- lemma
- morphological tags

/jetmijɔr/  jet + Verb + Neg + Prog1 + A3sg

Good results on NER/MT for Turkish->Uyghur, Hindi->Bengali transfer.

Data Augmentation via Reordering

[Zhou+ 2019]

- **Problem:** Source-target word order can differ significantly in methods that use monolingual pre-training

- **Solution:** Do re-ordering according to grammatical rules, followed by word-by-word translation to create pseudo-parallel data

Pivoting Methods

- Tons of data in English, fair amount of data in a relatively high-resourced language (HRL) and want to process a low-resourced language (LRL)
- Pivoting through HRL can take advantage of available resources!

**Zero-shot entity linking** by pivoting through related language w/ phonetic representations [Rijhwani+19]

**Data augmentation for NMT** using related language and unsupervised lexicon induction [Xia+19]
Active Learning
Creating Data

- Cross-lingual transfer is great, but no substitute for actual annotated data!

- **Active learning:** Ask human annotators to create data that maximally improves performance

- **What level of annotation?:**
  - *Sentence level* -- select hard-looking sentences
  - *Phrase-level* -- select hard-looking phrases

- **What criterion for selection?:**
  - *Uncertainty* -- phrases/sentences that look hard for the current model
  - *Representativeness* -- how well does it cover examples in the data?
• Phrase-level annotation

• Select phrases that are infrequent in parallel data (uncertain), but frequent in monolingual data (representative)

Active Learning+Cross-lingual Transfer

[Chaudhary+ 19]

• Train a cross-lingual model, gradually improve via monolingual annotation

• Select examples where the cross-lingual model has uncertain predictions

• Using both cross-lingual and active supervision improves significantly over using just one

Conclusion
The Low-resource NLP Toolbox

• Lots of paired data $<X, Y>$
  -> supervised learning
• Lots of source or target data $X$ or $Y$
  -> monolingual pre-training, back-translation
• Paired data in another, similar language $<X', Y>$ or $<X, Y'>$
  -> multilingual training, transfer
• Can ask speakers to do a little work to generate data
  -> active learning

Use any tool available to you!

Thank you to sponsors:
(views expressed here do not reflect views of the US government)