School of Computing and Information Systems
The University of Melbourne
COMP90042 NATURAL LANGUAGE PROCESSING (Semester 1, 2020)

Workshop exercises: Week 10

Discussion

1. What aspects of human language make automatic translation difficult?
2. What is **Information Extraction**? What might the "extracted" information look like?
   (a) What is **Named Entity Recognition** and why is it difficult? What might make it more difficult for persons rather than places, and *vice versa*?
   (b) What is the **IOB** trick, in a sequence labelling context? Why is it important?
   (c) What is **Relation Extraction**? How is it similar to NER, and how is it different?
   (d) Why are hand–written patterns generally inadequate for IE, and what other approaches can we take?

Programming

1. In the iPython notebook `11-machine-translation`, we provide an example of an encoder-decoder model for machine translation. Just like workshop-07, we'll use colab to run the notebook, as we'll need a GPU to speed up the training. Please refer to workshop-07 if you are looking for instructions to get set up for colab.
   - Modify the code to use GRU instead of LSTM, noting that GRU does not have a memory state (`state_c`).
   - Modify the code to do translation at the word-level instead of character-level. This will involve:
     – Using a French and English tokeniser (e.g. spaCy);
     – Creating a vocabulary for the input source and target language;
     – Replacing low frequency words with a special UNK token;
     – Changing the corpus reading function to create sequences of words for the training data;
     – Updating the training model and inference model to use the word vocabulary to look up words.

Get ahead

- Extend the encoder-decoder model in `11-machine-translation` to incorporate attention mechanism. You can use the formulation described in the lecture: dot product for comparing encoder and decoder hidden states; and concatenating the "context vector" with the decoder hidden state to predict the target word.