EXPERIMENT NO – 7

DIFFERENT MACHINE TRANSLATION TOOLS

MOSES:

1. *Moses is a statistical machine translation system that allows you to automatically train translation models for any language pair.*
2. *All you need is a collection of translated texts (parallel corpus).*
3. *Once you have a trained model, an efficient search algorithm quickly finds the highest probability translation among the exponential number of choices.*
4. *The training process in Moses takes in the parallel data and uses co-occurrences of words and segments (known as phrases) to infer translation correspondences between the two languages of interest.*
5. *In phrase-based machine translation, these correspondences are simply between continuous sequences of words, whereas in hierarchical phrase-based machine translation or syntax-based translation, more structure is added to the correspondences.*
6. *For instance a hierarchical MT system could learn that the German hat X gegessen corresponds to the English ate X, where the Xs are replaced by any German-English word pair.*
7. *The extra structure used in these types of systems may or may not be derived from a linguistic analysis of the parallel data.*
8. *Moses also implements an extension of phrase-based machine translation known as factored translation which enables extra linguistic information to be added to a phrase-based system.*
9. *The two main components in Moses are the training pipeline and the decoder. There are also a variety of contributed tools and utilities.*

10. *The training pipeline is really a collection of tools (mainly written in perl, with some in C++) which take the raw data (parallel and monolingual) and turn it into a machine translation model.*
11. *The decoder is a single C++ application which, given a trained machine translation model and a source sentence, will translate the source sentence into the target language.*

OpenNMT:

1. **OpenNMT** is an open source ecosystem for neural machine translation and neural sequence learning.
2. Neural machine translation (NMT) is a new methodology for machine translation that has led to remarkable improvements, particularly in terms of human evaluation, compared to rule-based and statistical machine translation (SMT) systems .
3. Originally developed using pure sequence-to-sequence models and improved upon using attention-based variants , NMT has now become a widely-applied technique for machine translation, as well as an effective approach for other related NLP tasks such as dialogue, parsing, and summarization.
4. As NMT approaches are standardized, it becomes more important for the machine translation and NLP community to develop open implementations for researchers to benchmark against, learn from, and extend upon.
5. With these goals in mind, we introduce OpenNMT, an open source framework for neural machine translation. OpenNMT is a complete NMT implementation. In addition to providing code for the core translation tasks.
6. In practice, there are also many other important aspects that improve the effectiveness of the base model.
7. Here we mention four areas:
    a. It is important to use a gated RNN such as an LSTM which help the model learn long-distance features within a text.
    b. Translation requires relatively large, stacked RNNs, which consist of several vertical layers of RNNs at each time step

c. Input feeding, where the previous attention vector is fed back into the input as well as the predicted word, has been shown to be quite helpful for machine translation
d. Test-time decoding is done through beam search where multiple hypothesis target predictions are considered at each time step. Implementing these correctly can be difficult, which motivates their inclusion in an NMT framework.

8. One nice aspect of NMT as a model is its relative compactness. When excluding Torch framework code, the Lua OpenNMT system including preprocessing is roughly 4K lines of code, and the Python version is less than 1K lines (although slightly less feature complete).
9. For comparison the Moses SMT framework including language modeling is over 100K lines. This makes the system easy to completely understand for newcomers.
10. The project is fully self-contained depending on a minimal number of external Lua libraries and including also a simple language independent reversible tokenization and detokenization tools.

Sockeye:

1. SOCKEYE is a versatile toolkit for research in the fast-moving field of NMT. Since the initial release, it has been used in at least 25 scientific publications, including winning submissions to WMT evaluations .
2. Based on the deep learning library MXNet , SOCKEYE also powers Amazon Translate, showing industrial strength performance in addition to the flexibility needed in academic environments.
3. Moreover, we are excited to see hardware manufacturers contributing optimizations to MXNet and SOCKEYE. Intel has demonstrated large performance gains for SOCKEYE inference on Intel Skylake processors.
4. SOCKEYE 2 significantly accelerates training with Horovod2 integration and MXNet's automatic mixed precision (AMP).
5. Horovod extends synchronous training to any number of GPUs (including across nodes) while AMP automatically detects and converts parts of the model that can run in reduced-precision mode (FP16) without loss of quality.

6. SOCKEYE also provides a data-driven alternative to the popular “inverse square root” learning schedule.
7. Keeps the same learning rate until validation perplexity does not increase for several checkpoints, at which time it reduces the learning rate and rewinds all model and optimizer parameters to the best previous point. Training concludes when validation perplexity reaches an extended plateau.
8. SOCKEYE 2 provides out-of-the-box support for quickly training strong transformer models for research or production.
9. Extensive configuration options and the simplified code base enable rapid development and experimentation.
10. We invite the community to contribute their ideas to SOCKEYE 2 and hope that the new programming model and performance improvements enable others to conduct effective and successful research.