A Toolset to Integrate OpenNMT into Production Workflow

Yu Gong
Product Globalization
VMware
Beijing, China
gongy@vmware.com

Demin Yan
Product Globalization
VMware
Palo Alto, USA
dyan@vmware.com

Abstract

In recent months, machine translation (MT) using deep learning has attracted attention for its improved quality over statistical MT. Harvard University and Systran introduced an open-source tool, OpenNMT, to the public for training neural machine translation models. OpenNMT is easy to use yet, there are still some limitations when applying it into an enterprise production environment.

In most enterprise production environments, output from the localization workflow is in Translation Memory eXchange (TMX) format. To feed this kind of human-translated parallel data into OpenNMT, users have to write their own tools or make use of some third-party tools to manipulate the data.

To quickly set up a workable machine translation engine with less cost and effort, we developed a toolset, called OMTS (OpenNMT Toolset) [1], to accelerate the process. OMTS contains two major features:

- TMX parsing and corpus cleaning;
- OpenNMT model training and controlling;
- RESTful APIs to call an OpenNMT model.

In the beginning, OMTS uses TMX file(s) as input, and then calls the corpus cleaning tool in m4loc (Moses for Localization)[2] to generate clean and tokenized corpus required by the pre-processing step in OpenNMT. A training job is automatically kicked off right after the corpus is ready to generate the final model.

OMTS evaluates the results by giving it a BLEU score. A dashboard gives the users a sense of how good the model is. Users also have an option to let OMTS automatically choose the best model (with the highest BLEU score). Finally, to integrate the model into localization workflow, a connector is required to link the model to the production environment. This connector is usually done by the localization management system (e.g. SDL WorldServer) provider and currently not in the scope of OMTS.

In conclusion, OMTS streamlines the process of creating workable NMT models by making use of the enterprise's own raw data and integrating it into the current localization workflow. With minimal effort, users are then able to set up their own OpenNMT systems.

[1] We're intended to get OMTS open source and it's currently in internal review process.
[2] https://github.com/achimr/m4loc