LibreOffice Translate

Offline AI-powered Machine Translation with a single Click

Built with: LibreOffice, PyTorch, OpenNMT

https://github.com/lernapparat/lotranslate/
Neural Machine Translation for LibreOffice

Thomas Viehmann

tv@lernapparat.de

ALMERIA | 13 Sept. 2019
• About me
• LibreOffice Translate – An Extension for Writer
• How much UI do we need?
• Under the hood
  • Neural Machine Translation
  • NMT models meet the real world
  • PyTorch & OpenNMT
  • How to train new models
  • Pains of a LibreOffice Python extension with many dependencies
• Better Integration with Writer / other parts LibreOffice
Hi, I’m Thomas Viehmann

My background:

- did some Pattern Recognition and Neural Networks in 2000, but moved on to mathematics
- Ph.D. in Mathematics, modelling patterns in magnets
- 9 years of insurance risk modelling
- Recently lots of modelling in the ML context
- 2 tiny LibreOffice patches

ML Training & Consultancy through: http://mathinf.eu/

Blog about ML stuff: http://lernapparat.de/
LibreOffice Translate

Idea: very simple (1-click) machine translation of text in Writer

https://github.com/lernapparat/lotranslate/

Development supported by the German Ministry of Research and Technology through the Prototype Fund. (Thank you!)
How much UI do we need?

Researched other Translation tools workflow: “Text→Sidebar→Translation → Edit → Insert”

What is the advantage of using the sidebar? → You see original and translation side by side.

(Happily, the Prototype Fund sponsored a coaching regarding UI / UX which helped in reflecting this.)
How much UI do we need?

LOTranslate use Annotations instead (also saves UI coding):

...annotations are optional
How much UI do we need?

There are interesting visualizations of model “innards”...

... but are they useful here without intervention options?

(Source: Strobelt et al.: Seq2Seq-Vis)
Oversimplified structure of NMT network

- Red arrows: Predictions. Attention “causal” and also in Encoder.
- Similar to the ones used by well-known online services, to recent news-making models (BERT, OpenAI GPT2,...)
- Trained through back-propagation of error term.
NMT models meet the real world

- Need to translate more than one sentence → need to split
- Want to preserve formatting
  → Use Attention to map
    (heuristics for “end-bias”)
  → Paragraph formatting?
    (currently clumsy bit-by-bit insertion)
PyTorch

- great Python Deep Learning framework,
- quite well-liked in deep learning research,
- large and very nice community

(I’m biased, though.)

OpenNMT

- Quite comprehensive framework for Neural Machine Translation and related tasks
- More library / application like than many other “code for the paper” style implementation.
Training new models

- Needs parallel corpus of sentences (no word alignment needed)
  EN-DE: 4.5 Million pairs
- Needs (at least one) “gaming” GPU – for 1-2 weeks
  (that is 30-70 KWh per model – compare to ~2.400 KWh per year for a family of 5 - and the GPU)
- Vocabulary preparation
- Training (this is what takes long)
- Evaluation (mostly “closeness” on a holdout set + inspection)
- Probably also want domain adaptation (i.e. specialize from “general” model to one specific for a domain, e.g. legal text).
Packaging the extension

- Quite a few Python modules as dependencies: OpenNMT, PyTorch, SynTok (to get sentences) + quite a few indirect ones
- Some are platform / Python-Version specific
- Current solution: build “simple” OXT
- Install dependencies in fresh LibreOffice with pip
- Copy site-packages into OXT zipfile (30MB → 150MB)
- Automate OXT build?

Probably want to shed some dependencies, maybe port to C++.
UI - Installation

- **Standard Extension**
- *(Only) for Windows:* all dependencies bundled
Install translation models (unpacked from ZIP file) in the new Options page.
Better integration with LibreOffice?

Some things I would like your input on:

• How much is MT a desired standard feature?
• Working on other than Writer text?
• Do we need more UI?
• Interest in more languages?
• What size of language models is acceptable? (x.xGB per language pair) Distribution channel?
How to make this more useful?

• Better integration with LibreOffice Writer (e.g. set the Language properties...)

• Would you train a model for your language (pairs)?
  What do you look for for training models?
  (except a speedup / being more economical)

• Open Corpora: [http://opus.nlpl.eu/](http://opus.nlpl.eu/)
  (incidentally uses translated UI strings as a source)

• NMT researchy (or catching up with NMT research):
  - Can we use dictionaries, too?
  - Improve using “weakly aligned” texts? (from gutenberg.org, LibreOffice documentation?)
    Related research: automatic filtering noisy corpora (WMT challenge 18 & 19)

Your other comments and observations
Thank you!

Contact:
Thomas Viehmann
https://mathinf.eu/
https://lernapparat.de/