Baseline

A Library for Rapid Modeling, Experimentation and Development of Deep Learning Algorithms targeting NLP

Daniel Pressel, Sagnik Ray Choudhury, Brian Lester, Yanjie Zhao, Matt Barta

NLP OSS Workshop @ ACL 2018

Baseline: A Deep NLP library built on these principles

- simplicity is best
  - Minimal dependencies, effective design patterns
  - Add value but never detract from a DL framework
  - A la carte design: take only what you need
- baselines should be strong, reflect NLP zeitgeist
- boilerplate code for training deep NLP models should be baked in
  - Flexible builtin loaders, datasets, embeddings, trainers, evaluation, baselines
  - 80% use-case should be trivial, the rest should be as simple as possible

Baseline: A Deep NLP library built on these principles

- experiments should be automatically reproducible and tracked
    - Models, hyper-parameters
    - Standard metrics and datasets facilitate better model comparisons
- research benefits from rapid development, automatic deployment
    - Training should be efficient, work on multiple GPUs where possible
    - Library should provide reusable components to accelerate development
- go where the user is: do not make them come to you!

Use Baseline code base if you want...

- A reusable harness to train models and track experiments
    - Focus on the models instead of the boilerplate
    - Define your configuration with a model and a configuration file
- Strong, well-tested deep baselines for common NLP tasks
    - Classification
    - Tagging
    - Seq2seq
    - Language Modeling
- Support for your favorite DL framework
    - TensorFlow, PyTorch and DyNet all supported
- Reusable components to build your own SoTA models

Use Baseline code base if you want...

- A Leaderboard to track progress of your models and HP configurations
- Support for auto-deployment into production (caveat: TF only)
- Built-in dataset and embedding downloads
- Strong models, with addon support for...
    - Transformer
    - ELMo
    - Gazetteers

Future

- More tasks!
- Even stronger baselines!
- Faster training!
- Recipes with pre-training using LMs
- local experiment repo, streaming support
    - For live monitoring and control from a frontend
- native framework optimized readers
- Better integration with other OSS projects
- HPO utilities
- Open experiment server
- Web interface for launching/management

Want to help build?

- PRs welcome!
- Codebase:
  - https://github.com/dpressel/baseline
- Public addons:
  - https://github.com/dpressel/baseline/tree/master/python/addons
- Contact Info
  - dpressel@gmail.com, @DanielPressel

Refs: Representations, Cross-Task

- *Distributed Representations of Words and Phrases and their Compositionality (Mikolov, Sutskever, Chen, Corrado, Dean)*
  - https://arxiv.org/abs/1310.4546
- *Exploiting Similarities among Languages for Machine Translation (Mikolov, Le, Sutskever)*
  - https://arxiv.org/abs/1309.4168
- *Efficient Estimation of Word Representations in Vector Space (Mikolov, Chen, Corrado, Dean)*
  - https://arxiv.org/abs/1301.3781
- *Deep contextualized word representations (Peters et al)*
  - https://export.arxiv.org/pdf/1802.05365
- *Finding Function in Form: Compositional Character Models for Open Vocabulary Word Representation (Ling et al)*
  - https://arxiv.org/pdf/1508.02096.pdf
- *Natural Language Processing (Almost) from Scratch (Collobert et al)*
  - http://jmlr.org/papers/volume12/collobert11a/collobert11a.pdf
- *Enriching Word Vectors with Subword Information (Bojanowski, Grave, Joulin, Mikolov)*
  - https://arxiv.org/abs/1607.04606

Refs: Classification and Neural Architecture

- *Convolutional Neural Networks for Sentence Classification (Kim)*
  - https://arxiv.org/abs/1408.5882
- *Rethinking the Inception Architecture for Computer Vision (Szegedy)*
  - https://arxiv.org/abs/1512.00567
- *Going Deeper with Convolutions (Szegedy et al)*
  - https://arxiv.org/abs/1409.4842
- *Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift (Ioffe/Szegedy)*
  - https://arxiv.org/abs/1502.03167
- *Hierarchical Attention Networks for Document Classification (Yanh et al)*
  - https://www.microsoft.com/en-us/research/publication/hierarchical-attention-networks-document-classification/
- *Deep Residual Learning for Image Recognition (He, Zhang, Ren, Sun)*
  - https://arxiv.org/pdf/1512.03385v1.pdf

Refs: Tagging

- *Learning Character-level Representations for Part-of-Speech Tagging (dos Santos, Zadrozny)*
  - http://proceedings.mlr.press/v32/santos14.pdf
  - https://rawgit.com/dpressel/Meetups/master/nlp-reading-group-2016-03-14/presentation.html#1
- *Boosting Named Entity Recognition with Neural Character Embeddings (dos Santos, Cıcero and Victor Guimaraes)*
  - http://www.aclweb.org/anthology/W15-3904
  - https://rawgit.com/dpressel/Meetups/master/nlp-reading-group-2016-03-14/presentation.html#1
- *Neural Architectures for Named Entity Recognition (Lample et al)*
  - https://arxiv.org/abs/1603.01360
- *End-to-end Sequence Labeling via Bi-directional LSTM-CNNs-CRF (Ma, Hovy)*
  - https://arxiv.org/abs/1603.01354
- *Reporting Score Distributions Makes a Difference: Performance Study of LSTM-networks for Sequence Tagging (Reimers, Gurevych)*
  - http://aclweb.org/anthology/D17-1035
- *Design Challenges and Misconceptions in Neural Sequence Labeling (Yang, Liang, Zhang)*
  - https://arxiv.org/pdf/1806.04470.pdf

Refs: Encoder Decoders

- *Sequence to Sequence Learning with Neural Networks (Sutskever, Vinyals, Le)*
  - https://arxiv.org/abs/1409.3215
- *Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation (Cho et al)*
  - https://arxiv.org/abs/1406.1078
- *Neural Machine Translation by Jointly Learning to Align and Translate (Bahdanau, Cho, Bengio)*
  - https://arxiv.org/abs/1409.0473
- *Attention Is All You Need (Vaswani et al)*
  - https://arxiv.org/pdf/1706.03762.pdf
- *Show and Tell: A Neural Image Caption Generator (Vinyals, Tosheb, Bengio, Erhan)*
  - https://arxiv.org/pdf/1411.4555v2.pdf
- *Effective Approaches to Attention-based Neural Machine Translation* (Luong, Pham, Manning)
  - https://nlp.stanford.edu/pubs/emnlp15_attn.pdf

Refs: Language Modeling

- *Recurrent Neural Network Regularization (Zaremba, Sutskever, Vinyals)*
  - https://arxiv.org/abs/1409.2329
- *Character-Aware Neural Language Models (Kim, Jernite, Sontag, Rush)*
  - https://arxiv.org/abs/1508.06615
- *Exploring the Limits of Language Modeling (Jozefowicz, Vinyals, Schuster, Shazeer, Wu)*
  - https://arxiv.org/pdf/1602.02410v2.pdf

OPENSEQ2SEQ

Oleksii Kuchaiev, Boris Ginsburg, Igor Gitman, Vitaly Lavrukhin, Carl Case,
Paulius Micikevicius, Jason Li, Vahid Noroozi, Ravi Teja Gadde

NVIDIA.

Overview

1. Toolkit for building sequence to sequence models
   - ✓ Neural Machine Translation
   - ✓ Automated Speech Recognition
   - ✓ Speech Synthesis
2. **Mixed Precision training***

3. Distributed training: multi-GPU and multi-node
4. Extendable
5. Open-source: https://github.com/NVIDIA/OpenSeq2Seq

* Micikevicius et al. “Mixed Precision Training” *ICLR 2018*

Usage & Core Concepts

Core concepts:

- Data Layer
- Encoder
- Decoder
- Loss

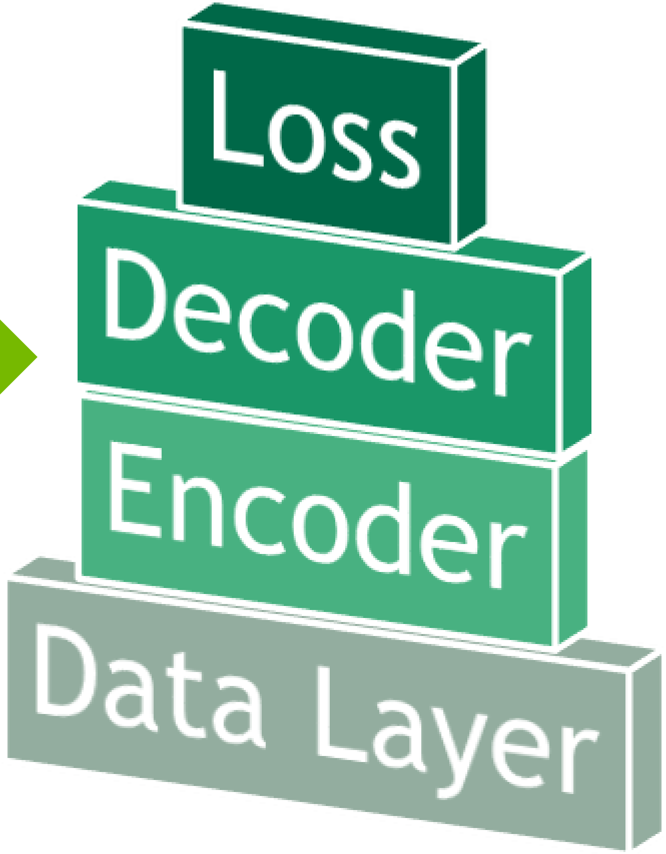
User can mix different encoders and decoders

Flexible Python-based config file

```
"encoder": BidirectionalRNNEncoderWithEmbedding,
"encoder_params": {
  "initializer": tf.glorot_uniform_initializer,
  "core_cell": tf.nn.rnn_cell.LSTMCell,
  "core_cell_params": {
      "num_units": 512,
      "forget_bias": 1.0,
  },
  "encoder_layers": 2,
  "encoder_dp_input_keep_prob": 0.8,
  "encoder_dp_output_keep_prob": 1.0,
  "encoder_use_skip_connections": False,
  "src_emb_size": 512,
  "use_swap_memory": True,
},

"decoder": RNNDecoderWithAttention,
"decoder_params": {
  "initializer": tf.glorot_uniform_initializer,
  "core_cell": tf.nn.rnn_cell.LSTMCell,
  "core_cell_params": {
      "num_units": 512,
      "forget_bias": 1.0,
  },
  "decoder_layers": 2,
```

Seq2Seq model



Mixed Precision Training - *float16*

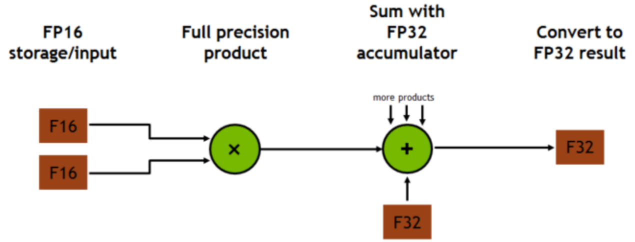
✓ **Train SOTA models faster and using less memory**

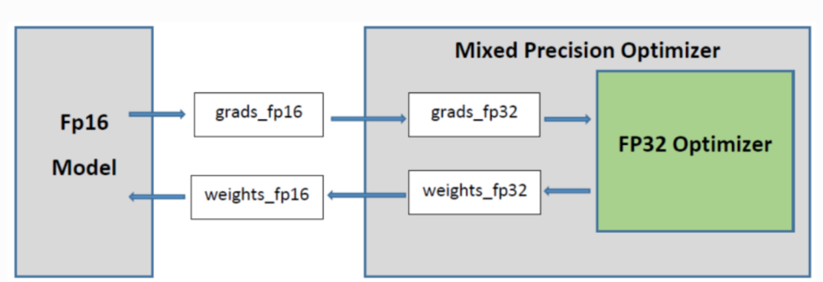
✓ **Keep hyperparameters and network unchanged**

Mixed Precision training*:

1. Use NVIDIA's Volta GPU (for *Tensor Core math*)
2. Maintain *float32* master copy of weights for weights update.
3. Use the *float16* weights for forward and back propagation
4. Apply loss scaling while computing gradients to prevent underflow during backpropagation

Tensor Core math



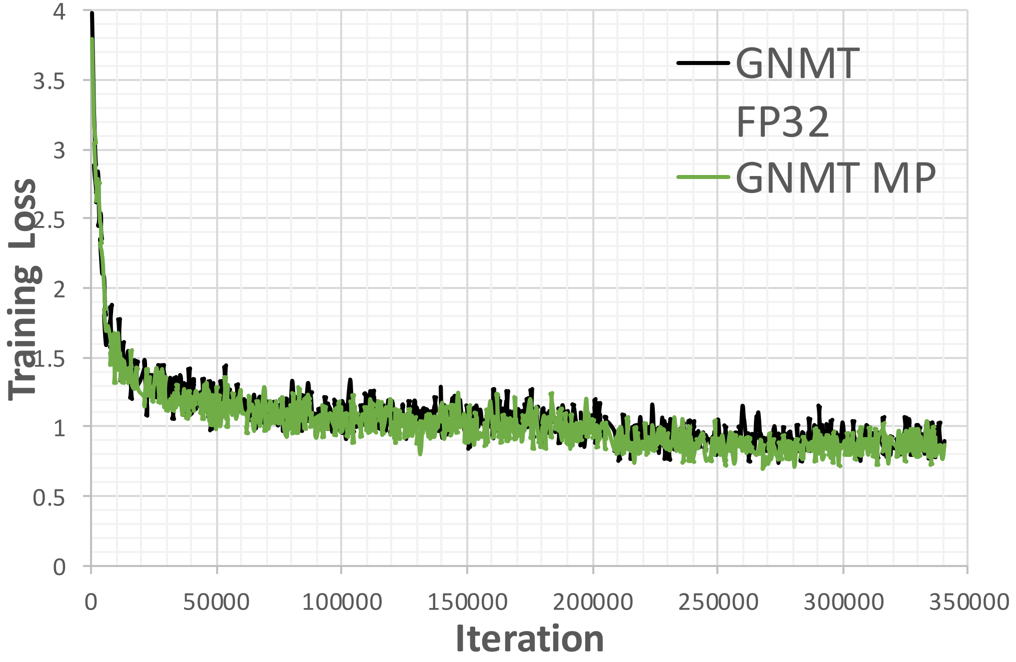


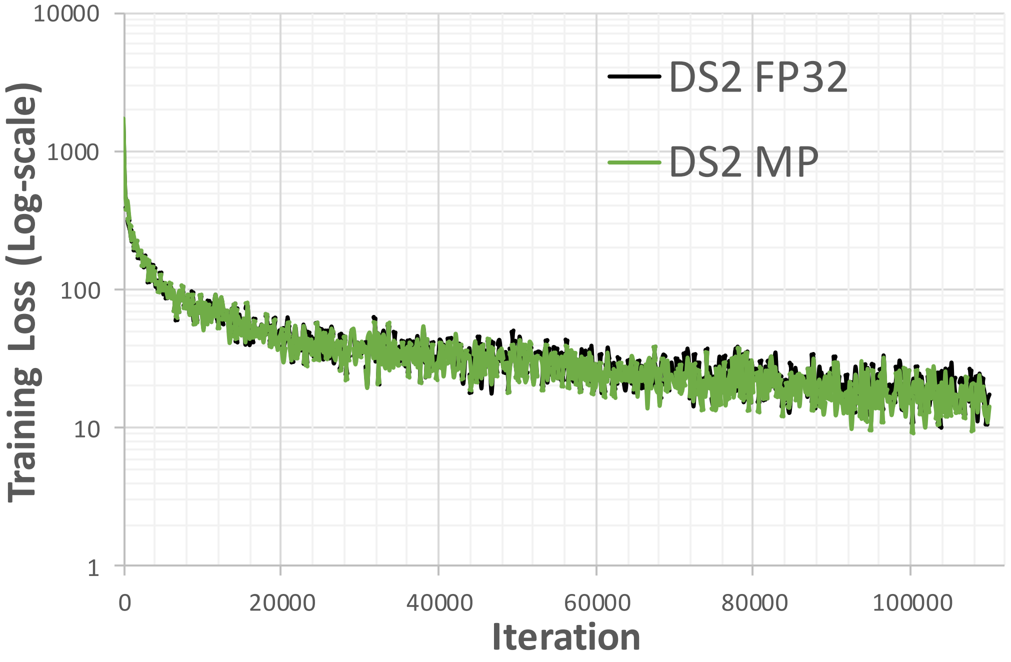
"Mixed precision" optimizer wrapper around any TensorFlow optimizer.

OpenSeq2Seq implements all of this on a base class level

* Micikevicius et al. "Mixed Precision Training" *ICLR 2018*

Mixed Precision Training





Convergence is the same for *float32* and *mixed precision* training. But it is faster and uses about 45% less memory

Summary

OpenSeq2Seq currently implements:

NMT: GNMT, Transformer, ConvSeq2Seq

ASR: DeepSpeech2, Wav2Letter

Speech Synthesis: Tachotron

Makes mixed precision and distributed training easy!

Code, Docs and pre-trained models:

https://github.com/NVIDIA/OpenSeq2Seq

Contributions are welcome!

SUMMA

Scalable Understanding of Multilingual Media

Open-source Software for Multilingual Media-Monitoring

Ulrich Germann,[1] Renārs Liepiņš,[2] Didzis Gosko,[2] Guntis Barzdins[2,3]

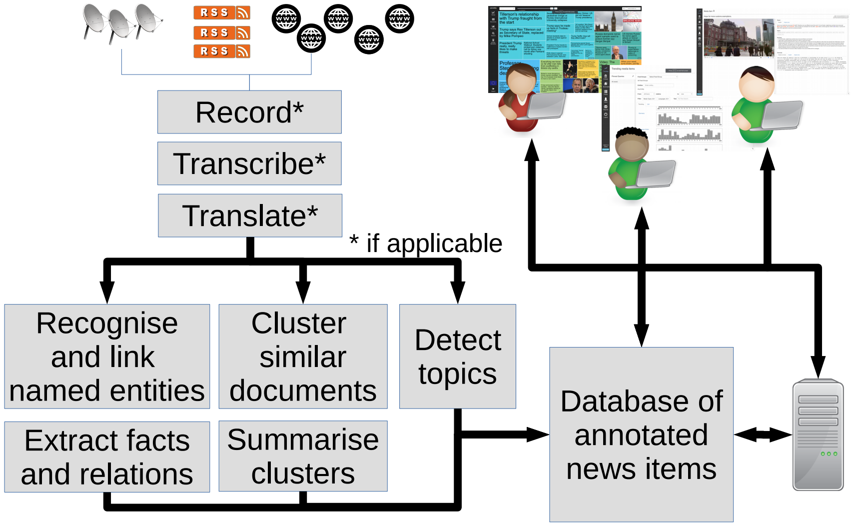
[1] University of Edinburgh; [2] Latvian News Agency; [3] University of Latvia

This work was conducted within the scope of the Research and Innovation Action *SUMMA*, which has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 688139.

Use Case 1: BBC Monitoring

https://www.facebook.com/BBCMonitoring/photos

Workflow


RSS
RSS
RSS
WWW
Record*
Transcribe*
Translate*
* if applicable
Recognise and link named entities
Cluster similar documents
Detect topics
Extract facts and relations
Summarise clusters
Database of annotated news items

NLP Technologies in SUMMA

ASR	Training: Kaldi [1]; transcription: CloudASR [2]
MT	Marian [3]
NE Recognition	Improved TurboEntityRecognizer [4]
NE Linking	Improved TurboParser [4]
Topic Detection	Hierarchical attention model [5]
Doc. Clustering	Online algorithm by Aggarwal & Yu [6]
Summarization	Extractive algorithm by Almeida et al. [7]
Semantic Parsing	AMR parser by Damonte et al. [8]
Timeline Creation	Cornegruta & Vlachos [9]
KB Population	Paikens et al. [10]

Could You Implement Your Technology in X?

Could You Implement Your Technology in X?

So how did we do it?

… come see the poster!

References

[1] D. Povey et al., 2011. "The Kaldi Speech Recognition Toolkit". In: *Proc. ASRU*.

[2] O. Klejch et al., 2015. "CloudASR: Platform and Service". In: *Proc. Int'l. Conf. on Text, Speech, and Dialogue*.

[3] M. Junczys-Dowmunt et al., 2018. "Marian: Fast Neural Machine Translation in C++". In: *ACL Demonstration Session*.

[4] A. F. Martins et al., 2009. "Concise Integer Linear Programming Formulations for Dependency Parsing". In: *Proc. ACL-IJCNLP*.

[5] Z. Yang et al., 2016. "Hierarchical Attention Networks for Document Classification". In: *Proc. NAACL*.

[6] C. C. Aggarwal & P. S. Yu, 2006. "A Framework for Clustering Massive Text and Categorical Data Streams". In: *Proc. SIAM Int'l. Conf. on Data Mining*. SIAM.

[7] M. B. Almeida & A. F. Martins, 2013. "Fast and Robust Compressive Summarization with Dual Decomposition and Multi-Task Learning." In: *ACL*.

[8] M. Damonte et al., 2017. "An Incremental Parser for Abstract Meaning Representation". In: *Proc. EACL*.

[9] S. Cornegruta & A. Vlachos, 2016. "Timeline Extraction Using Distant Supervision and Joint Inference". In: *Proc. EMNLP*.

[10] P. Paikens et al., 2016. "SUMMA at TAC Knowledge Base Population Task 2016". In: *Proc. TAC*.

OpenNMT
Open Source solution for Research … and Industry

Sasha Rush, Harvard NLP, **Jean Senellart** and Guillaume Klein, SYSTRAN

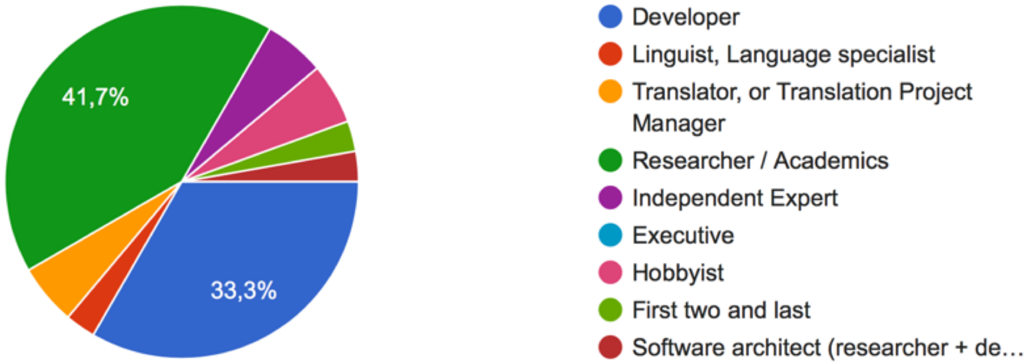
Vincent Nguyen, UBIQUS

What is OpenNMT?

- Generic framework for seq2seq Neural Machine Translation

  - … extending to many other applications
    - End-to-end Speech recognition
    - Img2Text
    - Dialog systems
    - Grammar checking

OpenNMT - 18 months after

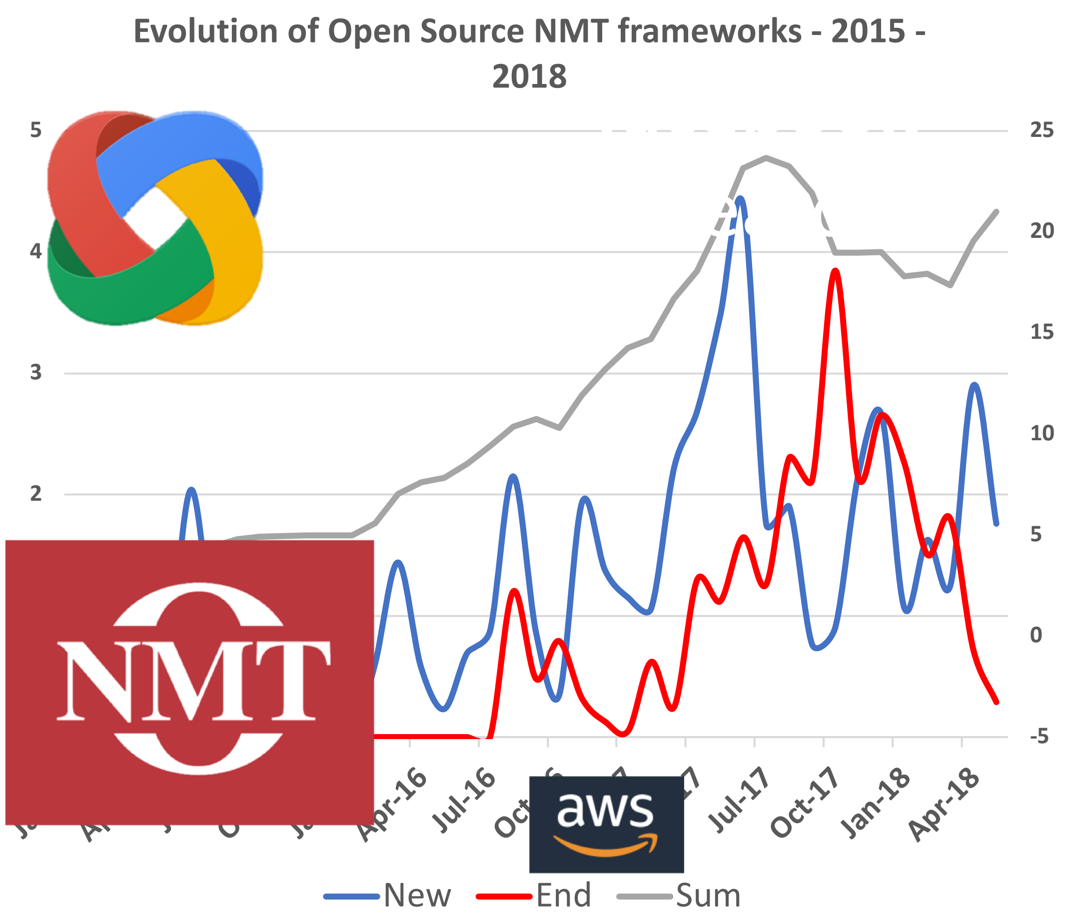


- 18 months since launch
- 3300 stars
- 1020 forks
- 4400 posts on the forum
- 100+ contributors (15 active)
- 18 major releases
- 6 complete code refactoring
- 600 unit tests

Only … 5000 lines of code

NMT Openness

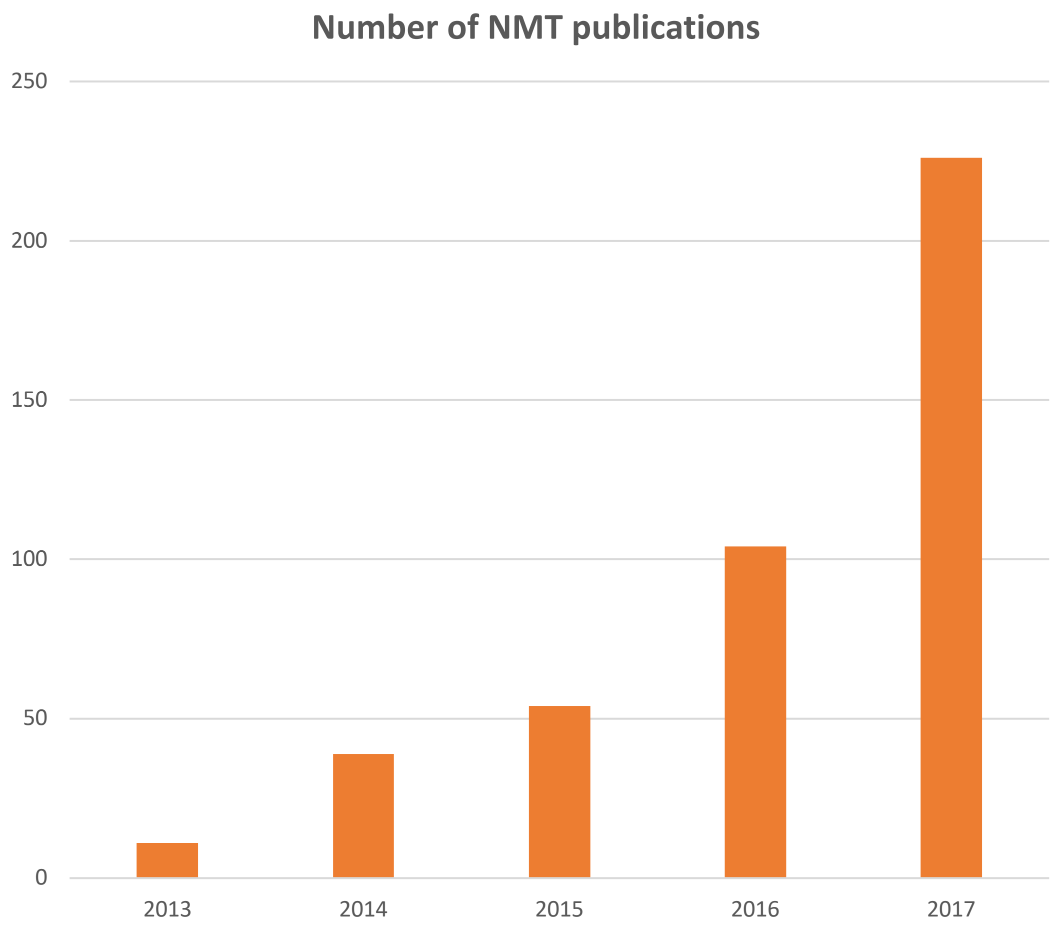
- About 2 new Open Source projects per month
  - Dominated by industry
- Systematic Publications for industrial deployments
  - GNMT, PNMT, Wipro, …
  - Microsoft
- Proof by number
  - DeepL, Omniscient



Very fast changing technology

- Huge number of publications

- Major paradigm change
  - RNN
  - CNN
  - Attention-Based
  - …

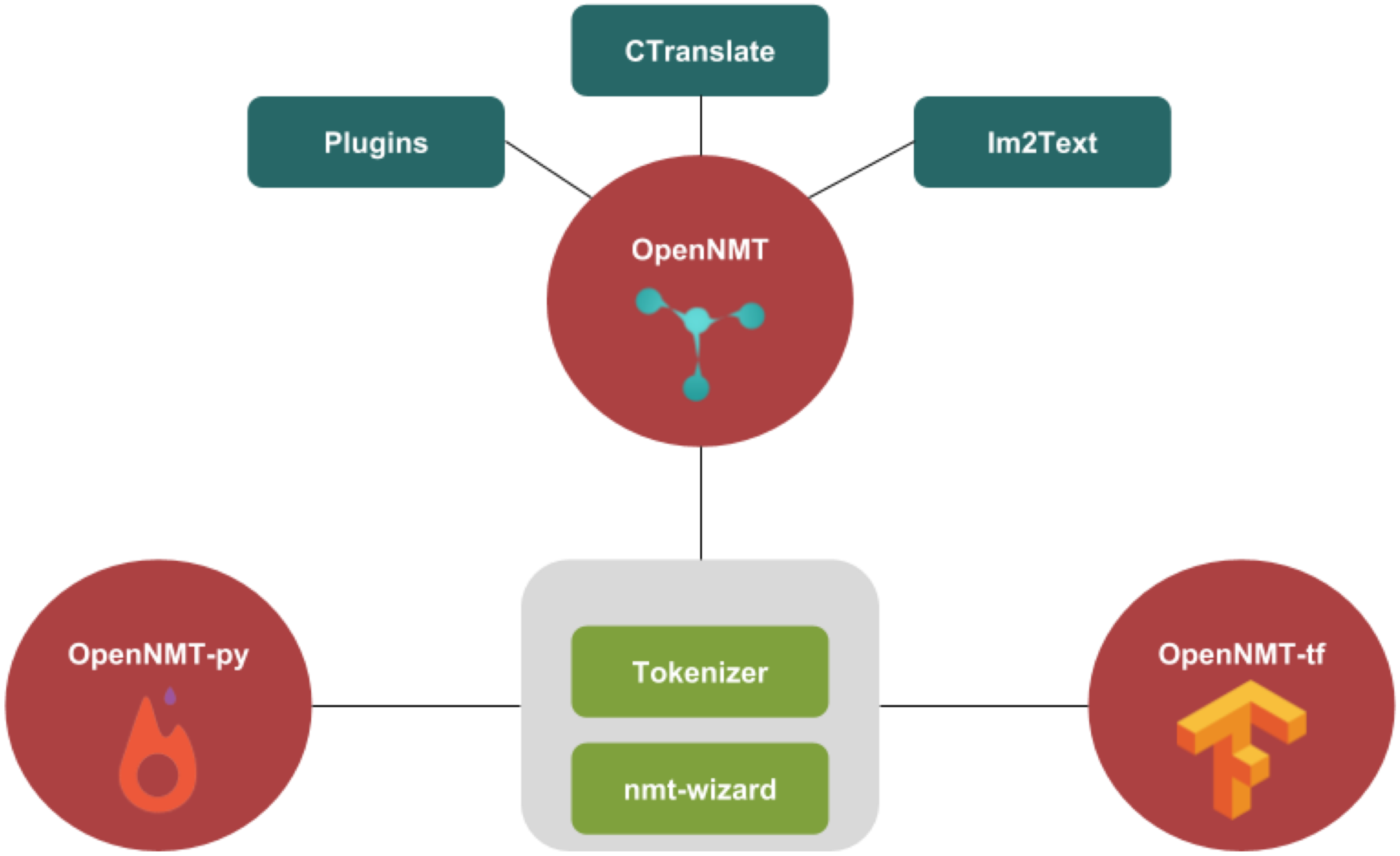


Open Source survival rules

- Open Source is not enough, the community expects:
  - Keep-up with new frameworks
  - Daily support
  - Sharing of data, recipes and good practices
  - Sharing of <u>negative findings</u>
  - Integration of the latest, brightest publications
  - Modularity
  - … Stability

- A lot of work – why are we fighting for this?

Current OpenNMT landscape



An evolving roadmap

OpenNMT	OpenNMT-py	OpenNMT-tf
maintenance mode:	**research-oriented:**	**production-oriented:**
same level of support but no major changes to be expected	flexible hackable	super-fast robust clean APIs

From Research – to … reproduction – to … production

- Come and see our poster!