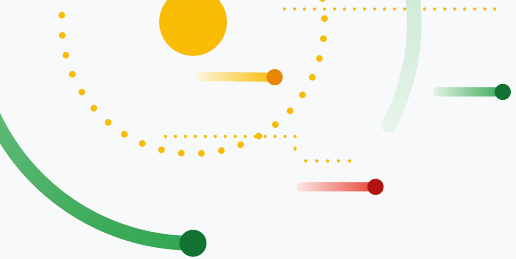
Google AI

Revisiting Character-Based Neural Machine Translation with Capacity and Compression

Colin Cherry*, **George Foster***,
Ankur Bapna, Orhan Firat, Wolfgang Macherey

To appear at EMNLP 2018

*Equal contribution



Google AI

Google Translate Montreal

- Colin Cherry, George Foster
- research into NLP topics relevant to MT
- work alongside Google Brain group in Montreal
- seeking researchers, interns...

The allure of translating characters

- SOTA neural MT is not trained end-to-end: depends on heuristic pre-processing steps including word fragmentation

- Word fragmentation (BPE) is a highly effective technique for open vocabulary MT, but:
  - usually relies on tokenization, needs to be optimized for different settings
  - can make questionable decisions, eg fling → fl + ing

- Can avoid these problems by translating character sequences
  - next step: bytes - handle all languages on equal footing

The perils of characters

- Sequences get longer by ~4x relative to BPE:
  - linear per-layer cost
  - quadratic attention cost
  - longer dependencies to capture
- Finer symbol granularity: potential for attention jitter
- Words are non-compositional in characters: model must memorize many different character sequences rather than using embedding table.

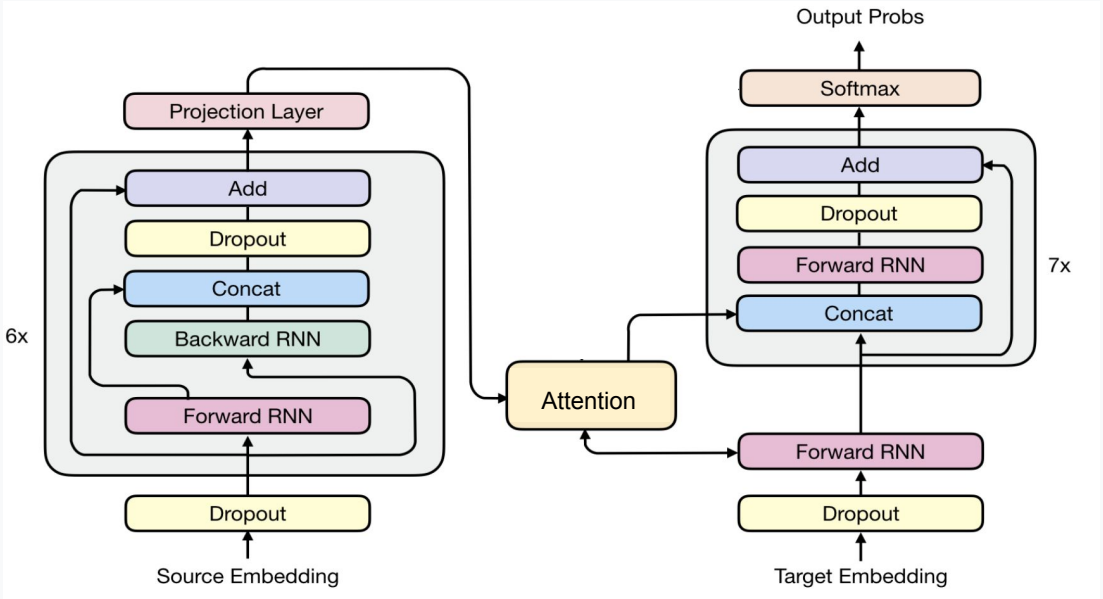
Contributions

- Large-scale empirical comparison of BPE and character-level NMT with deep models
- Investigation of techniques for encoder temporal compression, including first application of Hierarchical Multiscale architecture to NMT

Architecture

Deep architecture based on RNMT+, *Best of Both Worlds*, ACL 2018

- layer normalization
- large batches (16k symbols)
- dim 512
- used for BPE & char; voc sizes:
  - BPE32k voc
  - char 500
- **dropout tuned separately for BPE and char**



Corpora

Same corpora as *Fully character-level neural machine translation without explicit segmentation*, Lee et al, ACL 2017

- minus Russian-English (licence restriction)
- plus English-French (benchmarking)

corpus	train	dev	test
WMT15 Finnish-English	2.1M	1500	1370
WMT15 German-English	4.5M	3003	2169
WMT15 Czech-English	14.8M	3003	2056
WMT14 English-French	39.9M	3000	3003

Benchmark results

system	type	corpus	theirs	ours
Chen et al (2018)	BPE	EnFr	**41.0**	38.8
Wu et al (2016)	BPE		39.0	
Lee et al (2017)	Char	CsEn DeEn FiEn	22.5 25.8 13.1	**25.9** **31.6** **19.3**

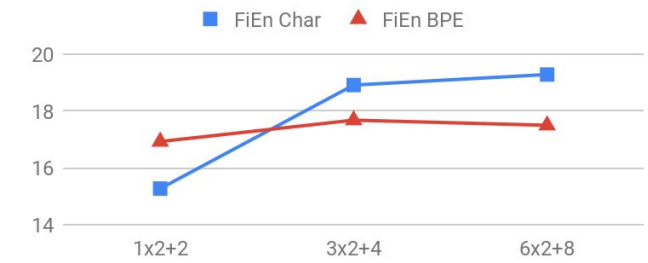
- Reasonably close to BPE SOTA on EnFr
- 3-6 BLEU better than char SOTA on Lee et al languages

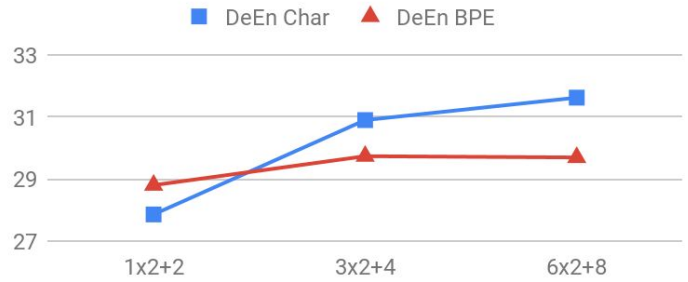
Characters versus BPE

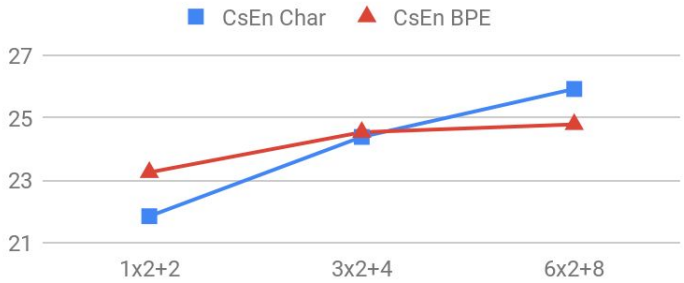
language	BPE	char	delta
English-French	38.8	**39.2**	0.4
Czech-English	24.8	**25.9**	1.1
German-English	29.7	**31.6**	1.9
Finnish-English	17.5	**19.2**	1.8

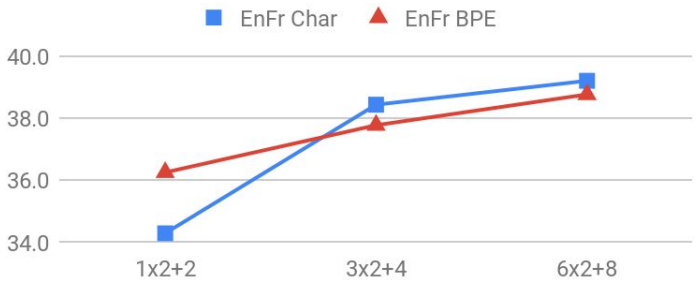
Results contradict those from the literature - we use higher-capacity models.

Effect of capacity

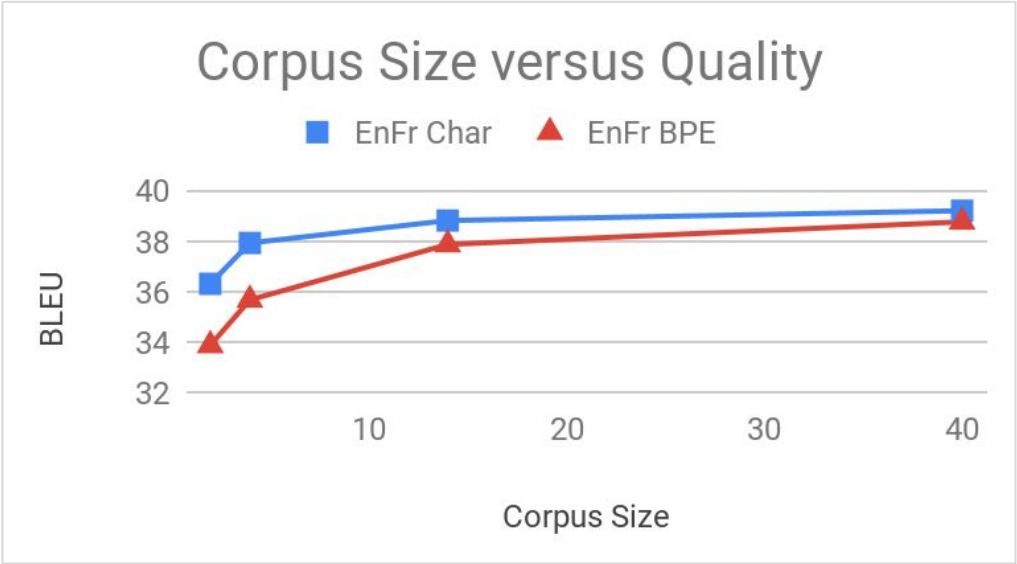






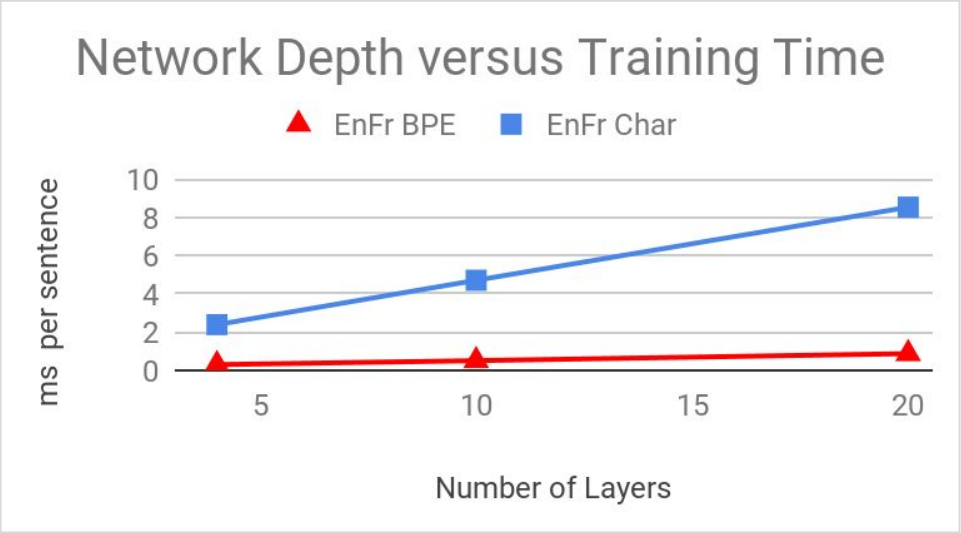


Effect of corpus size



Break-even point around 60-70M

Timing



- Character models train ~8x slower than BPE
- Adding ~5 layers to character model costs as much as attention

Qualitative comparison

Error	BPE	char
Lex choice	19	8
Compounds	13	1
Proper nouns	2	1
Morphology	2	2
Other	2	4
Dropped cont.	7	0

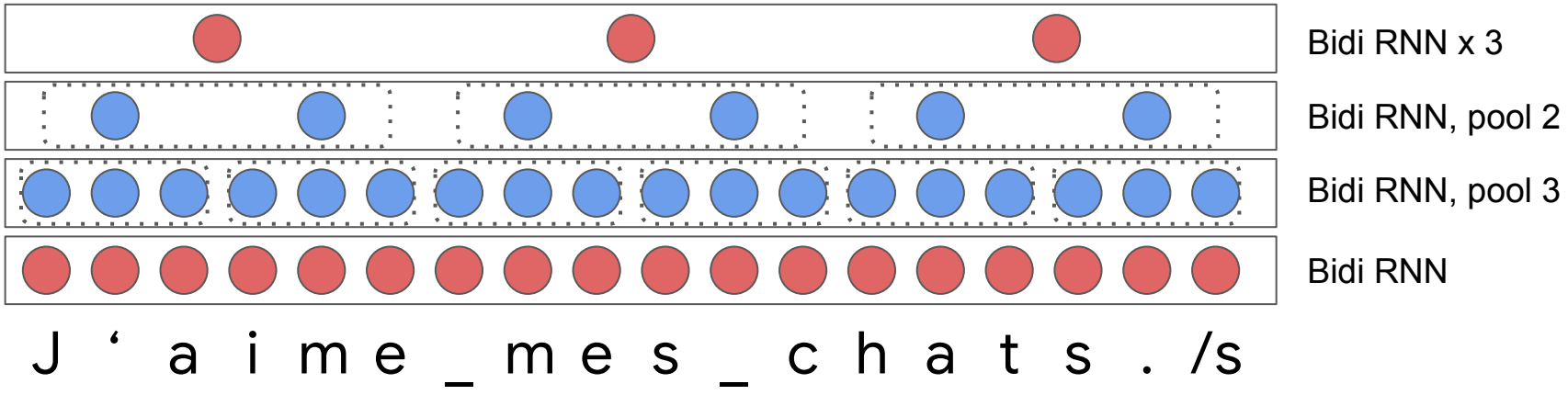
German-English

src	Für diejenigen, die in ländlichen und abgelegenen Regionen **des Staates**, lebten...
ref	Those living in regional and remote areas **of the state**
BPE	For those who lived in rural and remote regions...
char	For those who lived in rural and remote regions **of the state**
src	Überall im Land, in Tausenden von **Chemiestunden**, haben Schüler ihre **Bunsenbrenner** auf Asbestmatten abgestellt.
ref	Up and down the country, in myriad **chemistry lessons**, pupils have perched their **Bunsen burners** on asbestos mats.
BPE	Across the country, thousands of **chemists** have turned their **bullets** on asbestos mats.
char	Everywhere in the country, in thousands of **chemical hours**, students have parked their **bunsen burners** on asbestos mats.

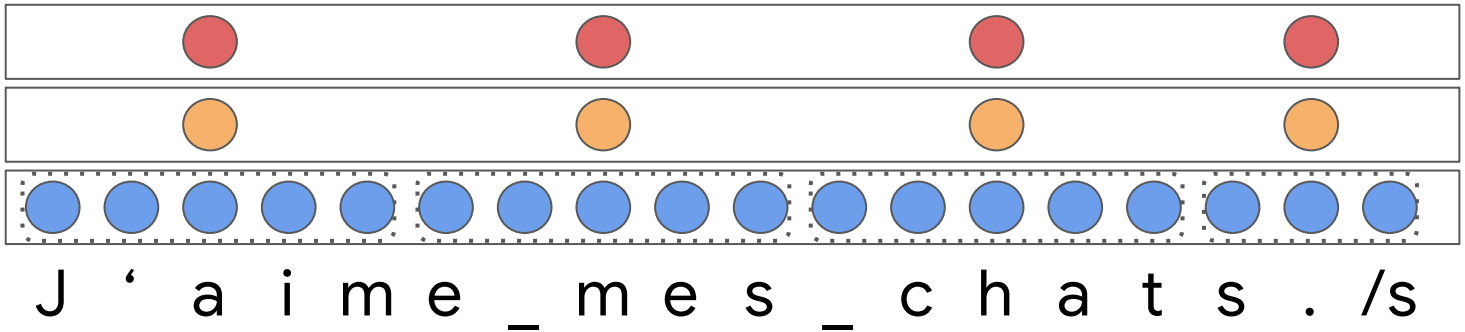
Compressing the source sequence

- Character-level NMT benefits quality, but incurs large computational cost
- Can we represent the source sequence using fewer vectors across time dimension, in order to save computation?
- Compared time/quality tradeoffs of four *temporal compression* techniques:
  - BPE, different vocabulary sizes
  - Fixed-schedule compression:
    - Pyramidal (Chan et al, 2016) - stack increasingly shorter LSTM layers
    - Convolutional (Lee et al, 2017) - pooled convolutional layers
  - Learned-schedule compression:
    - Hierarchical Multiscale LSTM (Chung et al, 2017) - selectively switch off computations - first application to MT

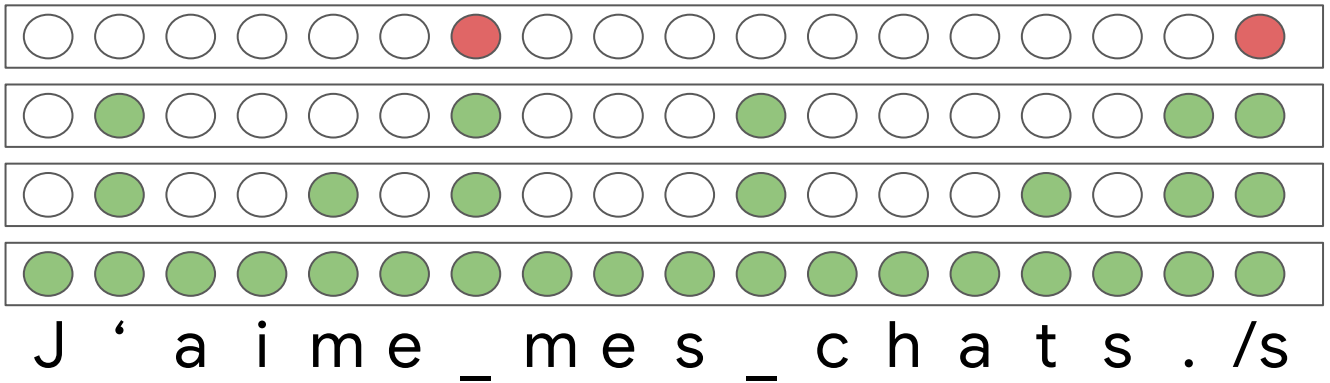
Pyramidal Architecture



Convolutional Architecture



Hierarchical Multiscale Architecture



Each state makes a binary decision (z) about the state above:

- z = 1 - state above does normal LSTM update (green)
- z = 0 - state above is just copied from previous (white)

z decisions come from step function, backprop through hard sigmoid

Adapting the Hierarchical Multiscale Architecture to NMT

- Impose hierarchical structure on z decisions: if z coming from below is 0, z passed up must also be 0
- Don’t flush: z = 0 causes no special behaviour in current state
- Initialize bias term for z gating to 1 - saturation pt of hard sigmoid
- Anneal slope of hard sigmoid closer to step function as training progresses
- Add loss term that penalizes per-layer z proportions of < 10% or >90% to avoid degenerate behaviour

Encoder temporal compression results (German-English)

Encoder	BPE size	BLEU	Computation
BiLSTM	char (500)	31.6	1.00
BiLSTM	1k	30.5	0.44
BiLSTM	2k	30.4	0.35
BiLSTM	4k	30.0	0.29
BiLSTM	8k	29.6	0.25
BiLSTM	16k	30.0	0.22
BiLSTM	32k	29.7	0.20
Pyramidal	char	30.0	0.47
Convolutional	char	28.0	0.20
HM, 3-layer	char	31.2	0.77
HM, 2-layer	char	30.9	0.89

- BPE suffers large drop going from char to 1k voc
- Pyramidal performs somewhat worse than BPE at similar computation rate (~50%)
- Convolutional (not scaled up) performs much worse than BPE at similar computation rate (~20%)
- HM 3-layer slightly worse than char, saves %25 computation

Conclusions

- Translating characters in NMT gives better quality than translating word fragments if models are sufficiently deep, especially if data is limited
- However, translating characters slows training by ~8x
- We investigated three architectures for reducing the amount of encoder computation - none offers a convincingly better time/quality tradeoff than BPE
- Future work: beat BPE and gain the quality advantage from using characters!