

# Natural Language Processing with Deep Learning

## CS224N/Ling284



Christopher Manning

Lecture 12: Information from parts of words:  
Subword Models



# Announcements (Changes!!!)

- Assignment 5 written questions
  - Will be updated tomorrow
- Final Projects due: Fri Mar 13, 4:30pm
- Survey





# Announcements

## Assignment 5:

- Adding convnets and subword modeling to NMT
- Coding-heavy, written questions-light
- The complexity of the coding is similar to A4, **but**:
- **We give you much less help!**
  - Less scaffolding, less provided sanity checks, no public autograder
  - You write your own testing code
- A5 is an exercise in learning to figure things out for yourself
- Essential preparation for final project and beyond
- You now have 7 days—budget time for training and debugging
- Get started soon!



# Lecture Plan

Lecture 12: Information from parts of words: Subword Models

1. A tiny bit of linguistics (10 mins)
2. Purely character-level models (10 mins)
3. Subword-models: Byte Pair Encoding and friends (20 mins)
4. Hybrid character and word level models (30 mins)
5. fastText (5 mins)



# 1. Human language sounds: Phonetics and phonology

- Phonetics is the sound stream – uncontroversial “physics”
- Phonology posits a small set or sets of distinctive, categorical units: **phonemes** or distinctive features
  - A perhaps universal typology but language-particular realization
  - Best evidence of categorical perception comes from phonology
    - Within phoneme differences shrink; between phoneme magnified

CONSONANTS (PULMONIC)

© 2005 IPA

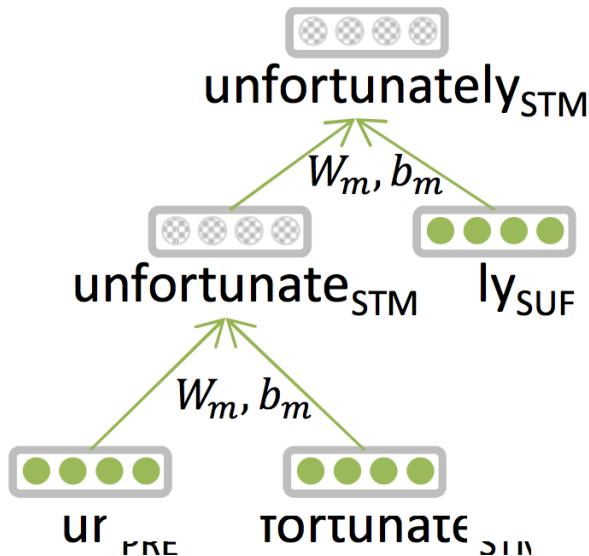
|                     | Bilabial | Labiodental | Dental | Alveolar | Postalveolar | Retroflex | Palatal | Velar | Uvular | Pharyngeal | Glottal |
|---------------------|----------|-------------|--------|----------|--------------|-----------|---------|-------|--------|------------|---------|
| Plosive             | p b      |             |        | t d      |              | t d       | c ɟ     | k g   | q ɢ    |            | ?       |
| Nasal               | m        | m̪          |        | n        |              | ɳ         | ɲ       | ŋ     | N      |            |         |
| Trill               | B        |             |        | r        |              |           |         |       | R      |            |         |
| Tap or Flap         |          | v̪          |        | f̪       |              | t̪        |         |       |        |            |         |
| Fricative           | ɸ β      | f v         | θ ð    | s z      | ʃ ʒ          | ʂ ʐ       | ç ɟ     | x ɣ   | χ ʁ    | ħ ʕ        | h ɦ     |
| Lateral fricative   |          |             | ɬ ɺ    |          |              |           |         |       |        |            |         |
| Approximant         |          | v̪          |        | j̪       |              | ɻ         | j̪      | w̪    |        |            |         |
| Lateral approximant |          |             |        | l̪       |              | ɭ         | ʎ       | L̪    |        |            |         |

caught  
cot  
cat



# Morphology: Parts of words

- Traditionally, we have morphemes as smallest **semantic** unit
  - [[un [[fortun(e) ]<sub>ROOT</sub> ate]<sub>STEM</sub>]<sub>STEM</sub>] ly]<sub>WORD</sub>  
subword mode => morpheme level meaning extraction !
- Deep learning: Morphology little studied; one attempt with recursive neural networks is (Luong, Socher, & Manning 2013)

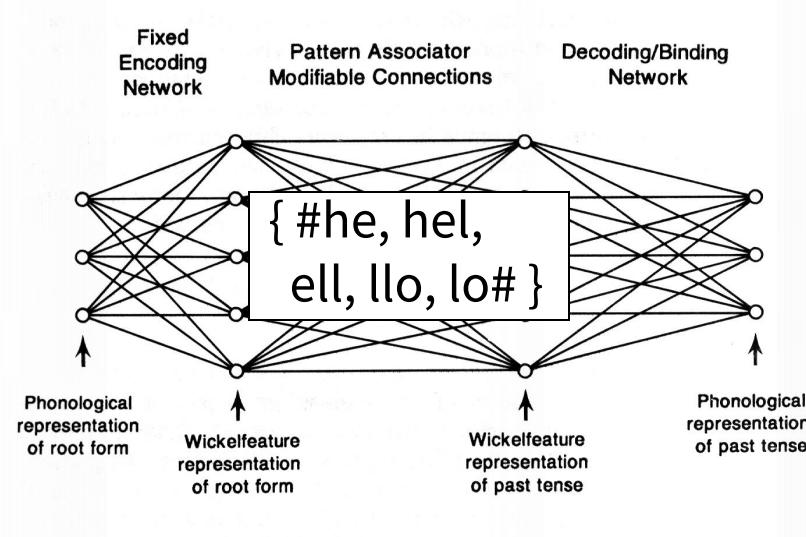


A possible way of dealing with a larger vocabulary – most unseen words are new morphological forms (or numbers)



# Morphology

- An easy alternative is to work with character  $n$ -grams
  - Wickelphones (English past tns Rumelhart & McClelland 1986)
  - Microsoft's DSSM (Huang, He, Gao, Deng, Acero, & Hect 2013)
- Related idea to use of a convolutional layer
- Can give many of the benefits of morphemes more easily??



# Words in writing systems

Writing systems vary in how they represent words – or don't

- No word segmentation 安理会认可利比亚问题柏林峰会成果
- Words (mainly) segmented: *This is a sentence with words.*
  - Clitics/pronouns/agreement?
    - Separated      **Je vous ai apporté** des bonbons
    - Joined              فقلناها = نا + قال + ها = so+said+we+it
  - Compounds?
    - Separated      life insurance company employee
    - Joined              Lebensversicherungsgesellschaftsangestellter

# Models below the word level

- Need to handle **large, open vocabulary**
  - Rich morphology: nejneobhospodařovávatelnějšímu  
tcheck words (“to the worst farmable one”)
  - Transliteration: Christopher ↪ Kryštof
  - Informal spelling:



Brianna @\_parsimonia\_ · 24h

Goooooooood Vibesssssss



@J0YUS · 1m

When idc, I really don't care.

Like my “I want space” is me shutting you out. My “**imma** go, u want something?” And u don't say nothing, then I'm not coming back sumn 4 u



# Character-Level Models

**1.** Word embeddings can be composed from character embeddings

- Generates embeddings for unknown words
- Similar spellings share similar embeddings
- Solves OOV problem

**2.** Connected language can be processed as characters

Both methods have proven to work very successfully!

- Somewhat surprisingly – traditionally, phonemes/letters weren't a semantic unit – but DL models compose groups

# **Below the word: Writing systems**

Most deep learning NLP work begins with language in its written form – it's the easily processed, found data

# But human language writing systems aren't one thing!

- Phonemic (maybe digraphs) jiyawu ngabulu
  - Fossilized phonemic thorough failure
  - Syllabic/moraic つゞく or korean
  - Ideographic (syllabic) <sup>hieroglyph</sup> 去年太空船二号坠毁
  - Combination of the above インド洋の島

ex : spanish/ Wambaya

# Wambaya

## English

# Inuktut

## Chinese

# Japanese

## 2. Purely character-level models

- We saw one good example of a purely character-level model last lecture for sentence classification:
  - Very Deep Convolutional Networks for Text Classification
  - Conneau, Schwenk, Lecun, Barrault. EACL 2017
- Strong results via a deep convolutional stack

# Purely character-level NMT models

- Initially, unsatisfactory performance
  - Vilar et al., 2007; Neubig et al., 2013
- Decoder only
  - Junyoung Chung, Kyunghyun Cho, Yoshua Bengio. arXiv 2016
- Then promising results
  - Wang Ling, Isabel Trancoso, Chris Dyer, Alan Black, arXiv 2015
  - **Thang Luong, Christopher Manning, ACL 2016**
  - Marta R. Costa-Jussà, José A. R. Fonollosa, ACL 2016

# English-Czech WMT 2015 Results

have big joined words/ good for char level

- Luong and Manning tested as a baseline a pure character-level seq2seq (LSTM) NMT system
- It worked well against word-level baseline
- But it was sslooooww
  - 3 weeks to train ... not that fast at runtime

| System  | BLEU |
|---|------|
| Word-level model (single; large vocab; UNK replace) | 15.7 |
| Character-level model (single; 600-step backprop)   | 15.9 |

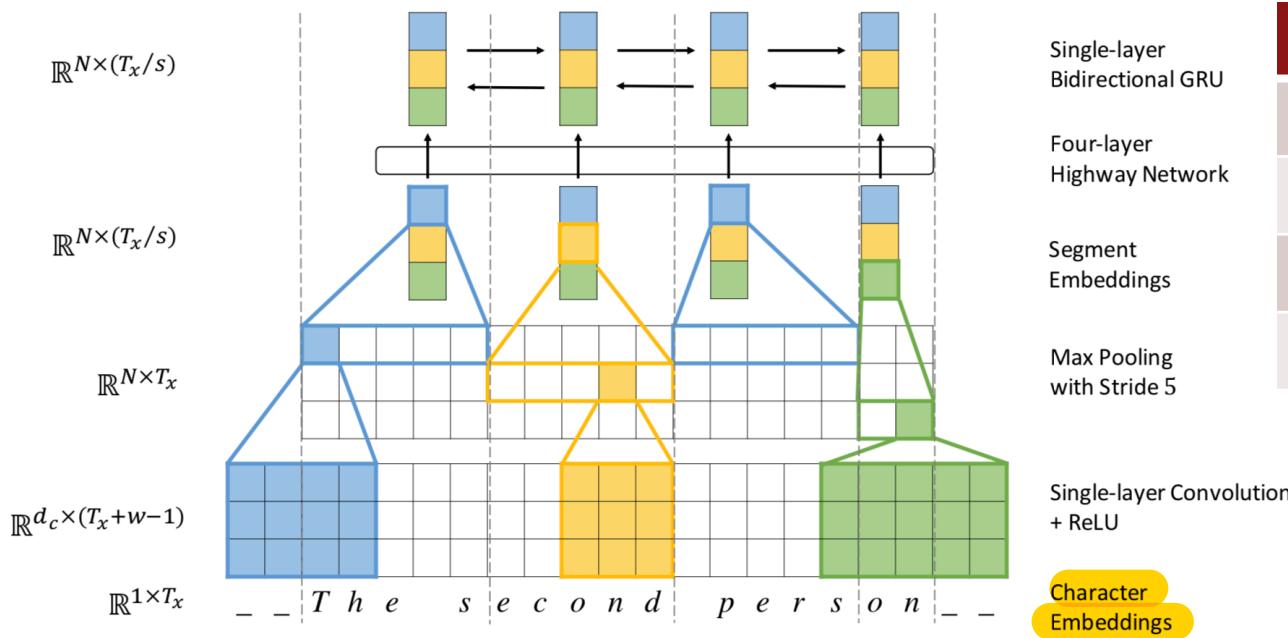
# English-Czech WMT 2015 Example

|        |   |
|--------|---|
| source | Her <b>11-year-old</b> daughter , <b>Shani Bart</b> , said it felt a little bit <b>weird</b>      |
| human  | Její <b>jedenáctiletá</b> dcera <b>Shani Bartová</b> prozradila , že je to trochu <b>zvláštní</b> |
| char   | Její <b>jedenáctiletá</b> dcera , <b>Shani Bartová</b> , říkala , že cítí trochu <b>divně</b>     |
| word   | Její <unk> dcera <unk> <unk> řekla , že je to trochu divné  |
|        | Její <b>11-year-old</b> dcera <b>Shani</b> , řekla , že je to trochu <b>divné</b>                 |

| System  | BLEU |
|---|------|
| Word-level model (single; large vocab; UNK replace) | 15.7 |
| Character-level model (single; 600-step backprop)   | 15.9 |

# Fully Character-Level Neural Machine Translation without Explicit Segmentation

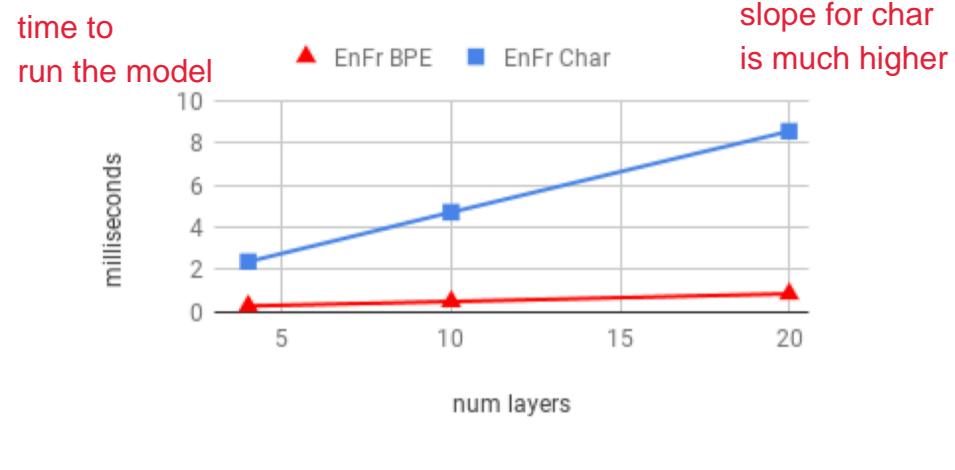
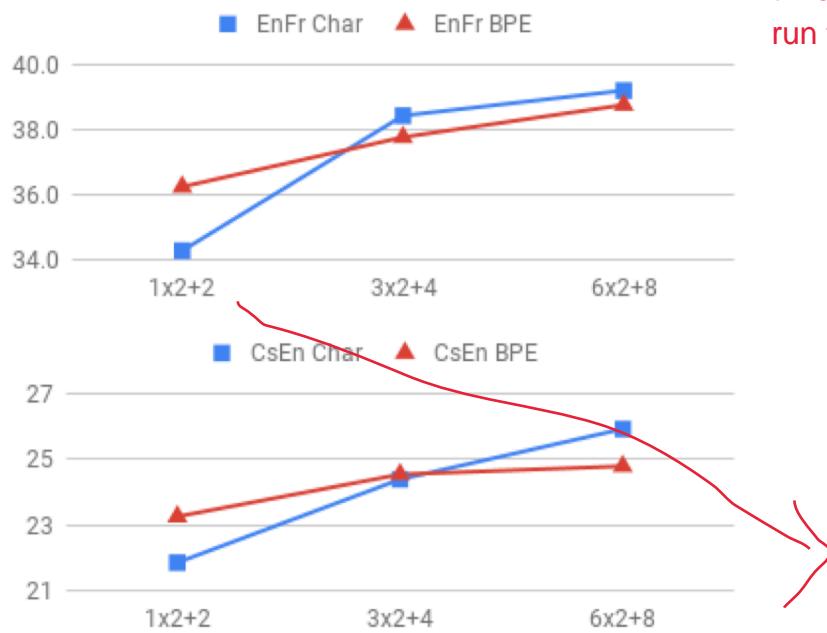
Jason Lee, Kyunghyun Cho, Thomas Hoffmann. 2017.  
Encoder as below; decoder is a char-level GRU



| Cs-En  | WMT 15 | Test |
|--------|--------|------|
| Source | Target | BLEU |
| Bpe    | Bpe    | 20.3 |
| Bpe    | Char   | 22.4 |
| Char   | Char   | 22.5 |

# Stronger character results with depth in LSTM seq2seq model

Revisiting Character-Based Neural Machine Translation with Capacity and Compression. 2018.  
Cherry, Foster, Bapna, Firat, Macherey, Google AI



1 shallow decoder + 2 layer decoder

### 3. Sub-word models: two trends

- Same architecture as for word-level model:
  - But use smaller units: “word pieces”
  - [Sennrich, Haddow, Birch, ACL’16a],  
[Chung, Cho, Bengio, ACL’16].
- Hybrid architectures:
  - Main model has *words*; something else for *characters*
  - [Costa-Jussà & Fonollosa, ACL’16],  
[Luong & Manning, ACL’16].

# Byte Pair Encoding



- Originally a **compression algorithm**:
  - Most frequent **byte** pair  $\mapsto$  a new **byte**.

Replace bytes with character ngrams

(though, actually, some people have done interesting things with bytes)

Rico Sennrich, Barry Haddow, and Alexandra Birch. **Neural Machine Translation of Rare Words with Subword Units**. ACL 2016.

<https://arxiv.org/abs/1508.07909>

<https://github.com/rsennrich/subword-nmt>

<https://github.com/EdinburghNLP/nematus>

# Byte Pair Encoding

- A word segmentation algorithm:
  - Though done as bottom up clustering
  - Start with a unigram vocabulary of all (Unicode) characters in data
  - Most frequent ngram pairs  $\mapsto$  a new ngram

# Byte Pair Encoding

- A word segmentation algorithm:
  - Start with a vocabulary of characters
  - Most frequent ngram pairs  $\mapsto$  a new ngram

*Dictionary*

5 low  
2 lower  
6 newest  
3 widest

*Vocabulary*

l, o, w, e, r, n, w, s, t, i, d

Start with all characters  
in vocab

# Byte Pair Encoding

- A word segmentation algorithm:
  - Start with a vocabulary of characters
  - Most frequent ngram pairs  $\mapsto$  a new ngram

*Dictionary*

5 low  
2 lower  
6 new es t  
3 wi d es t

*Vocabulary*

I, o, w, e, r, n, w, s, t, i, d, es

Add a pair (e, s) with freq 9

# Byte Pair Encoding

- A word segmentation algorithm:
  - Start with a vocabulary of characters
  - Most frequent ngram pairs  $\mapsto$  a new ngram

*Dictionary*

5 low  
2 lower  
6 new **est**  
3 wid **est**

*Vocabulary*

I, o, w, e, r, n, w, s, t, i, d, es, **est**

Add a pair (es, t) with freq 9

# Byte Pair Encoding

- A word segmentation algorithm:
  - Start with a vocabulary of characters
  - Most frequent ngram pairs  $\mapsto$  a new ngram

*Dictionary*

5 **l o w**  
2 **l o w e r**  
6 **n e w e s t**  
3 **w i d e s t**

*Vocabulary*

l, o, w, e, r, n, w, s, t, i, d, es, est, lo

Add a pair (l, o) with freq 7

# Byte Pair Encoding

- Have a target vocabulary size and stop when you reach it
- Do deterministic longest piece segmentation of words
- Segmentation is only within words identified by some prior tokenizer (commonly Moses tokenizer for MT)
- **Automatically decides** vocab for system
  - No longer strongly “word” based in conventional way

Top places in WMT 2016!  
Still widely used in WMT 2018

# Wordpiece/Sentencepiece model

- Google NMT (GNMT) uses a variant of this
  - V1: wordpiece model
  - V2: sentencepiece model
- Rather than char  $n$ -gram count, uses a greedy approximation to maximizing language model log likelihood to choose the pieces
  - Add  $n$ -gram that maximally reduces perplexity

# Wordpiece/Sentencepiece model

- Wordpiece model tokenizes inside words
- Sentencepiece model works from raw text
  - Whitespace is retained as special token (\_) and grouped normally
  - You can reverse things at end by joining pieces and recoding them to spaces
- <https://github.com/google/sentencepiece>
- <https://arxiv.org/pdf/1804.10959.pdf>

# Wordpiece/Sentencepiece model

- BERT uses a variant of the wordpiece model
  - (Relatively) common words are in the vocabulary:
    - *at, fairfax, 1910s*
  - Other words are built from wordpieces:
    - *hypatia* = *h* *##yp* *##ati* *##a*
- non initial wordpieces = *##xx*
- If you're using BERT in an otherwise word based model, you have to deal with this

# Wordpiece/Sentencepiece model

```
from transformers import BertModel, BertTokenizer
import torch

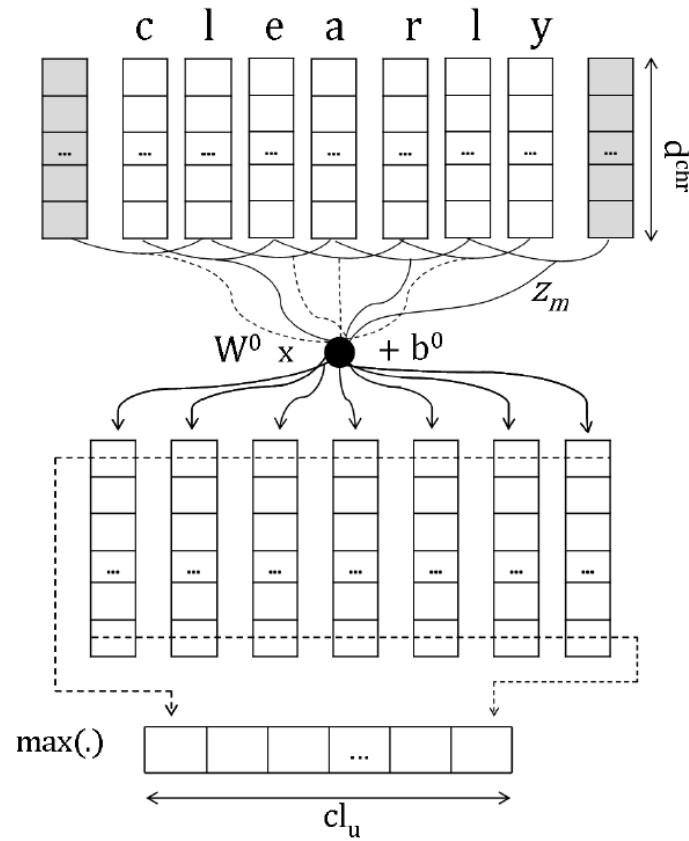
tokenizer =
    BertTokenizer.from_pretrained('bert-base-uncased')
model = BertModel.from_pretrained('bert-base-uncased')

inputs = torch.tensor(tokenizer.encode(
    "Hello, my dog is cute", add_special_tokens=True))
    .unsqueeze(0)  # Batch size 1
outputs = model(inputs)
```

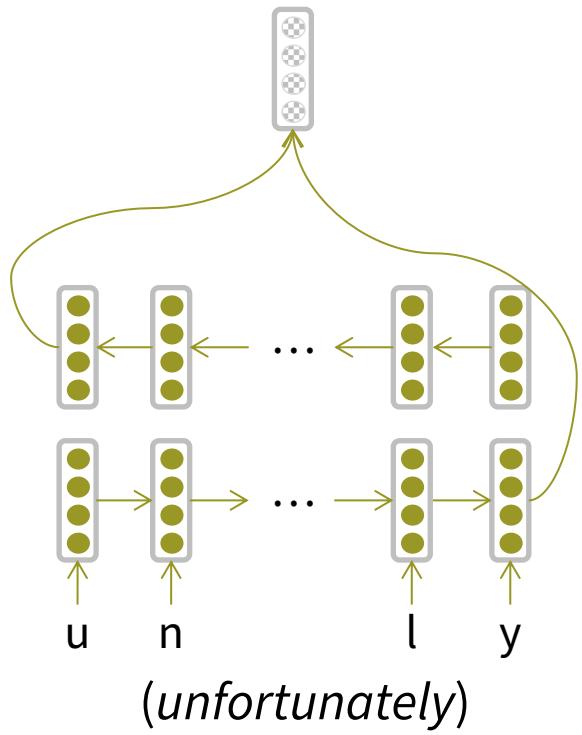
## 4. Character-level to build word-level

### Learning Character-level Representations for Part-of-Speech Tagging (Dos Santos and Zadrozny 2014)

- **Convolution** over characters to generate word embeddings
- Fixed window of word embeddings used for PoS tagging



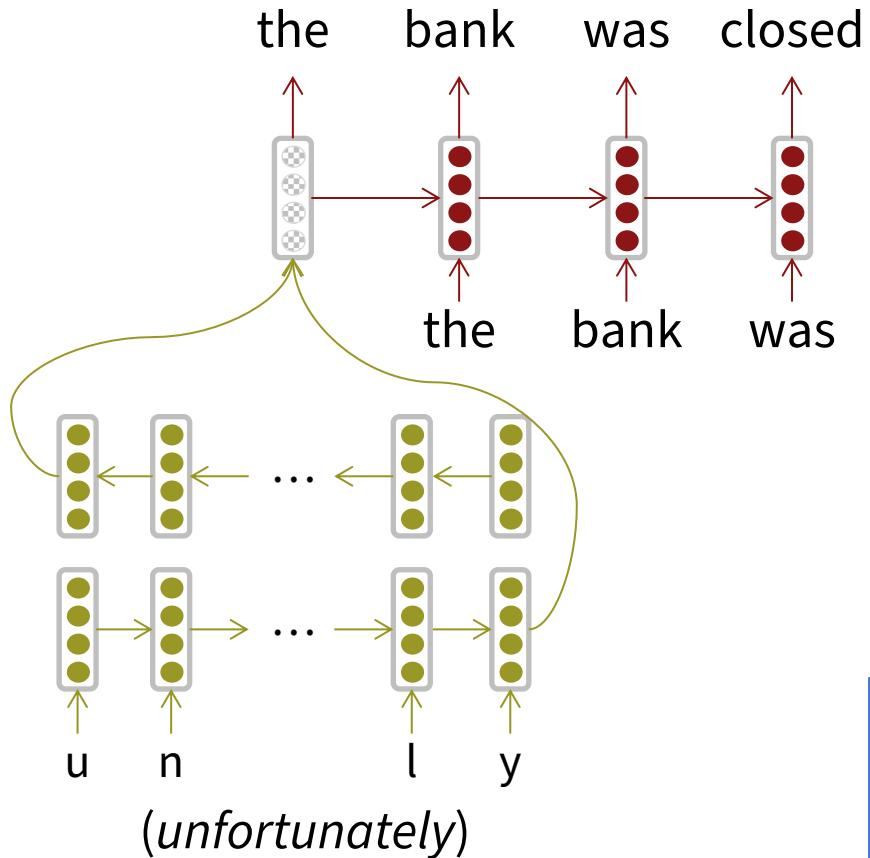
# Character-based LSTM to build word rep'ns



Bi-LSTM builds word representations

Ling, Luís, Marujo, Astudillo, Amir, Dyer, Black, Trancoso. **Finding Function in Form: Compositional Character Models for Open Vocabulary Word Representation**. EMNLP'15.

# Character-based LSTM



Recurrent Language Model

Bi-LSTM builds word representations

Used as LM and for POS tagging

Ling, Luís, Marujo, Astudillo, Amir, Dyer, Black, Trancoso. **Finding Function in Form: Compositional Character Models for Open Vocabulary Word Representation**. EMNLP'15.

# Character-Aware Neural Language Models

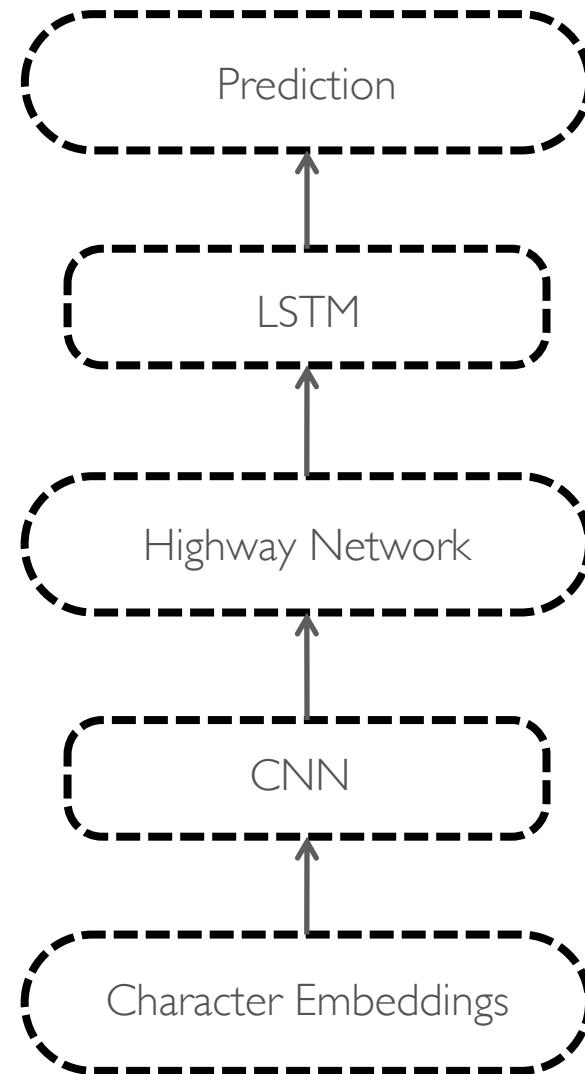
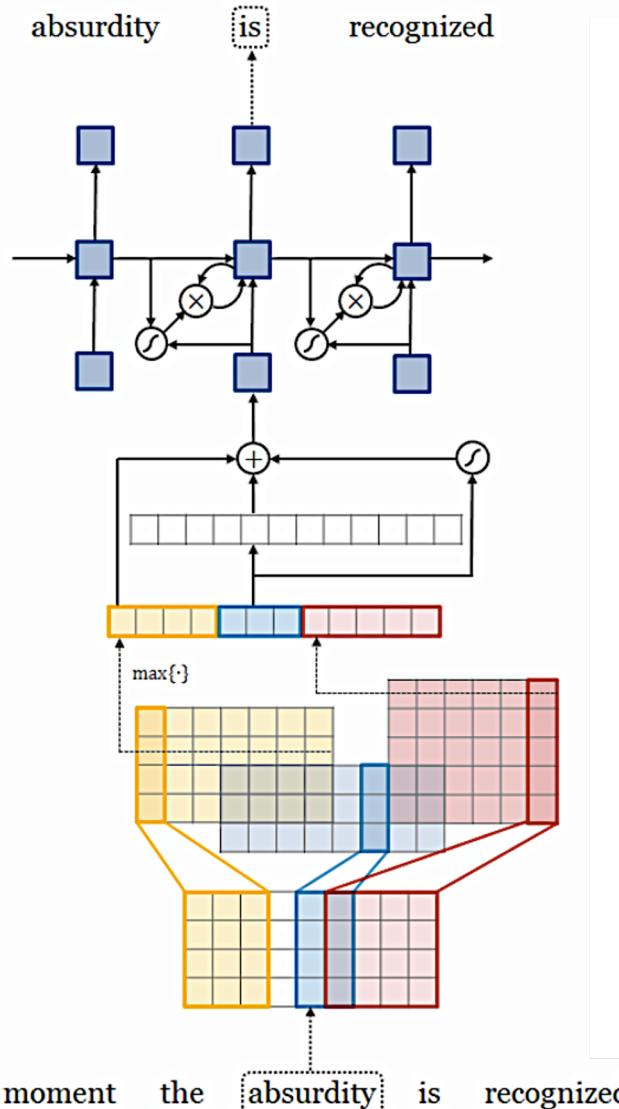
Yoon Kim, Yacine Jernite, David Sontag, Alexander M. Rush. 2015

A more complex/sophisticated approach

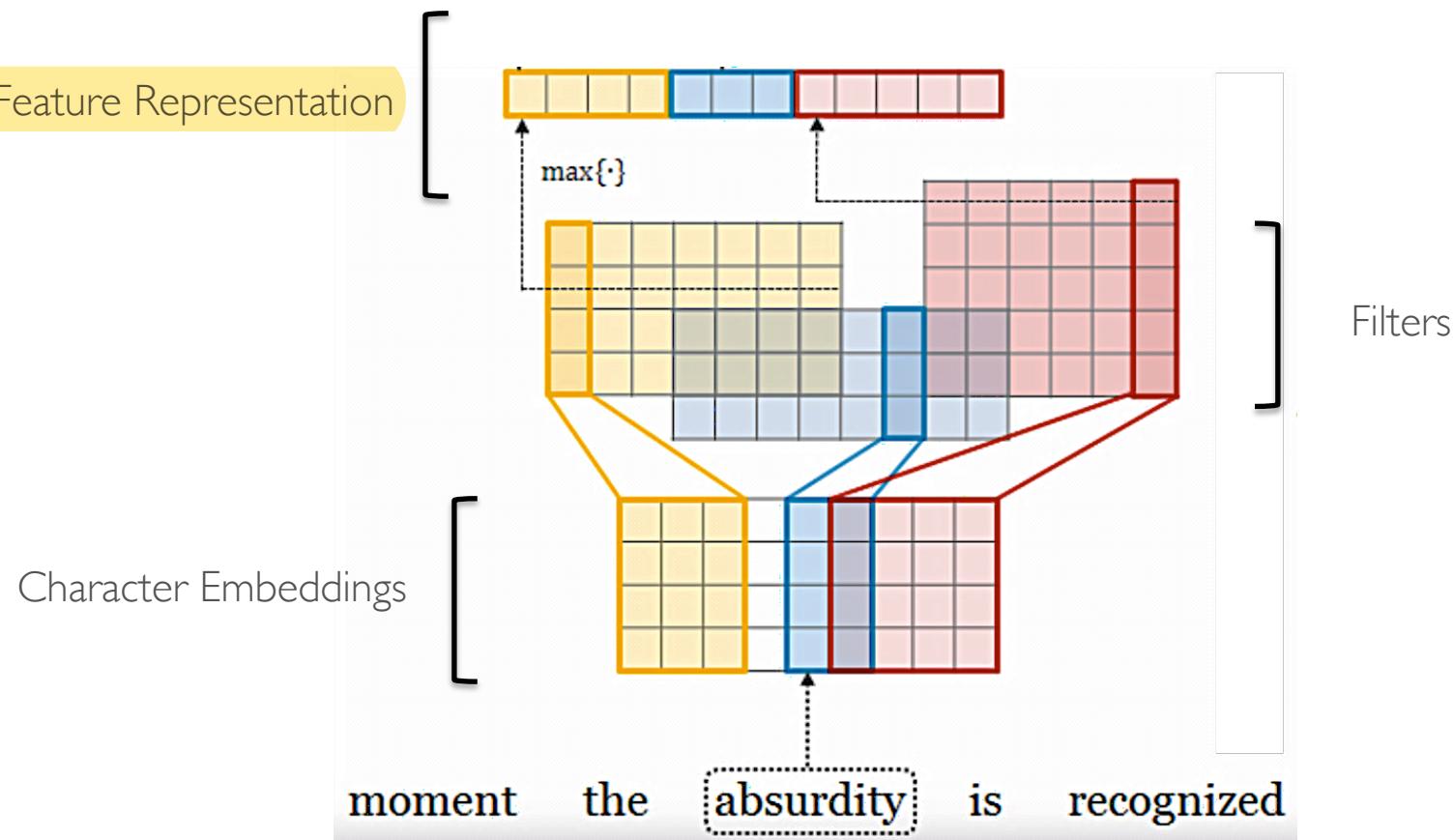
## Motivation

- Derive a powerful, robust language model effective across a variety of languages.
- Encode subword relatedness: *eventful*, *eventfully*, *uneventful*...
- Address rare-word problem of prior models.
- Obtain comparable expressivity with fewer parameters.

# Technical Approach



# Convolutional Layer



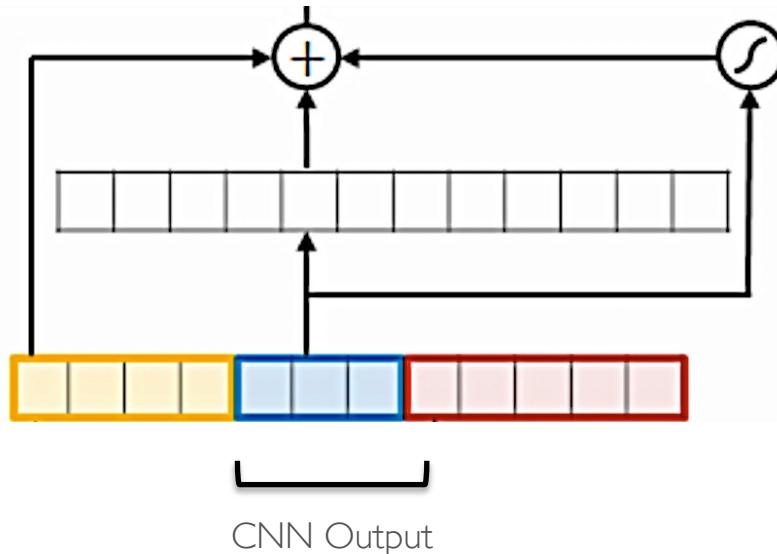
- Convolutions over character-level inputs.
- Max-over-time pooling (effectively n-gram selection).

# Highway Network (Srivastava et al. 2015)

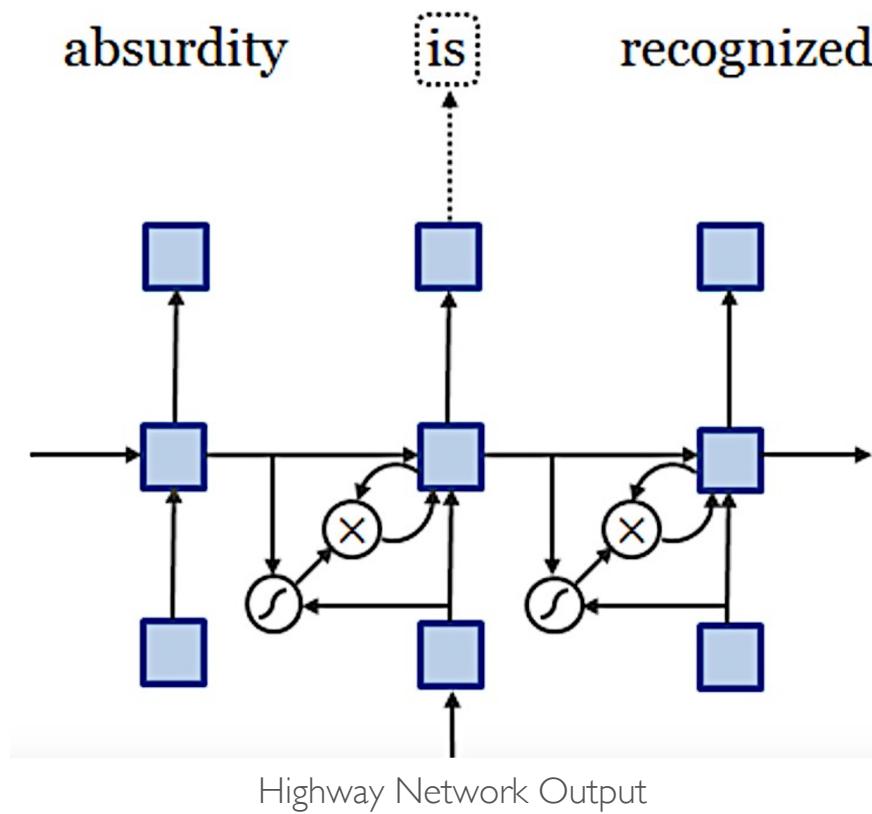
- Model  $n$ -gram interactions.
- Apply transformation while carrying over original information.
- Functions akin to an LSTM cell.

$$\mathbf{t} = \sigma(\mathbf{W}_T \mathbf{y} + \mathbf{b}_T)$$
$$\mathbf{z} = \mathbf{t} \odot g(\mathbf{W}_H \mathbf{y} + \mathbf{b}_H) + (1 - \mathbf{t}) \odot \mathbf{y}$$

↑                      ↑                      ↑  
Transform Gate      Input      Carry Gate



# Long Short-Term Memory Network



- Hierarchical Softmax to handle large output vocabulary.
- Trained with truncated backprop through time.

# Quantitative Results

|       |       | DATA-S     |            |            |            |            |            |
|-------|-------|------------|------------|------------|------------|------------|------------|
|       |       | Cs         | DE         | Es         | FR         | RU         | AR         |
| Botha | KN-4  | 545        | 366        | 241        | 274        | 396        | 323        |
|       | MLBL  | 465        | 296        | 200        | 225        | 304        | –          |
| Small | Word  | 503        | 305        | 212        | 229        | 352        | 216        |
|       | Morph | 414        | 278        | 197        | 216        | 290        | 230        |
|       | Char  | 401        | 260        | 182        | 189        | 278        | 196        |
| Large | Word  | 493        | 286        | 200        | 222        | 357        | 172        |
|       | Morph | 398        | 263        | 177        | 196        | 271        | 148        |
|       | Char  | <b>371</b> | <b>239</b> | <b>165</b> | <b>184</b> | <b>261</b> | <b>148</b> |

|       |       | DATA-L     |            |            |            |            |            |
|-------|-------|------------|------------|------------|------------|------------|------------|
|       |       | Cs         | DE         | Es         | FR         | RU         | EN         |
| Botha | KN-4  | 862        | 463        | 219        | 243        | 390        | 291        |
|       | MLBL  | 643        | 404        | 203        | 227        | <b>300</b> | 273        |
| Small | Word  | 701        | 347        | 186        | 202        | 353        | 236        |
|       | Morph | 615        | 331        | 189        | 209        | 331        | 233        |
|       | Char  | <b>578</b> | <b>305</b> | <b>169</b> | <b>190</b> | 313        | <b>216</b> |

Comparable performance  
with fewer parameters!

|   | PPL         | Size        |
|---|-------------|-------------|
| LSTM-Word-Small                               | 97.6        | 5 m         |
| LSTM-Char-Small                               | 92.3        | 5 m         |
| LSTM-Word-Large                               | 85.4        | 20 m        |
| <b>LSTM-Char-Large</b>                        | <b>78.9</b> | <b>19 m</b> |
| KN-5 (Mikolov et al. 2012)                    | 141.2       | 2 m         |
| RNN <sup>†</sup> (Mikolov et al. 2012)        | 124.7       | 6 m         |
| RNN-LDA <sup>†</sup> (Mikolov et al. 2012)    | 113.7       | 7 m         |
| genCNN <sup>†</sup> (Wang et al. 2015)        | 116.4       | 8 m         |
| FOFE-FNNLM <sup>†</sup> (Zhang et al. 2015)   | 108.0       | 6 m         |
| Deep RNN (Pascanu et al. 2013)                | 107.5       | 6 m         |
| Sum-Prod Net <sup>†</sup> (Cheng et al. 2014) | 100.0       | 5 m         |
| LSTM-1 <sup>†</sup> (Zaremba et al. 2014)     | 82.7        | 20 m        |
| LSTM-2 <sup>†</sup> (Zaremba et al. 2014)     | 78.4        | <b>52 m</b> |

# Qualitative Insights

|                               | In Vocabulary       |              |                      |                 |                    |  |
|-------------------------------|---------------------|--------------|----------------------|-----------------|--------------------|--|
|                               | <i>while</i>        | <i>his</i>   | <i>you</i>           | <i>richard</i>  | <i>trading</i>     |  |
| LSTM-Word                     | <i>although</i>     | <i>your</i>  | <i>conservatives</i> | <i>jonathan</i> | <i>advertised</i>  |  |
|                               | <i>letting</i>      | <i>her</i>   | <i>we</i>            | <i>robert</i>   | <i>advertising</i> |  |
|                               | <i>though</i>       | <i>my</i>    | <i>guys</i>          | <i>neil</i>     | <i>turnover</i>    |  |
|                               | <i>minute</i>       | <i>their</i> | <i>i</i>             | <i>nancy</i>    | <i>turnover</i>    |  |
| LSTM-Char<br>(before highway) | <i>chile</i>        | <i>this</i>  | <i>your</i>          | <i>hard</i>     | <i>heading</i>     |  |
|                               | <i>whole</i>        | <i>hhs</i>   | <i>young</i>         | <i>rich</i>     | <i>training</i>    |  |
|                               | <i>meanwhile</i>    | <i>is</i>    | <i>four</i>          | <i>richer</i>   | <i>reading</i>     |  |
|                               | <i>white</i>        | <i>has</i>   | <i>youth</i>         | <i>richter</i>  | <i>leading</i>     |  |
| LSTM-Char<br>(after highway)  | <i>meanwhile</i>    | <i>hhs</i>   | <i>we</i>            | <i>eduard</i>   | <i>trade</i>       |  |
|                               | <i>whole</i>        | <i>this</i>  | <i>your</i>          | <i>gerard</i>   | <i>training</i>    |  |
|                               | <i>though</i>       | <i>their</i> | <i>doug</i>          | <i>edward</i>   | <i>traded</i>      |  |
|                               | <i>nevertheless</i> | <i>your</i>  | <i>i</i>             | <i>carl</i>     | <i>trader</i>      |  |

# Qualitative Insights

## Out-of-Vocabulary

## *computer-aided*

*misinformed*

*looook*

## *computer-guided*

*computerized*

### *disk-drive*

## *computer*

informed

*performed*

*transformed*

inform

*look*

*cook*

*looks*

*shook*

### *computer-guided*

## *computer-driven*

## *computerized*

## *computer*

## *informed*

*performed*

*outperformed*

*transformed*

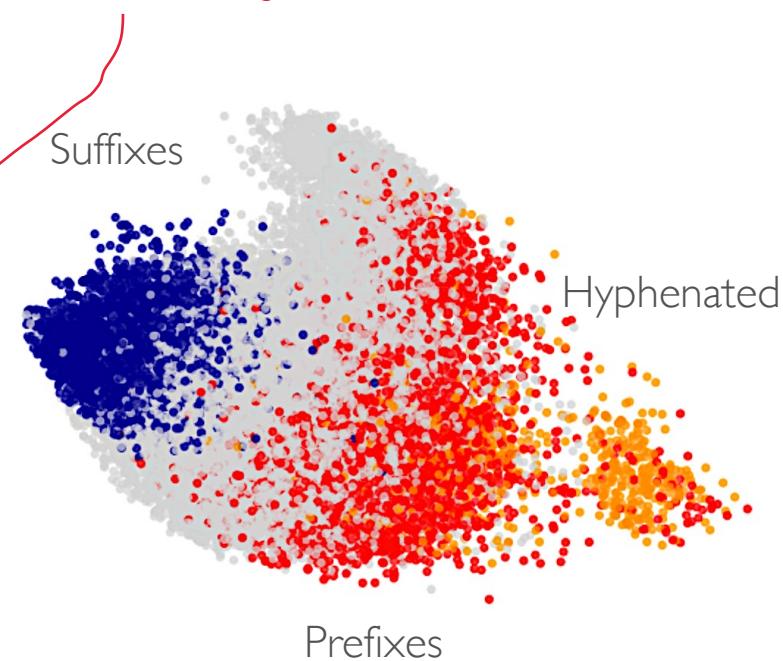
*look*

*looks*

*looked*

*looking*

not having the vocab in the word-level model



# Take-aways

- Paper questioned the necessity of using word embeddings as inputs for neural language modeling.
- CNNs + Highway Network over characters can extract rich semantic and structural information.
- Key thinking: you can compose “building blocks” to obtain nuanced and powerful models!

# Hybrid NMT

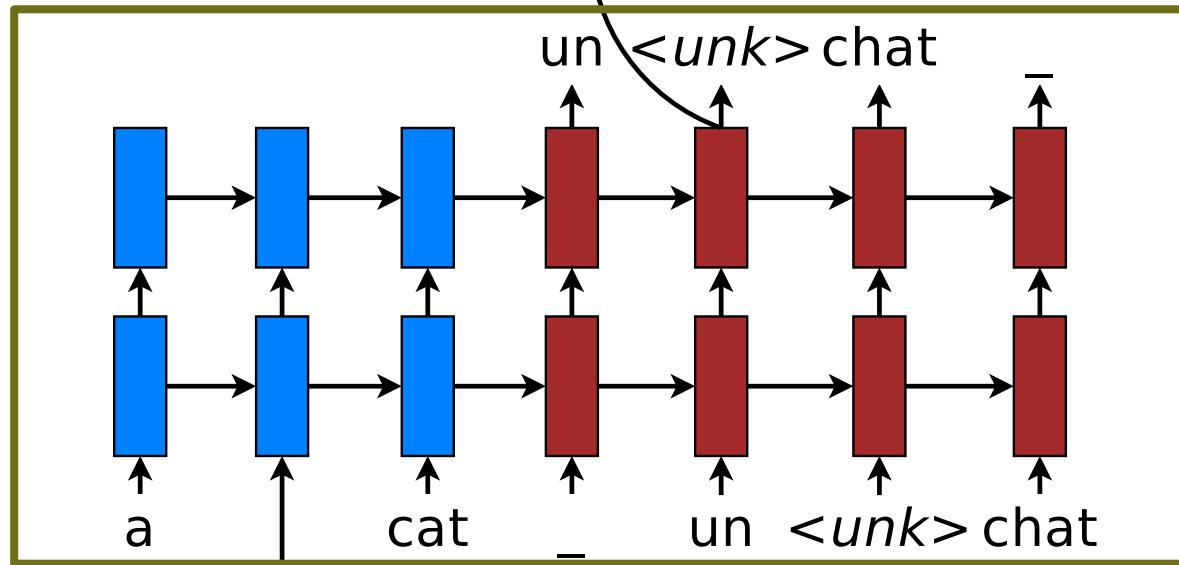


- A *best-of-both-worlds* architecture:
  - Translate mostly at the **word** level
  - Only go to the **character** level when needed
- More than **2 BLEU** improvement over a copy mechanism to try to fill in rare words

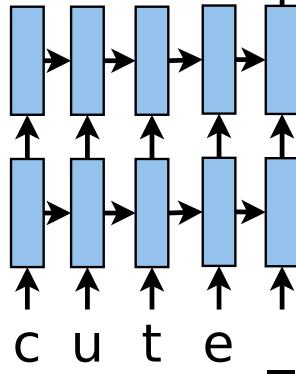
*Thang Luong and Chris Manning. Achieving Open Vocabulary Neural Machine Translation with Hybrid Word-Character Models. ACL 2016.*

# Hybrid NMT

Word-level  
(4 layers)



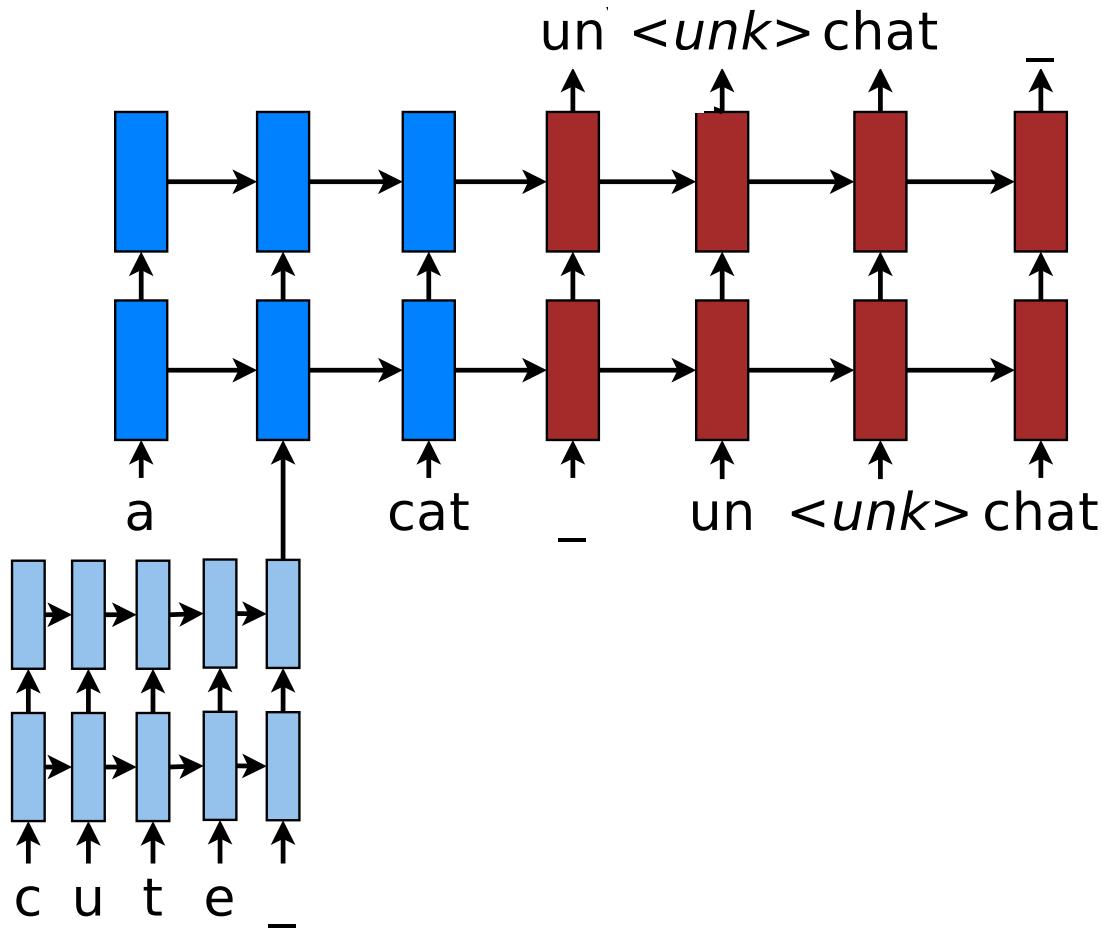
16 000 vocab size  
for words not in vocab, work  
out a char level representation



End-to-end training  
8-stacking LSTM layers.

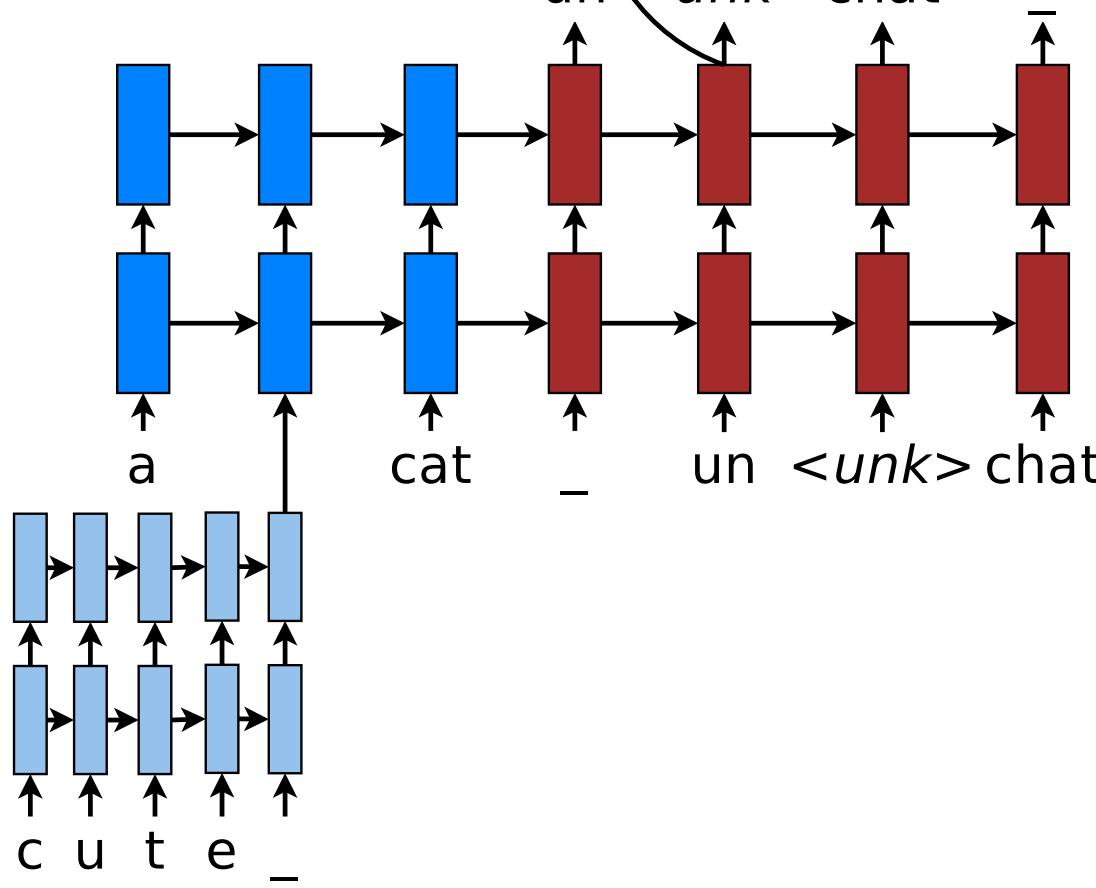
# 2-stage Decoding

- Word-level beam search



# 2-stage Decoding

- Word-level beam search
- Char-level beam search  
for <unk>



Init with word hidden states.

# English-Czech Results

- Train on WMT'15 data (12M sentence pairs)
  - newstest2015

| Systems                                   | BLEU |                                 |
|---|------|---------------------------------|
| Winning WMT'15 (Bojar & Tamchyna, 2015)   | 18.8 | 30x data<br>3 systems           |
| <b>Word-level</b> NMT (Jean et al., 2015) | 18.3 | Large vocab<br>+ copy mechanism |

# English-Czech Results

- Train on WMT'15 data (12M sentence pairs)
  - newstest2015

| Systems                                    | BLEU        |  |
|--|-------------|--|
| Winning WMT'15 (Bojar & Tamchyna, 2015)    | 18.8        | 30x data<br>3 systems  |
| <b>Word-level</b> NMT (Jean et al., 2015)  | 18.3        | Large vocab<br>+ copy mechanism  |
| <b>Hybrid</b> NMT (Luong & Manning, 2016)* | <b>20.7</b> |  Then SOTA! |

# Sample English-Czech translations

|        |  |
|--------|--|
| source | The author <b>Stephen Jay Gould</b> died 20 years after <b>diagnosis</b> . |
| human  | Autor <b>Stephen Jay Gould</b> zemřel 20 let po <b>diagnóze</b> .          |
| char   | Autor <b>Stepher Stepfer</b> zemřel 20 let po <b>diagnóze</b> .            |
| word   | Autor Stephen Jay <unk> zemřel 20 let po <unk> .                           |
|        | Autor <b>Stephen Jay Gould</b> zemřel 20 let po <b>po</b> .                |
| hybrid | Autor Stephen Jay <unk> zemřel 20 let po <unk> .                           |
|        | Autor <b>Stephen Jay Gould</b> zemřel 20 let po <b>diagnóze</b> .          |



# Sample English-Czech translations

|        |  |
|--------|--|
| source | The author <b>Stephen Jay Gould</b> died 20 years after <b>diagnosis</b> . |
| human  | Autor <b>Stephen Jay Gould</b> zemřel 20 let po <b>diagnóze</b> .          |
| char   | Autor <b>Stepher Stepfer</b> zemřel 20 let po <b>diagnóze</b> .            |
| word   | Autor Stephen Jay <unk> zemřel 20 let po <unk> .                           |
|        | Autor <b>Stephen Jay Gould</b> zemřel 20 let po <b>po</b> .                |
| hybrid | Autor Stephen Jay <unk> zemřel 20 let po <unk> .                           |
|        | Autor <b>Stephen Jay Gould</b> zemřel 20 let po <b>diagnóze</b> .          |

- *Char*-based: wrong name translation

# Sample English-Czech translations

|        |  |
|--------|--|
| source | The author <b>Stephen Jay Gould</b> died 20 years after <b>diagnosis</b> . |
| human  | Autor <b>Stephen Jay Gould</b> zemřel 20 let po <b>diagnóze</b> .          |
| char   | Autor <b>Stepher Stepfer</b> zemřel 20 let po <b>diagnóze</b> .            |
| word   | Autor Stephen Jay <unk> zemřel 20 let po <unk>.                            |
|        | Autor <b>Stephen Jay Gould</b> zemřel 20 let po <b>po</b> .                |
| hybrid | Autor Stephen Jay <unk> zemřel 20 let po <unk>.                            |
|        | Autor <b>Stephen Jay Gould</b> zemřel 20 let po <b>diagnóze</b> .          |

- *Word*-based: incorrect alignment

# Sample English-Czech translations

|        |  |
|--------|--|
| source | The author <b>Stephen Jay Gould</b> died 20 years after <b>diagnosis</b> . |
| human  | Autor <b>Stephen Jay Gould</b> zemřel 20 let po <b>diagnóze</b> .          |
| char   | Autor <b>Stepher Stepfer</b> zemřel 20 let po <b>diagnóze</b> .            |
| word   | Autor Stephen Jay <unk> zemřel 20 let po <unk> .                           |
| hybrid | Autor <b>Stephen Jay Gould</b> zemřel 20 let po <b>po</b> .                |
|        | Autor Stephen Jay <unk> zemřel 20 let po <unk> .                           |
|        | Autor <b>Stephen Jay Gould</b> zemřel 20 let po <b>diagnóze</b> .          |

- *Char*-based & *hybrid*: correct translation of **diagnóze**

# Sample English-Czech translation

|        |   |
|--------|---|
| source | Her <b>11-year-old</b> daughter , <b>Shani Bart</b> , said it felt a little bit <b>weird</b>      |
| human  | Její <b>jedenáctiletá</b> dcera <b>Shani Bartová</b> prozradila , že je to trochu <b>zvláštní</b> |
| word   | Její <unk> dcera <unk> <unk> řekla , že je to trochu divné  |
|        | Její <b>11-year-old</b> dcera <b>Shani</b> , řekla , že je to trochu <b>divné</b>                 |
| hybrid | Její <unk> dcera , <unk> <unk> , řekla , že je to <unk> <unk>                                     |
|        | Její <b>jedenáctiletá</b> dcera , <b>Graham Bart</b> , řekla , že cítí trochu <b>divný</b>        |

- Word-based: identity copy **fails**

# Sample English-Czech translation

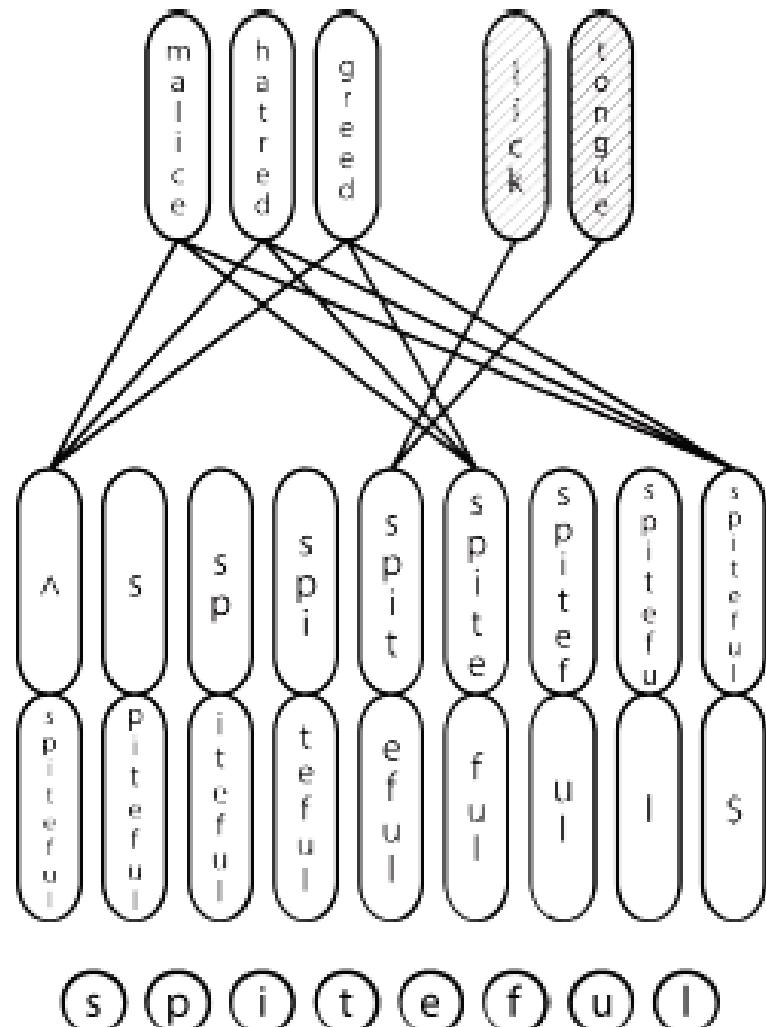
|        |   |
|--------|---|
| source | Her <b>11-year-old</b> daughter , <b>Shani Bart</b> , said it felt a little bit <b>weird</b>      |
| human  | Její <b>jedenáctiletá</b> dcera <b>Shani Bartová</b> prozradila , že je to trochu <b>zvlástní</b> |
| word   | Její <unk> dcera <unk> <unk> řekla , že je to trochu divné  |
| hybrid | Její <b>11-year-old</b> dcera <b>Shani</b> , řekla , že je to trochu <b>divné</b>                 |
|        | Její <unk> dcera , <unk> <unk> , řekla , že je to <unk> <unk>                                     |
|        | Její <b>jedenáctiletá</b> dcera , <b>Graham Bart</b> , řekla , že cítí trochu <b>divný</b>        |

- Hybrid: correct, **11-year-old** – **jedenáctiletá**
- Wrong: **Shani Bartová**

# 5. Char for word embeddings

A Joint Model for Word Embedding and Word Morphology  
(Cao and Rei 2016)

- Same objective as w2v, but using characters
- Bi-directional LSTM to compute embedding
- Model attempts to capture morphology
- Model can infer roots of words



# FastText embeddings

Enriching Word Vectors with Subword Information

Bojanowski, Grave, Joulin and Mikolov. FAIR. 2016.

<https://arxiv.org/pdf/1607.04606.pdf> • <https://fasttext.cc>

- Aim: a next generation efficient word2vec-like word representation library, but better for rare words and languages with lots of morphology
- An extension of the w2v skip-gram model with character  $n$ -grams

# FastText embeddings

- Represent word as char  $n$ -grams augmented with boundary symbols and as whole word:
- $\text{where} = \langle wh, whe, her, ere, re \rangle, \langle where \rangle$ 
  - Note that  $\langle her \rangle$  or  $\langle her \rangle$  is different from  $her$ 
    - Prefix, suffixes and whole words are special
- Represent word as sum of these representations.  
Word in context score is:
  - $s(w, c) = \sum_{g \in G(w)} \mathbf{z}_g^T \mathbf{v}_c$ 
    - Detail: rather than sharing representation for all  $n$ -grams, use “hashing trick” to have fixed number of vectors

# FastText embeddings

new model that uses n-gram

Word similarity  
dataset scores  
(correlations)

|    |        | sg | cbow      | sisg-     | sisg      |
|----|--------|----|-----------|-----------|-----------|
| AR | WS353  | 51 | 52        | 54        | <b>55</b> |
|    | GUR350 | 61 | 62        | 64        | <b>70</b> |
| DE | GUR65  | 78 | 78        | <b>81</b> | <b>81</b> |
|    | ZG222  | 35 | 38        | 41        | <b>44</b> |
| EN | RW     | 43 | 43        | 46        | <b>47</b> |
|    | WS353  | 72 | <b>73</b> | 71        | 71        |
| ES | WS353  | 57 | 58        | 58        | <b>59</b> |
| FR | RG65   | 70 | 69        | <b>75</b> | <b>75</b> |
| RO | WS353  | 48 | 52        | 51        | <b>54</b> |
| RU | HJ     | 59 | 60        | 60        | <b>66</b> |

clear gain

# FastText embeddings

## Differential gains on rare words

|                          | DE     |       | EN    |    | ES    | FR   |
|--------------------------|--------|-------|-------|----|-------|------|
|                          | GUR350 | ZG222 | WS353 | RW | WS353 | RG65 |
| Luong et al. (2013)      | -      | -     | 64    | 34 | -     | -    |
| Qiu et al. (2014)        | -      | -     | 65    | 33 | -     | -    |
| Soricut and Och (2015)   | 64     | 22    | 71    | 42 | 47    | 67   |
| sisg                     | 73     | 43    | 73    | 48 | 54    | 69   |
| Botha and Blunsom (2014) | 56     | 25    | 39    | 30 | 28    | 45   |
| sisg                     | 66     | 34    | 54    | 41 | 49    | 52   |