NLU lecture 6: Compositional character representations

Adam Lopez
alopez@inf.ed.ac.uk

Credits: Clara Vania
2 Feb 2018

Let's revisit an assumption in language modeling (& word2vec)

Probability simply requires us to obey the following rules (remember: V is finite):

$$P : V \rightarrow \mathcal{R}_+$$

$$\sum_{w \in V} P(w \mid w_{i-n+1}, \ldots, w_{i-1}) = 1$$

When does this assumption make sense for language modeling?

But words are not a finite set!

- Bengio et al.: "Rare words with frequency ≤ 3 were merged into a single symbol, reducing the vocabulary size to |V| = 16,383."
- Bahdanau et al.: "we use a shortlist of 30,000 most frequent words in each language to train our models. Any word not included in the shortlist is mapped to a special token ([UNK])."

Src | 日本 の 主要 作物 は 米 で あ る 。
Ref | the main crop of japan is rice .
Hyp | the _UNK is popular of _UNK . _EOS

What if we could scale softmax to the training data vocabulary? Would that help?

Idea: scale by partitioning

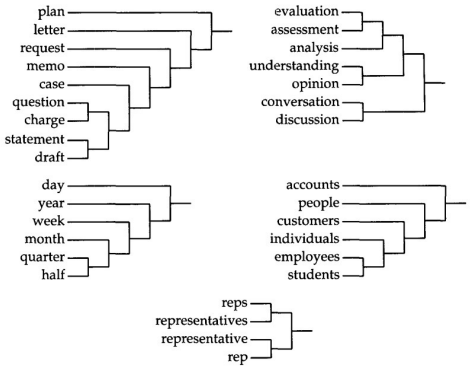
- Partition the vocabulary into smaller pieces.

$$p(w_i|h_i) = p(c_i|h_i)p(w_i|c_i, h_i)$$

Class-based LM

Idea: scale by partitioning

- Partition the vocabulary into smaller pieces hierarchically (*hierarchical* softmax).



Brown clustering: hard clustering based on mutual information

Idea: scale by partitioning

- Differentiated softmax: assign more parameters to more frequent words, fewer to less frequent words.

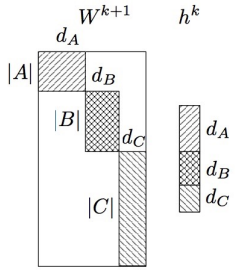
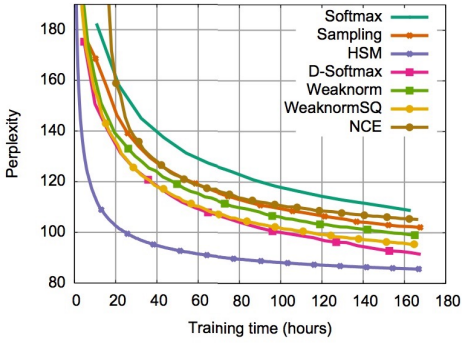


Figure 1: Final weight matrix W^{k+1} and hidden layer h^k for differentiated softmax for partitions A, B, C of the output vocabulary with embedding dimensions d_A, d_B, d_C; non-shaded areas are zero.

Source: Strategies for training large vocabulary language models. Chen, Auli, and Grangier, 2015

Partitioning helps



Dataset	Train	Test	Vocab	OOV
PTB	1M	0.08M	10k	5.8%
gigaword	4,631M	279M	100k	5.6%
billionW	799M	8.1M	793k	0.3%

Table 1: Dataset statistics. Number of tokens for train and test set, vocabulary size and ratio of out-of-vocabulary words in the test set.

Source: Strategies for training large vocabulary language models. Chen, Auli, and Grangier, 2015

Partitioning helps… but could be better

	PTB	gigaword	billionW
KN	141.2	57.1	70.2
Softmax	123.8	56.5	108.3
D-Softmax	121.1	52.0	91.2
Sampling	124.2	57.6	101.0
HSM	138.2	57.1	85.2
NCE	143.1	78.4	104.7
Weaknorm	124.4	56.9	98.7
WeaknormSQ	122.1	56.1	94.9
KN+Softmax	108.5	43.6	59.4
KN+D-Softmax	107.0	42.0	56.3
KN+Sampling	109.4	43.8	58.1
KN+HSM	115.0	43.9	55.6
KN+NCE	114.6	49.0	58.8
KN+Weaknorm	109.2	43.8	58.1
KN+WeaknormSQ	108.8	43.8	57.7

Table 2: Test perplexity of individual models and interpolation with Kneser-Ney.

Source: Strategies for training large vocabulary language models. Chen, Auli, and Grangier, 2015

Partitioning helps… but could be better

Noise contrastive estimation

	PTB	gigaword	billionW
KN	141.2	57.1	70.2
Softmax	123.8	56.5	108.3
D-Softmax	121.1	52.0	91.2
Sampling	124.2	57.6	101.0
HSM	138.2	57.1	85.2
NCE	143.1	78.4	104.7
Weaknorm	124.4	56.9	98.7
WeaknormSQ	122.1	56.1	94.9
KN+Softmax	108.5	43.6	59.4
KN+D-Softmax	107.0	42.0	56.3
KN+Sampling	109.4	43.8	58.1
KN+HSM	115.0	43.9	55.6
KN+NCE	114.6	49.0	58.8
KN+Weaknorm	109.2	43.8	58.1
KN+WeaknormSQ	108.8	43.8	57.7

Table 2: Test perplexity of individual models and interpolation with Kneser-Ney.

Source: Strategies for training large vocabulary language models. Chen, Auli, and Grangier, 2015

Partitioning helps… but could be better

Skip normalization step altogether

	PTB	gigaword	billionW
KN	141.2	57.1	70.2
Softmax	123.8	56.5	108.3
D-Softmax	121.1	52.0	91.2
Sampling	124.2	57.6	101.0
HSM	138.2	57.1	85.2
NCE	143.1	78.4	104.7
Weaknorm	124.4	56.9	98.7
WeaknormSQ	122.1	56.1	94.9
KN+Softmax	108.5	43.6	59.4
KN+D-Softmax	107.0	42.0	56.3
KN+Sampling	109.4	43.8	58.1
KN+HSM	115.0	43.9	55.6
KN+NCE	114.6	49.0	58.8
KN+Weaknorm	109.2	43.8	58.1
KN+WeaknormSQ	108.8	43.8	57.7

Table 2: Test perplexity of individual models and interpolation with Kneser-Ney.

Source: Strategies for training large vocabulary language models. Chen, Auli, and Grangier, 2015

Partitioning helps… but could be better

Room for improvement

	PTB	gigaword	billionW
KN	141.2	57.1	70.2
Softmax	123.8	56.5	108.3
D-Softmax	121.1	52.0	91.2
Sampling	124.2	57.6	101.0
HSM	138.2	57.1	85.2
NCE	143.1	78.4	104.7
Weaknorm	124.4	56.9	98.7
WeaknormSQ	122.1	56.1	94.9
KN+Softmax	108.5	43.6	59.4
KN+D-Softmax	107.0	42.0	56.3
KN+Sampling	109.4	43.8	58.1
KN+HSM	115.0	43.9	55.6
KN+NCE	114.6	49.0	58.8
KN+Weaknorm	109.2	43.8	58.1
KN+WeaknormSQ	108.8	43.8	57.7

Table 2: Test perplexity of individual models and interpolation with Kneser-Ney.

Source: Strategies for training large vocabulary language models. Chen, Auli, and Grangier, 2015

V is not finite

- Practical problem: softmax computation is linear in vocabulary size.
- **Theorem.** The vocabulary of word types is infinite.
  Proof 1. productive morphology, loanwords, “fleek”
  Proof 2. 1, 2, 3, 4, …

What set is finite?

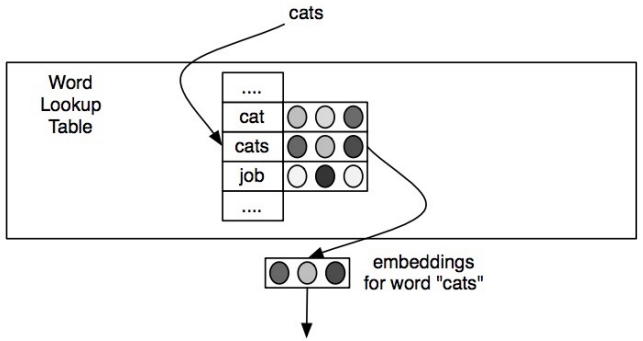
Characters.

More precisely, unicode code points.

🤨 Are you sure? 🤷‍♀️

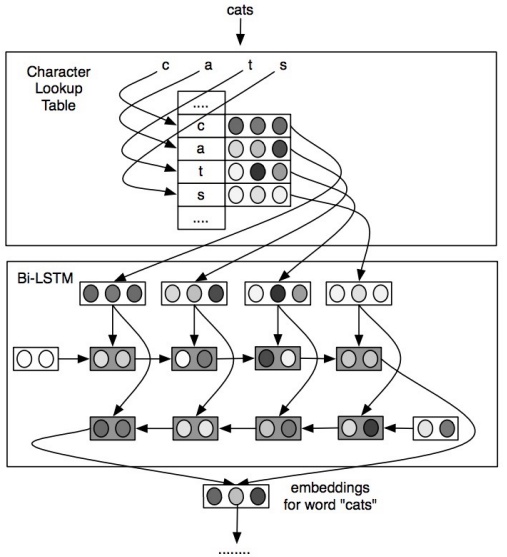
Not all characters are the same, because not all languages have alphabets. Some have syllabaries (e.g. Japanese kana) and/ or logographies (Chinese hànzì).

Rather than look up word representations…



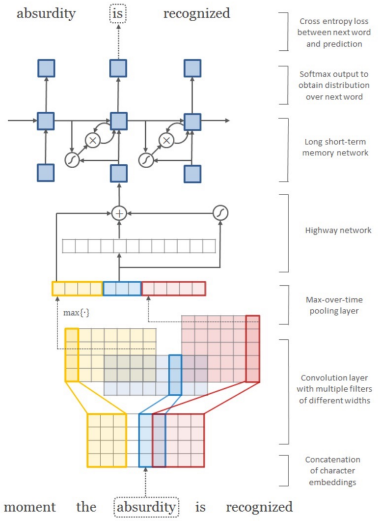
Source: Finding function in form: compositional character models for open vocabulary word representation, Ling et al. 2015

Compose character representations into word representations with LSTMs



Source: Finding function in form: compositional character models for open vocabulary word representation, Ling et al. 2015

Compose character representations into word representations with CNNs



Source: Character-aware neural language models, Kim et al. 2015

Character models actually work. Train them long enough, they generate words

increased	*John*	*Noahshire*	*phding*
reduced	Richard	Nottinghamshire	mixing
improved	George	Bucharest	modelling
expected	James	Saxony	styling
decreased	Robert	Johannesburg	blaming
targeted	Edward	Gloucestershire	christening

Table 2: Most-similar in-vocabular words under the C2W model; the two query words on the left are in the training vocabulary, those on the right are nonce (invented) words.

Source: Finding function in form: compositional character models for open vocabulary word representation, Ling et al. 2015

Character models actually work. Train them long enough, they generate words

anterest	hamburgo
artifactive	identimity
capacited	ipoteca
capitaling	nightmale
compensive	orience
dermitories	patholicism
despertator	pinguenas
dividement	sammitment
extremilated	tasteman
faxemary	understrumental
follect	wisholver

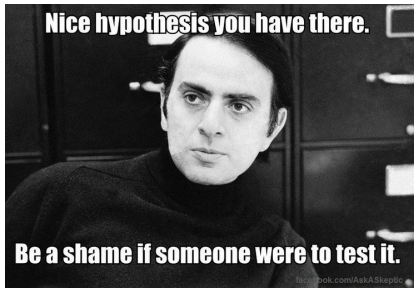
Wow, the disconversated vocabulations of their system are fantastics!

—Sharon Goldwater

How good are character-level NLP models?

tation algorithm. Ling et al. (2015b) indeed states that "[m]uch of the prior information regarding morphology, cognates and rare word translation among others, should be incorporated".

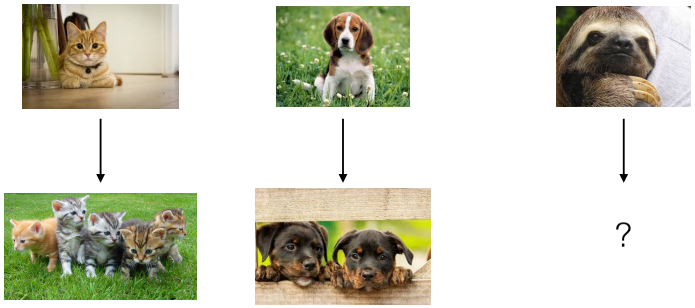
It however becomes unnecessary to consider these prior information, if we use a neural network, be it recurrent, convolution or their combination, directly on the unsegmented character sequence. The possibility of using a sequence of un-



Implied(?): character-level neural models learn everything they need to know about language.

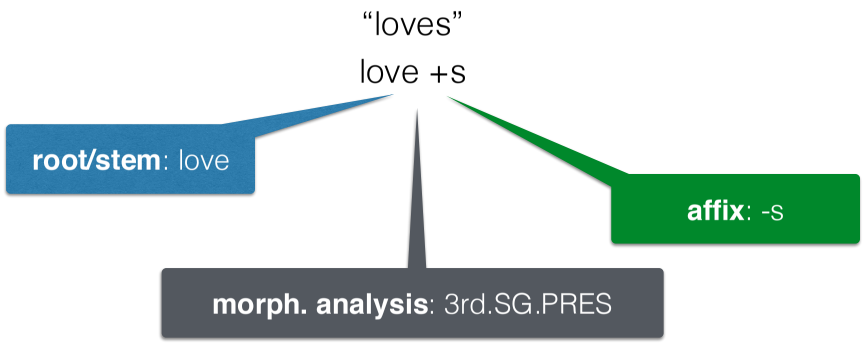
Word embeddings have obvious limitations

- Closed vocabulary assumption
- Cannot exploit functional relationships in learning



And **we** know a lot about linguistic structure

Morpheme: the smallest meaningful unit of language



The ratio of morphemes to words varies by language

Analytic languages
one morphemeper word

Vietnamese

Hai đứa bo? nhau là tại gia-đình thàng chông.
two individual leave each.other be because.of family guy husband.
'They divorced because of his family'

English

Synthetic languages
many morphemes per word

Turkish

West Greenlandic

Paasi-nngil-luinnar-para illa-juma-sutit.
understand-not-completely-1SG.SBJ.3SG.OBJ.IND come-want-2SG.PTCP
'I didn't understand at all that you wanted to come along.'

Morphology can change syntax or semantics of a word

"love" (VB)

Inflectional morphology

love (VB), love**s** (VB), lov**ing**(VB), lov**ed**(VB)

Derivational morphology

lover (NN), love**ly**(ADJ), lov**able**(ADJ)

Morphemes can represent one or more *features*

Agglutinative languages
one feature per morpheme

(Turkish)
oku-**r**-**sa**-**m**
read-AOR.COND.1SG
'If I read …'

Fusional languages
many features per morpheme

(English)
read-**s**
read-3SG.SG
'reads'

Words can have more than one stem

Affixation
one stem per word (English)

studying
study + ing

Compounding
many stems per word (German)

Rettungshubschraubernotlandeplatz
Rettung + s + **hubschrauber** + **not** + **lande**+ **platz**
rescue + LNK + helicopter + emergency + landing + place
'Rescue helicopter emergency landing pad'

Inflection is not limited to affixation

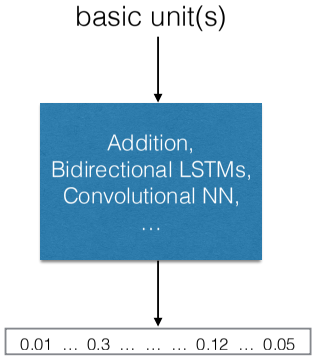
Base Modification	**drink**, **dr**a**nk**, **dr**u**nk**	(English)
Root & Pattern	**k**(a)**t**(a)**b**(a) write-PST.3SG.M *'he wrote'*	(Arabic)
Reduplication	ke**merah**~**merah**an red-ADJ *'reddish'*	(Indonesian)

There are many different ways to compute word representations from *subwords*

Basic Units of Representation

- Characters (Ling et al., 2015, Kim et al., 2016, Lee at al., 2016)
- Character n-grams (Sperr at al., 2013, Wieting et al., 2016, Bojanowski et al., 2016)
- Morphemes (Luong et al., 2013, Botha & Blunsom, 2014, Sennrich et al., 2016)
- Morphological analysis (Cotterel & Schütze, 2015, Kann & Schütze, 2016)

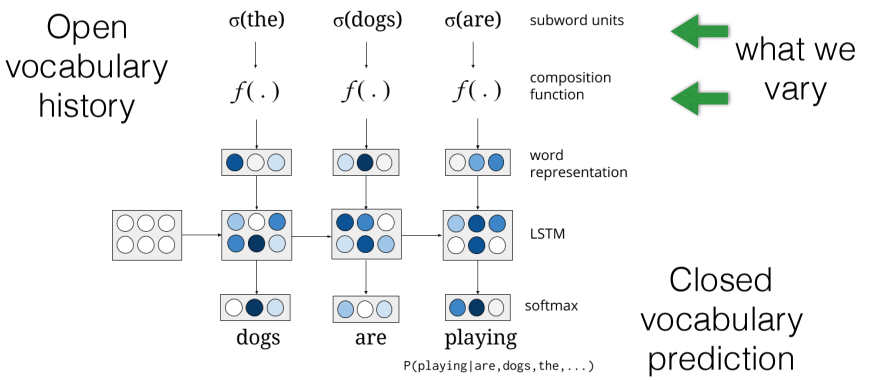
Compositional Function



We've revised morphology, so we have some questions about character models

- How do representations based on morphemes compared with those based on characters?
- What is the best way to compose subword representations?
- Do character-level models the same **predictive utility** as models with knowledge of morphology?
- How do different representations interact with languages of different morphological typologies?

Prediction problem: neural language modeling



Open vocabulary prediction is interesting, but our goal is to understand representations, not build a better neural LM.

Variable: Subword Unit

Unit	Examples
Morfessor	^want, s$
BPE	^w, ants$
char-trigram	^wa, wan, ant, nts, ts$
character	^, w, a, n, t, s, $
analysis	want+VB, +3rd, +SG, +Pres

Approximations to morphology

Annotated morphology

The last row is part of an oracle experiment: suppose you had an oracle that could tell you the true morphology. In this case, the oracle is a human annotator.

Variable: Composition Function

- Vector addition (except for characters)
- Bidirectional LSTMs
- Convolutional NN

Variable: Language Typology

Fusional (English)

read-s
read-3SG.SG
'reads'

Agglutinative (Turkish)

oku-r-sa-m
read-AOR.COND.1SG
'If I read …'

Root&Pattern (Arabic)

k(a)t(a)b(a)
write-PST.3SG.M
'he wrote'

Reduplication (Indonesian)

anak~anak
child-PL
'children'

Summary of perplexity: use bi-LSTMs over character trigrams

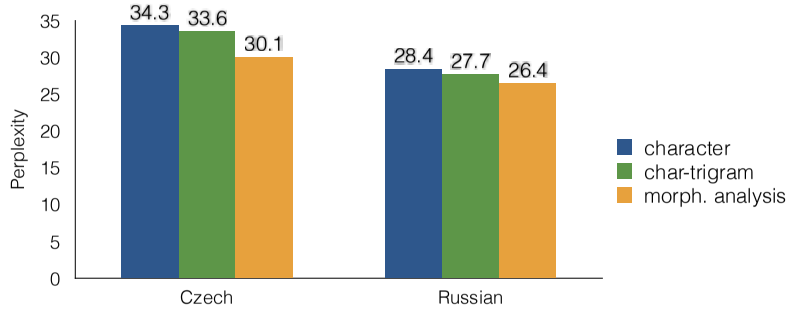
Language	word	character		char-trigrams		BPE		Morfessor		%imp
		bi-LSTM	CNN	add	bi-LSTM	add	bi-LSTM	add	bi-LSTM	
Czech	41.46	34.25	36.6	42.73	**33.59**	49.96	33.74	47.74	36.87	18.98
English	46.4	43.53	44.67	45.41	**42.97**	47.51	43.3	49.72	49.72	7.39
Russian	34.93	28.44	29.47	35.15	**27.72**	40.1	28.52	39.6	31.31	20.64
Finnish	24.21	20.05	20.29	24.89	**18.62**	26.77	19.08	27.79	22.45	23.09
Japanese	98.14	98.14	**91.63**	101.99	101.09	126.53	96.8	111.97	99.23	6.63
Turkish	66.97	54.46	55.07	**50.07**	54.23	59.49	57.32	62.2	62.7	25.24
Arabic	48.2	42.02	43.17	50.85	**39.87**	50.85	42.79	52.88	45.46	17.28
Hebrew	38.23	31.63	33.19	39.67	**30.4**	44.15	32.91	44.94	34.28	20.48
Indonesian	46.07	45.47	46.6	58.51	45.96	59.17	**43.37**	59.33	44.86	5.86
Malay	54.67	53.01	**50.56**	68.51	50.74	68.99	51.21	68.2	52.5	7.52

Still lots of work to do on unsupervised morphology…

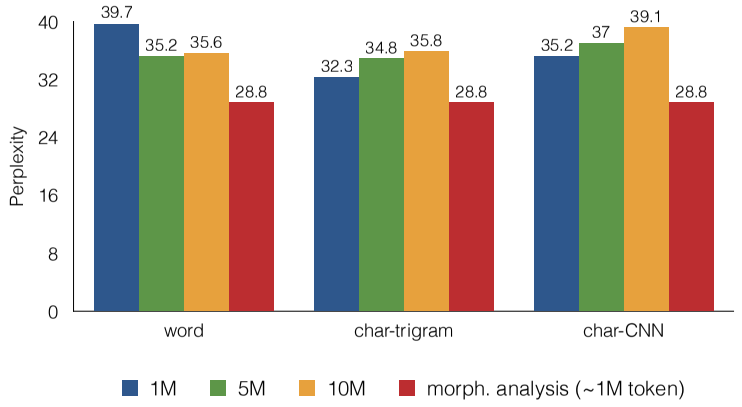
Do character-level models have the predictive utility of models with access to actual morphology? **NO**

no morphology → **actual morphology**

(^, r, e, a, d, s, $) ⟶ (read, VB, 3rd, SG, Present)



Can we close that gap by training character-level models on far more data? **NO**



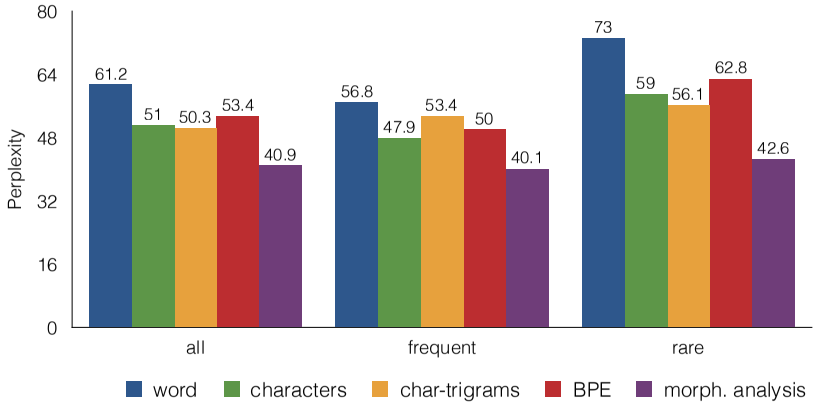
How do we know that the morphological annotations that make the difference?

- Measure **Targeted perplexity**: perplexity on specific subset of words in the test data
- Analyze perplexities when the **inflected words** of interest are in the most recent history: **nouns** and **verbs**

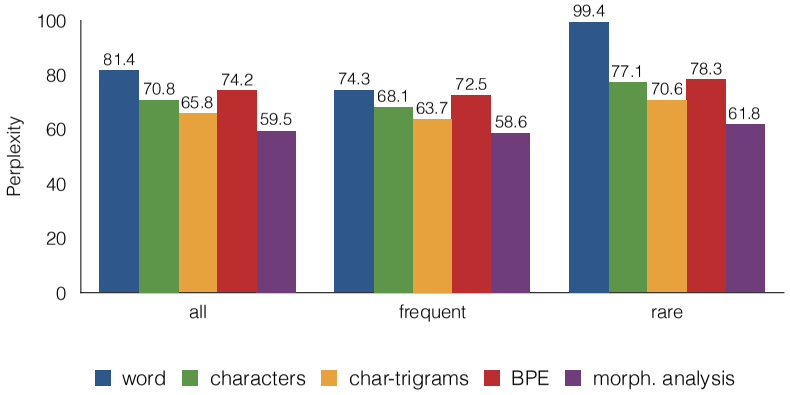
*Green **tea** or white **tea** ?*

*The sushi **is** great , and they **have** a great **selection** .*

Targeted perplexity of Czech nouns is lower when we use morphology



Targeted perplexity of Czech verbs is lower when we use morphology

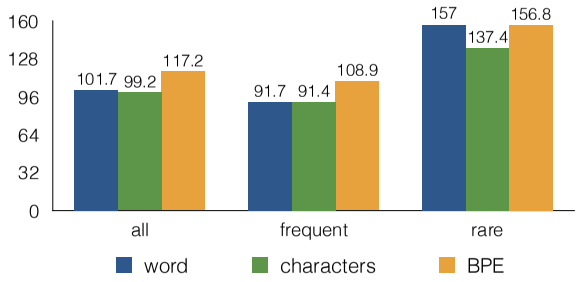


Character models are good at reduplication (no oracle, though)

Percentage of **full-reduplication** in the training data

Language	type-level (%)	token-level (%)
Indonesian	1.1%	2.6
Malay	1.3%	2.9

Targeted Perplexity



Different representations make different neighbors

Model	Frequent		Rare		Unknown	
	man	***including***	***unconditional***	***hydroplane***	***uploading***	***foodism***
word	person anyone children	like featuring include	nazi fairly joints	molybdenum your imperial	- - -	- - -
BPE bi-lstm	ii hill text	called **in**volve like	**un**intention**al** **un**generous **un**animous	emphasize heartbeat **hy**bridized	**up**beat **up**ris**ing** handl**ing**	vigilant**ism** pyrethrum pausanias
char trigrams bi-lstm	**ma**k vill cow	**includ**e **includ**es **in**dermin**ing**	**uncon**stitut**ional** **con**stitu**tional** **un**imolecular	selenocysteine guerrillas scrofula	drifted affected conflicted	tuaregs quft subjectiv**ism**
char bi-lstm	**may**r **man**y **may**	inclusion insularity include	relates **un**myelinated **unco**ordinated	**hydro**lyzed **hydr**aulics **hy**sterotomy	musagte mutualism mutualist	**for**mulas **for**mally fecal
char CNN	**mtn** **mann** nu**n**	**includ**e **includ**es ex**cluding**	**uncon**ventional **un**intentional **uncon**stitu**tional**	**hydro**xyproline **hydr**ate **hydr**angea	un**loading** **loading** **up**grading	**for**dham dada**ism** pop**ism**

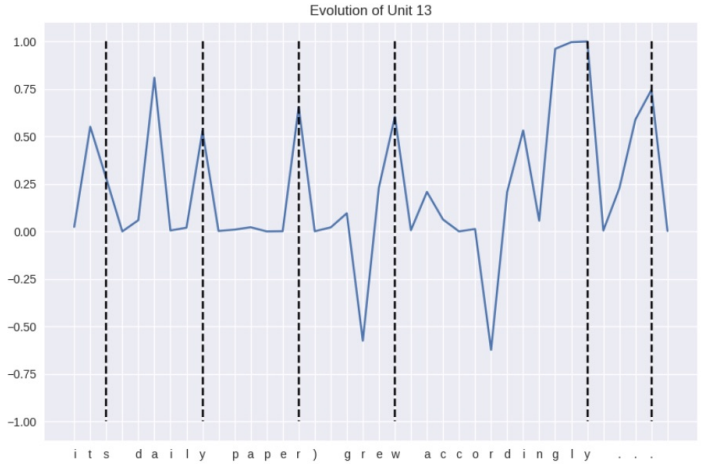
Different representations make different neighb…

Good at frequent words

Maybe they learn "word classes"?

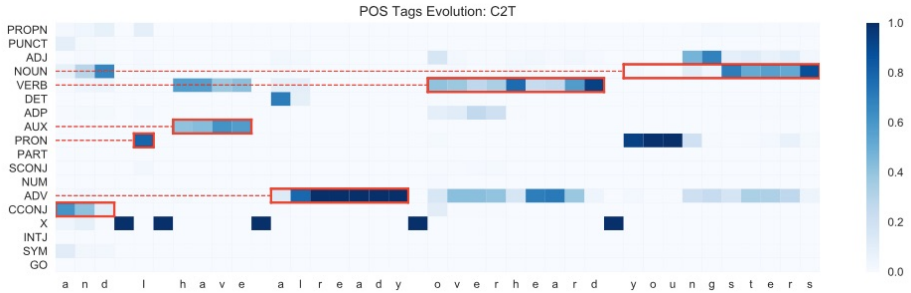
Model	Frequent		Rare		OOV	
	man	***including***	***unconditional***	***hydroplane***	***uploading***	***foodism***
word	person anyone children	like featuring include	nazi fairly joints	molybdenum your imperial	- - -	- - -
BPE bi-lstm	ii hill text	called **in**volve like	**un**intention**al** **un**generous **un**animous	emphasize heartbeat **hy**bridized	**up**beat **up**ris**ing** handl**ing**	vigilant**ism** pyrethrum pausanias
char trigrams bi-lstm	**ma**k vill cow	**includ**e **includ**es **in**dermin**ing**	**uncon**stitut**ional** **con**stitu**tional** **un**imolecular	selenocysteine guerrillas scrofula	drifted affected conflicted	tuaregs quft subjectiv**ism**
char bi-lstm	**may**r **man**y **may**	inclusion insularity include	relates **un**myelinated **unco**ordinated	**hydro**lyzed **hydr**aulics **hy**sterotomy	musagte mutualism mutualist	**for**mulas **for**mally fecal
char CNN	**mtn** **mann** nu**n**	**includ**e **includ**es ex**cluding**	**uncon**ventional **un**intentional **uncon**stitu**tional**	**hydro**xyproline **hydr**ate **hydr**angea	un**loading** **loading** **up**grading	**for**dham dada**ism** pop**ism**

Character NLMs learn word boundaries.



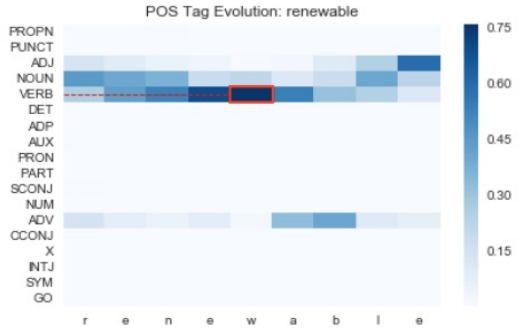
Source: Yova Kementchedjhieva, Morpho-syntactic awareness in a character-level language model 2017 Informatics M.Sc. thesis

…and memorize POS tags



Source: Yova Kementchedjhieva, Morpho-syntactic awareness in a character-level language model 2017 Informatics M.Sc. thesis

…and memorize POS tags



Source: Yova Kementchedjhieva, Morpho-syntactic awareness in a character-level language model 2017 Informatics M.Sc. thesis

What do we know about what NNs know about language?

- Still very little.
- Evidence suggests: nothing surprising. Lots of memorization, local generalization.
- NNs are great for simplicity of specification and end-to-end learning.
- But these things are not magic! We still don't have enough data, and these models could be better if they knew about morphology.
- But how do we do that?

What do NLMs learn about morphology?

- Character-level NLMs are great! Across typologies, but especially for agglutinative morphology.
- However, they **do not match** predictive accuracy of model with explicit knowledge of morphology (or POS).
- Qualitative analyses suggests that they learn **orthographic similarity** of affixes, and **forget meaning of root morphemes**.
- More generally, they appear to **memorize frequent subpatterns**.