CS6101: NLP with Deep Learning: Advanced Attention

Abhinav Kashyap,
Zining Zhang,
Nan Xiao,
Adam Goodge

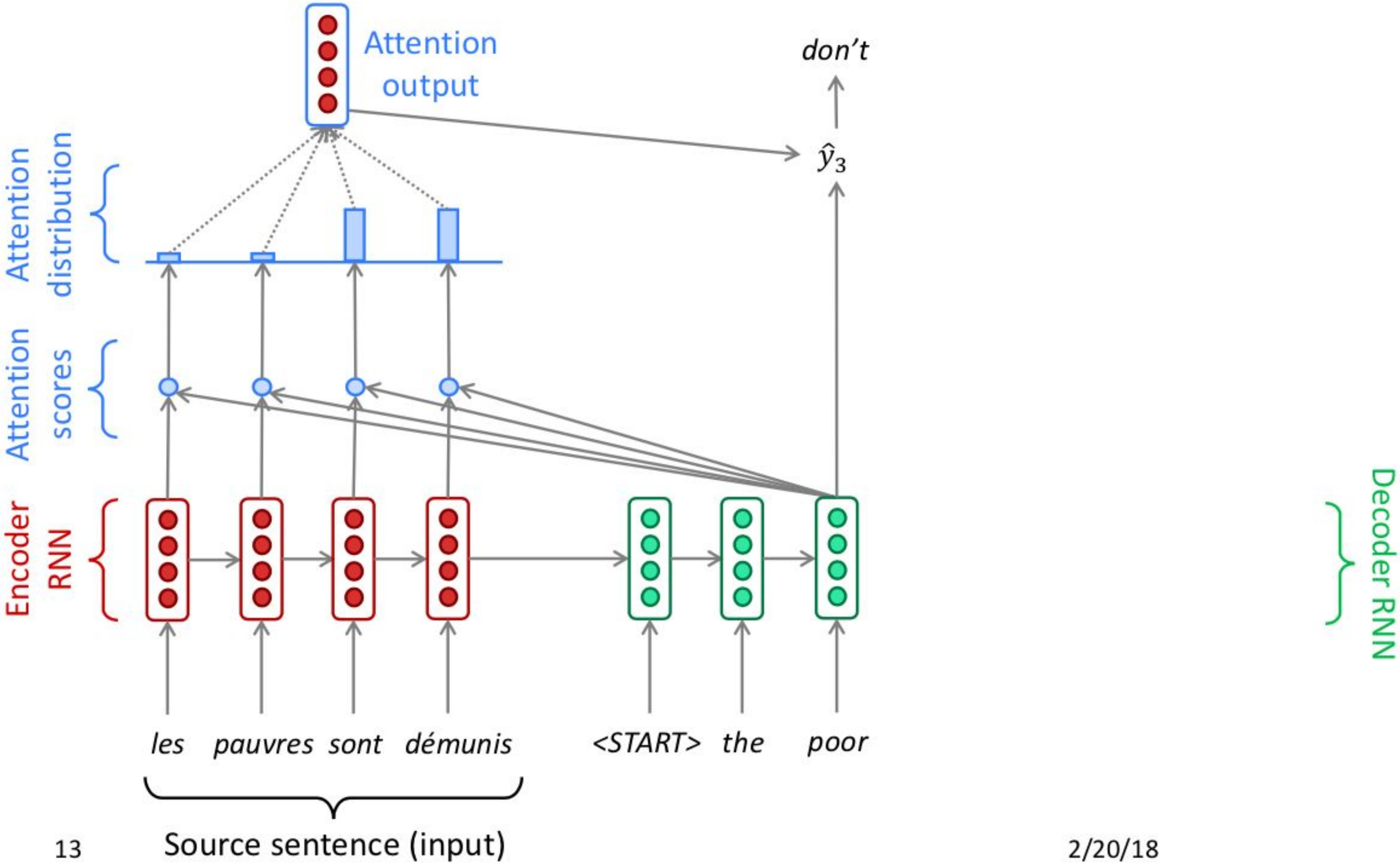
Attention Code Example

Abhinav Kashyap

Attention Application

Zining Zhang

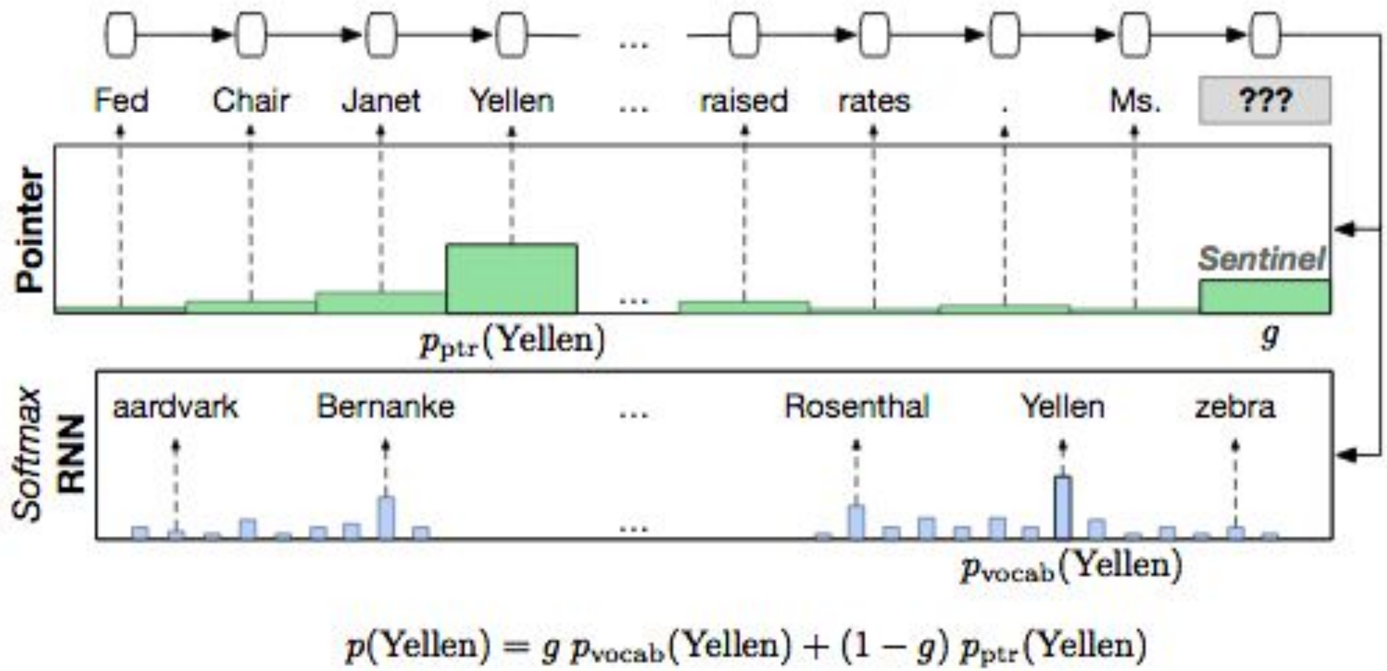
Seq2seq with attention



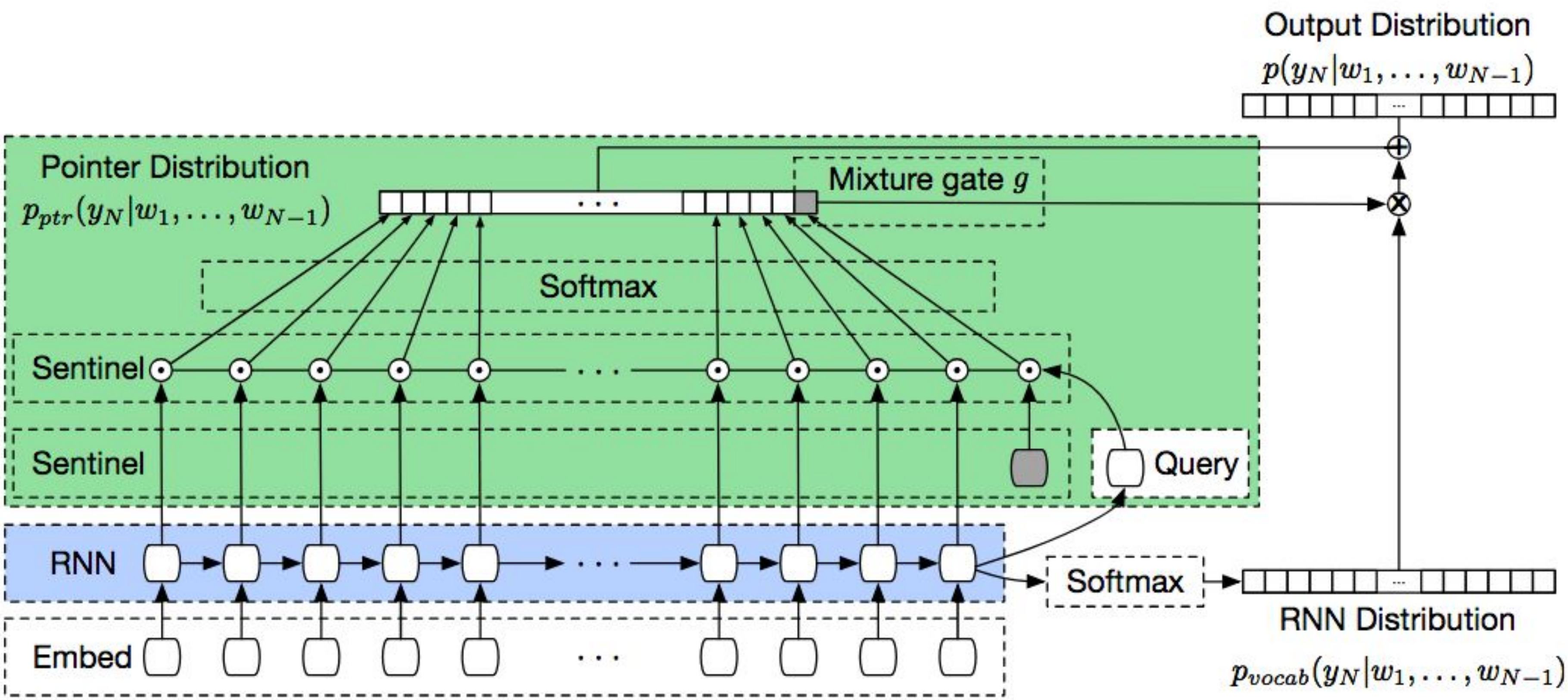
Attention is a general DL technique

- Not only for seq2seq
- More general definition:
  - Given a set of vectors: values
  - One vector as input: query
  - Attention: is a technique to get weighted sum on values based on the query
    - We sometimes say query ***attends*** to the values
- Example:
  - Decoder hidden state attends to encoder hidden states
- Score -> probability distribution -> weighted sum
- Variants:
  - Basic dot product attention: $$\boldsymbol{e}_i = \boldsymbol{s}^T \boldsymbol{h}_i \in \mathbb{R}$$
  - Multiplicative: $$\boldsymbol{e}_i = \boldsymbol{s}^T \boldsymbol{W} \boldsymbol{h}_i \in \mathbb{R}$$
  - Additive: $$\boldsymbol{e}_i = \boldsymbol{v}^T \tanh(\boldsymbol{W}_1 \boldsymbol{h}_i + \boldsymbol{W}_2 \boldsymbol{s}) \in \mathbb{R}$$

Attention Applications: Predict next word



Attention Applications: Predict next word- Pointer Sentinel



Model explained

$$q = \tanh(W h_{N-1} + b),$$

$$z_i = q^T h_i,$$

$$a = \text{softmax}(z),$$

$$p_{\text{ptr}}(w) = \sum_{i \in I(w,x)} a_i,$$

$$p(y_i|x_i) = g\, p_{\text{vocab}}(y_i|x_i) + (1-g)\, p_{\text{ptr}}(y_i|x_i).$$

Attention Applications: Summarization

- Long document:
  - Tony Blair has said he does not want to retire until he is 91 – as he unveiled plans to set up a ‘cadre’ of ex-leaders to advise governments around the world. The defiant 61-year-old former Prime Minister said he had ‘decades’ still in him and joked that he would ‘turn to drink’ if he ever stepped down from his multitude of global roles. He told Newsweek magazine that his latest ambition was to recruit former heads of government to go round the world to advise presidents and prime ministers on how to run their countries. In an interview with the magazine Newsweek Mr Blair said he did not want to retire until he was 91 years old Mr Blair said his latest ambition is to recruit former heads of government to advise presidents and prime ministers on how to run their countries Mr Blair said he himself had been ‘mentored’ by US president Bill Clinton when he took office in 1997. And he said he wanted to build up his organisations, such as his Faith Foundation, so they are ‘capable of changing global policy’. Last night, Tory MPs expressed horror at the prospect of Mr Blair remaining in public life for another 30 years. Andrew Bridgen said: ‘We all know weak Ed Miliband’s called on Tony to give his flailing campaign a boost, but the attention’s clearly gone to his head.’ (...)

- Summary:
  - The former Prime Minister claimed he has 'decades' of work left in him. Joked he would 'turn to drink' if he ever stepped down from global roles. Wants to recruit former government heads to advise current leaders. He was 'mentored' by US president Bill Clinton when he started in 1997.

Attention deficiency

Source:	die Teilnehmer der Proteste , die am Donnerstag um 6:30 AM morgens vor dem McDonald 's in der 40th Street und in der Madison Avenue begannen , forderten , dass **die Kassierer und Köche von Fast - Food - Restaurants** einen Mindestlohn von 15 US-Dollar die Stunde erhalten , was mehr als einer Verdoppelung des jetzigen Mindestlohns entspricht .
Reference:	Participants of the protest that began at 6.30 a.m. on Thursday near the McDonald's on 40th street and Madison Avenue demanded that cashiers and cooks of the fast food chain be paid at least 15 dollars/hour, i.e. more than double their present wages.
Candidate:	The protests that began on Thursday at 06:30 before the McDonald 's at McDonald 's at McDonald 's on 40th Street and Madison Avenue demanded that a minimum wage of 15 dollars would receive a minimum wage of 15 dollars per hour , equivalent to doubling the current minimum wage .
Source:	Special attention **is being paid** to the Tokyo **gubernatorial election** because it is perceived as a litmus test for the upcoming House of Councillors election , particularly in the metropolitan areas where **nonpartisan voters predominate** .
Reference:	都知事 選 の 結果 は , とくに 参院 選 に 向け 都市 部 に 多い 無党派 層 の 動向 を 占う もの と して 注目 さ れる 。
Candidate:	特に 首都 圏 で は , 特に 首都 圏 で は , 参院 選 の 試金石 と して 注目 さ れて いる 。

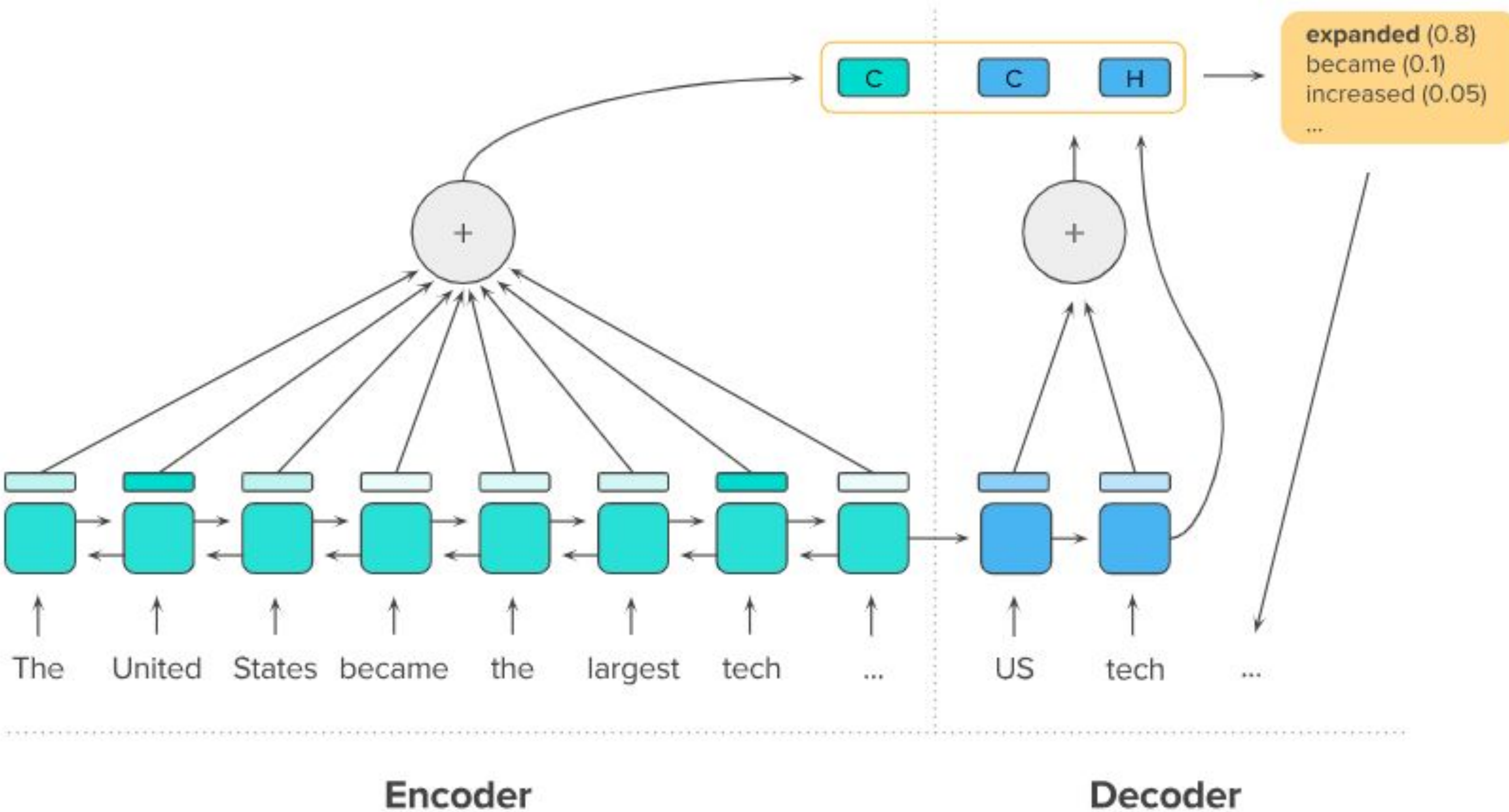
Using simple attention

- We could have repeated phrases
  - MT is used in single sentences

Intra-decoder attention for summarization

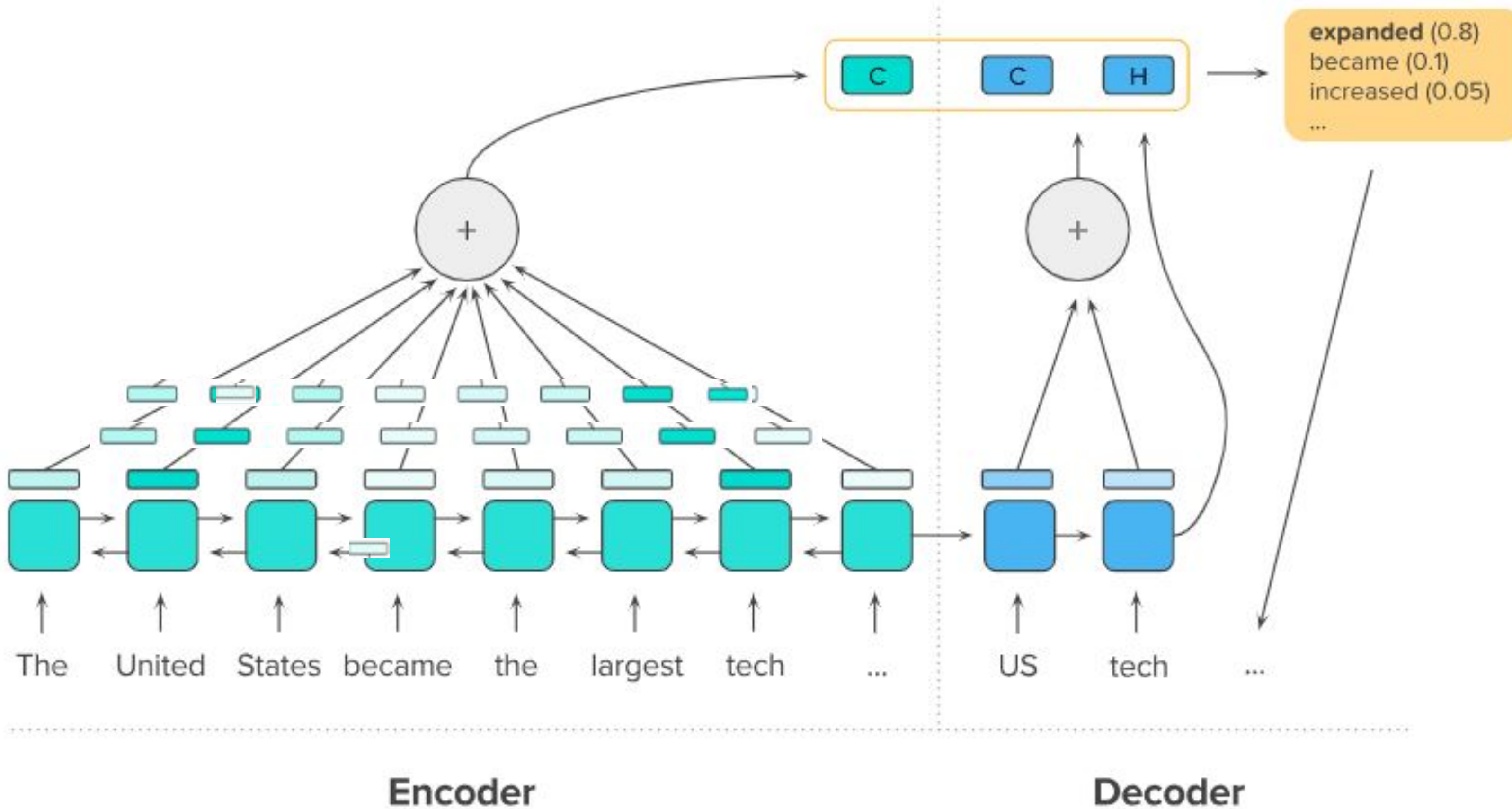
- A Deep Reinforced Model for Abstractive Summarization
    - Extractive vs abstractive
    - Attention during generation
    - Reinforcement learning(out of scope)

Model



expanded (0.8)
became (0.1)
increased (0.05)
...
C
C
H
+
+
The
United
States
became
the
largest
tech
...
US
tech
...
Encoder
Decoder

Model



C
C
H
expanded (0.8)
became (0.1)
increased (0.05)
...
The
United
States
became
the
largest
tech
...
US
tech
...
Encoder
Decoder

Details of the attention model: encoder

$$e_{ti} = f(h_t^d, h_i^e)$$

$$f(h_t^d, h_i^e) = {h_t^d}^T W_{\text{attn}}^e h_i^e$$

$$e'_{ti} = \begin{cases} exp(e_{ti}) & \text{if } t = 1 \\ \frac{exp(e_{ti})}{\sum_{j=1}^{t-1} \exp(e_{ji})} & \text{otherwise} \end{cases}$$

Details of the attention model: encoder

- score->score'->probability->context(weighted sum)

$$\boxed{\alpha_{ti}^{e} = \frac{e'_{ti}}{\sum_{j=1}^{n} e'_{tj}}}$$

$$c_t^e = \sum_{i=1}^{n} \alpha_{ti}^{e} h_i^e$$

Details of the attention model: decoder

$$e^d_{tt'} = {h^d_t}^T W^d_{\text{attn}} h^d_{t'}$$

$$\alpha^d_{tt'} = \frac{exp(e^d_{tt'})}{\sum_{j=1}^{t-1} exp(e^d_{tj})}$$

$$c^d_t = \sum_{j=1}^{t-1} \alpha^d_{tj} h^d_j$$

Details of attention mode: prediction

$$p(u_t = 1) = \sigma(W_u[h_t^d \| c_t^e \| c_t^d] + b_u)$$

$$p(y_t | u_t = 0) = \text{softmax}(W_{\text{out}}[h_t^d \| c_t^e \| c_t^d] + b_{\text{out}})$$
$$p(y_t = x_i | u_t = 1) = \alpha_{ti}^e$$

$$p(y_t) = p(u_t = 1)p(y_t | u_t = 1) + p(u_t = 0)p(y_t | u_t = 0)$$

Summarization result

cia documents reveal iot-specific televisions can be used to secretly record conversations .
cybercriminals who initiated the attack managed to commandeer a large number of internet-connected devices in current use .
cia documents revealed that microwave ovens can spy on you - maybe if you personally don't suffer the consequences of the sub-par security of the iot .

Internet of Things (IoT) security breaches have been dominating the headlines lately . WikiLeaks's trove of CIA documents revealed that internet-connected televisions can be used to secretly record conversations . Trump's advisor Kellyanne Conway believes that microwave ovens can spy on you - maybe she was referring to microwave cameras which indeed can be used for surveillance . And don't delude yourself that you are immune to IoT attacks , with 96 % of security professionals responding to a new survey expecting an increase in IoT breaches this year . Even if you personally don't suffer the consequences of the sub-par security of the IoT , your connected gadgets may well be unwittingly cooperating with criminals . Last October , Internet service provider Dyn came under an attack that disrupted access to popular websites . The cybercriminals who initiated the attack managed to commandeer a large number of internet-connected devices (mostly DVRs and cameras) to serve as their helpers . As a result , cybersecurity expert Bruce Schneier has called for government regulation of the IoT , concluding that both IoT manufacturers and their customers don't care about the security of the 8.4 billion internet-connected devices in current use . Whether because of government regulation or good old-fashioned self-interest , we can expect increased investment in IoT security technologies . In its recently-released TechRadar report for security and risk professionals , Forrester Research discusses the outlook for the 13 most relevant and important IoT security technologies , warning that " there is no single , magic security bullet that can easily fix all IoT security issues . " Based on Forrester's analysis , here's my list of the 6 hottest technologies for IoT security : IoT network security : Protecting and securing the network connecting IoT devices to back-end systems on the internet . IoT network security is a bit more challenging than traditional network security because there is a wider range of communication protocols , standards , and device capabilities , all of which pose significant issues and increased complexity . Key capabilities include traditional endpoint security features such as antivirus and antimalware as well as other features such as firewalls and intrusion prevention and detection systems . Sample vendors : Bayshore Networks , Cisco , Darktrace , and Senrio . IoT authentication : Providing the ability for users to authenticate an IoT device , including managing multiple users of a single device (such as a connected car) , ranging from simple static password/pins to more robust authentication mechanisms such as two-factor

Summarization result

The bottleneck is no longer access to information; now it's our ability to keep up.
AI can be trained on a variety of different types of texts and summary lengths.
A model that can generate long, coherent, and meaningful summaries remains an open research problem.

The last few decades have witnessed a fundamental change in the challenge of taking in new information. The bottleneck is no longer access to information; now it's our ability to keep up. We all have to read more and more to keep up-to-date with our jobs, the news, and social media. We've looked at how AI can improve people's work by helping with this information deluge and one potential answer is to have algorithms automatically summarize longer texts. Training a model that can generate long, coherent, and meaningful summaries remains an open research problem. In fact, generating any kind of longer text is hard for even the most advanced deep learning algorithms. In order to make summarization successful, we introduce two separate improvements: a more contextual word generation model and a new way of training summarization models via reinforcement learning (RL). The combination of the two training methods enables the system to create relevant and highly readable multi-sentence summaries of long text, such as news articles, significantly improving on previous results. Our algorithm can be trained on a variety of different types of texts and summary lengths. In this blog post, we present the main contributions of our model and an overview of the natural language challenges specific to text summarization.

References

- A deep reinforced model for abstractive summarization, 2017
- Temporal Attention Model for Neural Machine Translation,2016
- Pointer Sentinel Mixture Models, 2016

Scale up NMT

Nan Xiao

Preview

- Hopefully, you can see how useful and versatile attention is

- Next lecture we will go even further and cover a model that *only* has attention (The Transformer)

- But, for now, we will cover some tips and tricks to actually **scale up** machine translation.

What could be the problems?

At its core, NMT is a single deep neural network that is trained end-to-end with several advantages such as simplicity and generalization.

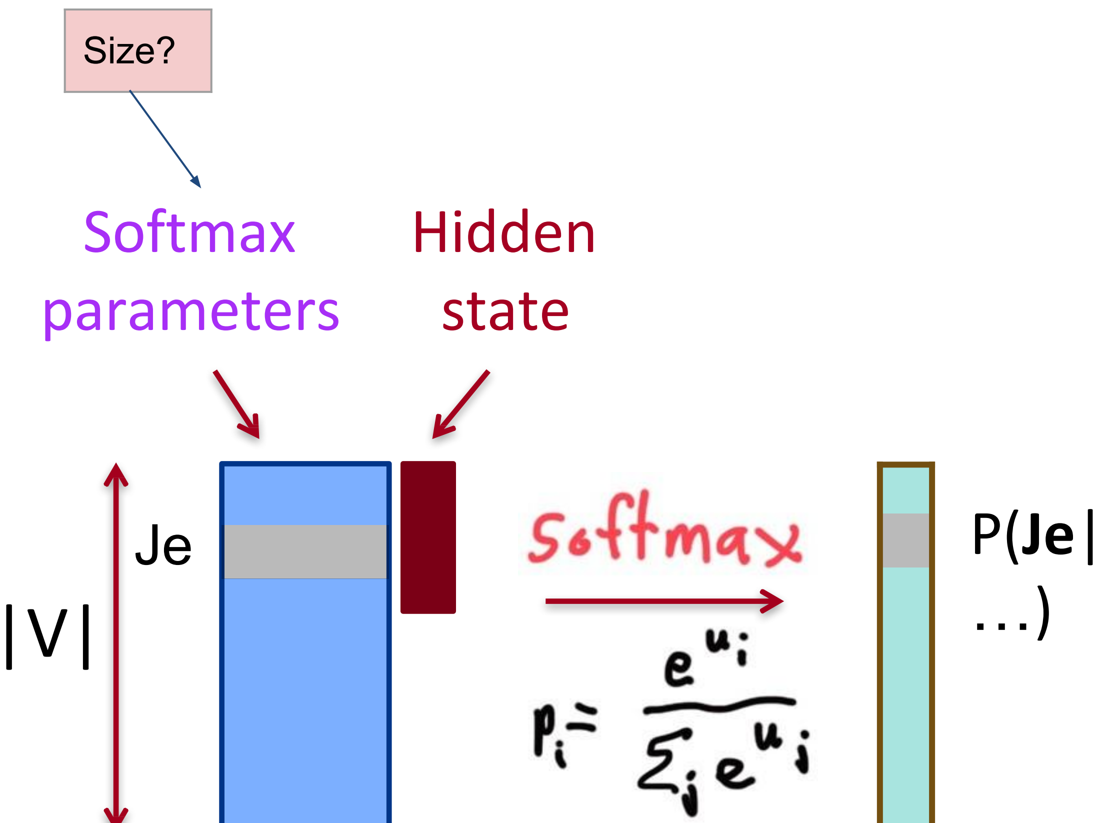
Extending NMT to more languages

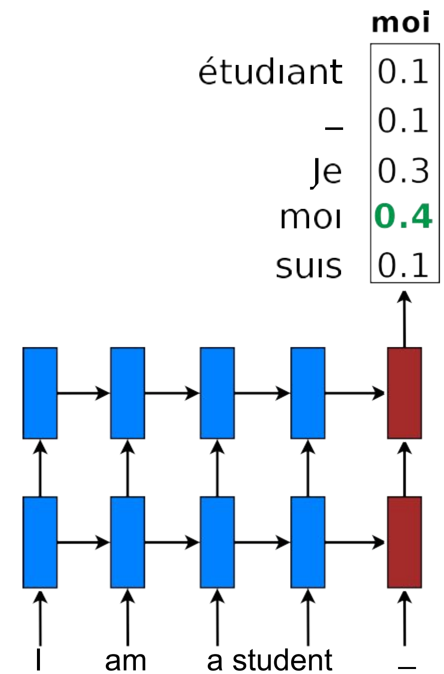
- “Copy” mechanisms are **not sufficient**.
  - Transliteration: Christopher ↦ Kryštof
  - Multi-word alignment: Solar system ↦ Sonnensystem

- Need to handle **large, open vocabulary**
  - Rich morphology:
    - nejneobhospodařovávatelnějšímu - Czech = “to the worst farmable one”
    - Donaudampfschiffahrtsgesellschaftskapitän – German = Danube steamship company captain
  - Informal spelling: goooooood morning !!!!!

Need to be able to operate at sub-word levels!

Dealing with a large output vocabulary in MT++

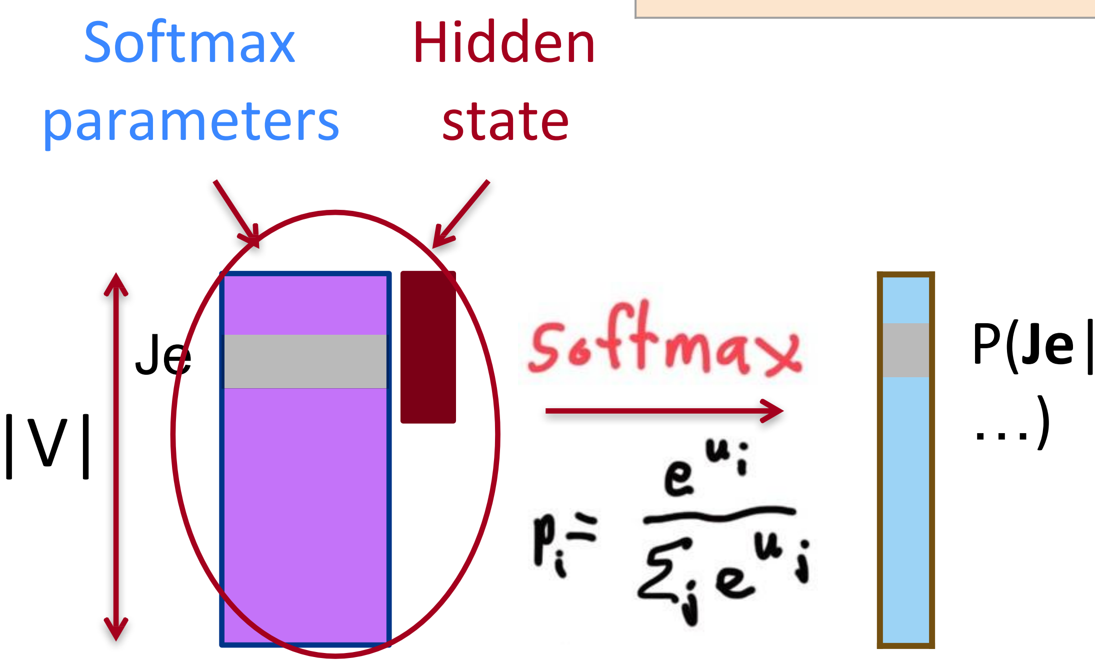


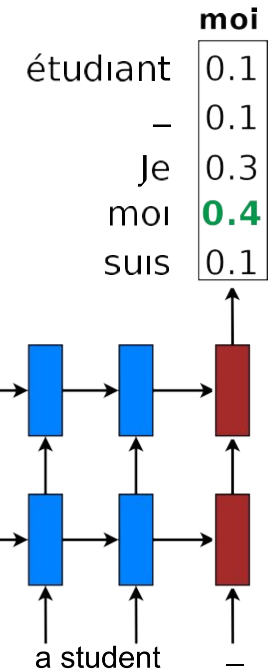


The word generation problem

- Word generation problem

size(vocabulary) x size(hidden state)



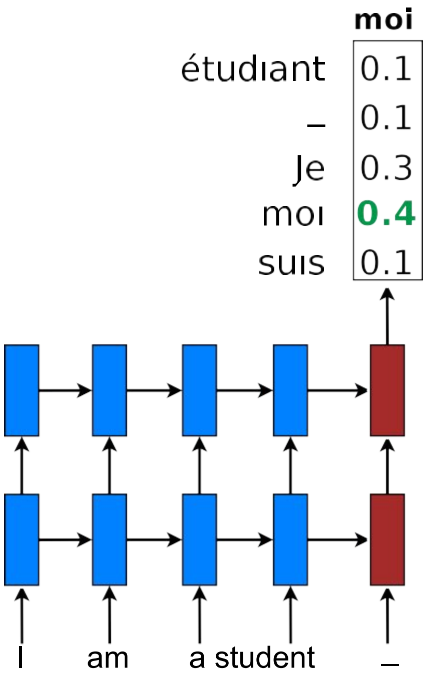


Softmax computation is expensive.

Just use smaller vocabulary?

The word generation problem

- Word generation problem
  - If vocabs are modest, e.g., 50K



The ecotax portico in Pont-de-Buis Le portique écotaxe de Pont-de-Buis

The <unk> portico in <unk> Le <unk> <unk> de <unk>

A usual practice is to construct a target vocabulary of the K most frequent words (a socalled shortlist), where K is often in the range of 30k (Bahdanau et al., 2015) to 80k (Sutskever et al., 2014). Any word not included in this vocabulary is mapped to a special token representing an unknown word [UNK].

This approach works well when there are only a few unknown words in the target sentence

First thought: scale the softmax

How to do softmax without doing so much computation?

- Lots of ideas from the neural LM literature!
- *Hierarchical models*: tree-structured vocabulary
  - [Morin & Bengio, AISTATS'05], [Mnih & Hinton, NIPS'09].
  - Complex, sensitive to tree structures.

Use CS trick to put it into tree structure to reduce the computation

- *Noise-contrastive estimation*: binary classification
  - [Mnih & Teh, ICML'12], [Vaswani et al., EMNLP'13].
  - Different noise samples per training example.*

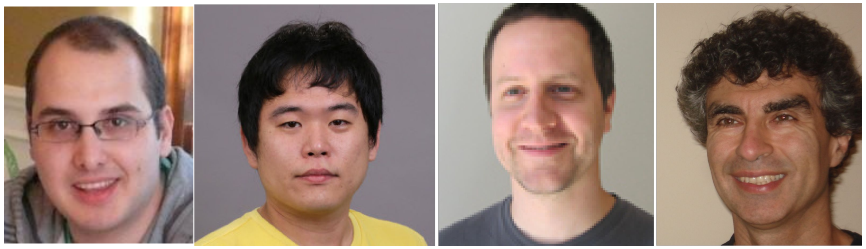
NCE: The basic idea is to convert **a multinomial classification problem** (as it is the problem of predicting the next word) **to a binary classification problem**. That is, instead of using softmax to estimate a true probability distribution of the output word, a binary logistic regression (binary classification) is used instead.

Avoid computation like used in word2vec

Not GPU-friendly

*We'll mention a simple fix for this!

Large-vocab NMT



Problem: training complexity as well as decoding complexity increase proportionally to the number of target words

- GPU-friendly.
- *Training*: a subset of the vocabulary at a time.
- *Testing*: smart on the set of possible translations.

Propose: a method based on importance sampling that allows us to use a very large target vocabulary without increasing training complexity

Fast at both train & test time.

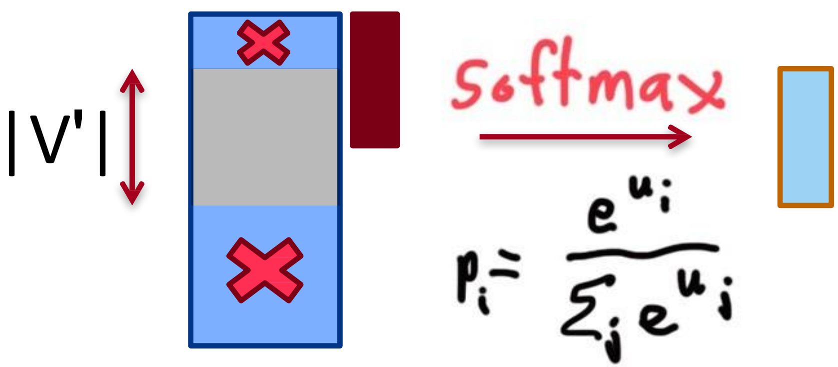
Sébastien Jean, Kyunghyun Cho, Roland Memisevic, Yoshua Bengio. ***On Using Very Large Target Vocabulary for Neural Machine Translation****. ACL'15.*

Ref: Denny Britz's notes - dennybritz/deeplearning-papernotes

Training

- Each time train on a smaller vocab V' $\ll$ V

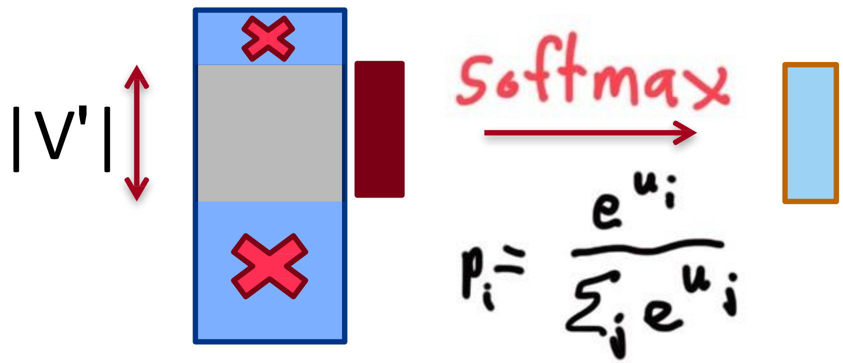
If you take a slice, most rare words won't be there



How do we select V'?

Training

- Each time train on a smaller vocab V' ≪ V



Computing partition function for softmax is the bottleneck. Use sampling-based approach.

- Partition training data in subsets:
  - Each subset has τ distinct target words, $|V'| = \tau$.

Training – *Segment data*

- Sequentially select examples: |V'| = 5.

she loves cats he likes
dogs

cats have tails
dogs have tails
dogs chase cats
she loves dogs
cats hate dogs

V' = {she, loves, cats, he, likes}

Training – *Segment data*

- Sequentially select examples: |V'| = 5.

she loves cats
he likes dogs
cats have tails
dogs have tails
dogs chase cats
she loves dogs
cats hate dogs

V' = {cats, have, tails, dogs, chase}

Training – *Segment data*

- Sequentially select examples: |V'| = 5.

she loves cats
he likes dogs
cats have tails
dogs have tails
dogs chase cats
she loves dogs
cats hate dogs

V' = {she, loves, dogs, cats, hate}

- *Practice*: $|V| = 500K$, $|V'| = 30K$ or $50K$.

We want to be fast in the Test time, how can we do that?

Testing – *Select candidate words*

In test time we want to use a much smaller vocabulary

- **K** most frequent words: unigram prob.

de,
,
la
.
et
des
les
...

Common functional words we always want to have in Softmax

Testing – *Select candidate words*

- K most frequent words: unigram prob.
- Candidate target words
  - K' choices per source word. K' = 3.

de,
,
la
.
et
des
les
...

elle
celle
ceci

She

aime
amour
aimer

loves

chats
chat
félin

cats

Training is handled with importance sampling. Decoding is handled with source-based candidate list.

Testing – *Select candidate words*

K'

Likely translation of each source word

elle
celle
ceci

She

aime
amour
aimer

loves

chats
chat
félin

cats

+

K

de,
,
la
.
et
des
les
...

Softmax only runs on Candidate list
-> reduce Softmax computation

= Candidate list

Issue: Candidate list is depended on source sentence, so it must be re-computed for each sentence.

- Produce translations within the candidate list
- *Practice*: $K' = 10$ or 20, $K = 15k, 30k$, or $50k$.

Reshuffling the dataset also results in a significant performance bump, but this operation is expensive.

More on large-vocab techniques

One way to understand BlackOut is to view it as an extension of the DropOut strategy to the output layer, wherein we use a discriminative training loss and a weighted sampling
scheme.

- "BlackOut: Speeding up Recurrent Neural Network Language Models with very Large Vocabularies" – [Ji, Vishwanathan, Satish, Anderson, Dubey, ICLR'16].
  - Good survey over many techniques.

- "Simple, Fast Noise Contrastive Estimation for Large RNN Vocabularies" – [Zoph, Vaswani, May, Knight, NAACL'16].
  - Use the same samples per minibatch. GPU efficient.

In normal NCE a dense matrix multiplication cannot be done. The reason is that the noise samples generated per training example will be different.

BUT
Scaling softmax is insufficient:
- new names, number…
- Theoretically we want to deal with infinite vocab in Test time

Sub-word NMT: two trends

From word level to sub-word level

- Same seq2seq architecture:
  - Use smaller units.
  - [Sennrich, Haddow, Birch, ACL'16a], [Chung, Cho, Bengio, ACL'16].

Improving Neural Machine Translation Models with Monolingual Data
A Character-Level Decoder without Explicit Segmentation for Neural Machine Translation

- Hybrid architectures:
  - RNN for *words* + something else for *characters*.
  - [Costa-Jussà & Fonollosa, ACL'16], [Luong & Manning, ACL'16].

Continuous Space Language Models for the IWSLT 2006 Task
Stanford Neural Machine Translation Systems for Spoken Language Domains

Byte Pair Encoding

- A compression algorithm:
  - Most frequent byte pair ↦ a new byte.

Replace bytes with character ngrams

making the NMT model
capable of open-vocabulary translation by
encoding rare and unknown words as sequences
of subword units

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

Byte Pair Encoding

Based on the intuition that various word classes are translatable via smaller units than words

- A word segmentation algorithm:
  - Start with a vocabulary of characters.
  - Most frequent ngram pairs ↦ a new ngram.

Byte Pair Encoding (BPE) (Gage, 1994) is a simple data compression technique that iteratively replaces the most frequent pair of bytes in a sequence with a single, unused byte.
Instead of merging frequent pairs of bytes, we merge characters or character sequences.

Algorithm 1 Learn BPE operations

```
import re, collections

def get_stats(vocab):
  pairs = collections.defaultdict(int)
  for word, freq in vocab.items():
    symbols = word.split()
    for i in range(len(symbols)-1):
      pairs[symbols[i],symbols[i+1]] += freq
  return pairs

def merge_vocab(pair, v_in):
  v_out = {}
  bigram = re.escape(' '.join(pair))
  p = re.compile(r'(?<!\S)' + bigram + r'(?!\S)')
  for word in v_in:
    w_out = p.sub(''.join(pair), word)
    v_out[w_out] = v_in[word]
  return v_out

vocab = {'l o w </w>' : 5, 'l o w e r </w>' : 2,
         'n e w e s t </w>':6, 'w i d e s t </w>':3}
num_merges = 10
for i in range(num_merges):
  pairs = get_stats(vocab)
  best = max(pairs, key=pairs.get)
  vocab = merge_vocab(best, vocab)
  print(best)
```

r · → r·
l o → lo
lo w → low
e r· → er·

Byte Pair Encoding

- A word segmentation algorithm:
  - Start with a vocabulary of characters.
  - Most frequent ngram pairs ↦ 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

Start with all characters in vocab

(Example from Sennrich)

Byte Pair Encoding

- A word segmentation algorithm:
  - Start with a vocabulary of characters.
  - Most frequent ngram pairs ↦ a new ngram.

Dictionary

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

Vocabulary

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

Add a pair (e, s) with freq 9

(Example from Sennrich)

Byte Pair Encoding

- A word segmentation algorithm:
  - Start with a vocabulary of characters.
  - Most frequent ngram pairs ↦ a new ngram.

Dictionary

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

Vocabulary

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

Add a pair (es, t) with freq 9

(Example from Sennrich)

Byte Pair Encoding

- A word segmentation algorithm:
  - Start with a vocabulary of characters.
  - Most frequent ngram pairs ↦ a new ngram.

Dictionary

5	**lo** w
2	**lo** w e r
6	n e w est
3	w i d est

Vocabulary

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

Add a pair (l, o) with freq 7

(Example from Sennrich)

Byte Pair Encoding

The main contribution of this paper is that we show that neural machine translation systems are capable of open-vocabulary translation by representing rare and unseen words as a sequence of subword units.
This is both simpler and more effective than using a back-off translation model.

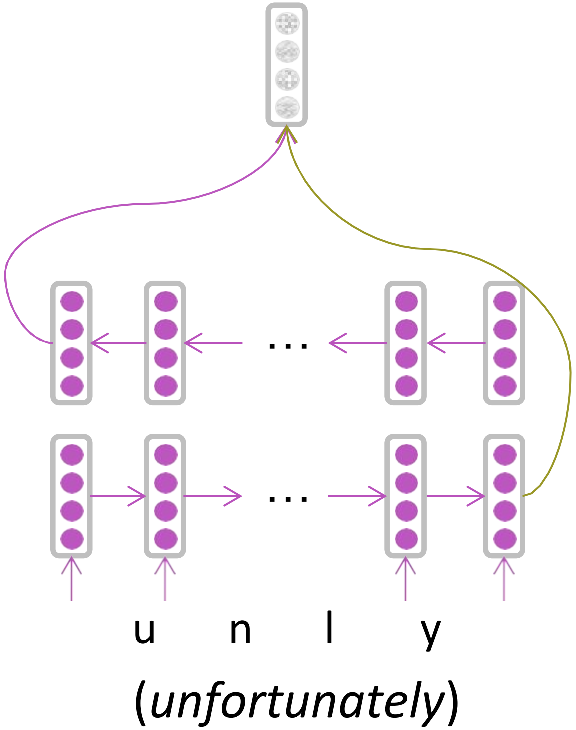
- A word segmentation algorithm:
  - Start with a vocabulary of characters.
  - Most frequent ngram pairs ↦ a new ngram.
- Automatically decide vocabs for NMT

choice of vocabulary size is somewhat arbitrary, and mainly motivated by comparison to prior work

Top places in WMT 2016!

https://github.com/rsennrich/nematus

Character-based LSTM

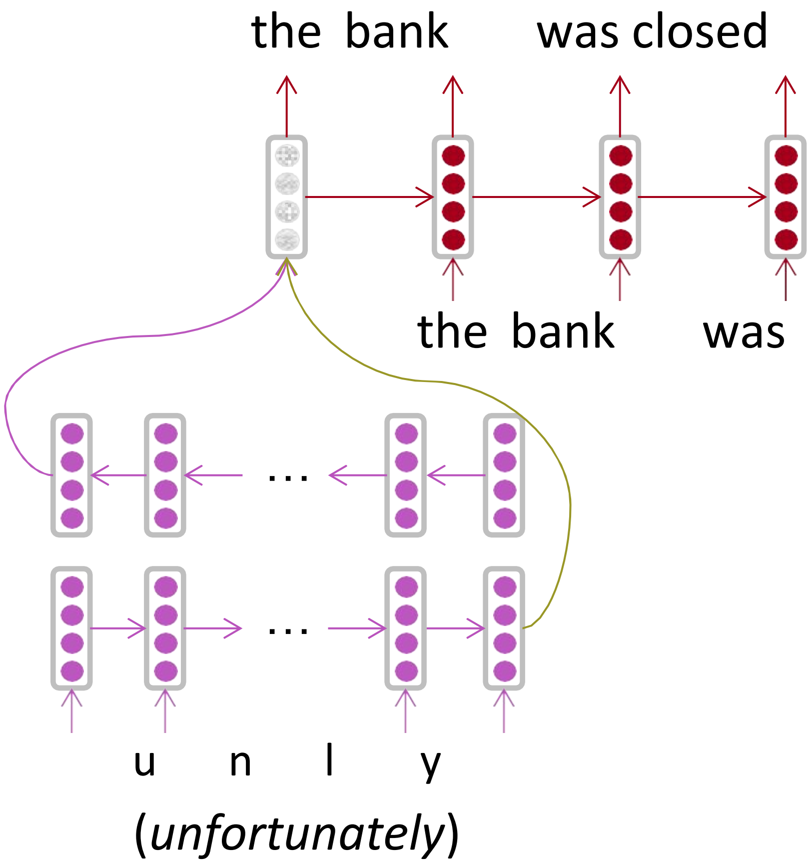


Benefits over traditional baselines are particularly pronounced in morphologically rich languages (e.g., Turkish)

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

a model for constructing vector representations of words by composing characters using bidirectional LSTMs

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.*

GitHub

Hybrid NMT


- A *best-of-both-worlds* architecture:
  - Translate mostly at the word level
  - Only go to the character level when needed.

To deal with open vocabulary NMT

- More than 2 BLEU improvement over a copy mechanism.

Code, data & models

The twofold advantage of such a hybrid approach is that it is much faster and easier to train than character-based ones; at the same time, it never produces unknown words as in the case of word-based models

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

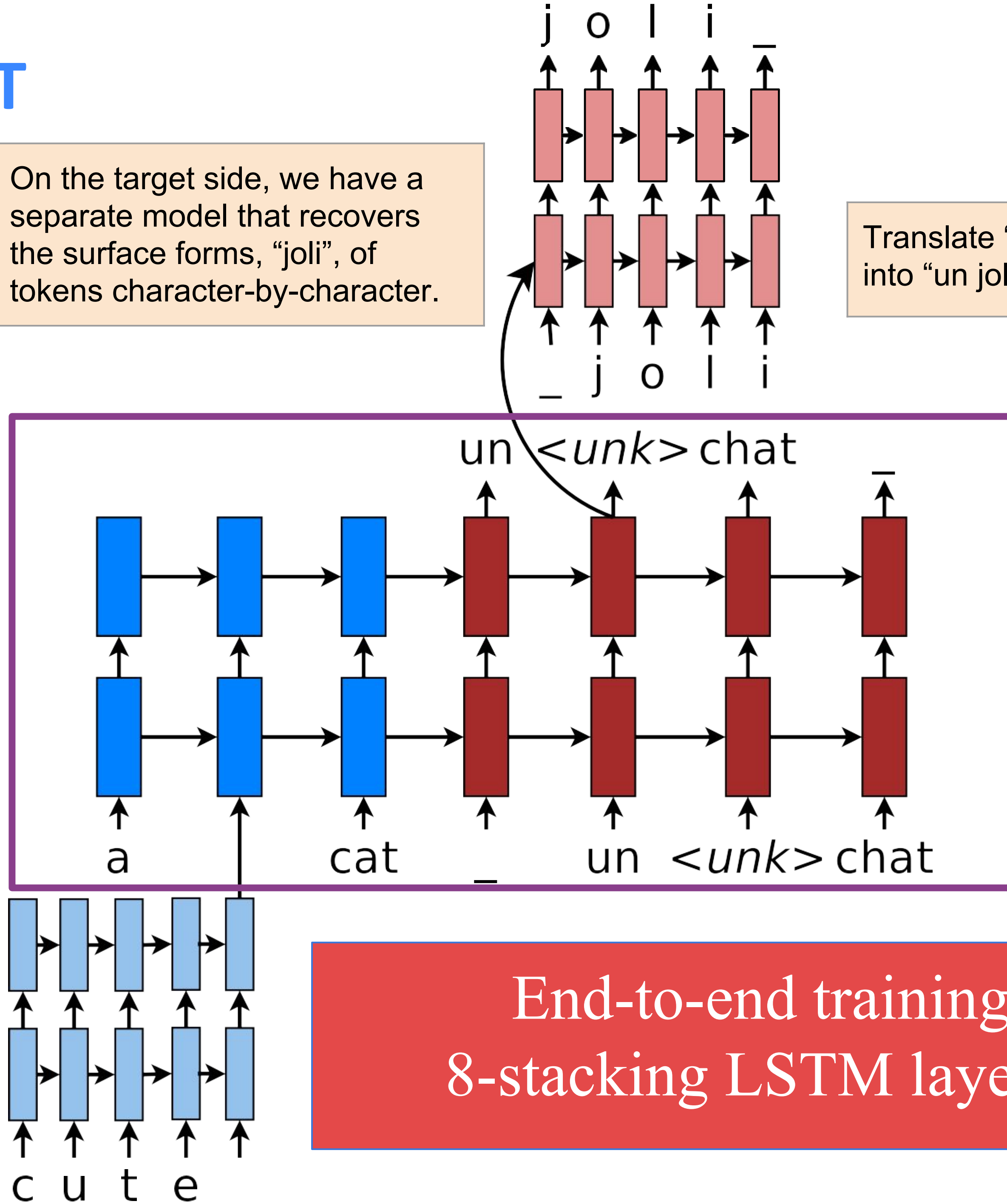
Previous work: Effective Approaches to Attention-based Neural Machine Translation

Hybrid NMT

On the target side, we have a separate model that recovers the surface forms, “joli”, of tokens character-by-character.

Translate “a cute cat” into “un joli chat”

Word-level
(4 layers)



On the source side, representations for rare words, “cute”, are computed on-the-fly using a deep recurrent neural network that operates at the character level.

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

2-stage Decoding

- Word-level beam search

The core of hybrid NMT is a deep LSTM encoder-decoder that translates at the word level

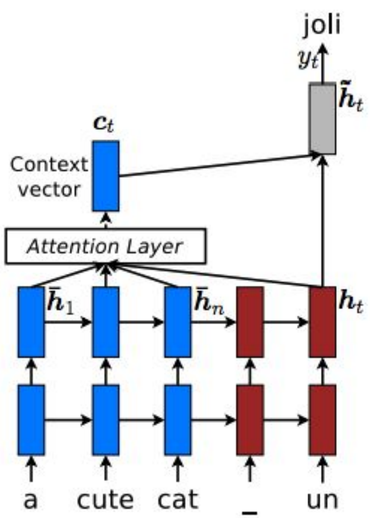
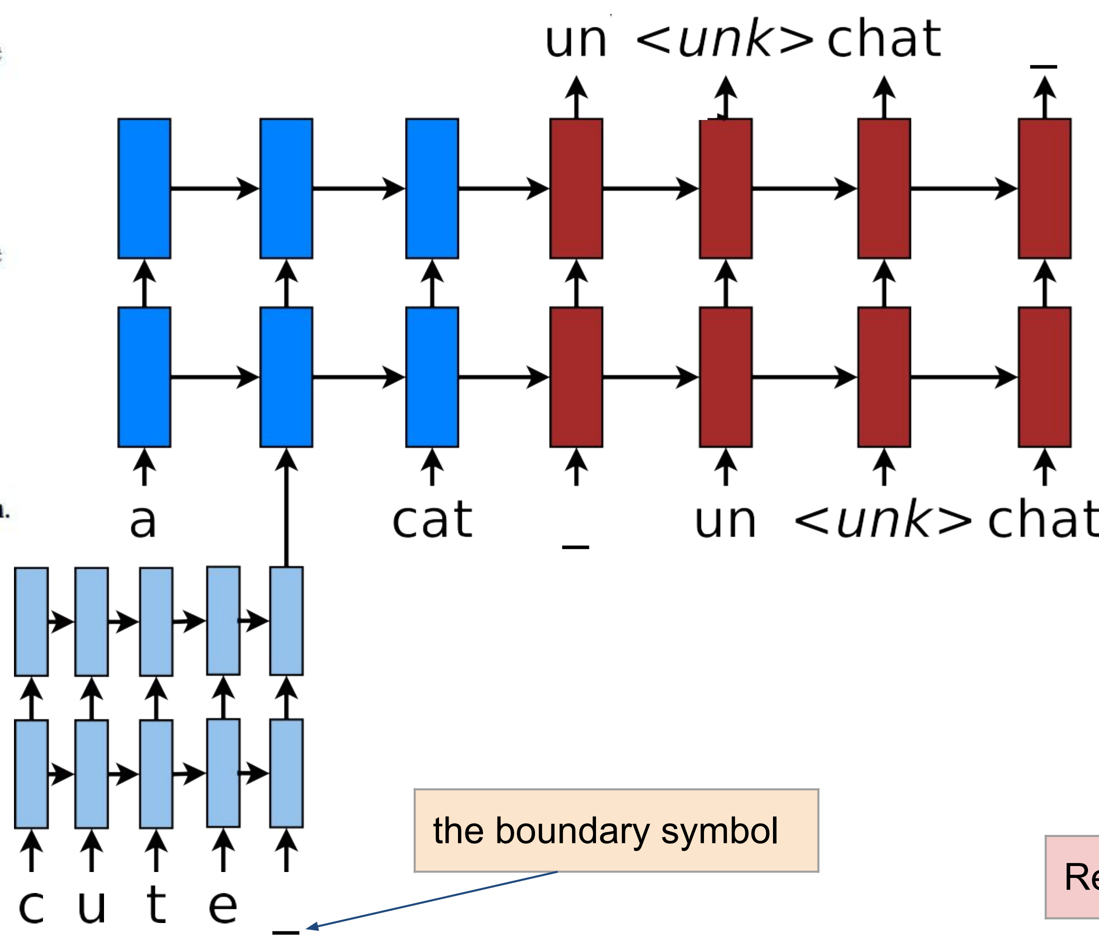


Figure 2: **Attention mechanism.**

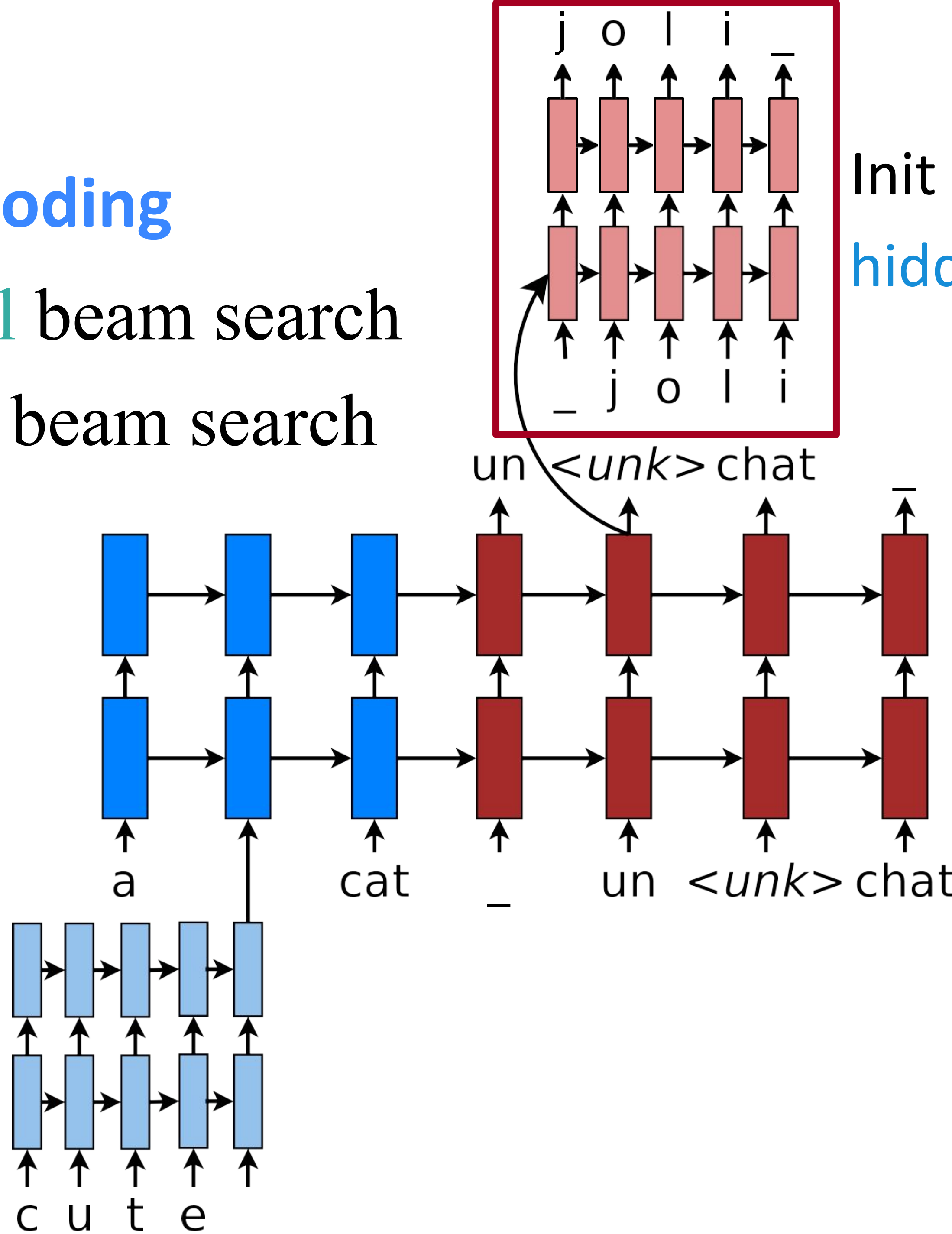


the boundary symbol

Ref. 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
Word-level NMT (Jean et al., 2015)	18.3

30x data 3 systems

Large vocab + copy mechanism

English-Czech Results

Recently, Ling et al. (2015b) attempt character-level NMT; however, the experimental evidence is weak. The authors demonstrate only small improvements over word-level baselines and acknowledge that there are no differences of significance. Furthermore, only small datasets were used without comparable results from past NMT work.

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

Systems	BLEU
Winning WMT’15 (Bojar & Tamchyna, 2015)	18.8
Word-level NMT (Jean et al., 2015)	18.3
Hybrid NMT (Luong & Manning, 2016)*	**20.7**

30x data
3 systems

Large vocab
+ copy mechanism

New SOTA

Sample English-Czech translations

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

- Word-based: identity copy fails.

Sample English-Czech translations

source	Her *11-year-old* daughter , *Shani Bart* , said it felt a little bit *weird*
human	Její **jedenáctiletá** dcera **Shani Bartová** prozradila , že je to trochu **zvláštní**
word	Její <unk> dcera <unk> <unk> řekla , že je to trochu divné
	Její **11-year-old** dcera **Shani ,** řekla , že je to trochu ***divné***
hybrid	Její <unk> dcera , <unk> <unk> , řekla , že je to <unk> <unk>
	Její **jed** Correct **etá** dcera , Wrong ***Bart*** , řekla , že cítí trochu Close

- Hybrid: correct, ***11-year-old*** – **jedenáctiletá**.

Quasi-RNN

Adam Goodge

Quasi-RNN

RNN

- Vanishing gradient problem is alleviated with LSTM and GRU
- Can only capture sequential data inputs - very slow and poor parallelization

CNN

- Much better parallelization
- Worse at capturing the sequential dependency of data

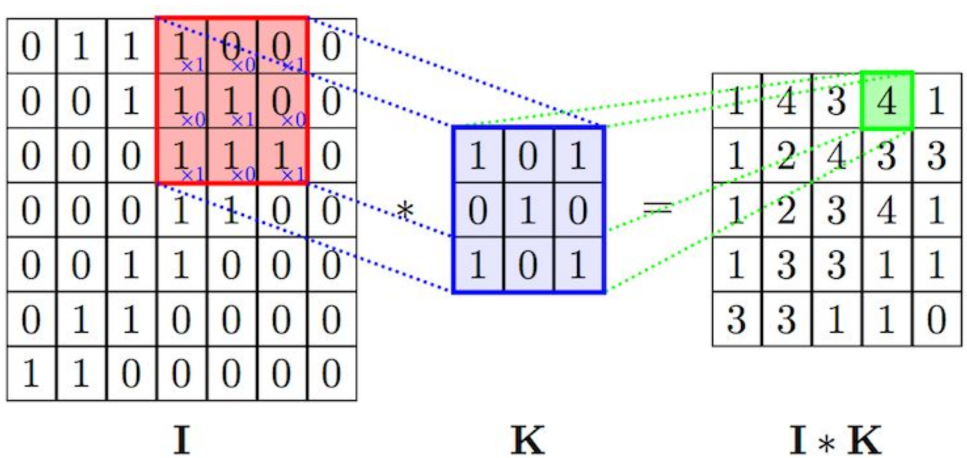
How to handle sequential dependencies between input data without creating sequential dependencies between the hidden states?

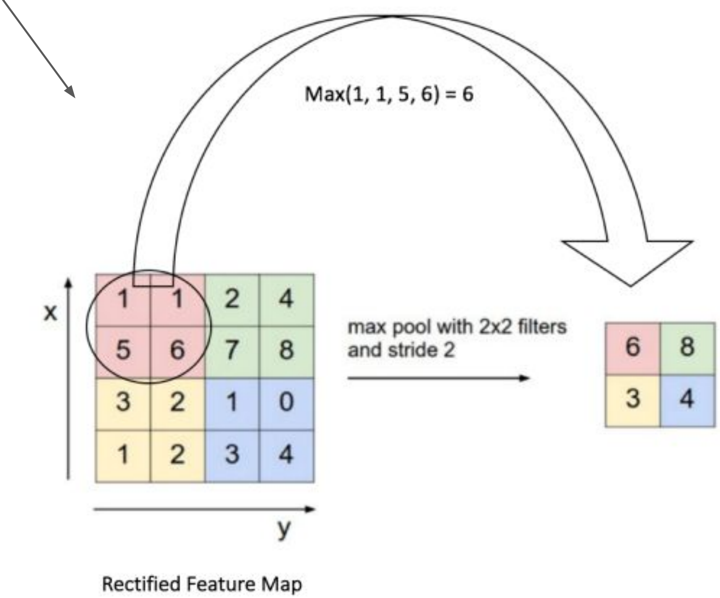
Solution?

Quasi-RNN!

CNN Primer

- Used widely in image recognition as well as NLP
- Applies a convolution -> ReLU -> Pooling -> Classification

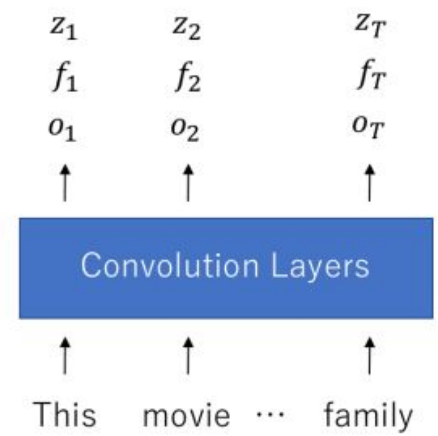


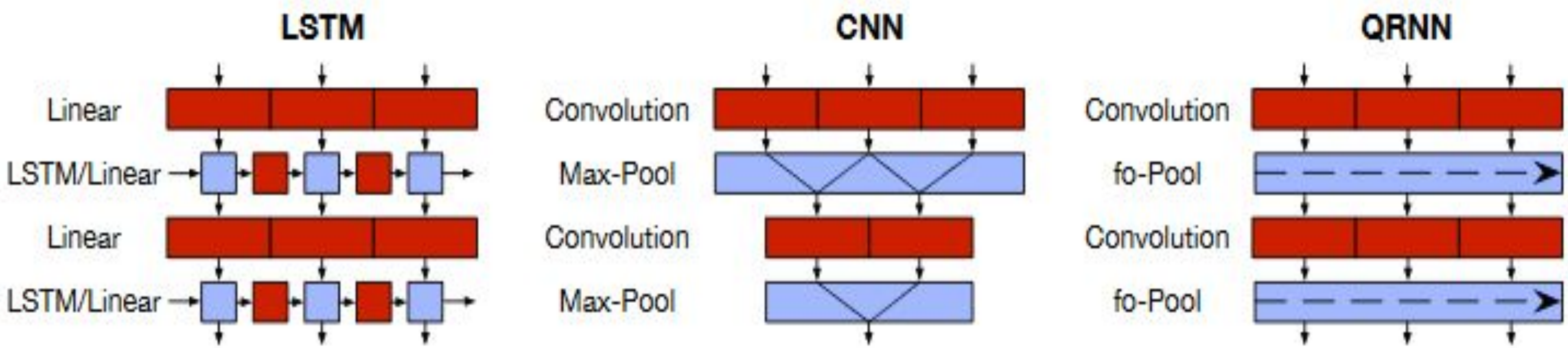


Source: https://tex.stackexchange.com/questions/437007/drawing-a-convolution-with-tikz

Source: https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/

Quasi-Recurrent Neural Network





- Parallelism computation across time:

$$\mathbf{z}_t = \tanh(\mathbf{W}_z^1 \mathbf{x}_{t-1} + \mathbf{W}_z^2 \mathbf{x}_t)$$
$$\mathbf{f}_t = \sigma(\mathbf{W}_f^1 \mathbf{x}_{t-1} + \mathbf{W}_f^2 \mathbf{x}_t)$$
$$\mathbf{o}_t = \sigma(\mathbf{W}_o^1 \mathbf{x}_{t-1} + \mathbf{W}_o^2 \mathbf{x}_t).$$

$$\mathbf{Z} = \tanh(\mathbf{W}_z * \mathbf{X})$$
$$\mathbf{F} = \sigma(\mathbf{W}_f * \mathbf{X})$$
$$\mathbf{O} = \sigma(\mathbf{W}_o * \mathbf{X}),$$

Pooling Variants

f-pooling only forget gate	$h_t = f_t \odot g_{t-1} + (1 - f_t) \odot z_t$
fo-pooling forget and output gate	$c_t = f_t \odot c_{t-1} + (1 - f_t) \odot z_t$ $h_t = o_t \odot c_t$
ifo-pooling input and forget gate	$c_t = f_t \odot c_{t-1} + i_t \odot z_t$ $h_t = o_t \odot c_t$

Key point: z_t f_t and o_t do not depend on the previous values. Essence of Q-RNN is to do the heavy computation in parallel, whilst doing minimal sequential processing in the pooling layers

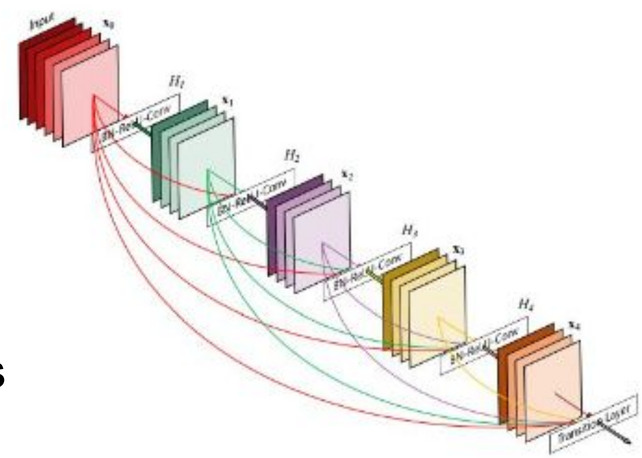
Regularization

- Extension of work by Kreuget et al. (2016)
- Keep the pooling state for a stochastic subset of channels, equivalent to stochastically setting a subset of f gate channels to 1

$$F = 1 - \text{dropout}(1 - \sigma(W_f * X))$$

Densely Connected Layers

- Authors found skip-connections helpful between layers (termed “dense convolution”)
- For L layers, this means a total of L(L-1) connections
- Improves gradient flow but parameter count is quadratic in number of layers



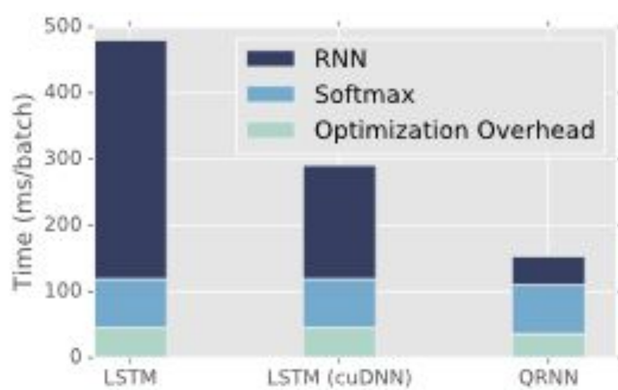
Source: https://pdfs.semanticscholar.org/4703/84314930a894e68394e8e8ed846eb214bbde.pdf

Q-RNNs for Language Modeling

- Better

Model	Parameters	Validation	Test
LSTM (medium) (Zaremba et al., 2014)	20M	86.2	82.7
Variational LSTM (medium) (Gal & Ghahramani, 2016)	20M	81.9	79.7
LSTM with CharCNN embeddings (Kim et al., 2016)	19M	—	78.9
Zoneout + Variational LSTM (medium) (Merity et al., 2016)	20M	84.4	80.6
Our models			
LSTM (medium)	20M	85.7	82.0
QRNN (medium)	18M	82.9	79.9
QRNN + zoneout ($p = 0.1$) (medium)	18M	82.1	78.3

- Faster



Batch size	Sequence length 32	64	128	256	512
8	**5.5x**	**8.8x**	**11.0x**	**12.4x**	**16.9x**
16	**5.5x**	**6.7x**	**7.8x**	**8.3x**	**10.8x**
32	**4.2x**	**4.5x**	**4.9x**	**4.9x**	**6.4x**
64	**3.0x**	**3.0x**	**3.0x**	**3.0x**	**3.7x**
128	**2.1x**	**1.9x**	**2.0x**	**2.0x**	**2.4x**
256	**1.4x**	**1.4x**	**1.3x**	**1.3x**	**1.3x**

Limitations

- Subsequent papers from the authors find character level Neural Language Modeling is done best by LSTM still (requires more complex long-term interactions)
- Shows that pooling to handle dependencies is not always successful
- ‘An Analysis of Neural Language Modeling at Multiple Scales’ (https://arxiv.org/abs/1803.08240)

Links

Q-RNN ICLR 2017 paper: https://openreview.net/pdf?id=H1zJ-v5x

Sample code: https://github.com/salesforce/pytorch-qrnn

Baidu DeepVoice: https://arxiv.org/pdf/1702.07825.pdf

http://research.baidu.com/Blog/index-view?id=91

Zoneout for RNNs: https://arxiv.org/pdf/1606.01305.pdf