CS6120: Lecture 7

Jiaji Huang

https://jiaji-huang.github.io

Recap: Transformer

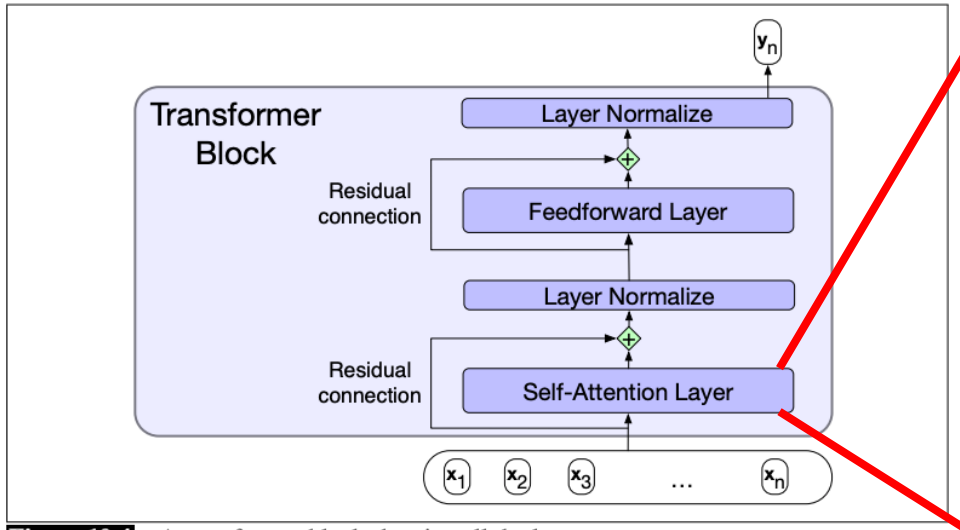


Figure 10.4 A transformer block showing all the layers.

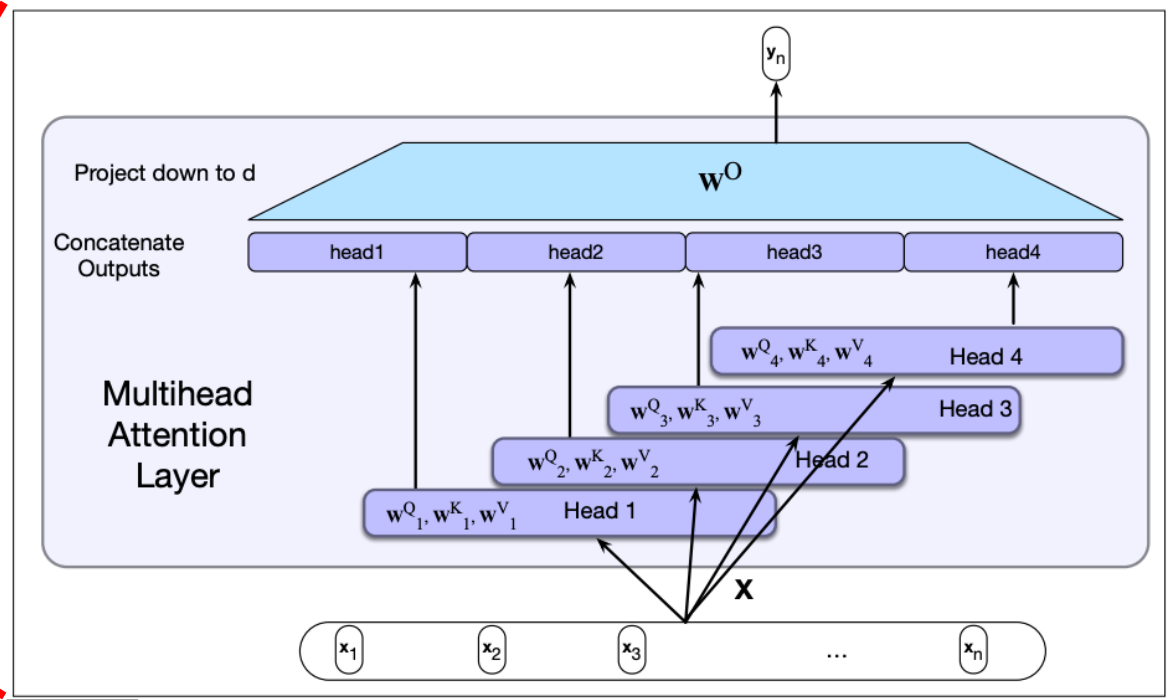


Figure 10.5 Multihead self-attention: Each of the multihead self-attention layers is provided with its own set of key, query and value weight matrices. The outputs from each of the layers are concatenated and then projected down to d, thus producing an output of the same size as the input so layers can be stacked.

Illustration from https://web.stanford.edu/~jurafsky/slp3/10.pdf

Recap: Typical Architectures

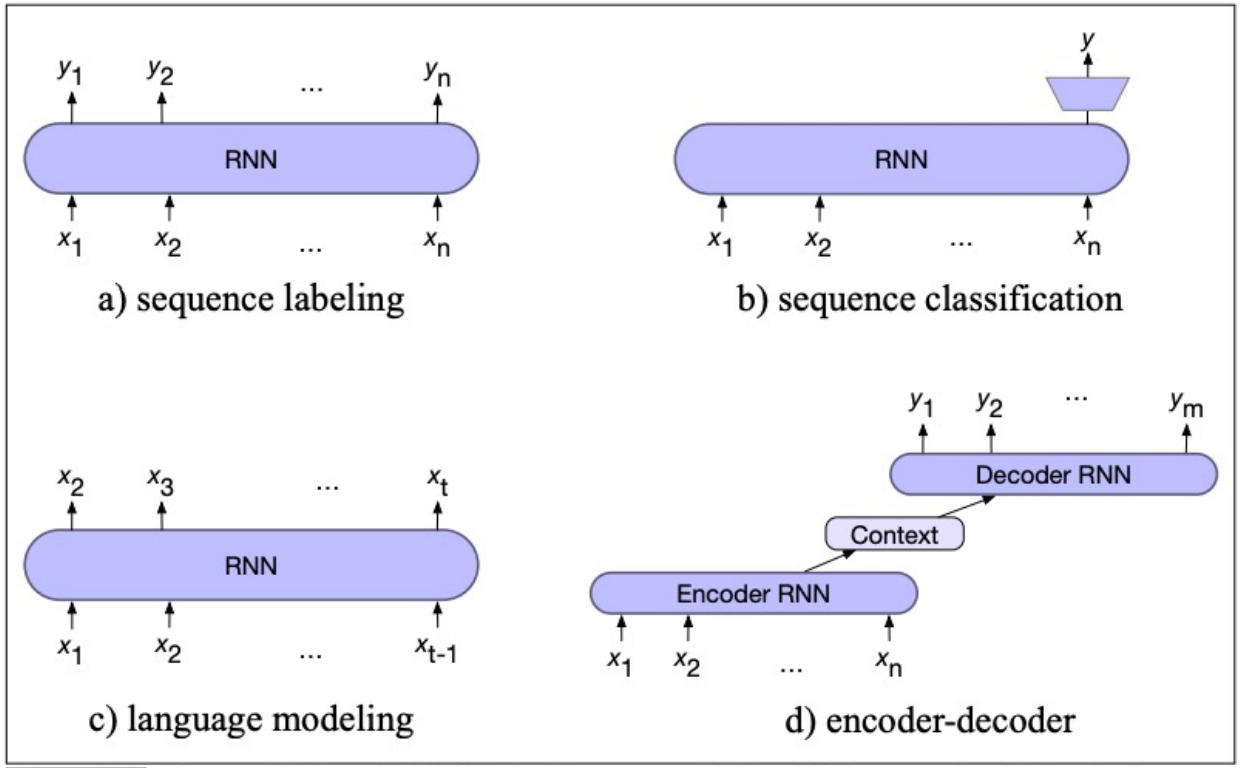


Figure 9.15 Four architectures for NLP tasks. In sequence labeling (POS or named entity tagging) we map each input token x_i to an output token y_i. In sequence classification we map the entire input sequence to a single class. In language modeling we output the next token conditioned on previous tokens. In the encoder model we have two separate RNN models, one of which maps from an input sequence **x** to an intermediate representation we call the **context**, and a second of which maps from the context to an output sequence **y**.

Illustration from https://web.stanford.edu/~jurafsky/slp3/9.pdf

Agenda

- Applications
  - Translation
  - Question Answering
- Other Modality
  - Speech to text
  - Text to Speech
  - Vision

Machine Translation is Hard

- Because of linguistic divergences
  - Morphology
  - Syntax
  - semantics
- Linguistic typology: studies cross-linguistic similarities and differences

Word Order Typology

- SVO: German, French, English, Mandarin
- SOV: Hindi, Japanese
- VSO: Irish, Arabic

English: *He wrote a letter to a friend*
Japanese: *tomodachi ni tegami-o kaita*
friend to letter wrote
Arabic: *katabt risāla li ṣadq*
wrote letter to friend

Illustration from https://web.stanford.edu/~jurafsky/slp3/13.pdf

Lexical Divergences

- En-> Es
  - bass -> lubnia/bajo
- En->Zh
  - Brother -> 哥/弟
- Word Sense Disambiguation
- Lexical Gap
  - Zh -> En: 孝 -> ?

Examples from https://web.stanford.edu/~jurafsky/slp3/13.pdf

Morphological Topology

- Base form: run
- Present tense: Running, past tense: ran
- Isolating languages: Vietnamese (1 morpheme per word)
- Polysynthetic language: Siberian Yupik (many morphemes per word)

Classical Approach

Statistical Machine Translation

- Bayesian Rule

$$T^* = arg\max_T P(T|S) = arg\max_T P(S|T)P(T)$$

- $P(S|T)$: translation model, faithfulness
- $P(T)$: language model, fluency
- IBM models:
  - Alignment a
  - $P(S|T) = \sum_a P(S, a|T)$

Alignment

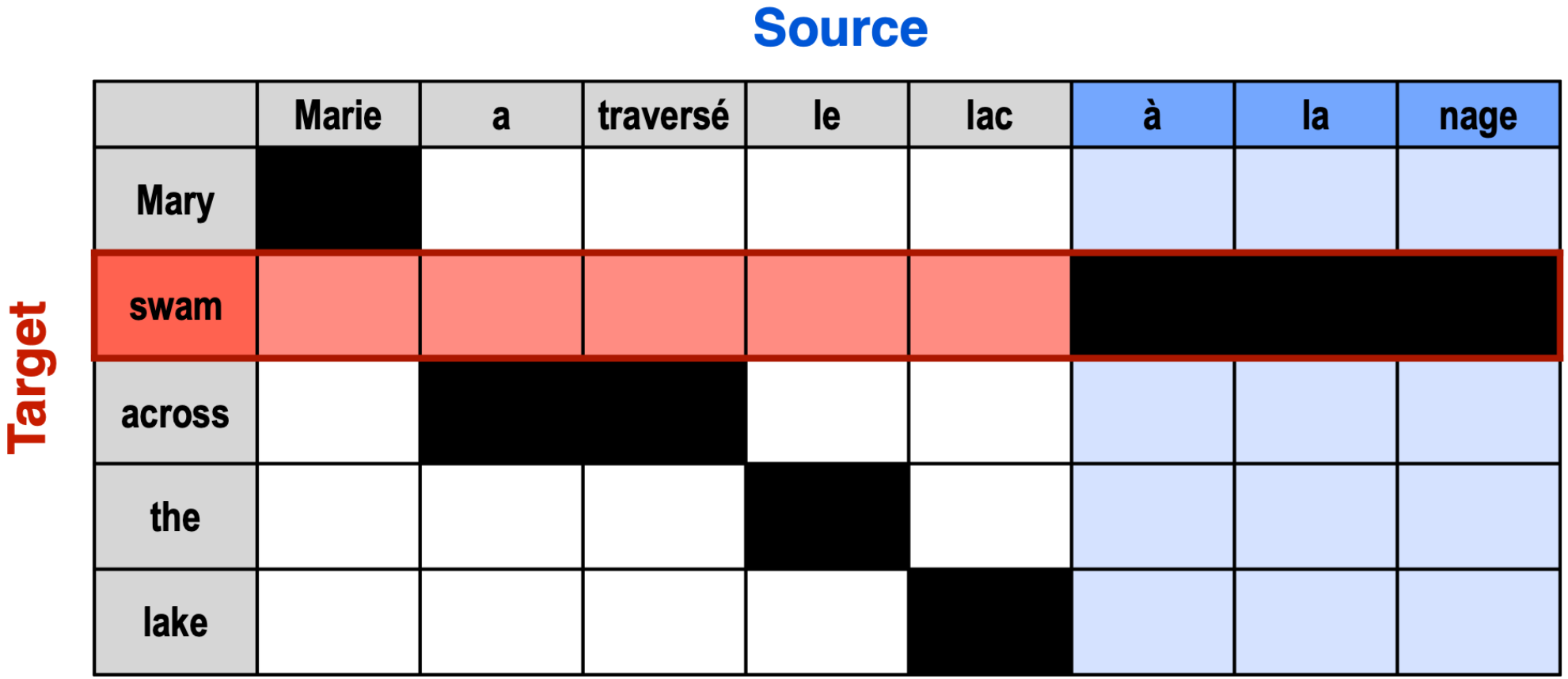
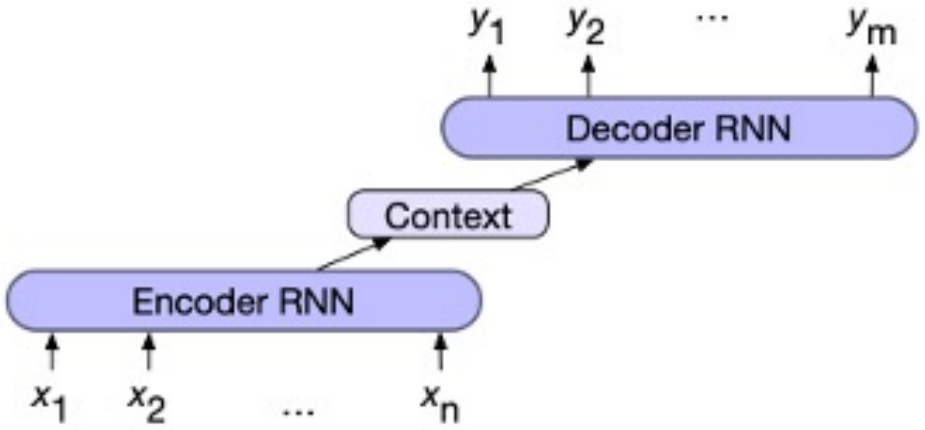


Illustration from https://courses.grainger.illinois.edu/CS447/fa2020/Slides/Lecture14.pdf

Modern Approach

Encoder-Decoder Model



$$\mathbf{h} = \texttt{encoder}(x)$$
$$y_{i+1} = \texttt{decoder}(\mathbf{h}, y_1, \ldots, y_i)) \quad \forall i \in [1, \ldots, m]$$

On the x_i's and y_i's

- Option1: Word, too huge vocabulary, cannot handle OOV
- Option2: character, too long input, inferior performance
- Option3: subwords
  - Several methods to obtain
    - Byte-pair encoding (BPE)
    - Wordpiece
    - Sentencepiece

Byte Pair Encoding (BPE)

- Merge the most frequent pair of tokens

corpus

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

vocabulary

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

corpus

```
5   l o w _
2   l o w e s t _
6   n e w er _
3   w i d er _
2   n e w _
```

vocabulary

```
_, d, e, i, l, n, o, r, s, t, w, er
```

Example from https://web.stanford.edu/~jurafsky/slp3/13.pdf

Byte Pair Encoding (BPE)

corpus	vocabulary
5 l o w _	_, d, e, i, l, n, o, r, s, t, w, er, er_
2 l o w e s t _	
6 n e w er_	
3 w i d er_	
2 n e w _	

corpus	vocabulary
5 l o w _	_, d, e, i, l, n, o, r, s, t, w, er, er_, ne
2 l o w e s t _	
6 ne w er_	
3 w i d er_	
2 ne w _	

Example from https://web.stanford.edu/~jurafsky/slp3/13.pdf

Drawback of BPE

- Small non-meaningful subwords

Natural Language Engineering (2020), **26**, pp. 375–382
doi:10.1017/S1351324920000145

CAMBRIDGE UNIVERSITY PRESS

EMERGING TRENDS

Emerging trends: Subwords, seriously?

Kenneth Ward Church

Baidu, USA
E-mail: kenneth.ward.church@gmail.com

Abstract

Subwords have become very popular, but the BERT[a] and ERNIE[b] tokenizers often produce surprising results. Byte pair encoding (BPE) trains a dictionary with a simple information theoretic criterion that sidesteps the need for special treatment of unknown words. BPE is more about training (populating a dictionary of word pieces) than inference (parsing an unknown word into word pieces). The parse at inference time can be ambiguous. Which parse should we use? For example, "electroneutral" can be parsed as electron-eu-tral or electro-neutral, and "bidirectional" can be parsed as bid-ire-ction-al and bi-directional. BERT and ERNIE tend to favor the parse with more word pieces. We propose minimizing the number of word pieces. To justify our proposal, a number of criteria will be considered: sound, meaning, etc. The prefix, bi-, has the desired vowel (unlike bid) and the desired meaning (bi is Latin for two, unlike bid, which is Germanic for offer).

Original:	`corrupted`	Original:	`Completely preposterous suggestions`
BPE:	`cor rupted`	BPE:	`Comple t ely prep ost erous suggest ions`

Example from https://web.stanford.edu/~jurafsky/slp3/13.pdf

Wordpiece

- Initialize with a set of all characters
- Repeat till there are V wordpieces
  - Train an n-gram language model, using the current set
  - Consider concatenating two word pieces, so that the resulting n-gram has biggest likelihood increase

SentencePiece

- A library implementing BPE and another method called *Unigram*
- How Unigram works
  - Fix token set, learn probabilistic split of words (into these tokens), via EM
  - Prune away subwords with low probabilities

Original:	corrupted	Original:	Completely preposterous suggestions
BPE:	cor rupted	BPE:	Comple t ely prep ost erous suggest ions
Unigram:	corrupt ed	Unigram:	Complete ly pre post er ous suggestion s

Example from https://web.stanford.edu/~jurafsky/slp3/13.pdf

What about Chinese

- Languages without space separating words
- Run segmenter first, e.g.,

Python Jieba

```
>>> import jieba
>>> s="我在中国科学技术大学读本科"
>>> " ".join(jieba.cut(s))
'我 在 中国 科学技术 大学 读 本科'
```

Architecture

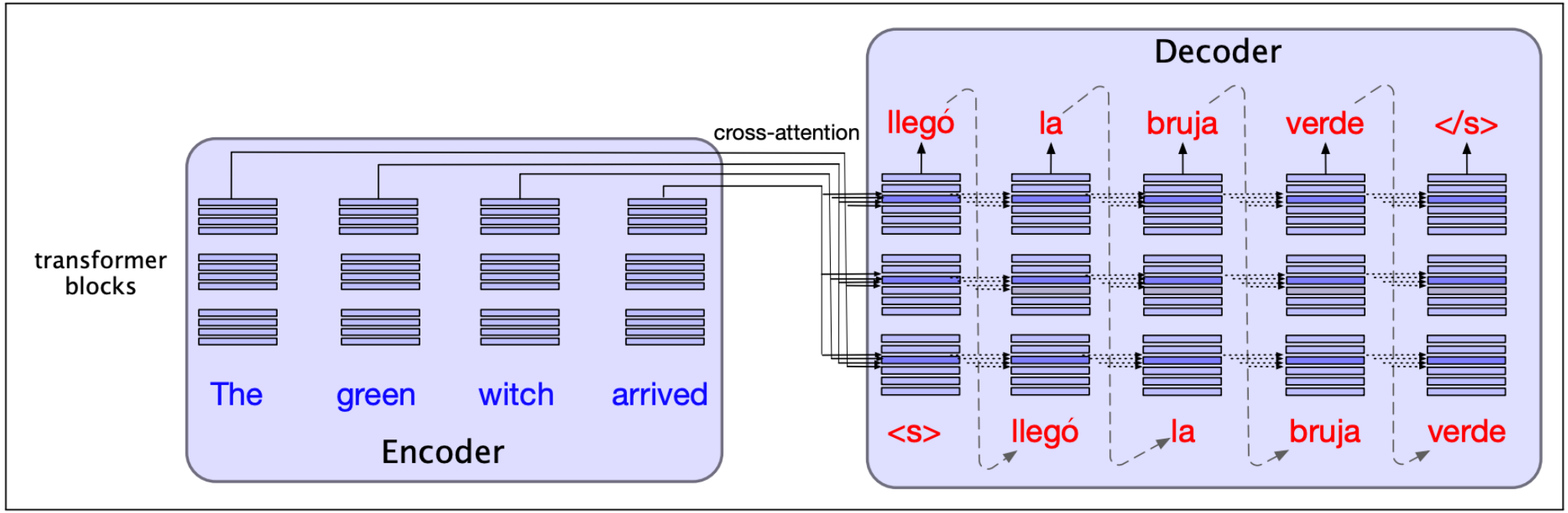
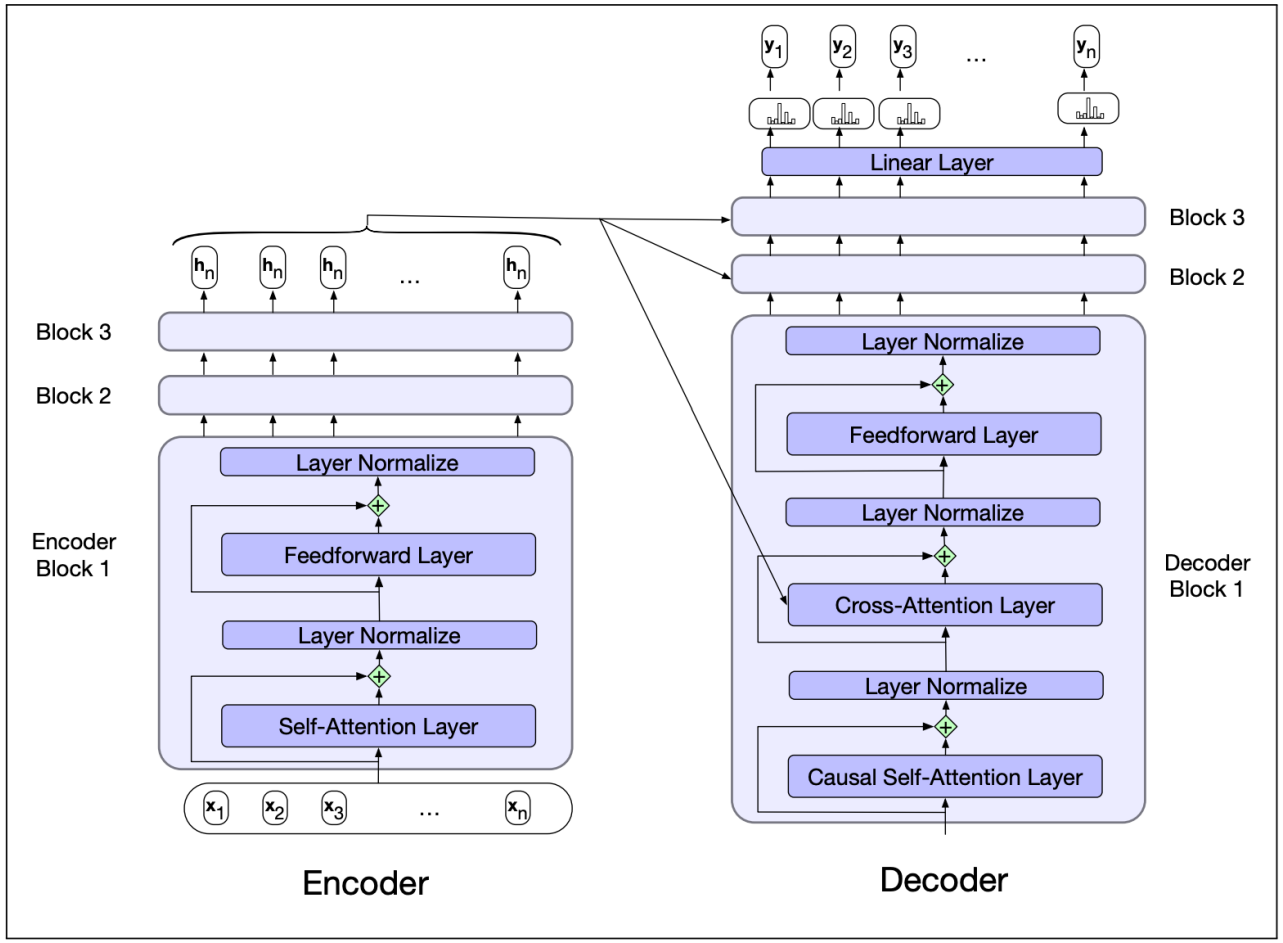


Figure 10.5 The encoder-decoder transformer architecture for machine translation. The encoder uses the transformer blocks we saw in Chapter 9, while the decoder uses a more powerful block with an extra **cross-attention** layer that can attend to all the encoder words. We'll see this in more detail in the next section.

Illustration from https://web.stanford.edu/~jurafsky/slp3/13.pdf

Zoom in for cross-attention

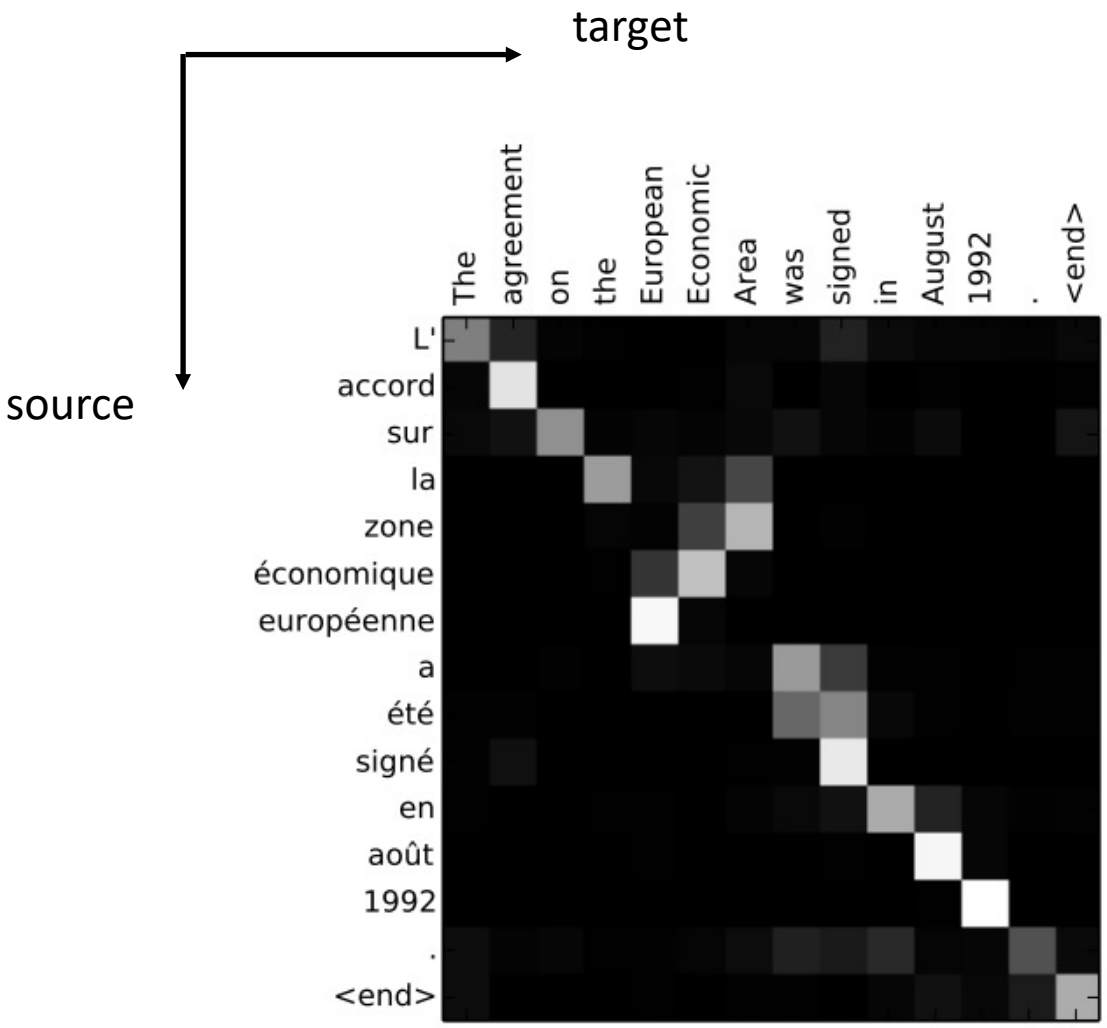


$$\mathbf{Q} = \mathbf{W}^{\mathbf{Q}}\mathbf{H}^{dec[i-1]};\ \mathbf{K} = \mathbf{W}^{\mathbf{K}}\mathbf{H}^{enc};\ \mathbf{V} = \mathbf{W}^{\mathbf{V}}\mathbf{H}^{enc}$$

$$\text{CrossAttention}(\mathbf{Q},\mathbf{K},\mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^{\intercal}}{\sqrt{d_k}}\right)\mathbf{V}$$

Illustration from https://web.stanford.edu/~jurafsky/slp3/13.pdf

Cross Attention is alignment



arXiv
https://arxiv.org › cs

Neural Machine Translation by Jointly Learning to Align ...

by D Bahdanau · 2014 · Cited by 31472 — With this new approach, we achieve a **translation** performance comparable to the existing state-of-the-art phrase-based **system** on the task of ...

Note the alignment is neither diagonal, nor triangular!

Extensions

- Simultaneous Translation

ACL Anthology
https://aclanthology.org › ...

STACL: Simultaneous Translation with Implicit Anticipation ...

by M Ma · 2019 · Cited by 194 — **Simultaneous translation**, which translates sentences before they are finished, is use- ful in many scenarios but is notoriously dif- ficult due to word-order ...

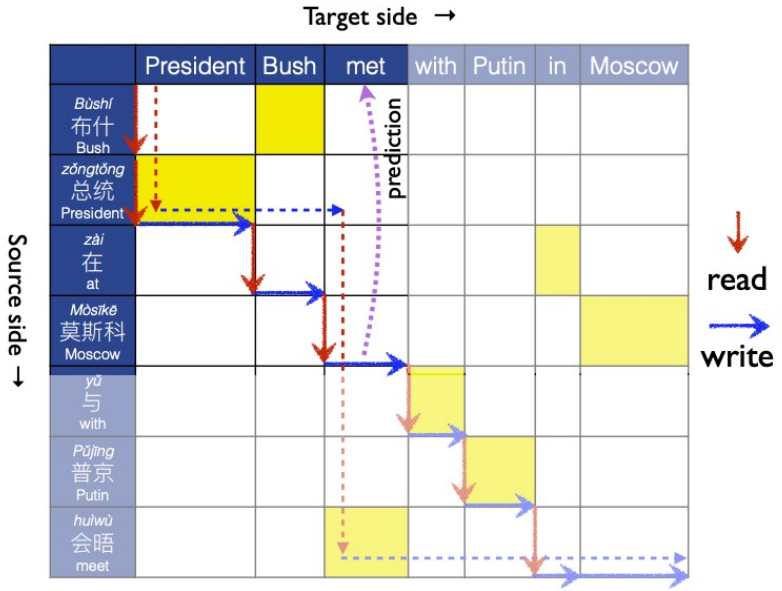


Figure 1: Our wait-k model emits target word y_t given source-side prefix $x_1 \ldots x_{t+k-1}$, often before seeing the corresponding source word (here k=2, outputing y_3="met" before x_7="*huìwù*"). Without anticipation, a 5-word wait is needed (dashed arrows). See also Fig. 2.

Extensions

- Multilingual Translation
- Language encoding token: l_s , l_t

$$\begin{aligned} \mathbf{h} &= \text{encoder}(x, l_s) \\ y_{i+1} &= \text{decoder}(\mathbf{h}, l_t, y_1, \ldots, y_i)) \quad \forall i \in [1, \ldots, m] \end{aligned}$$

Extensions

UNSUPERVISED MACHINE TRANSLATION
USING MONOLINGUAL CORPORA ONLY

Guillaume Lample † ‡ , **Alexis Conneau** † , **Ludovic Denoyer** ‡ , **Marc'Aurelio Ranzato** †
† Facebook AI Research,
‡ Sorbonne Universités, UPMC Univ Paris 06, LIP6 UMR 7606, CNRS

- Unsupervised Machine Translation
  - Source and target text has no correspondence
- Method
  - Initialize: Word-to-word translation (bilingual lexicon induction)
  - Train encoder-decoder by
    - Sample source sentence x, translation y using current model
    - Train as if supervised case, using (x, y)

Agenda

- Applications
  - Translation
  - Question Answering
- Other Modality
  - Speech to text
  - Text to Speech
  - Vision

Paradigms of QA

- Open domain QA
- Knowledge-based QA
- Language Model, e.g., ChatGPT

Open domain QA Setup

- Information Retrieval (IR)
- Reading Comprehension: extract a range of text as answer

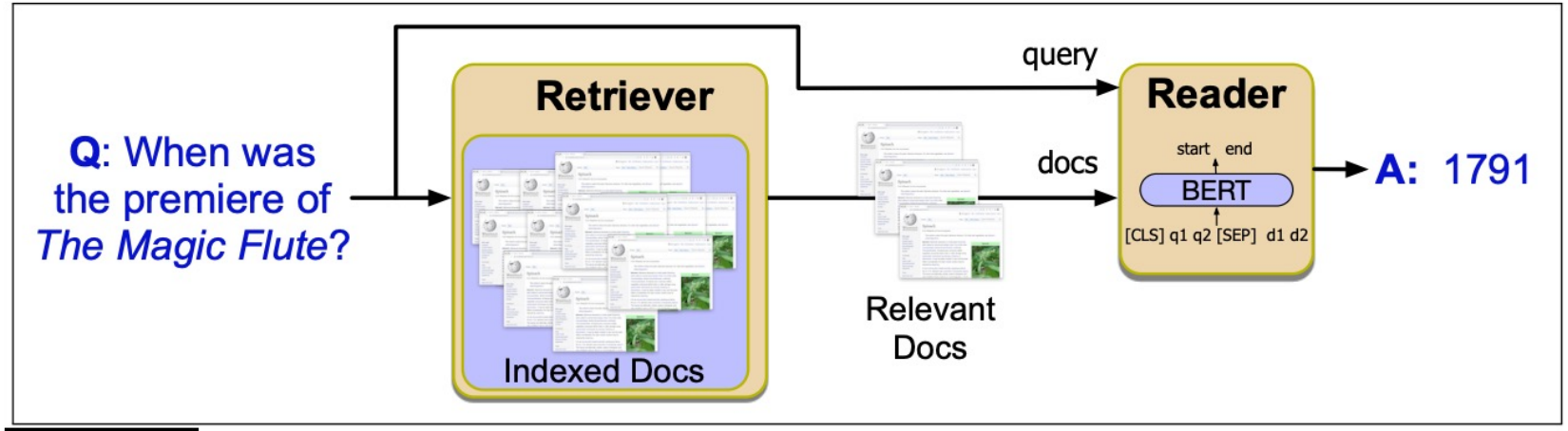


Figure 14.10 IR-based factoid question answering has two stages: **retrieval**, which returns relevant documents from the collection, and **reading**, in which a neural reading comprehension system extracts answer spans.

Examples from https://web.stanford.edu/~jurafsky/slp3/14.pdf

Information Retrieval

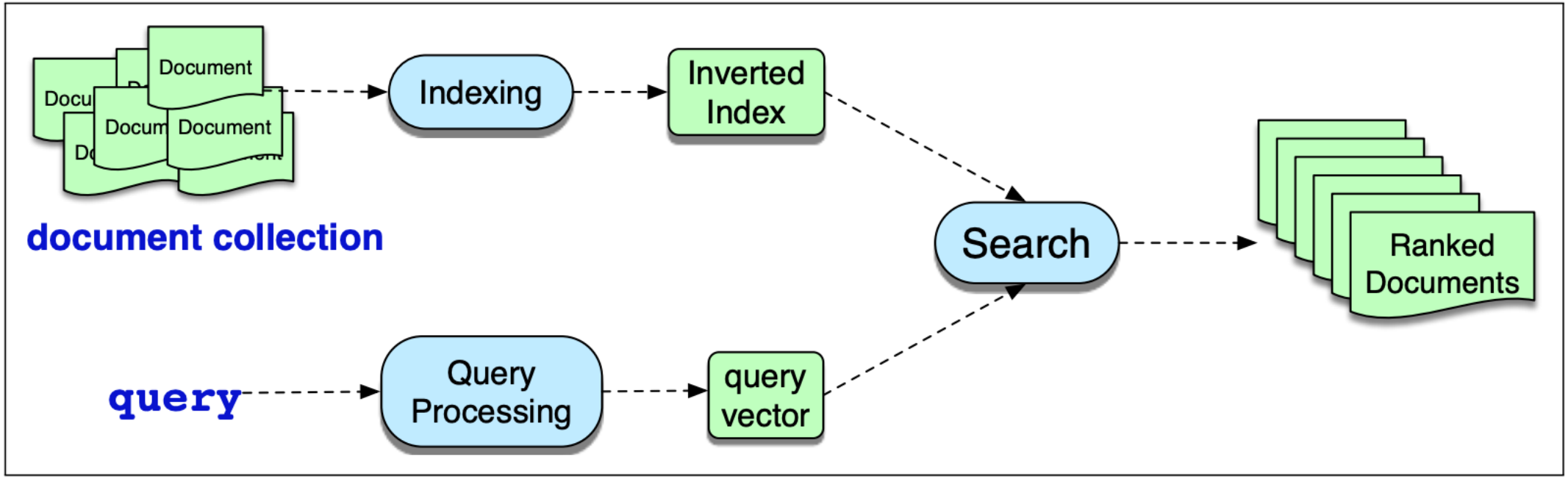


Figure 14.1 The architecture of an ad hoc IR system.

IR based on tf-idf

- Encode each document using tf-idf
- Recap for tf-idf

$$\text{tf}_{t,d} = \log_{10}(\text{count}(t,d)+1)$$

$$\text{idf}_t = \log_{10}\frac{N}{\text{df}_t}$$

$$\text{tf-idf}(t,d) = \text{tf}_{t,d} \cdot \text{idf}_t$$

- Score(q, d) = cos(q, d), where q, d are tf-idf vectors for query and document

IR based on Deep nets

- Drawback of tf-idf approach:
  - Query words must overlap with those in document
- Instead, we could encode with dense vectors

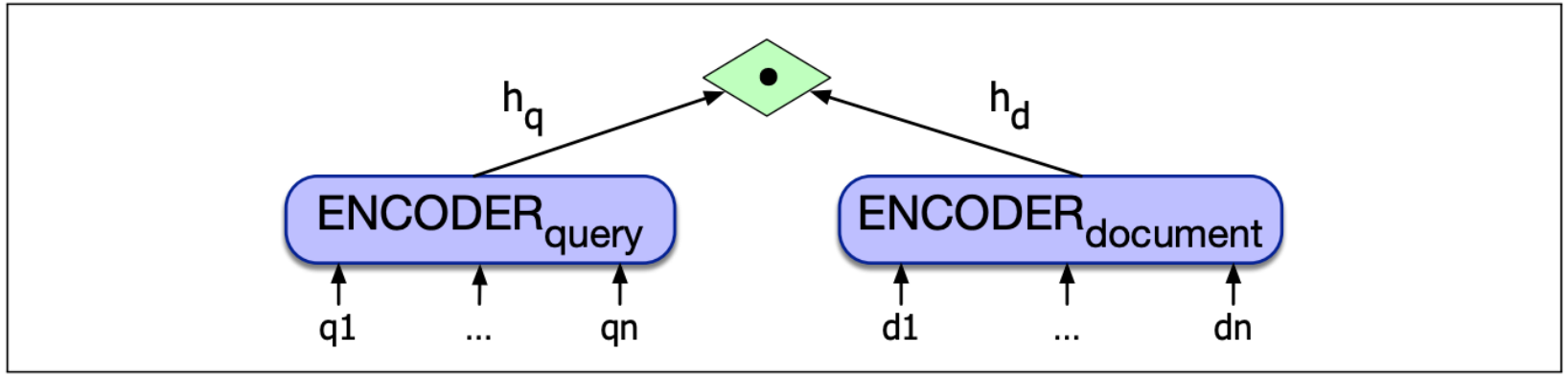


Figure 14.8 BERT bi-encoder for computing relevance of a document to a query.

Examples from https://web.stanford.edu/~jurafsky/slp3/14.pdf

Reading Comprehension

- In the retrieved doc, Find span of text as answer
- Example setup:
  - Input:
    - Question: How tall is Mt. Everest?
    - Passage: "… Reaching 29,029 feet at its summit, Mt. Everest stands …"
  - Return:
    - 29,029 feet
- Formulated into span labeling problem

Span labeling

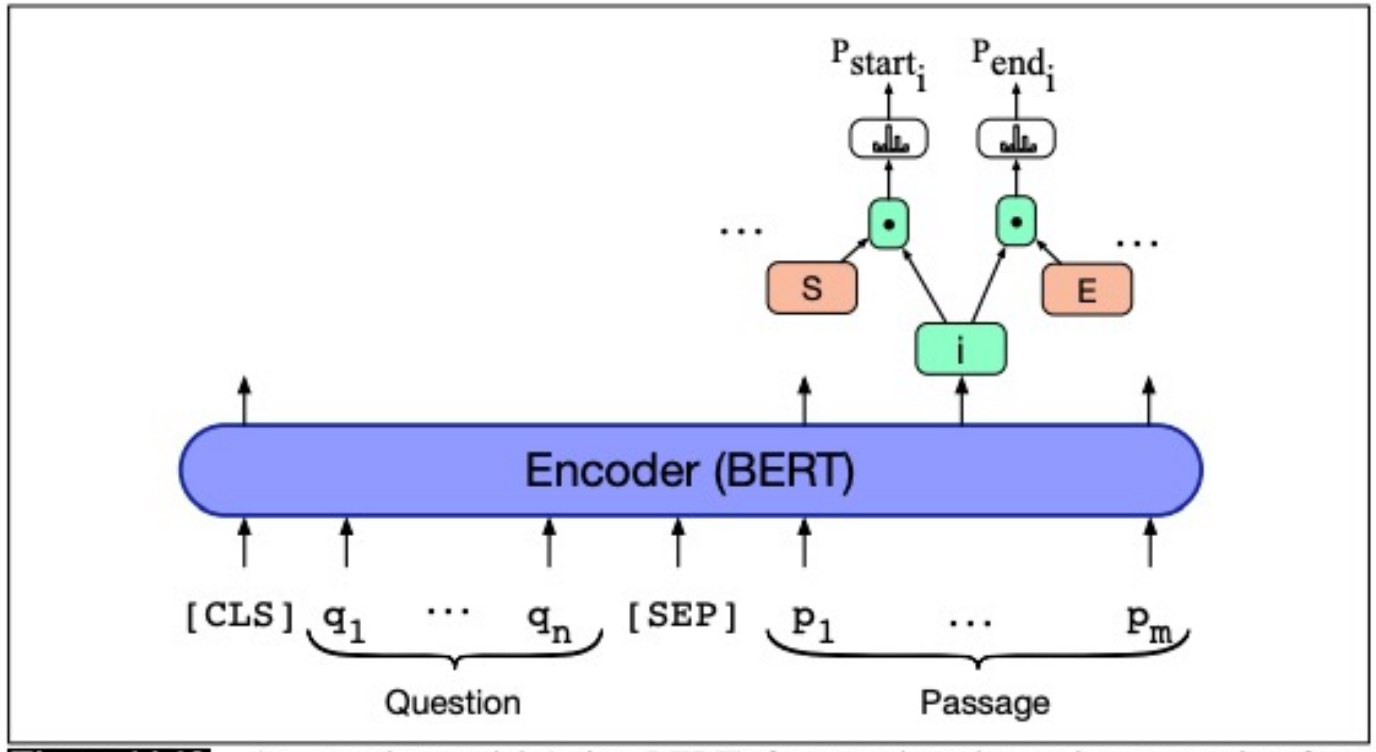


Figure 14.12 An encoder model (using BERT) for span-based question answering from reading-comprehension-based question answering tasks.

- Span start embedding S
- Span end embedding E

$$P_{\text{start}_i} = \frac{\exp(S \cdot p'_i)}{\sum_j \exp(S \cdot p'_j)}$$

$$P_{\text{end}_i} = \frac{\exp(E \cdot p'_i)}{\sum_j \exp(E \cdot p'_j)}$$

- Training loss

$$L = -\log P_{\text{start}_i} - \log P_{\text{end}_i}$$

Paradigms of QA

- Open domain QA
- Knowledge-based QA
- Language Model, e.g., ChatGPT

Knowledge-based QA

- Setup:
  - Input:
    - Question: where is Golden Gate Park?
    - RDF (Resource Description Framework) triples, e.g.,

Subject	predicate	object
Golden Gate Park	location	San Francisco

  - Return:

    San Francisco

Knowledge-based QA

- How do we know which triple can answer the question?

  “where is Golden Gate Park?”

- Entity linking
  - Link “Golden State park” to relevant triples
- Relation Linking
  - Link “where” to a triple with ”location” as predicate

Entity Linking

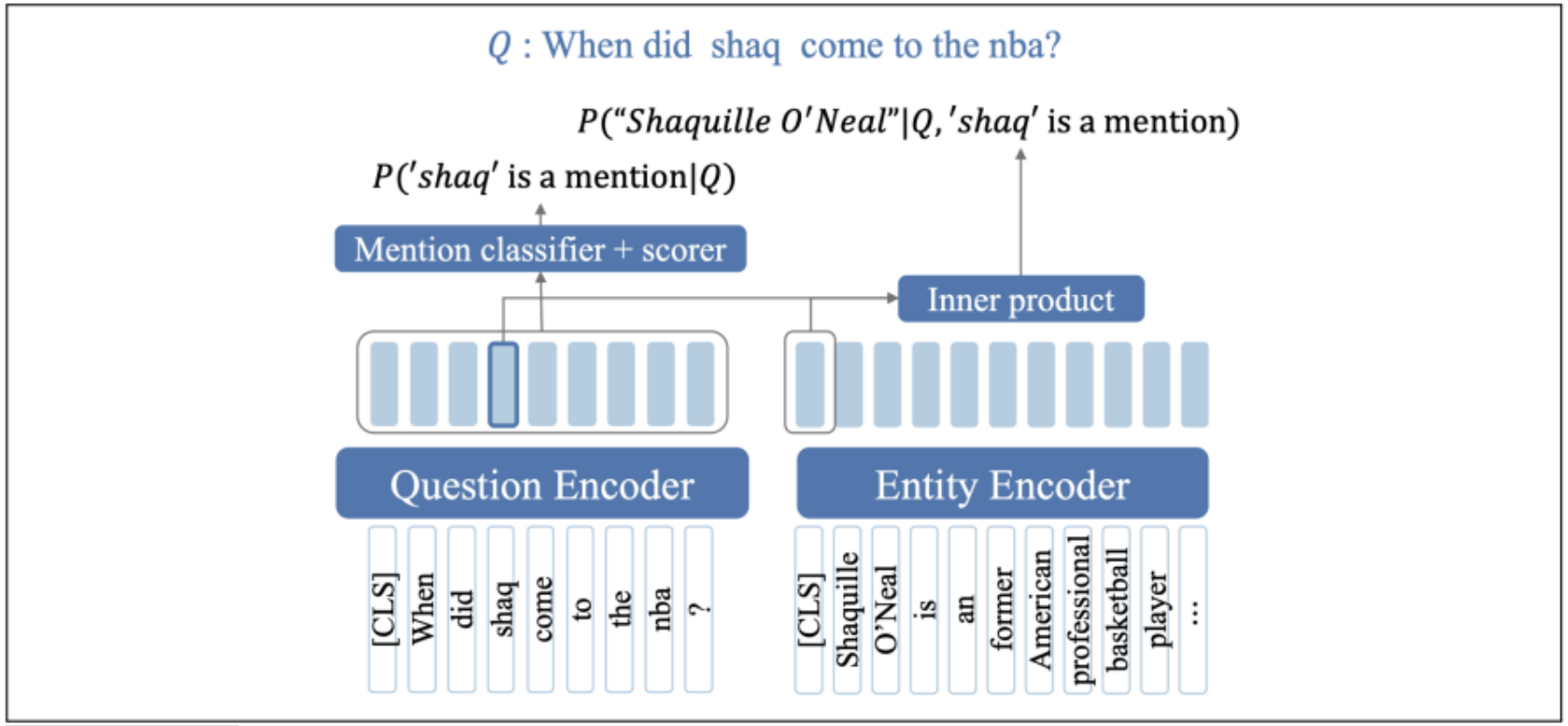


Figure 14.13 A sketch of the inference process in the ELQ algorithm for entity linking in questions (Li et al., 2020). Each candidate question mention span and candidate entity are separately encoded, and then scored by the entity/span dot product.

Illustration from https://web.stanford.edu/~jurafsky/slp3/14.pdf

Relation Linking

- Encode the question via an encoder

$$\mathbf{m}_r = \mathrm{BERT}_{\mathrm{CLS}}([\mathrm{CLS}]q_1 \cdots q_n[\mathrm{SEP}])$$

- Trainable vectors $\{r_i\}$ for each relation
- Compute score

$$s(\mathbf{m_r}, r_i) = \mathbf{m_r} \cdot \mathbf{w_{r_i}}$$

- Choose the relation with softmax probability

Paradigms of QA

- Open domain QA
- Knowledge-based QA
- Language Model, e.g., ChatGPT

Large Language Model (LLM) for QA

- Trained on huge corpora, LLM’s store knowledge
- Many open API’s

Risks and Concerns

- Factuality
  - Stochastic Parrot, fluent but false
- Harmfulness
  - Question: “How to make a bomb?”

On the Dangers of Stochastic Parrots: Can Language Models Be Too Big? 

Emily M. Bender*
ebender@uw.edu
University of Washington
Seattle, WA, USA

Timnit Gebru*
timnit@blackinai.org
Black in AI
Palo Alto, CA, USA

Angelina McMillan-Major
aymm@uw.edu
University of Washington
Seattle, WA, USA

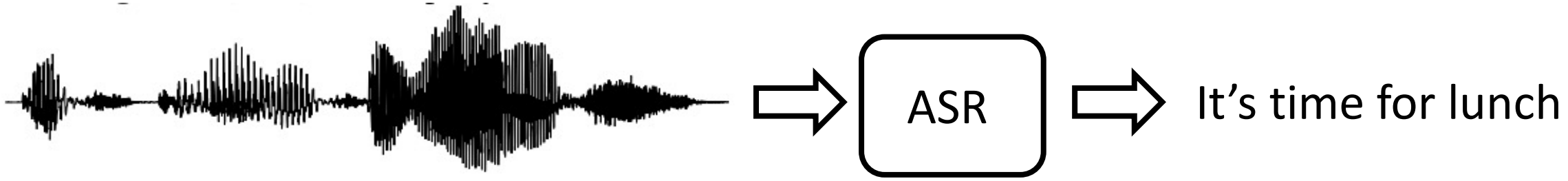
Shmargaret Shmitchell
shmargaret.shmitchell@gmail.com
The Aether

Agenda

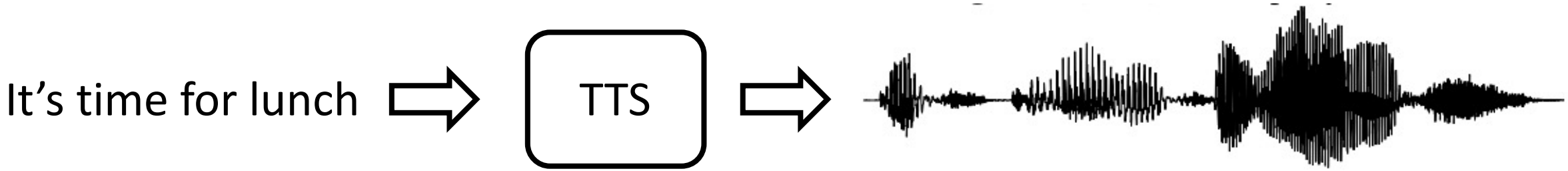
- Applications
  - Translation
  - Question Answering
- Other Modality
  - Speech to text
  - Text to Speech
  - Vision

Two Tasks

- Automatic Speech Recognition (ASR)



- Speech synthesis, or Text-to-Speech (TTS)



Word Error Rate (WER) on WSJ eval92

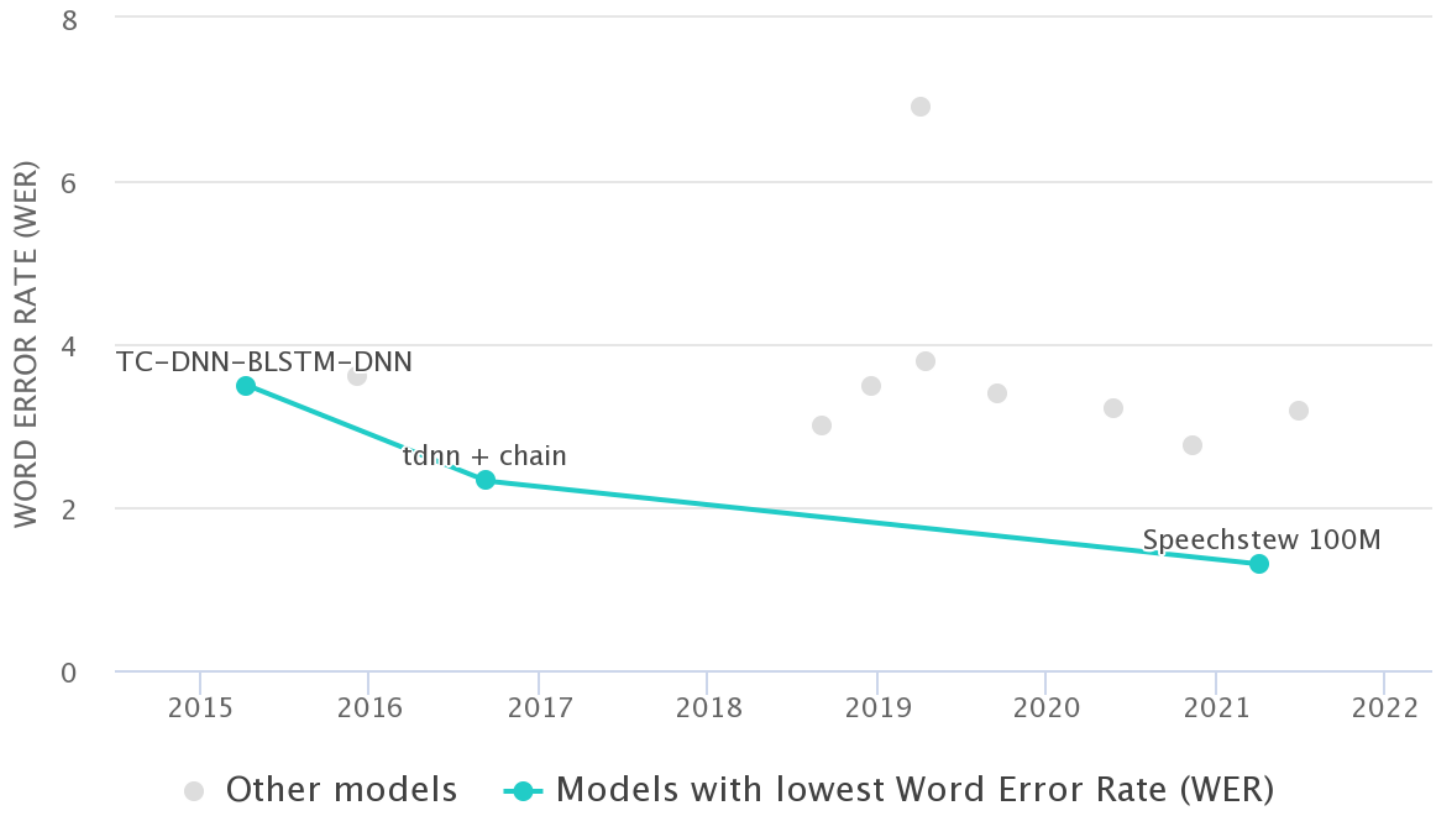


Image from https://paperswithcode.com/sota/speech-recognition-on-wsj-eval92

End-to-end ASR

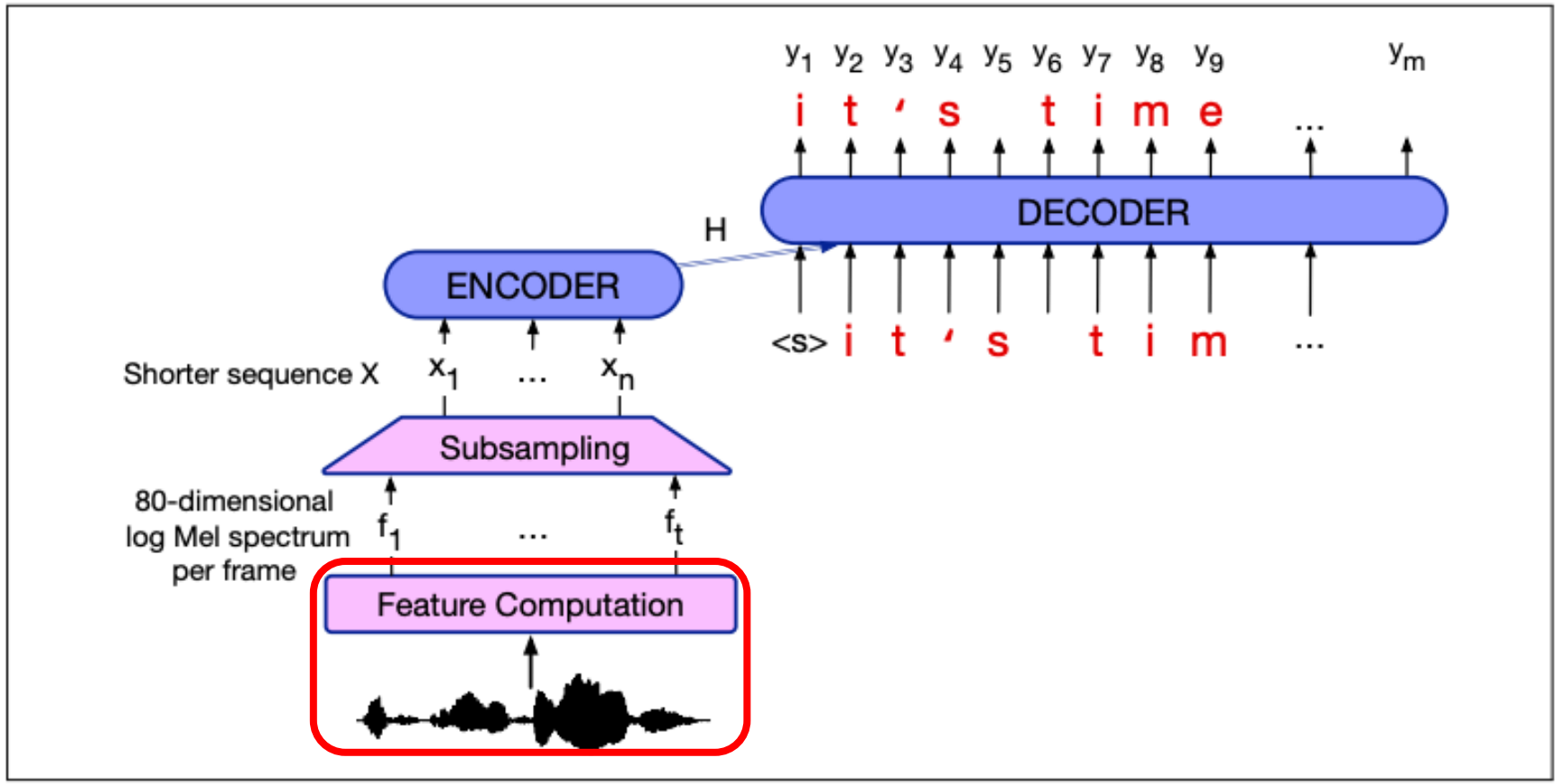


Figure 16.6 Schematic architecture for an encoder-decoder speech recognizer.

Illustration from https://web.stanford.edu/~jurafsky/slp3/16.pdf

Feature Computation

- Windowing
  - Rectangular window
    - Gibbs phenomenon
  - Hamming window

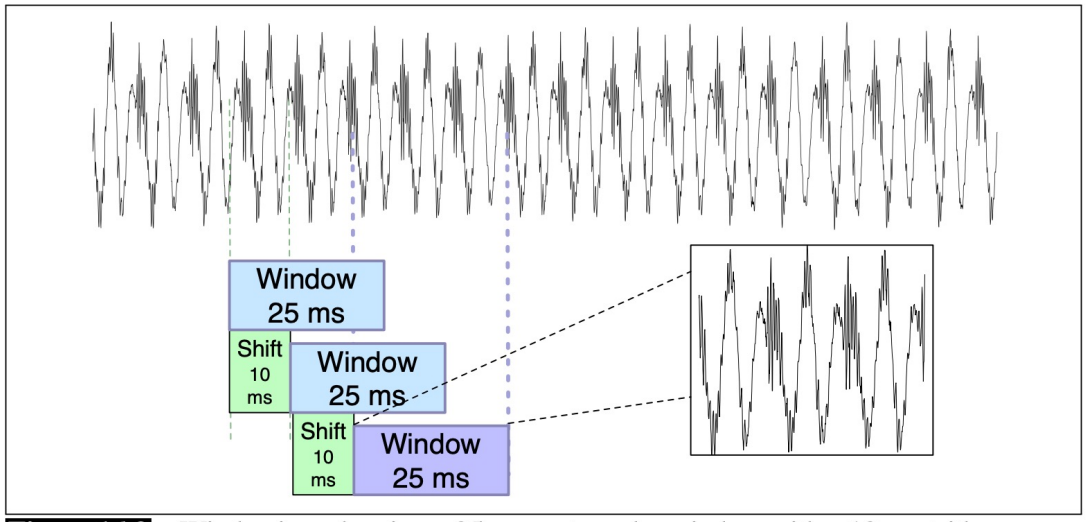
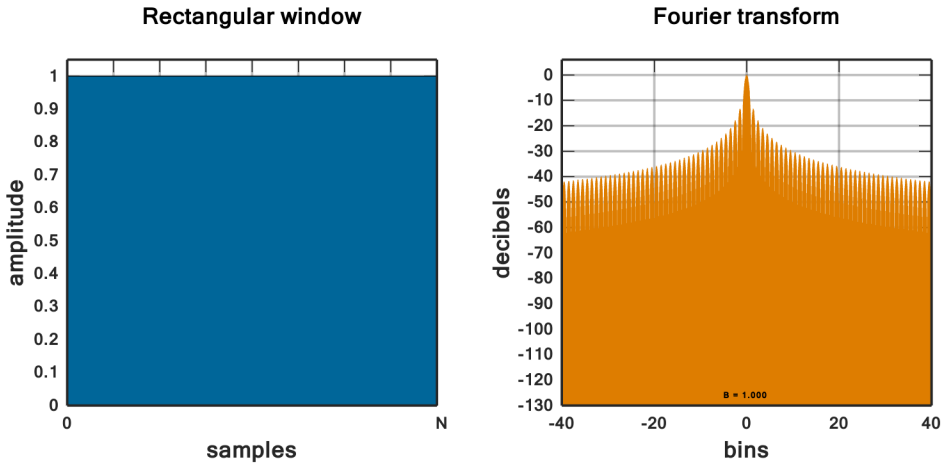


Figure 16.2 Windowing, showing a 25 ms rectangular window with a 10ms stride.



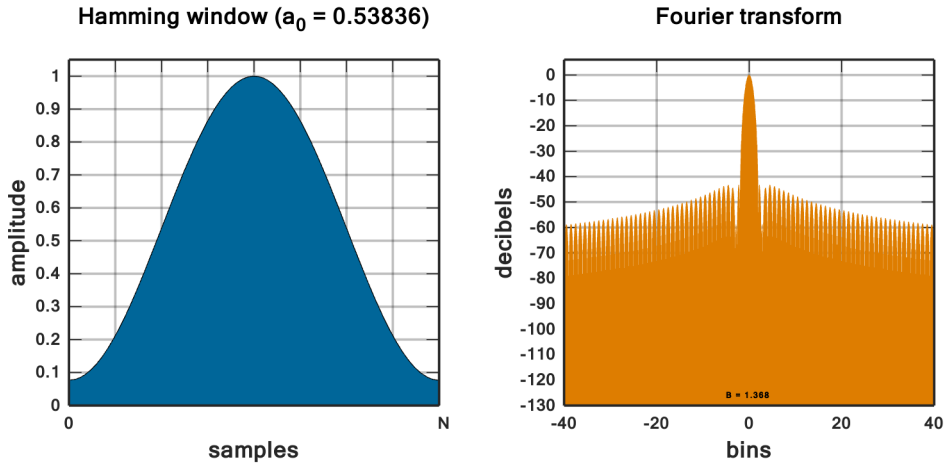


Illustration from https://web.stanford.edu/~jurafsky/slp3/16.pdf and https://en.wikipedia.org/wiki/Window_function

Convert to Frequency domain

- Apply FFT inside each window
- Apply Mel Filter banks
  - Why? Human hearing is less sensitive at higher frequency

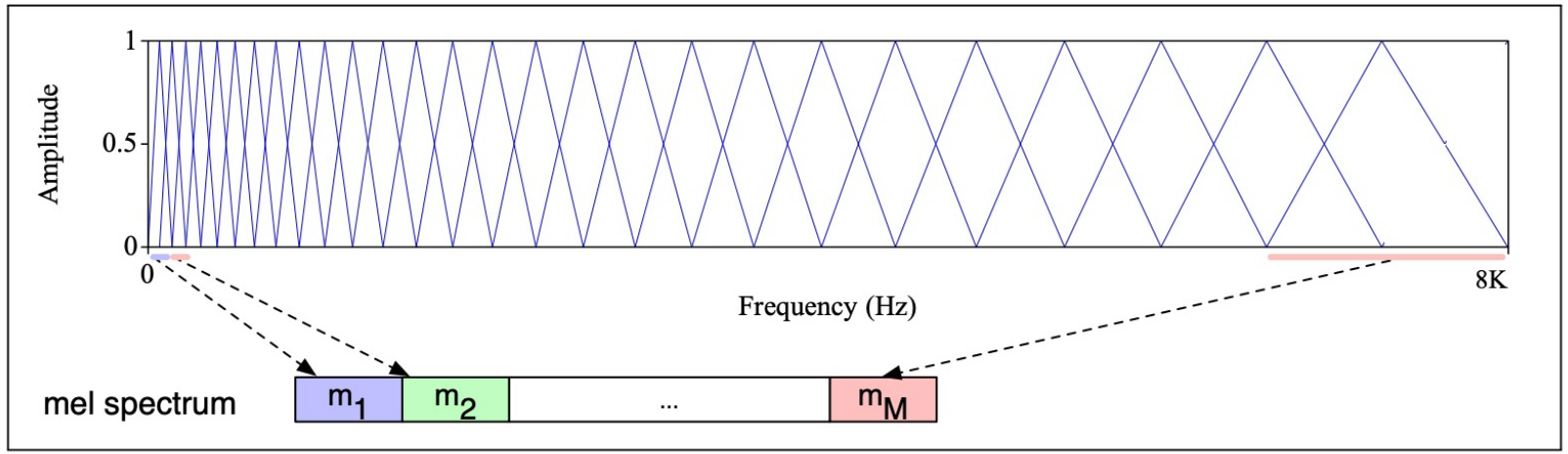
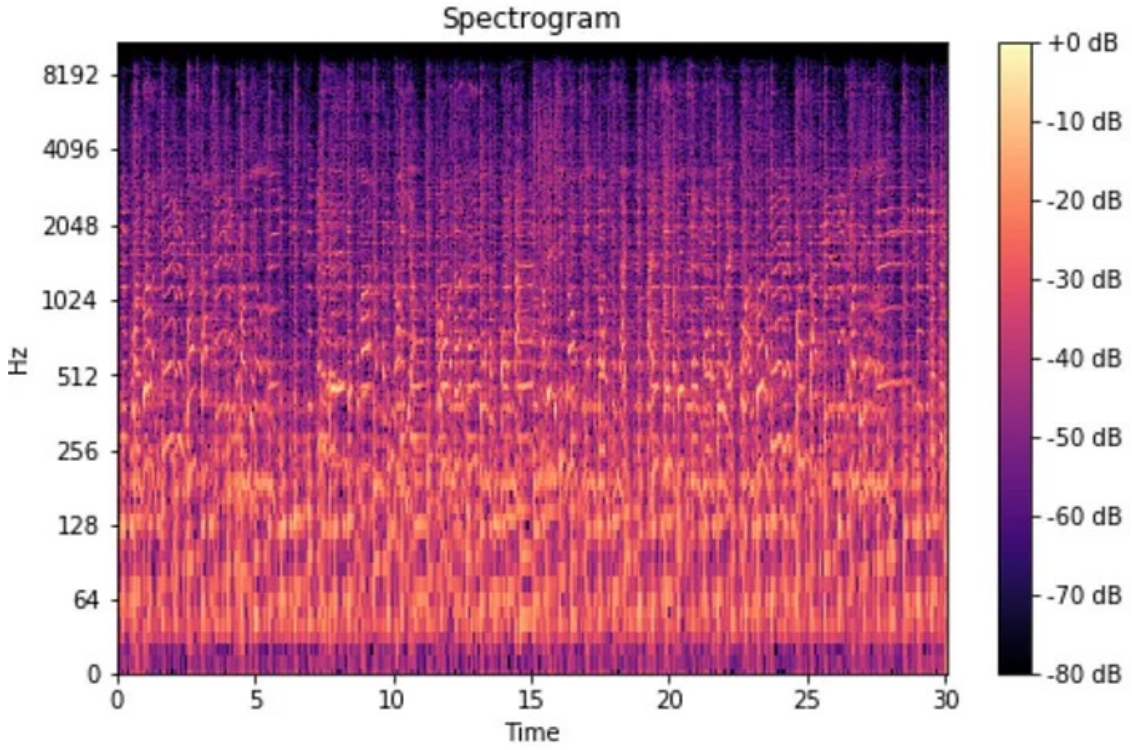


Figure 16.5 The mel filter bank (Davis and Mermelstein, 1980). Each triangular filter, spaced logarithmically along the mel scale, collects energy from a given frequency range.

Illustration from https://web.stanford.edu/~jurafsky/slp3/16.pdf

Before and after Mel



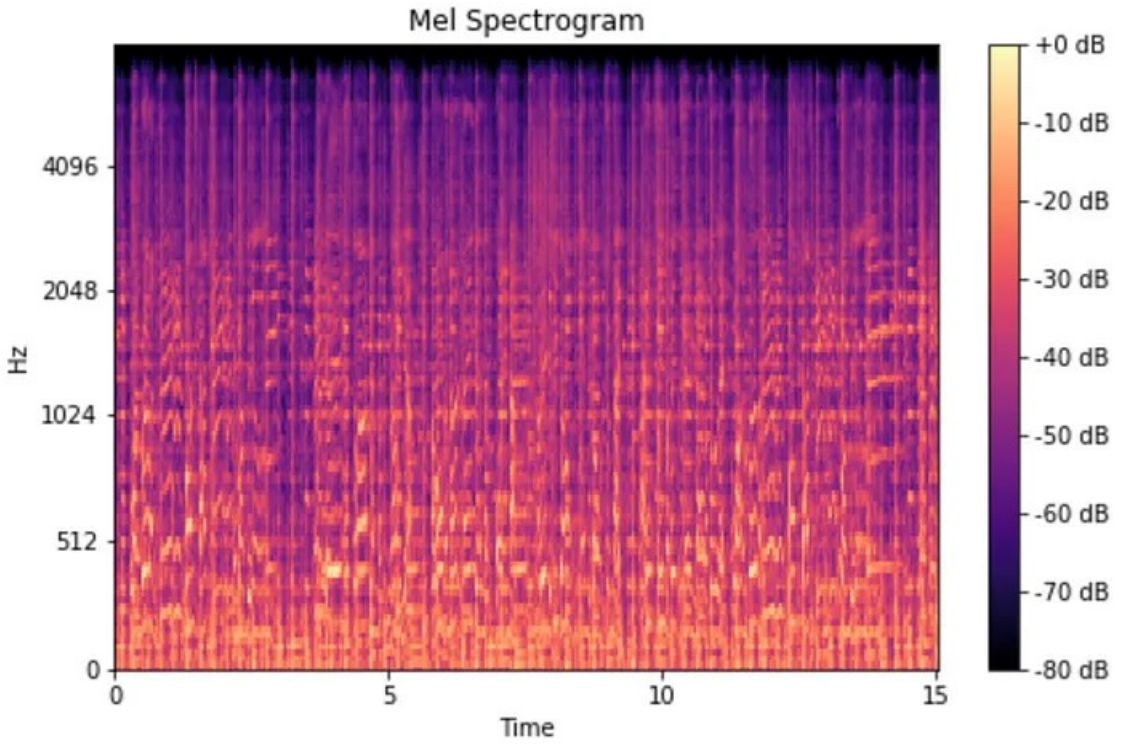


Illustration from https://medium.com/analytics-vidhya/understanding-the-mel-spectrogram-fca2afa2ce53

End-to-end ASR

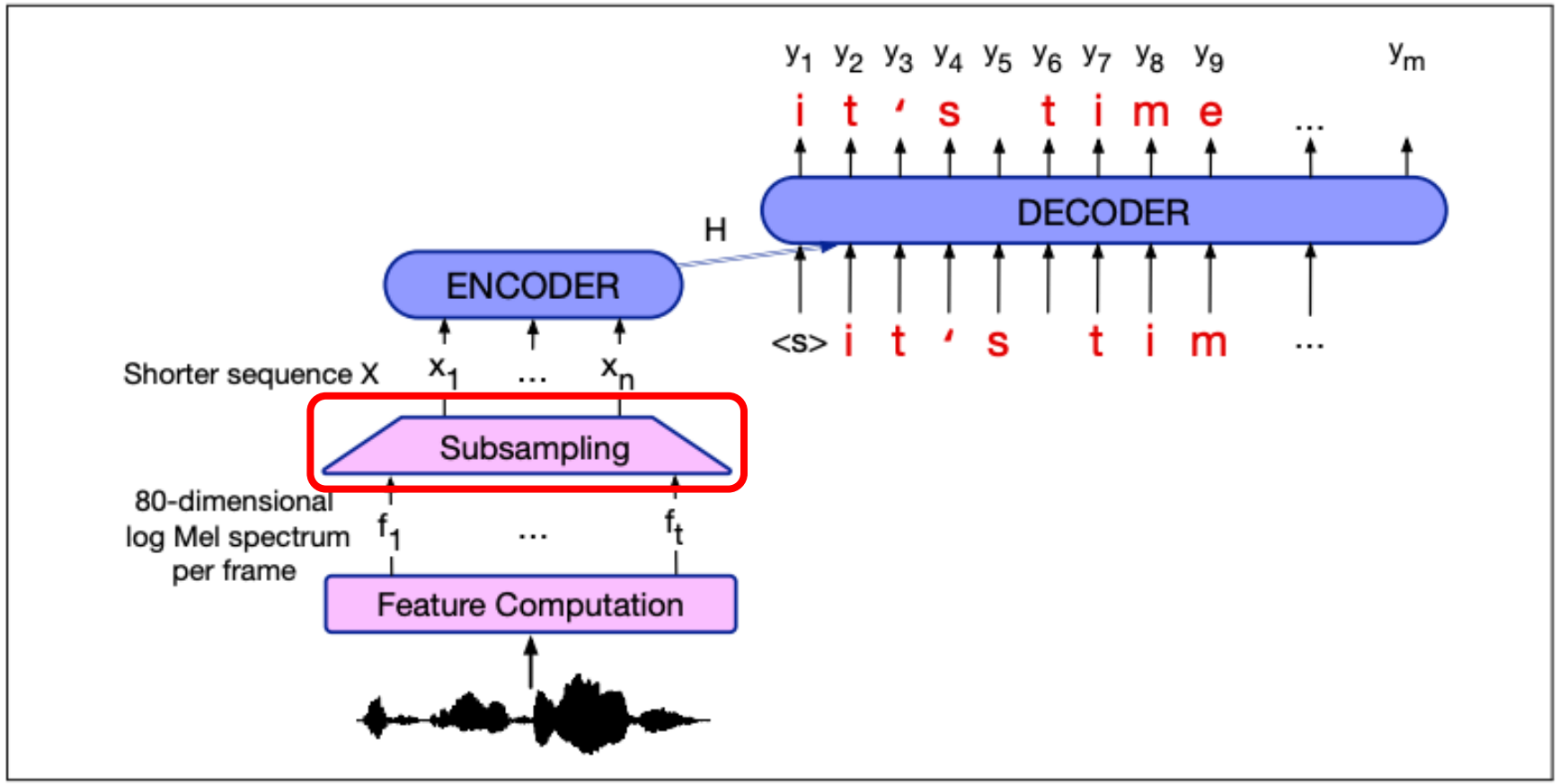


Figure 16.6 Schematic architecture for an encoder-decoder speech recognizer.

Illustration from https://web.stanford.edu/~jurafsky/slp3/16.pdf

Subsampling

- Input is very long sequence, e.g.,
  - 2s audio is 200 frames, assuming 10ms stride at windowing
- Ways to lower the frame rate:
  - Stack adjacent frames
  - 1D filter along time axis

End-to-end ASR

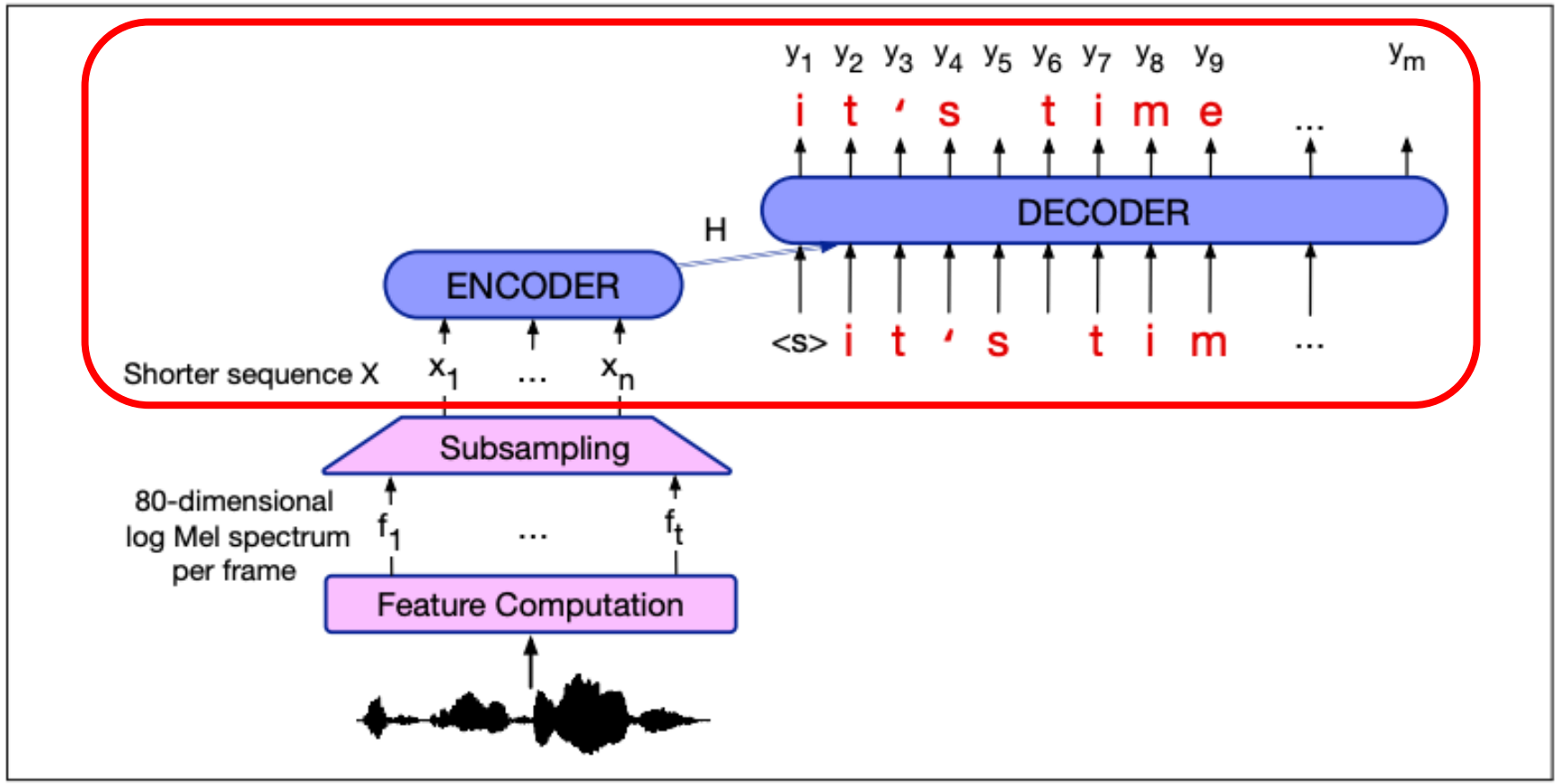
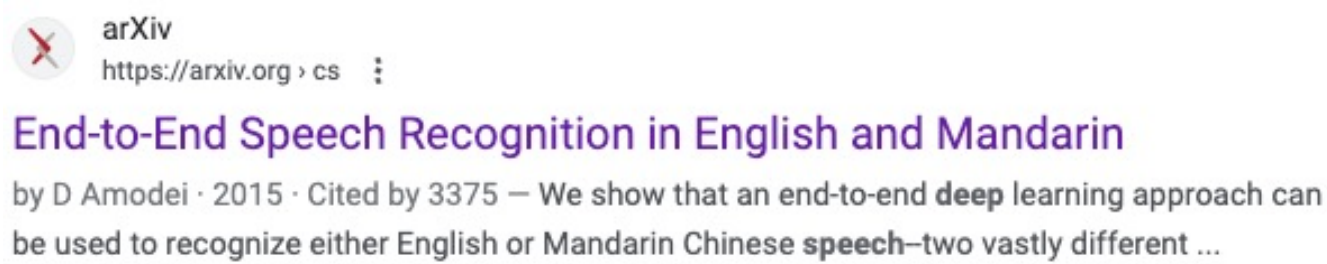


Figure 16.6 Schematic architecture for an encoder-decoder speech recognizer.

Illustration from https://web.stanford.edu/~jurafsky/slp3/16.pdf

What are the y’s

- Characters



- Words, less common

- Subwords, popular now

LSTM based Encoder-Decoder

LISTEN, ATTEND AND SPELL: A NEURAL NETWORK FOR LARGE VOCABULARY CONVERSATIONAL SPEECH RECOGNITION

William Chan — Carnegie Mellon University

Navdeep Jaitly, Quoc Le, Oriol Vinyals — Google Brain

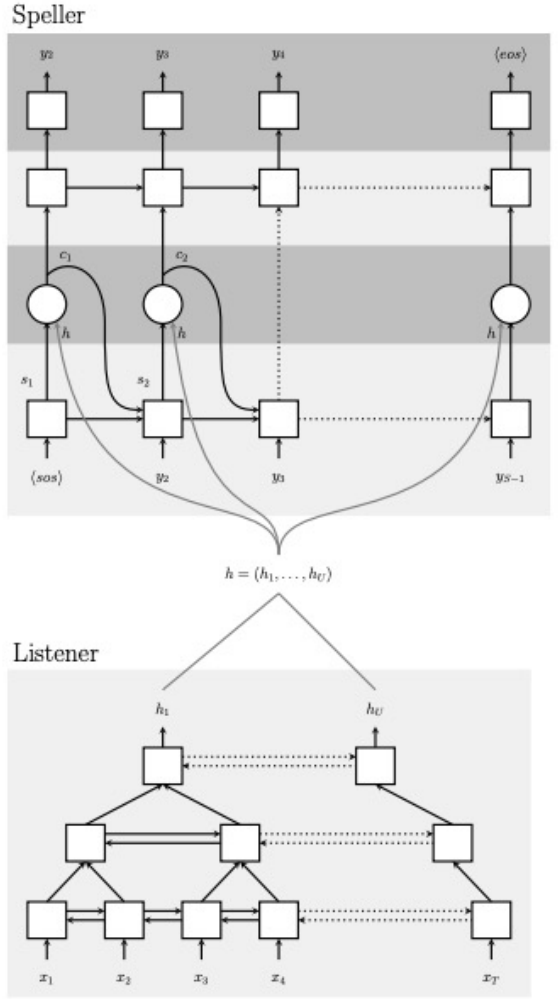


Fig. 1: Listen, Attend and Spell (LAS) model: the listener is a pyramidal BLSTM encoding our input sequence **x** into high level features **h**, the speller is an attention-based decoder generating the **y** characters from **h**.

Transformer based Encoder-Decoder

Conformer: Convolution-augmented Transformer for Speech Recognition

Anmol Gulati, James Qin, Chung-Cheng Chiu, Niki Parmar, Yu Zhang, Jiahui Yu, Wei Han, Shibo Wang, Zhengdong Zhang, Yonghui Wu, Ruoming Pang

Google Inc.

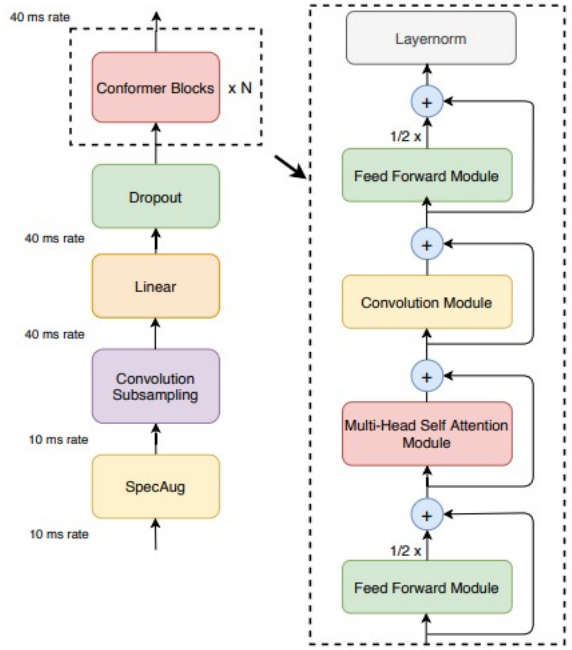


Figure 1: ***Conformer encoder model architecture.*** *Conformer comprises of two macaron-like feed-forward layers with half-step residual connections sandwiching the multi-headed self-attention and convolution modules. This is followed by a post layernorm.*

Table 2: *Comparison of Conformer with recent published models. Our model shows improvements consistently over various model parameter size constraints. At 10.3M parameters, our model is 0.7% better on testother when compared to contemporary work, ContextNet(S) [10]. At 30.7M model parameters our model already significantly outperforms the previous published state of the art results of Transformer Transducer [7] with 139M parameters.*

Method	#Params (M)	WER Without LM		WER With LM	
		testclean	testother	testclean	testother
Hybrid					
Transformer [33]	-	-	-	2.26	4.85
CTC					
QuartzNet [9]	19	3.90	11.28	2.69	7.25
LAS					
Transformer [34]	270	2.89	6.98	2.33	5.17
Transformer [19]	-	2.2	5.6	2.6	5.7
LSTM	360	2.6	6.0	2.2	5.2
Transducer					
Transformer [7]	139	2.4	5.6	2.0	4.6
ContextNet(S) [10]	10.8	2.9	7.0	2.3	5.5
ContextNet(M) [10]	31.4	2.4	5.4	**2.0**	4.5
ContextNet(L) [10]	112.7	**2.1**	4.6	**1.9**	4.1
Conformer (Ours)					
Conformer(S)	10.3	**2.7**	**6.3**	**2.1**	**5.0**
Conformer(M)	30.7	**2.3**	**5.0**	**2.0**	**4.3**
Conformer(L)	118.8	**2.1**	**4.3**	**1.9**	**3.9**

Decoding At Inference Time

- Greedy
  - Each time step take the most likely token and input to next step
- Beam Search
  - Recap

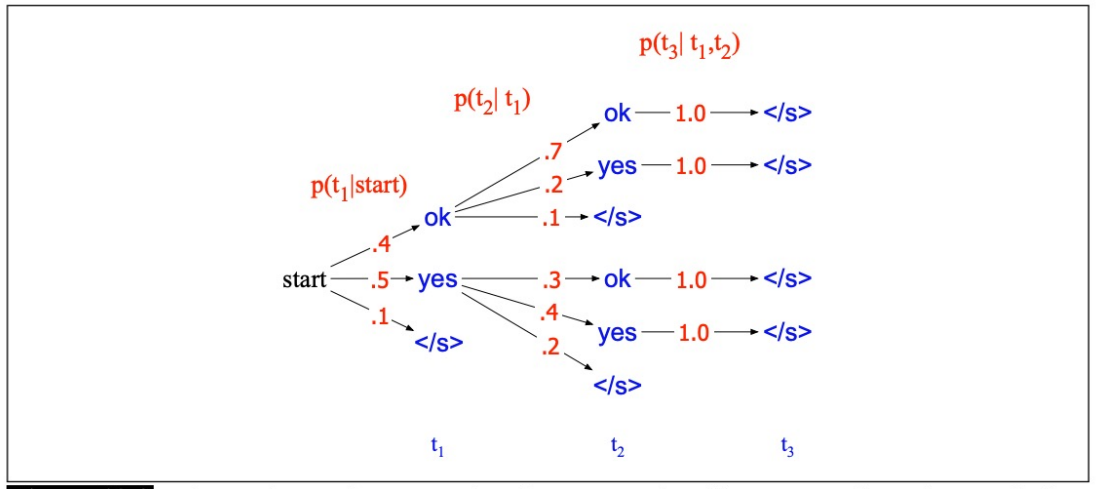


Figure 10.8 A search tree for generating the target string $T = t_1, t_2, \ldots$ from the vocabulary $V = \{\text{yes}, \text{ok}, \text{<s>}\}$, showing the probability of generating each token from that state. Greedy search would choose *yes* at the first time step followed by *yes*, instead of the globally most probable sequence *ok ok*.

Illustration from https://web.stanford.edu/~jurafsky/slp3/10.pdf

Connectionist Temporal Classification (CTC)

Connectionist Temporal Classification: Labelling Unsegmented Sequence Data with Recurrent Neural Networks

Alex Graves[1] ALEX@IDSIA.CH
Santiago Fernández[1] SANTIAGO@IDSIA.CH
Faustino Gomez[1] TINO@IDSIA.CH
Jürgen Schmidhuber[1,2] JUERGEN@IDSIA.CH
[1] Istituto Dalle Molle di Studi sull'Intelligenza Artificiale (IDSIA), Galleria 2, 6928 Manno-Lugano, Switzerland
[2] Technische Universität München (TUM), Boltzmannstr. 3, 85748 Garching, Munich, Germany

- Drawback of enc-dec architecture
  - Causal decoder is slow
- Number of Input speech frames >> Number of output tokens

Connectionist Temporal Classification (CTC)

- Introduce blank token "␣" for silence
- Same token could last for multiple frames

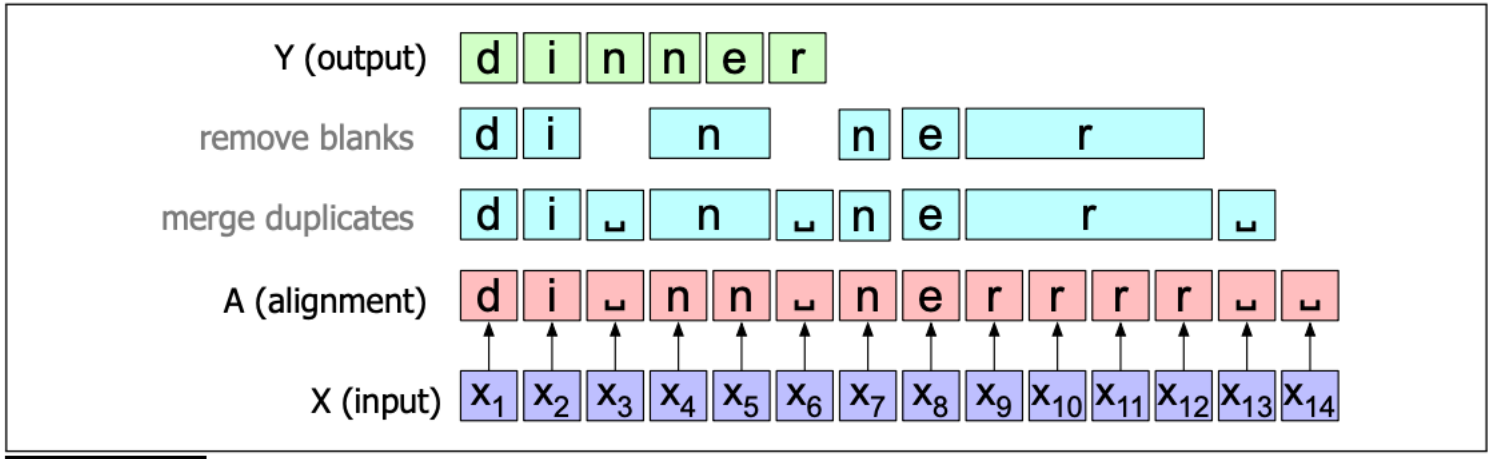


Figure 16.8 The CTC collapsing function B, showing the space blank character ␣; repeated (consecutive) characters in an alignment A are removed to form the output Y.

Illustration from https://web.stanford.edu/~jurafsky/slp3/16.pdf

Connectionist Temporal Classification (CTC)

- Alignment $A = [a_1, \dots, a_T]$
- Independence assumption

$$P_{\text{CTC}}(A|X) = \prod_{t=1}^{T} p(a_t|X)$$

- Training loss on (X, Y)

$$-\log \sum_{A:collapse(A)=Y} P_{CTC}(A|X)$$

Training with CTC

- Enumerating all alignments is infeasible

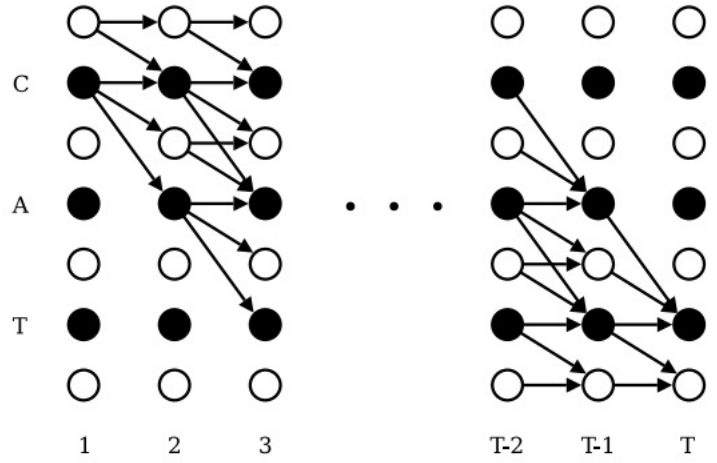


Figure 3. **illustration of the forward backward algorithm applied to the labelling 'CAT'**. Black circles represent labels, and white circles represent blanks. Arrows signify allowed transitions. Forward variables are updated in the direction of the arrows, and backward variables are updated against them.

Illustration from Graves *et. al*, 2006

Decoding with CTC

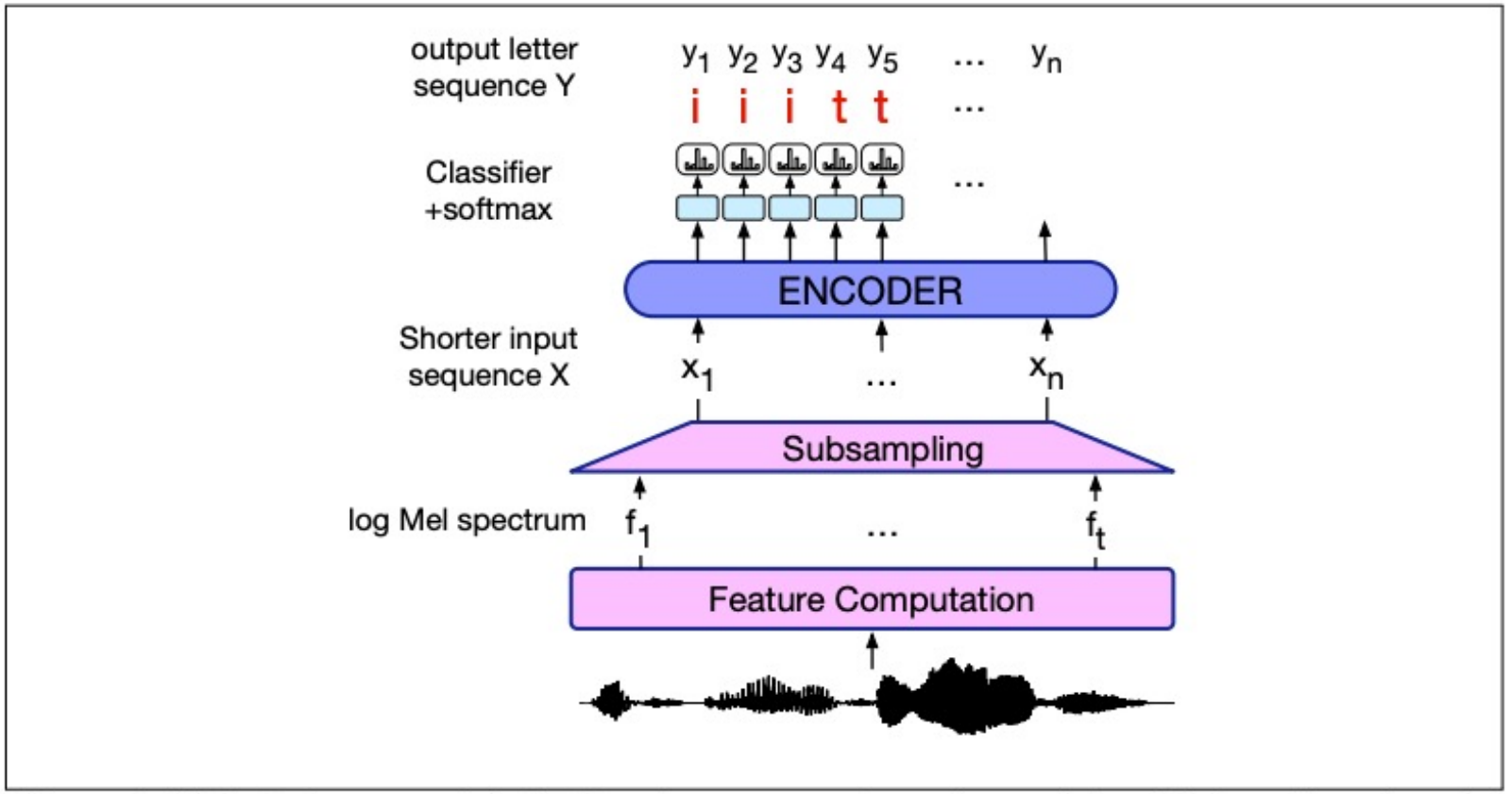


Figure 16.10 Inference with CTC: using an encoder-only model, with decoding done by simple softmaxes over the hidden state h_t at each output step.

Illustration from https://web.stanford.edu/~jurafsky/slp3/10.pdf

Other decoding issues

- Beam search can be applied
- Use a lexicon Trie to avoid obvious spelling errors

Discussion

- What are the key differences between CTC loss and enc-dec model?

Use Language Model

- “two” and “to” sounds alike
- How to make sure we decode the right one?
- Use Language Model
  - Easy way: rescoring N-best list
  - Hard way: add LM score at decoding
- We can finetune the language model for ASR

Large Margin Neural Language Model

Jiaji Huang[1] **Yi Li**[1] **Wei Ping**[1] **Liang Huang**[1,2*]
[1] Baidu Research, Sunnyvale, CA, USA
[2] School of EECS, Oregon State University, Corvallis, OR, USA

Evaluation Metric: Word Error Rate

- Edit (Levenshtein) distance between reference and decoded

$$\text{Word Error Rate} = 100 \times \frac{\text{Insertions} + \text{Substitutions} + \text{Deletions}}{\text{Total Words in Correct Transcript}}$$

- Implementation based on Dynamic programming

		H	Y	U	N	D	A	I
	0	1	2	3	4	5	6	7
H	1	0	1	2	3	4	5	6
O	2	1	1	2	3	4	5	6
N	3	2	2	2	2	3	4	5
D	4	3	3	3	3	2	3	4
A	5	4	4	4	4	3	2	3

$$\text{lev}_{a,b}(i,j) = \begin{cases} 0 & , i = j = 0 \\ i & , j = 0 \text{ and } i > 0 \\ j & , i = 0 \text{ and } j > 0 \\ \min \begin{cases} \text{lev}_{a,b}(i-1,j) + 1 \\ \text{lev}_{a,b}(i,j-1) + 1 \\ \text{lev}_{a,b}(i-1,j-1) + [a_i \neq b_j] \end{cases} & , \text{ else} \end{cases}$$

Example from https://www.cuelogic.com/blog/the-levenshtein-algorithm

Agenda

- Applications
  - Translation
  - Question Answering
- Other Modality
  - Speech to text
  - Text to Speech
  - Vision

Text to Speech

- Two steps:
  - Text to mel spectrum
  - mel spectrum to audio

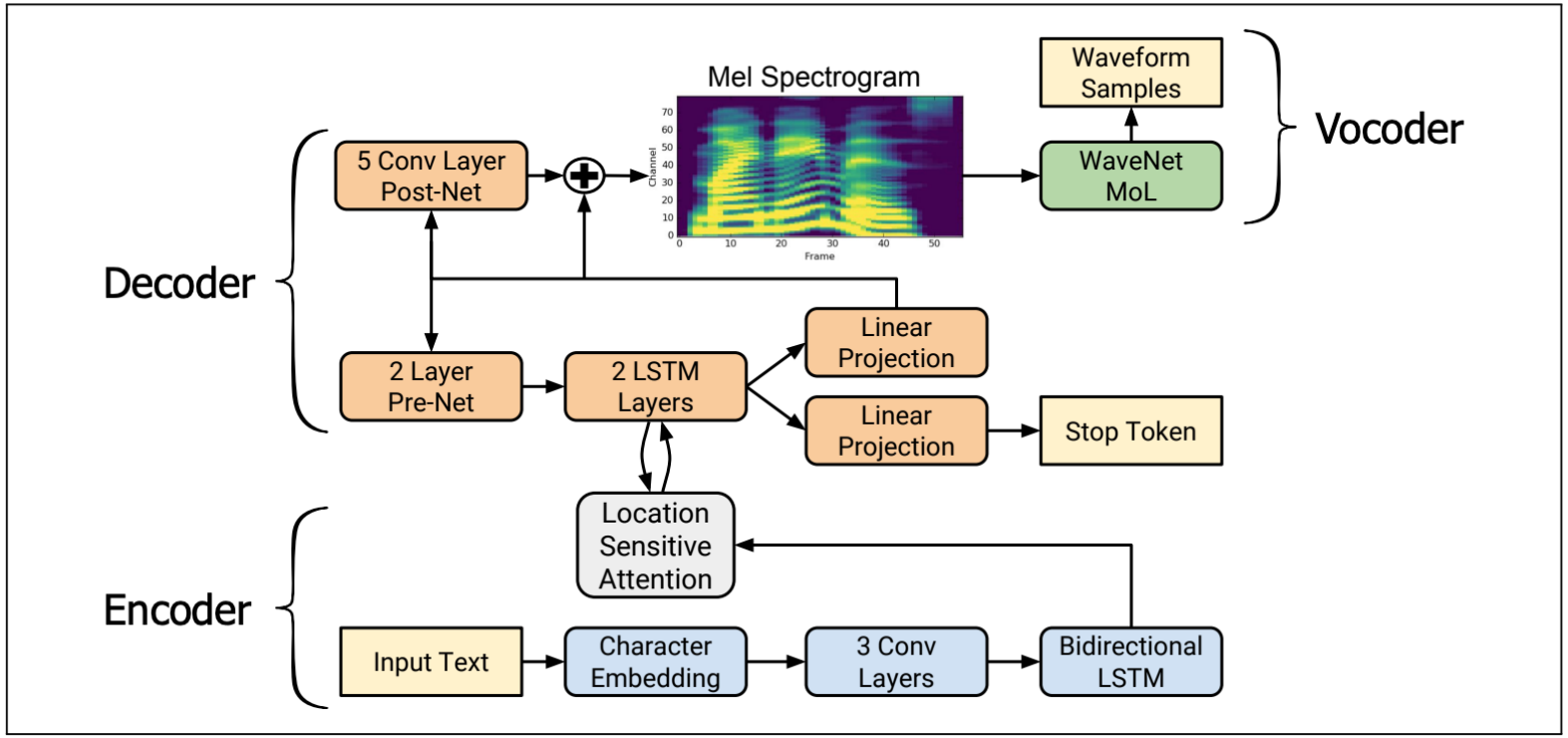


Figure 16.14 The Tacotron2 architecture: An encoder-decoder maps from graphemes to mel spectrograms, followed by a vocoder that maps to wavefiles. Figure modified from Shen et al. (2018).

Illustration from https://web.stanford.edu/~jurafsky/slp3/16.pdf

Vocoder

- Input: mel spectrogram
- Output: 8-bit mu-law audio samples

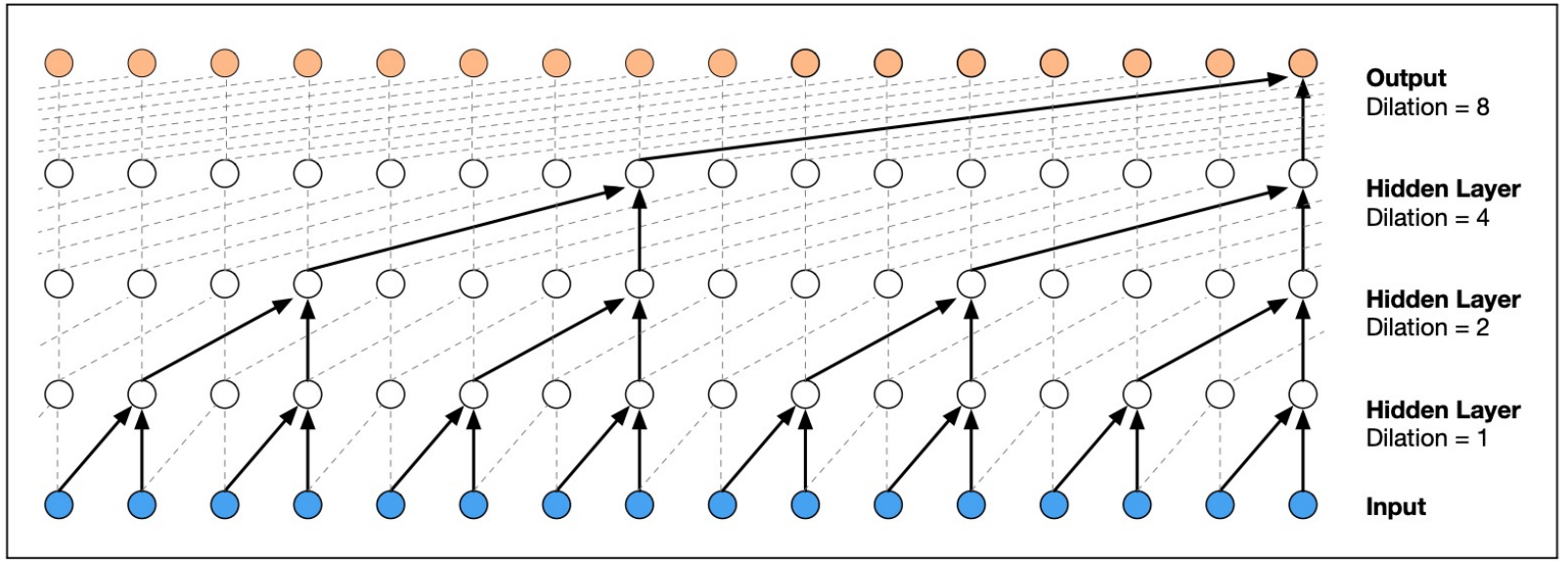


Figure 16.15 Dilated convolutions, showing one dilation cycle size of 4, i.e., dilation values of 1, 2, 4, 8. Figure from van den Oord et al. (2016).

Illustration from https://web.stanford.edu/~jurafsky/slp3/16.pdf

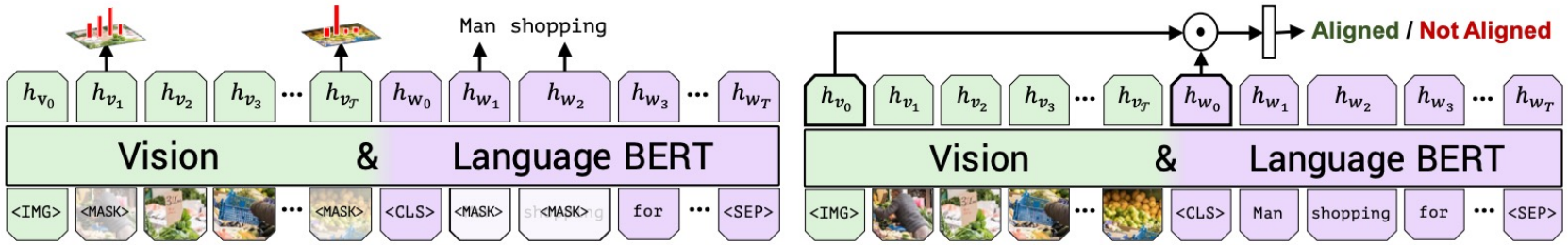
Agenda

- Applications
  - Translation
  - Question Answering
- Other Modality
  - Speech to text
  - Text to Speech
  - Vision

Joint modeling of Text and Images

ViLBERT: Pretraining Task-Agnostic Visiolinguistic Representations for Vision-and-Language Tasks

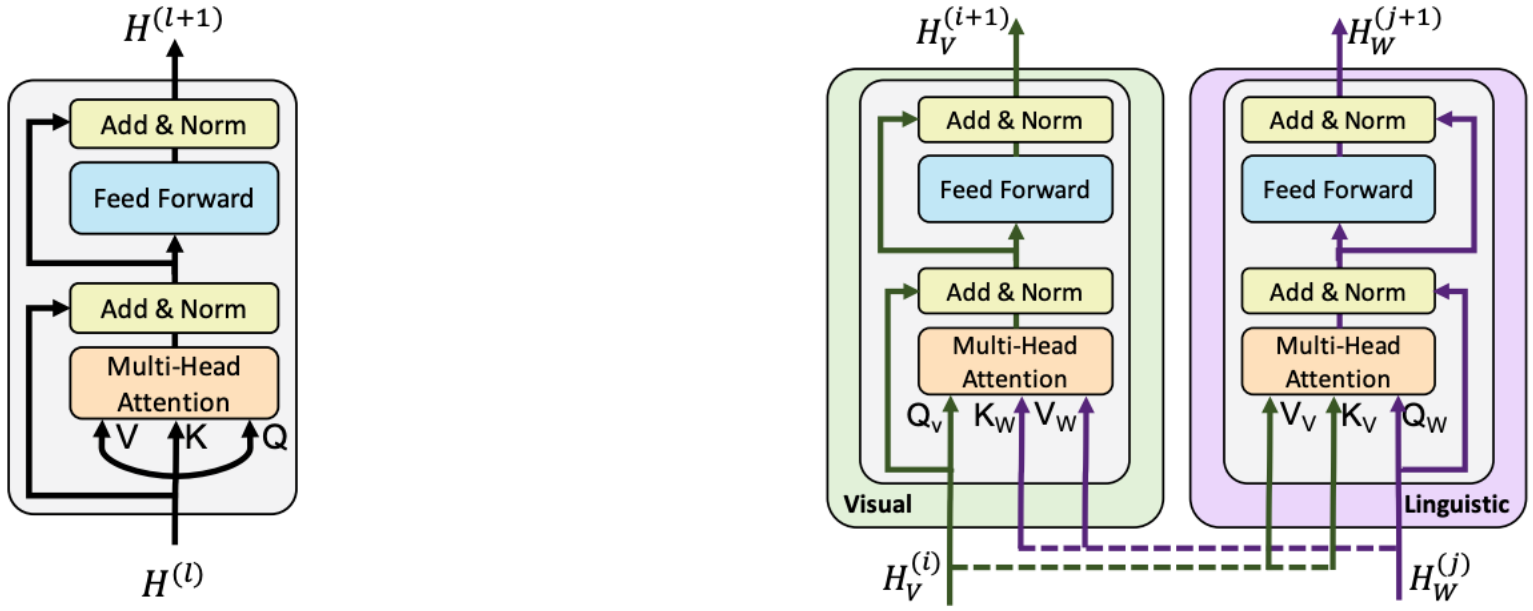
Jiasen Lu[1], **Dhruv Batra**[1,2], **Devi Parikh**[1,2], **Stefan Lee**[1,3]
[1]Georgia Institute of Technology, [2]Facebook AI Research, [3]Oregon State University



(a) Masked multi-modal learning (b) Multi-modal alignment prediction

Figure 3: We train ViLBERT on the Conceptual Captions [24] dataset under two training tasks to learn visual grounding. In masked multi-modal learning, the model must reconstruct image region categories or words for masked inputs given the observed inputs. In multi-modal alignment prediction, the model must predict whether or not the caption describes the image content.

Co-attention



(a) Standard encoder transformer block

(b) Our co-attention transformer layer

Figure 2: We introduce a novel co-attention mechanism based on the transformer architecture. By exchanging key-value pairs in multi-headed attention, this structure enables vision-attended language features to be incorporated into visual representations (and vice versa).

On Perplexity

Perplexity

- N classes, predicted probabilities $\{\hat{p}_i\}_{i=1}^N$
- Groundtruth probabilities $\{p_i\}_{i=1}^N$
- Cross entropy loss $\ell = -\sum_{i=1}^N p_i \ln \hat{p}_i$
- Perplexity e^{ℓ}
- Perplexity ≥ 1 as $\ell \geq 0$
- Higher perplexity, less accurate the model

Questions

- In a 3-way classification problem
- What would be the perplexity of worst classifier?
- If we know the 3rd class occurs twice more often the the 1st and 2nd class
- Can we build a classifier that reduces the perplexity, without any training data?