Machine Translation

Lori Levin and David Mortensen

March 14, 2023

Language Technologies Institute

Translation

➜ Mapping a "text" in a source language to a target language

"I went to the store to buy eggs" ⟶ *"Eu fui à loja comprar ovos"*

When did people start using computers to translate?

- Roughly around World War II
- Research stopped in the US for about 15-20 years after a 1967 report deemed it impossible
- Research resumed in the US in the early 1980s

What did early MT systems look like?

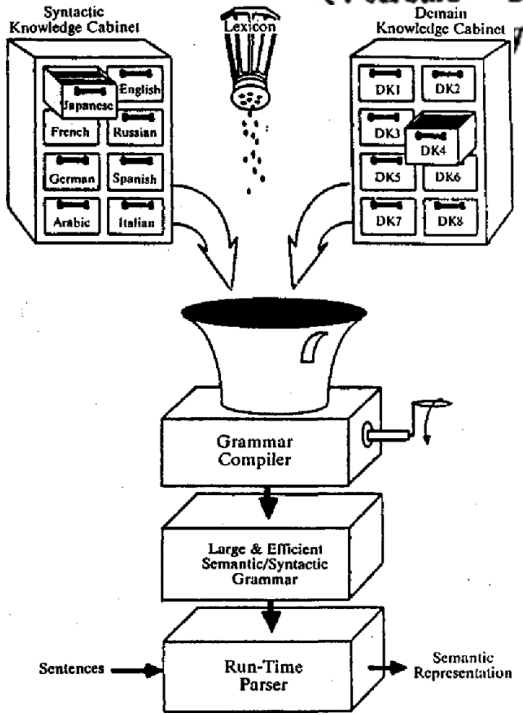
```
(<S> <--> (<V>)
     ((x0 = x1)))

(<S> <--> (<NP> <S>)
     (((x2 subj-case) = *defined*)
      ((x2 subj-case) = (x1 case))
      (x0 = x2)
      ((x0 subj) = x1)))

(<S> <--> (<NP> <S>)
     (((x2 obj-case) = *defined*)
      ((x2 obj-case) = (x1 case))
      (x0 = x2)
      ((x0 obj) = x1)))

(<S> <--> (<NP> <S>)
     (((x2 obj2-case) = *defined*)
      ((x2 obj2-case) = (x1 case))
      (x0 = x2)
      ((x0 obj2) = x1)))
```

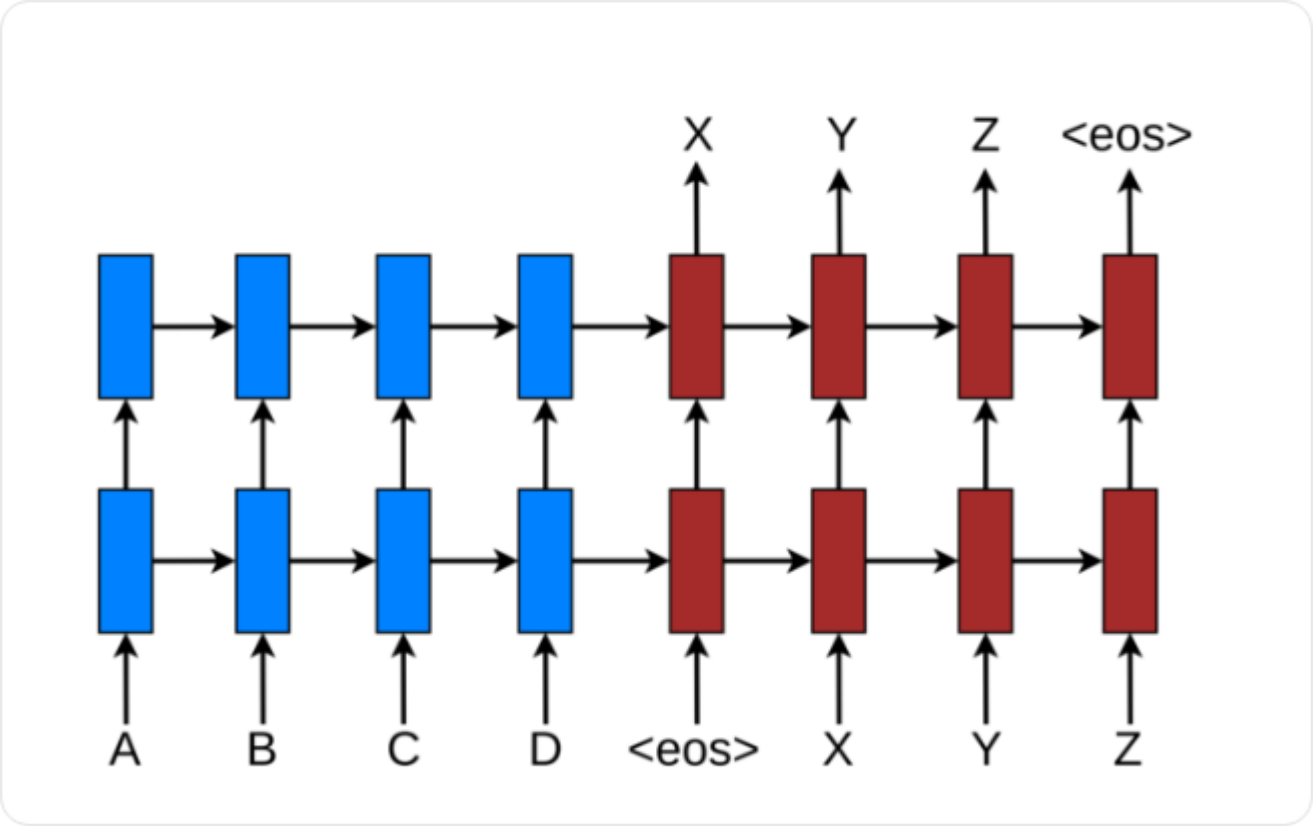
```
(emap *insert
   <=1=> insert ((CAT v) (SUBCAT trans))
   (role =sem (*physical-action))
   (:agent =syn (SUBJECT))
   (:theme =syn (DOBJECT))
   (:theme =syn (PPADJUNCT
      ((PREP into) (CAT n)))))
```



Human linguists wrote rules. And we had to use clever pictures to explain to people what we were doing.

The Language Technologies Institute was founded as the Center for Machine Translation by Jaime Carbonell and Masaru Tomita in the 1980's.

What will your homework look like?



Learning to translate from data

Since the late 1980's, Machine Translation researchers have been using *parallel corpora* to train Machine Translation systems. This is what we will be talking about today, and in the next lecture.

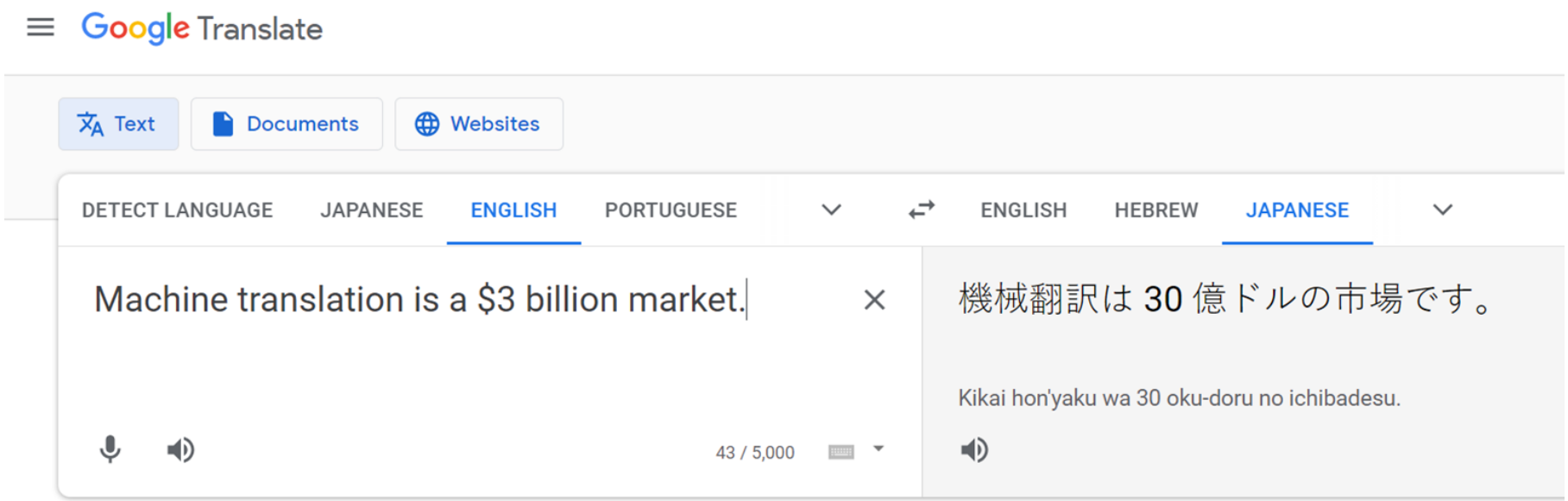
	ENGLISH	MANDARIN
1	i **wanna** live in a wes anderson world	我想要生活在Wes Anderson的世界里
2	Chicken soup, corn never truly digests. **TMI**.	鸡汤吧，玉米神马的从来没有真正消化过.恶心
3	To DanielVeuleman **yea iknw imma** work on that	对DanielVeuleman说，是的我知道，我正在向那方面努力
4	**msg 4** Warren G his **cday** is today 1 **yr** older.	发信息给Warren G，今天是他的生日，又老了一岁了。
5	Where **the hell** have you been all these years?	这些年你**TMD**到哪去了

	ENGLISH	ARABIC
6	It's **gonna** be a warm week!	الاسبو ع **الياي** حر
7	onni this gift only **4 u**	أوني هذة الهدية فقط لك
8	sunset in aqaba **:)**	غروب الشمس في العقبة:)
9	RT @MARYAMALKHAWAJA: there is a call for widespread protests in #bahrain **tmrw**	هناك نداء لظاهرات في عدة مناطق غدا

Wang Ling, *Microblogs as Parallel Corpora*, 2013

Machine Translation is a $3 billion market

Translation of text



Machine Translation is a $3 billion market

Translation of speech

Lori: Alexa, how do you say, “I hate this movie” in Japanese.

Alexa: “I hate this movie” in Japanese is “Kono eiga wa kirai da.”

Lori: Alexa, how do you say, “I hate this movie in Japanese” in Japanese.

Alexa: “I hate this movie in Japanese” in Japanese is “Kono eiga wa nihongo de wa kirai da.”

More impressively, real time translation of meetings is also now viable.

Most translation is still done by human translators

Translation and Localization Industry Grows 11.8% in 2021 to USD 26.6bn

https://slator.com/translation-localization-industry-grows-in-2021-to-usd-26bn/

What kinds of translations must be done by humans or at least checked and corrected by humans?

Checking and correcting of machine translation by humans is called *post-editing*.

Evacuation Ladder

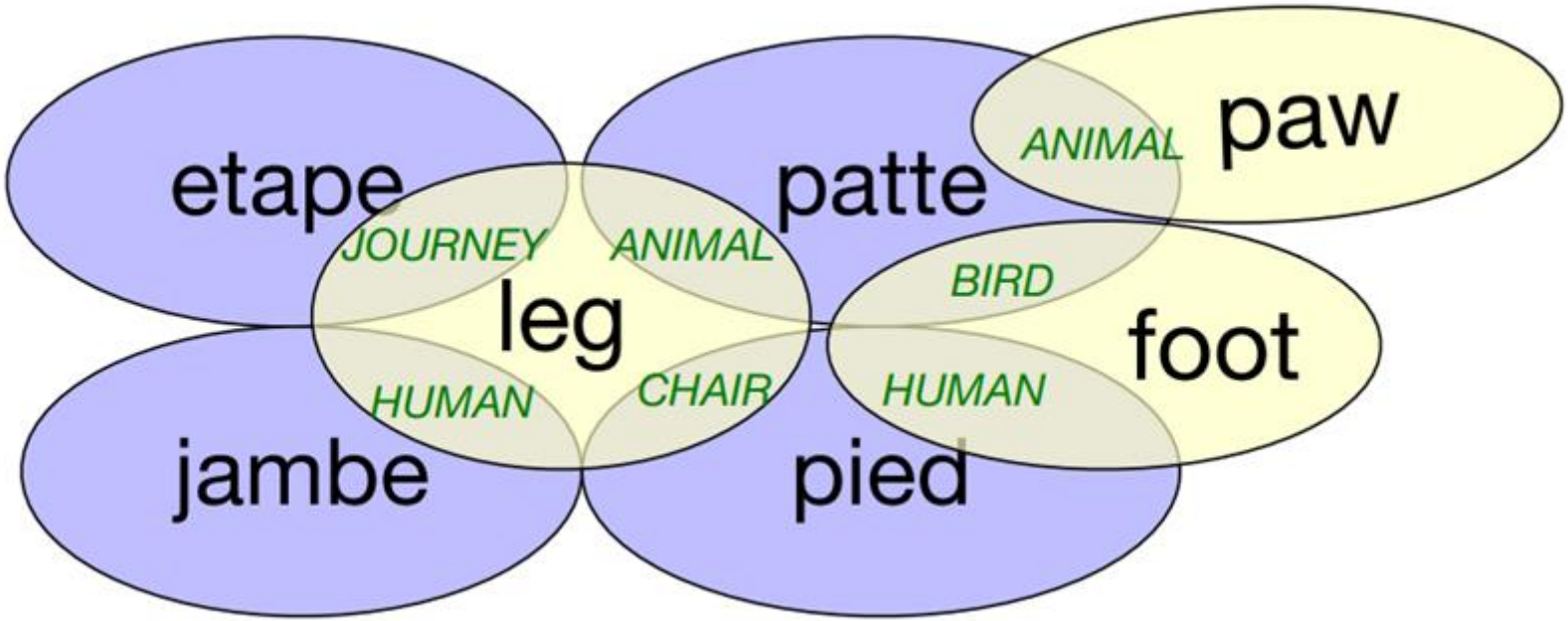
Don’t yell

https://www.languageconnections.com/blog/7-hilarious-machine-translation-mistakes/

Why is translation difficult?

Why not just look up each word in a dictionary and translate word-by-word?

This is what you get when you look words up in a dictionary (English and French)



Jurafsky & Martin *Speech and Language Processing* 2nd ed.

Many-to-many mapping of words

Why not translate word-by-word

- The grammars of some languages make distinctions that other languages don’t make:

- Russian *kniga* translates to English as *the book* or *a book.*
  - English grammar makes a distinction in definiteness. Russian grammar does not.
- English *it* translates to French il/le (masculine) or elle/la (feminine).
- English a translates to French as un (masculine) or une (feminine).
  - *Une chaise* (a chair) vs *un livre* (a book)
  - French grammar makes a distinction in Gender. English grammar does not.

Why not translate word-by-word:
Using different numbers of words to say the same thing

uygarlaştıramadıklarımızdanmışsınızcasına

"(behaving) as if you are among those whom we were not able to civilize"

uygar	"civilized"
+laş	"become"
+tır	"cause to"
+ama	"not able"
+dık	past participle
+lar	plural
+ımız	first person plural possessive ("our")
+dan	ablative case ("from/among")
+mış	past
+sınız	second person plural ("y'all")
+casına	finite verb → adverb ("as if")

This produces a sparcity problem (if, e.g., you have a one-hot encoding of word-sized tokens).

Why not translate word-by-word: the words are in a different order.

(10.3) English: *He wrote a letter to a friend* **verb-medial**

Japanese: *tomodachi ni tegami-o kaita* **verb-final**
friend to letter wrote

Arabic: *katabt risāla li ṡadq* **verb-initial**
wrote letter to friend

What differences do you see in the order of words and the number of words?

Grammar also doesn’t match up nicely

יש ספר לתלמיד

יש ספר על השולחן

How do you solve this?

There are 3,344,720 speakers of ***Tajik*** in Tajikistan (one of the Central Asian republics of the former Soviet Union) and another million speakers in surrounding countries.

дуусти хуби ҳамсоай сумо	a good friend of your neighbor
ҳамсоай дуусти хуби сумо	a neighbor of your good friend
ҳамсоай хуби дуусти сумо	a good neighbor of your friend

Above are three phrases in Tajik with their English translations. Your task is to give the English translations of all four Tajik words. The possibilities are simply "good," "friend," "neighbor," and "your." The order of the words – which is not the same order as in English! – does the rest.

дуусти	_____
ҳамсоай	_____
хуби	_____
сумо	_____

Problem by Adriana Solovyova.

What insight does it give you into machine translation?

What is difficult about translation?

Lori's opinion as a linguist and someone who built MT rules and lexicons by hand for more than 20 years

- People in NLP and MT have reduced "language divergences" to six major word order features from WALS, or seven lexical features (Jurafsky and Martin section 10.1.2).
- Other people talk about language typology as a system of "morphosyntactic strategies", of which there is not a limited number.
  - An example of a morphosyntactic strategy follows on the next slide

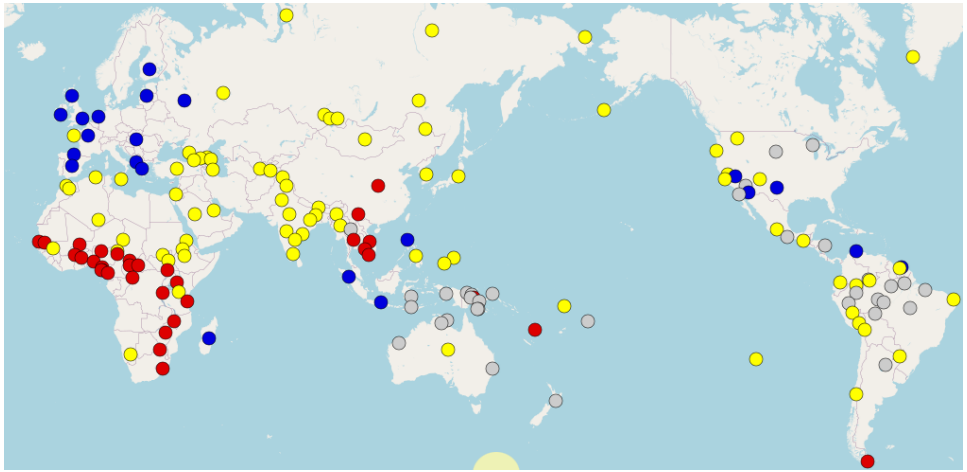
THE WORLD ATLAS
OF LANGUAGE STRUCTURES
ONLINE

Home | Features | Chapters | Languages | References | Authors

Feature 121A: Comparative Constructions

Values

●	Locational	78
●	Exceed	33
●	Conjoined	34
●	Particle	22



Yellow dot languages say: X is big from Y, or X is big to Y
Red dot languages say: X is big, exceeds Y
Grey dot languages say: X is big, Y is small
Blue dot languages say: X is big than Y

There are many variations within each color.
And there are thousands of things like this.

But the picture is not so gloomy

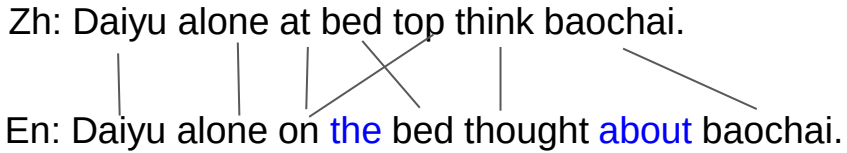
- In spite of the gloomy perspective, MT researchers have made progress on using data from typologically similar languages to improve MT, or by using a multilingual model trained on many typologically different languages.

Why is translation difficult?

錨玉自在枕上感念寶釵。。。又聽見窗外竹梢焦葉之上,

dai yu zi zai zhen shang gan nian bao chai...you ting jian chuang wal zhu shao xiang ye

From *"Dream of the Red Chamber"*, Cao Xue Qin (1792)



Literal translation: but we can't limit ourselves to literal translation because of availability.

Zh: Chinese
En: English

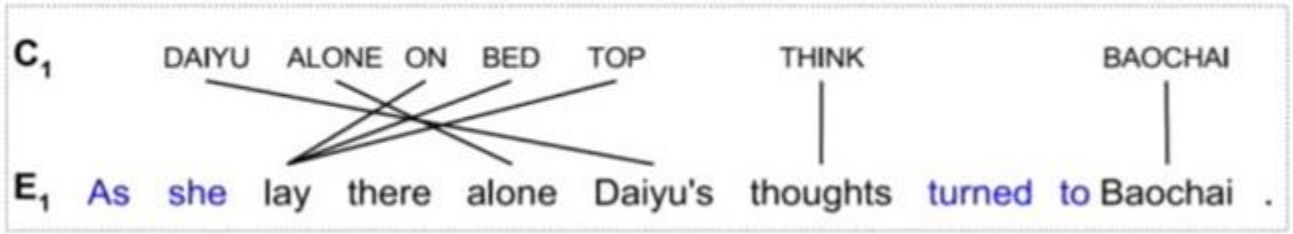
Patrick Fernandez, Graham Neubig, Xinyi Wang

Why is translation difficult?

錨玉自在枕上感念寶釵。。。又聽見窗外竹梢焦葉之上,

dai yu zi zai zhen shang gan nian bao chai...you ting jian chuang wal zhu shao xiang ye

From *"Dream of the Red Chamber"*, Cao Xue Qin (1792)



Literary translation: Parallel data is more likely to be like this.

Zh: Chinese
En: English

Patrick Fernandez, Graham Neubig, Xinyi Wang

Preparing for Machine Translation

- Collect a parallel corpus
- Align sentences (not covering: see Jurafsky and Martin section 10.7.2)
- Tokenization (not covering: see Jurafsky and Martin Chapter 2 and section 10.7.1)
    - Split words into sub-word units, e.g., using BPE (Byte Pair Encoding)

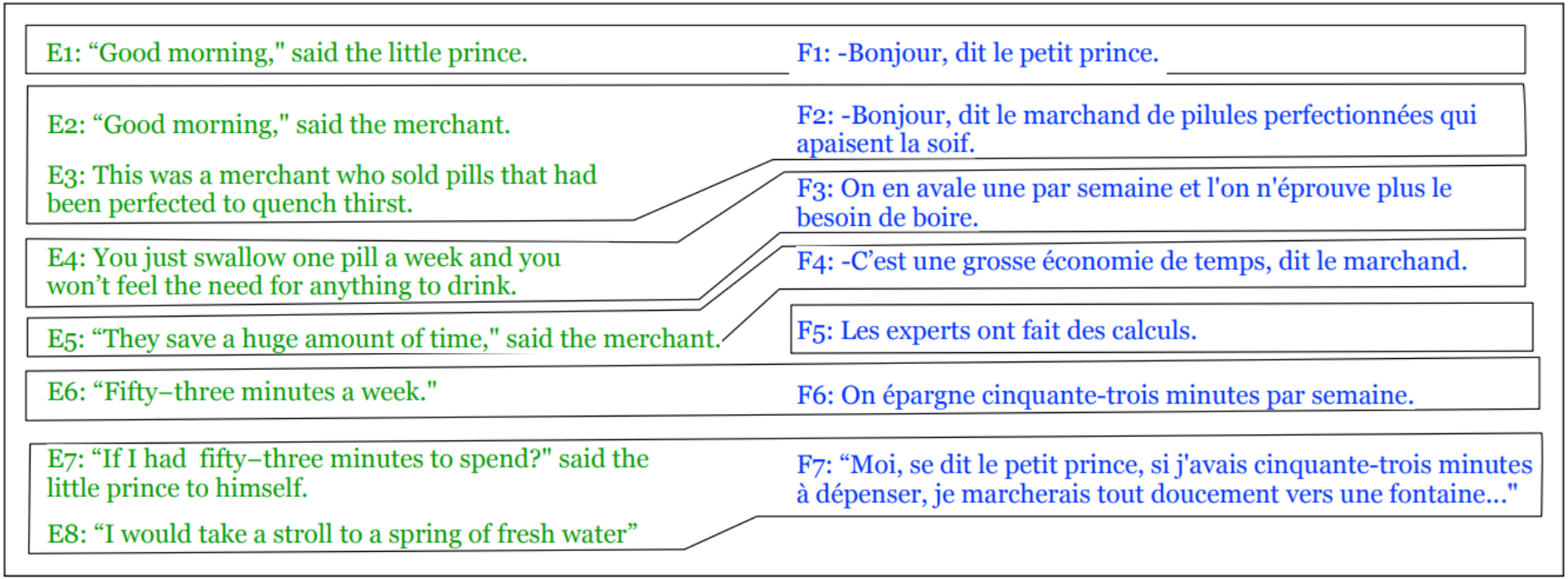


Figure 10.17 A sample alignment between sentences in English and French, with sentences extracted from Antoine de Saint-Exupery’s *Le Petit Prince* and a hypothetical translation. Sentence alignment takes sentences $e_1, ..., e_n$, and $f_1, ..., f_n$ and finds minimal sets of sentences that are translations of each other, including single sentence mappings like (e_1,f_1), (e_4,f_3), (e_5,f_4), (e_6,f_6) as well as 2-1 alignments (e_2/e_3,f_2), (e_7/e_8,f_7), and null alignments (f_5).

Acquiring a parallel corpus

- Examples (Jurafsky and Martin, section 10.7.2)
  - Europarl: Proceedings of the European Parliament; 21 languages; up to 2 million sentences
  - United Nations Parallel Corpus: 10 million sentences in Arabic, Chinese, English, French, Russian, Spanish
  - OpenSubtitles: movie and TV subtitles
  - ParaCrawl: 223 million sentences in 23 EU languages
- What about the other 7000 languages?
  - For many languages, the only parallel text is the Christian Bible.
  - Low-resource MT is a large area of research
    - How to leverage monolingual texts (backtranslation)
    - Humans in the loop
    - Leverage multilingual models

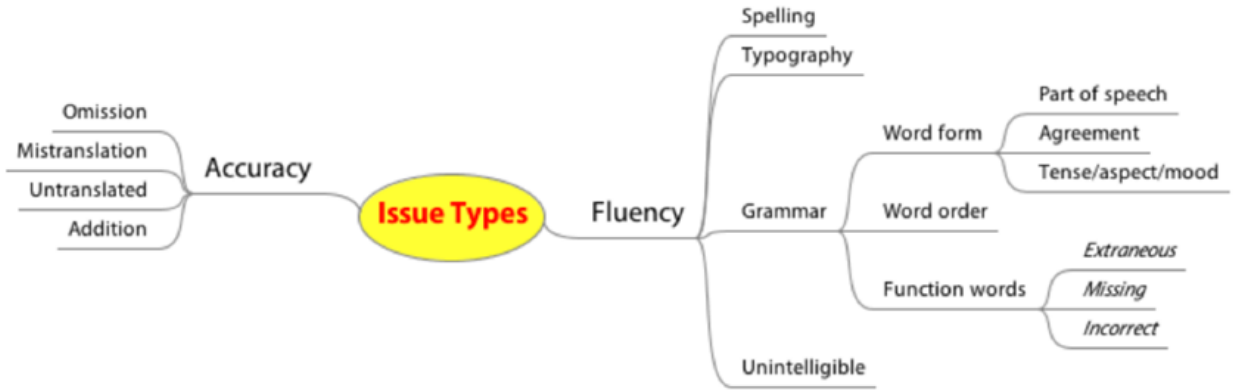
How to evaluate machine translation?

How do you know if a translation is good?

Ask humans to evaluate machine translation output

- ➔ Ask humans to *rate* translations
  - ◆ Adequacy and Fluency

  - ◆ Multidimensional Quality Metrics (MQM)



Evaluation by humans is not practical

When you are in a research-and-development cycle.

As you iterate through versions of your system, you need a faster way to know how well you are doing.

Evaluating machine translation

Machine Translation output is evaluated by comparing it to translations that are produced by human translators.

The translations that are produced by human translators are called “reference translations”.

From: https://slidetodoc.com/evaluating-the-output-of-machine-translation-systems-alon/

Reference: (Produced by a human translator)

The Iraqi weapons are to be handed over to the army within two weeks

MT Output:

In two weeks Iraq’s weapons will give army

BLEU (Papineni et al, ACL-2002)

Start by counting how many n-grams in the MT output are also found in a reference translation.

MT output 1: A cat sat on the mat.

*Reference 1: **The cat** is **on** the **mat**.*

*Reference 2: There is **a** cat on the mat.*

Unigrams: 5/6
A yes
Cat yes
Sat no
On yes
The yes
Mat yes

Bigrams: 3/5
A cat yes
Cat sat no
Sat on no
On the yes
The mat yes

Trigrams: 1/4
A cat sat no
Cat sat on no
Sat on the no
On the mat yes

A metric should correlate with human judgement: if a human judges a translation to be better than another, the metric should give it a better score.

MT output 1: A cat sat on the mat.

*Reference 1: **The cat** is **on** the **mat**.*

*Reference 2: There is **a** cat on the mat.*

MT output 2: Cats stay in the living room.

MT output 2 has fewer n-grams in common with the reference translations, and it is clearly not an accurate translation. So this is good. Counting n-grams confirms our judgement that MT output 2 is worse than MT output 1.

Some MT outputs cause problems for n-gram counting

MT output 3 has 10/10 matching unigrams, and 4/9 matching bigrams.

MT output 1: A cat sat on the mat.

*Reference 1: **The cat** is **on** the **mat**.*

*Reference 2: There is **a** cat on the mat.*

MT output 3: The cat the cat the cat the the cat cat.

Clipping the counts: only count an n-gram up to the maximum number of times it occurs in any one reference translation.

The can be counted twice
Cat can be counted once
The cat can be counted once

Brevity Penalty

MT output 4: The cat

*Reference 1: **The cat** is **on** the **mat**.*

*Reference 2: There is **a** cat on the mat.*

$$\text{BP} = \begin{cases} 1 & \text{if } c > r \\ e^{(1-\frac{r}{c})} & \text{if } c \leq r \end{cases}$$

c is the length of the MT candidate translation
r is the closest reference translation length

The Brevity Penalty is less than 1 when the MT output is short compared to the reference.

We want BP to be less than 1 because we will multiply it by the BLEU score to decrease the BLEU score.

BLEU Score

$$\mathrm{BLEU} = \mathrm{BP} \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right)$$

BP is the Brevity Penalty
N is the longest n-gram used in the metric (usually 4)
although it is 3 in the example on the right
w_n is usually ¼
1/3 in the example on the right
p_n is the precision of n-grams of length n
as shown on the right

Unigrams: ⅝
A yes
Cat yes
Sat no
On yes
The yes
Mat yes

Bigrams: ⅗
A cat yes
Cat sat no
Sat on no
On the yes
The mat yes

Trigrams: ¼
A cat sat no
Cat sat on no
Sat on the no
On the mat yes

Issues with BLEU

- Based on precision
  - Is each n-gram in the candidate translation in a reference translation
- But not based on recall
  - Which n-grams in the reference translations are included in the candidate translation

Recall

Unigrams: 6/8

The	yes
The	yes
Cat	yes
Is	no
On	yes
Mat	yes
There	no
A	yes

Bigrams: 3/9

The cat	no
Cat is	no
Is on	no
On the	yes
The mat	yes
There is	no
Is a	no
A cat	yes
Cat on	no

Trigrams: 2/8

The cat is	no
Cat is on	no
Is on the	no
On the mat	yes
There is a	no
Is a cat	no
Cat on the	no
On the mat	yes

MT output 1: A cat sat on the mat.

*Reference 1: **The cat** is **on** the **mat**.*

*Reference 2: There is **a** cat on the mat.*

Precision

Unigrams: 5/6

A	yes
Cat	yes
Sat	no
On	yes
The	yes
Mat	yes

Bigrams: 3/5

A cat	yes
Cat sat	no
Sat on	no
On the	yes
The mat	yes

Trigrams: 1/4

A cat sat	no
Cat sat on	no
Sat on the	no
On the mat	yes

BLEU counts *words* and n-grams of *words*

- Doesn’t work as well on languages with a lot of morphology
  - Spanish dormir (sleep): duermo, duermes, duerme, dormimos, dormís, duramen, duerma, duermas, durmamos, durmáis, durman, dormí, dormiste, durmió, dormimos, dormisteis, durmieron, and many more.

BLEU counts *words* and n-grams of *words*

- It can be affected by how the words are tokenized
  - Italian: Il tavolo è **nella** stanza
  - the table is in-the room
  - Some tokenizers break *nella* into two words and some don’t.

  - Arabic: أعطها للطالب
  - 'aetaha liltaalib
  - give-it to-the-student
  - Some tokenizers break *'aetaha* and *liltaalib* into *aeta ha* and *l il taalib* and some don’t.

  - For example, some word tokenizers take prepositions off of the beginning of Italian and Arabic words, and some don’t.

chrF: Character F-Score

- Remove spaces between words
- Count *character* n-gram overlaps between the candidate translation and the reference translation
- Compute the average of both precision and recall

chrP percentage of character 1-grams, 2-grams, ..., k-grams in the hypothesis that occur in the reference, averaged.

chrR percentage of character 1-grams, 2-grams,..., k-grams in the reference that occur in the hypothesis, averaged.

chrF: Character F-Score

- Parameter, k, is the longest n-gram you want to consider
- Weighting parameter β is usually set to 2
  - Weights recall twice as much as precision

$$\text{chrF}\beta = (1+\beta^2)\frac{\text{chrP}\cdot\text{chrR}}{\beta^2\cdot\text{chrP}+\text{chrR}}$$

chrF: Character F-Score

The textbook walks through this example in section 10.8.2

REF:	witness for the past,	
HYP1:	witness of the past,	chrF2,2 = .86
HYP2:	past witness	chrF2,2 = .62

Limitations of chrF and other overlap metrics

- Local
  - If you move a big phrase it might not change the chrF score much
    - This medication is good for people who get really horrible migraines.
    - For people who get really horrible migraines, this medication is good.

    - Teachers who give long lectures like students who stay awake.
    - Students who stay awake like teachers who give long lectures.

- Sentence-based
  - Does not measure discourse coherence, which affects translation quality

Recurrent Neural Networks (RNNs): Review

RNN for Machine Translation

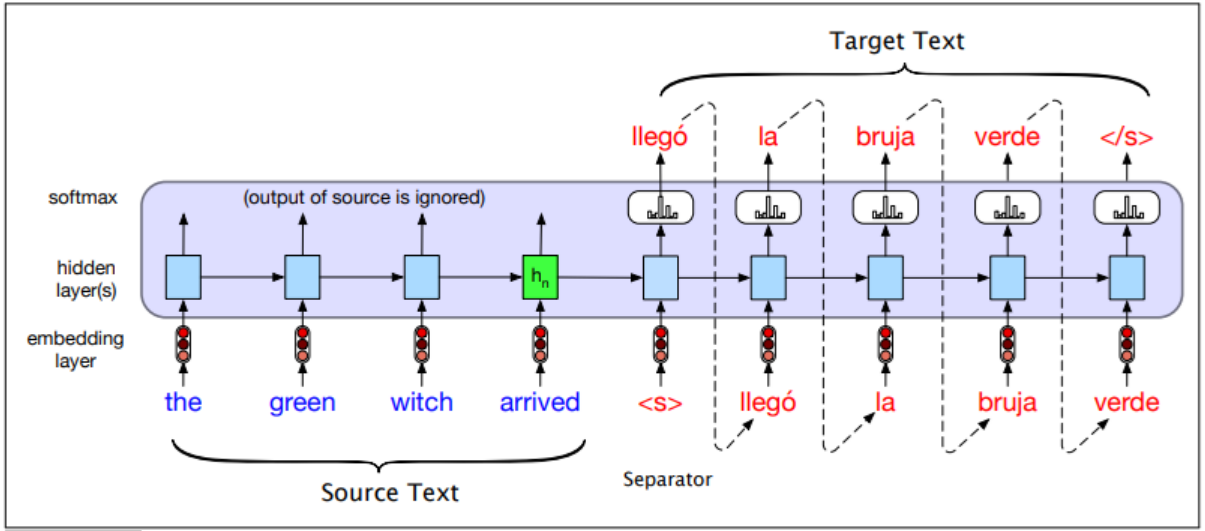
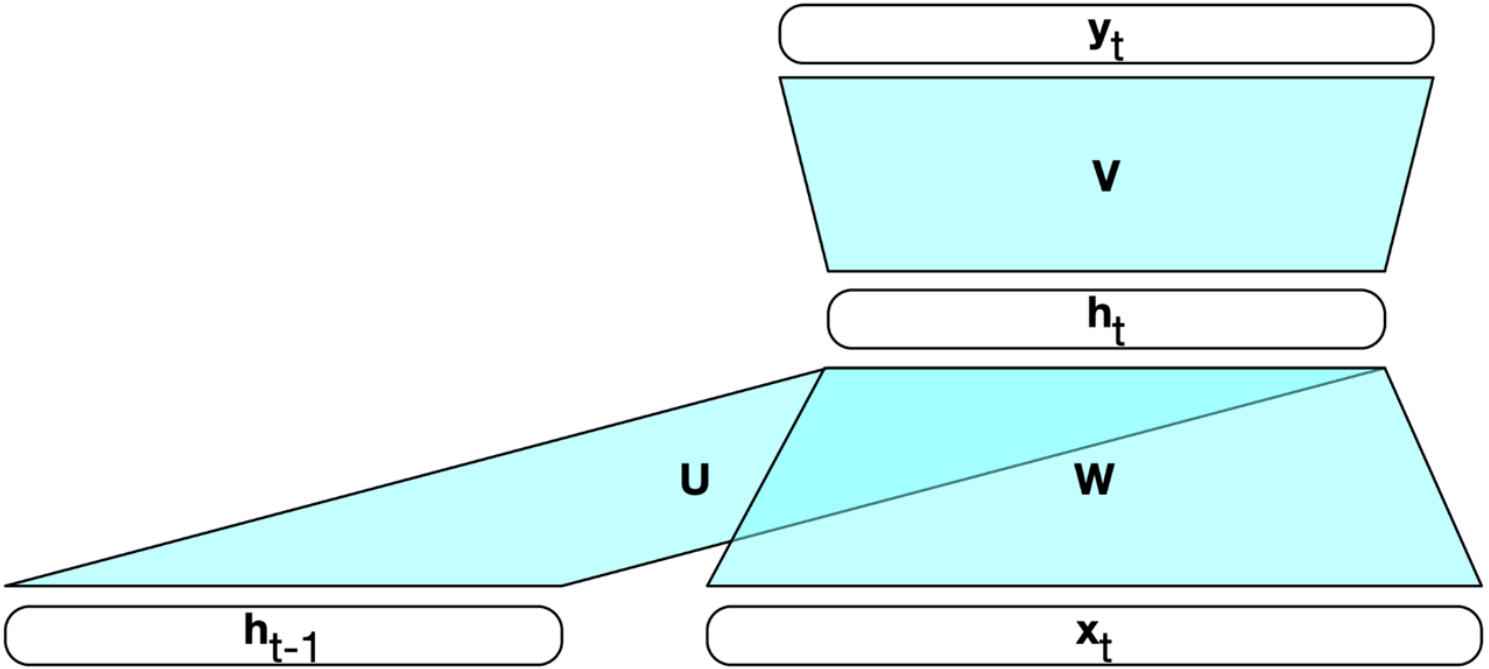


Figure 10.4 Translating a single sentence (inference time) in the basic RNN version of encoder-decoder approach to machine translation. Source and target sentences are concatenated with a separator token in between, and the decoder uses context information from the encoder's last hidden state.

Lexicon:

llegó arrived
la the
bruja witch
verde green

A Simple RNN



RNNs “Recur” across Multiple Time Steps

A simple RNN has three facets, just like a simple FFNN:

1. An input layer, ***x***
2. A hidden layer, ***h***
3. An output layer, ***y***

What makes it different is that it works through time, incorporating information from the preceding time step as well as the current input:

$\boldsymbol{h}_t$ is always the sum of $\boldsymbol{U}\boldsymbol{h}_{t-1}$ and $\boldsymbol{W}\boldsymbol{x}_t$

Where **U** and **W** are weight matrices with $d_h \times d_h$ and $d_h \times d_{in}$ dimensions

A More Formal Definition of an RNN

$$\mathbf{h}_t = g(\mathbf{U}\mathbf{h}_{t-1} + \mathbf{W}\mathbf{x}_t)$$

$$\mathbf{y}_t = f(\mathbf{V}\mathbf{h}_t)$$

Where (assuming no embedding layer):

- d_h is the number of dimensions in $\mathbf{h}$
- $\mathbf{U}$ is a $d_h \times d_h$ matrix of real numbers
- $\mathbf{W}$ is a $d_h \times d_{in}$ matrix of real numbers
- $\mathbf{V}$ is a $d_{out} \times d_h$ matrix of real numbers
- g and f are activation functions

These dimensions, matrices, and functions are the same across all time steps.

Recurrent Neural Net

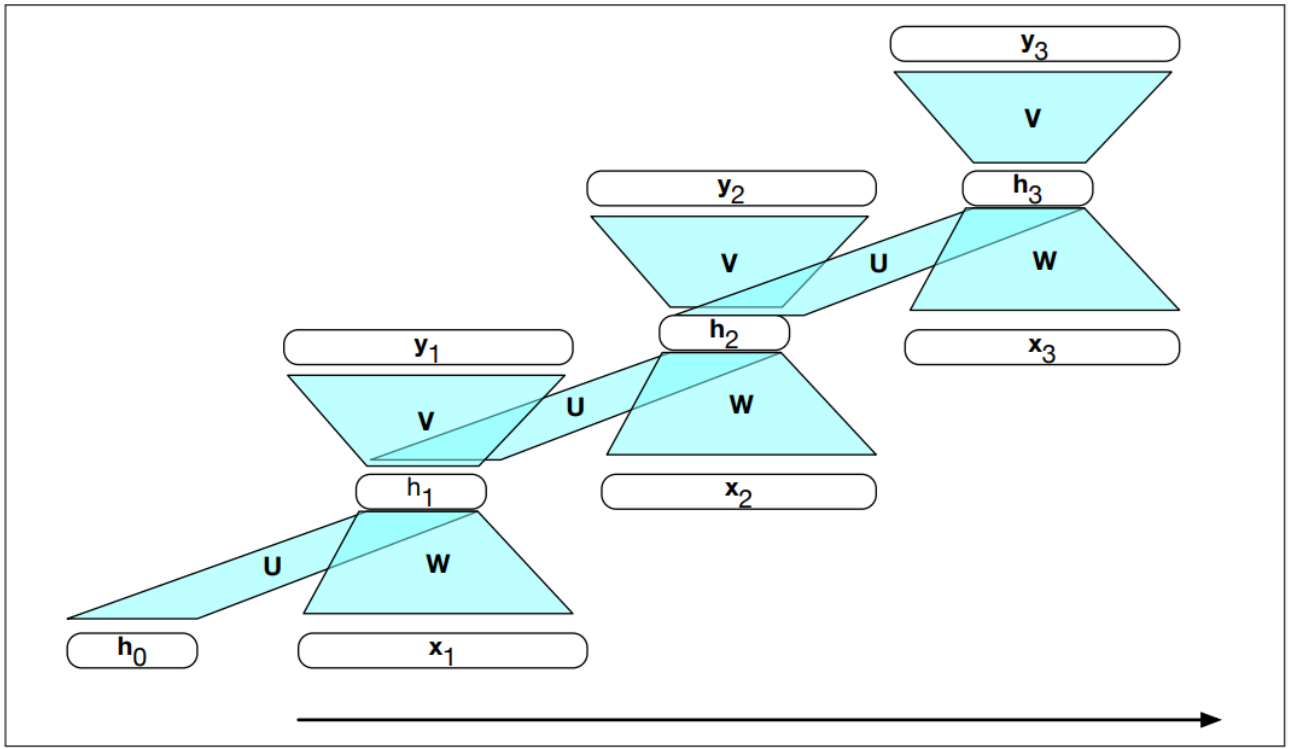


Figure 9.5 A simple recurrent neural network shown unrolled in time. Network layers are recalculated for each time step, while the weights **U**, **V** and **W** are shared in common across all time steps.

Step through a sentence one word at a time, starting with the start symbol.

There are four matrices:

E (not shown): a matrix where each column is a word embedding for a different word. The number of rows is the length of the hidden layer.

U: multiply this weight matrix by the hidden layer of time step t-1.

W: multiply this weight matrix by input t to get h at time t.

V: multiply this weight matrix by the hidden layer at time t.

Recurrent Neural Net

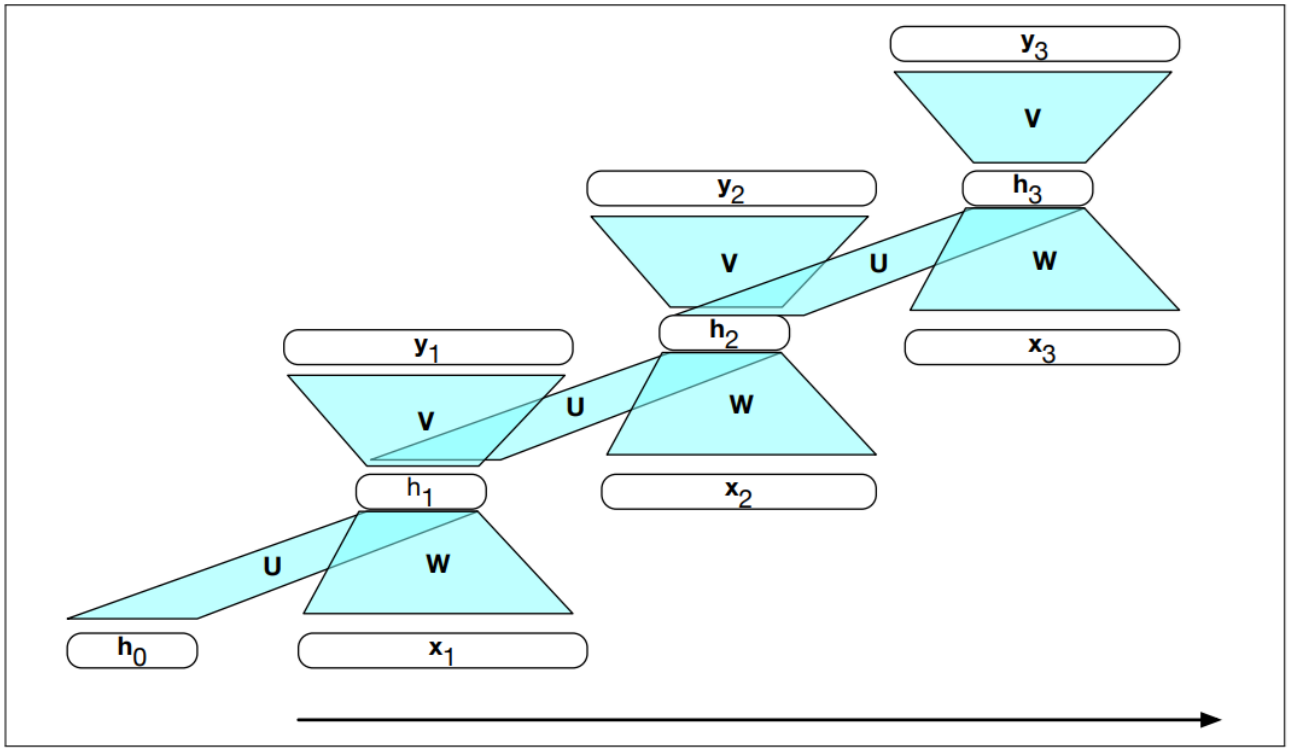


Figure 9.5 A simple recurrent neural network shown unrolled in time. Network layers are recalculated for each time step, while the weights **U**, **V** and **W** are shared in common across all time steps.

$$\mathbf{e}_t = \mathbf{E}\mathbf{x}_t$$
$$\mathbf{h}_t = g(\mathbf{U}\mathbf{h}_{t-1} + \mathbf{W}\mathbf{e}_t)$$
$$\mathbf{y}_t = \text{softmax}(\mathbf{V}\mathbf{h}_t)$$

x_t is a one-hot vector, identifying a word

e_t is the embedding of x_t from the embedding matrix E

Recurrent Neural Net

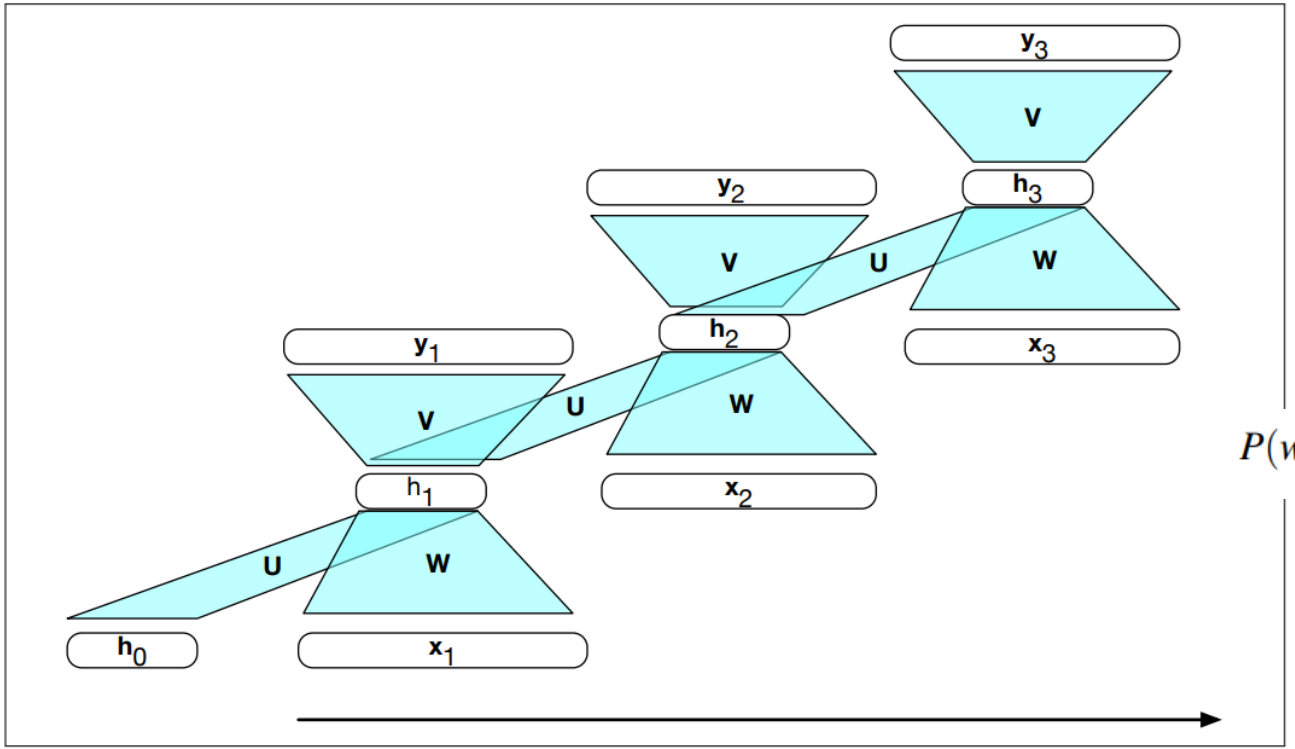


Figure 9.5 A simple recurrent neural network shown unrolled in time. Network layers are recalculated for each time step, while the weights **U**, **V** and **W** are shared in common across all time steps.

If you are using this as a language model, y_t is a probability distribution showing the probabilities of possible next words.

$$P(w_{t+1} = i | w_1, \ldots, w_t) = \mathbf{y}_t[i]$$

The *i*th position of y_t is the probability that the next word is i.

So long and thanks for all the fish

From Douglas Adams, *Hitchhiker's Guide to the Galaxy* sci-fi comedy book (also radio and film) series.

Dolphins said this to humans (who thought they were training them with fish as a reward) as they left Earth.

Generating language with an RNN

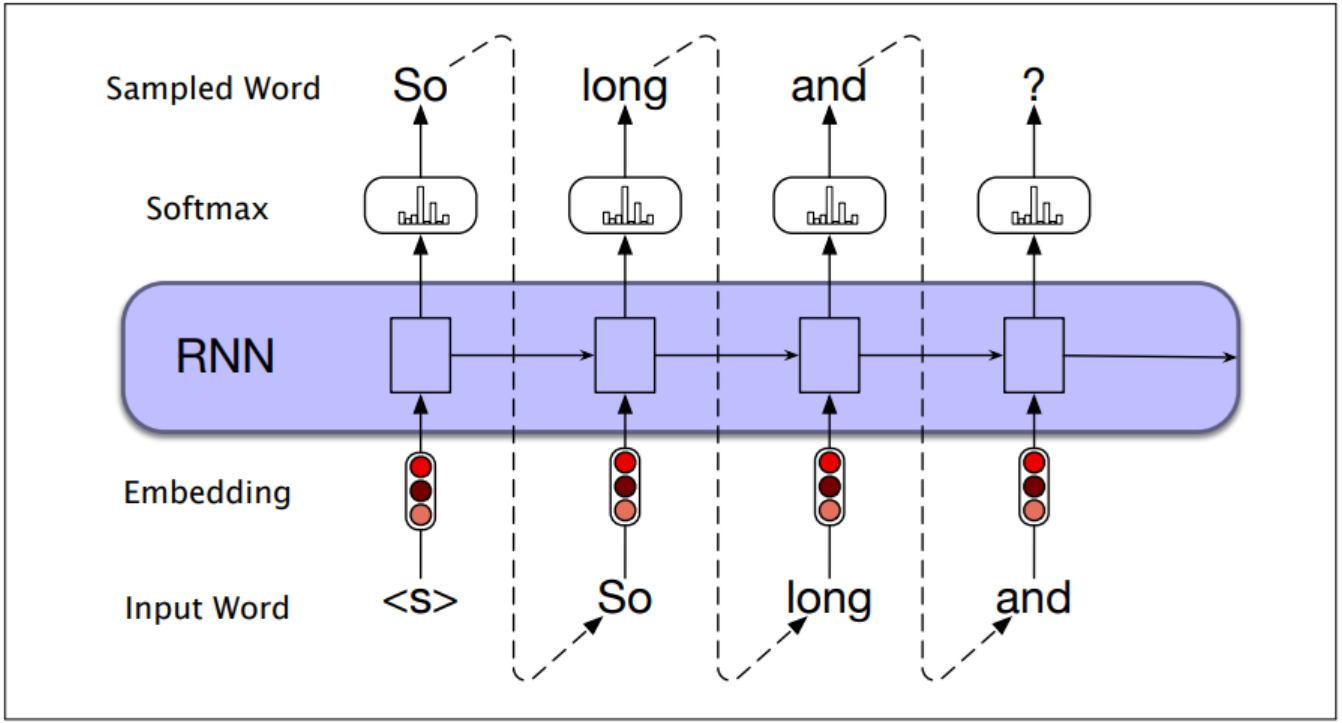


Figure 9.9 Autoregressive generation with an RNN-based neural language model.

Autoregressive generation:

"The word generated at each timestep is conditioned on the word selected by the network from the previous step."

Autoregressive generation is used in machine translation, question answering, and summarization.

Generating language with an RNN

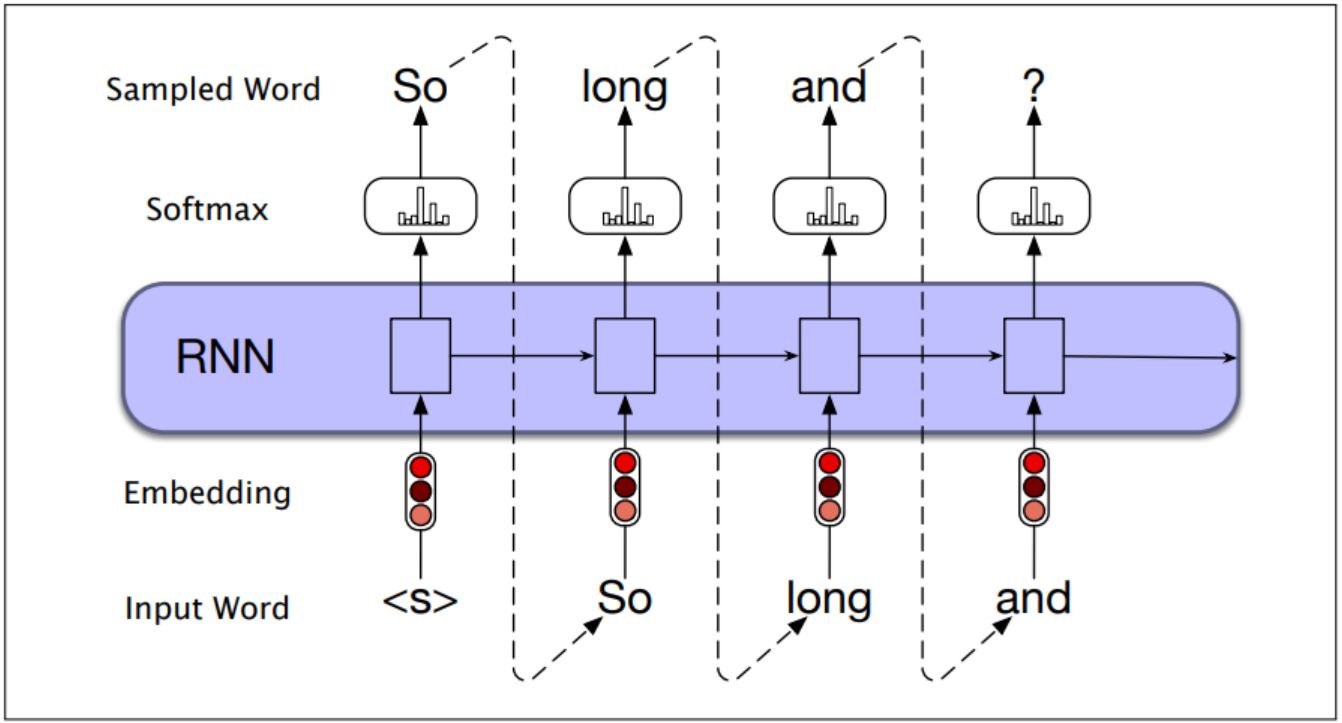


Figure 9.9 Autoregressive generation with an RNN-based neural language model.

Autoregressive generation is used in machine translation, question answering, and summarization.

Instead of starting with a start symbol, start with a richer context.

In Machine Translation, the context is the source language sentence.

RNN for Machine Translation

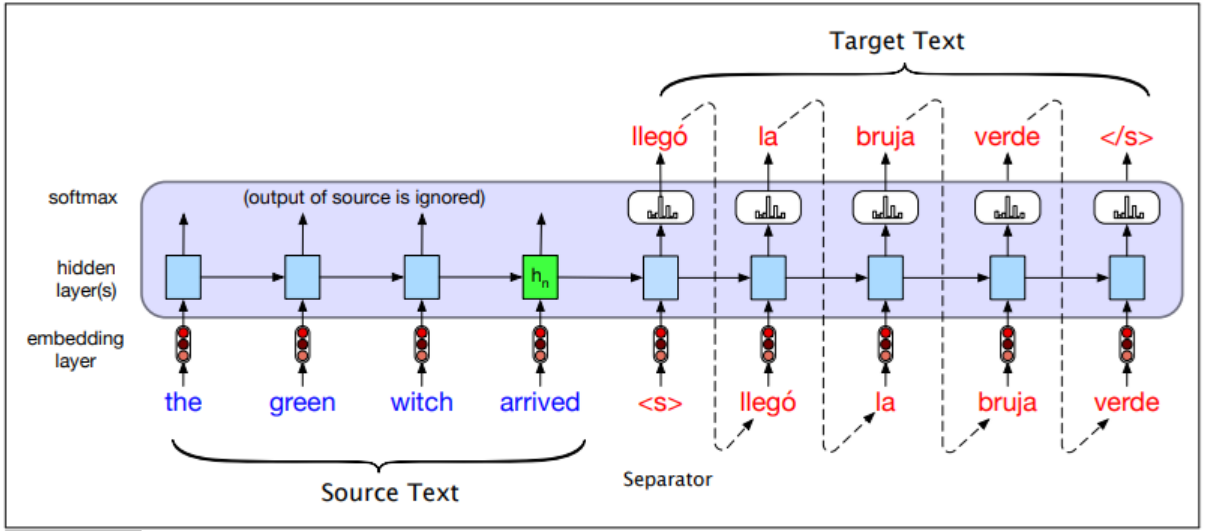


Figure 10.4 Translating a single sentence (inference time) in the basic RNN version of encoder-decoder approach to machine translation. Source and target sentences are concatenated with a separator token in between, and the decoder uses context information from the encoder's last hidden state.

Lexicon:

llegó arrived
la the
bruja witch
verde green

Stacking RNNs

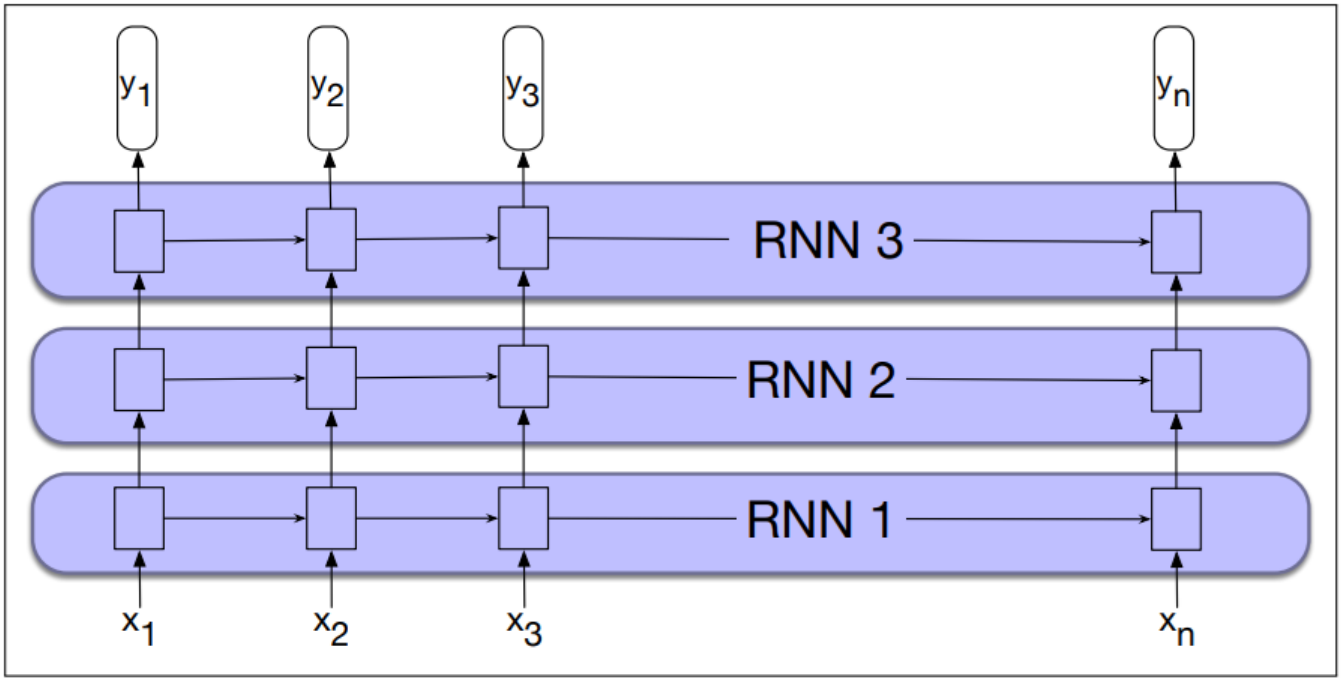
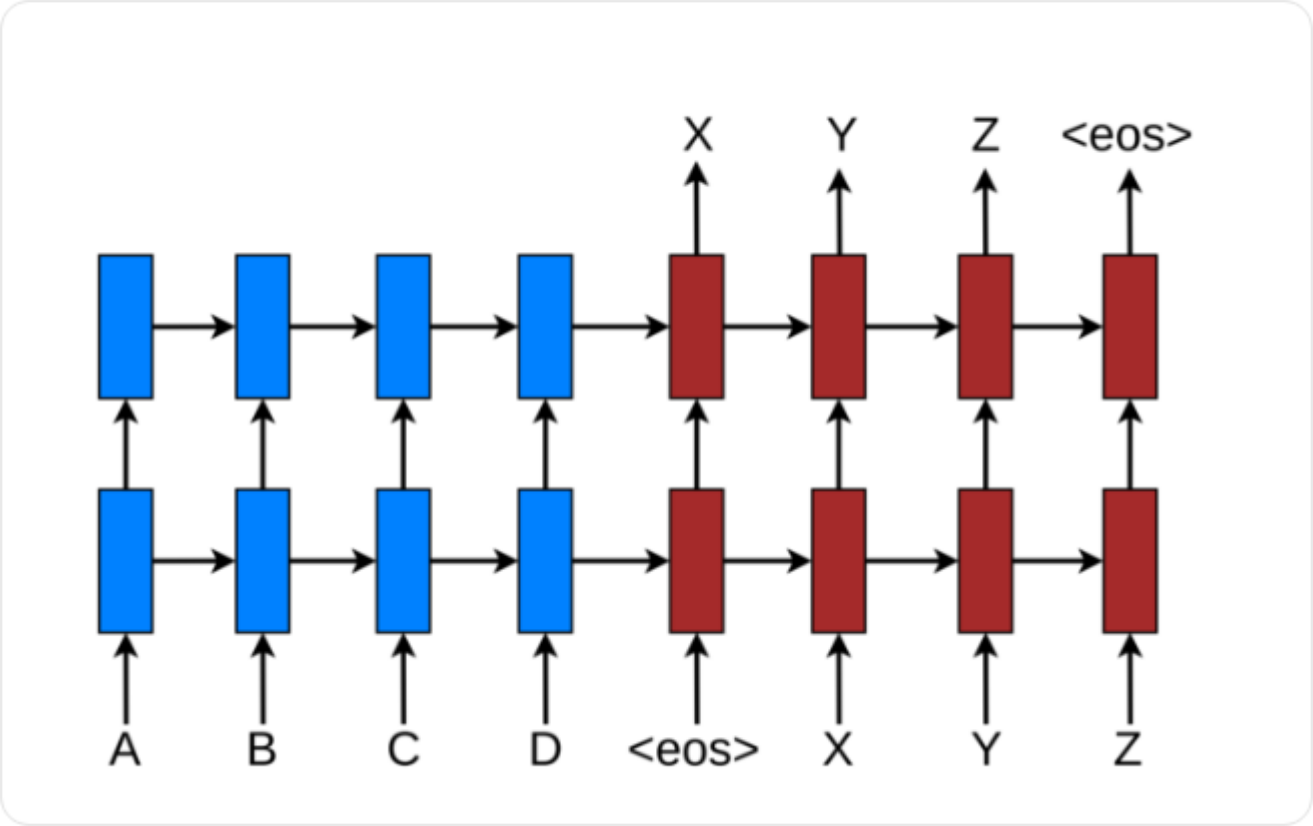


Figure 9.10 Stacked recurrent networks. The output of a lower level serves as the input to higher levels with the output of the last network serving as the final output.

The output of each time step is used by the next time step and by the same time step in the next layer.

When you probe inside the layers, you find that each layer learns a different level of abstraction about syntax, morphology, meaning, dependencies between words, etc.

What will your homework look like?



The problem of only moving left-to-right

- When you see “I know that” you don’t have enough information to choose a part of speech or translation for “that”
  - I know that.
    - “That” is a pronoun.
  - I know [that student].
    - “That” is a determiner.
  - I know [that you yawned].
    - “That” is a *complementizer*, a word that marks the beginning of an embedded clause (which linguists call a complement clause).

The problem of only moving left-to-right

- When you get to the word “man”, you don’t have enough information to know if it is a noun or a verb.
  - [The old man] is rowing.
  - The old [man the boats].

Bidirectional RNNs

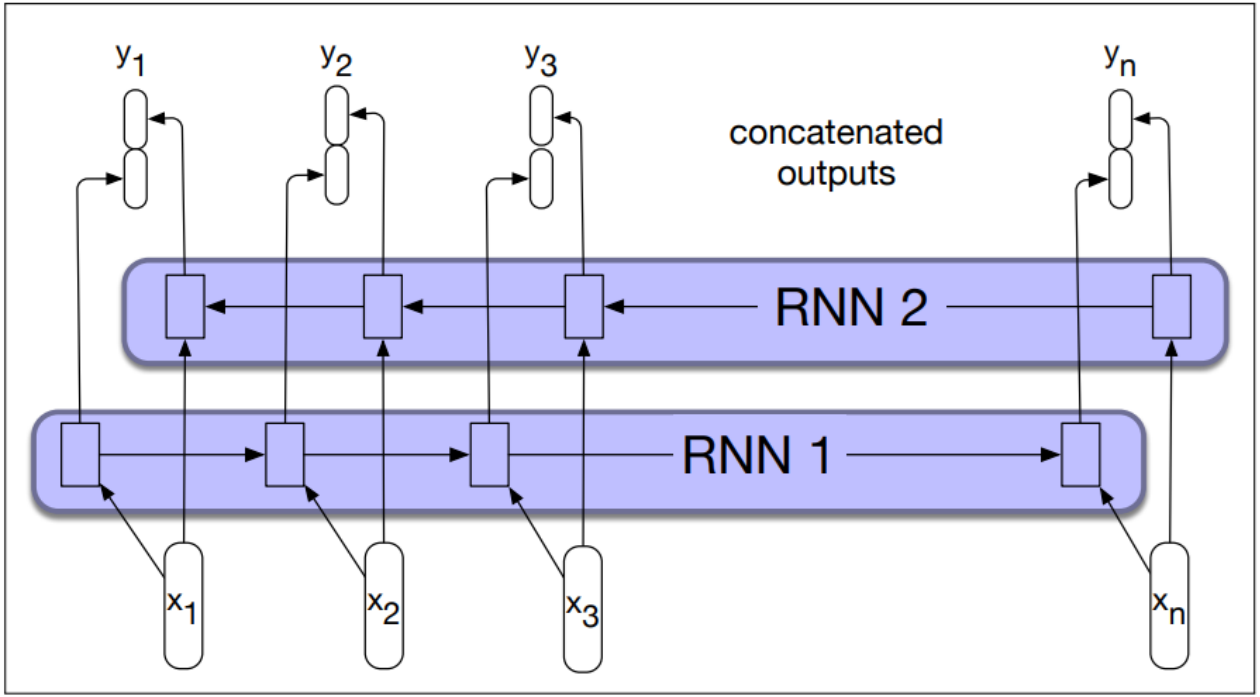


Figure 9.11 A bidirectional RNN. Separate models are trained in the forward and backward directions, with the output of each model at each time point concatenated to represent the bidirectional state at that time point.

If you have access to the entire input string all at once, you can run the RNN in both directions and concatenate the outputs.

$$\mathbf{h}_t^f = RNN_{forward}(\mathbf{x}_1, \ldots, \mathbf{x}_t)$$

$$\mathbf{h}_t^b = RNN_{backward}(\mathbf{x}_t, \ldots \mathbf{x}_n)$$

$$\begin{aligned}\mathbf{h}_t &= [\mathbf{h}_t^f ; \mathbf{h}_t^b] \\ &= \mathbf{h}_t^f \oplus \mathbf{h}_t^b\end{aligned}$$

Weaknesses of RNNs

- Vanishing Gradients:
  - During training, repeated multiplication drives gradients to zero
  - Result: information from many timesteps away is lost
  - The **flights** the airline cancelled **was**/**were** full.