

ANTONIS ANASTASOPOULOS
CS499 INTRODUCTION TO NLP

NEURAL LANGUAGE MODELS



<https://cs.gmu.edu/~antonis/course/cs499-spring21/>

STRUCTURE OF THIS LECTURE

- 1** Quick LM Recap
- 2** Recurrent Neural Nets
- 3** Evaluation

LANGUAGE MODELING

LANGUAGE MODELING

The goal is to obtain a model to compute:

$$P(X) = \prod_{i=1}^I P(x_i | x_1, \dots, x_{i-1})$$

What if we could have **infinite history**, instead of relying on finite n-grams?

**AN ALTERNATIVE:
FEATURIZED LOG-LINEAR MODELS**

AN ALTERNATIVE: FEATURIZED MODELS

Calculate features of the context

Based on the features, calculate probabilities

Optimize feature weights using gradient descent, etc.

EXAMPLE:

Previous words: "giving a"

a
the
talk
gift
hat
...

Words we're
predicting

$$b = \begin{pmatrix} 3.0 \\ 2.5 \\ -0.2 \\ 0.1 \\ 1.2 \\ \dots \end{pmatrix}$$

How likely
are they?

$$w_{1,a} = \begin{pmatrix} -6.0 \\ -5.1 \\ 0.2 \\ 0.1 \\ 0.5 \\ \dots \end{pmatrix}$$

How likely
are they
given prev.
word is "a"?

$$w_{2,giving} = \begin{pmatrix} -0.2 \\ -0.3 \\ 1.0 \\ 2.0 \\ -1.2 \\ \dots \end{pmatrix}$$

How likely
are they
given 2nd prev.
word is "giving"?

$$s = \begin{pmatrix} -3.2 \\ -2.9 \\ 1.0 \\ 2.2 \\ 0.6 \\ \dots \end{pmatrix}$$

Total
score

SOFTMAX

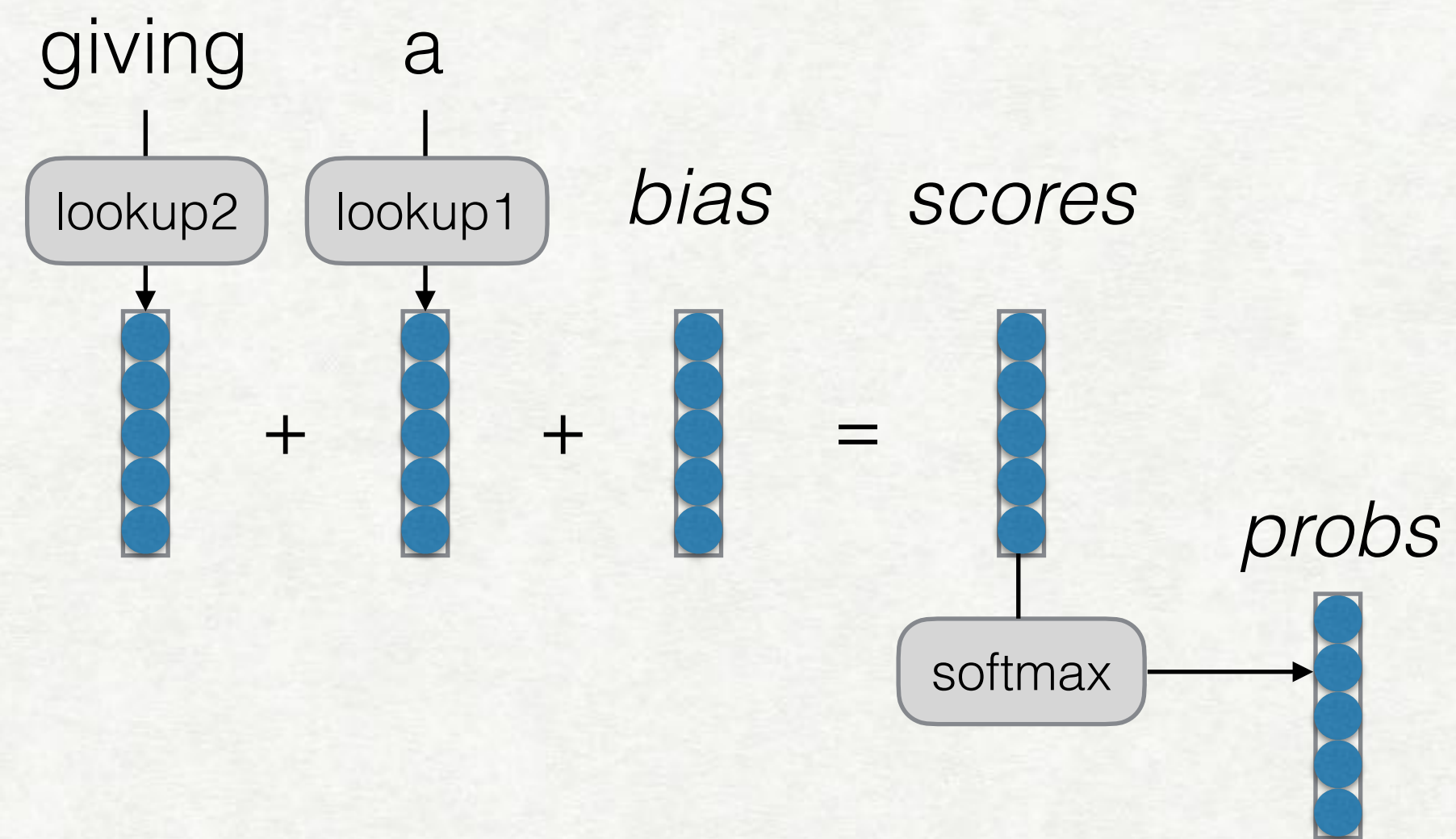
Convert scores into probabilities by taking the exponent and normalizing (softmax)

$$P(x_i | x_{i-n+1}^{i-1}) = \frac{e^{s(x_i | x_{i-n+1}^{i-1})}}{\sum_{\tilde{x}_i} e^{s(\tilde{x}_i | x_{i-n+1}^{i-1})}}$$

$$s = \begin{pmatrix} -3.2 \\ -2.9 \\ 1.0 \\ 2.2 \\ 0.6 \\ \dots \end{pmatrix} \longrightarrow p = \begin{pmatrix} 0.002 \\ 0.003 \\ 0.329 \\ 0.444 \\ 0.090 \\ \dots \end{pmatrix}$$

A COMPUTATION GRAPH VIEW

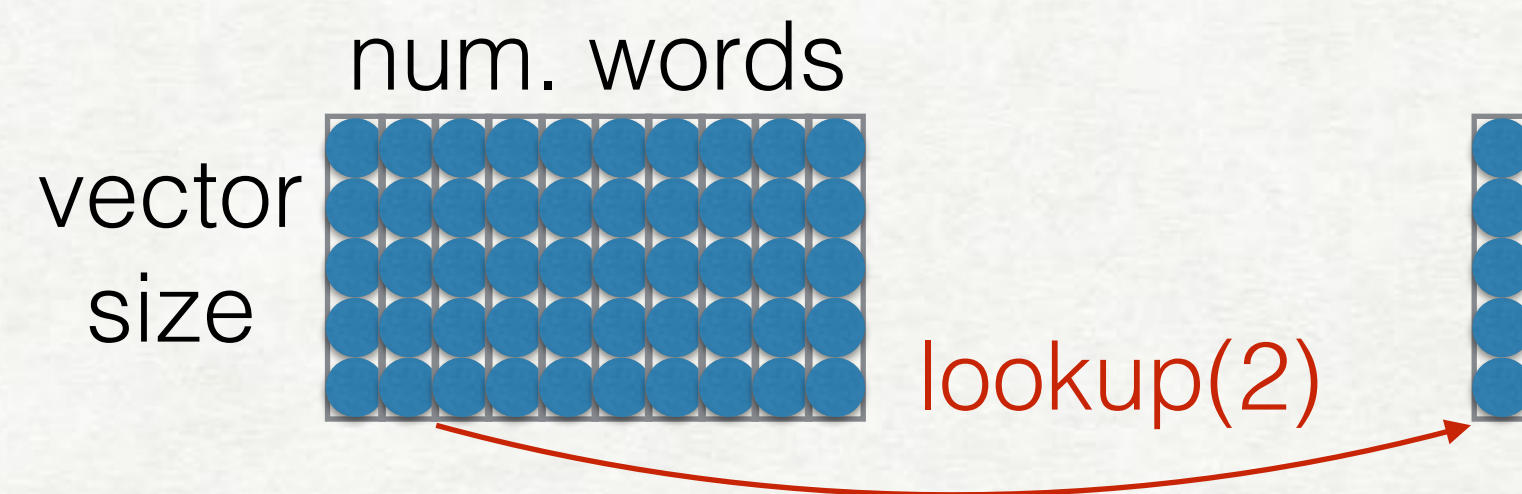
Each word has a vector of weights for each tag



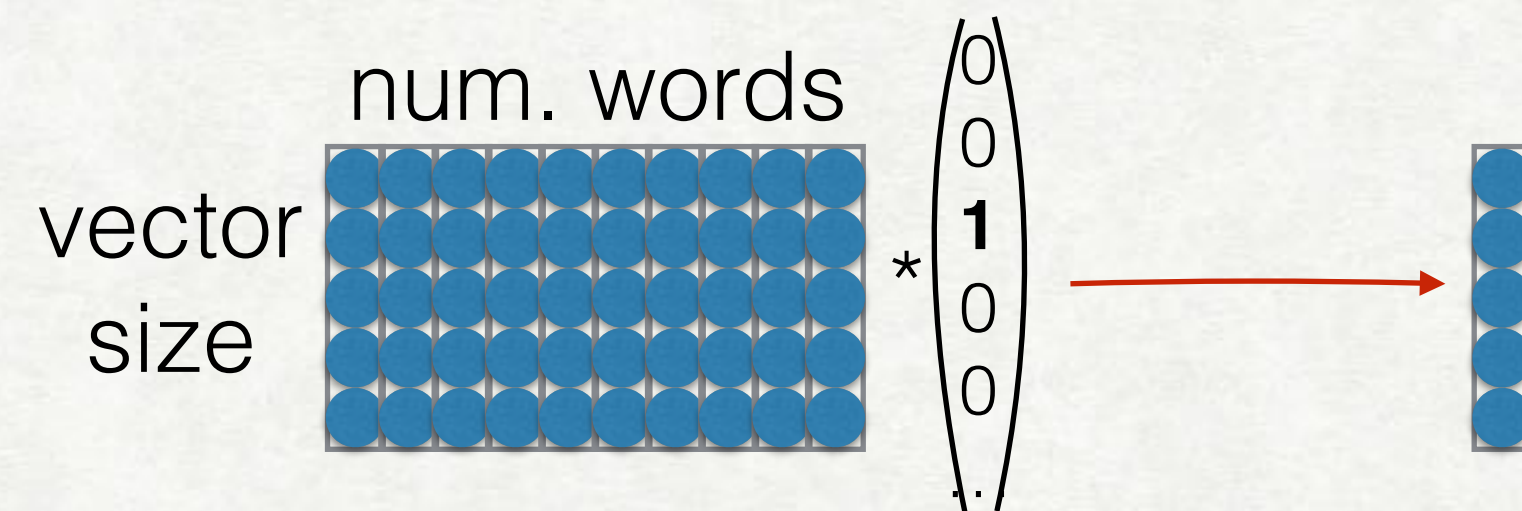
Each vector is size of output vocabulary

A NOTE: "LOOKUP"

Lookup can be viewed as "grabbing" a single vector from a big matrix of word embeddings



Similarly, can be viewed as multiplying with an "one-hot" vector



The former tends to be faster

TRAINING A NEURAL MODEL

To train, we calculate a “loss function” (a measure of how bad our predictions are), and move the parameters to reduce the loss

The most common loss function for probabilistic models is “negative log likelihood”

TRAINING A MODEL

Reminder: to train, we calculate a “loss function” (a measure of how bad our predictions are), and move the parameters to reduce the loss

The most common loss function for probabilistic models is “negative log likelihood”

If element 3
(or zero-indexed, 2)
is the correct answer:

$$p = \begin{pmatrix} 0.002 \\ 0.003 \\ \boxed{0.329} \\ 0.444 \\ 0.090 \\ \dots \end{pmatrix} \xrightarrow{-\log} 1.112$$

CHOOSING A VOCABULARY

UNKNOWN WORDS

Necessity for UNK words

We won't have all the words in the world in training data

Larger vocabularies require more memory and computation time

Common ways:

Frequency threshold (usually $\text{UNK} \leq 1$)

Rank threshold

UNKNOWN WORDS

A very large number of published documents contain text only. They often look boring, and they are often written in obscure language, using mile-long sentences and cryptic technical terms, using one font only, perhaps even without headings. Such style, or lack of style, might be the one you are strongly expected to follow when writing eg scientific or technical reports, legal documents, or administrative papers. It is natural to think that such documents would benefit from a few illustrative images. (However, just adding illustration might be rather useless, if the text remains obscure and unstructured.)

UNKNOWN WORDS

a very large number of published documents contain text only . they often look boring , and they are often written in obscure language , using mile-long sentences and cryptic technical terms , using one font only , perhaps even without headings . such style, or lack of style, might be the one you are strongly expected to follow when writing eg scientific or technical reports , legal documents , or administrative papers . it is natural to think that such documents would benefit from a few illustrative images . (however , just adding illustration might be rather useless , if the text remains obscure and unstructured .)

lowercase + tokenize

UNKNOWN WORDS

a very large number of published documents contain text only . they often look boring , and they are often written in obscure language , using **mile-long** sentences and cryptic technical terms , using one font only , perhaps even without headings . such style, or lack of style, might be the one you are strongly expected to follow when writing eg scientific or technical reports , legal documents , or **administrative** papers . it is natural to think that such documents would benefit from a few illustrative images . (however , just adding **illustration** might be rather useless , if the text remains obscure and **unstructured** .)

Find rare words (e.g. with $\text{freq} < 2$)

UNKNOWN WORDS

a very large number of published documents contain text only . they often look boring , and they are often written in obscure language , using **UNK** sentences and cryptic technical terms , using one font only , perhaps even without headings . such style, or lack of style, might be the one you are strongly expected to follow when writing eg scientific or technical reports , legal documents , or **UNK** papers . it is natural to think that such documents would benefit from a few illustrative images . (however , just adding **UNK** might be rather useless , if the text remains obscure and **UNK** .)

Substitute with UNK

EVALUATION AND VOCABULARY

Important: the vocabulary must be the same over models you compare

EVALUATION AND VOCABULARY

Important: the vocabulary must be the same over models you compare

Or more accurately, all models must be able to generate the test set (it's OK if they can generate *more* than the test set, but not less)

e.g. Comparing a character-based model to a word-based model is fair, but not vice-versa

RECURRENT NEURAL NETWORKS

LINEAR MODELS CAN'T LEARN FEATURE COMBINATIONS

farmers eat steak → **high**

cows eat steak → **low**

farmers eat hay → **low**

cows eat hay → **high**

These can't be expressed by linear features

What can we do?

Remember combinations as features (individual scores for "farmers eat", "cows eat")

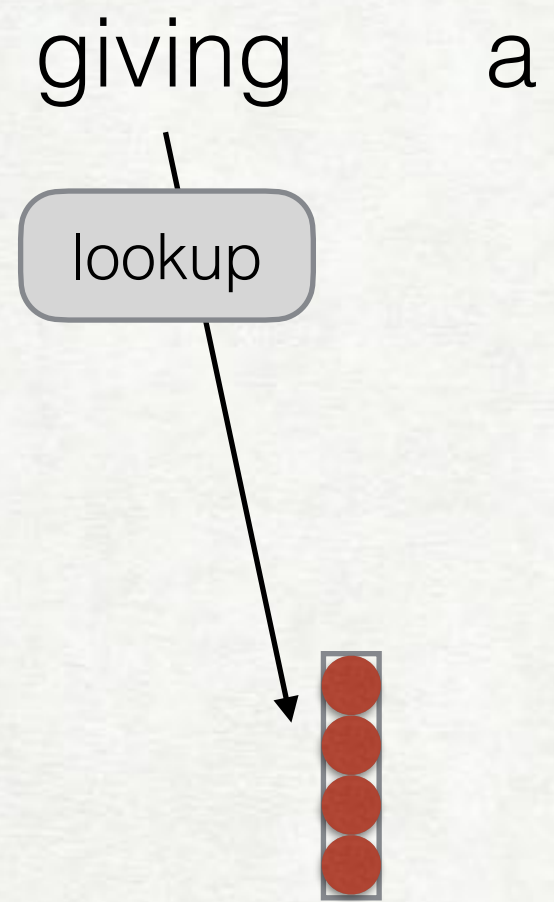
→ Feature space explosion!

Neural nets

NEURAL LANGUAGE MODELS

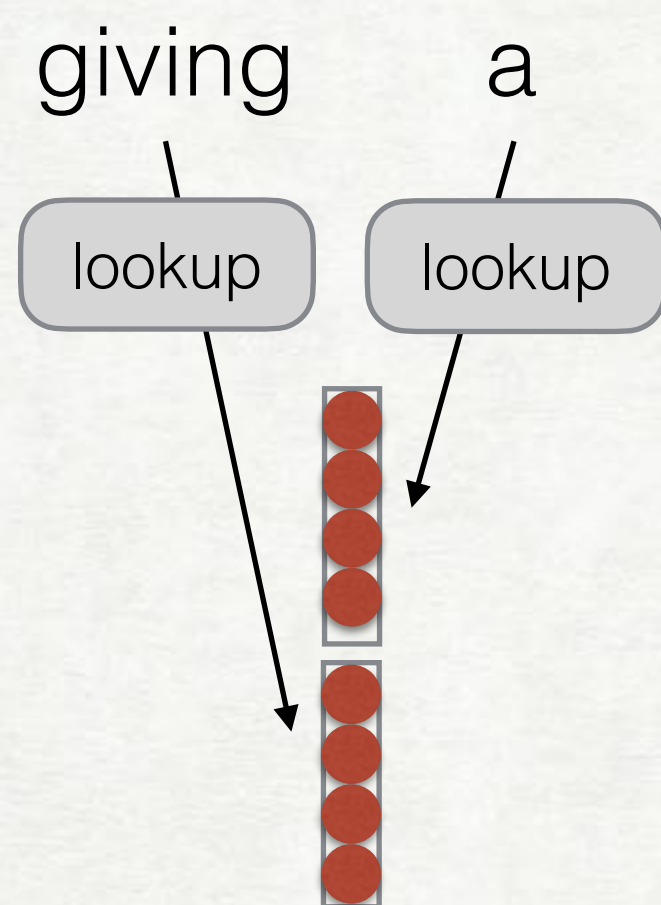
(See Bengio et al. 2004)

NEURAL LANGUAGE MODELS



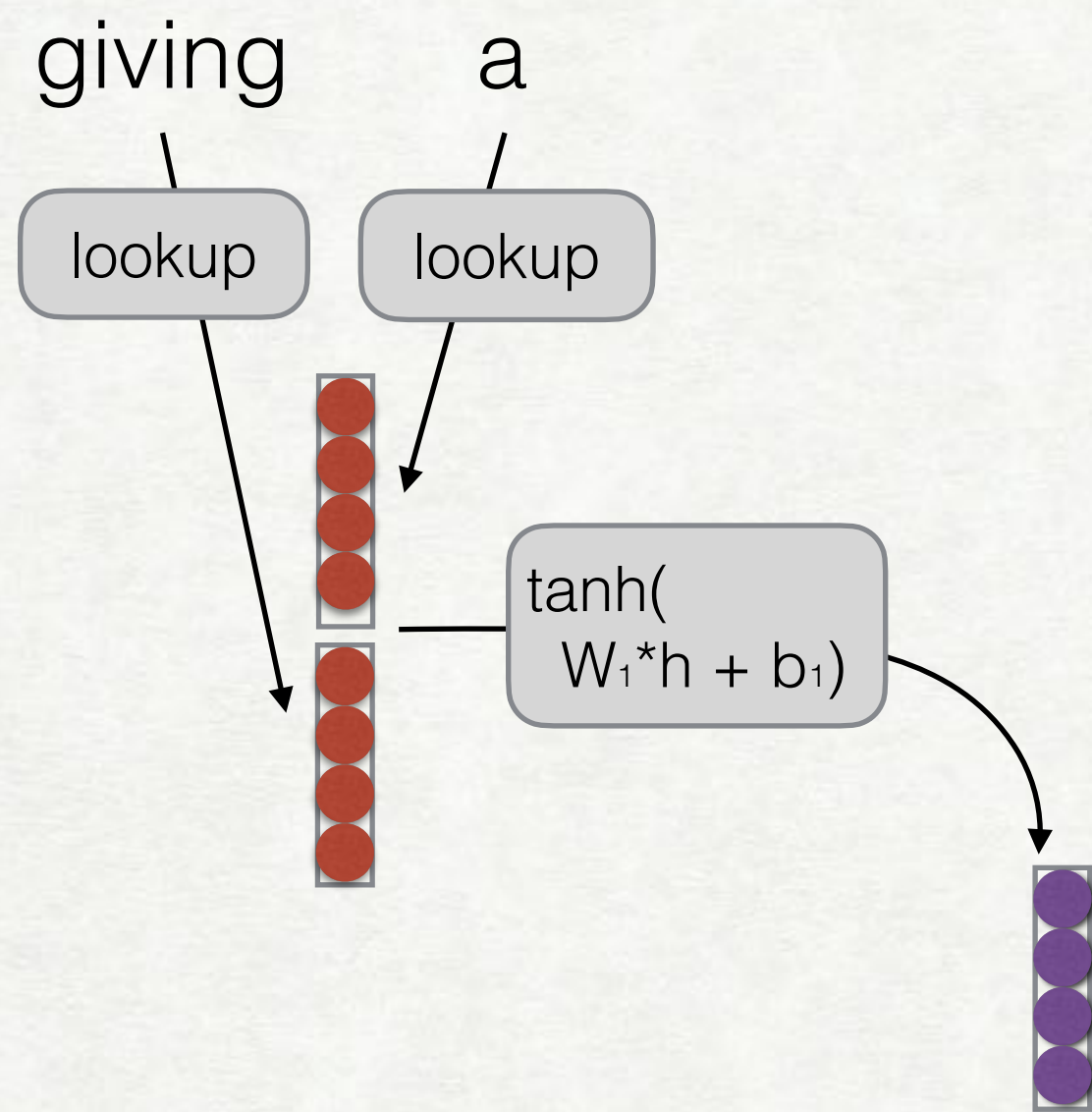
(See Bengio et al. 2004)

NEURAL LANGUAGE MODELS



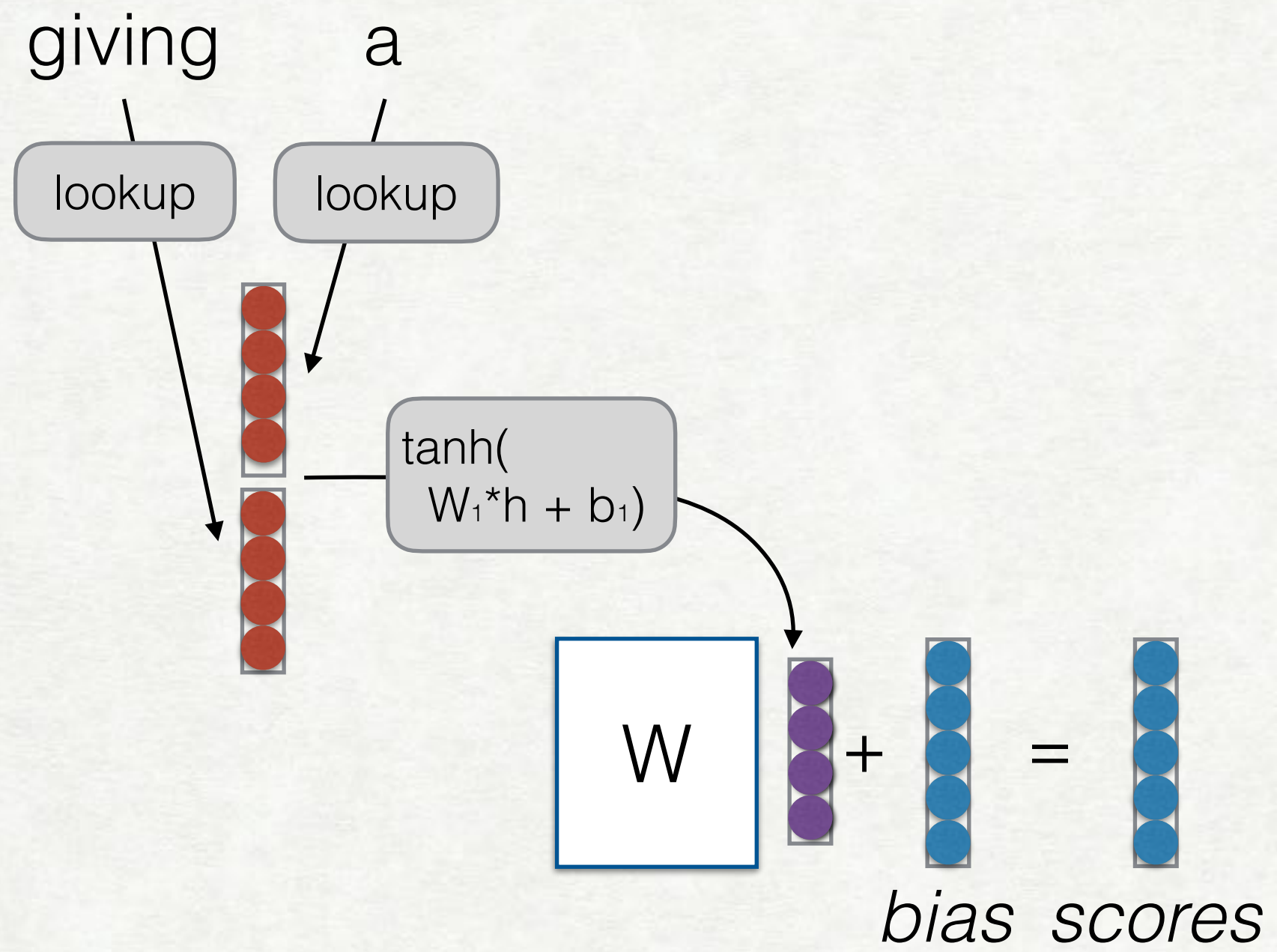
(See Bengio et al. 2004)

NEURAL LANGUAGE MODELS



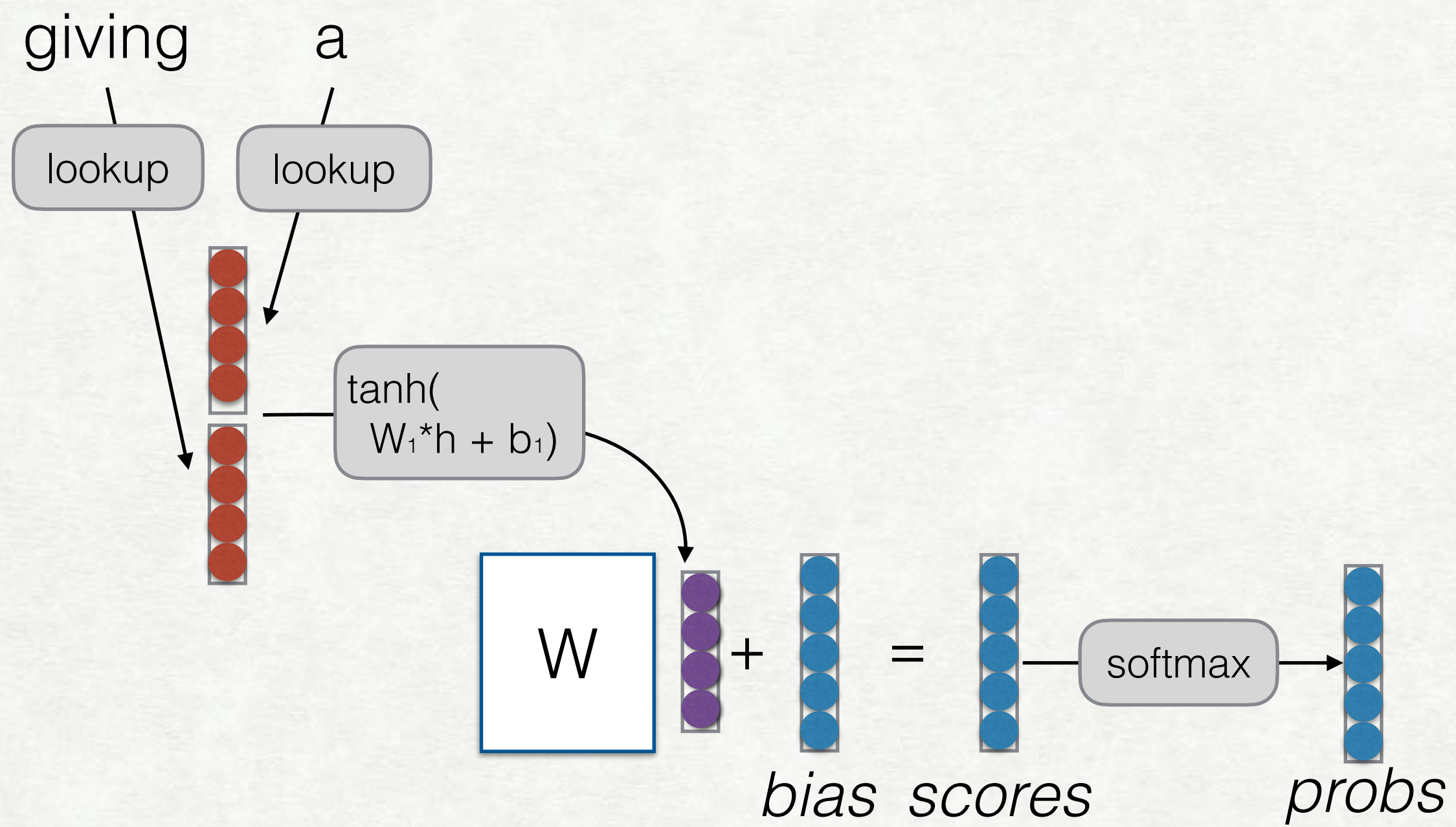
(See Bengio et al. 2004)

NEURAL LANGUAGE MODELS



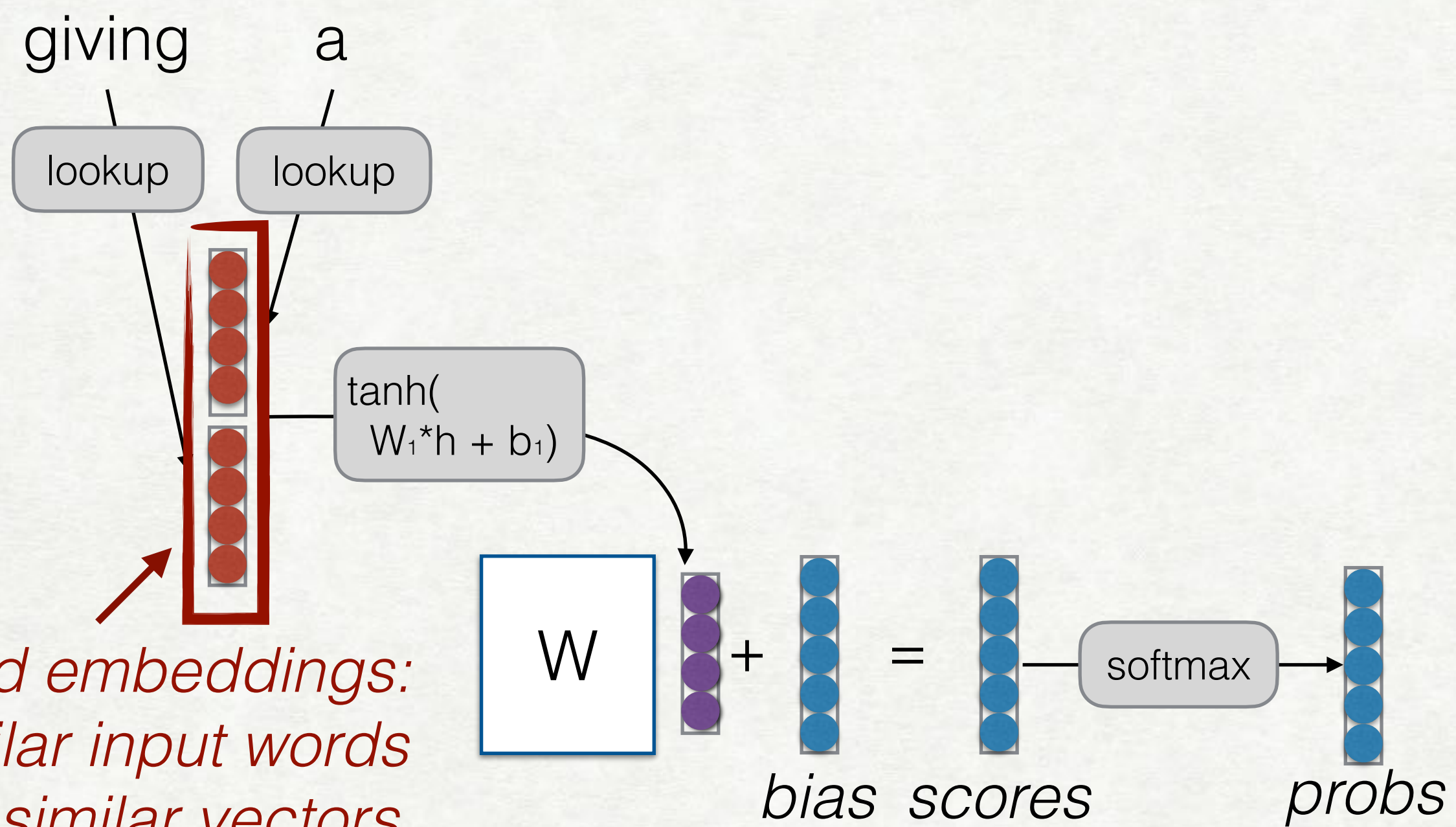
(See Bengio et al. 2004)

NEURAL LANGUAGE MODELS



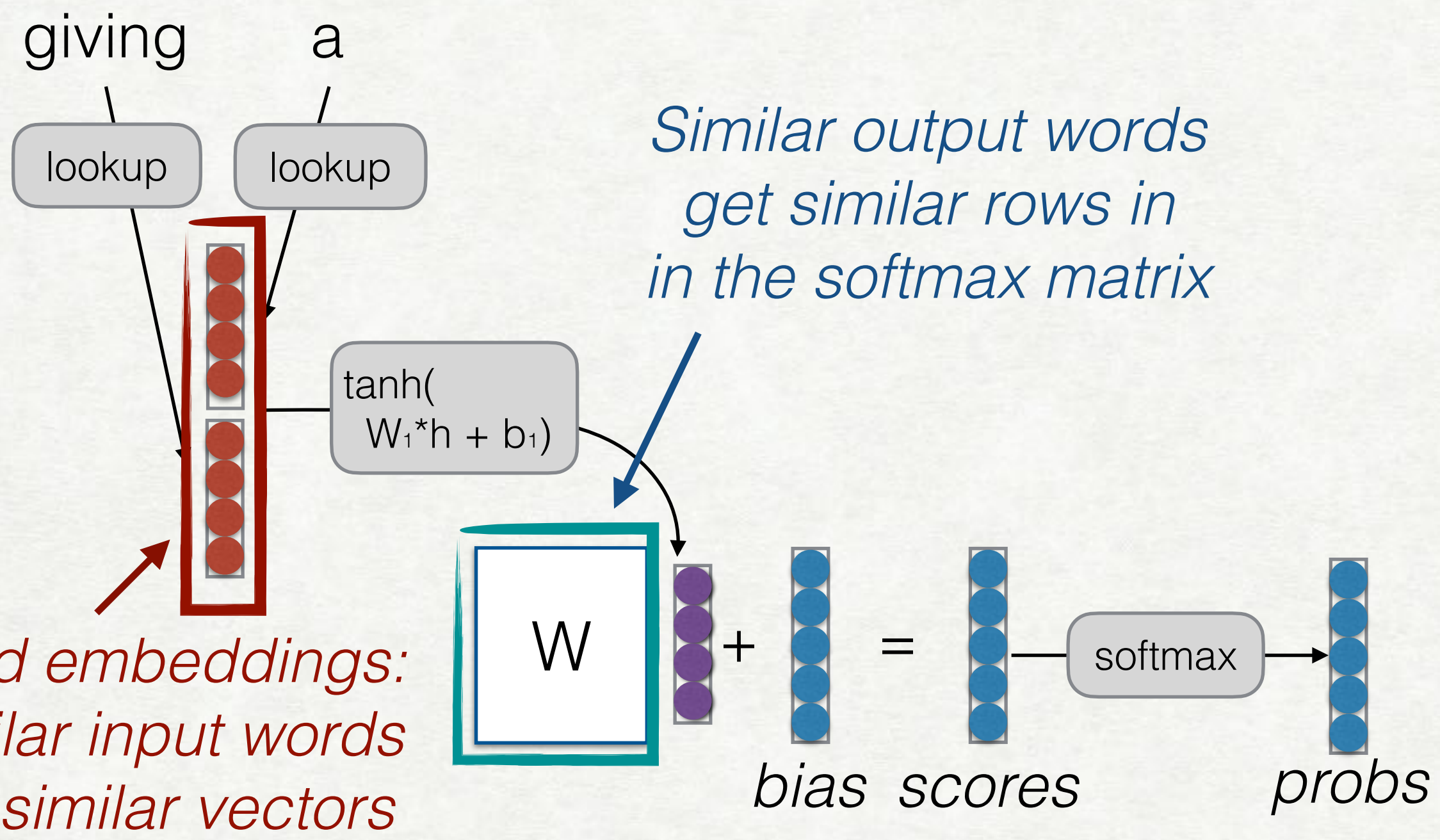
(See Bengio et al. 2004)

WHERE IS STRENGTH SHARED?



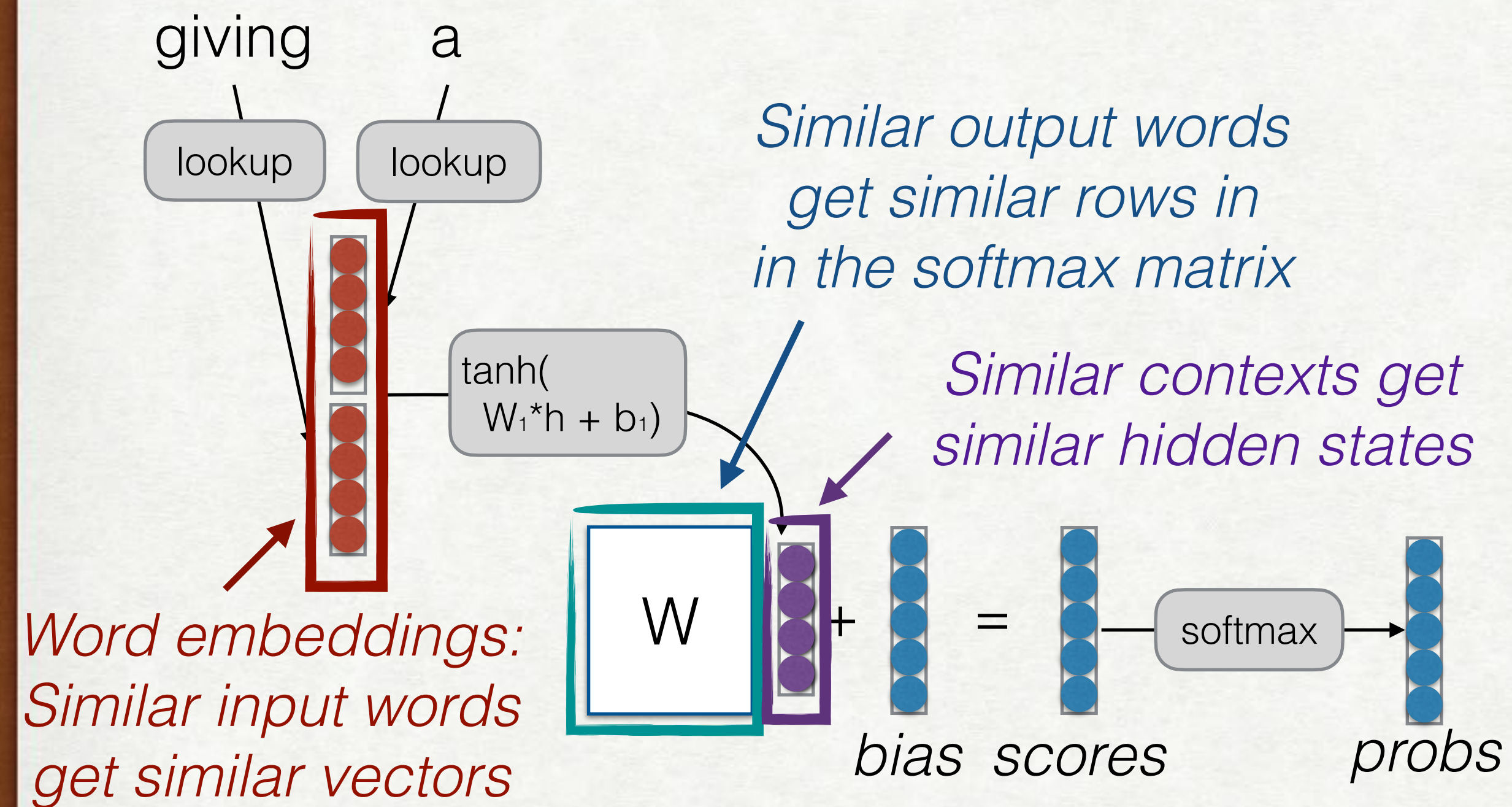
(See Bengio et al. 2004)

WHERE IS STRENGTH SHARED?



(See Bengio et al. 2004)

WHERE IS STRENGTH SHARED?



(See Bengio et al. 2004)

WHAT PROBLEMS ARE HANDLED?

Cannot share strength among **similar words**

she bought a car she bought a bicycle
she purchased a car she purchased a bicycle

→ solved, and similar contexts as well! 😊

Cannot condition on context with intervening words

Dr. Jane Smith Dr. Gertrude Smith

→ solved! 😊

Cannot handle **long-distance dependencies**

for tennis class he wanted to buy his own racquet
for programming class he wanted to buy his own computer

→ not solved yet 😞

LONG-DISTANCE DEPENDENCIES IN LANGUAGE

Agreement in number, gender, etc.

He does not have very much confidence in **himself**.

She does not have very much confidence in **herself**.

LONG-DISTANCE DEPENDENCIES IN LANGUAGE

Agreement in number, gender, etc.

He does not have very much confidence in **himself**.

She does not have very much confidence in **herself**.

Selectional preference

The **reign** has lasted as long as the life of the **queen**.

The **rain** has lasted as long as the life of the **clouds**.

CAN BE COMPLICATED!

What is the referent of "it"?

The trophy would not fit in the brown suitcase because it was too **big**.

Trophy

The trophy would not fit in the brown suitcase because it was too **small**.

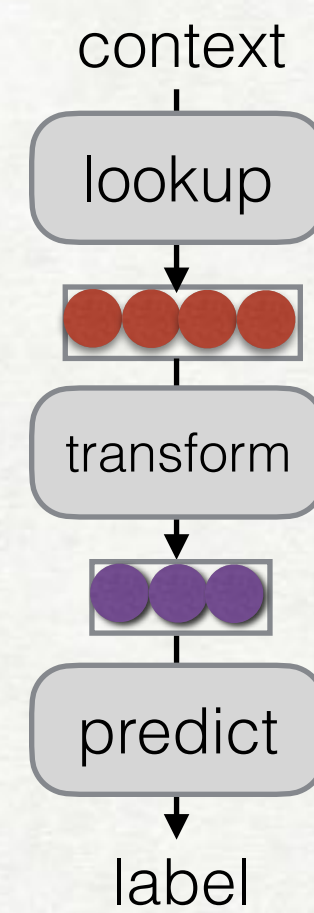
Suitcase

(from Winograd Schema Challenge:
<http://commonsensereasoning.org/winograd.html>)

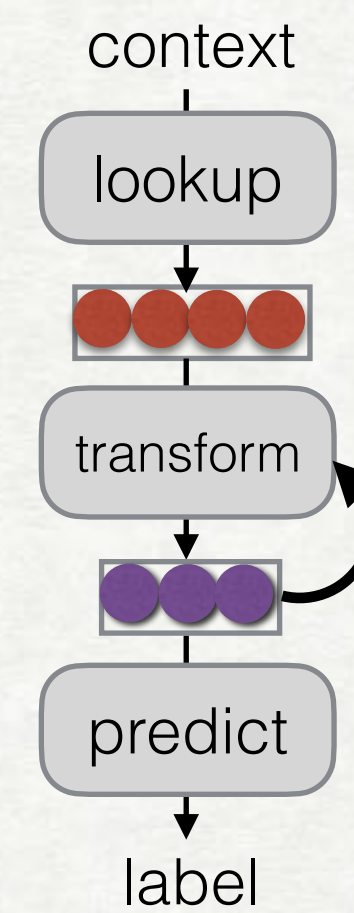
RECURRENT NEURAL NETWORKS (ELMAN 1990)

Tools to “remember” information

Feed-forward NN

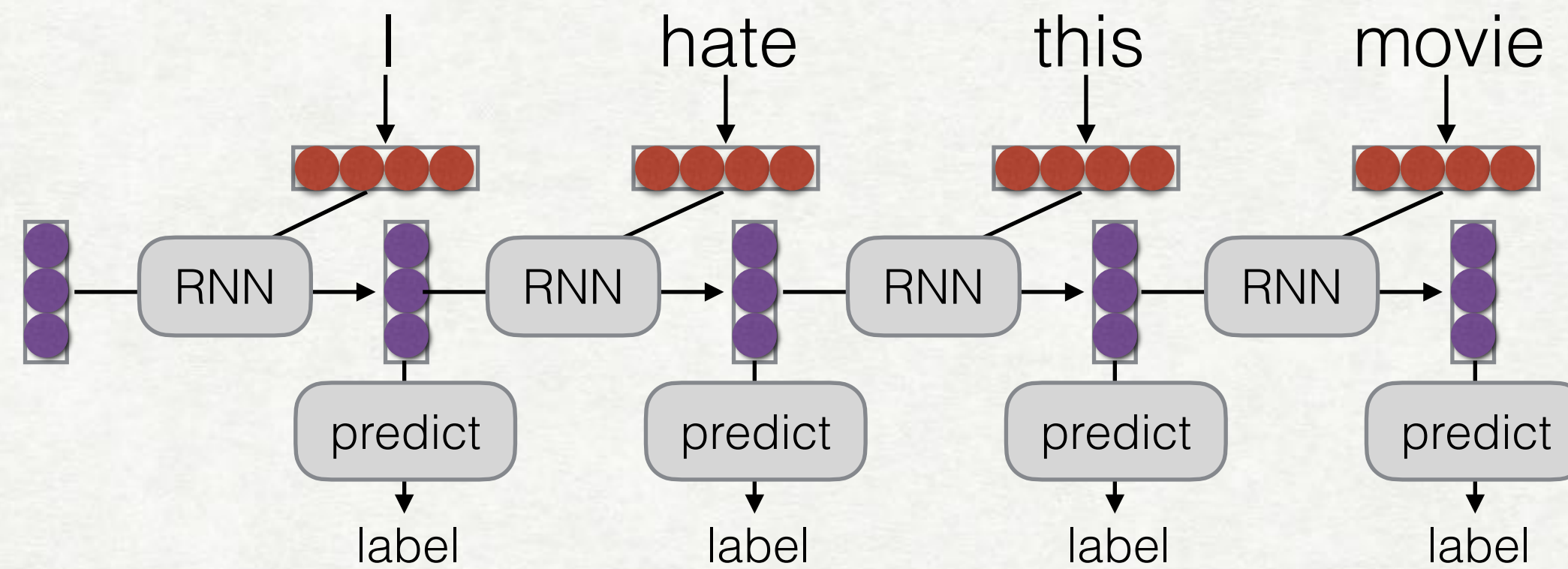


Recurrent NN



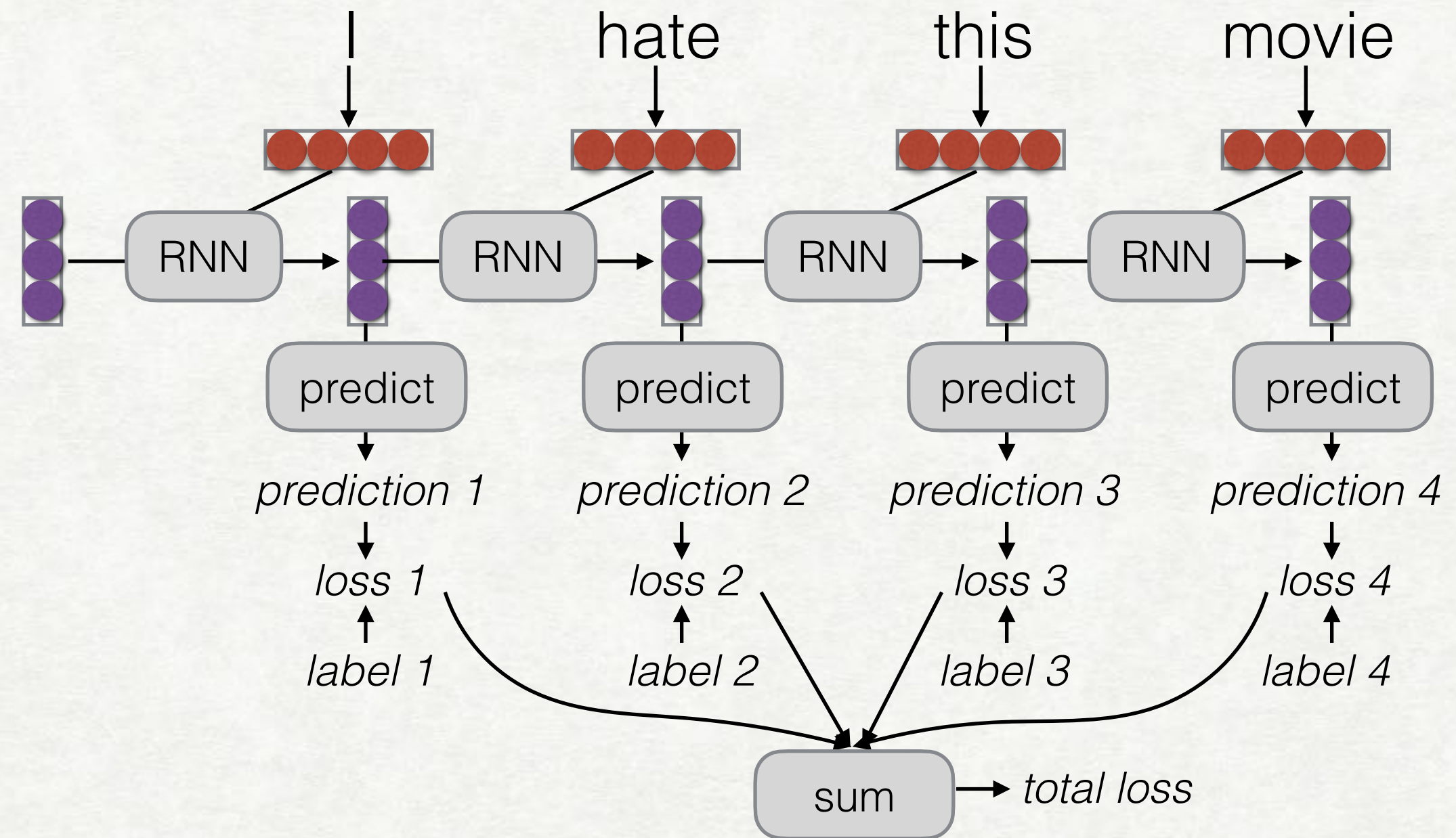
UNROLLING IN TIME

What does processing a sequence look like?



TRAINING RNNs

Calculating the loss

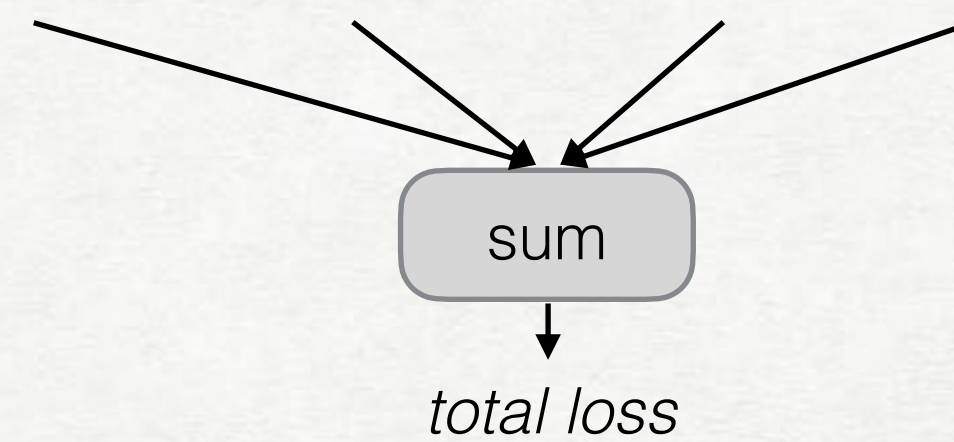


RNN TRAINING

The unrolled graph is a well-formed (DAG) computation graph—we can run backprop

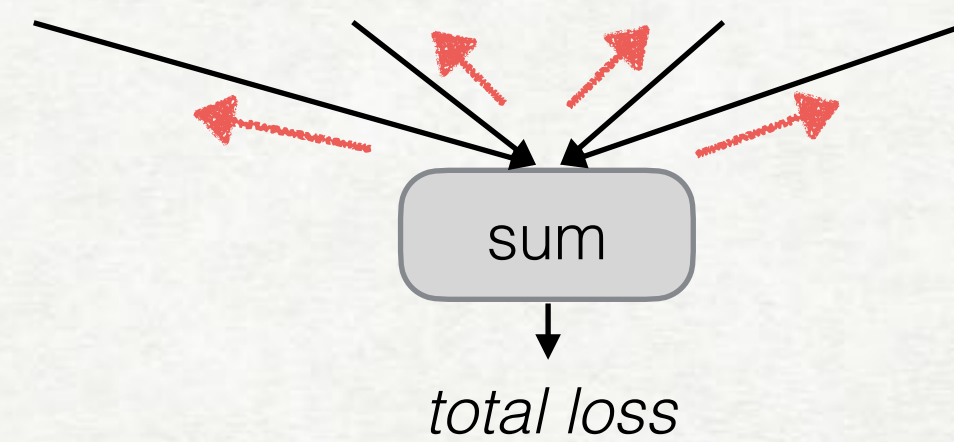
RNN TRAINING

The unrolled graph is a well-formed (DAG) computation graph—we can run backprop



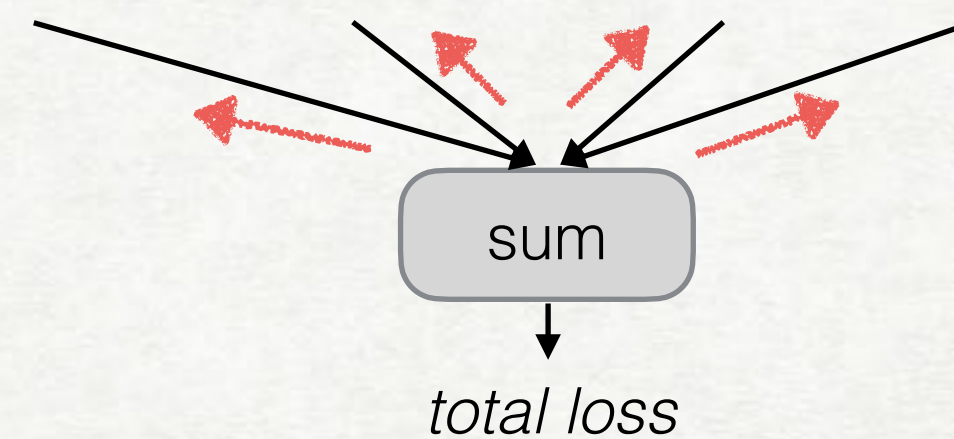
RNN TRAINING

The unrolled graph is a well-formed (DAG) computation graph—we can run backprop



RNN TRAINING

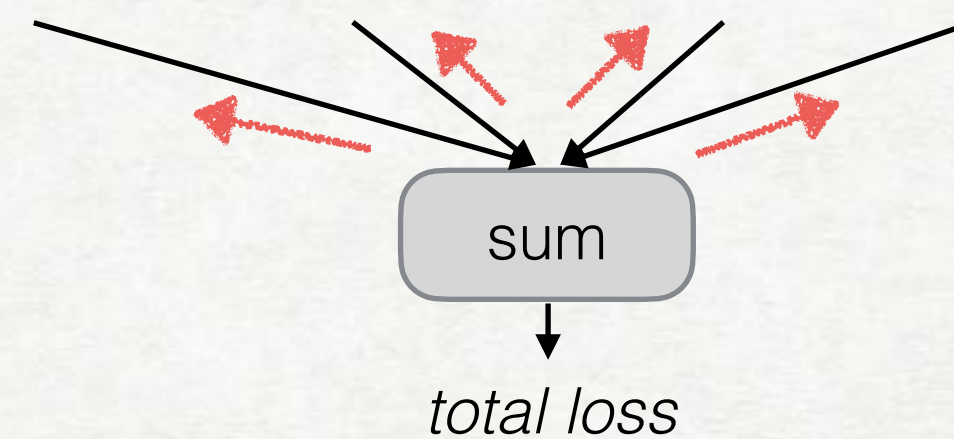
The unrolled graph is a well-formed (DAG) computation graph—we can run backprop



Parameters are tied across time, derivatives are aggregated across all time steps

RNN TRAINING

The unrolled graph is a well-formed (DAG) computation graph—we can run backprop



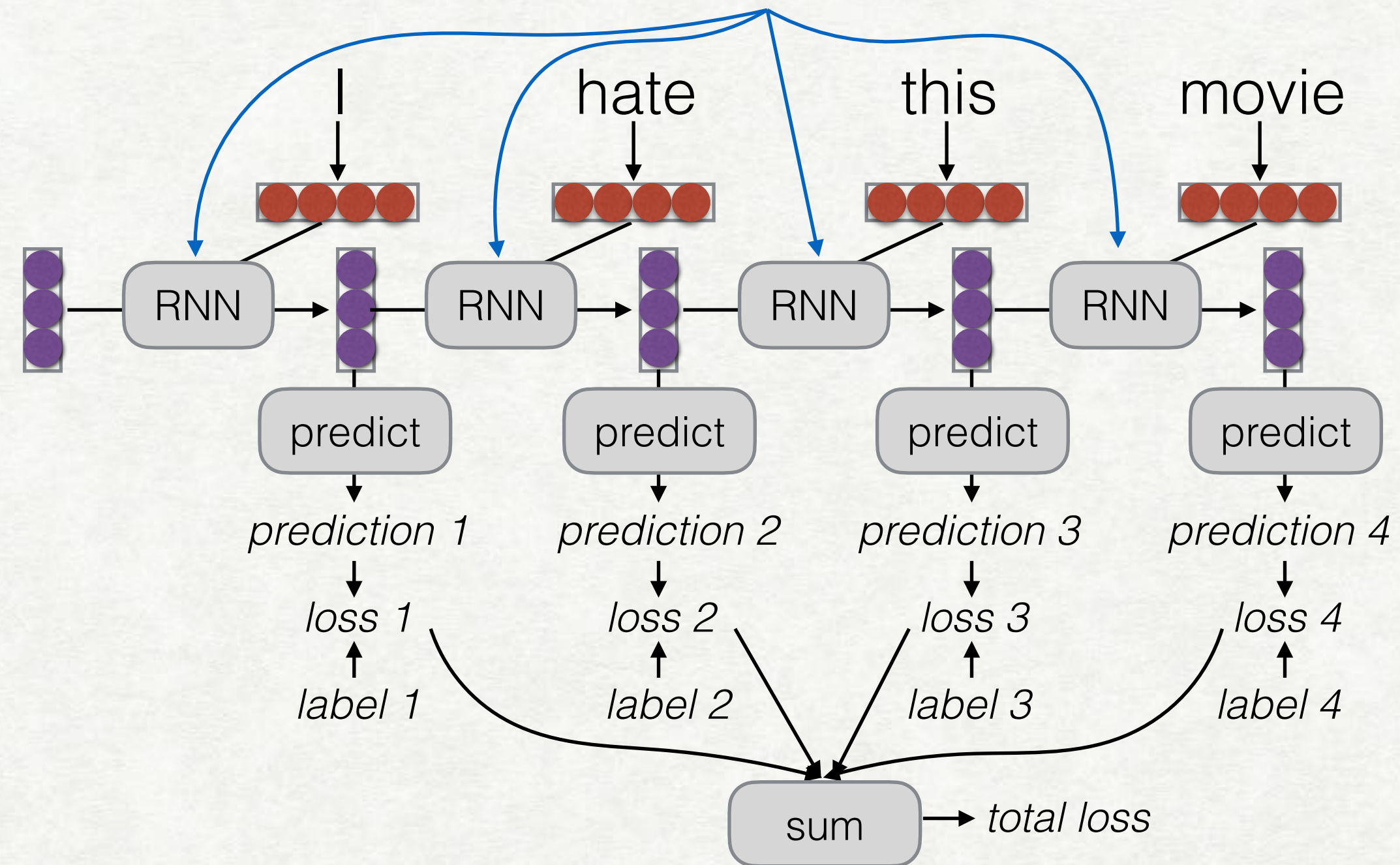
Parameters are tied across time, derivatives are aggregated across all time steps

This is historically called “backpropagation through time” (BPTT)

PARAMETER TYING

Calculating the loss

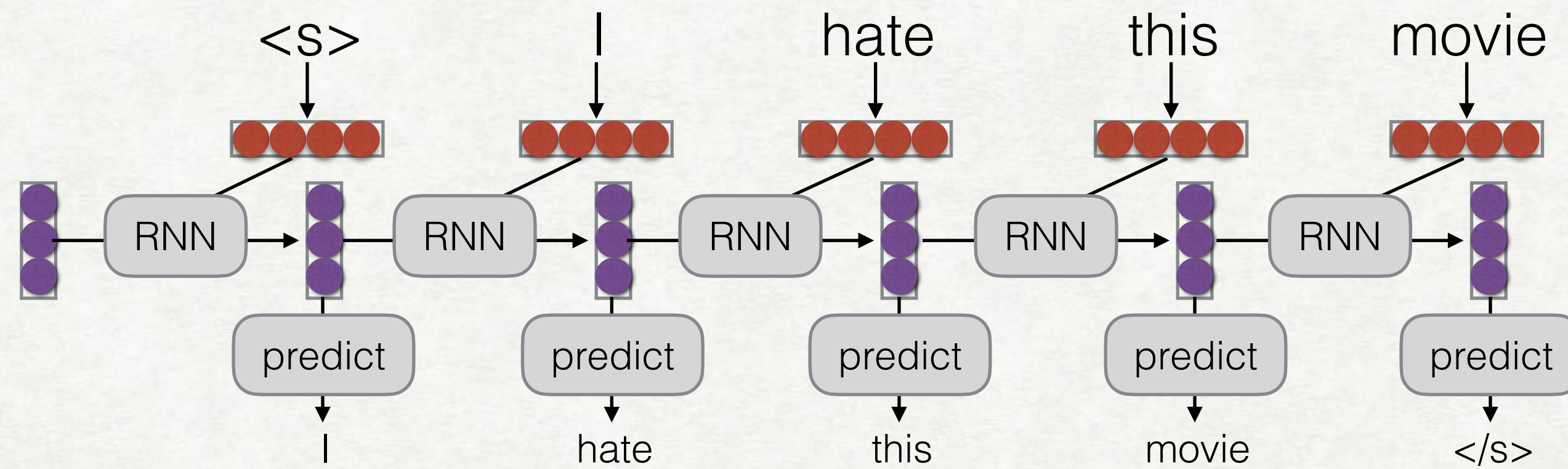
Parameters are shared! Derivatives are accumulated.



APPLICATIONS OF RNNs

E.G. LANGUAGE MODELING

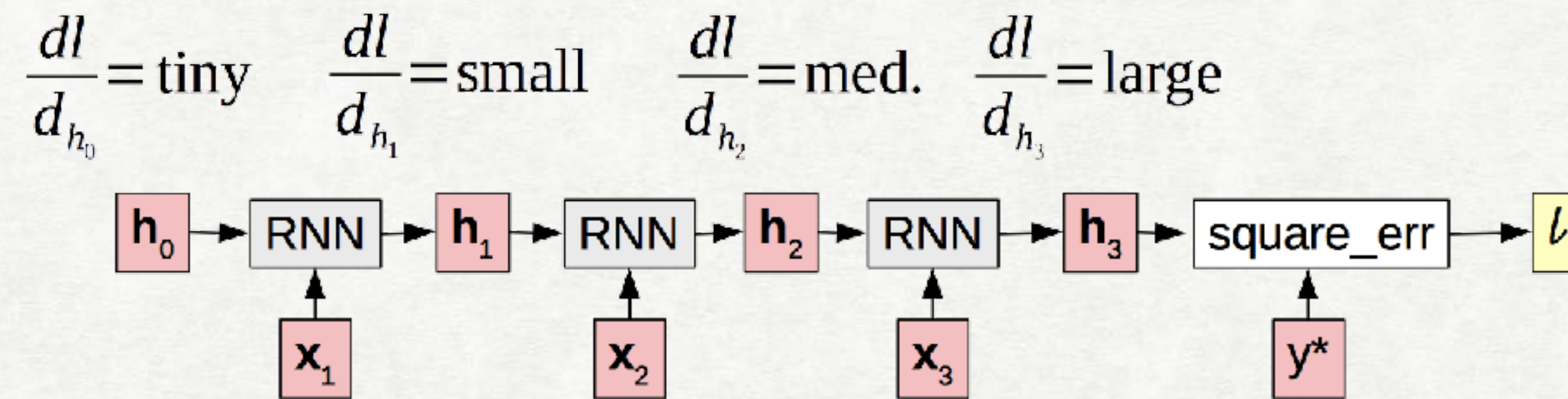
Language modeling is like a tagging task, where each tag is the next word!



VANISHING GRADIENTS

VANISHING GRADIENT

Gradients decrease as they get pushed back



Why? "Squashed" by non-linearities or small weights in matrices

A SOLUTION: LONG SHORT-TERM MEMORY
(HOCHREITER AND SCHMIDHUBER 1997)

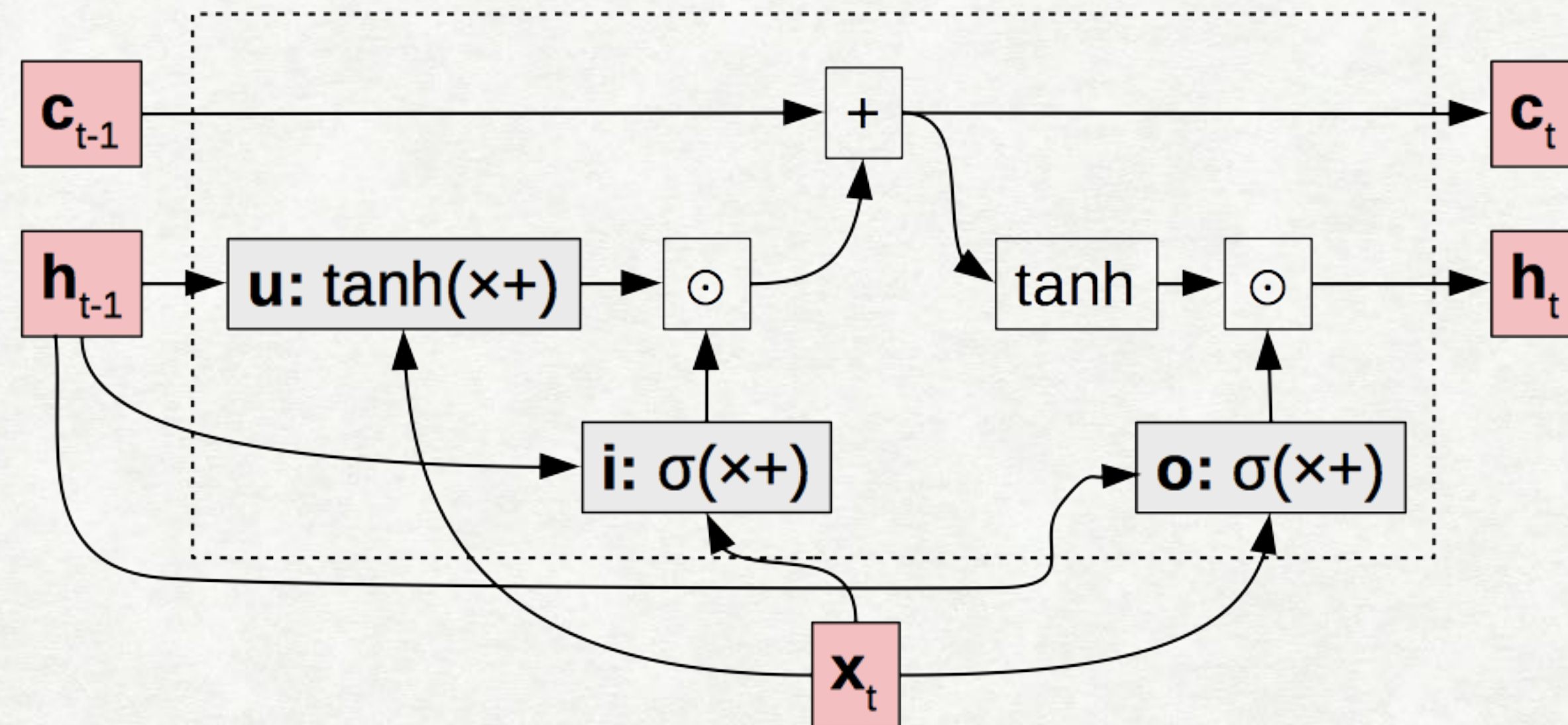
Basic idea: make additive connections between time steps

Addition does not modify the gradient, no vanishing

Gates to control the information flow

LSTM STRUCTURE

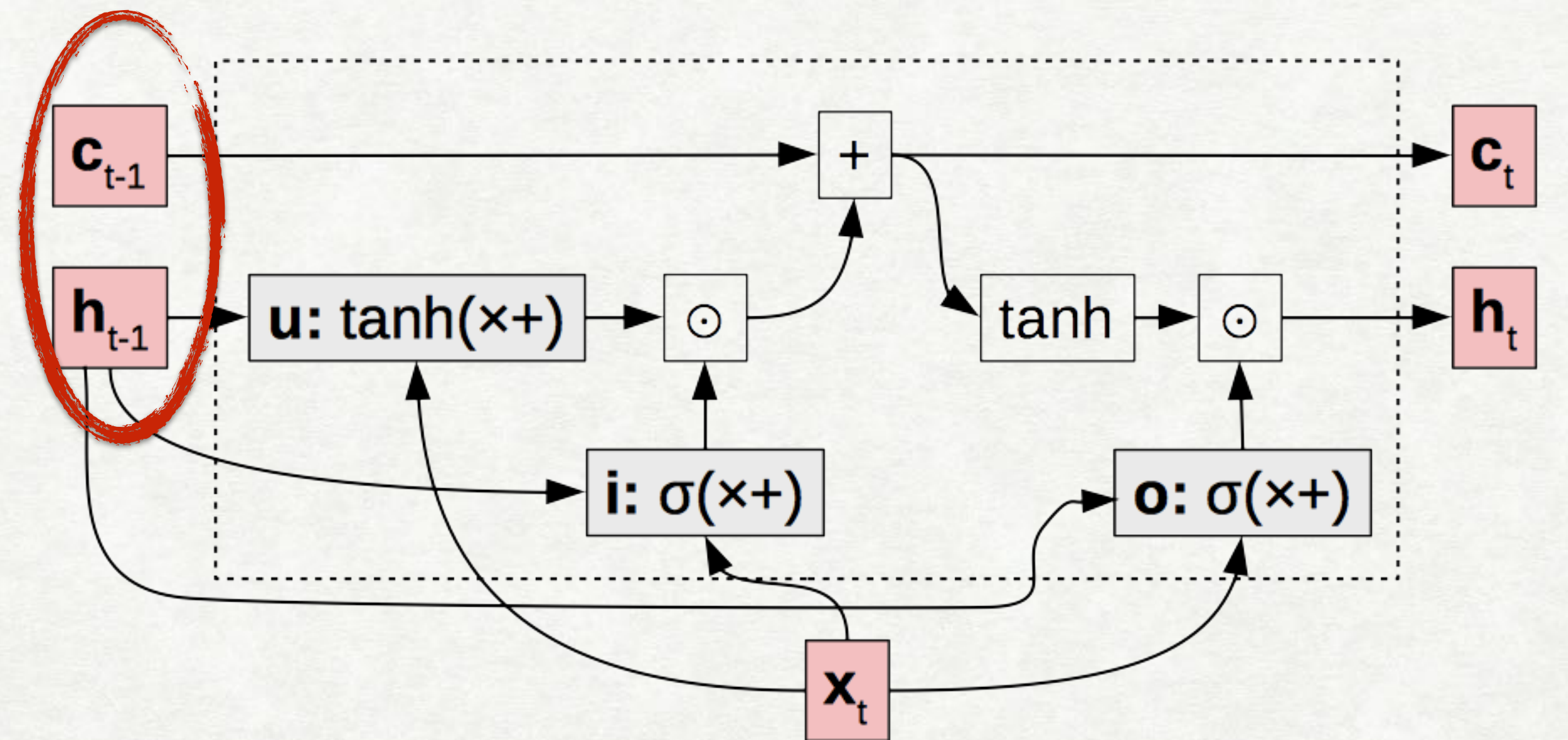
Most important idea: we want an additive connection between time steps.



update u : what value do we try to add to the memory cell?
input i : how much of the update do we allow to go through?
output o : how much of the cell do we reflect in the next state?

LSTM STRUCTURE

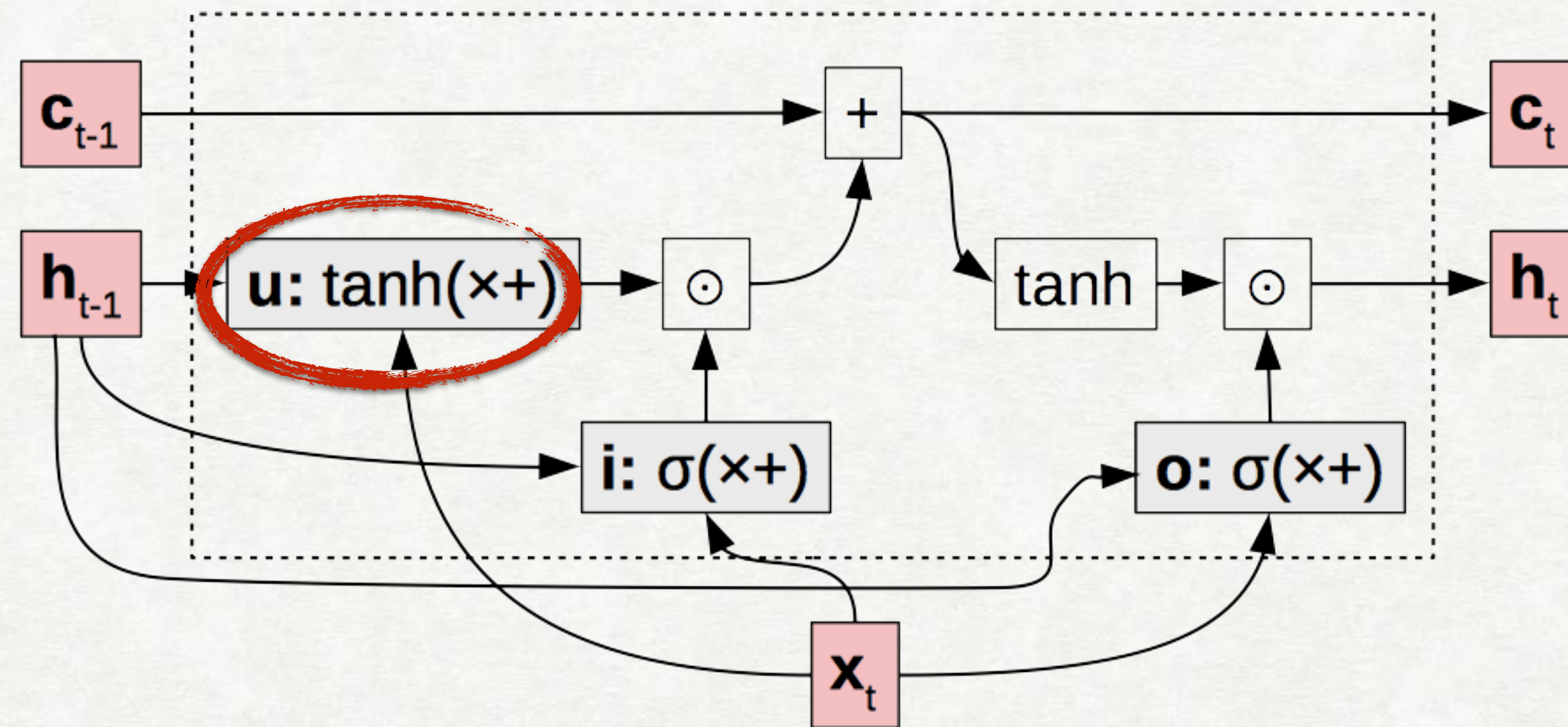
Most important idea: we want an additive connection between time steps.



update u : what value do we try to add to the memory cell?
input i : how much of the update do we allow to go through?
output o : how much of the cell do we reflect in the next state?

LSTM STRUCTURE

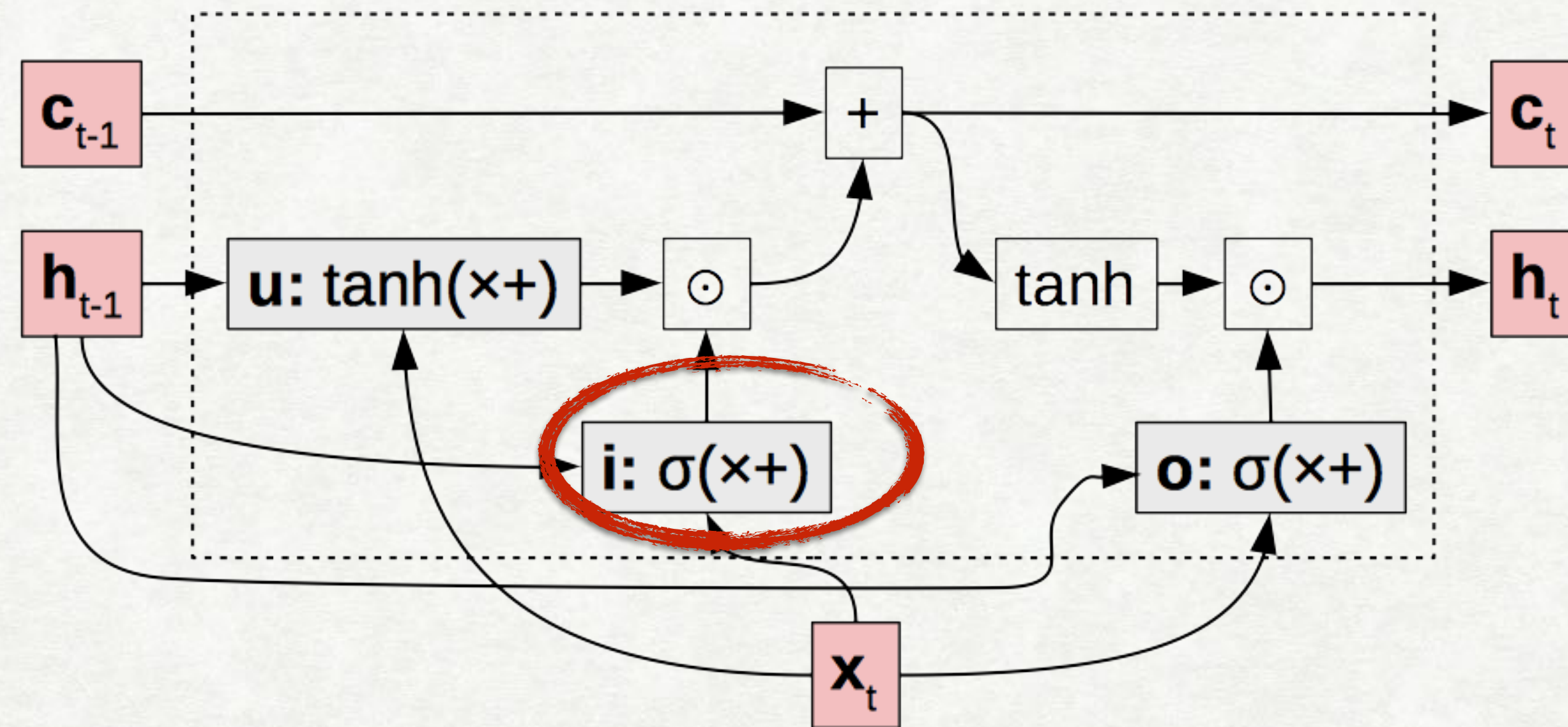
Most important idea: we want an additive connection between time steps



update u : what value do we try to add to the memory cell?
input i : how much of the update do we allow to go through?
output o : how much of the cell do we reflect in the next state?

LSTM STRUCTURE

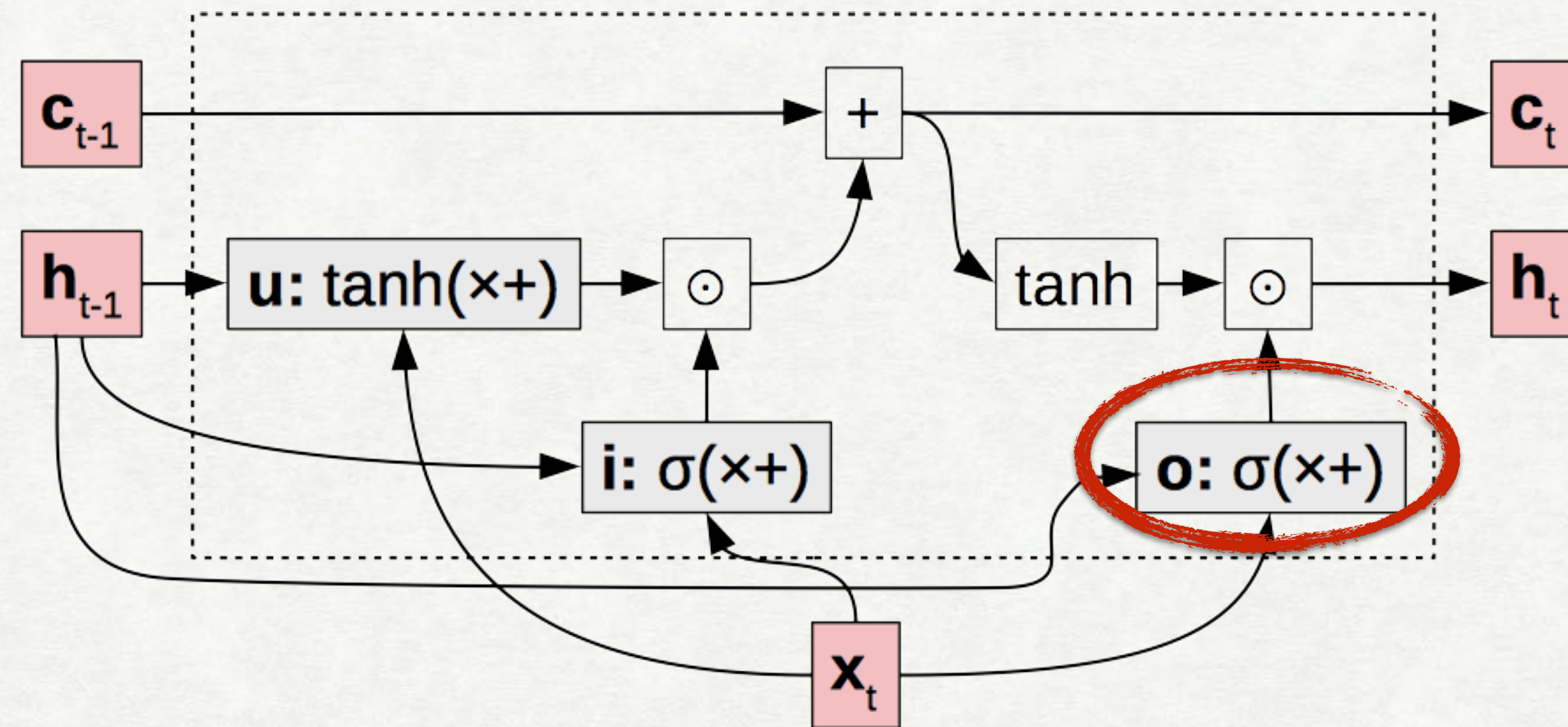
Most important idea: we want an additive connection between time steps.



update u : what value do we try to add to the memory cell?
input i : how much of the update do we allow to go through?
output o : how much of the cell do we reflect in the next state?

LSTM STRUCTURE

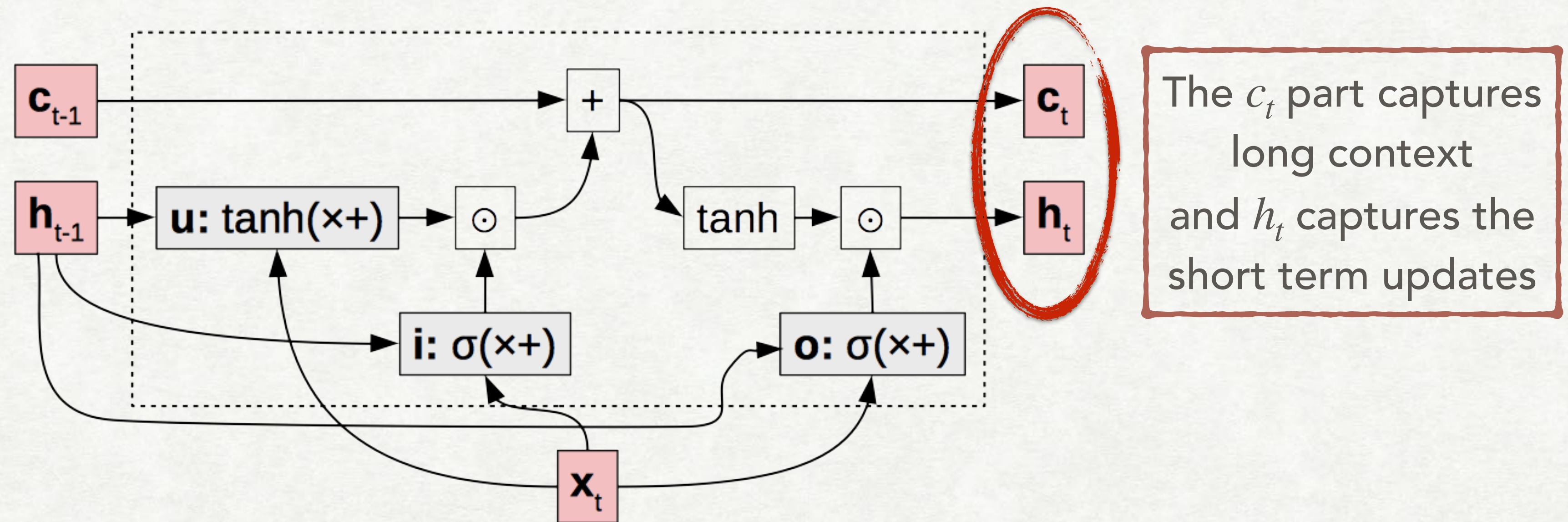
Most important idea: we want an additive connection between time steps.



update u : what value do we try to add to the memory cell?
input i : how much of the update do we allow to go through?
output o : how much of the cell do we reflect in the next state?

LSTM STRUCTURE

Most important idea: we want an additive connection between time steps.



The c_t part captures long context and h_t captures the short term updates

update u : what value do we try to add to the memory cell?
input i : how much of the update do we allow to go through?
output o : how much of the cell do we reflect in the next state?

WHAT CAN LSTMS LEARN? (1)

(KARPATHY ET AL. 2015)

Additive connections make single nodes surprisingly interpretable

Cell sensitive to position in line:

```
The sole importance of the crossing of the Berezina lies in the fact that it plainly and indubitably proved the fallacy of all the plans for cutting off the enemy's retreat and the soundness of the only possible line of action--the one Kutuzov and the general mass of the army demanded--namely, simply to follow the enemy up. The French crowd fled at a continually increasing speed and all its energy was directed to reaching its goal. It fled like a wounded animal and it was impossible to block its path. This was shown not so much by the arrangements it made for crossing as by what took place at the bridges. When the bridges broke down, unarmed soldiers, people from Moscow and women with children who were with the French transport, all--carried on by vis inertiae--pressed forward into boats and into the ice-covered water and did not, surrender.
```

Cell that turns on inside quotes:

```
"You mean to imply that I have nothing to eat out of... On the contrary, I can supply you with everything even if you want to give dinner parties," warmly replied Chichagov, who tried by every word he spoke to prove his own rectitude and therefore imagined Kutuzov to be animated by the same desire.
```

```
Kutuzov, shrugging his shoulders, replied with his subtle penetrating smile: "I meant merely to say what I said."
```

Cell that robustly activates inside if statements:

```
static int __dequeue_signal(struct sigpending *pending, sigset_t *mask, siginfo_t *info)
{
    int sig = next_signal(pending, mask);
    if (sig) {
        if (current->notifier) {
            if (sigismember(current->notifier_mask, sig)) {
                if (!(current->notifier)(current->notifier_data)) {
                    clear_thread_flag(TIF_SIGPENDING);
                    return 0;
                }
            }
        }
        collect_signal(sig, pending, info);
    }
    return sig;
}
```

A large portion of cells are not easily interpretable. Here is a typical example:

```
/* Unpack a filter field's string representation from user-space
 * buffer. */
char *audit_unpack_string(void **bufp, size_t *remain, size_t len)
{
    char *str;
    if (!*bufp || (len == 0) || (len > *remain))
        return ERR_PTR(-EINVAL);
    /* Of the currently implemented string fields, PATH_MAX
     * defines the longest valid length.
     */
}
```

Cell that turns on inside comments and quotes:

```
/* Duplicate LSM field information. The lsm_rule is opaque, so
 * re-initialized. */
static inline int audit_dupe_lsm_field(struct audit_field *df,
                                     struct audit_field *sf)
{
    int ret = 0;
    char *lsm_str;
    /* our own copy of lsm_str */
    lsm_str = kstrdup(sf->lsm_str, GFP_KERNEL);
    if (unlikely(!lsm_str))
        return -ENOMEM;
    df->lsm_str = lsm_str;
    /* our own (refreshed) copy of lsm_rule */
    ret = security_audit_rule_init(df->type, df->op, df->lsm_str,
                                   (void **) &df->lsm_rule);
    /* Keep currently invalid fields around in case they
     * become valid after a policy reload. */
    if (ret == -EINVAL) {
        pr_warn("audit rule for LSM '%s' is invalid\n",
                df->lsm_str);
        ret = 0;
    }
    return ret;
}
```

Cell that is sensitive to the depth of an expression:

```
#ifdef CONFIG_AUDITSYSCALL
static inline int audit_match_class_bits(int class, u32 *mask)
{
    int i;
    if (classes[class]) {
        for (i = 0; i < AUDIT_BITMASK_SIZE; i++)
            if (mask[i] & classes[class][i])
                return 0;
    }
    return 1;
}
```

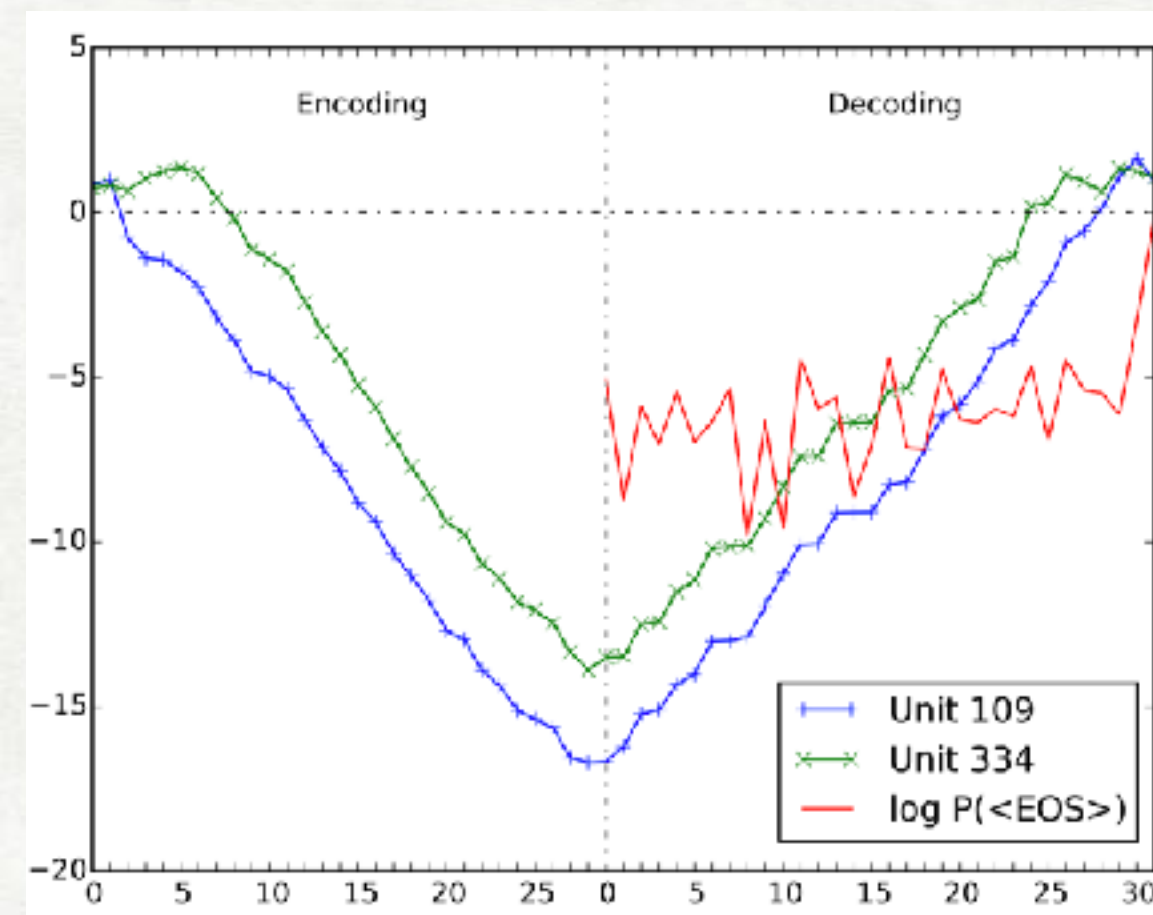
Cell that might be helpful in predicting a new line. Note that it only turns on for some *):

```
char *audit_unpack_string(void **bufp, size_t *remain, size_t len)
{
    char *str;
    if (!*bufp || (len == 0) || (len > *remain))
        return ERR_PTR(-EINVAL);
    /* Of the currently implemented string fields, PATH_MAX
     * defines the longest valid length.
     */
    if (len > PATH_MAX)
        return ERR_PTR(-ENAMETOOLONG);
    str = kmalloc(len + 1, GFP_KERNEL);
    if (unlikely(!str))
        return ERR_PTR(-ENOMEM);
    memcpy(str, *bufp, len);
    str[len] = 0;
    *bufp += len;
    *remain -= len;
    return str;
}
```


WHAT CAN LSTMS LEARN? (2)

(SHI ET AL. 2016, RADFORD ET AL. 2017)

Count length of sentence



Sentiment(?)

25 August 2003 League of Extraordinary Gentlemen: Sean Connery is one of the all time greats and I have been a fan of his since the 1950's. I went to this movie because Sean Connery was the main actor. I had not read reviews or had any prior knowledge of the movie. The movie surprised me quite a bit. The scenery and sights were spectacular, but the plot was unreal to the point of being ridiculous. In my mind this was not one of his better movies it could be the worst. Why he chose to be in this movie is a mystery. For me, going to this movie was a waste of my time. I will continue to go to his movies and add his movies to my video collection. But I can't see wasting money to put this movie in my collection

I found this to be a charming adaptation, very lively and full of fun. With the exception of a couple of major errors, the cast is wonderful. I have to echo some of the earlier comments -- Chynna Phillips is horribly miscast as a teenager. At 27, she's just too old (and, yes, it DOES show), and lacks the singing "chops" for Broadway-style music. Vanessa Williams is a decent-enough singer and, for a non-dancer, she's adequate. However, she is NOT Latina, and her character definitely is. She's also very STRIDENT throughout, which gets tiresome. The girls of Sweet Apple's Conrad Birdie fan club really sparkle -- with special kudos to Brigitta Dau and Chiara Zanni. I also enjoyed Tyne Daly's performance, though I'm not generally a fan of her work. Finally, the dancing Shriners are a riot, especially the dorky three in the bar. The movie is suitable for the whole family, and I highly recommend it.

NEXT CLASS PREVIEW

Language models produce good representations!

BERT and family