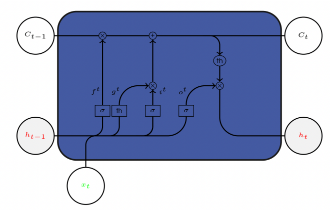


LSTM And GRU

Vincent Barra
LIMOS, UMR 6158 CNRS, Université Clermont Auvergne

Sequence modeling

- Handle variable-length inputs
- Share parameters across the sequence
-

Sequence modeling

- Handle variable-length inputs
- Share parameters across the sequence
- Keep track of long-term dependencies

Paul is a small boy and is hungry. He goes to the restaurant and eats a lot.

LIMOS LABORATOIRE D'INFORMATIQUE, DE MODÉLISATION ET D'OPTIMISATION DES SYSTÈMES

cnrs

uca UNIVERSITÉ Clermont Auvergne

Sequence modeling

- Handle variable-length inputs
- Share parameters across the sequence
- Keep track of long-term dependencies

Paul is a small boy and is hungry. He goes to the restaurant and eats a lot.

Solutions

- Activation function: using ReLU prevents σ' from shrinking the gradients when $x > 0$
- Weights and bias initialisation can help ($\boldsymbol{W} = \boldsymbol{I}, \boldsymbol{b}$=0)
- Think of another architecture

LIMOS LABORATOIRE D'INFORMATIQUE, DE MODÉLISATION ET D'OPTIMISATION DES SYSTÈMES

cnrs

UCA UNIVERSITÉ Clermont Auvergne

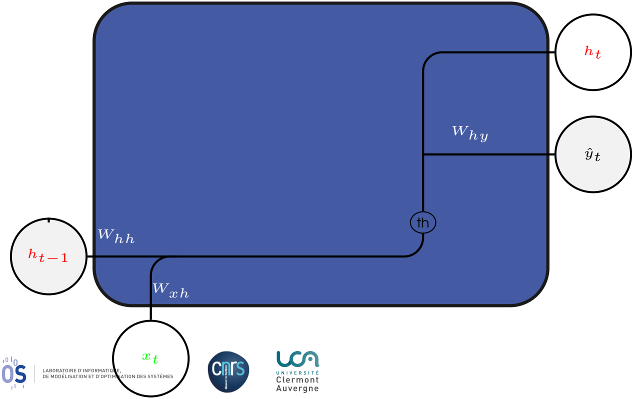
Recall: RNN

1. Update hidden state

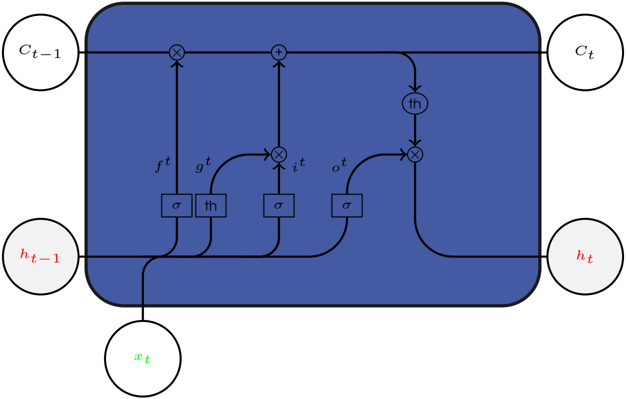
$$h_t = tanh(W_{xh}^T x_t + W_{hh}^T h_{t-1})$$

2. Compute output vector

$$\hat{y}_t = W_{hy}^T h_t$$



LSTM



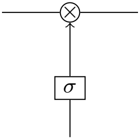
LSTM

Properties

- Long Short Term Memory networks rely on gated cell to allow information tracking through time, even for several timesteps
- Contains computational blocks controling information flow
- Key concept: a persistent cell state C_t module, representation of past history
- in Keras: `tensorflow.keras.layers.LSTM(num_units)`

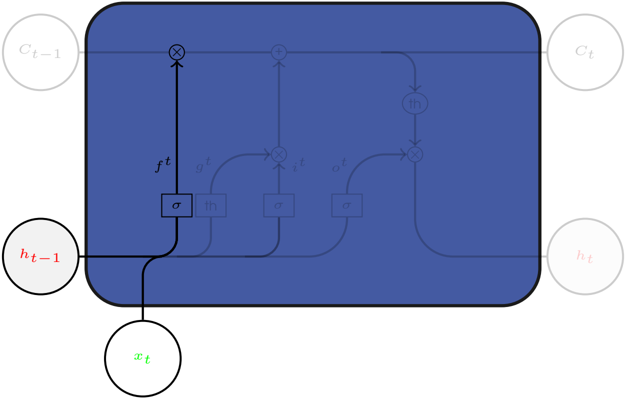
Gates

- Information is added or removed using gate structures.
- can let information through using eg. sigmoïd and pointwise multiplication



Inside LSTM: Forget gate

Forget irrelevant parts of the previous state



INSIDE LSTM: FORGET GATE

$$f^t = \sigma\left(\boldsymbol{W_f}^\top [\boldsymbol{x_t}, h_{t-1}] + b_f\right)$$

example: the state keeps the gender of the heroe in a text, for a good use of pronouns. If a new heroe appears, forget the gender.

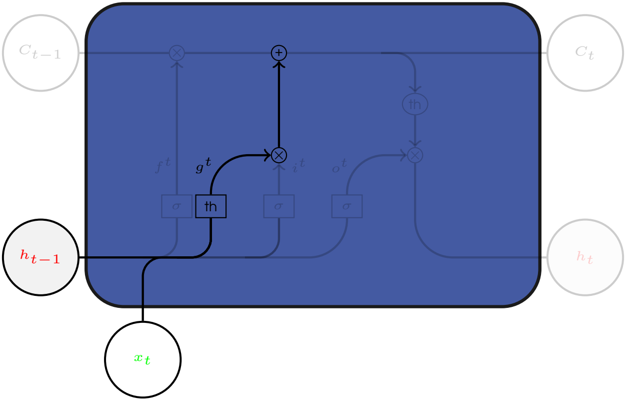
- $f_t = 0$: completely forgetting previous state
- $f_t = 1$: completely keeping previous state
- b_f should be initilized with large values so that initially $f_t \approx 1$

LABORATOIRE D'INFORMATIQUE, DE MODÉLISATION ET D'OPTIMISATION DES SYSTÈMES

UNIVERSITÉ Clermont Auvergne

Inside LSTM: Input neuron

Classic activation layer.

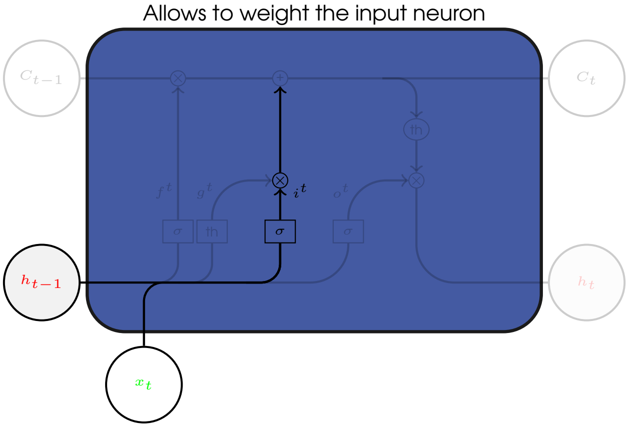


$$g^t = tanh\left(\boldsymbol{W_C}^\top [\boldsymbol{x_t}, h_{t-1}] + b_C\right)$$

LIMOS LABORATOIRE D'INFORMATIQUE, DE MODÉLISATION ET D'OPTIMISATION DES SYSTÈMES

UCA UNIVERSITÉ Clermont Auvergne

Inside LSTM: Update gate



LIMOS LABORATOIRE D'INFORMATIQUE, DE MODÉLISATION ET D'OPTIMISATION DES SYSTÈMES

cnrs

UCA UNIVERSITÉ Clermont Auvergne

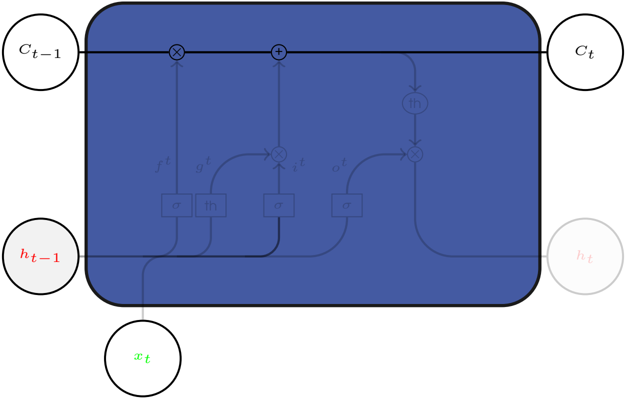
Inside LSTM: Update gate

$$i^t = \sigma\left(\boldsymbol{W_i}^\top \left[\boldsymbol{x_t}, h_{t-1}\right] + b_i\right)$$

- Decide the importance of the contribution of the input neuron at each timestep.
- g^t and i^t store relevant new information in the current state.

Inside LSTM: Update step

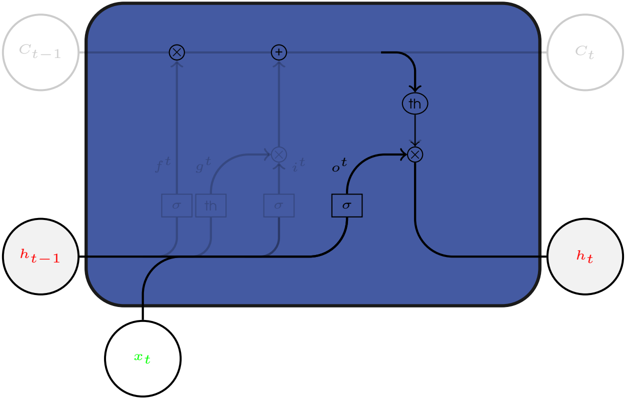
LSTM selectively update cell state values



$C_t = g^t.i^t + C_{t-1}.f^t$: assures that the derivatives of the loss w.r.t. C_t does not vanish.

Inside LSTM: Output gate

Controls what information is sent to the next timestep



$$o^t = \sigma\left(\boldsymbol{W_o}^\top [\boldsymbol{x_t}, h_{t-1}] + b_o\right) \qquad h_t = o^t tanh(C_t)$$

LIMOS LABORATOIRE D'INFORMATIQUE, DE MODÉLISATION ET D'OPTIMISATION DES SYSTÈMES

cnrs

UCA UNIVERSITÉ Clermont Auvergne

In summary:

$$\begin{aligned}
f^t &= \sigma\left(\boldsymbol{W_f}^\top [\boldsymbol{x_t}, h_{t-1}] + b_f\right) \\
g^t &= tanh\left(\boldsymbol{W_C}^\top [\boldsymbol{x_t}, h_{t-1}] + b_C\right) \\
i^t &= \sigma\left(\boldsymbol{W_i}^\top [\boldsymbol{x_t}, h_{t-1}] + b_i\right) \\
C_t &= g^t.i^t + C_{t-1}.f^t \\
o^t &= \sigma\left(\boldsymbol{W_o}^\top [\boldsymbol{x_t}, h_{t-1}] + b_o\right) \\
h_t &= o^t tanh(C_t)
\end{aligned}$$

LABORATOIRE D'INFORMATIQUE, DE MODÉLISATION ET D'OPTIMISATION DES SYSTÈMES

Key concepts

Concepts

1. Maintain a separate state C_t from what is outputted
2. Use gated to control the information flow
   - can forget information (f^t)
   - can store relevant information from the x_t (g^t)
   - can selectively update state (i^t)
   - can return a filtered version of the state (o^t)
3. introduction of self-loops to produce paths where gradients can flow for long durations
4. Since the backward flow from C_t to C_{t-1} is direct, backpropagation through time is computed with uninterrupted gradient flow.

Example

- Task: predicting the next word of a sentence based on previous ones.
- Hypothesis: the cell state C_t contains the gender of the subject

Paul is a clever guy. He has his own car. Cathy and Mary are their sisters.

1. First step (f^t): C_{t-1} contains the gender of the subject (Paul) to use proper pronouns (his). If a new subject arrive (Cathy and Mary), we may want to forget the old gender.
2. Second step: (g^t,i^t): what kind of information we want to store in C_t ? Here we may want to add the gender of the new subject (Cathy and Mary) to C_t
3. Third step (o^t): what we want to output. Since we see a new subject, we may want to output information relevant to verb (are)(eg. singular or plural subject)

LIMOS LABORATOIRE D'INFORMATIQUE, DE MODÉLISATION ET D'OPTIMISATION DES SYSTÈMES

UNIVERSITÉ Clermont Auvergne

Variants

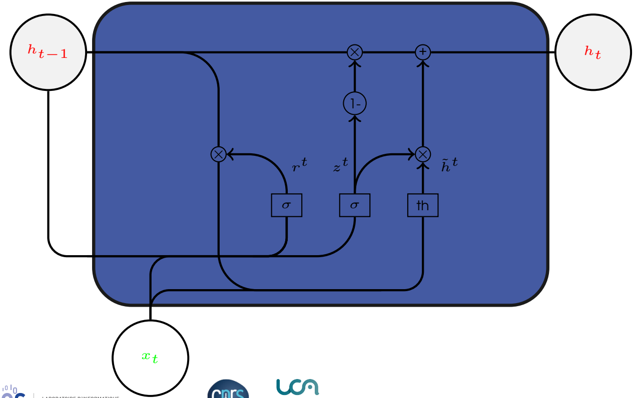
- LSTM with peephole connections
- Gates have access to C_{t-1}
- Bi-directional recurrent networks
- Gated Recurrent Units (GRU)
- Skip LSTMs
- ...

LIMOS LABORATOIRE D'INFORMATIQUE, DE MODÉLISATION ET D'OPTIMISATION DES SYSTÈMES

Gated Recurrent Units

Simplified version of LSTM:

- no o^t neither C_t
- only two gates
- easier to train



LIMOS LABORATOIRE D'INFORMATIQUE, DE MODÉLISATION ET D'OPTIMISATION DES SYSTÈMES cnrs UCA UNIVERSITÉ Clermont Auvergne

Gated Recurrent Units

- reset gate r^t: determines how to combine $\boldsymbol{x_t}$ with h_{t-1}
- update gate z^t: what quantity of memory must be preserved (like a combination of f^t and g^t)

$$
\begin{aligned}
r^t &= \sigma\left(\boldsymbol{W_r}^\top [\boldsymbol{x_t}, h_{t-1}] + b_r\right)\\
z^t &= \sigma\left(\boldsymbol{W_z}^\top [\boldsymbol{x_t}, h_{t-1}] + b_z\right)\\
\tilde{h}^t &= tanh\left(\boldsymbol{W}^\top \left[\boldsymbol{x_t}, r^t h_{t-1}\right] + b_h\right)\\
h_t &= \left(1 - z^t\right) h_{t-1} + z^t \tilde{h}^t
\end{aligned}
$$

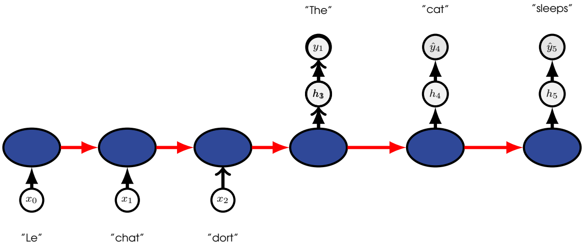
Gating network signals control how the present input and previous memory are used to update the current activation and produce the current state.

LABORATOIRE D'INFORMATIQUE, DE MODÉLISATION ET D'OPTIMISATION DES SYSTÈMES

Université Clermont Auvergne

Machine translation

Many-to-Many architecture



- the sentence in the source language is a sequence
- it is transformed in a latent space by an LSTM encoder
- the sentence in the target language is a sequence
- it is captured by a LSTM decoder

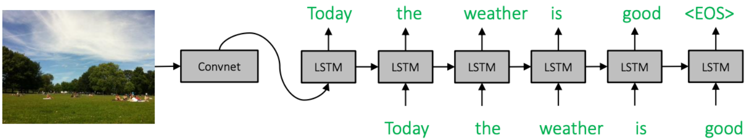
LIMOS LABORATOIRE D'INFORMATIQUE, DE MODÉLISATION ET D'OPTIMISATION DES SYSTÈMES

cnrs

UCA UNIVERSITÉ Clermont Auvergne

Image captioning

Many-to-Many architecture



Timeseries modeling

Environmental modeling

Keras

```
model = Sequential()
# 1 layer LSTM with batch size = 128
model.add(LSTM(128, input_shape=(...)))
model.add(Dense(..))
# Predicting the next word
model.add(Activation('softmax'))
model.compile(loss='binary_crossentropy',optimizer='adam', metrics=['accuracy'])
print(model.summary())


# Using embeddings
embedding_size = 32
model = Sequential()
model.add(Embedding(top_words, embedding_size, input_length=max_length))
model.add(LSTM(100))
model.add(Dense(1, activation='sigmoid'))
model.compile(loss='binary_crossentropy',optimizer='adam', metrics=['accuracy'])
print(model.summary())
```

LABORATOIRE D'INFORMATIQUE,
DE MODÉLISATION ET D'OPTIMISATION DES SYSTÈMES