Рекуррентные нейронные сети

Шаталов Николай Алексеевич

МГУ имени М. В. Ломоносова, факультет ВМК, кафедра ММП

15 марта 2018 г.

Содержание

Вспомним "обычные"нейросети

Нейросети прямого распространения (feed-forward neural networks) - нейросети, в которых информация передается только вперед по сети от слоя к слою.

Ограничения:

- Фиксированный размер входных и выходных данных
- С математической точки зрения ведет себя как обычная функция $y = f(x)$.

Вспомним "обычные"нейросети

Нейросети прямого распространения (feed-forward neural networks) - нейросети, в которых информация передается только вперед по сети от слоя к слою.

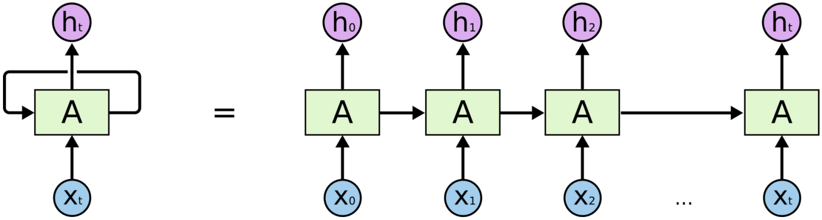
Ограничения:

- Фиксированный размер входных и выходных данных
- С математической точки зрения ведет себя как обычная функция $y = f(x)$.

Что если мы хотим работать с последовательностями, в которых важно не только содержание, но и порядок, в котором следует информация?

Рекуррентные нейронные сети

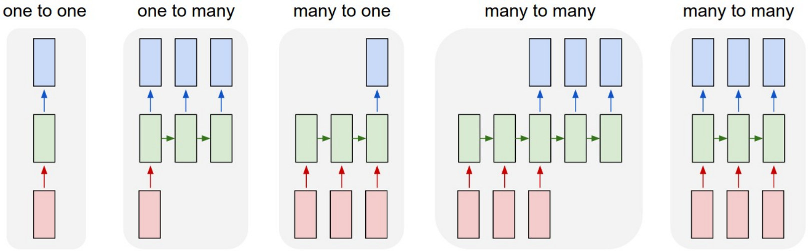
Рекуррентные нейронные сети содержат обратные связи. Например, нейроны могут передавать информацию о предыдущем состоянии нейронной сети.



- Может работать с последовательностью входных данных (условно бесконечной), так как имеет эффект памяти
- Ведет себя как программа (RNN – Тьюринг-полные)

1. Базовая модель рекуррентной нейронной сети
2. Примеры задач
3. Backpropagation Through Time
4. Архитектура LSTM

Последовательности



Примеры задач:

1. (Без RNN) Классификация изображений
2. Описание изображений
3. Анализ тональности
4. Машинный перевод (асинхронные последовательности)
5. Классификация видео, где требуется разметить каждый кадр (синхронные последовательности)

Последовательная обработка (не последовательностей)

1. RNN учится читать номера домов
2. RNN учится раскрашивать номера домов

Игрушечный пример. Генерация файлов LaTeX

For $\bigoplus_{n=1,\dots,m}$ where $\mathcal{L}_{m_\bullet} = 0$, hence we can find a closed subset $\mathcal{H}$ in $\mathcal{H}$ and any sets $\mathcal{F}$ on X, U is a closed immersion of S, then $U \to T$ is a separated algebraic space.

Proof. Proof of (1). It also start we get

$$S = \mathrm{Spec}(R) = U \times_X U \times_X U$$

and the comparicoly in the fibre product covering we have to prove the lemma generated by $\coprod Z \times_U U \to V$. Consider the maps M along the set of points Sch_{fppf} and $U \to U$ is the fibre category of S in U in Section, ?? and the fact that any U affine, see Morphisms, Lemma ??. Hence we obtain a scheme S and any open subset $W \subset U$ in $Sh(G)$ such that $\mathrm{Spec}(R') \to S$ is smooth or an

$$U = \bigcup U_i \times_{S_i} U_i$$

which has a nonzero morphism we may assume that f_i is of finite presentation over S. We claim that $\mathcal{O}_{X,x}$ is a scheme where $x, x', s'' \in S'$ such that $\mathcal{O}_{X,x'} \to \mathcal{O}'_{X',x'}$ is separated. By Algebra, Lemma ?? we can define a map of complexes $\mathrm{GL}_{S'}(x'/S'')$ and we win. □

To prove study we see that $\mathcal{F}|_U$ is a covering of $\mathcal{X}'$, and $\mathcal{T}_i$ is an object of $\mathcal{F}_{X/S}$ for $i > 0$ and $\mathcal{F}_p$ exists and let $\mathcal{F}_i$ be a presheaf of $\mathcal{O}_X$-modules on $\mathcal{C}$ as a $\mathcal{F}$-module. In particular $\mathcal{F} = U/\mathcal{F}$ we have to show that

$$\widetilde{M}^\bullet = \mathcal{I}^\bullet \otimes_{\mathrm{Spec}(k)} \mathcal{O}_{S,s} - i_X^{-1}\mathcal{F})$$

is a unique morphism of algebraic stacks. Note that

$$\mathrm{Arrows} = (Sch/S)^{opp}_{fppf}, (Sch/S)_{fppf}$$

and

$$V = \Gamma(S, \mathcal{O}) \longmapsto (U, \mathrm{Spec}(A))$$

is an open subset of X. Thus U is affine. This is a continuous map of X is the inverse, the groupoid scheme S.

Proof. See discussion of sheaves of sets. □

The result for prove any open covering follows from the less of Example ??. It may replace S by $X_{spaces, étale}$ which gives an open subspace of X and T equal to S_{Zar}, see Descent, Lemma ??. Namely, by Lemma ?? we see that R is geometrically regular over S.

Lemma 0.1. *Assume (3) and (3) by the construction in the description.*

Suppose $X = \lim |X|$ (by the formal open covering X and a single map $\underline{Proj}_X(\mathcal{A}) = \mathrm{Spec}(B)$ over U compatible with the complex

$$Set(\mathcal{A}) = \Gamma(X, \mathcal{O}_{X,\mathcal{O}_X}).$$

When in this case of to show that $\mathcal{Q} \to \mathcal{C}_{Z/X}$ is stable under the following result in the second conditions of (1), and (3). This finishes the proof. By Definition ?? (without element is when the closed subschemes are catenary. If T is surjective we may assume that T is connected with residue fields of S. Moreover there exists a closed subspace $Z \subset X$ of X where U in X' is proper (some defining as a closed subset of the uniqueness it suffices to check the fact that the following theorem

(1) *f is locally of finite type. Since $S = \mathrm{Spec}(R)$ and $Y = \mathrm{Spec}(R)$.*

Proof. This is form all sheaves of sheaves on X. But given a scheme U and a surjective étale morphism $U \to X$. Let $U \cap U = \coprod_{i=1,\dots,n} U_i$ be the scheme X over S at the schemes $X_i \to X$ and $U = \lim_i X_i$. □

The following lemma surjective restrocomposes of this implies that $\mathcal{F}_{x_0} = \mathcal{F}_{x_0} = \mathcal{F}_{X,\dots,0}$.

Lemma 0.2. *Let X be a locally Noetherian scheme over S, $E = \mathcal{F}_{X/S}$. Set $\mathcal{I} = \mathcal{J}_1 \subset \mathcal{I}'_n$. Since $\mathcal{I}^n \subset \mathcal{I}^n$ are nonzero over $i_0 \leq \mathfrak{p}$ is a subset of $\mathcal{J}_{n,0} \circ \overline{A}_2$ works.*

Lemma 0.3. *In Situation ??. Hence we may assume $\mathfrak{q}' = 0$.*

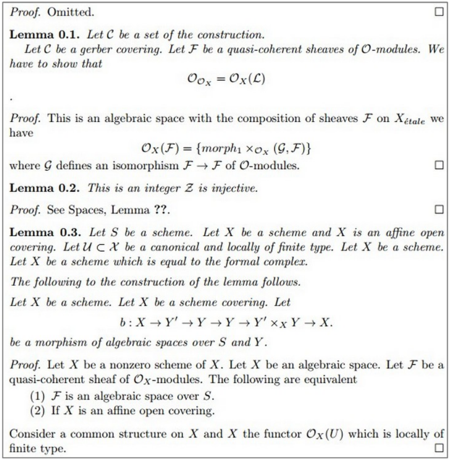
Proof. We will use the property we see that $\mathfrak{p}$ is the mext functor (??). On the other hand, by Lemma ?? we see that

$$D(\mathcal{O}_{X'}) = \mathcal{O}_X(D)$$

where K is an F-algebra where δ_{n+1} is a scheme over S. □

Рекуррентная нейронная сеть, обученная на исходнике учебника по алгебре, пытается генерировать новые файлы.

Игрушечный пример. Генерация файлов LaTeX



Proof. Omitted. □

Lemma 0.1. *Let $\mathcal{C}$ be a set of the construction.*

Let $\mathcal{C}$ be a gerber covering. Let $\mathcal{F}$ be a quasi-coherent sheaves of $\mathcal{O}$-modules. We have to show that

$$\mathcal{O}_{\mathcal{O}_X} = \mathcal{O}_X(\mathcal{L})$$

.

Proof. This is an algebraic space with the composition of sheaves $\mathcal{F}$ on $X_{\acute{e}tale}$ we have

$$\mathcal{O}_X(\mathcal{F}) = \{morph_1 \times_{\mathcal{O}_X} (\mathcal{G}, \mathcal{F})\}$$

where $\mathcal{G}$ defines an isomorphism $\mathcal{F} \to \mathcal{F}$ of $\mathcal{O}$-modules. □

Lemma 0.2. *This is an integer $\mathcal{Z}$ is injective.*

Proof. See Spaces, Lemma ??. □

Lemma 0.3. *Let S be a scheme. Let X be a scheme and X is an affine open covering. Let $\mathcal{U} \subset \mathcal{X}$ be a canonical and locally of finite type. Let X be a scheme. Let X be a scheme which is equal to the formal complex.*

The following to the construction of the lemma follows.

Let X be a scheme. Let X be a scheme covering. Let

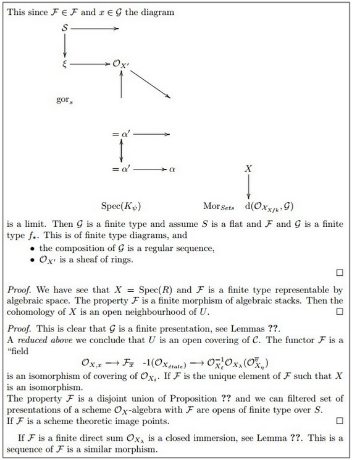
$$b : X \to Y' \to Y \to Y \to Y' \times_X Y \to X.$$

be a morphism of algebraic spaces over S and Y.

Proof. Let X be a nonzero scheme of X. Let X be an algebraic space. Let $\mathcal{F}$ be a quasi-coherent sheaf of $\mathcal{O}_X$-modules. The following are equivalent

(1) $\mathcal{F}$ is an algebraic space over S.
(2) If X is an affine open covering.

Consider a common structure on X and X the functor $\mathcal{O}_X(U)$ which is locally of finite type. □



This since $\mathcal{F} \in \mathcal{F}$ and $x \in \mathcal{G}$ the diagram

$$\mathcal{S} \longrightarrow \qquad \xi \longrightarrow \mathcal{O}_{X'} \qquad \mathrm{gor}_s \qquad =\alpha' \longrightarrow \qquad =\alpha' \longrightarrow \alpha \qquad X \qquad \mathrm{Spec}(K_\psi) \qquad \mathrm{Mor}_{Sets} \quad \mathrm{d}(\mathcal{O}_{\mathcal{X}_{X/k}}, \mathcal{G})$$

is a limit. Then $\mathcal{G}$ is a finite type and assume S is a flat and $\mathcal{F}$ and $\mathcal{G}$ is a finite type f_*. This is of finite type diagrams, and

- the composition of $\mathcal{G}$ is a regular sequence,
- $\mathcal{O}_{X'}$ is a sheaf of rings.

□

Proof. We have see that $X = \mathrm{Spec}(R)$ and $\mathcal{F}$ is a finite type representable by algebraic space. The property $\mathcal{F}$ is a finite morphism of algebraic stacks. Then the cohomology of X is an open neighbourhood of U. □

Proof. This is clear that $\mathcal{G}$ is a finite presentation, see Lemmas ??.
A *reduced above* we conclude that U is an open covering of $\mathcal{C}$. The functor $\mathcal{F}$ is a "field

$$\mathcal{O}_{X,x} \longrightarrow \mathcal{F}_{\overline{x}} \;\; \text{-1}(\mathcal{O}_{X_{\acute{e}tale}}) \longrightarrow \mathcal{O}_{X_\ell}^{-1}\mathcal{O}_{X_\lambda}(\mathcal{O}_{X_\eta}^{\overline{v}})$$

is an isomorphism of covering of $\mathcal{O}_{X_i}$. If $\mathcal{F}$ is the unique element of $\mathcal{F}$ such that X is an isomorphism.
The property $\mathcal{F}$ is a disjoint union of Proposition ?? and we can filtered set of presentations of a scheme $\mathcal{O}_X$-algebra with $\mathcal{F}$ are opens of finite type over S.
If $\mathcal{F}$ is a scheme theoretic image points. □

If $\mathcal{F}$ is a finite direct sum $\mathcal{O}_{X_\lambda}$ is a closed immersion, see Lemma ??. This is a sequence of $\mathcal{F}$ is a similar morphism.

Рекуррентная нейронная сеть, обученная на исходнике учебника по алгебре, пытается генерировать новые файлы.

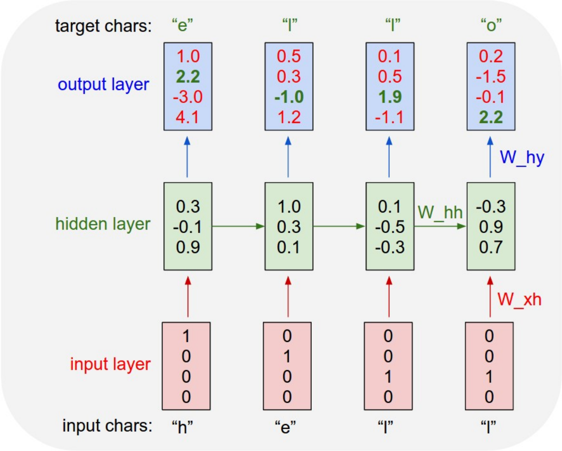
Игрушечный пример. Генерация файлов LaTeX

Пример выдачи модели:

```
\begin{proof}
We may assume that $\mathcal{I}$ is an abelian ...
\begin{enumerate}
\item \hyperref[setain-construction-phantom]{Lemma}
\item The following are equivalent
\begin{enumerate}
\item $\mathcal{F}$ is an $\mathcal{O}_X$-module.
\end{lemma}
```

Можно заметить, что модель генерирует код с ошибками. Например, в начале модель открыла блок proof, а закрыла уже блок lemma.

Игрушечная модель. Character-Level Language Model



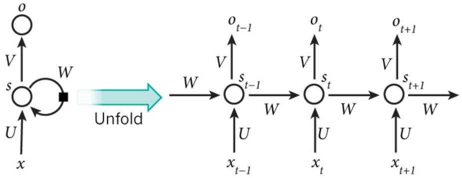
Модель учится по символу предсказывать его продолжение.
Обучающая выборка: последовательность сиволов „hello“.

Опишем модель рекуррентной нейронной сети.

$$
\begin{aligned}
s_t &= \tanh\left(U x_t + W s_{t-1}\right), \\
\widehat{y}_t &= \operatorname{softmax}\left(V s_t\right).
\end{aligned}
$$

Функция потерь – кросс-энтропия.

$$
\begin{aligned}
E_t\left(y_t, \hat{y}_t\right) &= -\sum_i y_t^i \log \hat{y}_t^i, \\
E_t\left(y, \hat{y}\right) &= \sum_t E_t\left(y_t, \hat{y}_t\right) = -\sum_t \sum_i y_t^i \log \hat{y}_t^i.
\end{aligned}
$$



Backpropagation Through Time. Параметр V

Найдем градиент функции потерь по параметрам U, V и W.
Градиент разбивается на сумму компонент: $\frac{\partial E}{\partial W} = \sum\limits_t \frac{\partial E_t}{\partial W}$.

Градиент функции потерь по параметру V:

$$\frac{\partial E_t}{\partial V} = \frac{\partial E_t}{\partial \hat{y}_t}\frac{\partial \hat{y}_t}{\partial V} = \frac{\partial E_t}{\partial \hat{y}_t}\frac{\partial \hat{y}_t}{\partial z_t}\frac{\partial z_t}{\partial V} = (\hat{y}_t - y_t) \otimes s_t,$$

где $z_t = V s_t$.

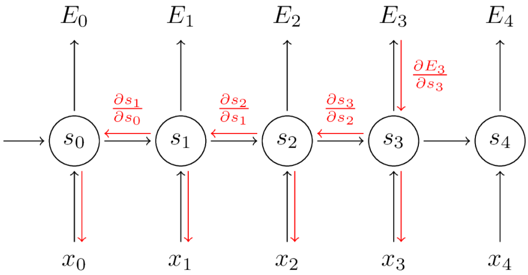
Заметим, что $\frac{\partial E_t}{\partial V}$ зависит только от значений текущего шага.

Backpropagation Through Time. Параметры W и U

Градиент функции потерь по параметру W (аналогично U):

$$\frac{\partial E_t}{\partial W} = \frac{\partial E_t}{\partial \hat{y}_t}\frac{\partial \hat{y}_t}{\partial z_t}\frac{\partial z_t}{\partial s_t}\frac{\partial s_t}{\partial W} = (\hat{y}_t - y_t)\frac{\partial z_t}{\partial s_t}\sum_{k=0}^{t}\frac{\partial s_t}{\partial s_k}\frac{\partial s_k}{\partial W}.$$

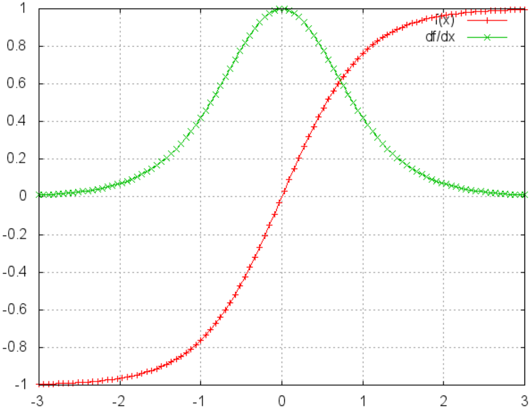
Вспомним, что $s_t = \tanh\left(Ux_t + Ws_{t-1}\right)$.



$$\frac{\partial s_t}{\partial W} = \frac{\partial s_t}{\partial W} + \frac{\partial s_t}{\partial s_{t-1}}\frac{\partial s_{t-1}}{\partial W} + \frac{\partial s_t}{\partial s_{t-1}}\frac{\partial s_{t-1}}{\partial s_{t-2}}\frac{\partial s_{t-2}}{\partial W} + \ldots = \sum_{k=0}^{t}\frac{\partial s_t}{\partial s_k}\frac{\partial s_k}{\partial W}.$$

Исчезающий градиент

Функция активации $f(x) = \tanh(x)$.



Сильное зануление при больших по модулю аргументах.

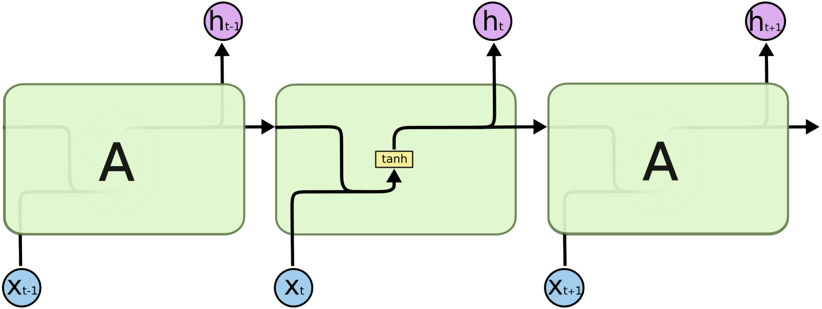
Исчезающий градиент

Модель начинает довольно быстро забывать информацию, а это критично в многих задачах.

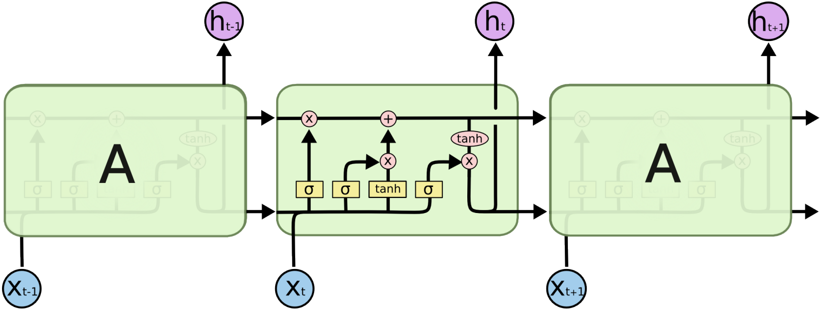
Как решать проблему?

- Правильный подбор начальных значений параметров W, U
- Использовать функцию активации ReLU
- Использовать Long Short-Term Memory (LSTM, 1997)
- Использовать Gated Recurrent Unit (GRU, 2014)

Стандартная RNN.

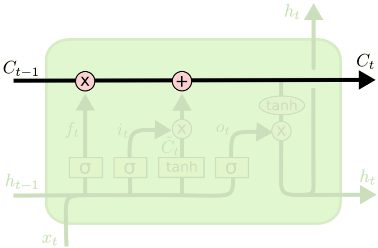


LSTM – Long Short Term Memory network.
В LSTM такая же цепочная структура, но теперь 4 слоя.



Cell State

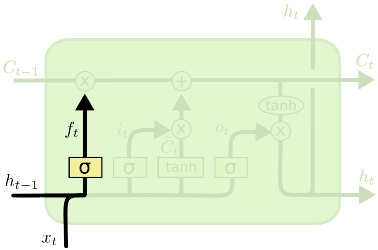
Главный ключ к LSTM – cell state.



LSTM может убирать или добавлять информацию в cell state, контролируемую специальными структурами - gates.

Первый gate

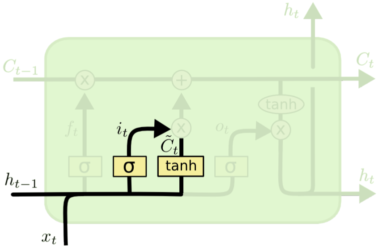
Forget gate layer – что нам больше не нужно хранить?



$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] \ + \ b_f\right)$$

Второй и третий gate

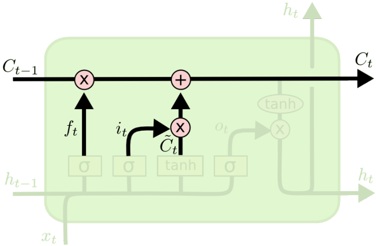
Input gate layer – какие значения нужно обновить?
$\tanh$ layer – новые кандидаты $\tilde{C}_t$, которые нужно сохранить.



$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] \ + \ b_i\right)$$
$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] \ + \ b_C)$$

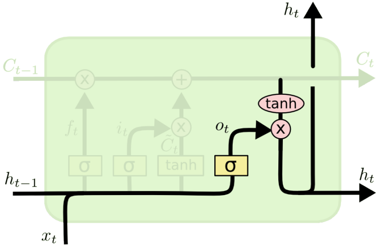
Обновление

Обновляем состояние LSTM.



$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

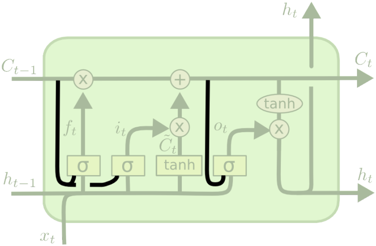
Решаем что LSTM должна выдать.
Фильтрация с помощью cell state.



$$o_t = \sigma\left(W_o \,[h_{t-1}, x_t] \;+\; b_o\right)$$
$$h_t = o_t * \tanh\left(C_t\right)$$

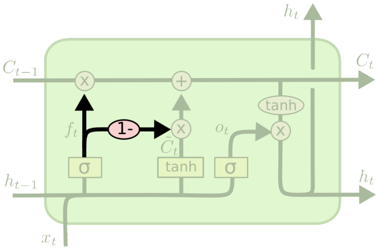
Разновидности LSTM. Peephole connections

Позволяем каждому gate заглядывать в cell state.



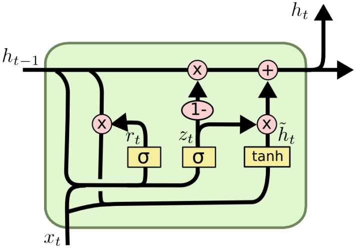
$$f_t = \sigma\left(W_f \cdot [\boldsymbol{C_{t-1}}, h_{t-1}, x_t] \;+\; b_f\right)$$
$$i_t = \sigma\left(W_i \cdot [\boldsymbol{C_{t-1}}, h_{t-1}, x_t] \;+\; b_i\right)$$
$$o_t = \sigma\left(W_o \cdot [\boldsymbol{C_t}, h_{t-1}, x_t] \;+\; b_o\right)$$

Вместо того, чтобы отдельно решать что забыть и что обновить, забываем только те значения, на место которых придут новые обновленные.



$$C_t = f_t * C_{t-1} + (\mathbf{1} - \boldsymbol{f_t}) * \tilde{C}_t$$

Более простая модель LSTM, особо популярная в последнее время.
Она объединяет forget и input layer, а также объединяет cell state и hidden state.



$$z_t = \sigma\left(W_z \cdot [h_{t-1}, x_t]\right)$$

$$r_t = \sigma\left(W_r \cdot [h_{t-1}, x_t]\right)$$

$$\tilde{h}_t = \tanh\left(W \cdot [r_t * h_{t-1}, x_t]\right)$$

$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t$$

Ссылки (кликабельны)

- The Unreasonable Effectiveness of Recurrent Neural Networks. Andrej Karpathy blog
- RNN tutorial on WildML
- Gradients for an RNN. Carter N Brown
- Understanding LSTM Networks. colah's blog