

pred tem smo si ogledali nevronske mreže, ki obdelujejo tabelarne podatke
(1. del prejšnjih predavanj) – grafomski n.m.
→ ni optimalno za slike (obdelano pixlov)

Rekurenčne nevronske mreže

Ljupčo Todorovski

Univerza v Ljubljani, Fakulteta za matematiko in fiziko
Institut Jožef Stefan, Odsek za tehnologije znanja (E8)

Maj 2023

za sliko trdimo



ta pixel je povezan z vsemi svojimi sosedji

Pregled predavanja

1 Motivacija

dolgi kratkoročni spomin
razne last odgov na vprašanje kaj (kak naredijo s sekvencami)

2 Rekurenčne nevronske mreže

- Osnovna ideja in težave
- Enota LSTM, dolgi kratkoročni spomin
long short term memory
- GRU, rekurenčna enota z vrti

na osnovi tečaj bošča predstavljeni te dve rednici

3 Posplošitvi rekurenčnih NM

- Rekurenčna enota z vrti za dvojiška drevesa
- Mehanizem pozornosti

Zaporedni podatki in časovne vrste

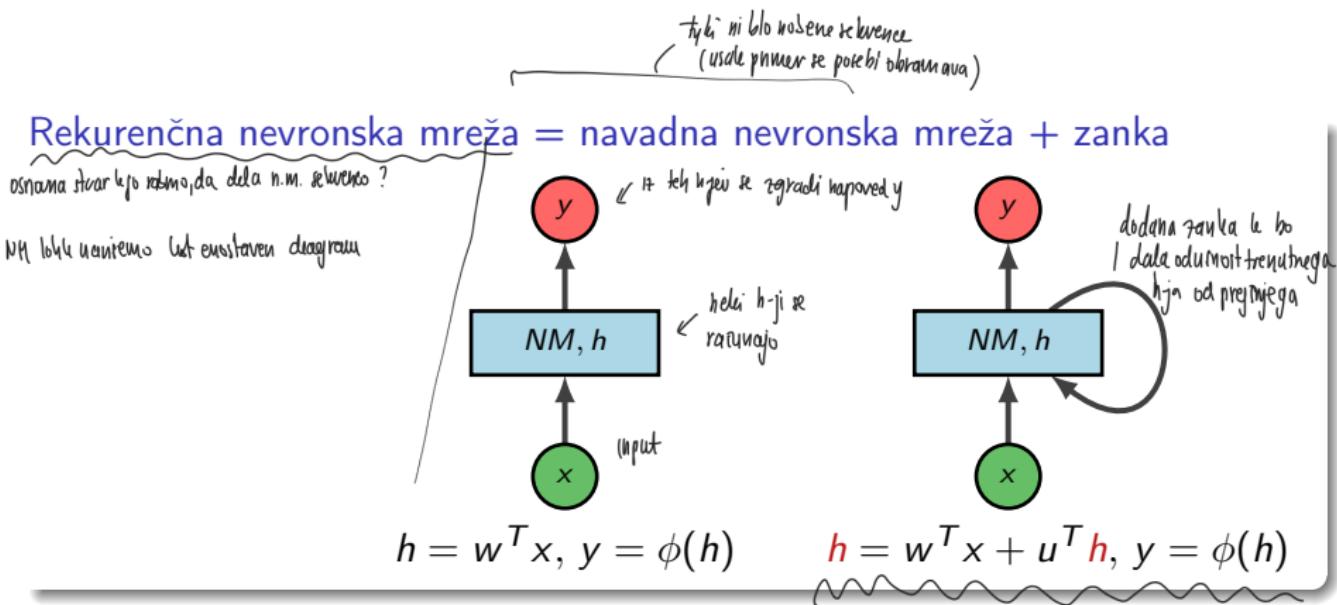
Primeri zaporednih podatkov

- Besedila so zaporedja besed, besede zaporedja črk
- Zvočni in video posnetki

Časovne vrste

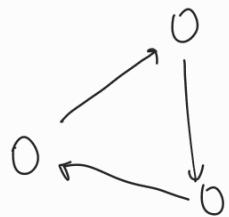
- Finančne časovne vrste, indeksi, cene, volumni trgovanja
- Temperatura, padavine, veter, kakovost zraka
- Zaporedja celih števil, npr. 1, 1, 2, 3, 5, 8, 13, ...
- Število obolelih z opazovano nalezljivo boleznijo

Ideja: Nevronske mreže dodamo zanko



To ni več usmerjena nevronska mreža, težave z računanjem vrednosti h .

Příklad primitivní zakávky s neuronami

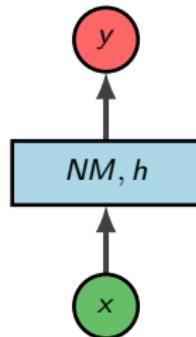


Tyto neuroni povídají + zakávka. → mezi sebou si po zakávce správce
správce ⇒ en za druhim
je uplynulo

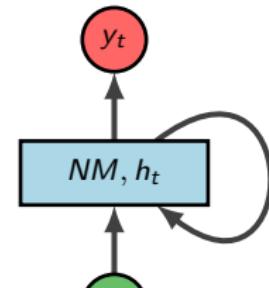
to lze shrnout do funkce $f(x) = \alpha f_1(x) + \beta f_2(x)$
(α je trojici lze říct dle délky) až β
sau trojici v ∞

Rešitev težave: časovni indeks t

Rekurenčna nevronska mreža = navadna nevronska mreža + zanka + čas



$$h = w^T x, \quad y = \phi(h)$$



$$h_t = w^T x_t + u^T h_{t-1}, \quad y_t = \phi(h_t)$$

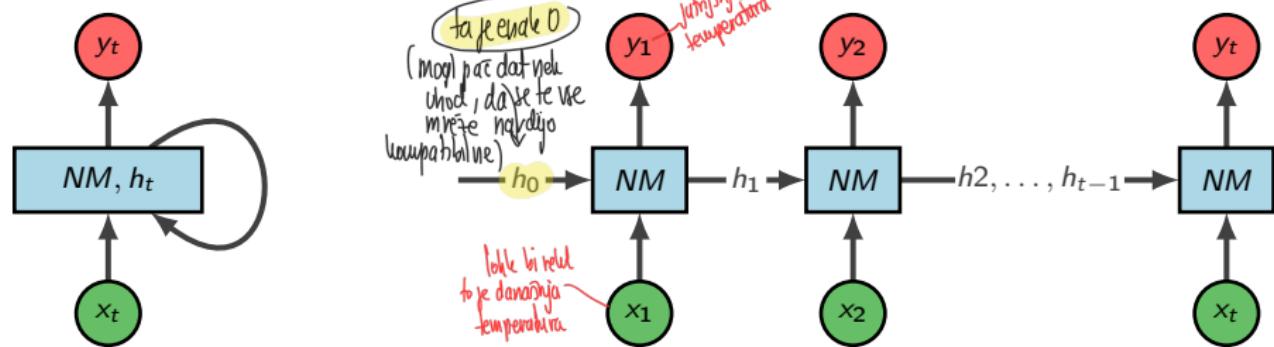
ZAPIS ČASOVNI INDEKS ... hocemo povezati da h_{t-1}
ni isti kot h_t (h_t je en časovni trenutek kasnej
kot h_{t-1})

Na prvi pogled ne reši težav, a poglejmo interpretacijo indeksa t .

Indeks pravzaprav pretvori RNM v usmerjeno NM

$$y_1 = x_2$$

Rekurenčna nevronska mreža = usmerjena nevronska mreža



$$h_t = w^T x_t + u^T h_{t-1}, \quad y_t = \phi(h_t), \quad h_0 = 0$$

"ODKROVANA NEVRONSKA MREŽA"

vidimo kakov ta NM je zato da na njej relacije

Plasti ustrezajo časovnemu indeksu: vzvratno razširjanje napake skozi čas.

Vloga stanja h v RNN

Stanje h v RNN predstavlja spomin

- Ker velja $h_t = f(x_t, h_{t-1})$
- Si lahko h_{t-1} **zapomni** prejšnje vhode x_1, x_2, \dots, x_{t-1}
- In zato obravnava x_t v kontekstu prejšnjih vhodov, t.j., h_{t-1}

Težave osnovne, *vanilla* RNN

VRN - vzvratno razširjene napake

Veliko število parametrov

- Ponavljanje NM povzroči naraščanje števila sinaps in uteži
- Rešitev: tako kot pri GNM, tudi tukaj si mreže NM delijo iste uteži

Kratkoročni spomin

- Vpliv x_1 na h_t skozi čas bledi, ker
- Na vsakem vmesnem koraku $2, 3, \dots, t$ uporabimo funkcijo f
- Rešitev: funkcije, ki ne zmanjšujejo vrednosti argumentov, npr. ReLU

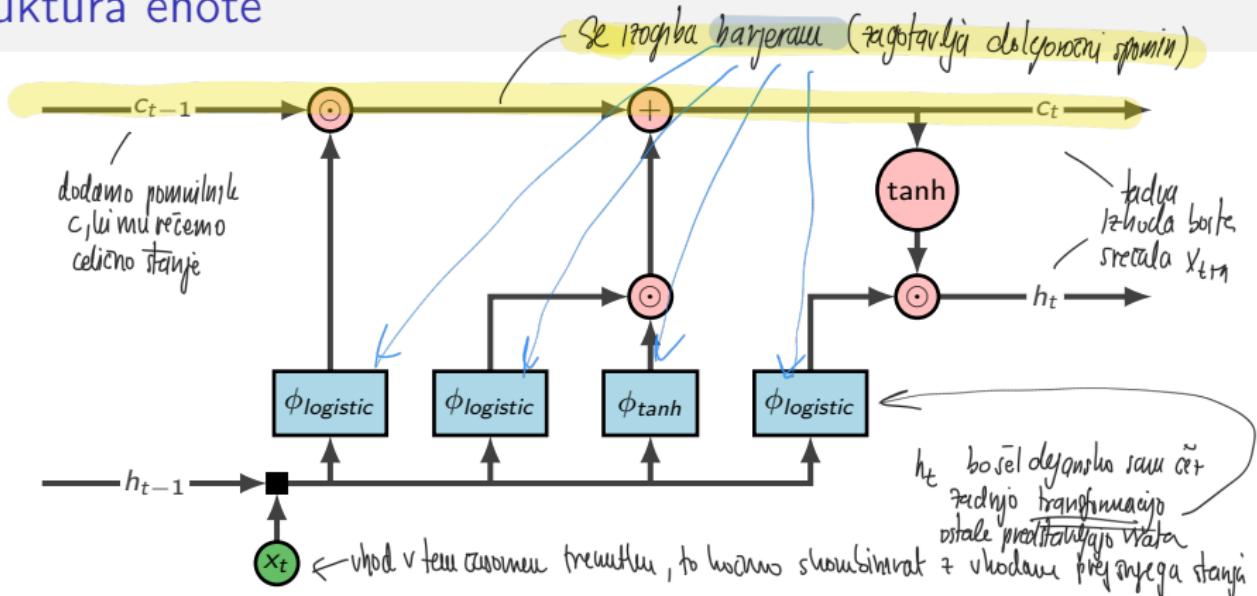
Krčenje gradientov zaradi vzvratnega razširjanja napake.

Rešitve druge in tretje težave

Arhitekturi NM, ki omogočata vzdrževanje dolgoročnega spomina

- LSTM, *Long Short-Term Memory*, dolgi kratkoročni spomin
- GRU, *Gated Recurrent Unit*, rekurenčna enota z vrtiti

Struktura enote



- $x_t \in \mathbb{R}^p$ je vhodni vektor, črna škatlica je operacija stika vektorjev
- $c_t, h_t \in \mathbb{R}^m$ sta vektorja **celičnega** in **skritega stanja**, $c_0 = h_0 = 0$
- Pravokotniki so enoplastne NM z označenimi funkcijami aktivacije
- Krogci roza barve so operacije nad komponentami vektorjev

Zakaj ima enota dva stanja c_t in h_t ?

Skrito stanje, h_t

- Je **podobno** skritemu stanju osnovne RNN, skrbi za **spomin**
- Shranjuje zgodovino vhodnih vektorjev x_t

Celično stanje, c_t

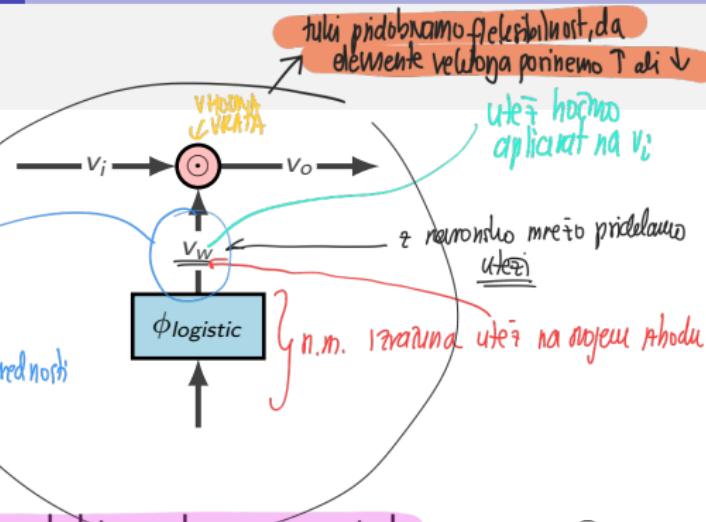
- Se **malo spreminja** znotraj enote, dve linearni operaciji
- Na ta način poskrbi za **dolgoročni spomin** RNN
- **Vrata** enote LSTM upravljajo ohranjanje informacije v tem stanju

Vrata, gate

kontrolirajo pretok informacij

tko ge 

možno da te
uteži igrajo vlogo
vlogo pri prehodu
↳ te uteži so jasne vrednosti
med 0 in 1



Hadamardov produkt po komponentah $v_o = v_w \odot v_i$

$$(0.1, 0.9) \odot (10, 100) = (0.1 \cdot 10, 0.9 \cdot 100) = (1, 90)$$

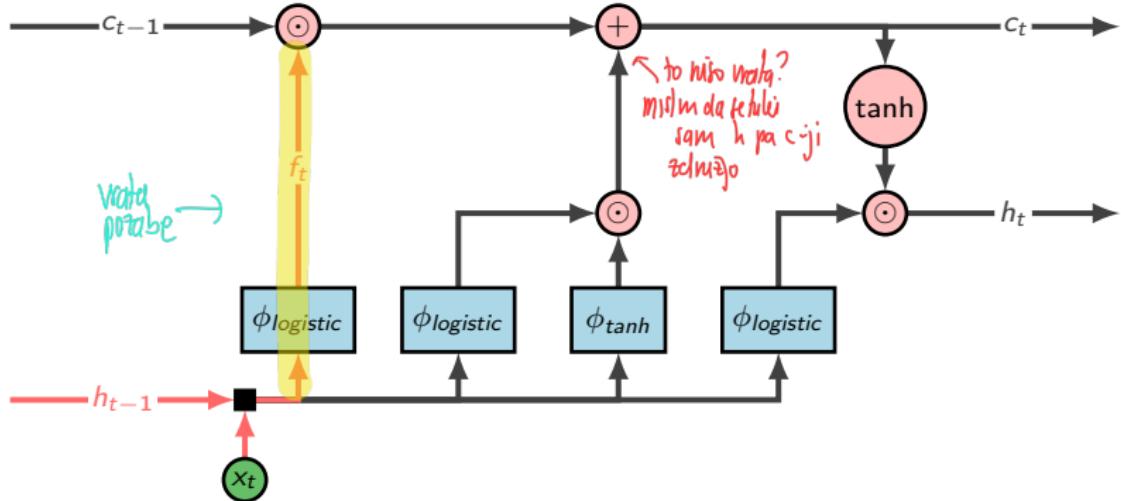
Vrata kontorolirajo pretok informacije iz vhoda v izhod

- $v_i, v_o \in \mathbb{R}^m$ sta vhodni in izhodni vektor, vektor $v_w \in (0, 1)^m$
- Logistična funkcija aktivacije poskrbi, da so komponente v_w med 0 in 1
- Komponenta v_w z vrednostjo blizu 0 **ne prepušča** ustrezne kom. v_i v v_o
- Komponenta v_w z vrednostjo blizu 1 **prepušča** ustrezno komp. v_i v v_o

Vrata za pozabite, forget gate

1. vrata, ki jih je vpeljal LSTM.

o mogocjoj centriranji od c-ja (to je ta dolgoročni spomin)
da negrejo vsi h v dolgoročni spomin

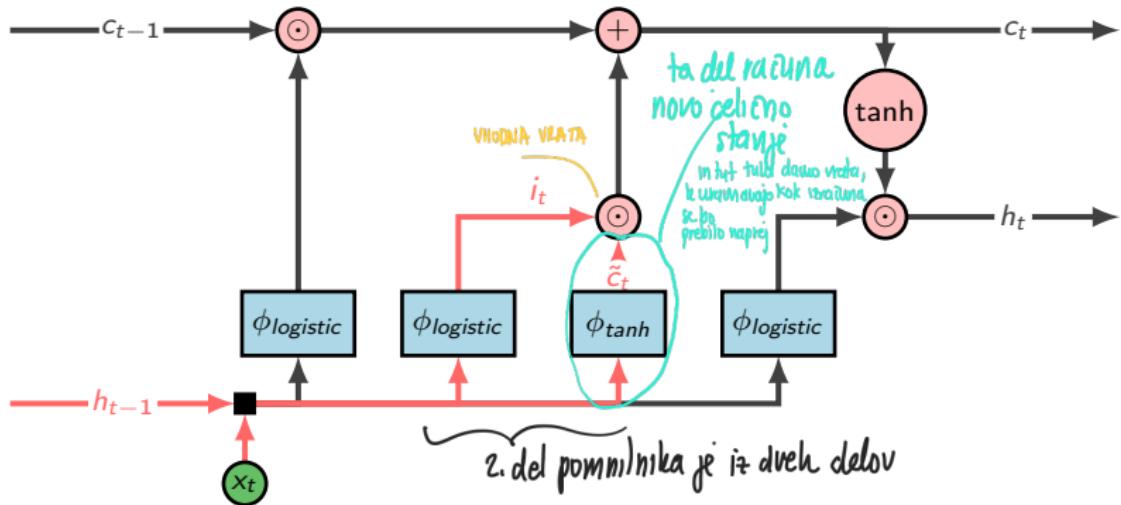


Kontrola pretoka informacije iz prejšnjega v trenutno celično stanje

$$f_t = \phi_{logistic}(W_f x_t + U_f h_{t-1} + w_{0f})$$

Matriki uteži $W_f \in \mathbb{R}^{m \times p}$ in $U_f \in \mathbb{R}^{m \times m}$, vektor pristransnosti $w_{0f} \in \mathbb{R}^m$.

Vhodna vrata, *input gate* in ustvarjanje nove informacije

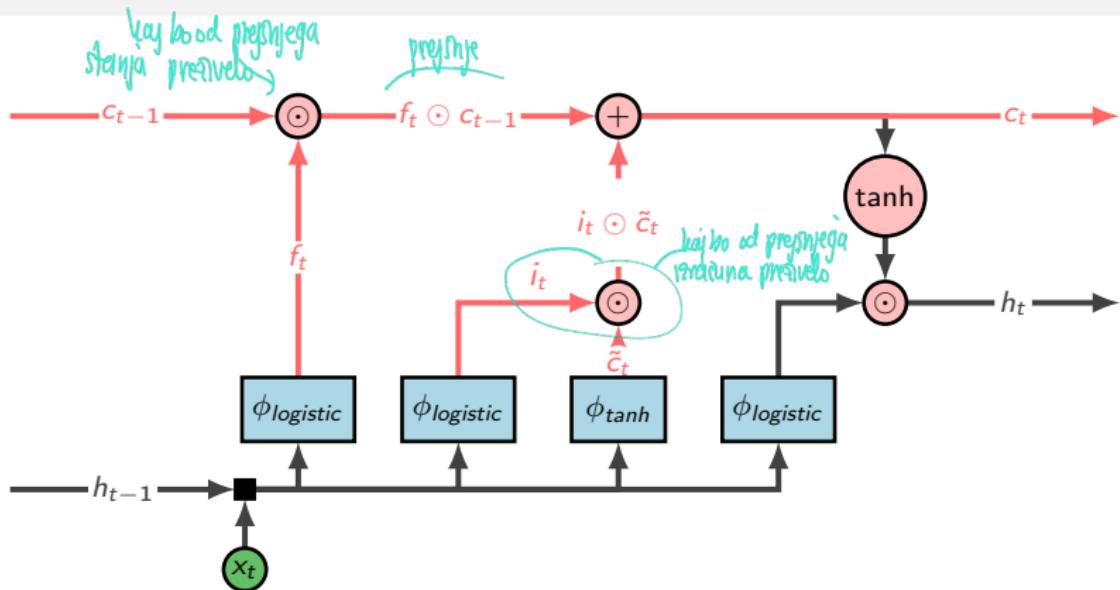


Ustvarjanje in kontorla pretoka nove informacije \tilde{c}_t za celično stanje

$$i_t = \phi_{logistic}(W_i x_t + U_i h_{t-1} + w_{0i}), \quad \tilde{c}_t = \phi_{tanh}(W_c x_t + U_c h_{t-1} + w_{0c})$$

Matrike uteži $W_i, W_c \in \mathbb{R}^{m \times p}$ in $U_i, U_c \in \mathbb{R}^{m \times m}$, vektorja $w_{0i}, w_{0c} \in \mathbb{R}^m$.

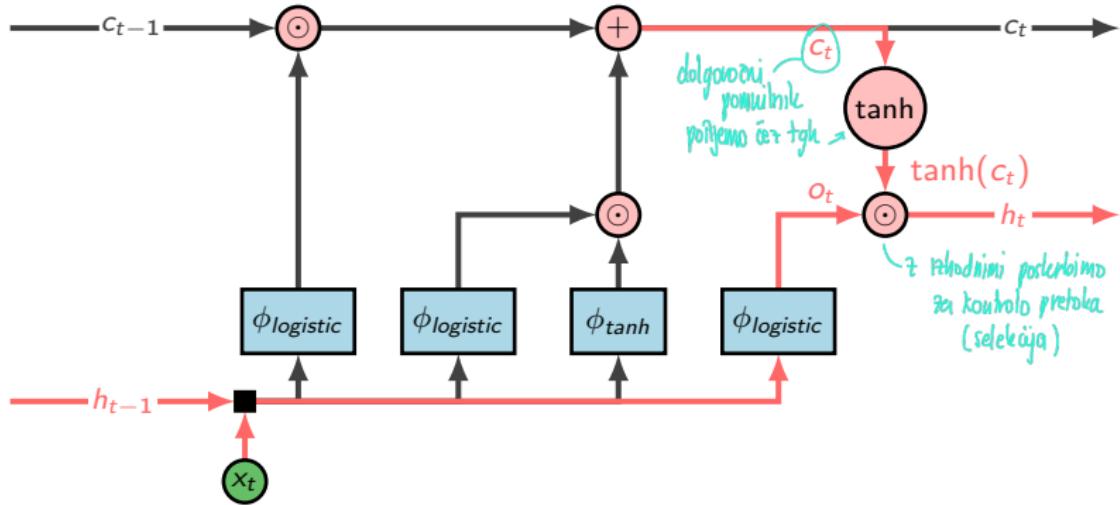
Ustvarjanje novega celičnega stanja



Kombiniranje pozabe in nove informacije v celično stanje enote c_t

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t$$

Izhodna vrata, exit gate in ustvarjanje novega skritega stanja



Ustvarjanje in kontorla pretoka nove informacije za skrito stanje enote h_t

$$o_t = \phi_{logistic}(W_o x_t + U_o h_{t-1} + w_{0o}), \quad h_t = o_t \odot \tanh(c_t)$$

Matriki uteži $W_o \in \mathbb{R}^{m \times p}$ in $U_o \in \mathbb{R}^{m \times m}$ ter vektor $w_{0o} \in \mathbb{R}^m$.

Delovanje enote

$$f_t = \phi_{logistic}(W_f x_t + U_f h_{t-1} + w_{0f})$$

$$i_t = \phi_{logistic}(W_i x_t + U_i h_{t-1} + w_{0i})$$

$$\tilde{c}_t = \phi_{tanh}(W_c x_t + U_c h_{t-1} + w_{0c})$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t$$

$$o_t = \phi_{logistic}(W_o x_t + U_o h_{t-1} + w_{0o})$$

$$h_t = o_t \odot \tanh(c_t)$$

Poimenovanje vektorjev

$f_t \in (0, 1)^m$ vektor aktivacije vrat pozabe

$i_t \in (0, 1)^m$ vektor aktivacije vhodnih vrat

$\tilde{c}_t \in (-1, 1)^m$ vektor aktivacije celičnega stanja

$c_t \in \mathbb{R}^m$ celično stanje enote

$o_t \in (0, 1)^m$ vektor aktivacije izhodnih vrat

$h_t \in (-1, 1)^m$ skrito stanje, tudi izhodni vektor enote

$x_t \in \mathbb{R}^p$ vhodni vektor enote

Povzetek delovanja enote

Celično stanje c_t kontrolira razmerje kratko- in dolgo- ročnega spomina

- Vrata pozabe skrbijo za kratkoročni spomin
- Vhodna vrata in ustvarjanje nove informacije \tilde{c}_t pa za dolgoročnega

Stanje h_t , ustvarjeno s kombinacijo

- Celičnega stanja, transformiranega s funkcijo tanh
- Izhodnih vrat, ki določijo pretok informacije iz celičnega v skrito stanje

Vrata hkrati omilijo učinek krčenja gradientov.

Poenostavitev enote LSTM

(Iz LSTM gremo na poenostavljenjo "arhitekturo")

Enota ima zgolj skrito stanje h_t

- Ukinja torej celično stanje
- Oziroma ga združi s skritim stanjem

// reset gate

Enota ima dvoje vrat, vrata ponastavljanja in posodabljanja

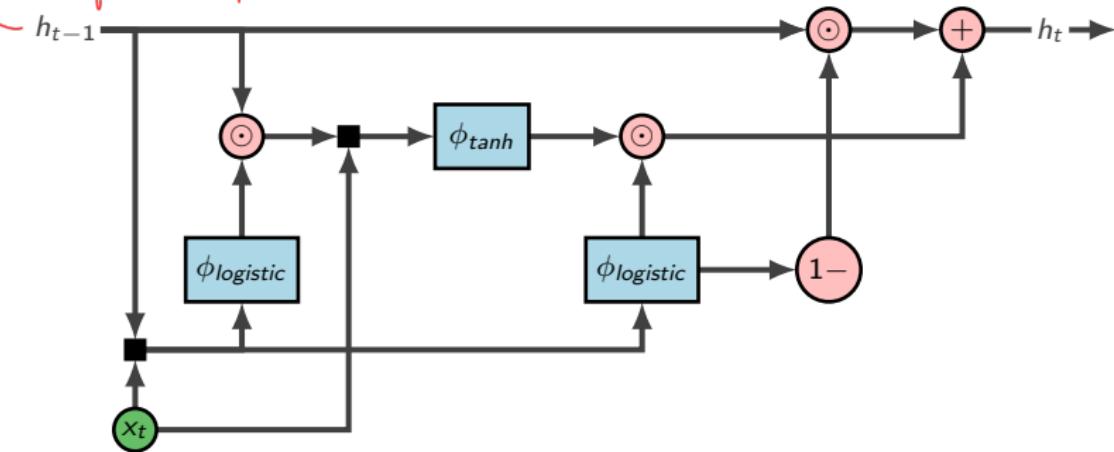
- **Vrata ponastavljanja** združijo funkcijo vhodnih vrat in vrat pozabe
- **Vrata posodabljanja** uravnavajo pretok informacije v stanje enote

\ update (slurki da te spomin nadgrajuje)

Struktura enote

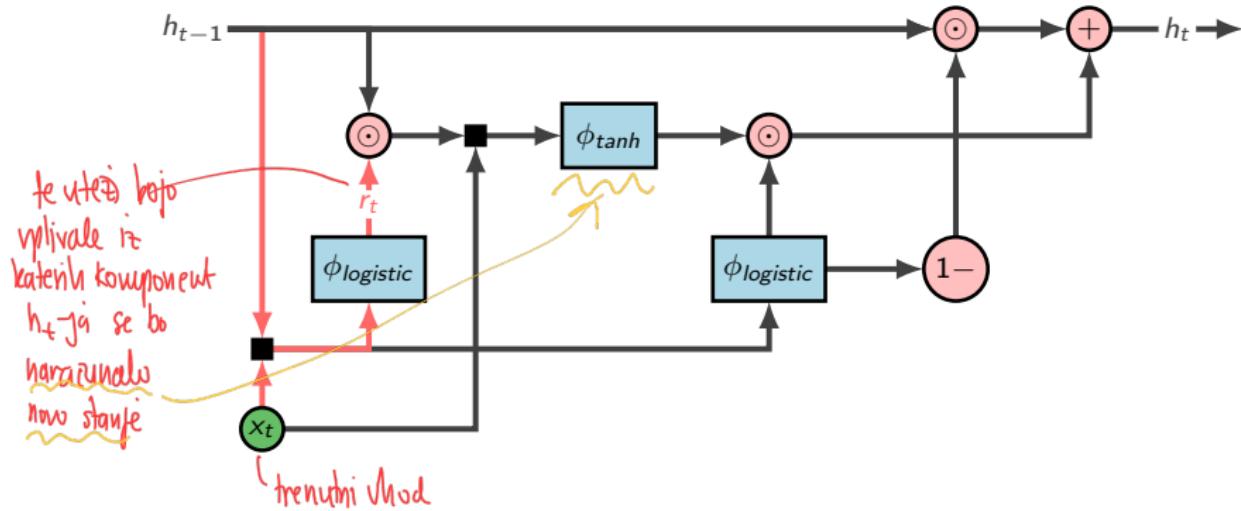
prejšnje stanje stanjuječi se na predela

prej so ble 4 modre skatice



- $x_t \in \mathbb{R}^p$ je vhodni vektor, črna kvadratka sta operaciji stika vektorjev
- $h_t \in \mathbb{R}^m$ je vektor **skritega stanja**, $h_0 = 0$
- Pravokotniki so enoplastne NM z označenimi funkcijami aktivacije
- Krogci roza barve so operacije nad komponentami vektorjev

Vrata ponastavljanja, *reset gate*

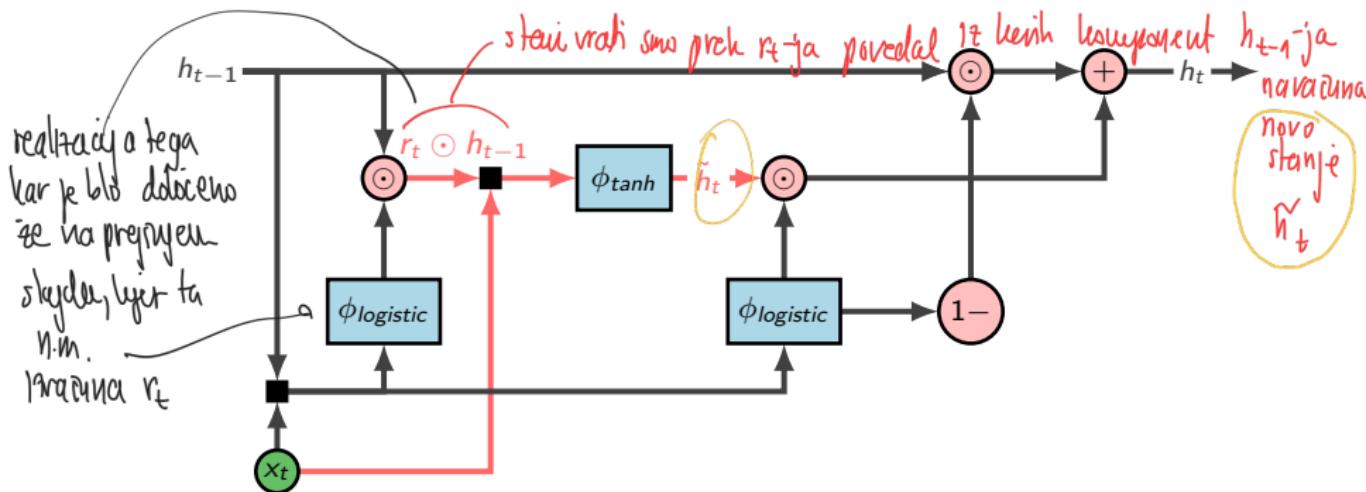


Kontrola pretoka informacije iz prejšnjega skritega stanja h_{t-1}

$$r_t = \phi_{logistic} (W_r x_t + U_r h_{t-1} + w_{0r})$$

Matriki uteži $W_r \in \mathbb{R}^{m \times p}$ in $U_r \in \mathbb{R}^{m \times m}$ ter vektor pristranskosti $w_{0r} \in \mathbb{R}^m$

Ustvarjanje nove informacije \tilde{h}_t za skrito stanje

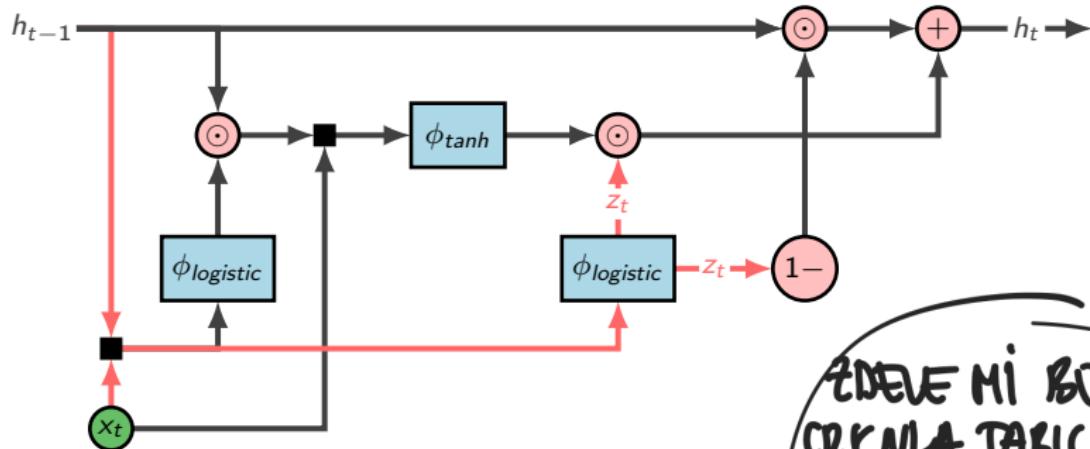


Nova informacija \tilde{h}_t za skrito stanje upošteva vektor uteži vrat r_t

$$\tilde{h}_t = \phi_{tanh}(W_h x_t + U_h(r_t \odot h_{t-1}) + w_{0h})$$

Matriki uteži $W_r \in \mathbb{R}^{m \times p}$ in $U_h \in \mathbb{R}^{m \times m}$ ter vektor $w_{0h} \in \mathbb{R}^m$.

Vrata posodabljanja, update gate

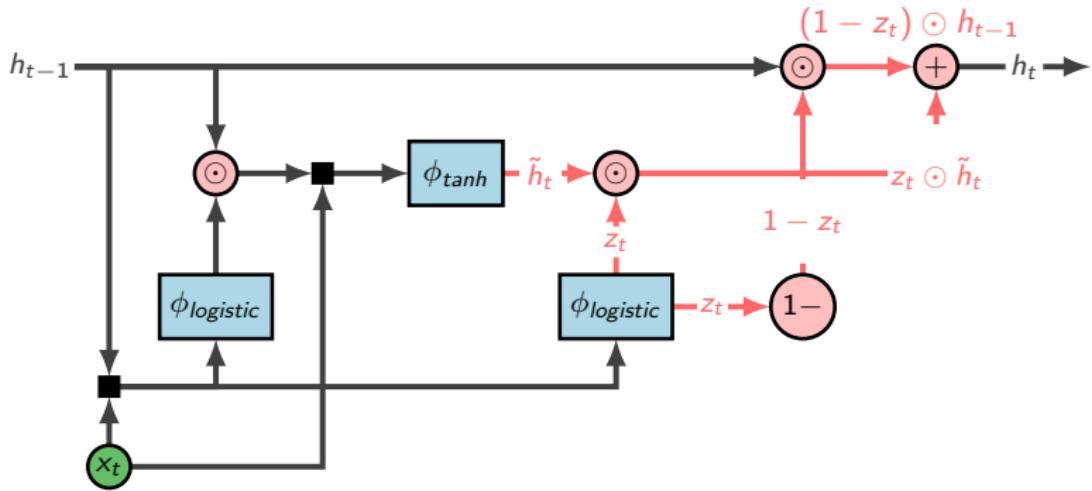


Kontrola pretoka nove informacije \tilde{h}_t v skrito stanje enote

$$z_t = \phi_{logistic} (W_z x_t + U_z h_{t-1} + w_{0z})$$

Matriki uteži $W_z \in \mathbb{R}^{m \times p}$ in $U_z \in \mathbb{R}^{m \times m}$ ter vektor $w_{0z} \in \mathbb{R}^m$.

Ustvarjanje skritega stanja enote h_t



Kombinira prejšnje stanje h_{t-1} in informacijo \tilde{h}_t v stanje enote h_t

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

Delovanje enote

$$\begin{aligned}r_t &= \phi_{\text{logistic}}(W_r x_t + U_r h_{t-1} + w_{0r}) \\ \tilde{h}_t &= \phi_{\tanh}(W_h x_t + U_h (r_t \odot h_{t-1}) + w_{0h})\end{aligned}$$

$$\begin{aligned}z_t &= \phi_{\text{logistic}}(W_z x_t + U_z h_{t-1} + w_{0z}) \\ h_t &= z_t \odot \tilde{h}_t + (1 - z_t) \odot h_{t-1}\end{aligned}$$

Poimenovanje vektorjev

$$r_t \in (0, 1)^m$$

vektor aktivacije vrat ponastavljanja

$$\tilde{h}_t \in (-1, 1)^m$$

vektor aktivacije kandidatne informacije za skrito stanje

$$z_t \in (0, 1)^m$$

vektor aktivacije vrat posodabljanja

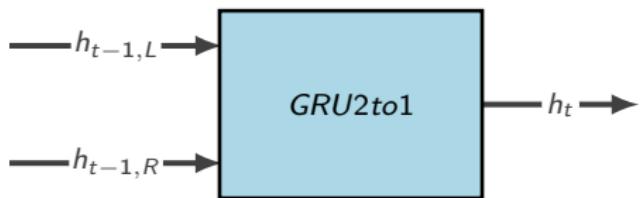
$$h_t \in (-1, 1)^m$$

skrito stanje, tudi izhodni vektor enote

$$x_t \in \mathbb{R}^p$$

vhodni vektor enote

Težava: dva prejšnja skrita stanja namesto enega



Vsako skrito stanje ustreza enemu nasledniku vozlišča

- Prejšnje skrito stanje levega naslednika $h_{t-1,L}$
- Prejšnje skrito stanje desnega naslednika $h_{t-1,R}$
- Rekurzivna uporaba osnovne enote, ki sledi drevesu, na tabli

Rešitev

Vektor prejšnjega stanja zamenjamo s stikom $h_L \| h_R$

$$r_t = \phi_{logistic}(W_r x_t + U_r(h_{t-1,L} \| h_{t-1,R}) + w_{0r})$$

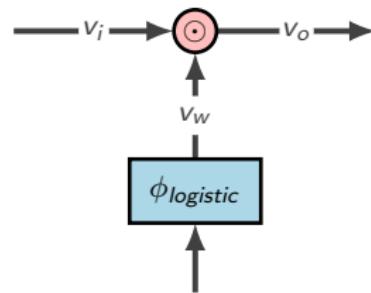
$$\tilde{h}_t = \phi_{tanh}(W_h x_t + U_h(r_t \odot (h_{t-1,L} \| h_{t-1,R})) + w_{0h})$$

$$z_t = \phi_{logistic}(W_z x_t + U_z(h_{t-1,L} \| h_{t-1,R}) + w_{0z})$$

Pri računanju stanja enote upoštevamo oba prejšnja skrita stanja

$$h_t = z_t \odot \tilde{h}_t + \frac{1 - z_t}{2} \odot h_{t-1,L} + \frac{1 - z_t}{2} \odot h_{t-1,R}$$

Poglejmo še enkrat delovanje vrat

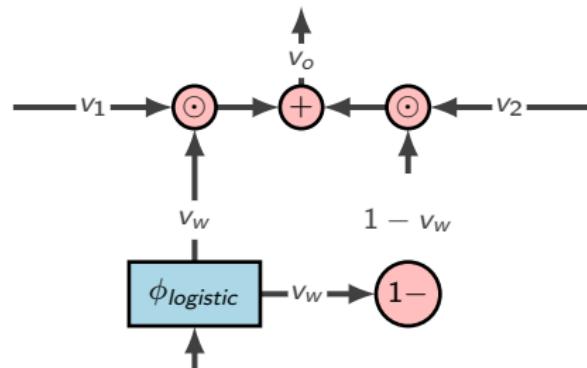


Hadamardov produkt po komponentah $v_o = v_w \odot v_i$

Vrata kontorolirajo pretok informacije iz vhoda v_i v izhod v_o

- Komponenta v_w z vrednostjo blizu 0 **ne prepušča** ustrezne kom. v_i v v_o
- Komponenta v_w z vrednostjo blizu 1 **prepušča** ustrezno komp. v_i v v_o

Vrata lahko uporabimo tudi za izbiro, GRU

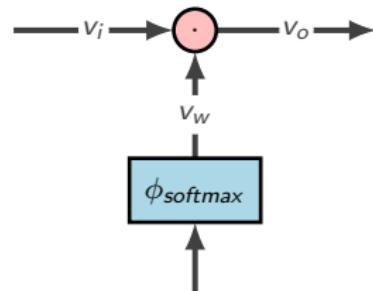


$$v_o = v_w \odot v_1 + (1 - v_w) \odot v_2$$

Vrata izbirajo med pretokom informacije iz v_1 ali v_2 v v_o

- Komponenta v_w z vrednostjo blizu 0 prepušča ustrezeno komp. v_2 v v_o
- Komponenta v_w z vrednostjo blizu 1 prepušča ustrezeno komp. v_1 v v_o

Posplošimo izbiro na poljubno število vhodov



$$\text{Skalarni produkt } v_o = v_w \cdot v_i$$

Vrata kontrolirajo pretok komponent vhoda v_i v izhod v_o

- Komponenta v_w z vrednostjo blizu 0 ne prepušča ustrezne kom. v_i v v_o
- Komponenta v_w z vrednostjo blizu 1 prepušča ustrezno komp. v_i v v_o
- **Mehanizem pozornosti:** usmerja pozornost na pomembne komp. vhoda

Reference in implementacije

Rekurenčne nevronske mreže

- (Hochreiter in Schmidhuber 1997): Dolgi kratkoročni spomin, LSTM
- (Cho in ost. 2014): Rekurenčna enota z vrati, GRU
- (Olah 2015): *Understanding LSTM networks*, odličen uvod

Posplošitvi

- (Mežnar in ost. 2023): Rekurenčna enota z vrati za dvojiška drevesa
- (Bahdanau in ost. 2014): Mehanizem pozornosti za prevajanje

Različice RNN so dostopne v standardnih knjižnicah za globoko učenje.