**SNE 10 (RNN)**

## 5. Uczenie głębokie (Deep learning) – Dalszy ciąg

### 5.3 Recurrent Neural Network (RNN)

Rekurencyjne sieci neuronowe

Static NN

$x = f(u), \;\; y = g(x)$

$$\begin{array}{c} y \\ \uparrow \\ x \\ \uparrow \\ u \end{array}$$

Dynamic NN (Neural automaton, Neural Turing machine) $\supset$ RNN

$x_t = f(x_{t-1}, u_t), \;\; y_t = g(x_t) \;\; (1 \leq t \leq T)$

$$\begin{array}{ccccccccccccc}
y_0 & & y_1 & & & & y_{t-1} & & y_t & & & & y_{T-1} & & y_T \\
\uparrow & & \uparrow & & & & \uparrow & & \uparrow & & & & \uparrow & & \uparrow \\
x_0 & \longrightarrow & x_1 & \longrightarrow & \cdots & \longrightarrow & x_{t-1} & \longrightarrow & x_t & \longrightarrow & \cdots & \longrightarrow & x_{T-1} & \longrightarrow & x_T \\
\uparrow & & \uparrow & & & & \uparrow & & \uparrow & & & & \uparrow & & \uparrow \\
u_0 & & u_1 & & & & u_{t-1} & & u_t & & & & u_{T-1} & & u_T \\
 & & & & & & & & & & & & & & \\
{\scriptstyle t=0} & & {\scriptstyle t=1} & & & & {\scriptstyle t-1} & & {\scriptstyle t} & & & & {\scriptstyle t=T-1} & & {\scriptstyle t=T}
\end{array}$$

($u_t = F(x_t)$ Feedback, Sprzężenie zwrotne)

**Sformułowanie RNN**

Sygnały wejściowe (input)

Rozważamy $U$ jako zero-wa warstwa $X^{(0)}$.

$$U = X^{(0)} = \coprod_{t=0}^{T} U_t = \coprod_{t=0}^{T} X_t^{(0)}$$

$$U_t = X_t^{(0)} = \{u_t(1), \ldots, u_t(P)\} = \{x_t^{(0)}(1), \ldots, x_t^{(0)}(P)\}$$

$u_t(p) = x_t^{(0)}(p) = (u_{t;1}(p), \ldots, u_{t;i_0}(p), \ldots, u_{t;m}(p)) = (x_{t;1}^{(0)}(p), \ldots, x_{t;i_0}^{(0)}(p), \ldots, x_{t;n_0}^{(0)}) \in \{0,1\}^m \subset \mathbb{R}^m$

$(1 \leq p \leq P; n_0 = m)$

Sygnały wyjściowe (output)

Rozważamy $Y$ jako $(R+1)$-ta warstwa $X^{(R+1)}$.

$$Y = X^{(R+1)} = \coprod_{t=0}^{T} Y_t = \coprod_{t=0}^{T} X_t^{(R+1)}$$

$$Y_t = X_t^{(R+1)} = \{y_t(1), \ldots, y_t(P)\} = \{x_t^{(R+1)}(1), \ldots, x_t^{(R+1)}(P)\}$$

$y_t(p) = x_t^{(R+1)}(p) = (y_{t;1}(p), \ldots, y_{t;i_{R+1}}(p), \ldots, y_{t;\ell}(p)) = (x_{t;1}^{(R+1)}(p), \ldots, x_{t;i_{R+1}}^{(R+1)}(p), \ldots, x_{t;n_{R+1}}^{(R+1)}) \in \mathbb{R}^\ell$

$(1 \leq p \leq P; n_{R+1} = \ell)$

Sygnały nauczyciela

$$Z = \coprod_{t=0}^{T} Z_t$$
$$Z_t = \{z_t(1), \dots, z_t(P)\}$$
$$z_t(p) = (z_{t;1}(p), \dots, z_{t;i_{R+1}}(p), \dots, z_{t;\ell}(p)) \in \mathbb{R}^\ell$$

$(1 \le p \le P)$

Warstwy środkowe $X$

$$X = \coprod_{t=0}^{T} X_t$$

$X_t$ składa się z kilk warstw.

$$X_t = \coprod_{r=1}^{R} X_t^{(r)}$$
$$X_t^{(r)} = \{x_t^{(r)}(1), \dots, x_t^{(r)}(P)\}$$
$$x_t^{(r)}(p) = (x_{t;1}^{(r)}(p), \dots, x_{t;i_r}^{(r)}, \dots, x_{t;n_r}^{(r)}(p))$$

$(1 \le p \le P)$

Wówczas dla $t = 0$ przekształcenie

$$x_t^{(r)} \to x_t^{(r+1)}$$

w

$$x_t^{(0)} = u_t \quad \to \quad x_t^{(1)} \quad \to \quad \cdots \quad \to \quad x_t^{(r)} \quad \to \quad x_t^{(r+1)} \quad \to \quad \cdots \quad \to \quad x_t^{(R)} \quad \to \quad x_t^{(R+1)} = y_t$$

można wyrazić wzorem

$$x_{t;i_{r+1}}^{(r+1)}(p) = f(\sum_{i_r=1}^{n_r} w_{i_{r+1},i_r}^{(r)} x_{t;i_r}^{(r)}(p))$$

$(0 \le r \le R,\ 1 \le p \le P)$.

Natomiast dla $1 \le t \le T$ przekształcenie

$$\begin{array}{ccc} & & x_{t-1}^{(r+1)} \\ & & \downarrow \\ x_t^{(r)} & \to & x_t^{(r+1)} \end{array}$$

w

$$\begin{array}{ccccccccccccccc} & & x_{t-1}^{(1)} & & & & x_{t-1}^{(r)} & & x_{t-1}^{(r+1)} & & & & x_{t-1}^{(R)} & & \\ & & \downarrow & & & & \downarrow & & \downarrow & & & & \downarrow & & \\ x_t^{(0)} = u_t & \to & x_t^{(1)} & \to & \cdots & \to & x_t^{(r)} & \to & x_t^{(r+1)} & \to & \cdots & \to & x_t^{(R)} & \to & x_t^{(R+1)} = y_t \end{array}$$

można wyrazić wzorem

$$x_{t;i_{r+1}}^{(r+1)}(p) = f(\sum_{i_r=1}^{n_r} w_{i_{r+1},i_r}^{(r)} x_{t;i_r}^{(r)}(p) + \sum_{i'_{r+1}=1}^{n_{r+1}} w'^{(r+1)}_{i_{r+1},i'_{r+1}} x_{t-1;i'_{r+1}}^{(r+1)}(p)) \quad (i_{r+1} = 1, \dots, n_{r+1})$$

$(0 \le r \le R,\ 1 \le p \le P)$.

**Cel RNN**
Niech

$$E_t = \frac{1}{2}\sum_{p=1}^{P}\sum_{i=1}^{\ell}(y_{t;i}(p) - z_{t;i}(p))^2 \quad (0 \le t \le T).$$

Celem RNN jest minimalizcja błędu

$$E = E(w^{(r)}_{i_{r+1},i_r}; r = 0,\dots,R, i_r = 1,\dots,n_r, i_{r+1} = 1,\dots,n_{r+1};;$$

$$w'^{(r+1)}_{i_{r+1},i'_{r+1}}; r = 0,\dots,R-1, i_{r+1} = 1,\dots,n_{r+1}, i'_{r+1} = 1,\dots,n_{r+1}) := \sum_{t=0}^{T} E_t$$

za pomocą metody gradientu.

**(1a) Gradienty** $t = 0$
$(0 \le r \le R)$

$$\frac{\partial E_t}{\partial w^{(r)}_{i_{r+1},i_r}}$$

$$= \sum_{p=1}^{P}\sum_{i_{R+1}=1}^{n_{R+1}}\sum_{i_R=1}^{n_R}\sum_{i_{R-1}=1}^{n_{R-1}}\cdots\sum_{i_{r+2}=1}^{n_{r+2}} \frac{\partial E_t}{\partial x^{(R+1)}_{t;i_{R+1}}(p)} \frac{\partial x^{(R+1)}_{t;i_{R+1}}(p)}{\partial x^{(R)}_{t;i_R}(p)} \frac{\partial x^{(R)}_{t;i_R}(p)}{\partial x^{(R-1)}_{t;i_{R-1}}(p)} \cdots \frac{\partial x^{(r+3)}_{t;i_{r+3}}(p)}{\partial x^{(r+2)}_{t;i_{r+2}}(p)} \frac{\partial x^{(r+2)}_{t;i_{r+2}}(p)}{\partial x^{(r+1)}_{t;i_{r+1}}(p)} \frac{\partial x^{(r+1)}_{t;i_{r+1}}(p)}{\partial w^{(r)}_{i_{r+1},i_r}}.$$

Szczegółowe wyrazy powyżej możemy obliczyć następująco:

$$\frac{\partial E_t}{\partial x^{(R+1)}_{t;i_{R+1}}(p)} = \frac{\partial E_t}{\partial y_{t;i_{R+1}}(p)} = y_{t;i_{R+1}}(p) - z_{t;i_{R+1}}(p),$$

$$\frac{\partial x^{(\alpha+1)}_{t;i_{\alpha+1}}(p)}{\partial x^{(\alpha)}_{t;i_\alpha}(p)} = f'(\sum_{i_\beta=1}^{n_\alpha} w^{(\alpha)}_{i_{\alpha+1},i_\beta} x^{(\alpha)}_{t;i_\beta}(p)) w^{(\alpha)}_{i_{\alpha+1},i_\alpha} \quad (\alpha = r+1,\dots,R),$$

$$\frac{\partial x^{(r+1)}_{t;i_{r+1}}(p)}{\partial w^{(r)}_{i_{r+1},i_r}} = f'(\sum_{i_\alpha=1}^{n_r} w^{(r)}_{i_{r+1},i_\alpha} x^{(r)}_{t;i_\alpha}(p)) x^{(r)}_{t;i_r}(p).$$

$(t = 0)$

**(1b) Gradienty** $1 \le t \le T$
$(0 \le r \le R)$

$$\frac{\partial E_t}{\partial w^{(r)}_{i_{r+1},i_r}}$$

$$= \sum_{p=1}^{P}\sum_{i_{R+1}=1}^{n_{R+1}}\sum_{i_R=1}^{n_R}\sum_{i_{R-1}=1}^{n_{R-1}}\cdots\sum_{i_{r+2}=1}^{n_{r+2}} \frac{\partial E_t}{\partial x^{(R+1)}_{t;i_{R+1}}(p)} \frac{\partial x^{(R+1)}_{t;i_{R+1}}(p)}{\partial x^{(R)}_{t;i_R}(p)} \frac{\partial x^{(R)}_{t;i_R}(p)}{\partial x^{(R-1)}_{t;i_{R-1}}(p)} \cdots \frac{\partial x^{(r+3)}_{t;i_{r+3}}(p)}{\partial x^{(r+2)}_{t;i_{r+2}}(p)} \frac{\partial x^{(r+2)}_{t;i_{r+2}}(p)}{\partial x^{(r+1)}_{t;i_{r+1}}(p)} \frac{\partial x^{(r+1)}_{t;i_{r+1}}(p)}{\partial w^{(r)}_{i_{r+1},i_r}}.$$

Szczegółowe wyrazy powyżej możemy obliczyć następująco:

$$\frac{\partial E_t}{\partial x^{(R+1)}_{t;i_{R+1}}(p)} = \frac{\partial E_t}{\partial y_{t;i_{R+1}}(p)} = y_{t;i_{R+1}}(p) - z_{t;i_{R+1}}(p),$$

$$\frac{\partial x^{(\alpha+1)}_{t;i_{\alpha+1}}(p)}{\partial x^{(\alpha)}_{t;i_\alpha}(p)} = f'(\sum_{i_\beta=1}^{n_\alpha} w^{(\alpha)}_{i_{\alpha+1},i_\beta} x^{(\alpha)}_{t;i_\beta}(p) + \sum_{i'_{\alpha+1}=1}^{n_{\alpha+1}} w'^{(\alpha+1)}_{i_{\alpha+1},i'_{\alpha+1}} x^{(\alpha+1)}_{t-1;i'_{\alpha+1}}(p)))w^{(\alpha)}_{i_{\alpha+1},i_\alpha} \quad (\alpha = r+1,\ldots,R),$$

$$\frac{\partial x^{(r+1)}_{t;i_{r+1}}(p)}{\partial w^{(r)}_{i_{r+1},i_r}} = f'(\sum_{i_\alpha=1}^{n_r} w^{(r)}_{i_{r+1},i_\alpha} x^{(r)}_{t;i_\alpha}(p) + \sum_{i'_{r+1}=1}^{n_{r+1}} w'^{(r+1)}_{i_{r+1},i'_{r+1}} x^{(r+1)}_{t-1;i^r_{r+1}}(p)))x^{(r)}_{t;i_r}(p).$$

$(1 \le t \le T)$

**(2) Gradienty (recurrent part)** $1 \le t \le T$
$(0 \le r \le R-1)$

$$\frac{\partial E_t}{\partial w'^{(r+1)}_{i_{r+1},i'_{r+1}}}$$

$$= \sum_{p=1}^{P} \sum_{i_{R+1}=1}^{n_{R+1}} \sum_{i_R=1}^{n_R} \sum_{i_{R-1}=1}^{n_{R-1}} \cdots \sum_{i_{r+2}=1}^{n_{r+2}} \frac{\partial E_t}{\partial x^{(R+1)}_{t;i_{R+1}}(p)} \frac{\partial x^{(R+1)}_{t;i_{R+1}}(p)}{\partial x^{(R)}_{t;i_R}(p)} \frac{\partial x^{(R)}_{t;i_R}(p)}{\partial x^{(R-1)}_{t;i_{R-1}}(p)} \cdots \frac{\partial x^{(r+3)}_{t;i_{r+3}}(p)}{\partial x^{(r+2)}_{t;i_{r+2}}(p)} \frac{\partial x^{(r+2)}_{t;i_{r+2}}(p)}{\partial x^{(r+1)}_{t;i_{r+1}}(p)} \frac{\partial x^{(r+1)}_{t;i_{r+1}}(p)}{\partial w'^{(r+1)}_{t;i_{r+1},i'_{r+1}}}$$

$(1 \le t \le T)$
Szczegółowe wyrazy powyżej możemy obliczyć następująco:

$$\frac{\partial E_t}{\partial x^{(R+1)}_{t;i_{R+1}}(p)} = \frac{\partial E_t}{\partial y_{t;i_{R+1}}(p)} = y_{t;i_{R+1}}(p) - z_{t;i_{R+1}}(p),$$

$$\frac{\partial x^{(\alpha+1)}_{t;i_{\alpha+1}}(p)}{\partial x^{(\alpha)}_{t;i_\alpha}(p)} = f'(\sum_{i_\beta=1}^{n_\alpha} w^{(\alpha)}_{i_{\alpha+1},i_\beta} x^{(\alpha)}_{t;i_\beta}(p) + \sum_{i'_{\alpha+1}=1}^{n_{\alpha+1}} w'^{(\alpha+1)}_{i_{\alpha+1},i'_{\alpha+1}} x^{(\alpha+1)}_{t-1;i'_{\alpha+1}}(p)))w^{(\alpha)}_{i_{\alpha+1},i_\alpha} \quad (\alpha = r+1,\ldots,R),$$

$$\frac{\partial x^{(r+1)}_{t;i_{r+1}}(p)}{\partial w'^{(r+1)}_{i_{r+1},i'_{r+1}}} = f'(\sum_{i_r=1}^{n_r} w^{(r)}_{i_{r+1},i_r} x^{(r)}_{t;i_r}(p) + \sum_{j'_{r+1}=1}^{n_{r+1}} w'^{(r+1)}_{i_{r+1},j'_{r+1}} x^{(r+1)}_{t-1;j'_{r+1}}(p))x^{(r+1)}_{t-1;i'_{r+1}}(p) \quad (i_{r+1} = 1,\ldots,n_{r+1})$$

$(1 \le t \le T)$

Szkic algorytmu

$$E = E(w^{(r)}_{i_{r+1},i_r}; r = 0,\ldots,R, i_r = 1,\ldots,n_r, i_{r+1} = 1,\ldots,n_{r+1};;$$

$$w'^{(r+1)}_{i_{r+1},i'_{r+1}}; r = 0,\ldots,R-1, i_{r+1} = 1,\ldots,n_{r+1}, i'_{r+1} = 1,\ldots,n_{r+1}) \to \text{Minimum lokalne}$$

za pomocą metody gradientu wygląda następująco.
**Algorytm**

$$w^{(r)\mathrm{new}}_{i_{r+1},i_r} = w^{(r)\mathrm{old}}_{i_{r+1},i_r} - c\frac{\partial E}{\partial w^{(r)}_{i_{r+1},i_r}}(w^{(r)\mathrm{old}}_{i_{r+1},i_r}; r = 0,\ldots,R, i_r = 1,\ldots,n_r, i_{r+1} = 1,\ldots,n_{r+1};;$$

$$w'^{(r+1)\mathrm{old}}_{i_{r+1},i'_{r+1}}; r = 0,\ldots,R-1, i_{r+1} = 1,\ldots,n_{r+1}, i'_{r+1} = 1,\ldots,n_{r+1})$$

$$w'^{(r+1)\mathrm{new}}_{i_{r+1},i'_{r+1}} = w'^{(r+1)\mathrm{old}}_{i_{r+1},i'_{r+1}} - c\frac{\partial E}{\partial w^{(r)}_{i_{r+1},i_r}}(w^{(r)\mathrm{old}}_{i_{r+1},i_r}; r = 0,\ldots,R, i_r = 1,\ldots,n_r, i_{r+1} = 1,\ldots,n_{r+1};;$$

$w'^{(r+1)\text{old}}_{i_{r+1},i'_{r+1}}; r = 0, \ldots, R-1, i_{r+1} = 1, \ldots, n_{r+1}, i'_{r+1} = 1, \ldots, n_{r+1})$

**Dodatek.** NETtalk
Learning, Then Talking (*The New York Times*, 16.08.1988)
Terrence Joseph Sejnowski (1947–) computational neuroscience
Institute for Neural Computation, University of California at San Diego (UCSD)
PhD in physics from Princeton University in 1978 with John Hopfield
co-invented the Boltzmann machine with Geoffrey Hinton

NETtalk był znanym zastosowaniem propagacji wstecznej w dziedzinie przetwarzania języka naturalnego (NLP=Natural Language Processing). Mimo tego, że nie używa się z RNNu (RNNu nie było w 80tych latach!), myślę, że można rozważać NETtalk jako pionierski prototyp RNNu dla NLP.

$u \in \{0,1\}^m \mapsto x \in \mathbb{R}^n \mapsto y \in \{0,1\}^\ell$
$m = (26+3) \times 7 = 203, n = 80 \sim 120, \ell = 26$

Przykład
$u = $ __t**a**ble $\mapsto y = ([\text{ei}], \text{accent}, \ldots)$
$u = $ tabl**e**___ $\mapsto y = ([\text{non}], \text{no accent}, \ldots)$