

Deep learning

RNN & autres architectures

Vincent Lefieux



RNN

Compléments

En guise de
conclusion

Références

Plan

RNN

Compléments

En guise de
conclusion

Références

Plan

RNN

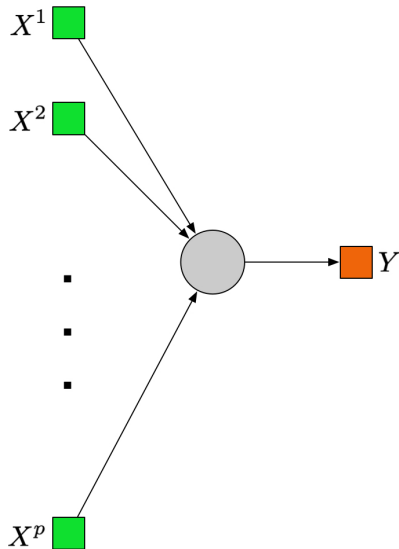
Compléments

En guise de
conclusion

Références

- ▶ Les **réseaux de neurones récurrents (RNN)** permettent de traiter des **séquences**, à savoir des :
 - ▶ données liées entre elles,
 - ▶ des données de taille variable.
- ▶ Les RNN sont particulièrement adaptés au traitement de **séries temporelles**, de **textes** et de **sons**.
- ▶ Les neurones d'un DNN ne conservent aucune mémoire, on va considérer des **neurones récurrents**.

Neurone classique



RNN

Compléments

En guise de
conclusion

Références

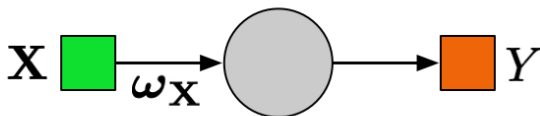
Neurone classique : représentation condensée

RNN

Compléments

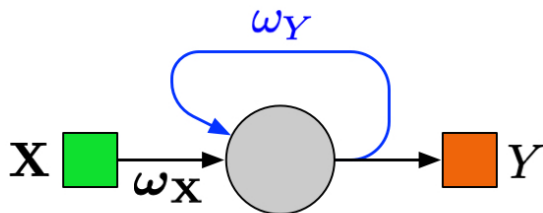
En guise de
conclusion

Références



$$Y = \varphi(\omega_0 + \omega_{\mathbf{X}}^{\top} \mathbf{X})$$

Neurone récurrent



$$Y_t = \varphi \left(\omega_{\mathbf{X}}^{\top} \mathbf{X}_t + \omega_Y Y_{t-1} + \omega_0 \right)$$

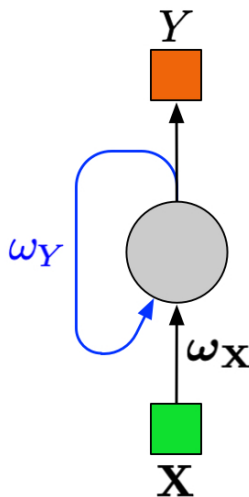
RNN

Compléments

En guise de
conclusion

Références

Neurone récurrent : représentation usuelle



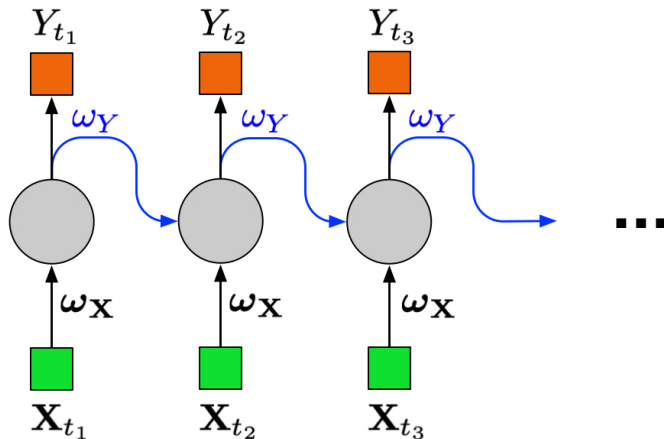
RNN

Compléments

En guise de
conclusion

Références

Neurone récurrent : version dépliée



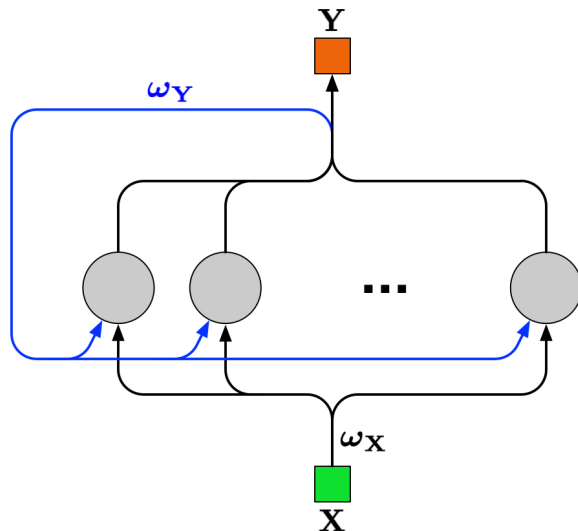
RNN

Compléments

En guise de
conclusion

Références

Unité récurrente



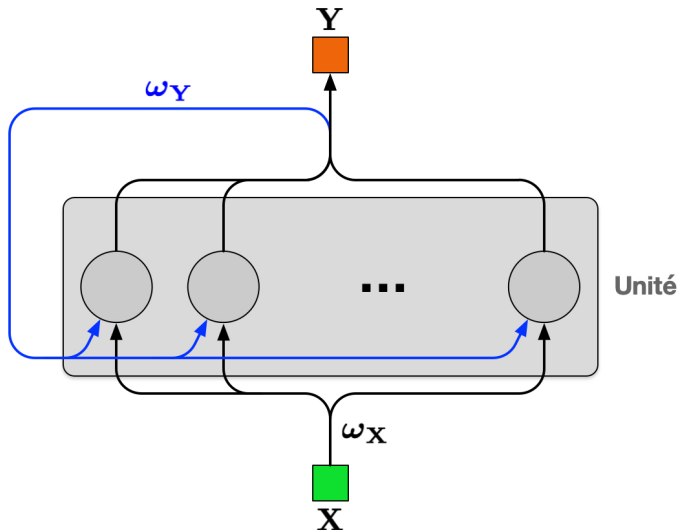
RNN

Compléments

En guise de
conclusion

Références

Unité récurrente



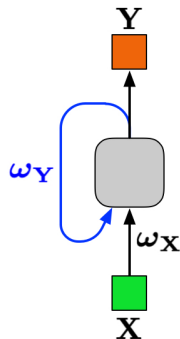
RNN

Compléments

En guise de
conclusion

Références

Unité récurrente III



$$\mathbf{Y}_t = \varphi \left(\omega_{\mathbf{X}}^{\top} \mathbf{X}_t + \omega_{\mathbf{Y}}^{\top} \mathbf{Y}_{t-1} + \omega_0 \right)$$

RNN

Compléments

En guise de
conclusion

Références

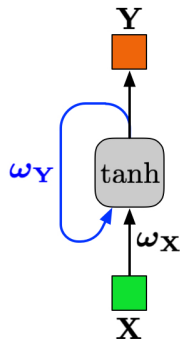
Unité récurrente IV

RNN

Compléments

En guise de
conclusion

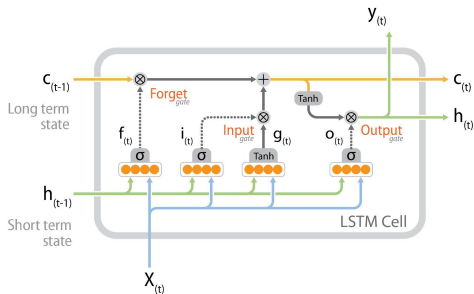
Références



$$\mathbf{Y}_t = \tanh (\omega_{\mathbf{X}}^{\top} \mathbf{X}_t + \omega_{\mathbf{Y}}^{\top} \mathbf{Y}_{t-1} + \omega_0)$$

- ▶ Convergence lente.
- ▶ Problème d'*exploding gradient* / *vanishing gradient*.
- ▶ Mémoire très faible.

LSTM



Long short-term memory (LSTM)¹

Gated recurrent unit (GRU)²

¹ Sepp Hochreiter, Jürgen Schmidhuber, (1997) [LSTM]

² Kyunghyun Cho et al, (2014) [GRU]

Source : (?)

RNN

Compléments

En guise de conclusion

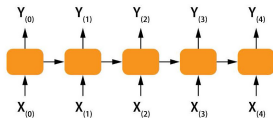
Références

$$\begin{aligned}
 f_{(t)} &= \sigma(W_{xf}^T X_{(t)} + W_{hf}^T h_{(t-1)} + b_f) \\
 i_{(t)} &= \sigma(W_{xi}^T X_{(t)} + W_{hi}^T h_{(t-1)} + b_i) \\
 g_{(t)} &= \tanh(W_{xg}^T X_{(t)} + W_{hg}^T h_{(t-1)} + b_g) \\
 o_{(t)} &= \sigma(W_{xo}^T X_{(t)} + W_{ho}^T h_{(t-1)} + b_o) \\
 c_{(t)} &= f_{(t)} \otimes c_{(t-1)} + i_{(t)} \otimes g_{(t)} \\
 y_{(t)} &= h_{(t)} = o_{(t)} \otimes \tanh(c_{(t)})
 \end{aligned}$$

with :

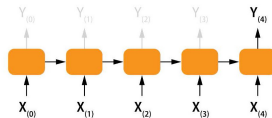
$X_{(t)} \in \mathbb{R}^d$	input vector
$f_{(t)} \in \mathbb{R}^h$	forget gate's activation vector
$i_{(t)} \in \mathbb{R}^h$	input gate's activation vector
$o_{(t)} \in \mathbb{R}^h$	output gate's activation vector
$g_{(t)} \in \mathbb{R}^h$	current entry vector
$h_{(t)}, y_{(t)} \in \mathbb{R}^h$	hidden state or output vector
$c_{(t)} \in \mathbb{R}^h$	cell state vector
\otimes	Hadamard product
σ	sigmoid function
W_k	weights matrix
b_k	bias vector

Applications potentielles



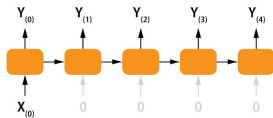
Serie to serie

Example : Time serie prediction



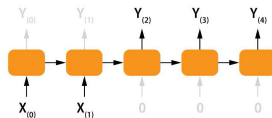
Serie to vector

Example : Sentiment analysis



Vector to serie

Example : Image annotation



Encoder-decoder

Example : Language Translation

Source : (?)

RNN

Compléments

En guise de
conclusion

Références

Motivation

- ▶ $\mathbf{X} \in \mathbb{R}^p$: Vecteur des entrées (*inputs*).
- ▶ $\mathbf{F} \in \mathbb{R}^h$: Vecteur d'activation de la porte d'oubli (*forget gate*).
- ▶ $\mathbf{I} \in \mathbb{R}^h$: Vecteur d'activation de la porte d'entrée (*input gate*).
- ▶ $\mathbf{O} \in \mathbb{R}^h$: Vecteur d'activation de la porte de sortie (*output gate*).
- ▶ $\mathbf{C} \in \mathbb{R}^h$: Vecteur d'état de l'unité (*cell state*).
- ▶ $\mathbf{H} \in \mathbb{R}^h$: Vecteur d'état caché (*Forget*).
- ▶ $\mathbf{Y} \in \mathbb{R}^h$: Vecteur des sorties (*ouputs*).

⊙ désigne le produit de Hadamard. σ désigne la fonction d'activation sigmoïde.

$$\mathbf{F}_t = \sigma \left(\omega_{\mathbf{F},\mathbf{C}}^\top \mathbf{C}_{t-1} + \omega_{\mathbf{F},\mathbf{X}}^\top \mathbf{X}_t + \omega_{\mathbf{F},0} \right) ,$$

$$\mathbf{I}_t = \sigma \left(\omega_{\mathbf{I},\mathbf{C}}^\top \mathbf{C}_{t-1} + \omega_{\mathbf{I},\mathbf{X}}^\top \mathbf{X}_t + \omega_{\mathbf{I},0} \right) ,$$

$$\mathbf{G}_t = \tanh \left(\omega_{\mathbf{G},\mathbf{C}}^\top \mathbf{C}_{t-1} + \omega_{\mathbf{G},\mathbf{X}}^\top \mathbf{X}_t + \omega_{\mathbf{G},0} \right) ,$$

RNN

Compléments

En guise de
conclusion

Références

Plan

RNN

Compléments

Autoencoders

GAN

En guise de
conclusion

Références

Plan

RNN

Compléments

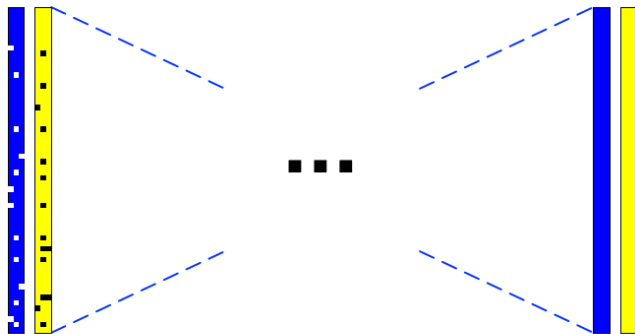
Autoencoders

GAN

En guise de
conclusion

Références

Principe



RNN

Compléments

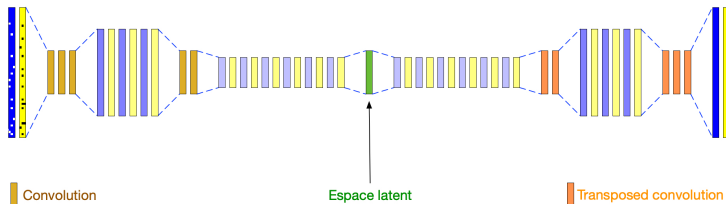
Autoencoders

GAN

En guise de
conclusion

Références

Architecture I



RNN

Compléments

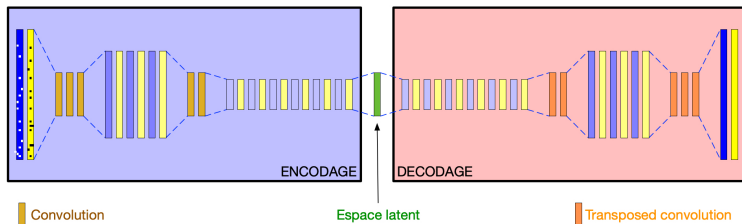
Autoencoders

GAN

En guise de
conclusion

Références

Architecture II



RNN

Compléments

Autoencoders

GAN

En guise de
conclusion

Références

Plan

RNN

Compléments

Autoencoders

GAN

En guise de
conclusion

Références

Quelques images I



RNN

Compléments

Autoencoders

GAN

En guise de
conclusion

Références

Quelques images II



RNN

Compléments

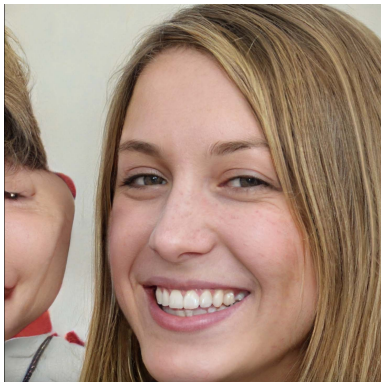
Autoencoders

GAN

En guise de
conclusion

Références

Quelques images III



RNN

Compléments

Autoencoders

GAN

En guise de
conclusion

Références

Quelques images



Source : <https://www.thispersondoesnotexist.com/>

RNN

Compléments

Autoencoders

GAN

En guise de
conclusion

Références

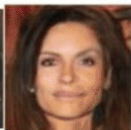
Quelques images



2014



2015



2016



2017



2018

RNN

Compléments

Autoencoders

GAN

En guise de
conclusion

Références

Generative Adversarial Networks

RNN

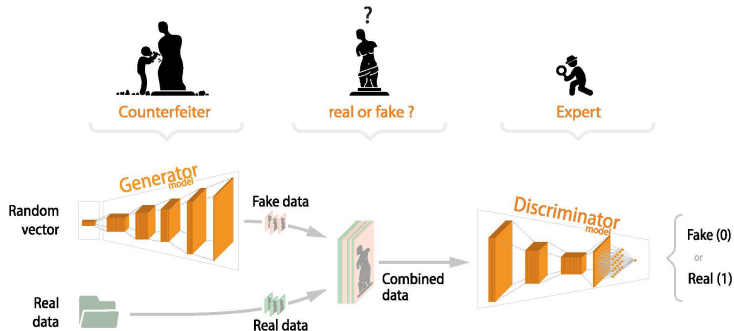
Compléments

Autoencoders

GAN

En guise de
conclusion

Références



Source : (?)

Plan

RNN

Compléments

**En guise de
conclusion**

Références

Pour terminer

RNN

Compléments

En guise de
conclusion

Références

- ▶ Un domaine en pleine effervescence. . .
- ▶ pas à l'abri d'un prochain « hiver »
- ▶ Des questions à ne pas oublier :
 - ▶ *Data drift*.
 - ▶ Droit et éthique.

Arias, S., E. Maldonado et J.-L. Parouty. 2022, «Fidle», URL <https://gricad-gitlab.univ-grenoble-alpes.fr/talks/fidle/-/wikis/home>.

Chollet, F. 2020, *L'apprentissage profond avec Python*, Machinelearning.fr.