Redes neurais para textos: rnn, lstm e cnn 1d

Prof. Dr. Ricardo Primi

Tópicos

- RNN
- LSTM
- CNN
- CNN 1d

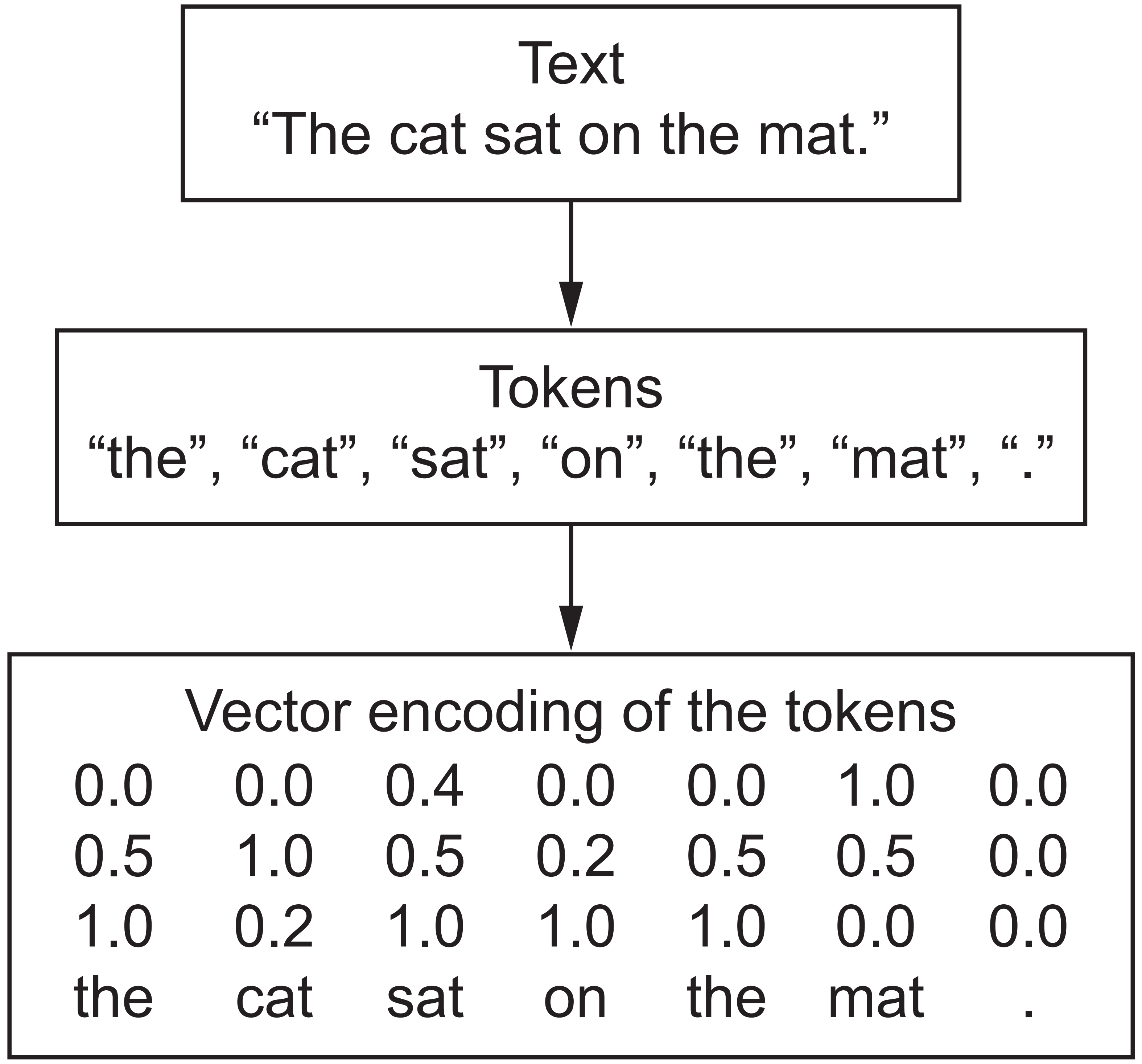


Figure 6.1 From text to tokens to vectors

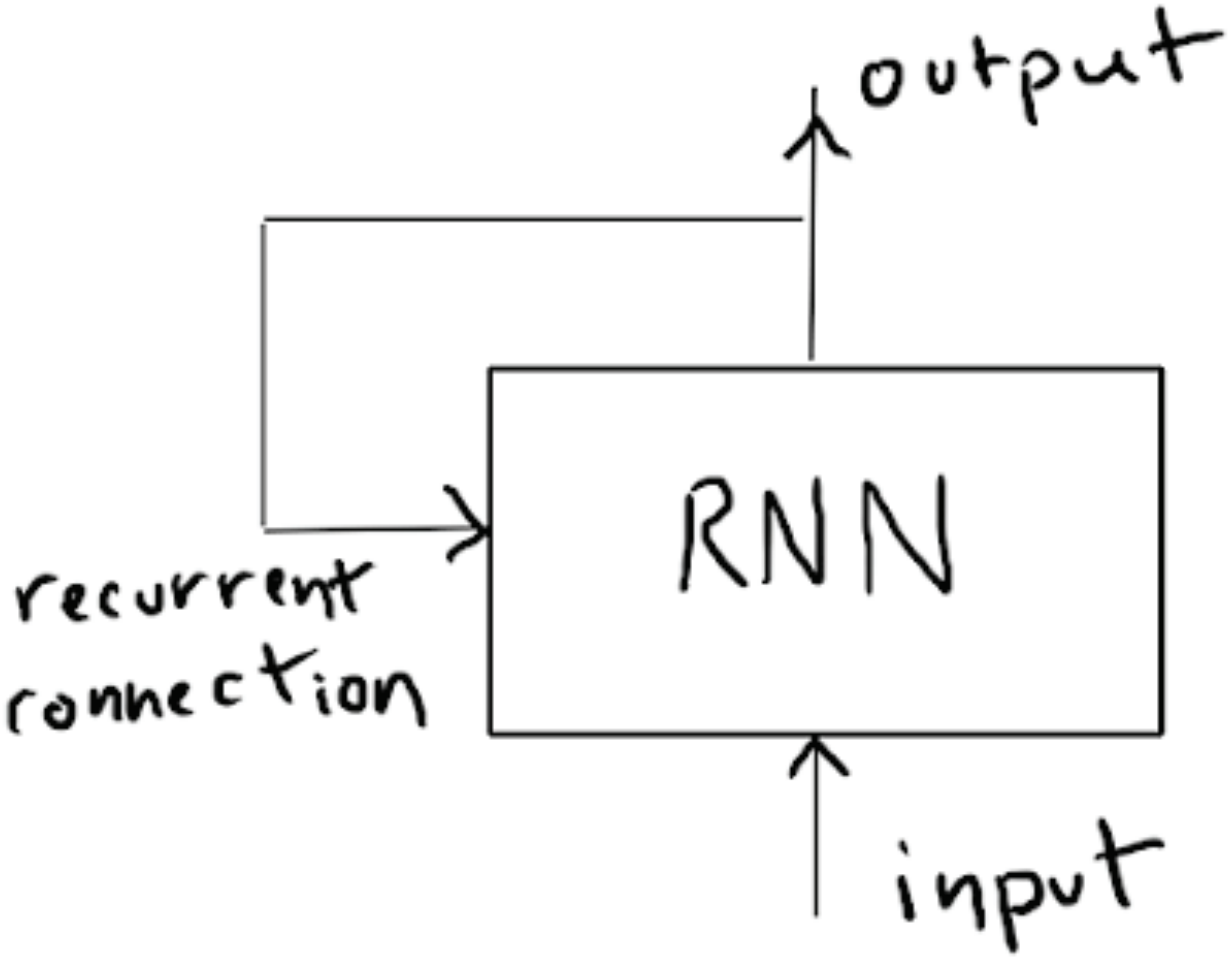
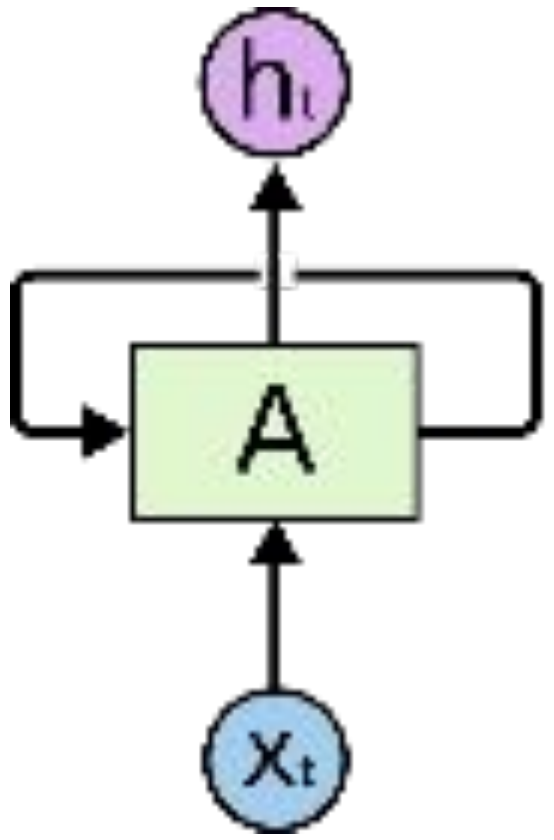
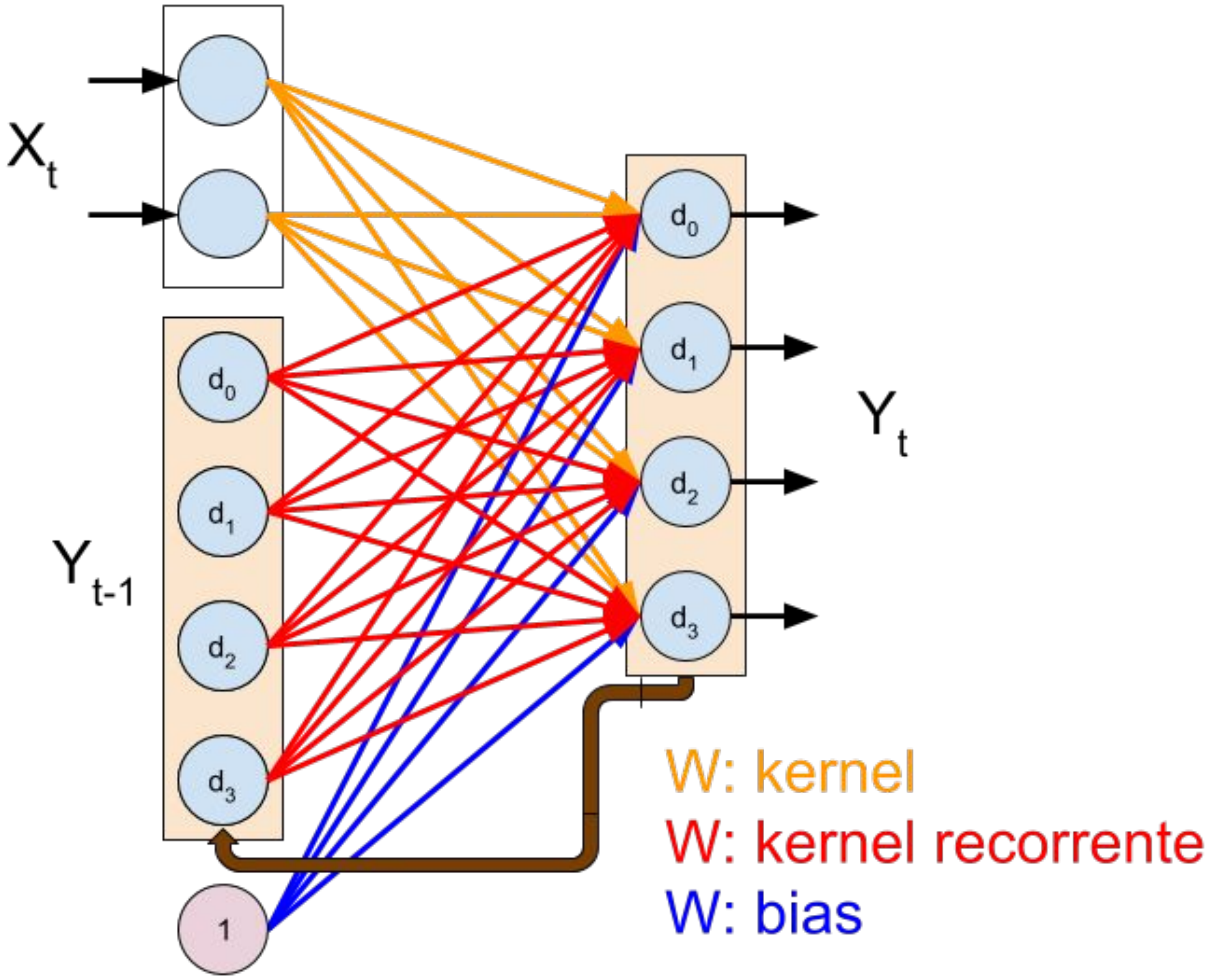


Figure 6.7 A recurrent network: a network with a loop

Redes Neurais Recorrentes





NeuralMind TECHNOLOGIES

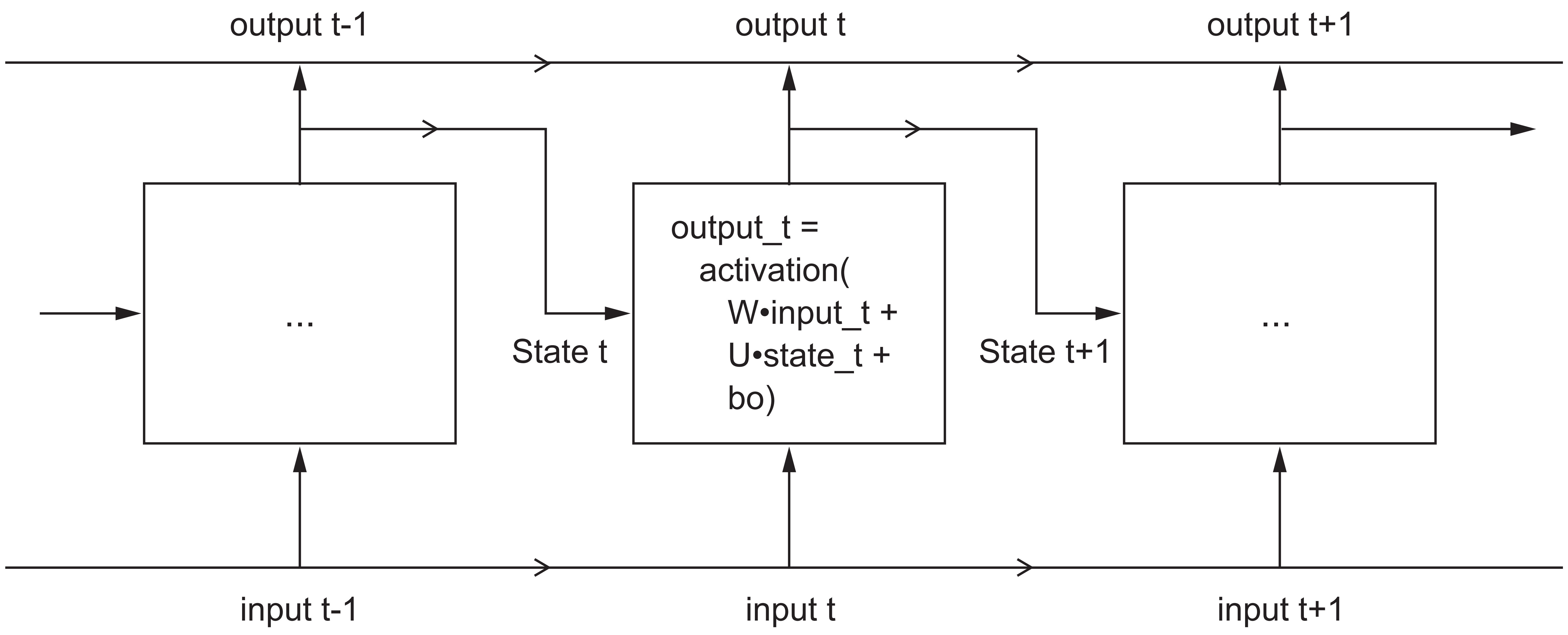


Figure 6.8 A simple RNN, unrolled over time

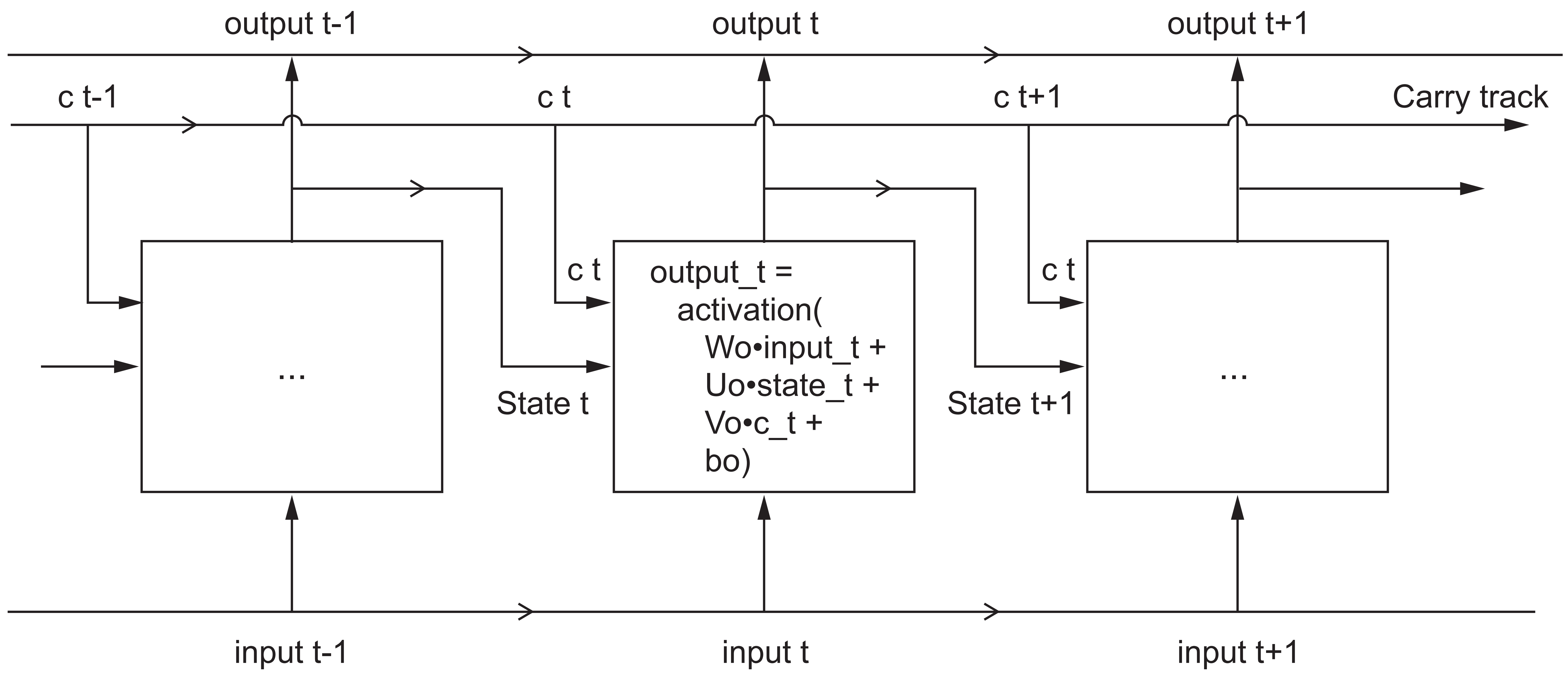
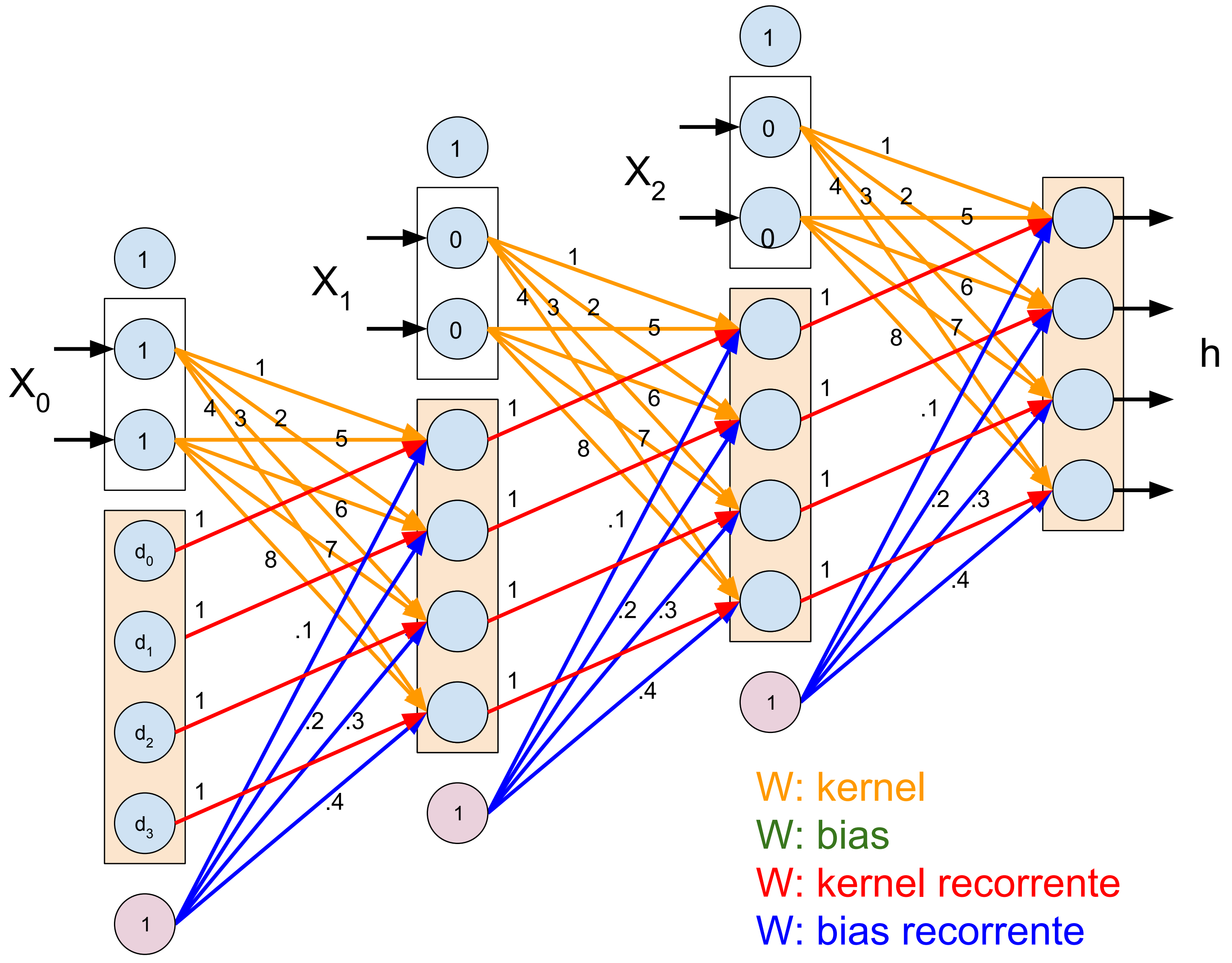
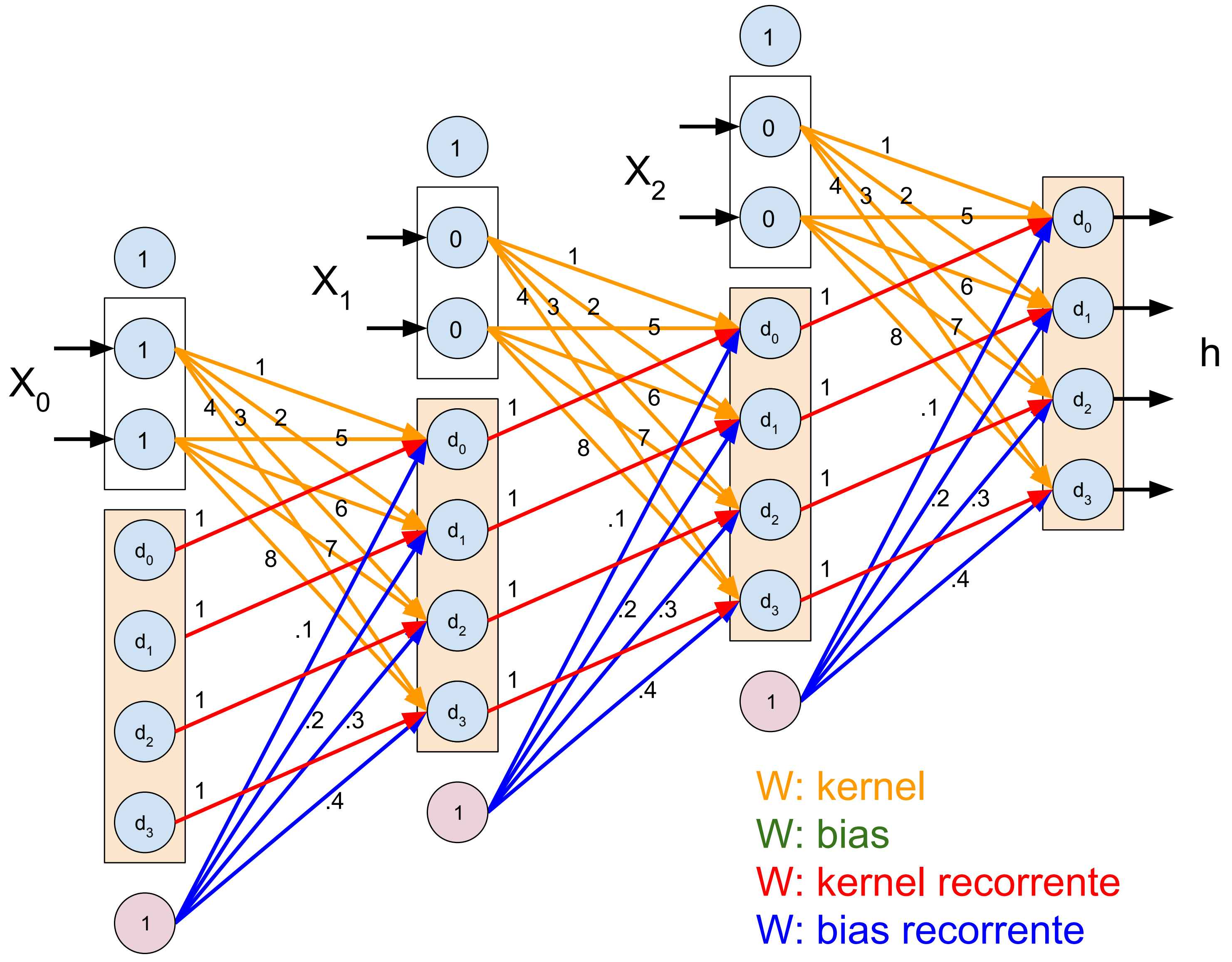
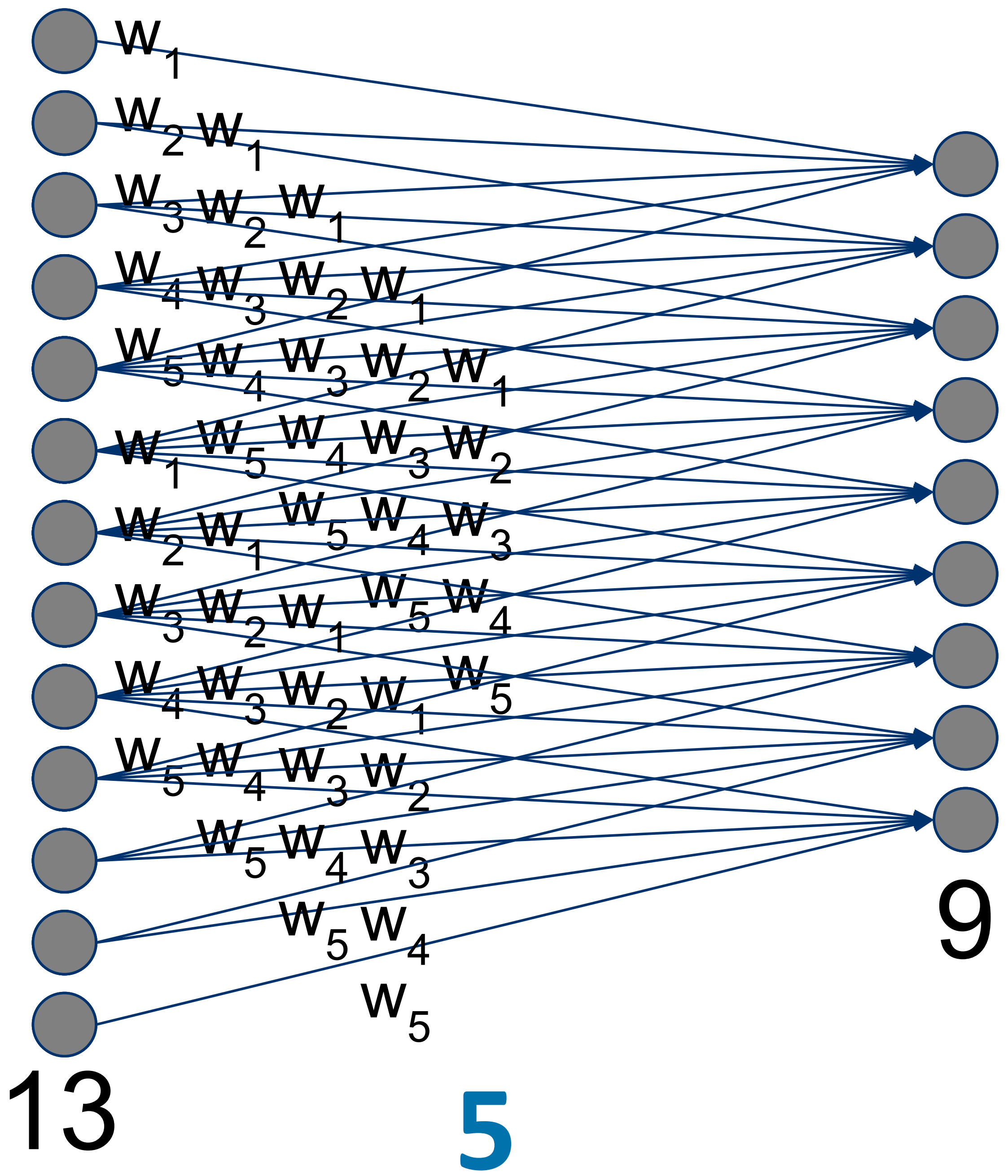


Figure 6.11 Going from a simple RNN to an LSTM: adding a carry track


X0
X1
X2
h
W: kernel
W: bias
W: kernel recorrente
W: bias recorrente


X0
X1
X2
h
W: kernel
W: bias
W: kernel recorrente
W: bias recorrente

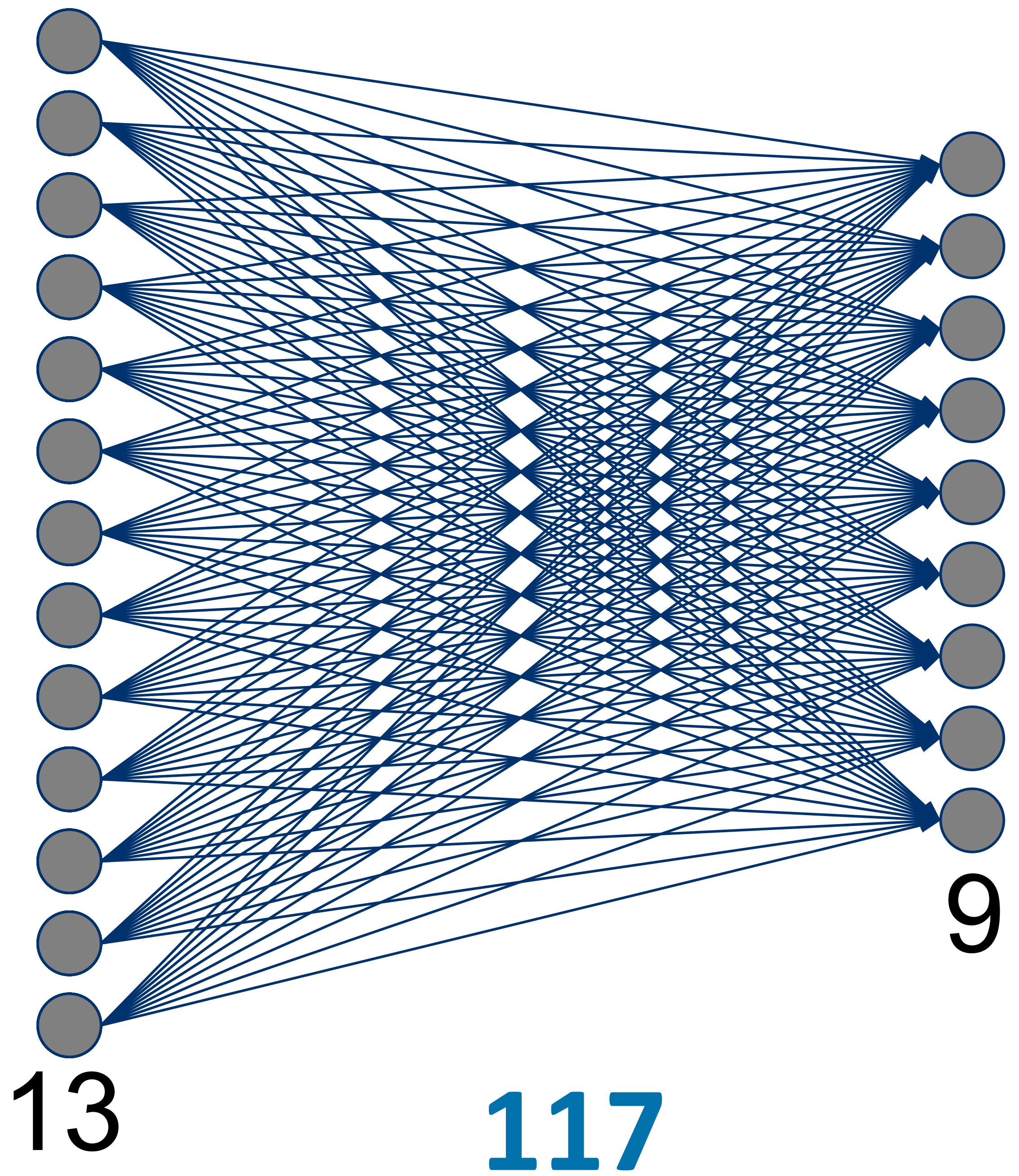
Rede Convolucional



5 parâmetros

Imagem
Vídeo
Som
Texto
Séries temporais

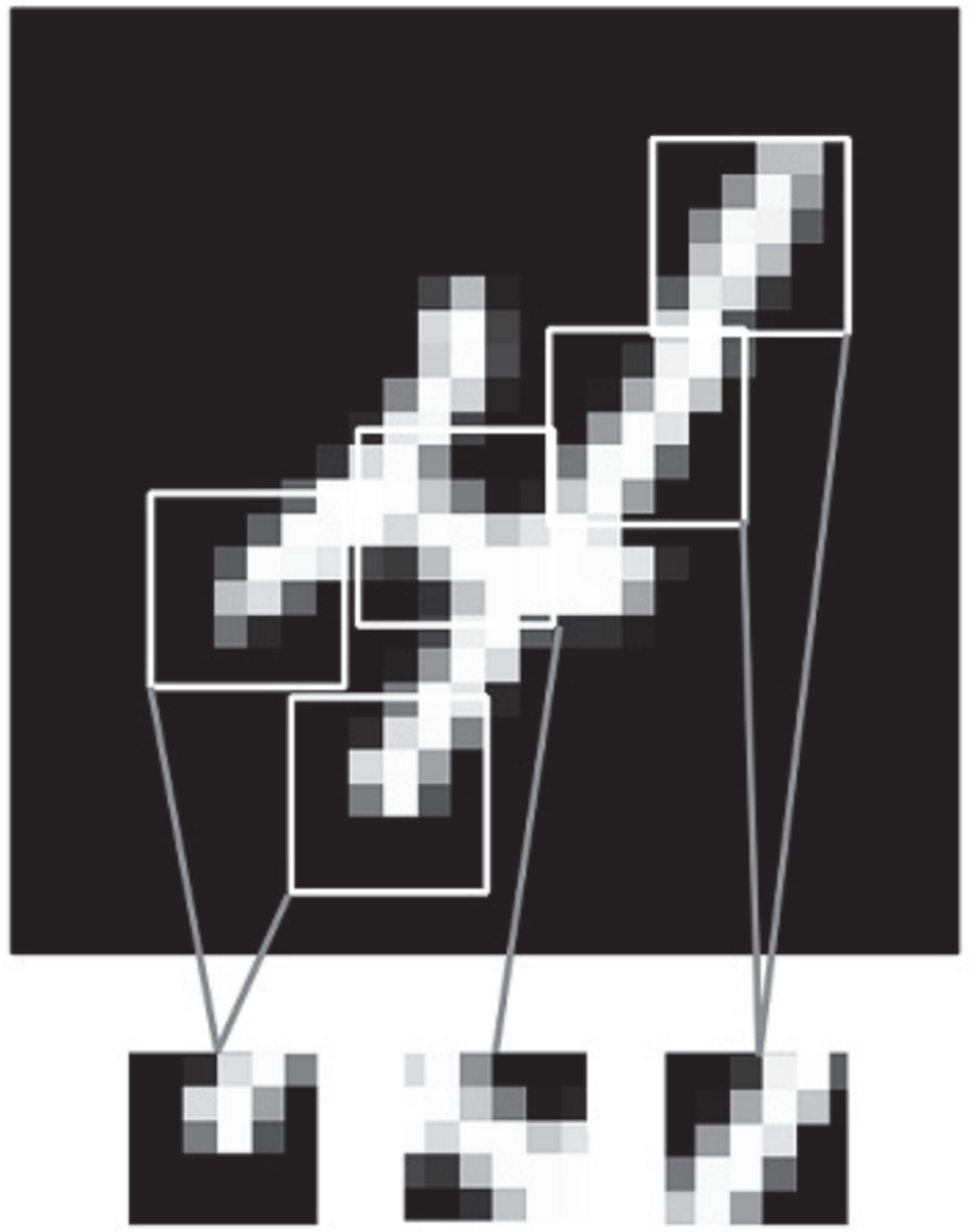
Rede Tradicional



117 parâmetros

CNN's

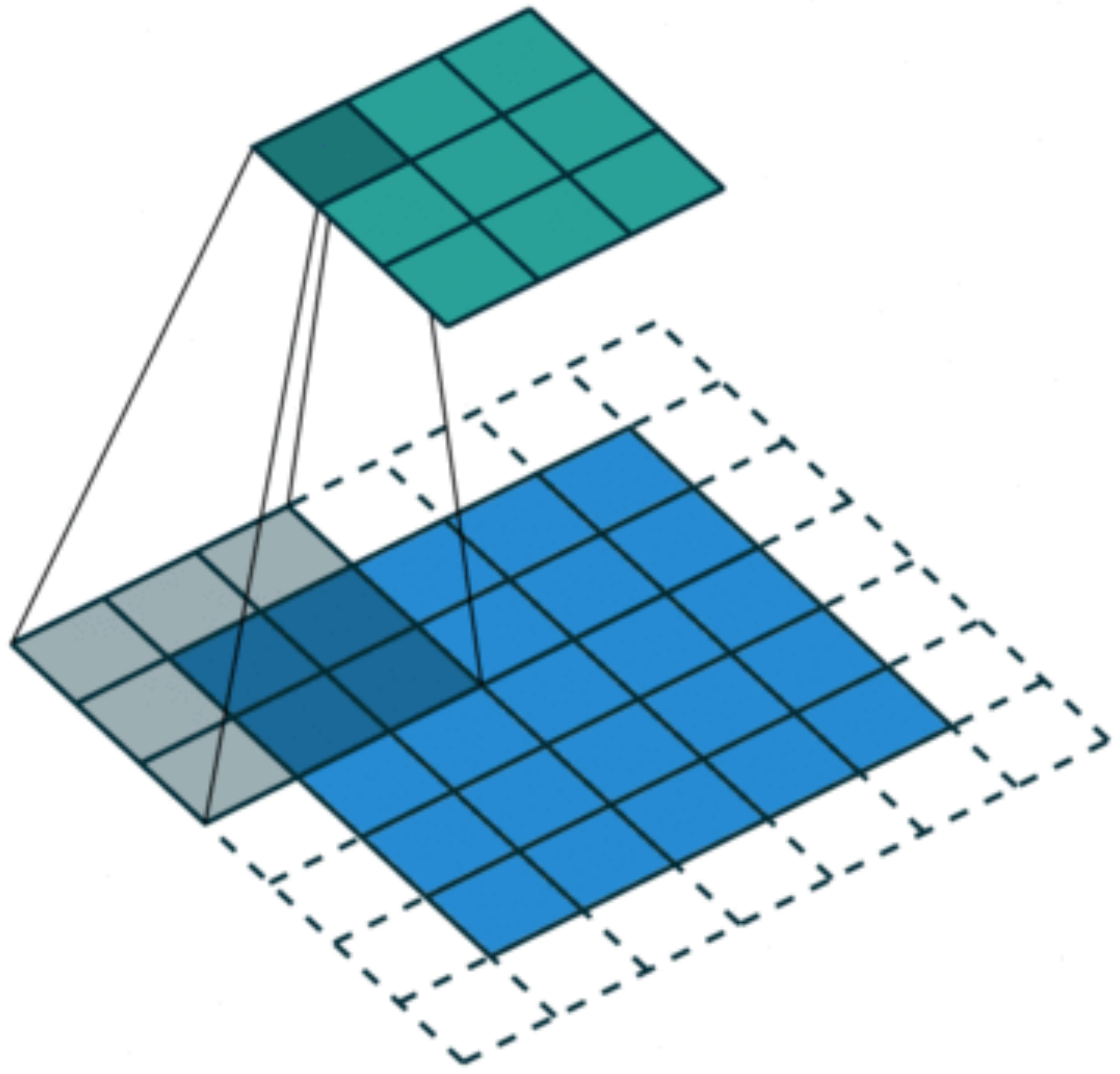
A principal diferença com relação à MLP é que a as camadas 'densas' aprendem padrões globais dos inputs, enquanto convoluções aprendem padrões **locais** dos inputs.



Fonte: Deep Learning with R

Uma convolução

- Definimos uma matriz de pesos (em cinza na representação ao lado)
- Andamos com essa matriz de pesos para cada parte da imagem (em azul ao lado).
- Esses pesos são multiplicados e depois somados para gerar uma nova 'imagem' (em verde).

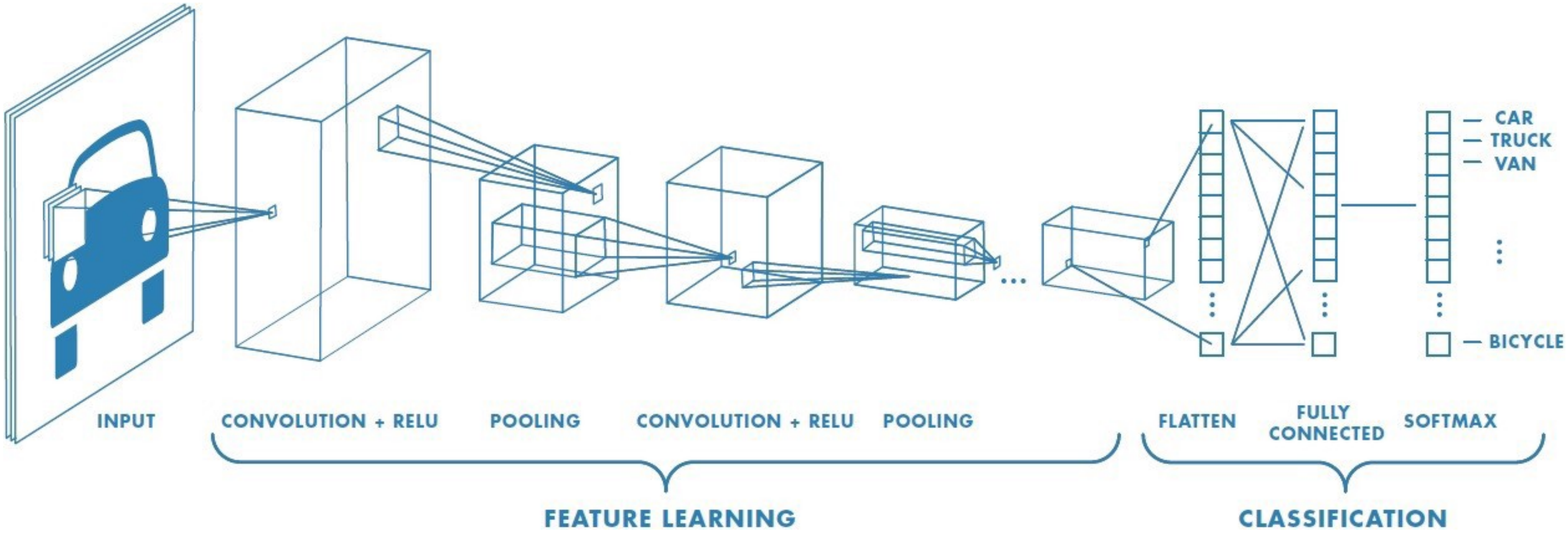


Fonte: Conv arithmetic

https://towardsdatascience.com/intuitively-understanding-convolutions-for-deep-learning-1f6f42faee1

Resumo:

- Mesclamos algumas camadas de convolução e max pooling, diminuindo a altura e largura das imagens e aumentando a profundidade.
- Depois transformamos em uma matriz e fazemos um modelo de classificação logístico usual.



Entendendo uma unidade CNN

Uma unidade é um filtro de imagem!

http://setosa.io/ev/image-kernels/

Mais vídeos:

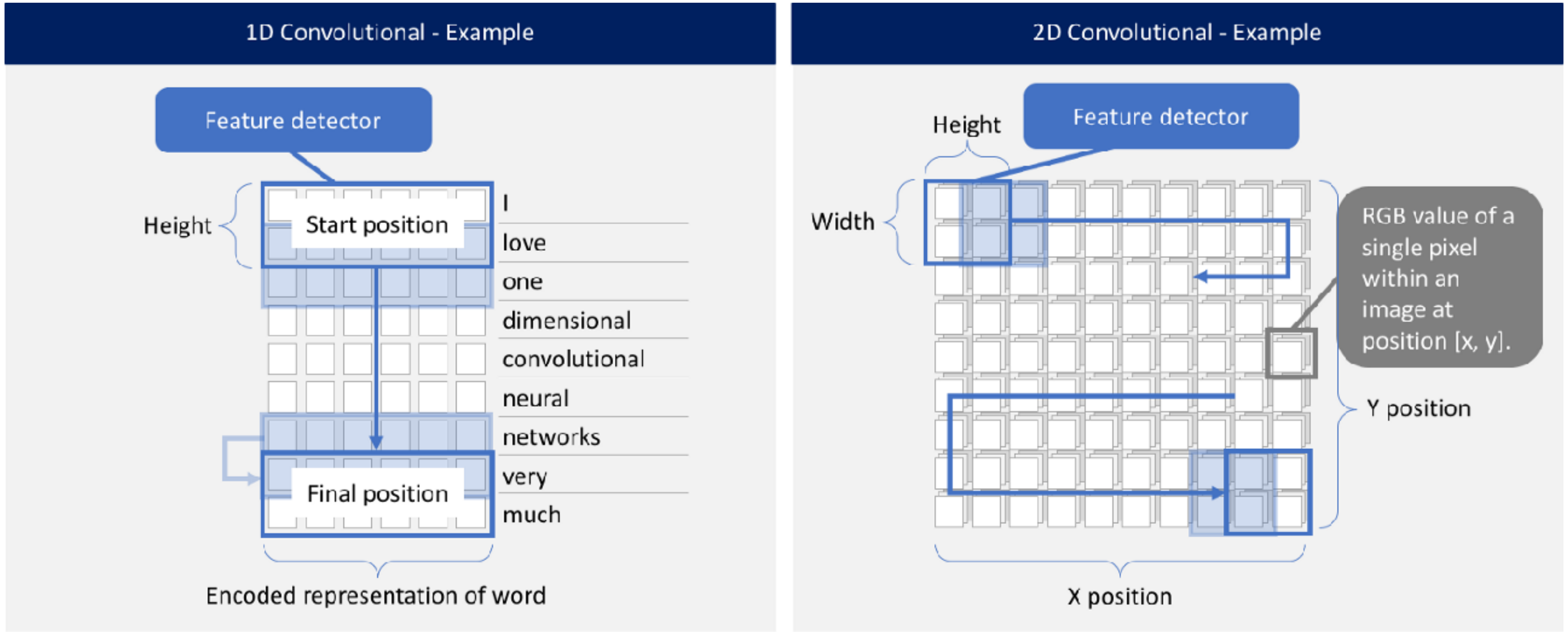
https://yosinski.com/deepvis

https://towardsdatascience.com/intuitively-understanding-convolutions-for-deep-learning-1f6f42faee1

https://playground.tensorflow.org

CNN 1d

Ao invés de uma matriz na entrada um vetor de embeddings



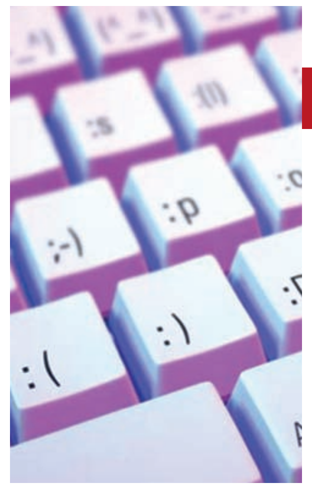
In this example for natural language processing, a sentence is made up of 9 words. Each word is a vector that represents a word as a low dimensional representation. The feature detector will always cover the whole word. The height determines how many words are considered when training the feature detector. In our example, the height is two. In this example the feature detector will iterate through the data 8 times.

In this example for computer vision, each pixel within the image is represented by its x- and y position as well as three values (RGB). The feature detector has a dimension of 2 x 2 in our example. The feature detector will now slide both horizontally and vertically across the image.

https://blog.goodaudience.com/introduction-to-1d-convolutional-neural-networks-in-keras-for-time-sequences-3a7ff801a2cf

CNN 1d

Avaliando B5 a partir de textos!!



Editor: **Erik Cambria,** Nanyang Technological University, Singapore, cambria@ntu.edu.sg

Deep Learning-Based Document Modeling for Personality Detection from Text

Navonil Majumder, *Instituto Politécnico Nacional*
Soujanya Poria, *Nanyang Technological University*
Alexander Gelbukh, *Instituto Politécnico Nacional*
Erik Cambria, *Nanyang Technological University*

Personality is a combination of an individual's behavior, emotion, motivation, and thought-pattern characteristics. Our personality has great impact on our lives; it affects our life choices, well-being, health, and numerous other preferences. Automatic detection of a person's personality traits has many important practical applications. In the context of sentiment analysis,[1] for example, the products and services recommended to a person should be those that have been positively evaluated by other users with a similar personality type. Personality detection can also be exploited for word polarity disambiguation in sentiment lexicons,[2] as the same concept can convey different polarity to different types of people. In mental health diagnosis, certain diagnoses correlate with certain personality traits. In forensics, knowing personality traits helps reduce the circle of suspects. In human resources management, personality traits affect one's suitability for certain jobs.

Personality is typically formally described in terms of the Big Five personality traits,[3] which are the following binary (yes/no) values:

- *Extroversion* (EXT). Is the person outgoing, talkative, and energetic versus reserved and solitary?
- *Neuroticism* (NEU). Is the person sensitive and nervous versus secure and confident?
- *Agreeableness* (AGR). Is the person trustworthy, straightforward, generous, and modest versus unreliable, complicated, meager, and boastful?
- *Conscientiousness* (CON). Is the person efficient and organized versus sloppy and careless?
- *Openness* (OPN). Is the person inventive and curious versus dogmatic and cautious?

Texts often reflect various aspects of the author's personality. In this article, we present a method to extract personality traits from stream-of-consciousness essays using a convolutional neural network (CNN). We trained five different networks, all with the same architecture, for the five personality traits (see the "Previous Work in Personality Detection" sidebar for more information). Each network was a binary classifier that predicted the corresponding trait to be positive or negative.

To this end, we developed a novel document-modeling technique based on a CNN features extractor. Namely, we fed sentences from the essays to convolution filters to obtain the sentence model in the form of n-gram feature vectors. We represented each individual essay by aggregating the vectors of its sentences. We concatenated the obtained vectors with the Mairesse features,[4] which were extracted from the texts directly at the preprocessing stage; this improved the method's performance. Discarding emotionally neutral input sentences from the essays further improved the results.

For final classification, we fed this document vector into a fully connected neural network with one hidden layer. Our results outperformed the current state of the art for all five traits. Our implementation is publicly available and can be downloaded freely for research purposes (see http://github.com/senticnet/personality-detection).

1541-1672/17/$33.00 © 2017 IEEE
Published by the IEEE Computer Society

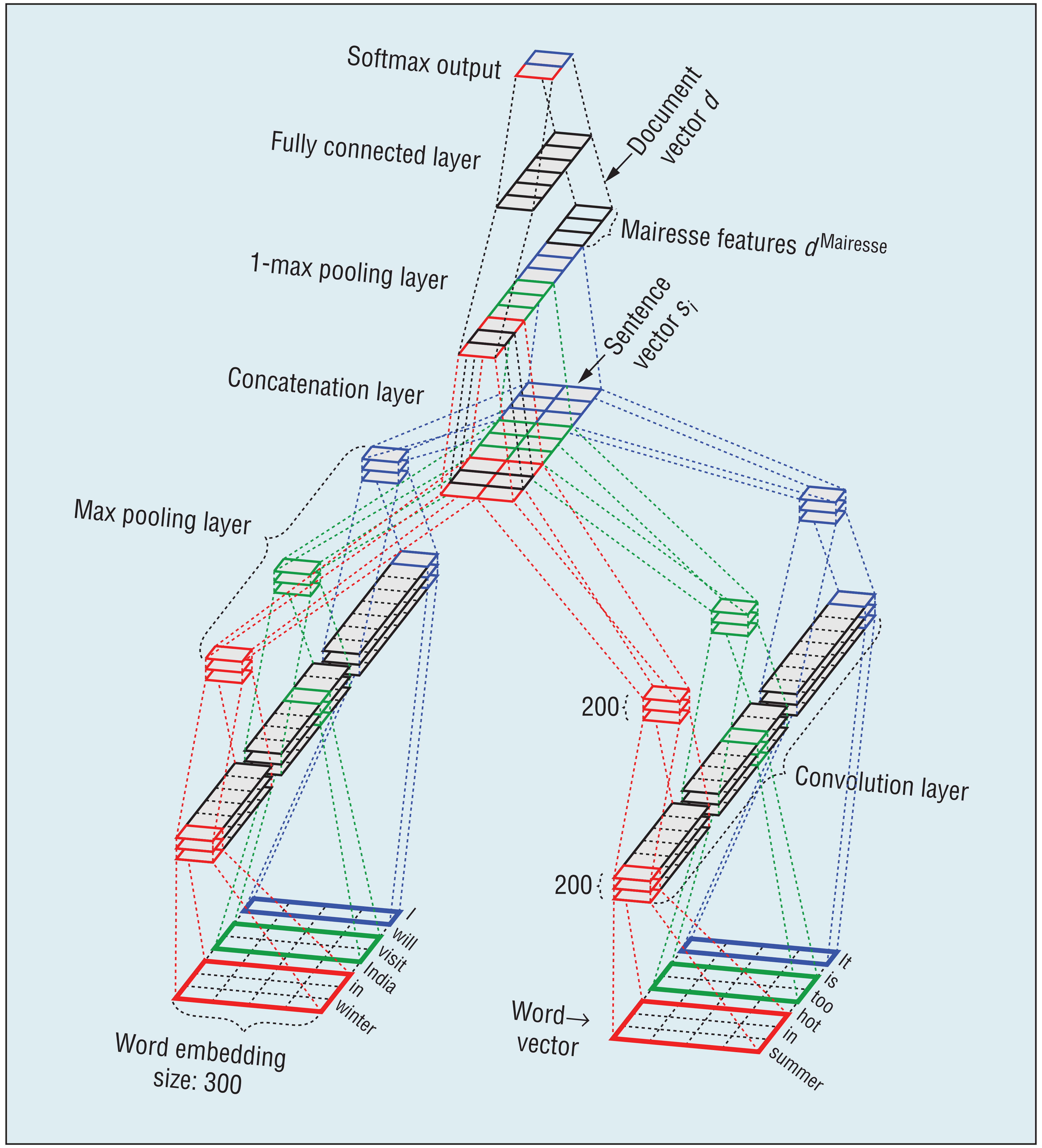


Figure 1. Architecture of our network. The network consists of seven layers. The input layer (shown at the bottom) corresponds to the sequence of input sentences (only two are shown). The next two layers include three parts, corresponding to trigrams, bigrams, and unigrams. The dotted lines delimit the area in a previous layer to which a neuron of the next layer is connected—for example, the bottom-right rectangle shows the area comprising three word vectors connected with a trigram neuron.

Table 1. Accuracy obtained with different configurations.

Document vector *d*	Filter	Classifier	Convolution filter	Personality traits				
				EXT	NEU	AGR	CON	OPN
N/A	N/A	Majority	N/A	51.72	50.02	53.10	50.79	51.52
Word *n*-grams	Not used	SVM	N/A	51.72	50.26	53.10	50.79	51.52
Mairesse[12]	N/A	SVM	N/A	55.13	58.09	55.35	55.28	59.57
Mairesse (our experiments)	N/A	SVM	N/A	55.82	58.74	55.70	55.25	60.40
Published state of the art per trait[12]	N/A	N/A	N/A	56.45	58.33	56.03	**56.73**	60.68
CNN	N/A	MLP	1, 2, 3	55.43	55.08	54.51	54.28	61.03
CNN	N/A	MLP	2, 3, 4	55.73	55.80	55.36	55.69	61.73
CNN	N/A	SVM	2, 3, 4	54.42	55.47	55.13	54.60	59.15
CNN + Mairesse	N/A	MLP	1, 2, 3	54.15	57.58	54.64	55.73	61.79
CNN + Mairesse	N/A	SVM	1, 2, 3	55.06	56.74	53.56	56.05	59.51
CNN + Mairesse	N/A	sMLP/FC	1, 2, 3	54.61	57.81	55.84	**57.30**	62.13
CNN + Mairesse	Used	sMLP/MP	1, 2, 3	**58.09**	57.33	**56.71**	56.71	61.13
CNN + Mairesse	Used	MLP	1, 2, 3	55.54	58.42	55.40	56.30	**62.68**
CNN + Mairesse	Used	SVM	1, 2, 3	55.65	55.57	52.40	55.05	58.92
CNN + Mairesse	Used	MLP	2, 3, 4	55.07	**59.38**	55.08	55.14	60.51
CNN + Mairesse	Used	SVM	2, 3, 4	56.41	55.61	54.79	55.69	61.52
CNN + Mairesse	Used	MLP	3, 4, 5	55.38	58.04	55.39	56.49	61.14
CNN + Mairesse	Used	SVM	3, 4, 5	56.06	55.96	54.16	55.47	60.67

*Bold indicates the best result for each trait.