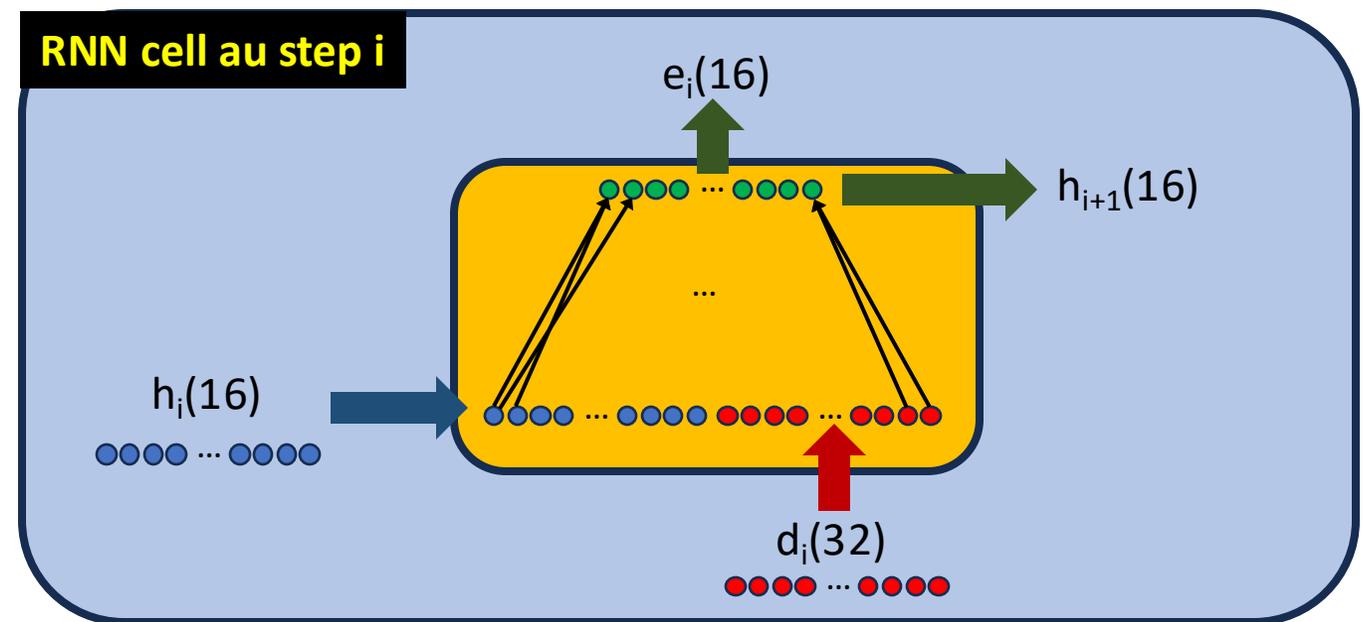


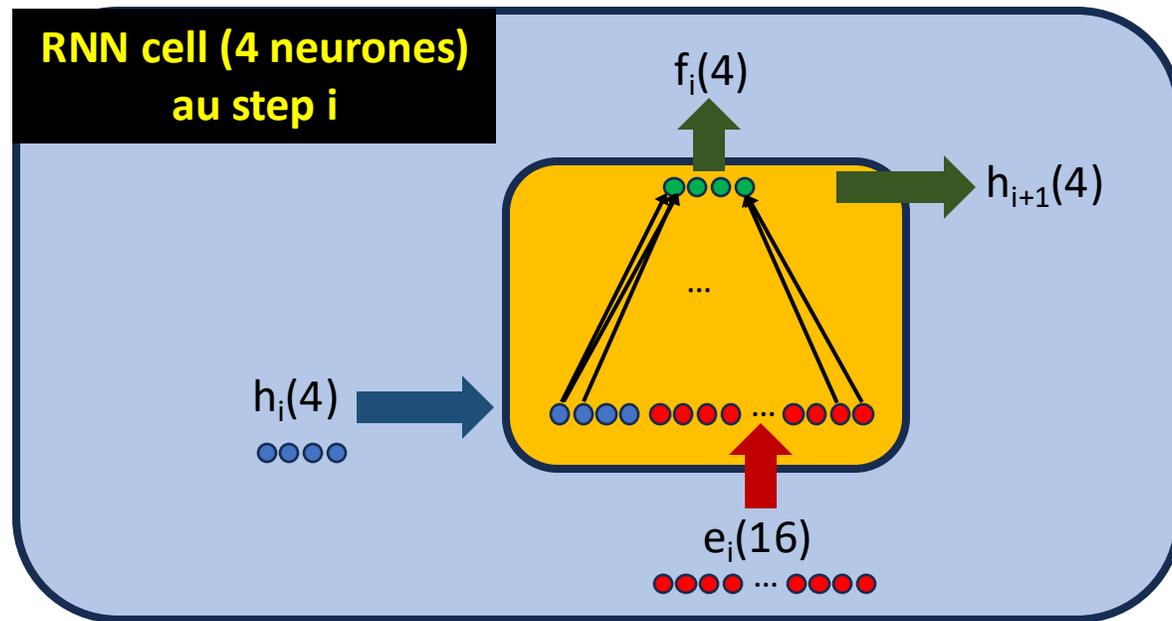
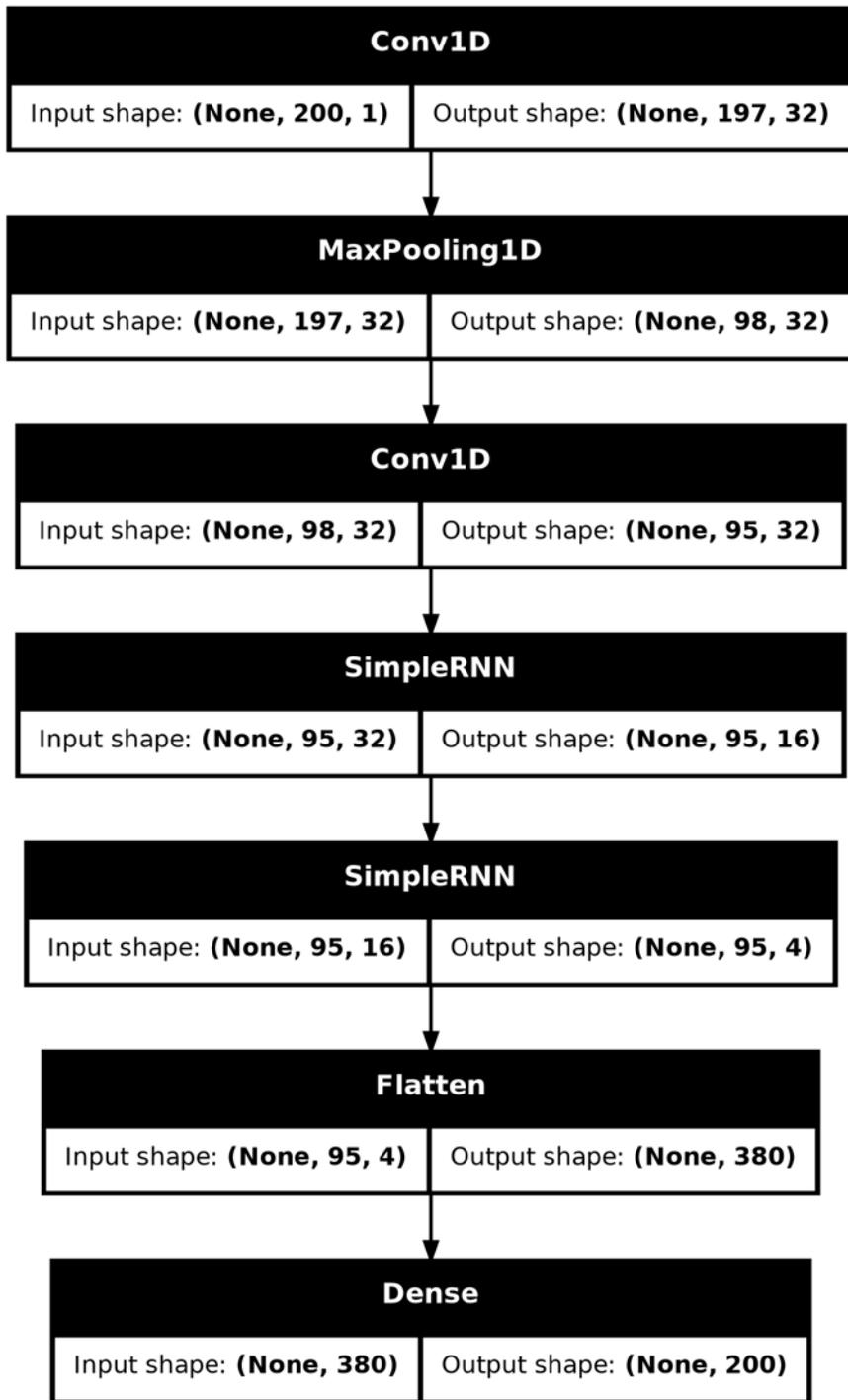
1. Notre entrée est une time serie de 200 échantillons (**$a(200,1)$**), fait passer ça dans 32 filtres de convolutions de taille 4.

2. La sortie est une matrice **$b(197 \times 32)$** , on fait un coup de max pooling, ie on prend les vecteurs de taille 32 2 par 2 et on garde le max des 2 (element-wise)

3. La sortie est une matrice **$c(98 \times 32)$** , on refait le même conv1D avec 32 autres filtres

4. La sortie est une matrice **$d(95 \times 32)$** , c'est ça qui va passer dans le premier RNN, qui a 16 neurones. Ce qui veut dire que la cellule RNN va conserver l'info du step précédent dans un vecteur h de taille 16





5. La sortie du RNN une matrice **e(95,16)**. L'état de la cellule RNN doit être conservé dans le vecteur h de taille 16 (16 neurones) pour chacun des 95 steps. Le vecteur h au step i a pris en compte les infos des steps 0 à i-1. En pratique il va perdre assez vite la mémoire, c'est pour ça qu'on a les LSTMs ensuite.



6. Après le deuxième RNN à 4 neurones on a une matrice f(95x4) qui passe dans une couche flatten



7. Le vecteur résultant, de taille 380, passe dans un réseau dense qui ramène tout à la taille initiale