



GRANDE ÉCOLE  
D'ACTUARIAT  
ET DE GESTION  
DES RISQUES

INSTITUT DE SCIENCE FINANCIÈRE ET D'ASSURANCES

---

Master 2 DARM

# SPIKING NEURAL NETWORKS

BAH Boubacar

KOFFI Mahounou

Avril 2023

# Table des matières

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Réseaux de neurones</b>	<b>3</b>
<b>3</b>	<b>Spiking Neural Network (SNN)</b>	<b>5</b>
3.1	Définition et principes du SNN . . . . .	5
3.2	Architecture et fonctionnement du modèle SNN . . . . .	5
3.3	Différents modèles de SNN . . . . .	6
3.3.1	Le modèle Integrate-and-Fire (IF) . . . . .	7
3.3.2	Le modèle Leaky Integrate and Fire (LIF) . . . . .	7
3.3.3	Le modèle d'Izhikevich . . . . .	7
3.4	SNN vs Autres modèles(ANN) . . . . .	8
3.5	Les méthodes d'apprentissage pour les SNN . . . . .	8
<b>4</b>	<b>Les défis de l'apprentissage dans les SNN</b>	<b>9</b>
<b>5</b>	<b>Application</b>	<b>9</b>
5.1	Méthodologie . . . . .	9
5.2	Résultats expérimentaux . . . . .	10
<b>6</b>	<b>Conclusion</b>	<b>11</b>

# 1 Introduction

La profusion de données, la facilité de leur acquisition et l'avancée technologique expliquent en partie l'intérêt croissant du monde scientifique pour les réseaux de neurones artificiels.

En effet, ces concepts datant des années quarante ont suscité un intérêt relativement récent et cet engouement ne cesse de croître ces dernières années. Ainsi, grâce à la masse de données à disposition des chercheurs et des outils de plus en plus performants, les réseaux de neurones artificiels ont vu leurs performances s'accroître rapidement, avec des formalisations de réseaux de plus en plus complexes à l'image du fonctionnement des réseaux neuronaux biologiques. Nous sommes en effet loin des premières formalisations du neurone formel<sup>1</sup> et du perceptron<sup>2</sup>.

Ces concepts ont conduits aux premières générations de réseaux de neurones multicouches basés sur le perceptron (MLP : MultiLayer Perceptron), qui ont été formalisés par ROSENBLATT (1958). Le principe de rétropropagation à quant à lui été appliqué aux MLP par WERBOS (1981) et sa mise en application a permis d'atteindre la seconde génération des réseaux de neurones profonds constituant le Deep Learning.

Les recherches les plus récentes quant à elles s'orientent vers des méthodes plus sophistiquées avec pour principal objectif d'optimiser les capacités des réseaux de neurones, mais aussi de répondre à des problématiques comme la quantité importante d'énergie nécessaire au fonctionnement des modèles d'apprentissage profond. Ils visent également à prendre en compte des paramètres supplémentaires afin de simuler au mieux le fonctionnement des réseaux neuronaux biologiques. Les évolutions actuelles portent en effet sur la troisième génération de réseaux de neurones basés sur une architecture à impulsions comme les réseaux neuronaux biologiques. Ces réseaux neuronaux à impulsions (SNN : Spike Neural Networks) constituent le sujet de notre analyse et dans la suite de ce rapport, nous ferons une présentation structurée et détaillée de ce modèle d'apprentissage.

Dans la première partie de notre analyse, nous présenterons le principe et l'architecture des SNN, ainsi que leur fonctionnement et leur apport au domaine de l'apprentissage non supervisé.

La seconde partie de notre analyse consistera en une application numérique basée sur des données réelles, permettant ainsi d'illustrer ce concept.

---

1. MCCULLOCH et PITTS (1943)

2. ROSENBLATT (1957)

## 2 Réseaux de neurones

Les neurones sont des cellules excitables du système nerveux qui servent à recevoir et transmettre des informations nerveuses. L'excitabilité est une propriété qui rend les neurones capables de réagir aux stimulations et de les transformer en impulsions. Quant à la transmission appelée aussi conductivité, elle leur sert à transmettre l'information reçue. On parle de réseau de neurones en raison de l'interconnexion des neurones et la dépendance entre eux. En effet ils forment un réseau important de réception et de transmission de messages nerveux.

L'étude des (réseaux de) neurones et de leur fonctionnement constitue la discipline des neurosciences et elles s'appliquent à plusieurs domaines comme la psychologie, la biologie, la médecine ou encore l'informatique. L'application des neurosciences dans le cadre de l'informatique consiste à reproduire le fonctionnement des réseaux de neurones biologiques en créant des réseaux neuronaux artificiels ou formels. La figure 1 suivante présente la structure d'un neurone biologique avec une analogie pour la constitution d'un neurone formel.

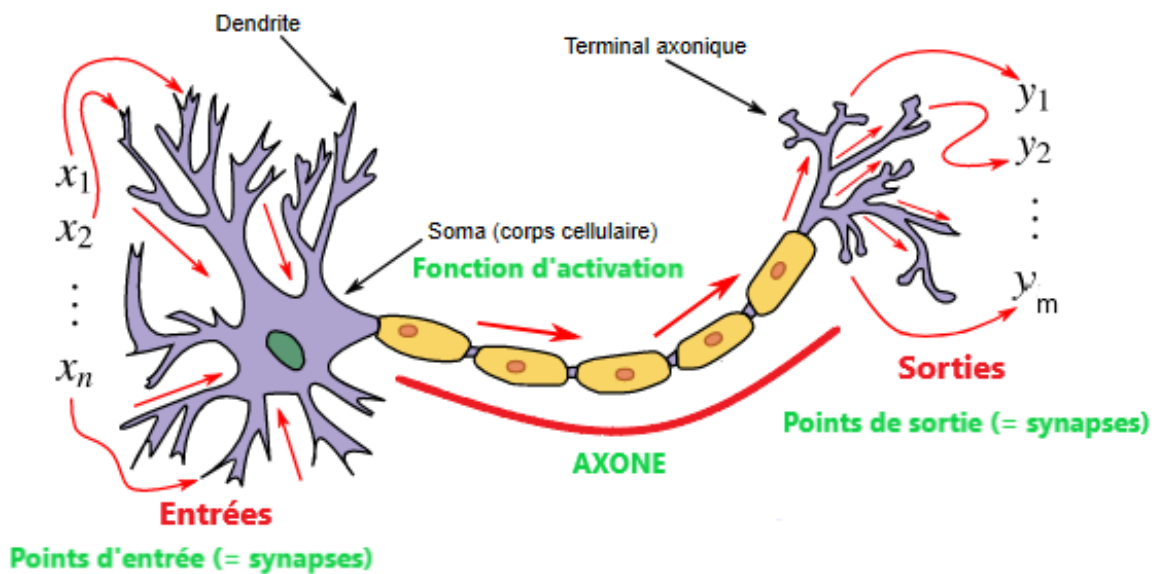


FIGURE 1 – Neurone biologique et analogie neurone formel

Un neurone formel est selon la formalisation de MCCULLOCH et PITTS (1943) une fonction non linéaire dépendant de paramètres appelés poids ou coefficients. Les variables de cette fonction sont ici représentées par les entrées du neurone et la sortie est la valeur qu'on obtient après le processus.

La figure 2 présente la modélisation mathématique d'un neurone artificiel avec :

- $x_i$  les entrées ou données,
- $w_{ij}$  les poids ou coefficients qui permettent de pondérer la transmission des entrées,

- La fonction de combinaison ou d'intégration  $\Sigma$  qui sert à calculer la sortie du neurone en fonction de chacun des poids attribués
- $\varphi$  une fonction d'activation ou de transfert qui donne la sortie du neurone.
- le vecteur  $\theta_j$  composé des  $j$  sorties.

Soient les vecteurs  $X = \begin{pmatrix} x_1 \\ \dots \\ x_n \end{pmatrix}$  et  $W = \begin{pmatrix} w_{11} \\ \dots \\ w_{n1} \end{pmatrix}$  ; l'équation correspondant à cette formalisation mathématique s'écrit :

$$\theta_j = \varphi(\Sigma) = \sum_{i=1}^n X_i \times W_{ij}.$$

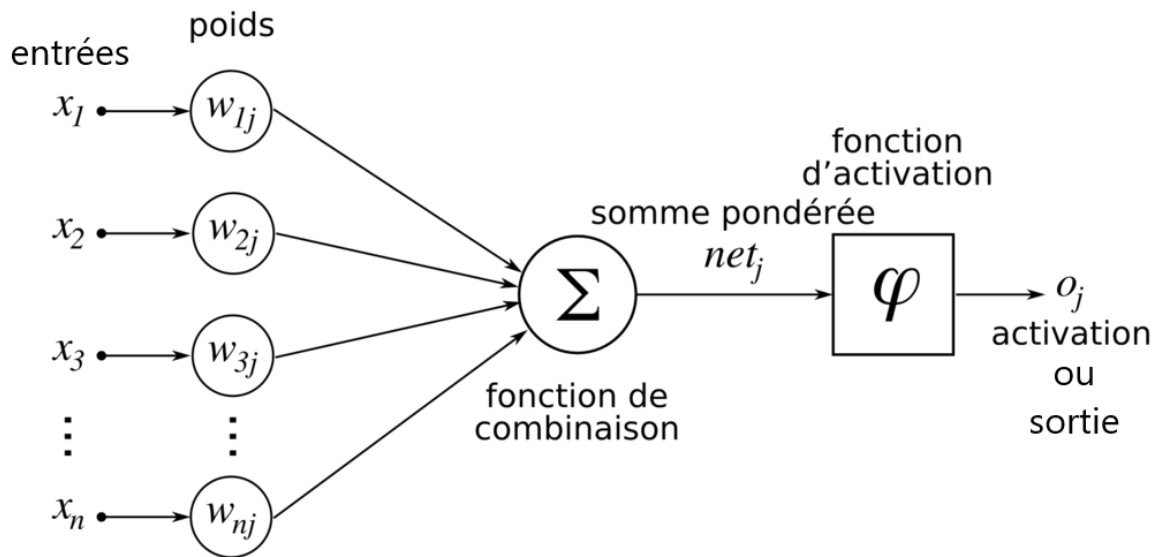


FIGURE 2 – Neurone artificielle

Les réseaux de neurones formels sont ainsi des algorithmes ou calculs réalisés et implémentés avec un ordinateur, dans le but de simuler le plus fidèlement possible le comportement de réseaux de neurones naturels.<sup>3</sup> Pour ce faire, on procède à l'apprentissage du réseau de neurones qui est la phase pendant laquelle l'algorithme reçoit des données en entrée afin de lui permettre d'apprendre à simuler un comportement à partir des exemples qui lui sont fournis. Le réseau est ainsi capable grâce à l'apprentissage de reproduire le comportement attendu et aussi de reconnaître des déviations ou des différences. Les réseaux de neurones peuvent être construits par exemple pour la détection de fraudes dans des données bancaires, pour prédire le salaire d'un individu en fonction de paramètres reçus en entrée. Ou même de reconnaître des images, par exemple d'écriture manuscrite.

Malgré leur imitation qui se veut proche des réseaux de neurones naturels, les réseaux neuronaux artificiels de seconde génération ne prennent pas en compte toutes les propriétés du système nerveux. Par exemple, la question de la communication entre

---

3. Source : [Universalis](#)

neurones qui est naturellement basée sur des transmissions successives de signal et dans un intervalle de temps donné, n'est pas prise en compte dans ces réseaux classiques. La Spike-timing-dependent plasticity (STDP) <sup>4</sup> ou plasticité dépendante du temps d'occurrence des impulsions, est un processus biologique de modification du poids des synapses qui n'est pas intégré dans le comportement des réseaux de neurones classiques.

Cette prise en compte de la temporalité dans le processus d'échanges entre neurones est l'objet (entre autres) des Spiking Neural Network (SNN) qui sont des réseaux de neurones impulsionnels et constituent une nouvelle génération de réseaux plus proche du fonctionnement des réseaux biologiques.

## 3 Spiking Neural Network (SNN)

### 3.1 Définition et principes du SNN

Tout comme les réseaux neuronaux biologiques, les SNN fonctionnent sur le principe d'impulsions intervenant au niveau des échanges entre neurones. Ces réseaux constituent une nouvelle génération de réseaux artificiels qui est très sophistiquée car les SNN imitent le mieux le comportement des neurones naturels. Par exemple, ils fonctionnent sur la redescende d'activation et leur encodage est binaire et tient compte du temps entre l'émission et la réception du signal. Ces caractéristiques font des SNN des modèles qui sont plus difficiles à mettre en place et à interpréter car ils n'utilisent pas des techniques usuelles comme la descente de gradient.

Néanmoins, leur architecture à l'avantage de limiter la perte d'information en mettant en avant des échanges entre neurones proches. En effet, le signal passe d'un neurone à un autre en fonction de leur proximité. Ce qui implique que le message ne passera pas nécessairement par tous les neurones mais uniquement à ceux qui sont les plus proches du neurone émetteur. Toutes ces caractéristiques font des SNN des modèles plus performants que les autres réseaux de neurones classiques même si leur complexité ne facilite pas leur construction et leur usage. De plus, ils sont énergivores et coûteux, même si depuis quelques années de nombreux progrès ont été fait. Les avancées sont surtout dues aux meilleures performances des outils d'implémentation et de la profusion des données pour entraîner les modèles.

L'un des paramètres important est ainsi le moment d'émission du signal et non sa durée ou son amplitude. De plus, on tient compte du laps de temps entre deux transmissions d'impulsions : il s'agit de la période réfractaire.

### 3.2 Architecture et fonctionnement du modèle SNN

La figure 6 suivante présente l'architecture d'un réseau de neurones à impulsions (SNN), qui se compose d'une couche d'entrée et d'une couche de traitement. Ce traitement est basé sur un réseau « Winner-Take-All » (WTA) avec des neurones excitatoires

---

4. ABBOTT, SONG et MILLER (2000)

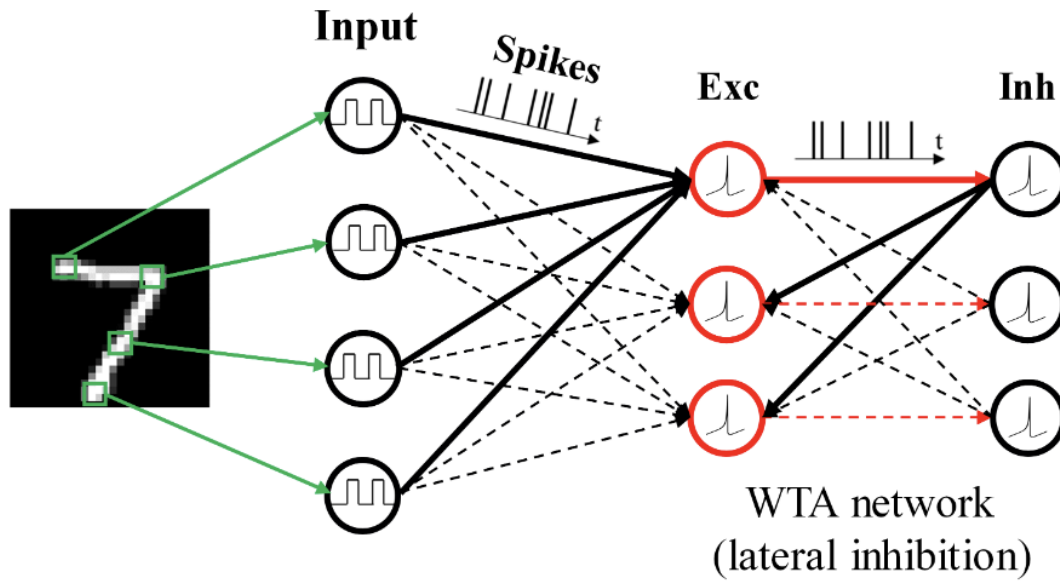


FIGURE 3 – Architecture d'un SNN

Source : [Researchgate](#)

(Exc) connectés à des neurones inhibiteurs (Inh)<sup>5</sup> pour induire une inhibition latérale.

La figure 6 montre un exemple de reconnaissance du chiffre 7 qui est extrait de la base de données MNIST de caractères manuscrits.

Les techniques de rétropropagation traditionnelles ne fonctionnent pas directement avec les Spiking Neural Networks (SNN) en raison de la nature des fonctions d'activation utilisées dans les SNN.

Pour les fonctions d'activation dans le cadre de SNN, on préfère utiliser des fonctions non-différentiables. En effet, les fonctions d'activation utilisées pour les SNN doivent produire des pics d'activité au niveau des émissions et échanges de signaux. Par définition, ces fonctions ne sont pas dérivables et donc en raison de leur nature, il n'est pas possible d'utiliser la rétropropagation usuelle.

Il existe néanmoins des méthodes de calcul alternatif pour les gradients et ces méthodes pour y remédier. L'approche la plus utilisée dans ce cas est l'**algorithme SpikeProp** proposé par BOHTÉ, KOK et POUTRÉ (2000).

### 3.3 Différents modèles de SNN

Plusieurs modèles de SNN ont été développés, et nous pouvons citer à titre d'exemples l'AdEx (Adaptive Exponential Integrate-and-Fire Model), le SEM (Single Exponential Model) et le DEM (Double Exponential Model). Les plus connus sont les modèles Integrate-and-Fire (IF) et les modèles Leaky-Integrate-and-Fire (LIF) que nous allons présenter dans la suite de notre analyse.

5. L'excitabilité d'un neurone (cellule nerveuse) est sa capacité de générer et de conduire rapidement un influx électrique à partir des dendrites ou du corps cellulaire, puis le long de l'axone jusqu'aux synapses. Des neurotransmetteurs sont excitateurs et d'autres sont inhibiteurs. (Source : [Psychomedia](#))

### 3.3.1 Le modèle Integrate-and-Fire (IF)

Dans ce modèle, le courant d'entrée est intégré au fil du temps jusqu'à ce que le potentiel membranaire atteigne un seuil fixé. Lorsque le potentiel membranaire dépasse ce seuil, le neurone « émet une impulsion » et le potentiel membranaire est réinitialisé à une valeur de repos. Le modèle **Integrate-and-Fire** s'écrit sous la forme :

$$\tau_m \frac{dV(t)}{dt} = -V(t) + RI(t) \quad (1)$$

où :

- $\tau_m$  est la constante de temps membranaire,
- $V(t)$  est le potentiel membranaire à un temps donné  $t$ ,
- $R$  est la résistance membranaire,
- et  $I(t)$  est le courant d'entrée à un temps donné  $t$ .

Lorsque le potentiel membranaire atteint un certain seuil, le neurone émet une impulsion, ce qui correspond au « fire » (feu) du modèle Integrate-and-Fire.

### 3.3.2 Le modèle Leaky Integrate and Fire (LIF)

Le modèle Leaky Integrate-and-Fire (LIF) reflète la diffusion d'ions qui se produit à travers la membrane lorsque l'équilibre n'est pas atteint dans la cellule en introduisant un terme de « fuite » dans le modèle IF. En raison de sa simplicité et de son faible coût de calcul, le modèle LIF et ses variantes sont largement utilisés . Ce modèle est représenté par l'équation suivante :

$$C_m \frac{dV_m}{dt} = -G_L(V_m - E_L) + I_{syn}(t) \quad (2)$$

Si  $v_m \geq v_\theta$ ,  $v_m \leftarrow v_{peak}$  alors  $v_m \leftarrow v_{reset}$  Le seuil de tension est représenté par  $v_\theta$ , le potentiel d'action par  $v_{peak}$ , et le potentiel de repos par  $v_{reset}$ . Lorsque la tension atteint le seuil  $v_\theta$  (souvent fixé à 1 pour simplifier), le neurone génère un potentiel d'action, puis le potentiel est réinitialisé à zéro pendant une période réfractaire  $\tau_{ref}$  qui limite la fréquence de décharge du neurone.

### 3.3.3 Le modèle d'Izhikevich

Le modèle d'Izhikevich est un modèle mathématique qui décrit le comportement des neurones à décharge. Contrairement aux modèles plus complexes, tels que le modèle de « Hodgkin-Huxley », le modèle d'Izhikevich est basé sur des équations différentielles ordinaires relativement simples, qui permettent une simulation rapide et efficace des réseaux neuronaux.

$$\frac{dV_m}{dt} = k(v_m - E_L)(v_m - v_t) - u + I_{syn}(t) \quad (3)$$



$$\frac{du(t)}{dt} = a(b((v_m - E_m) - u)) \quad (4)$$

si  $V_m \geq V_0$ , alors  $V_m \leftarrow c, u \leftarrow u + d$  où  $u$  représente l'activation des courants ioniques  $K^+$  et l'inactivation des courants ioniques  $Na^+$  ( $pA$ ), et  $v_t$  est le potentiel de seuil instantané ( $mV$ ).

### 3.4 SNN vs Autres modèles(ANN)

La principale différence entre un réseau de neurones classique (ANN) et un SNN réside dans la notion temporelle qui est intégré dans ce dernier. Dans les ANN, l'inférence est effectuée sur la base d'une seule passe d'alimentation avant, à travers le réseau. Un SNN, en revanche, est constitué d'un réseau de neurones qui communiquent à travers une séquence d'impulsions binaires sur un certain nombre d'étapes de temps  $T$  appelé latence d'inférence du SNN.

Propriétés	ANN	SNN
Représentation de l'information	scalaires	Impulsions
Paradigme d'apprentissage	Rétropopagation	Plasticité / RP
Plate-forme	VLSI <sup>6</sup>	Neuromorphic VLSI

TABLE 1 – Comparaison des propriétés entre réseaux neuronaux biologiques, ANN et SNN

### 3.5 Les méthodes d'apprentissage pour les SNN

- Apprentissage supervisé : l'entraînement est fait sur des données étiquetées pour apprendre à prédire des sorties cibles. On utilise des algorithmes classiques comme le rétropropagation du gradient pour ajuster les poids synaptiques.
- Apprentissage par renforcement : les SNN apprennent à générer des stratégies optimales par essais-erreurs en interagissant avec un environnement. On utilise des algorithmes comme le Q-learning<sup>7</sup>
- Apprentissage non supervisé : les SNN structurent automatiquement les données non étiquetées pour apprendre des régularités. On utilise des algorithmes comme l'auto-encodeur pour compresser les données en un code latent.

---

7. le Q-learning est un algorithme d'apprentissage par renforcement. Il ne nécessite aucun modèle initial de l'environnement. La lettre 'Q' désigne la fonction qui mesure la qualité d'une action exécutée dans un état donné du système. [Wikipedia](#)

## 4 Les défis de l'apprentissage dans les SNN

- La dépendance à l'historique : les SNN ont une grande dépendance aux entrées passées. Leur boucle de rétroaction peut ralentir l'apprentissage et rendre le SNN sensible aux perturbations.
- La complexité de l'architecture : ces modèles sont basés sur des mécanismes biologiques complexes, ce qui nécessite une conception d'architecture spécifique pour fonctionner efficacement.
- La complexité de l'apprentissage : les réseaux de neurones à impulsions sont plus complexes à entraîner que les réseaux de neurones artificiels classiques. Ils nécessitent souvent des algorithmes d'apprentissage spécifiques, tels que les algorithmes d'apprentissage par renforcement, pour optimiser les performances.
- Le manque de données d'apprentissage : les SNN nécessitent souvent plus de données d'apprentissage que les réseaux de neurones classiques pour obtenir de bonnes performances. Cela peut être dû à la complexité accrue de l'architecture des SNN.
- L'interprétabilité : Les SNN peuvent être plus difficiles à interpréter que les réseaux de neurones classiques en raison de la nature discrète des impulsions. Cela peut rendre la compréhension des décisions prises par le réseau plus difficile.

## 5 Application

### 5.1 Méthodologie

Dans la partie pratique, nous mettrons en place un modèle de reconnaissance d'images(chiffres). On travaillera sur le jeu de données **MNIST**, qui est composé d'un ensemble d'apprentissage de 60 000 images de 28 x 28 pixels, ainsi que d'un ensemble de test de 10 000 images. Les détails de nos choix des différentes méthodes seront expliqués dans le code ainsi que l'interprétation des résultats.

Nous avons implémenté deux modèles différents. Les deux ont plutôt de bon score de prédiction mais le deuxième est beaucoup plus coûteux en terme de temps. Cette différence est due aux nombres de calculs faits dans batch et chaque époque de chaque batch. Nous montrerons les détails des sorties des deux modèles dans la sous section suivante.

```

60000 images d'entraînement
10000 images de test
Perte par epoch
[Epoch 1] Perte: 0.435
[Epoch 2] Perte: 0.155
[Epoch 3] Perte: 0.103
[Epoch 4] Perte: 0.076
[Epoch 5] Perte: 0.059
[Epoch 6] Perte: 0.046
[Epoch 7] Perte: 0.038
[Epoch 8] Perte: 0.030
[Epoch 9] Perte: 0.025
[Epoch 10] Perte: 0.020
Précision du réseau sur les 10000 images de test : 98.08 %
Le temps total est de 66s

Process finished with exit code 0

```

FIGURE 4 – Modèle 1

## 5.2 Résultats expérimentaux

Ce modèle fait moins de calcul .En effet ,Il calcule la perte globale par epoch.Il fait une bonne prédiction sur les données d'apprentissage car nous pouvons remarquer une baisse significative dans les différents epoch et il a une bonne performance globale.

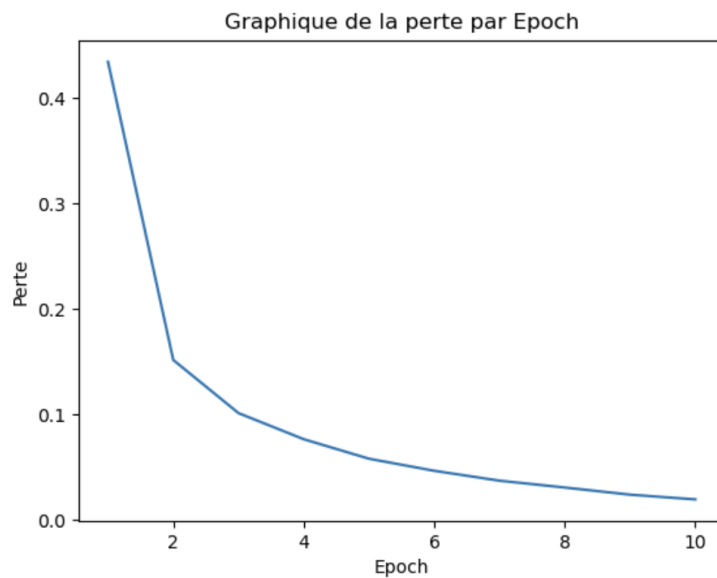


FIGURE 5 – Courbe de la perte dans les epochs

Ce graphique montre comment évolue la perte pour les dix différents Epochs. Nous pouvons donc remarquer la baisse significative de la perte précédemment mentionnée.

Ce modèle calcule la performance dans chaque Epoch et la performance dans chaque batch de chaque Epoch. Ce qui fait un grand nombre de boucles et de calculs.

```
Batch 0: Loss = 0.0878172293305397
Batch 100: Loss = 0.022081753239035606
Batch 200: Loss = 0.0909414067864418
Batch 300: Loss = 0.16124394536018372
Batch 400: Loss = 0.1188330203294754
Taux de réussite:98.06%
Epoch 10
Batch 0: Loss = 0.04333774745464325
Batch 100: Loss = 0.02628766931593418
Batch 200: Loss = 0.09035807102918625
Batch 300: Loss = 0.01965722255408764
Batch 400: Loss = 0.09678816050291061
Taux de réussite:98.02%

Performance globale: 98.14%
Le temps total est de 396s

Process finished with exit code 0
```

FIGURE 6 – Modèle 2

Ce modèle nous permet de savoir dans quel batch de chaque Epoch le SNN a obtenu le meilleur taux de réussite pendant l'entraînement. Connaître le batch le plus performant peut aider à comprendre comment le SNN apprend et comment optimiser le processus d'entraînement.

## 6 Conclusion

En conclusion, les SNN peuvent être efficaces pour certaines tâches spécifiques telles que la reconnaissance de la parole et la traduction automatique, mais leur architecture plus complexe les rend plus coûteux en termes de consommation d'énergie.

Ce sont des modèles prometteurs pour le traitement des séries temporelles, la modélisation de concepts abstraits ou l'intelligence artificielle générale. Bien que leur apprentissage et leur interprétabilité doivent encore progresser, les recherches futures permettront de relever ces défis et d'exploiter pleinement leur potentiel. Les SNN pourraient révolutionner de nombreux domaines, de la compréhension de texte à la prise de décision en passant par la mémoire ou le raisonnement.

## Références

- [1] W. McCULLOCH et W. PITTS. « A Logical Calculus of the Ideas Immanent in Nervous Activity ». In : *Bulletin of Mathematical Biophysics* vol. 5 (1943), 115-133.
- [2] F. ROSENBLATT. *The Perceptron, A perceiving and recognizing automaton*. Report NO. 85-460-1. Cornell Aeronautical Laboratory, 1957.
- [3] F. ROSENBLATT. « The Perceptron: A Probabilistic Model for Information Storage and Organization in The Brain ». In : *Psychological Review* vol. 65.6 (1958).
- [4] P. WERBOS. *Applications of Advances in Nonlinear Sensitivity Analysis*. Proceedings of the 10th IFIP Conference, 1981, 762-770.
- [5] L. ABBOTT, S. SONG et K. MILLER. « Competitive Hebbian learning through spike-timing-dependent synaptic plasticity. » In : *Nature Neuroscience* vol. 3.9 (2000), p. 919-926. DOI : [10.1038/78829](https://doi.org/10.1038/78829).
- [6] S. BOHTÉ, J. KOK et H. La POUTRÉ. « SpikeProp: backpropagation for networks of spiking neurons ». In : *The European Symposium on Artificial Neural Networks*. 2000.
- [7] W. GERSTNER et W. KISTLER. *Spiking Neuron Models : Single Neurons, Populations, Plasticity*. Cambridge University Press, 2002, 93-146. DOI : [10.1017/CB09780511815706.005](https://doi.org/10.1017/CB09780511815706.005).
- [8] F. POLUNAK et A. KASINSKI. « Introduction to spiking neural networks: Information processing, learning and applications ». In : *Acta neurobiologiae experimentalis* vol. 70.4 (2011), p. 409-433.
- [9] K. YAMAZAKI et al. « Spiking Neural Networks and Their Applications : A Review. » In : *Brain Sci.* vol. 12 (2022), 93-146. DOI : [10.3390/brainsci12070863](https://doi.org/10.3390/brainsci12070863).
- [10] OPENAI. *ChatGPT*. 2023. URL : <https://chat.openai.com/chat>.