

Reconnaissance de la métrique des poèmes arabes par les réseaux de neurones artificiels

Hafedh El Ayech¹, Amine Mahfouf², Adnane Zribi³

Institut Supérieur de Gestion de Tunis – Département Informatique

¹ hafedh.elayech@edunet.tn

² amine.mahfouf@cgi.com

³ adn@gnet.tn

Résumé

Nous avons construit un système capable de reconnaître les modes de composition pour les poèmes arabes, nous décrivons dans cet article les différents modules du système. Le recours à une technique d'apprentissage artificiel pour classer une séquence phonétique de syllabes est justifiable par le fait que nous avons imité le processus d'apprentissage naturel humain suivi par les poètes pendant des siècles. Les réseaux de neurones artificiels de type Perceptron multicouches ont montré un pouvoir très puissant de classification. **Mots-clés** : réseau de neurones, perceptron multi couches, classification, poèmes arabes, syllabes, analyse phonétique.

Abstract

We have created a new system to recognize the different composition modes of Arab poetry. In this article we describe the different modules of this system. The use of an artificial learning technique for grouping a phonetic sequence of syllables is explained by the fact that we imitate the human process learning.

Keywords: artificial neural networks, multi layer perceptron, classification, arabic poetry, syllables, phonetic analysis.

1. Introduction générale

L'objet de ce travail n'est pas l'élaboration d'un formalisme ni d'un ensemble de règles permettant de décrire les poèmes arabes. Un tel formalisme et de telles règles existent depuis les origines de la poésie arabe. L'on parle même de la science du '*aroudh*' (علم العروض). Il s'agit plutôt pour nous de traduire le formalisme et les règles existantes en termes informatiques. Nous établirons des règles, nous construirons des algorithmes et nous collectionnerons des données de base qui nous permettront, pensons-nous, de représenter les connaissances de la science de la métrique. Les buts derrière la réalisation d'un tel système de reconnaissance sont multiples : d'abord pour l'aspect EAO (Enseignement Assisté par Ordinateur), ensuite pour le tri et la classification automatique des poèmes. Les règles du '*aroudh*' sont nombreuses et souvent nuancées de manière à les rendre difficilement exploitables. Par ailleurs, trouver la bonne représentation informatique, voire la représentation optimale pour une règle souvent formulée en un langage des plus libres, n'est pas simple. Ce papier se présente en trois sections. La première section présentera la terminologie du '*aroudh*', la problématique traitée et la démarche à suivre pour réaliser le système. Nous consacrerons la deuxième section à traiter le module de traduction d'un vers de poème à une séquence phonétique composée de syllabes. Enfin la troisième section traite le module de classification de la séquence phonétique à sa classe adéquate parmi 18 classes qui présentent les modes de compositions réguliers dans la poésie arabe.

2. Présentation du système et Problématique

2.1. Historique

Autrefois, durant plusieurs siècles, les poètes arabes composaient des poèmes spontanément sans nécessairement savoir écrire ou lire. Les modes de composition étaient inspirés directement de la nature ; les poètes apprennent des milliers d'exemples de vers de poèmes pour pouvoir composer correctement leurs poèmes. À titre d'exemple le mode 'Khabab' symbolisait le rythme du trot du cheval. Cette façon de composer des vers était donc spontanée jusqu'à l'arrivée du linguiste grammairien Khalil El Farahidi au VIII^e siècle. Ce dernier a défini les modes de composition et toutes les règles qu'il faut suivre pour l'écriture des poèmes. Cette méthode consiste à regrouper les mètres dans des classes appelées cercles d'El Khalil (Sammoud, 1996), mais cette méthode n'est pas tout à fait irréprochable puisqu'elle ne considère que la forme originale des séquences phonétiques des mètres.

2.2. Définitions des règles de la métrique

2.2.1. L'écriture métrique

La langue arabe ne s'écrit pas toujours comme elle se prononce. Ceci pose un problème dans l'écriture et la reconnaissance des poèmes car les règles du 'aroudh' reposent sur la prononciation. De ce fait, le vers dans sa forme originale est transformé en une écriture de prononciation pour faire apparaître des syllabes. Cette écriture s'appelle l'écriture métrique *الكتابة العروضية*. Exemple : هذا الرجل معظم devint هَذَا رَجُلٌ مَعْظَمٌ

Plus loin dans cet article, nous verrons comment cette écriture n'a pas été prise en compte de manière systématique.

2.2.2. Principe général et Terminologie

Dans la métrique la première tâche à faire est de distinguer les syllabes courtes des syllabes longues afin de déterminer ensuite le mode de composition (*bahr* بحر) du vers. Dans ce contexte la notion de voyellation est très importante dans la mesure où pour distinguer les syllabes on doit voir la prononciation de la lettre avec sa voyellation et avec son successeur.

Il existe quatre types de voyellation : voyellation courte َ ُ ِ , voyellation longue (Tanwin) ً ٌ ٍ , voyellation muette (Soukoun) ◌◌ et la chadda ّ.

On appelle **lettre consonne** toute lettre qui peut supporter tous les types de voyellation comme بَ حَ تَ . En contrepartie, est appelée **lettre voyelle** toute lettre qui peut ne pas supporter de voyellation (و ي) ou ne supporte pas de voyellation (ا). Une **syllabe** est dite **courte** si et seulement si la lettre de départ a pour correspondant une voyellation courte et la lettre qui la suit immédiatement a une voyellation courte ou longue. C'est le cas dans يَدٌ . Une **syllabe** est dite **longue** si et seulement si la lettre a pour correspondant une voyellation courte et son successeur est soit une lettre voyelle soit une lettre qui a une voyelle muette. C'est le cas dans قَفْ , مَ (voyellations longues) ou dans يَدٌ . Pour les besoins de la reconnaissance des mètres, une codification doit être effectuée. Elle est en fait très simple : les syllabes courtes seront symbolisées par le signe v, les syllabes longues seront symbolisées par le signe - . Après transformation des syllabes courtes et longues en des symboles v et - , les combinaisons obtenues formeront ce qu'on appelle des *Tafila* تَفِيلَة. Par exemple la séquence de codes - - v - est une *Tafila* et sera, pour des raisons de facilité, prononcée فَعْلَانٌ . Une *Tafila* est donc une séquence de syllabes. C'est la séquence de *Tafila* composant la première

moitié d'un vers (*sadr* - صدر) qui va permettre d'identifier le mètre (*bahr* - بحر) du vers. Par exemple, le *Moutakareb* المتقارب est composé d'une séquence de quelques *Tafiila* - - v فعولن. Ce mètre accepte toutefois quelques permissions de déformation impliquant plusieurs séquences différentes. Le *Tawil* الطويل quant à lui est composé des deux *Tafiila* - - - v مفاعيلن et -- v فعولن. Nous allons considérer 18 modes de compositions (ou mètres), dont sept composés d'une seule *Tafiila* et les onze restants composés de deux *Tafiila*. Nous les présentons ci-dessous avec les différentes *Tafiila* qui les composent ainsi que les déformations acceptées de celles-ci (زحافات) :

Original	فعولن	مفاعيلن	فاعلاتن	فاعلن	مستفعلن	مُتفاعِلن	مفاعِلتن	مفعولات
Les Zihafs	فعولٌ	مفاعيلٌ	فِعالَتِن فاعلات	فِعالِن - فِعالِن	مُتَفِعلِن مُتَفِعلِن	مُتَفاعِلِن	مُفَاعِلَتِن	مُفَعولات

Tableau 1. Les différentes *Tafiila* et leurs déformations permises

A	المتقارب	G	الكامل	M	المجتث
B	الهزج	H	الوافر	N	السرّيع
C	الرّمّل	I	الطويل	O	المنسرح
D	المتدارك خيب	J	البسيط	P	المديد
E	المتدارك المحدث	K	مخلع البسيط	Q	المقتضب
F	الرّجز	L	الخفيف	R	المضارع

Tableau 2. Liste de 18 mètres (modes de composition à reconnaître)

Annonçons déjà que la méthodologie du grammairien Khalil El Farahidi se base sur la notion de *Tafiila* seulement pour classer les vers. Nous préférons traiter la séquence phonétique complète du vers.

2.3. État de l'art

Mostageer (1980) a essayé de décrire les mètres sous forme d'indices numériques à la place des *Tafiila*. Cette méthode numérique ne sort pas outre mesure des règles de la métrique et son application donne un aspect de coordination aux mètres. Elle permet de faire une distinction numérique à chaque *bahr* et un classement dans des ensembles qui facilitent leur reconnaissance. La démarche adoptée se base sur le bloc de syllabes courtes et longues de chaque *bahr*. L'extraction de l'indice numérique se fait sur le rang de l'apparition des syllabes courtes dans le bloc de syllabes. La théorie classique et traditionnelle considère que la *Tafiila* est la base de la reconnaissance d'un *bahr*. Mostageer propose de remplacer la *Tafiila* par des numéros, ce qui engendre une évolution vers une codification prête pour le traitement automatique.

Exemples :

- le Moutakareb	- - v	- - v	- - v	- - v	المتقارب
	10	7	4	1	
- le Tawil	- - v	- - v	- - v	- - v	الطويل
	11	8	4	1	

Cette idée est intéressante mais incomplète dans la mesure où elle n'est valable que pour la forme originale des mètres. En se basant sur la syllabe courte comme étant déterminante de l'indice identifiant le *bahr*, la méthode se trouve souvent dans un conflit extrême. C'est que la syllabe courte peut apparaître dans un autre rang à cause de ce qu'on appelle les déformations

consonnes sont : G,S,e,ç,p,è,D,é,b,E,g,B,R,T,F,t,Q,C,k,J,L,H,M,q,N,d,h,V,r,ù,Z,s, respectivement

ث,ش,ء,ص,ض,أ,ط,إ,ظ,آ,ع,ب,غ,ت,ف,ة,ق,ث,ك,ج,ل,ح,م,خ,ن,د,ه,ذ,و,ر,ؤ,ز,س,

- Une lettre voyelle est une lettre qui appartient à l'ensemble $Iv = \{W, Y, y, I\}$ { و ي ا }.
- Si W (و) ou Y (ي) portent une voyellation, elles ne seront pas considérées comme des voyelles mais comme des consonnes, sinon elles seront considérées comme voyelle.
- y (ى) et I (ا) sont des lettres purement voyelles (**lpv**), elles ne portent pas de voyellation.

Dans notre approche pour la reconnaissance des syllabes, nous avons essayé de simuler le travail manuel que fait l'Homme pour la reconnaissance des syllabes longues et courtes. L'algorithme que nous présentons pour ce faire est simple. Il consiste à considérer d'abord un caractère de la chaîne à traiter et à regarder sa voyellation correspondante, puis à faire de même pour le caractère suivant pour enfin prendre la décision, selon les règles de la métrique, s'il s'agit d'une syllabe longue ou courte. La figure 2 explique l'algorithme utilisé pour la reconnaissance des syllabes.

La voyellation utilisée est représentée par des abréviations dont voici l'explication :

VC : voyellation courte, **VL** : voyellation longue, **VCC** : voyellation chadda courte

VCL : voyellation chadda longue, **VM** : voyellation muette (ou 'soukoun'), **LV** : lettre voyelle et **LPV** : lettre purement voyelle. Le pointeur « Pas » ne peut se pointer que sur une lettre consonne porteuse d'une VC ou d'une VCC ou une lettre consonne porteuse d'une VL ou d'une VCL.

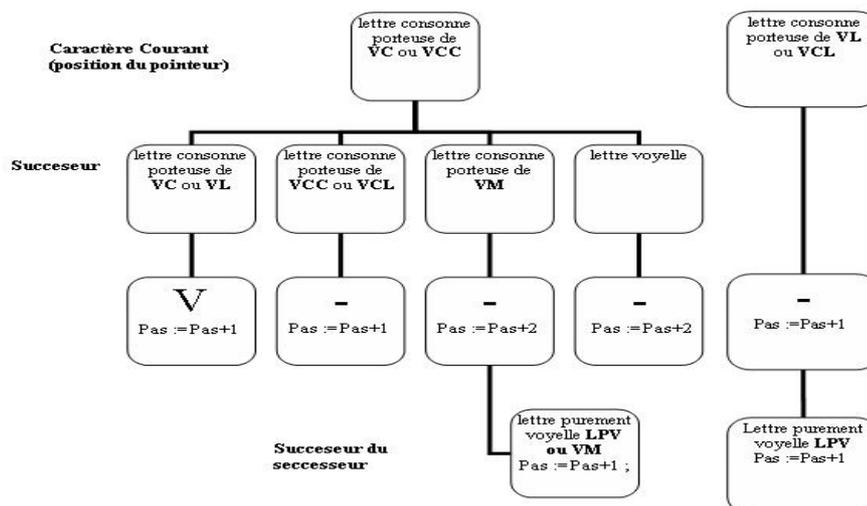


Figure 2. La génération de la séquence des syllabes

4. Reconnaissance du mètre avec un réseau de neurones

4.1. L'architecture des réseaux de neurones

Un réseau de neurones est un graphe orienté et pondéré. Les nœuds de ce graphe sont des automates simples appelés neurones formels dotés d'un état interne qui représente son activation, par lequel ils influencent les autres neurones du réseau. Cette activité se propage dans le graphe le long d'arcs pondérés appelés liens (ou poids) synaptiques. L'état du réseau entier est composé de l'activation de ses neurones constitutifs et des matrices des poids synaptiques reliant une couche à la suivante, chacune étant une matrice dans laquelle s'inscrivent les poids des liens. La réponse o_j d'un neurone j est donnée comme suit :

$$o_j = f(\text{net}_j) = f\left(\sum_{i=1}^n x_i w_{ij}\right) \quad \text{avec} \quad f(x) = \frac{1}{1 + e^{-x}}$$

On peut répartir les neurones du réseau en trois ensembles : l'ensemble des neurones d'entrée correspondant aux neurones qui perçoivent les données du problème (Jodouin, 1994) ; l'ensemble des neurones de sortie qui est un sous ensemble de neurones dont l'activation sera interprétée comme la réponse du réseau ; l'ensemble des neurones cachés correspondant aux neurones cachés, c'est-à-dire ceux qui ne sont ni à l'entrée ni à la sortie du réseau. Les neurones cachés n'ont pas de liens directs avec l'extérieur et agissent donc par l'intermédiaire d'autres neurones. Un réseau disposant de neurones cachés est souvent plus puissant qu'un réseau sans neurones cachés. Bien qu'il existe une variété d'architectures et de topologies de connexions, nous nous intéressons essentiellement à l'architecture multicouches appelée MLP (Multi Layer Perceptron) et la structure généralisée appelée feed-forward non récurrente. Les neurones sont arrangés par couches ; il n'y a pas de connexion entre neurones d'une même couche et les connexions ne se font qu'avec les neurones des couches avales. Chaque neurone d'une couche est connecté à tous les neurones de la couche suivante et à celle-ci seulement, bien que dans les réseaux généralisés de type feedforward, on peut trouver des connexions qui peuvent sauter directement depuis la couche d'entrée vers la couche de sortie. Nous adopterons la notation suivante pour désigner l'architecture : $\text{MLP}(n_{\text{couche1}}, n_{\text{couche2}}, \dots, n_{\text{couche}i}, \dots, n_{\text{couche}l})$, avec n_i est le nombre de neurones pour la couche i .

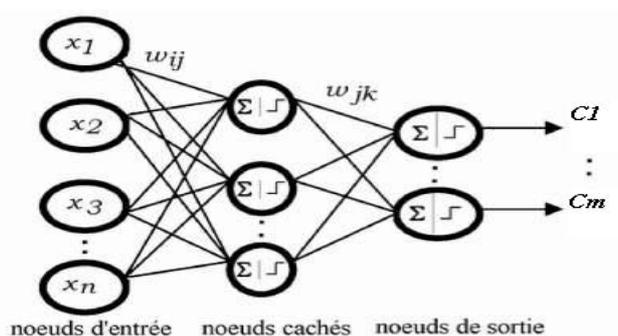


Figure 3. Réseau de type MLP à 3 couches

4.2. Classification par le Perceptron multi couches

Nous abordons un problème d'apprentissage supervisé (où le réseau doit connaître la réponse qu'il aurait dû donner) qui consiste à construire (ou estimer) une fonction $f(x)$ à partir des données observées $(x_1, y_1), \dots, (x_n, y_n)$, de telle sorte que $f(x)$ soit une bonne

approximation de la réponse souhaitée Y . Souvent on choisit f de manière à minimiser la somme des erreurs quadratiques (dans notre cas, c'est la fonction coût ou fonction Objectif)

$$F = \sum_t (Y_t - f(X_t))^2$$

L'algorithme de rétro-propagation basé sur la technique du gradient est bien adapté à la résolution de ce problème. La tâche de ce type d'application a pour but d'associer à une entrée donnée l'étiquette d'une forme connue. Les neurones d'entrées peuvent être des valeurs réelles ou binaires ; quant aux neurones de sorties, qui présentent les classes, les valeurs désirées sont souvent strictement binaires (les réponses données par le réseau sont comprises entre 0 et 1). Lorsqu'on présente dans la phase d'apprentissage un vecteur d'entrée (représentant des caractéristiques), on désire avoir une seule unité (classe) qui vaut 1 et les autres unités (classes) valant 0 au niveau de la couche de sortie. Une fois le réseau entraîné (les pondérations de synapses étant déjà ajustées), lorsque nous voulons reconnaître la classe d'un vecteur de caractéristiques qui se présente en entrée, nous allons avoir des valeurs réelles comprises dans l'intervalle $[0,1]$ (tout dépend de la fonction de transfert au niveau de la couche de sortie, il s'agit de la fonction sigmoïde unipolaire dans notre cas). La classe la plus probable est celle qui comporte la valeur la plus proche de 1. Dans d'autres modélisations, il y aura un seul neurone actif parmi les neurones de sorties tandis que les autres seront inactifs.

4.3. Apprentissage d'un réseau de neurones

Rappelons qu'il s'agit d'atteindre des valeurs de poids optimales qui minimisent la fonction objectif ; pour ce faire on utilise l'algorithme de rétro-propagation qui exige une architecture ayant au moins une couche cachée. De plus, la fonction de transfert, qui transforme l'activation en réponse au niveau d'une couche cachée, doit être non-linéaire. Le réseau de neurones MLP procède par un entraînement avec des exemples connus. L'ensemble de vecteurs d'entraînement est défini par $\{(x_p, y_p) | p=1, 2, \dots, P\}$; le vecteur x_p est alimenté à travers la couche d'entrée. Le réseau calculera son vecteur de sortie après le passage du stimulus à travers les couches cachées du réseau, le vecteur o_p est comparé à celui que l'on désire obtenir y_p , et on en déduit le vecteur d'erreur $y_p - o_p$. Le critère que l'on doit minimiser est la fonction objectif (ou fonction coût) qui est la somme des carrés des erreurs (en Anglais SSE : Sum of Squared Errors). Elle est exprimée de la manière suivante :

$$F = \sum_p F_p = \frac{1}{2} \sum_p \sum_k (y_{pk} - o_{pk})^2$$

où p est l'indice du vecteur d'apprentissage, o_{pk} est la sortie calculée du neurone k correspondant au vecteur d'apprentissage p , P étant le nombre total de vecteurs d'apprentissage et N étant le nombre de neurones de sortie.

L'erreur calculée sera rétropropagée à travers le réseau, et les poids w_{ij} seront modifiés. L'algorithme de rétropropagation utilise la technique de la descente du gradient pour minimiser la distance entre la sortie désirée et la sortie obtenue par le réseau :

$$\Delta w_{ij} = -\eta \frac{\partial F_p}{\partial w_{ij}}$$

la formule de mise à jour des poids synaptiques sera : $w_{ij}^{p+1} = w_{ij}^p + \Delta w_{ij}$

L'algorithme de rétro-propagation peut être résumé par les étapes suivantes après avoir fixé l'architecture du réseau (le nombre de couches, le nombre de neurones dans chaque couche,

les paramètres d'apprentissage comme le choix de la fonction de transfert pour les neurones des couches cachées et de sortie ainsi que le paramètre η appelé gain d'apprentissage) :

1. Initialiser aléatoirement les poids du réseau et initialiser le nombre d'itération n à zéro.
2. Présenter premièrement les vecteurs d'entrée à partir des données d'apprentissage dans le réseau.
3. Envoyer le vecteur p d'entrée à travers le réseau pour obtenir une sortie

$$o_{pk} = f\left(\sum_j w_{jk} f\left(\sum_m w_{mj} f\left(\dots f\left(\sum_i w_{il} x_i\right)\right)\right)\right)$$

4. Calculer un signal d'erreur entre la sortie réelle et la sortie désirée $O_p - y_p$, calculer la somme carré des erreurs (la fonction objectif F à minimiser) et Incrémenter n .
5. Envoyer le signal d'erreur en arrière à travers le réseau
 - Pour chaque unité de sortie k , calculer :

$$\delta_k = (o_k - y_k) f'(net_k)$$

- Pour chaque unité cachée j , calculer :

$$\delta_j = f'(net_j) \sum_k \delta_k w_{jk}$$

6. Corriger les poids pour minimiser l'erreur par la mise à jour suivante :

$$\Delta w_{ij}^{(n+1)} = \eta \delta_j o_i + \alpha \Delta w_{ij}^{(n)}$$

où α est une constante appelée momentum servant à améliorer le processus d'apprentissage.

7. Répéter les étapes 2-6 avec le prochain vecteur d'entrée jusqu'à ce que l'erreur soit suffisamment petite ou jusqu'à ce qu'un nombre d'itérations maximal fixé au préalable soit atteint.

4.4. Description du réseau de neurones pour Classifier une séquence de syllabes

L'architecture choisie exige 15 neurones pour la couche d'entrée et 18 pour la couche de sortie ; quant aux couches cachées, on peut choisir librement le nombre de neurones en testant des architectures différentes jusqu'à aboutir à la configuration la plus optimale. Le vecteur caractéristique $(x_1, x_2, \dots, x_{15})$ d'entrée est composé d'une séquence de syllabes. Chaque attribut x_i prend : la valeur 0 si la syllabe correspondante est courte (v), la valeur 1 si la syllabe est longue (-) ou la valeur -1 si la syllabe est absente. Le choix de 15 neurones en entrée revient à la longueur de la séquence la plus longue dans la base des exemples d'apprentissage, toute séquence de longueur inférieure à 15 sera concaténée par des -1, c'est-à-dire que les attributs x_i restants auront la valeur -1. Le choix de 18 neurones pour la couche de sortie est très évident puisque nous avons 18 classes (18 modes ou 'bahr' de composition) à reconnaître.

Exemple :

Soit la séquence d'entrée vv-v-vv-v-vv-v-, le vecteur d'entrée équivalent qui se présente au réseau est (0,0,1,0,1,0,0,1,0,1,0,0,1,0,1), le vecteur Y désiré à la sortie est (0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0)

Soit une autre séquence d'entrée v--v--v--v--, le vecteur équivalent est (0,1,1,0,1,1,0,1,1,0,1,1,-1,-1,-1), son vecteur Y désiré à la sortie est

(1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0). Nous avons fixé le paramètre η à 0.25 et le paramètre α à 0.9. La base des exemples d'apprentissage était extraite d'une centaine de poèmes dont on connaît les modes de compositions, nous avons utilisé le module de génération de séquence phonétique décrit dans la section n°3 pour créer cette base. La taille de la base est de 1481 exemples.

4.5. Sélection de l'architecture du réseau de neurones classifieur

Nous avons déjà décidé du nombre de neurones en entrées et celui en sortie, nous n'avons pas encore décidé du choix du nombre de couches cachées ainsi que du nombre de neurones pour chaque couche. Nous allons tenter plusieurs configurations possibles et nous allons en choisir une selon des critères de mesure de performances. Le premier critère est la valeur de la fonction objectif, le deuxième étant de tester le réseau déjà entraîné sur un échantillon de test de taille pareille de celle qui a servi à l'apprentissage et nous noterons le pourcentage de reconnaissance pour chaque mode de composition ainsi que la moyenne des pourcentages des reconnaissances pour les 18 modes de compositions.

Enfinement nous nous basons aussi sur le critère d'information d'Akaike et sur le critère d'information bayésien connus dans la littérature pour évaluer la performance d'un modèle : plus ces critères donnent des valeurs petites, plus le modèle est performant.

$$AIC(p) = n \log \left(\frac{SSE}{n} \right) + 2p \qquad BIC(p) = n \log \left(\frac{SSE}{n} \right) + p + p \log n$$

Où p représente le nombre de paramètres à estimer (dans notre cas p représente le nombre de synapses reliant les neurones entre eux), SSE est la somme des carrés des erreurs et n correspond au nombre d'observations utilisé pour ajuster le modèle. Pour les Perceptrons multi couches le nombre de paramètres p est donnée comme suit :

$$p = \sum_{i=1}^{l-1} nbneurons_i \times nbneurons_{i+1}$$

où l est le nombre total de couches dans le réseau.

Le tableau 3 montre les résultats descriptifs pour différentes architectures MLP après apprentissage et test. Le critère d'arrêt de l'apprentissage a été fixé à 500 époques, le mot époque voulant dire le nombre de fois que la base d'apprentissage a été totalement parcourue. Les résultats sont impressionnants et le pouvoir de classification de notre modèle est très puissant.

MLP(15,10,18)	MLP(15,18,18)	MLP(15,26,18)	MLP(15,18,18,18)	MLP(15,26,26,18)
SSE 3,51265E-9	SSE 3,67195768E-9	SSE 3,7990436423E-9	SSE 1,54839220E-8	SSE 1,1635843E-7
A: المتقارب 85,714%	A: المتقارب 96,428%	A: المتقارب 96,428%	A: المتقارب 96,42857%	A: المتقارب 100%
B: الهزج 33,333%	B: الهزج 100%	B: الهزج 100%	B: الهزج 100%	B: الهزج 100%
C: الرَّمَل 94,444%	C: الرَّمَل 100%	C: الرَّمَل 100%	C: الرَّمَل 97,222222%	C: الرَّمَل 100%
D: المتدارك خيب 75,8%	D: المتدارك خيب 86,20%	D: المتدارك خيب 86,206%	D: المتدارك خيب 100%	D: المتدارك خيب 75,862%
E: المتدارك المحدث 96,6%	E: المتدارك المحدث 100%	E: المتدارك المحدث 100%	E: المتدارك المحدث 86,666%	E: المتدارك المحدث 100%
F: الرّجَز 85,897%	F: الرّجَز 93,589%	F: الرّجَز 93,5897435%	F: الرّجَز 93,589%	F: الرّجَز 88,461%
G: الكامل 87,5%	G: الكامل 96,875%	G: الكامل 96,875%	G: الكامل 96,875%	G: الكامل 96,875%
H: الوافر 90%	H: الوافر 80%	H: الوافر 80%	H: الوافر 80%	H: الوافر 70%
I: الطويل 100%	I: الطويل 100%	I: الطويل 100%	I: الطويل 100%	I: الطويل 100%
J: البسيط 100%	J: البسيط 100%	J: البسيط 100%	J: البسيط 100%	J: البسيط 100%
K: مخلع البسيط 100%	K: مخلع البسيط 100%	K: مخلع البسيط 100%	K: مخلع البسيط 100%	K: مخلع البسيط 100%
L: الخفيف 100%	L: الخفيف 100%	L: الخفيف 100%	L: الخفيف 100%	L: الخفيف 100%

K:مخلع البسيط:100%	M:المجتث 100%	M:المجتث 100%	M:المجتث 100%	M:المجتث 100%
L:الخفيف 100%	N:التريع 100%	N:التريع 100%	N:التريع 100%	N:التريع 100%
M:المجتث 83,33%	O:المنسرح 100%	O:المنسرح 100%	O:المنسرح 100%	O:المنسرح 100%
N:التريع 100%	P:المديد 100%	P:المديد 100%	P:المديد 100%	P:المديد 100%
O:المنسرح 85,714%	Q:المقتضب 100%	Q:المقتضب 100%	Q:المقتضب 100%	Q:المقتضب 100%
P:المديد 100%	R:المضارع 100%	R:المضارع 100%	R:المضارع 100%	R:المضارع 100%
Q:المقتضب 100%	La moyenne est : 97,3944561%	La moyenne est : 97,3944561983%	La moyenne est : 97,26567799%	La moyenne est : 96,177700412%
R:المضارع 100%	AIC =-23331,22853	AIC =-25077,027768	AIC =-23914,38523	AIC =-23657,20050
La moyenne est : 89,91476967%	BIC =-23331,22853	BIC =-25077,027768	BIC =-22213,25757	BIC =-21200,01610
AIC =-21287,91049				
BIC =- 21287,910490				

Tableau 3. Les taux de reconnaissance après la phase de test

Les deux architectures MLP (15,18,18) et MLP (15,26,18) sont les meilleures. Nous opterons pour la deuxième parce qu'elle a engendré les valeurs minimales pour les critères AIC et BIC.

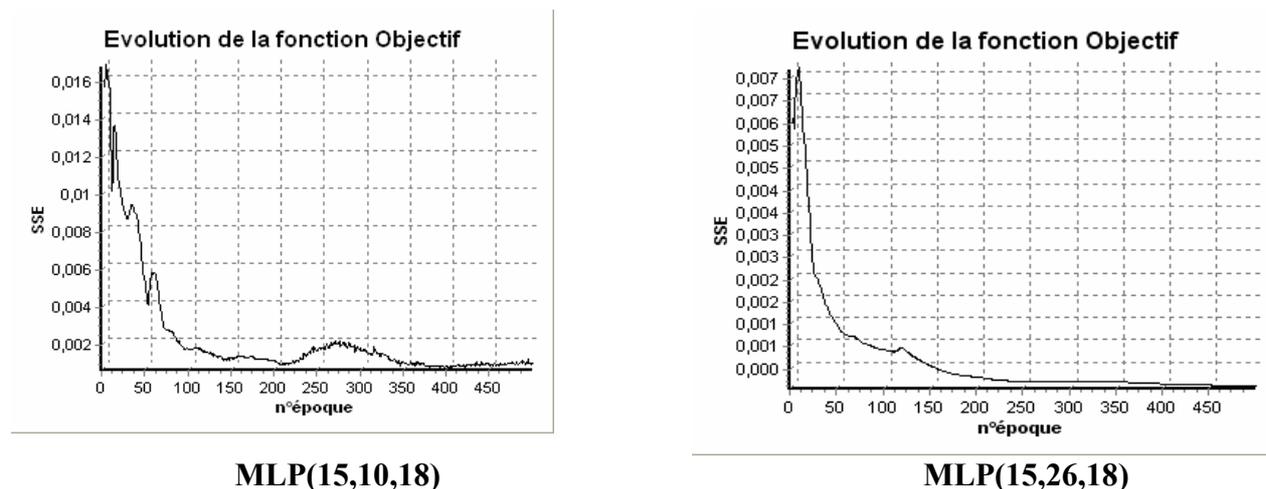


Figure 4. Evolution de la somme des carrés des erreurs durant l'apprentissage

5. Conclusion et Perspectives

Les résultats obtenus sont très encourageants avec des taux d'erreurs relativement faibles. Le système développé a grandement profité de la capacité des réseaux de neurones. Nous pourrions étendre ce travail en testant d'autres modèles probabilistes de classification de séquences comme les chaînes de Markov cachées. L'avantage de l'utilisation d'un modèle d'apprentissage est la possibilité qu'acquiert notre système à s'adapter pour reconnaître plus que les 18 modes connus si d'aventure nous venions à disposer d'exemples de nouveaux modes non cités par El Khalil le fondateur de la science de la métrique arabe.

Références

ABDI H. (1994). *Les réseaux de neurones*. Sciences et technologies de la connaissance. Presses Universitaires de Grenoble.

- CHEN W., CHEN S., LIN C. (1996). « A speech recognition method based on the sequential multi-layer perceptrons. *Neural Networks* 9 (4) : 655-669.
- CORNUÉJOLS A., MICLET L., KODRATOFF Y. (2003). *Apprentissage artificiel. Concepts et algorithmes*. Eyrolles.
- FREY P.W., SLATE D.J. (1991). « Letter Recognition Using Holland-style Adaptive Classifiers ». In *Machine Learning* 6 (2).
- JODOUIN J.F. (1994). *Les réseaux de neurones, principes et définitions*. Hermès, Paris.
- KECMAN V. (2001). *Learning and Soft Computing, Support Vector Machines, Neural Networks, and Fuzzy Logic Models*. The MIT Press, Cambridge.
- MOSTAGEER A. (1980). "الأدلة الرقمية في البحور الشعرية". *Les indices numériques dans les mètres de la poésie arabe*.
- SAMMOUD N. (1996). *Le Aroudh Précis*. دار شوقي للنشر تونس.
- WHITE H. (1990). « Connectionist nonparametric regression : Multilayered feedforward networks can learn arbitrary mapping ». In *Neural Networks* 3 : 535-550.