Chapitre VI: Classification des textes prophétiques basée sur les réseaux de neurones*

1. Introduction

Ce chapitre présente une étude comparative de deux modèles de réseaux de neurones artificiels à savoir le Perceptron Multi-Couches «Multi Layer Perceptron (MLP)» et les Fonctions à Base Radiale «Radial Basis Function (RBF)» pour la classification thématique des textes prophétiques. La classification automatique des textes arabes basée sur des réseaux de neurones artificiels n'a pas été explorée en détail jusqu'à présent. Dans ce chapitre, un corpus arabe sera utilisé pour construire et tester le modèle de réseaux de neurones artificiels. Des méthodes de représentation des documents et d'assignation des poids qui reflètent l'importance des termes dans ces documents seront discutées. Chaque document arabe sera représenté par un schéma terme-document, et comme le nombre des mots uniques dans la collection est très grand, la décomposition en valeurs singulières (Singular Value Decomposition ou SVD) sera utilisée pour sélectionner les attributs les plus pertinents pour le processus de classification. Les résultats expérimentaux montrent que les modèles de réseaux de neurones MLP et RBF, utilisant la décomposition en valeur singulière, sont capables d'atteindre de hautes performances. Les résultats montrent aussi que le classificateur MLP dépasse le classificateur RBF et que le modèle de réseaux de neurones utilisant la SVD est meilleur que le modèle original des réseaux de neurones.

Le reste de ce chapitre est organisé comme suit: la Section 2 présente les aspects principaux des deux modèles réseaux de neurones: Perceptron Multi-Couches «MLP» et Fonctions à Base Radiale «RBF». La Section 3 fournit le détail d'utilisation des modèles de réseaux de neurones pour la classification des textes arabes basés sur la Décomposition en Valeurs Singulières (SVD), méthode utilisée pour la réduction de l'espace d'attributs. Les résultats expérimentaux seront détaillés dans la Section 4. La Section 5 conclue le chapitre.

2. Classification par réseaux de neurones

Dans cette section, deux types de réseaux de neurones sont discutés brièvement avec référence aux structures et aux paramètres. Les différences principales parmi ceux-ci sont aussi discutées brièvement [Bishop, 1995] [Haykin, 1999] [Wasserman, 2000].

Un réseau de neurones est un réseau de nœuds tel que chaque nœud représente un modèle mathématique d'un neurone biologique. Ces réseaux ont une grande capacité d'auto-apprentissage, ils sont tolérants aux fautes et ils ont une grande immunité au bruit. Dans la dernière décennie, les réseaux neuraux ont été largement utilisés dans des applications de la vie courante, grâce à leur caractéristique d'apprentissage approximatif des réponses provenues de la plupart des systèmes de traitement. Ce comportement peut être modélisé d'une façon qui rend faisables les futures estimations des entrées similaires et qui donne de bons résultats.

Dans la pratique, il existe deux types d'architectures des réseaux de neurones, les réseaux de neurones «*feed-forward*» ou «passe-avant» et les réseaux de neurones «*feed-back*» ou «récurrents» qui sont appliqués dans des domaines de problèmes totalement différents [Haykin,

* Une grande partie de ce chapitre a été publiée dans les articles suivants :

1. F.Harrag, A. Hamdi-Cherif, , et E. El-Qawameh, Performance of MLP and RBF Neural Networks on Arabic Text Categorization Using SVD, *Neural Network World Journal*, ISSN 1210-0552, Vol.20, N.4, pp. 441-459, 2010.
2. F. Harrag et E. El-Qawameh, Improving Arabic Text Categorization Using Neural Network with SVD, *Journal of Digital Information Management*, ISSN 0972-7272, Vol.8, N.4, pp. 233-239, 2010.

1999]. Dans notre étude, deux types de réseaux de neurones sans boucles rétroactives sont utilisés et leurs performances sont testés dans le domaine de classification de texte, le Perceptron Multi-couches (MLP) et les Fonction à Base Radiale (RBF) sont les deux classificateurs les plus appropriés pour l'approximation des fonction interactives [Haykin, 1999].

La comparaison de ces deux architectures de réseaux de neurones a été déjà étudiée dans plusieurs domaines de recherche, tel que les systèmes dynamiques [Park et al., 2002], la compensation des canaux dans le traitement du signal [Jianping et al., 2002], la reconnaissance de la voix [Finan et al., 1996] et dans tout autre domaine où une estimation efficace, stable et avec de basse ressources en temps réel est exigée (les réseaux de neurones utilisés ont un petit nombre de nœuds). Les réseaux de neurones sont très populaires dans le domaine d'apprentissage, ils peuvent traiter le problème de classification des textes de deux manières différentes : linéairement et non- linéairement, les deux approches peuvent atteindre de bonnes performances [Park et al., 2002]. Ils ont été largement appliqués par beaucoup de chercheurs pour classer des documents avec différents types de vecteurs d'attributs. [Wermeter, 2000] a utilisé le titre du document comme vecteur pour la classification des textes. [Lam & Lee, 1999] ont utilisé la méthode d'analyse en composantes principales (ACP) comme une technique de réduction d'attributs pour les données d'entrée des réseaux de neurones.

2.1. Le Perceptron Multi-couches (MLP)

Les réseaux de neurones Multi-Couches sont beaucoup utilisés dans la modélisation des processus physiques non linéaires. Leur fonctionnement est le plus souvent basé sur le principe d'apprentissage supervisé. On ne présente ici que le cas des réseaux de neurones statique i.e. sans bouclage des sorties dans le réseau. On considère un modèle de réseau de neurones $g(x,w)$ avec x le vecteur des variables en entrée et w le vecteur des paramètres du réseau. Dans le cas d'un apprentissage supervisé, on considère aussi l'existence de N exemples où chaque exemple k est constitué d'un vecteur d'entrée x_k et d'un vecteur de sortie (ou de mesure) y_k associé à x_k.

Considérons qu'il existe une fonction $y=f(x)$ qui lie l'entrée x à la sortie y. On va alors utiliser le réseau de neurones pour qu'il approxime au mieux la fonction f. Pour cela on définit le résidu $r_k=y_k - g(x_k,w)$, et en utilisant la base d'apprentissage, on cherche donc à déterminer w de façon à minimiser r_k.

De façon informelle, le réseau peut se mettre en œuvre de la façon suivante :

- Déterminer les entrées pertinentes ;
- Récupérer les données nécessaires à l'apprentissage ;
- Définir la topologie du réseau (nombre de couches cachées) ;
- Estimer la valeur des paramètres (les poids synaptique ...) par apprentissage ;
- Évaluer la performance du réseau obtenu.

Utilisant un réseau de neurones MLP, le problème de classification des textes peut être résolu en utilisant l'algorithme de rétro-propagation du gradient de l'erreur. Cet algorithme a été appliqué dans l'identification des systèmes, la reconnaissance des motifs, la classification, le traitement d'images et le traitement des langages naturels, etc. La décision de classification pour tout document de dimension raisonnable est basée sur une évidence combinée de plusieurs sources, chaque mot dans un document sera donc considéré comme une source pour classer ce document [Rajan et al., 2009]. Il y a beaucoup de modèles de réseaux de neurones artificiels à savoir les réseaux du gradient standard et les réseaux du gradient stochastique. Dans notre étude, on a employé un réseau de neurones (MLP) à trois couches, de progression en avant, avec une fonction d'activation tangente hyperbolique (*tanh*) dans la couche cachée, suivi par une couche de sortie linéaire. Le réseau de neurones est entrainé par un algorithme de rétro-

propagation du gradient de l'erreur. Les entrées sont les composants des vecteurs de documents, les sorties sont les catégories des documents. La structure d'un réseau de neurones (MLP) de trois couches à rétro-propagation du gradient de l'erreur est montrée dans la Figure 6.1.

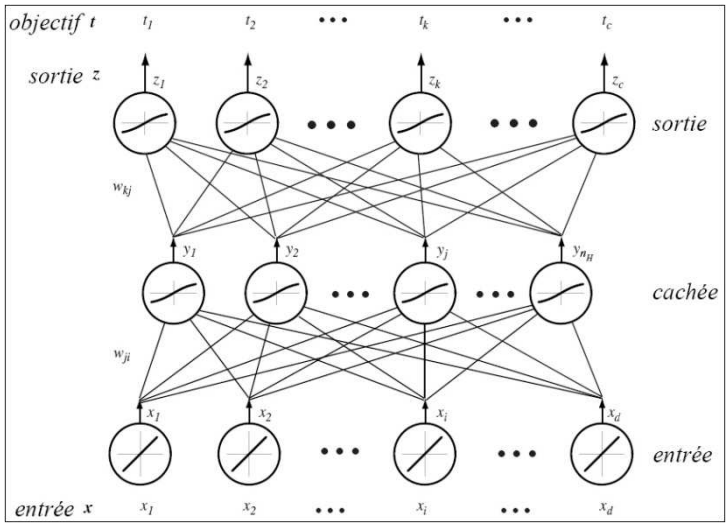

Fig. 6.1. Un réseau de neurones (MLP) typique de trois couches à rétro-propagation du gradient de l'erreur

2.2. Les Fonctions à bases radiales

Les Fonctions à bases radiales (RBF) est un modèle de réseaux de neurones proposé par [Powell, 1987], [Broomhead and Loewe, 1988], [Moody and Darken, 1989] et [Poggio and Girosi, 1989], ils sont apparues à la fin des années 80 comme une variante des réseaux de neurones. Cependant, leurs racines se retrouvent dans les techniques de reconnaissance de forme les plus anciennes comme les fonctions de potentiel (traduction de potential functions), le clustering et l'approximation fonctionnelle. Un RBF est constitué uniquement de 3 couches : la couche d'entrée qui retransmet les entrées sans distorsion, la couche cachée RBF qui contient les neurones RBF et la couche de sortie, une simple couche contenant une fonction linéaire. Chaque couche est entièrement connectée à la suivante. La Figure 6.2 montre la structure typique d'un Réseau de Neurone RBF [Chathura et al., 2008].

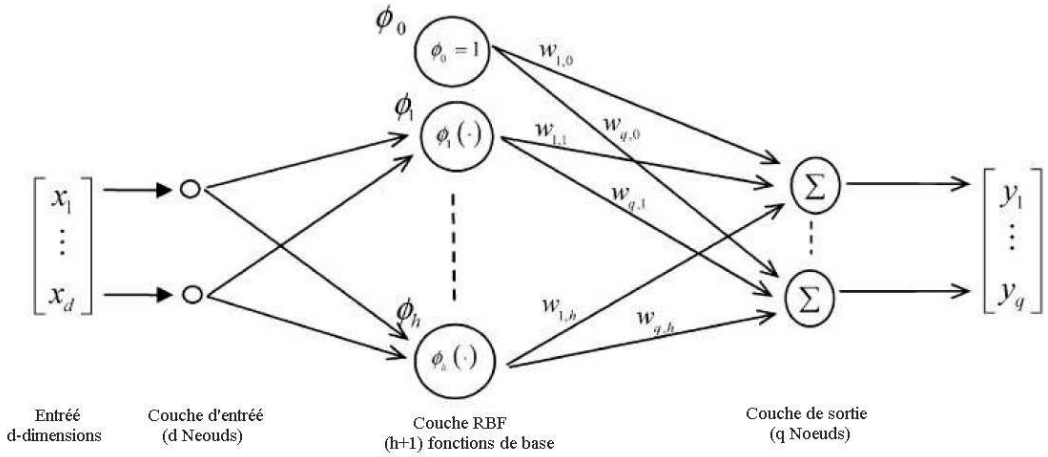

Fig. 6.2. Structure typique d'un réseau de neurones RBF

Le nombre de nœuds dans la couche d'entrée, la couche cachée et la couche de sortie est *d*, *h+1* et *q* respectivement. De la couche d'entrée à la couche cachée, chaque composant du vecteur d'entrée transmet l'information à chaque nœud caché sans aucun changement. Chaque cellule de la couche cachée utilise une fonction noyau (kernel function), telle que la Gaussienne, comme fonction d'activation. Cette fonction est centrée au point spécifié par le vecteur de poids associé à la cellule, la sortie du nœud caché est définie par :

$$\emptyset_i = \exp\left(-\sum_{j=1..d}\left(x_{kj} - c_{ij}\right)^2/\delta_i\right), i = 1,2,\ldots..,h+1 \qquad (6.1)$$

Où, x_{ij} est le *j*^{ième} composant du *k*^{ième} exemple d'entrée, c_{ij} est le *j*^{ième} composant du centre c_i de ce nœud, δ_i est le paramètre de graduation. La valeur de sortie du *n*^{ième} nœud du réseau $(1 \leq n \leq m)$ est définie par:

$$y_n = \sum_{j=1..h+1} \emptyset_i w_{nj}, i = 1,2,\ldots,h+1 \qquad (6.2)$$

Où w_{nj} est le poids de connexion du *j*^{ième} nœud caché au *n*^{ième} nœud de sortie.

Chaque neurone RBF contient une gaussienne qui est centrée sur un point de l'espace d'entrée. Pour une entrée donnée, la sortie du neurone RBF est la hauteur de la gaussienne en ce point. La fonction gaussienne permet aux neurones de ne répondre qu'à une petite région de l'espace d'entrée, région sur laquelle la gaussienne est centrée. La sortie du réseau est une combinaison linéaire des sorties des neurones RBF multipliés par le poids de leur connexion respective.

Il y a 4 paramètres principaux à régler dans un réseau RBF :

1. Le nombre de neurones RBF (nombre de neurones dans l'unique couche cachée).
2. La position des centres des gaussiennes de chacun des neurones.
3. La largeur de ces gaussiennes.
4. Le poids des connexions entre les neurones RBF et le(s) neurone(s) de sortie.

Toute modification d'un de ces paramètres entraîne directement un changement du comportement du réseau.

2.3. Comparaison des réseaux de neurones MLP / RBF

Il existe plusieurs points de différence entre ces deux grandes familles de réseaux de neurones. Une différence importante est qu'un MLP est la plupart des temps un réseau Multi-couches, alors que chaque réseau de neurones RBF consiste seulement en une couche caché avec des neurones de fonction à bases radiales. Dans une implémentation typique, les nœuds cachés et les nœuds de sortie d'un réseau de neurones MLP partagent un modèle commun de calcul neuronal, alors que les neurones de la couche cachée d'un réseau de neurones RBF jouent un rôle totalement différent comparés aux nœuds de la couche de sortie qui sont la plupart des temps linéaires et exécutant une combinaison linéaire des réponses de neurones de la couche cachée. En plus, la nature gaussienne des fonctions d'activation des neurones RBF contiennent des réponses aux régions locales (approximateurs locaux) de l'espace de sortie.

En revanche, chaque neurone MLP peut répondre par une valeur qui appartient à n'importe quel point de l'espace de sortie (approximateurs globaux). Cette localité qui est observée sur les principales réponses du RBF conduit à un nombre important de neurones et c'est la raison théorique pour laquelle des petits réseaux de neurones MLP sont plus adaptables aux problèmes spécifiques comparés aux réseaux de neurones RBF [Papaioannou et al., 2006].

3. Modèle de classification des textes arabes

3.1. Remplissage de la Matrice Document -Terme

Cette section discute comment le modèle de réseaux de neurones proposé est utilisé pour la classification des textes arabes. Le traitement par la méthode des réseaux de neurones implique trois phases principales, à savoir le pré-traitement de données, l'apprentissage et le test, comme il est représenté dans Figure 6.3. Dans notre cas, la phase d'extraction d'attributs fait référence au pré-traitement des données en utilisant la méthode de décomposition en valeurs singulières (SVD), détaillé ci-dessous. Pour l'ensemble d'apprentissage, une fois nous obtenons les attributs sélectionnés, nous les introduisons dans le réseau de neurones pour générer en sortie un classificateur du texte. Pour chaque texte de l'ensemble de test, nous utilisons le classificateur pour vérifier l'efficacité du modèle des réseaux de neurones.

Avant que nous embarquions dans la description du processus de classification, nous devons introduire la matrice Document-Terme (D-T) comme proposé par [Salton & Buckley, 1988]. La matrice (D-T) décrit la fréquence des termes qui appartiennent à une collection donné de documents. Dans cette matrice, chaque terme utilisé dans un texte de la collection est représenté par une colonne et chaque ligne correspond à un document. Les éléments a_{ij} de la matrice (D-T) donnent les fréquences de termes utilisés dans les textes. Il existe plusieurs formules pour déterminer la valeur que devrait prendre chaque entrée de la matrice, la formule « *Term Frequency Inverse Document Frequency* » (TF-IDF) est la plus citée dans ce domaine.

Le processus de la classification comporte plusieurs étapes. Premièrement, tous les textes sont transformés en une matrice Document-Terme (D-T). Deuxièmement, la dimensionnalité est réduite en se basant sur la SVD pour la transformation de la matrice (D-T). Troisièmement, la matrice réduite $(D\text{-}T)^r$ est divisée en deux parties: ensemble d'apprentissage et ensemble de test. Généralement, l'ensemble de test représente le un tiers du corpus entier. L'ensemble d'apprentissage est alors utilisé pour l'entrainement de modèle réseau de neurone. Finalement, les textes de l'ensemble de test sont présentés au classificateur réseau de neurones pour l'assignation des catégories déjà apprises.

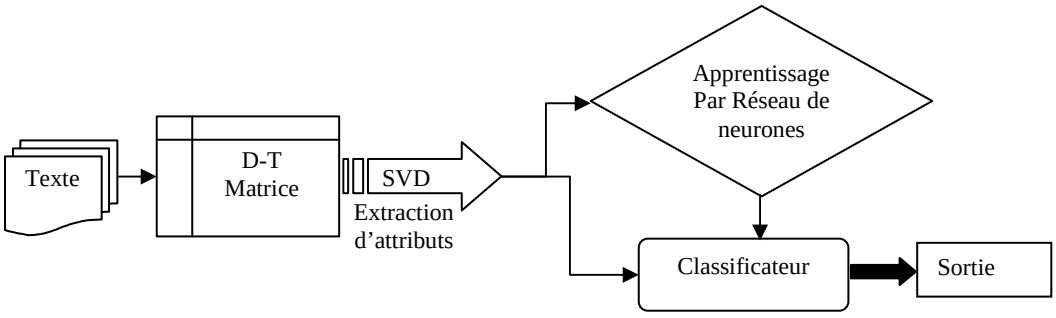

Fig. 6.3. Vue d'ensemble du classificateur de texte neuronal basé sur la technique SVD.

3.2. Pré-traitement

3.2.1. Phases initiales

Un document textuel dans un système de classification de texte passe par les étapes suivantes : conversion des documents, élimination des mots vides et lemmatisation [Larkey et al. 2002] [Syiam et al. 2006] Après avoir appliqué ces routines de pré-traitement, le document passe par un processus d'indexation qui consiste en trois phases majeures: construction du super vecteur, sélection des termes et/ou réduction de dimensionnalité et pondération [Liu et al., 2003]. Finalement, le système de classification est construit en apprenant les caractéristiques de chaque catégorie dans l'ensemble d'apprentissage. Une fois le système est construit, son efficacité peut être évaluée en l'appliquant lui-même sur un autre ensemble de test et en

vérifiant le degré de correspondance entre les décisions prises par le système de classification et celles encodés dans les catégories du corpus.

3.2.2. La décomposition en valeur singulière

La décomposition en valeur singulière (Singular Value Decomposition ou SVD) produit généralement un nouvel espace de représentation d'observations qui prend en considération les descripteurs initiaux en conservant la proximité entre les exemples de base [Rakotomalala & Mhamdi, 2006]. Ces nouveaux attributs connus comme "facteurs" ou "variables latentes" ont plusieurs propriétés très avantageuses [Hastie et al., 2001]:

a) Leur interprétation permet très souvent de détecter des modèles dans l'espace initial.

b) Un nombre très réduit de ces facteurs permet de restaurer les informations contenues dans les données.

c) Les nouveaux attributs forment une base orthogonale qui permet aux algorithmes d'apprentissage tels que l'analyse discriminante linéaire de bien fonctionner.

Ce processus est souvent utilisé dans l'analyse des données séquentielles [Wall et al., 2003] ou dans la recherche d'informations [Husbands et al., 2000] qui sont deux champs où le nombre initial de descripteurs est très grands d'où la réduction de dimensionnalité devient cruciale avant toute analyse de données. Pour la décomposition en valeurs singulières, le système est défini par une matrice de co-occurrence *Document-Mot* dans laquelle chaque ligne représente un document et chaque colonne représente un mot. Les éléments de la matrice contiennent les fréquences avec lesquelles les mots paraissent dans les documents. Regardant cette matrice, on trouvera que les documents sont représentés comme des vecteurs dans un espace de très haute dimensionnalité où les axes orthonormés représentent les mots du vocabulaire. Considérons ces vecteurs de mots c.-à-d. les colonnes de la matrice de co-occurrence *Document-Mot*, si nous multiplions la matrice de co-occurrence par son transposée, nous obtenons la matrice de corrélation des mots. Cette matrice indique combien les mots sont reliés entre eux et combien ils portent une information similaire. Si une décomposition en valeurs propres est exécutée sur cette matrice, les axes d'informations les plus significatives peuvent être choisies et les mots peuvent être projetés dans un sous-espace sémantique. Exécuter la décomposition en valeurs propres de la matrice de corrélation est équivalent à exécuter la décomposition en valeurs singulières de la matrice de co-occurrence. Les vecteurs singuliers les plus significatifs sont alors censés définir un sous-espace sémantique latent. Les vecteurs singuliers les moins significatifs sont négligés puisque ils portent généralement du bruit et de l'information redondante.

Les différentes étapes de calcul de la décomposition en valeurs singulières sont les suivantes : premièrement, la matrice de co-occurrence *Document-Mot* notée $\underline{A}$ est calculé. Ensuite, nous appliquons la technique de décomposition en valeurs singulières (SVD) sur $\underline{A}$. $\underline{A}$ nous donne donc le produit de trois autres matrices:

$$\underline{A} = \underline{U}\,\underline{S}\,\underline{V}^T \qquad (6.3)$$

Où $\underline{U}$ et $\underline{V}$ sont les matrices de *gauche* et de *droite* des vecteurs singuliers et $\underline{S}$ est la matrice *diagonale* des valeurs singulières, c.-à-d. la racine carrée non-négative de valeurs propres de $\underline{A}.\underline{A}^T$. Les premières colonnes de $\underline{U}$ et $\underline{V}$ définissent les vecteurs propres orthonormés associés avec les valeurs propres non-nulles de $\underline{A}.\underline{A}^T$ et $\underline{A}^T.\underline{A}$ respectivement. En choisissant les n valeurs singulières les plus grandes, l'espace peut être réduit en plus en éliminant un peu du bruit dû au style et aux mots non-significatives:

$$\underline{A}_n = \underline{U}_n\,\underline{S}_n\,\underline{V}_n^T \qquad (6.4)$$

Dans cet espace n-dimensionnel le $i^{\text{ième}}$ mot w_i est encodé comme suit:

$$\underline{x}_i = \underline{u}_i\,\underline{S}_n \,/\, ||\underline{u}_i\,\underline{S}_n|| \qquad (6.5)$$

Où $\underline{u_i}\ \underline{S_n}$ représente la $i^{ième}$ ligne de la matrice $\underline{U_n}\ \underline{S_n}$.

La décomposition en valeur singulière est un processus efficaces de nettoyage de données, en sélectionnant les *p* meilleurs facteurs, nous rejetons l'information négligeable contenue dans les données. Donc, c'est possible de reconstruire une version approximative des données originales à partir des facteurs sélectionnés et des vecteurs projetés.

3.3. Apprentissage du réseau neurones

Une fois l'architecture d'un réseau de neurones choisie, il est nécessaire d'effectuer un apprentissage pour déterminer les valeurs des poids permettant à la sortie du réseau de neurones d'être aussi proche que possible de l'objectif fixé. Dans le cas d'un problème de régression, il s'agit d'approcher une fonction continue, dans le cas d'un problème de classification supervisée, il s'agit de déterminer une surface de séparation. Cet apprentissage s'effectue grâce à la minimisation d'une fonction, appelée fonction de coût, calculée à partir des exemples de la base d'apprentissage et de la sortie du réseau de neurones; cette fonction détermine l'objectif à atteindre.

L'apprentissage consiste à calculer les pondérations optimales des différentes liaisons, en utilisant un échantillon. La méthode la plus utilisée est la rétro-propagation (ou back-propagation). On entre les valeurs des cellules d'entrée et en fonction de l'erreur obtenue en sortie (le delta), on corrige les poids accordés aux pondérations. C'est un cycle qui est répété jusqu'à ce que la courbe d'erreurs du réseau ne soit croissante (il faut bien prendre garde de ne pas sur-entraîner un réseau de neurones qui deviendra alors moins performant). Une fois le réseau calculé, il faut procéder à des tests pour vérifier que le réseau réagit bien comme on le souhaite : c'est la validation. Il y a plusieurs méthodes pour faire cela : la validation croisée, le bootstrapping…etc. La méthode la plus simple étant de garder une partie de l'échantillon réservé à l'apprentissage, pour la validation et faire ainsi une validation hors-échantillon.

4. Expériences

Pour évaluer les performances de nos classificateurs neuronales, Nous avons utilisé un corpus spécifique de Traditions Prophétiques ou "Hadiths" collectés à partir de l'Encyclopédie des Traditions Prophétique (Alkutub Altissâa - "Les Neuf Livres") [Harf, 1997]. Ce corpus est caractérisé par la spécialisation de son domaine. Il inclut 453 documents distribués sur quatorze catégories (Voir Chapitre IV, Table 1). Notre ensemble de données contient 5743 termes de 14 catégories majeures. Le nombre de mots collectés de toutes les catégories après la phase du pré-traitement est de 1065 mots. Les mots uniques sont donc identifiés et arrangés par leur fréquence d'apparition. Les mots de longueur moins de 3 octets, les caractères de ponctuation Arabes, les mots vides Arabes, les mots très fréquents et les mots peu fréquents sont supprimés de la liste. Suivant les étapes du modèle de classification des textes, nous avons appliqués un processus de lemmatisation à base de racine en utilisant le programme *Detect-Root* de Karim Darwish[1]. Nous avons utilisé la moitié des données pour tester le classificateur et la moitié pour apprendre le système de classification des textes.

4.1. Pré-traitement et données d'entrée

Le but de nos expériences est d'évaluer les performances de l'algorithme des réseaux de neurones artificiels pour la classification des textes arabes en utilisant le corpus décrit dans le paragraphe précédent.

Après l'application des processus d'élimination des mots vides et de lemmatisation sur l'ensemble des documents, nous les représenterons dans une matrice de fréquence Document-

[1] Le programme Detect-root de Karim Darwish est utilisé pour la lemmatisation, disponible sur le site: *http://www.glue.umd.edu/~kareem/research/*.

Terme *($Doc_i \times Terme_j$)*. *Doc_i* fait référence à chaque document existant dans le corpus où *$i = 1...n$*. *$Terme_j$* fait référence à chaque terme existant dans le corpus où *$j = 1...m$*. Le calcul des poids des termes x_{ij} de chaque mot w_j est réalisé en utilisant une méthode décrite par [Salton & Buckley, 1988], elle est donné par l'équation suivante:

$$X_{ij} = TF_{ij} \times idf_j \qquad (6.7)$$

Où la fréquence du document df_j représente le nombre total des documents du corpus qui contient le mot w_j. La fréquence inverse du document idf_j est égale à : $log(n/df_j)$ où n est le nombre total des documents dans le corpus [Selamat & Omatu, 2003].

4.1.1. Données d'entrée pour le réseau de neurones.

Après la fin de l'étape pré-traitement, un vocabulaire contenant tous les mots uniques du corpus est créé. Le nombre de mots distincts dans le corpus est très grand, il égal à 1065. Chaque mot dans le vocabulaire représente un vecteur d'attributs. Chaque vecteur d'attributs contient les poids Document-Termes. La haute dimensionnalité des vecteurs d'attributs en entrée du réseau de neurones n'est pas pratique, car elle cause une grande dégradation de performances pour le réseau de neurones [Selamat & Omatu, 2003]. Par conséquent, la décomposition en valeur singulière a été utilisée pour réduire le nombre des vecteurs d'attributs originaux *m=1065* en un nombre réduit de facteurs singuliers. Dans notre cas et après plusieurs expérimentations, nous avons sélectionné la valeur *d=530*, ce paramètre donne de bons résultats pour la classification des textes arabe comparée à d'autres paramètres comme entrées aux réseaux de neurones.

Pour avoir une idée sur les données après l'application de la transformation SVD, nous projetons dans la Figure 6.4, les premiers 200 vecteurs de facteurs après transformation avec les 1065 attributs de l'ensemble d'apprentissage. La réduction de la dimensionnalité permettra une visualisation 2-D des données textuelles, avec les quatorze groupes séparés du corpus (par exemple: *C1 = Faith* pour la catégorie «*Foi*» et *C14=Personnel states* pour la catégorie «*états personnels*»). La coordonnée *x* est obtenue en multipliant la première colonne de la matrice *V* (de SVD) par la matrice réduite *S*, avec seulement les premières 200 valeurs singulières. La coordonnée *y* est calculée par la multiplication de la deuxième colonne de *V* par la matrice réduite *S*, avec les deux cents facteurs SVD. Les cercles noirs sont les vecteurs de la catégorie «*Adorations*», alors que les carrés blancs sont les vecteurs de la catégorie «*Biographie*». De la Figure 6.4, nous observons que les vecteurs qui correspondent aux catégories peuvent être facilement distingués.

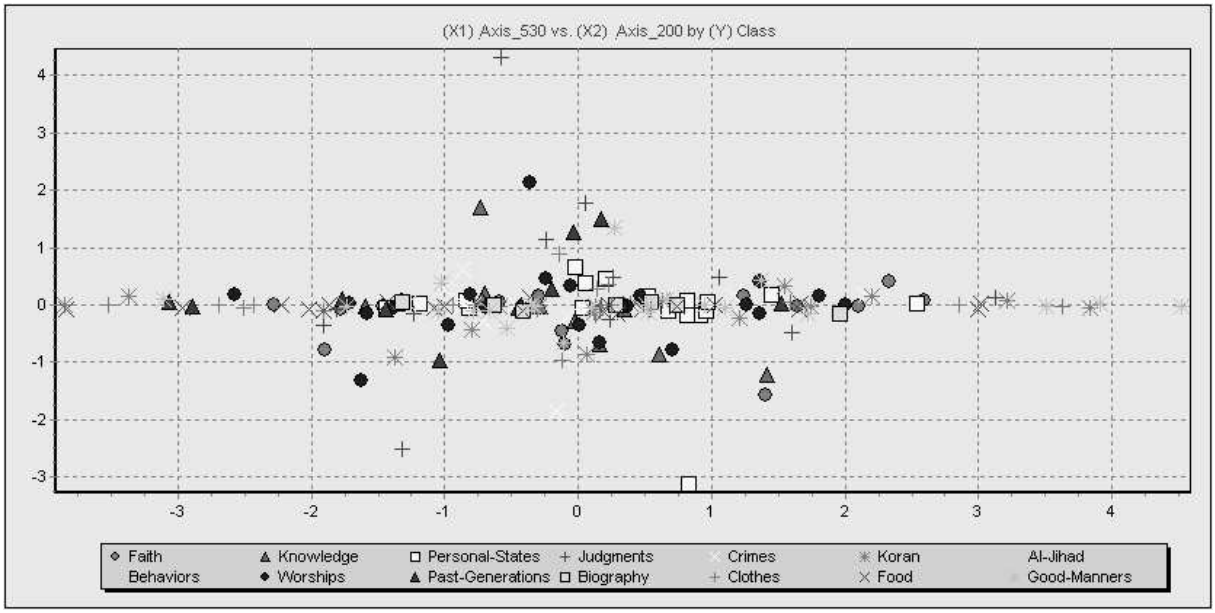

Fig. 6.4. Projection des 200 premiers facteurs du l'ensemble de 1065 attributs.

Le graphe de chargement des facteurs pour la proportion accumulée des valeurs propres est montré dans la Figure 6.5. La valeur de *d* contribue de 77.14 % des proportions des vecteurs d'attributs originaux.

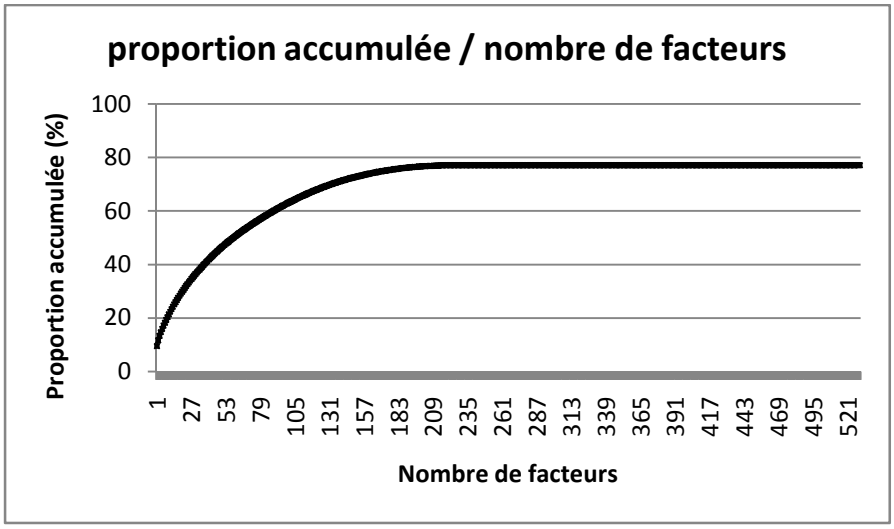

Fig. 6.5. Proportion accumulée des facteurs singuliers produite par SVD.

4.1.2. Caractérisation du réseau de neurones.

Le nombre de nœuds de la couche d'entrée (*p*) est égale à 1065, le nombre des facteurs SVD (*d*) est égale à 530. Le nombre de nœuds de la couche cachée (*q*) est égal à 30. L'approche erreur et test a été utilisée pour trouver le nombre convenable de couches cachées qui fournissent la bonne exactitude de classification en se basant sur les données d'entrée du réseau de neurones. Le nombre de couches de sortie (*r*) est égal à 14 qui correspond au nombre de catégories dans le corpus du *Hadith*. Le Tableau 6.1 montre les différents paramètres des réseaux de neurones MLP et RBF utilisés.

	Type	
	MLP	**RBF**
Architecture		
Utiliser une couche cachée	oui	Oui
Neurones dans la couche cachée	30	14
Groupe d'attributs	Non	Class
Paramètres d'apprentissage		
Taux d'apprentissage (q)	0.050	0.050
Transformation d'attributs	non	Non
Proportion de l'ensemble de validation	0.2	0.2
Critères d'arrêt		
Nombre d'itérations (I)	1000	1000
Seuil du taux d'erreur	0.001	0.001
Vérification de l'erreur de stagnation	non	non

Tableau 6.1. Les Paramètres des réseaux de neurones MLP et RBF.

4.2. Résultats et analyses

4.2.1. Erreur d'apprentissage

Pour voir comment l'exactitude s'améliore en fonction du nombre croissant d'itérations, la Figure 6.6 et la Figure 6.7 donnent le taux d'erreur pendant le processus d'apprentissage pour les modèles originaux des réseaux de neurones MLP et RBF. La Figure 6.8 et la Figure 6.9 donnent le taux d'erreur pour les modèles de réseaux de neurones MLP et RBF à base de SVD. L'erreur carrée moyenne (ECM) est mesurée pour l'apprentissage répété avec le nombre croissant d'époques.

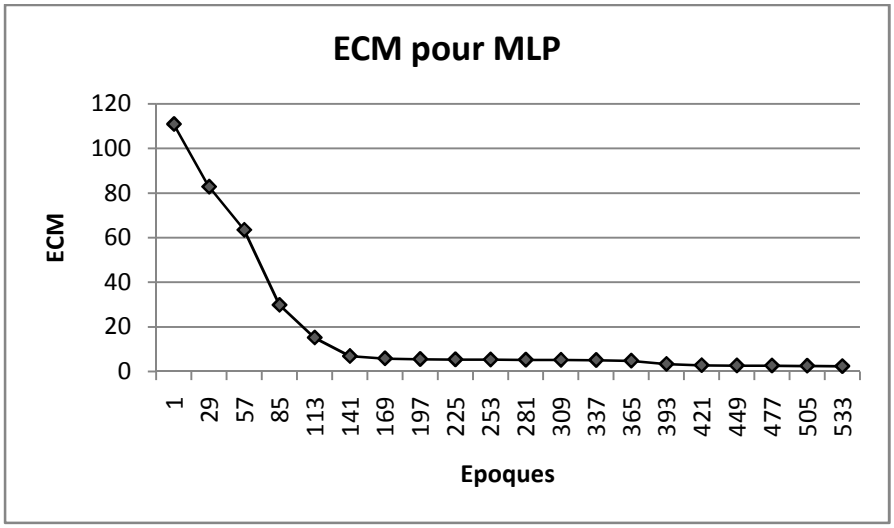

Fig. 6.6. ECM en fonction d'époques pour la classification des textes prophétiques en utilisant le modèle original du réseau de neurones MLP.

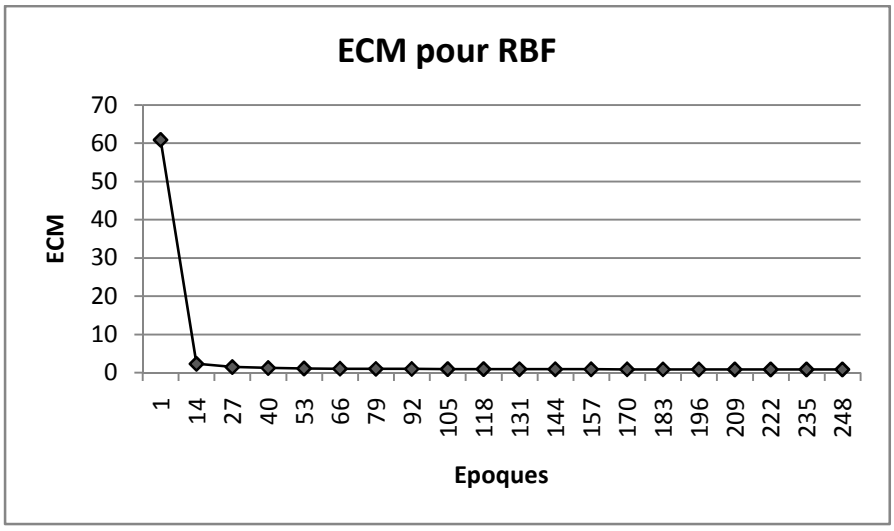

Fig. 6.7. ECM en fonction d'époques pour la classification des textes prophétiques en utilisant le modèle original du réseau de neurones RBF.

L'erreur d'apprentissage diminue d'une manière exponentielle en fonction du nombre croissant d'époques. Nous avons trouvé que l'erreur est extrêmement réduite après approximativement 141 époques pour le réseau de neurones MLP et après approximativement 14 époques pour le réseau de neurones RBF, ce qui peut être considéré comme un point assez rapide de convergence pour un système pratique.

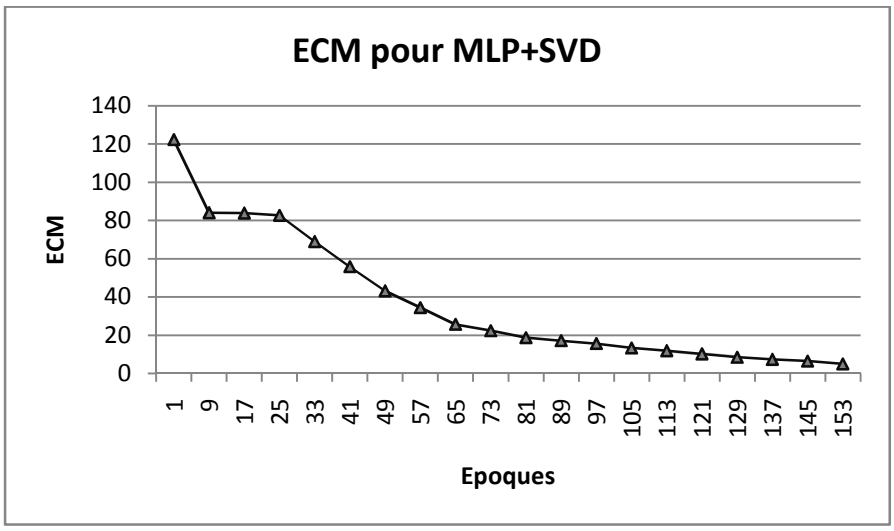

Fig. 6.8. ECM en fonction d'époques pour la classification des textes prophétiques en utilisant le modèle à base SVD du réseau de neurones MLP.

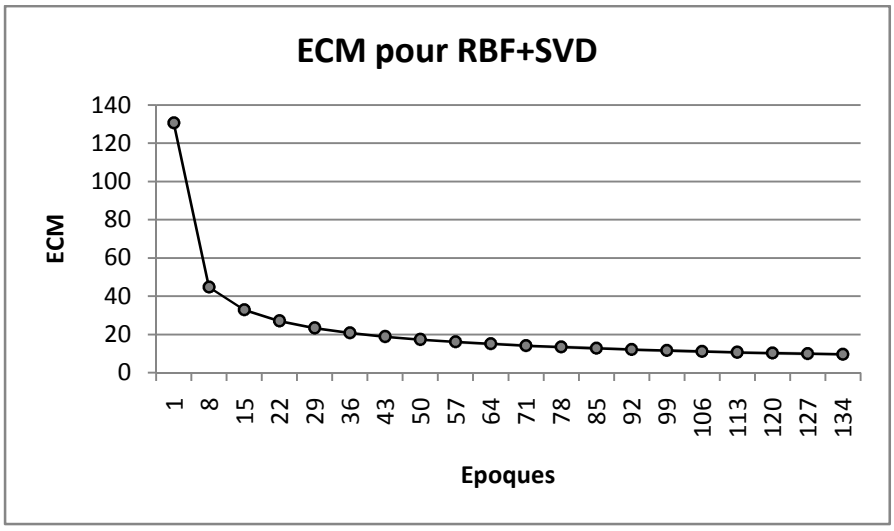

Fig. 6.9. ECM en fonction d'époques pour la classification des textes prophétiques en utilisant le modèle à base SVD du réseau de neurones RBF.

Pour les réseaux de neurones à base de SVD, nous arrêterons l'apprentissage après 179 époques pour le réseau de neurones MLP et après 159 époques pour le réseau de neurones RBF. La raison pour laquelle nous choisissons ces valeurs comme critères d'arrêt est que d'après nos expériences, nous avons trouvé que ces paramètres permettent d'obtenir un bon compromis entre l'exactitude et l'efficacité du système de classification.

4.2.2. Résultats de classification

Dans nos expériences, nous comparons les performances du système en variant le nombre de dimension *k*, le nombre de nœuds d'entrée des réseaux de neurones est égal à la dimension des vecteurs *documents*. Pour les modèles des réseaux de neurones à base de SVD, les dimensions varient entre 20 et 530, et pour les modèles originaux des réseaux de neurones à base de l'espace vectoriel, le nombre de vecteurs est égale 1065.

Les performances de notre système de classification sont évaluées par les mesures de rappel, précision et macro-moyenne de la mesure F_1. Les valeurs de la macro-moyenne de la mesure F_1 sont basées sur les valeurs de précision et rappel.

La technique SVD a été testée pour sa capacité d'encoder les informations importantes d'un espace d'attributs de haute dimensionnalité en un autre espace d'attributs réduit de dimension très inférieure. Nous avons utilisé une méthode semblable pour les tests en variant et en réduisant la dimensionnalité. Dans les figures qui suivent, Nous présentons respectivement le rappel moyen, la précision moyenne et la macro-moyenne de la mesure F_1 en fonction du nombre de dimensions pour les réseaux de neurones MLP et le RBF.

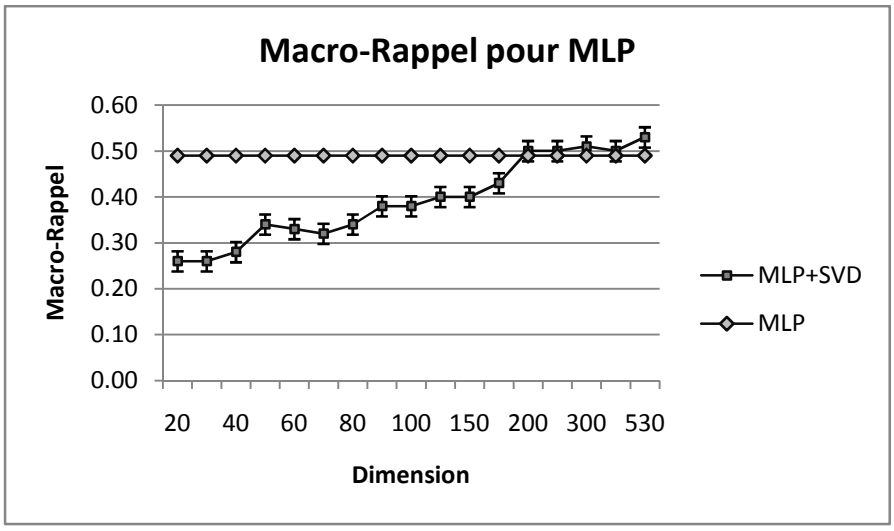

Fig. 6.10. Macro-Rappel en fonction du nombre de dimension pour les deux modèles du réseau de neurones MLP.

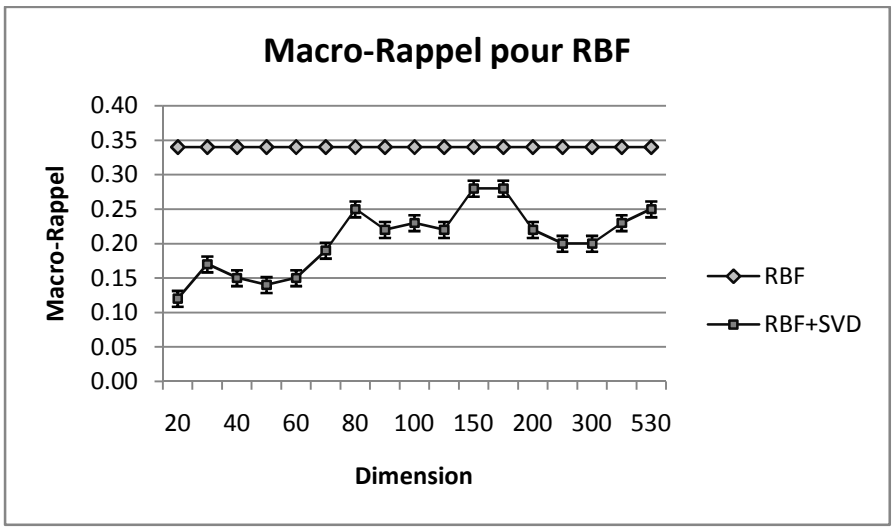

Fig. 6.11. Macro-Rappel en fonction du nombre de dimension pour les deux modèles du réseau de neurones RBF.

La Figure 6.10 représente la comparaison des valeurs du rappel moyen pour les deux modèles MLP original et MLP à base de SVD, la dimension varie entre 20 à 530. La valeur du rappel moyen pour MLP est de 49% alors que les valeurs du rappel moyen pour MLP à base de SVD appartiennent à l'intervalle [26%-53%]. Les meilleures valeurs du rappel moyen pour MLP+SVD sont 50%, 50%, 51%, 50% et 53% pour les dimensions 200, 250, 300, 400 et 530 respectivement. La Figure 6.11 représente la comparaison des valeurs du rappel moyen pour les deux modèles RBF original et RBF à base de SVD. Les dimensions de SVD sont entre 20 et 530. La valeur du rappel moyen pour RBF est de 34% et les valeurs du rappel moyen pour RBF à base de SVD sont dans l'intervalle [12%-25%]. La plus grande valeur du rappel moyenne pour RBF+SVD est de 28% pour les deux dimensions 150 et 175.

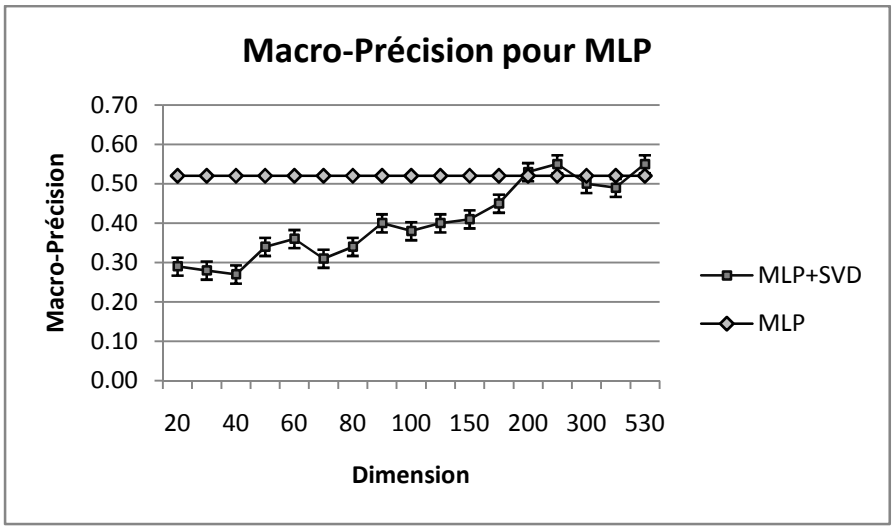

Fig. 6.12. Macro-Précision en fonction du nombre de dimension pour les deux modèles du réseau de neurones MLP.

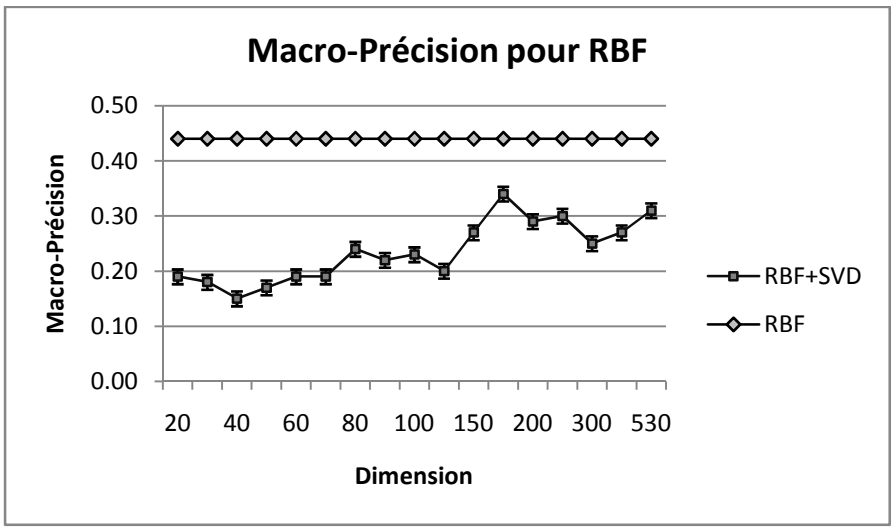

Fig. 6.13. Macro-Précision en fonction du nombre de dimension pour les deux modèles du réseau de neurones RBF.

La Figure 6.12 représente la comparaison des valeurs de précision moyenne pour les deux modèles MLP original et MLP à base de SVD. La valeur de précision moyenne pour MLP est de 52%. Les valeurs de précision moyenne pour MLP+SVD sont dans l'intervalle [29%-55%]. Les meilleures valeurs de précision moyennes pour MLP+SVD sont 53%, 55%, 55% pour les dimensions 200, 250 et 530 respectivement. La Figure 6.13 représente la comparaison des valeurs de précision moyenne pour les deux modèles RBF et RBF+SVD. La précision moyenne pour RBF est de 44% et les valeurs de précision moyennes pour RBF+SVD sont entre 19% et 31%. La plus haute valeur est de 34% pour la dimension 175.

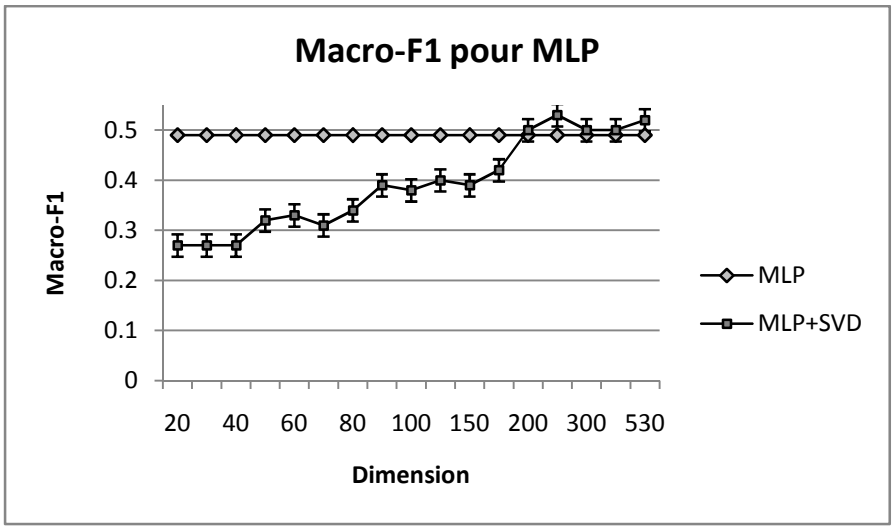

Fig. 6.14. Macro-F_1 en fonction de dimension pour MLP.

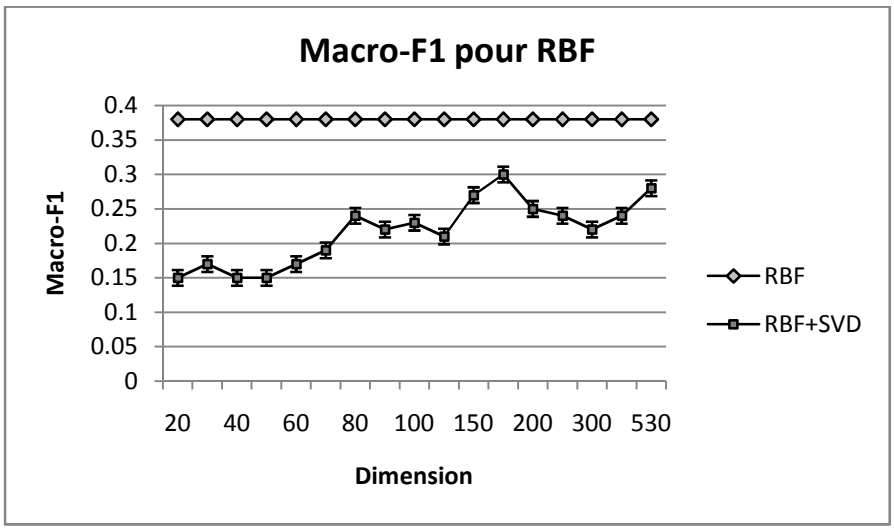

Fig. 6.15. Macro-F_1 en fonction de dimension pour RBF.

La Figure 6.14 et La Figure 6.15 montrent l'effet du nombre des facteurs d'entrée sur les performances de classification des réseaux de neurones avec un nombre de dimension égal à 530. En général, les valeurs de la mesure Macro-F_1 augmentent en fonction du nombre croissant des facteurs d'entrée. La valeur de la mesure Macro-F_1 avec tous les attributs (1065) est de 49% pour MLP et de 38% pour RBF. La valeur de la mesure Macro-F_1 en utilisant SVD (530 dimensions) est entre 22% et 53% pour MLP et entre 15% et 30% pour RBF.

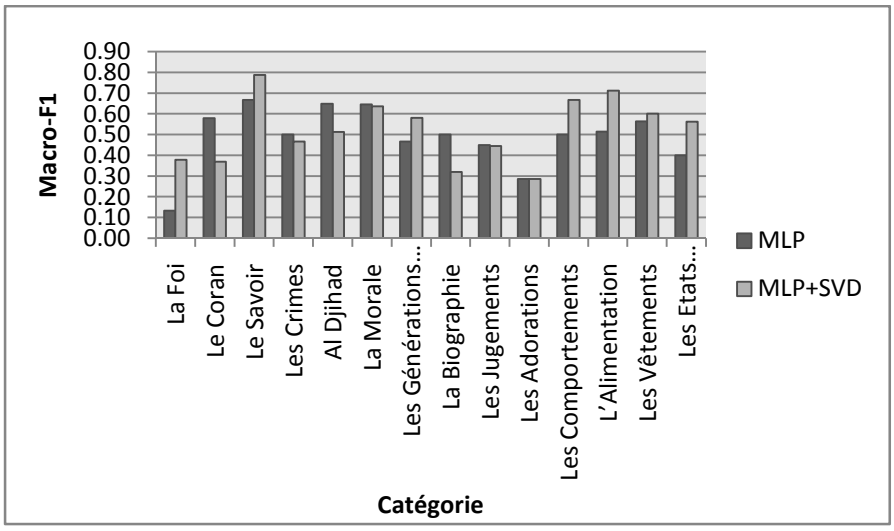

Fig. 6.16. Comparaison des résultats de classification pour MLP.

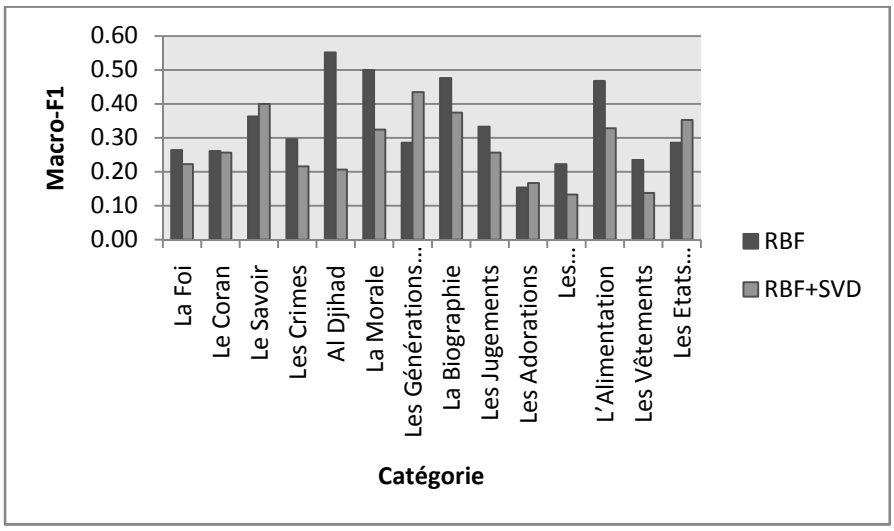

Fig. 6.17. Comparaison des résultats de classification pour RBF.

La Figure 6.16 et la Figure 6.17 montrent les valeurs de la mesure Macro-F_1 pour chaque catégorie du corpus. De la Figure 6.16, on peut voir que la mesure F_1 atteint ses plus hautes valeurs 67% et 79% pour la catégorie (Savoir) pour les deux modèles MLP et MLP +SVD, et les plus basses valeurs de 13% pour la catégorie (Foi) pour MLP et 29% pour la catégorie (Adorations) pour MLP+SVD. De la Figure 6.17 nous pouvons voir que pour RBF, la plus haute valeur pour la mesure F_1 est 55% pour la catégorie (Al-Jihad), et la valeur la plus basse est 15% pour la catégorie (Adorations). Pour RBF+SVD, la plus haute valeur est 43% pour la catégorie (Générations antérieures) et la plus basse valeur est de 13% pour la catégorie (Comportements).

4.3. Discussion

Des expériences précédentes, nous pouvons remarqués que le classificateur des textes arabes à base des réseaux de neurones est capable d'obtenir des valeurs raisonnables pour les mesures de rappel, de précision et F_1. Les résultats montrent que les réseaux de neurones avec un apprentissage basé sur l'algorithme de «Back-Propagation» sont efficaces pour une telle tâche de catégorisation de texte. Pour le réseau de neurones MLP, nous pouvons remarqués que les valeurs des mesures Macro-rappel, Macro-précision et Macro-F_1 pour la classification des textes arabes augmentent et devient plus stables avec l'utilisation de SVD comme une technique d'extraction d'attributs. Avec un nombre croissant de facteurs, le taux d'exactitude augmente

progressivement et à la dimension 530 le réseau de neurones MLP obtient sa meilleure performance de 52%. Pour le RBF, les résultats montrent que la performance de classification est très instable quand le nombre de facteurs augmente, le modèle original du RBF est donc plus performant que le modèle RBF à base de SVD.

Les résultats montrent aussi que l'utilisation de SVD est efficace pour réduire la taille de l'espace d'attributs de 1065 à 530, et maintenir en même temps des valeurs moyennes pour les mesures de précision, rappel et F1. Ce qui représente un taux de la réduction de 50%. En outre, nous trouvons aussi que les nouveaux attributs sélectionnés par SVD peuvent décrire les caractéristiques de textes arabes mieux que les attributs originaux. Il est donc montré qu'ajoutant des attributs insignifiants n'améliore nécessairement pas la performance de classification.

La dimension réduite des vecteurs a grandement contribuée pour la baisse du temps de calcul de la phase d'apprentissage pour les deux modèles des réseaux de neurones à rétro-propagation du gradient de l'erreur. Le temps de calcul pour le réseau de neurones MLP à base de SVD est inferieur à celui du réseau de neurones MLP original. Pour un nombre de dimensions égal à 530, le temps de calcul pour MLP à base de SVD est de 52.4 s. Plus rapide que celui du modèle MLP original qui est égal à 191.7 s. La même remarque peut être faite pour RBF, puisque le temps de calcul du modèle RBF à base de SVD est égal à 984.5 s comparé à celui du RBF original avec un temps de calcul égal à 3157.9 s.

Comparant les deux architectures des réseaux de neurones MLP et RBF, il parait que MLP dépasse RBF pour les deux modèles (original et à base de SVD). Pour cette raison, il est estimé que le réseau de neurones MLP est plus approprié que le réseau de neurones RBF pour la catégorisation des textes arabe.

5. Conclusion

Dans ce chapitre, nous avons développé un modèle de classification des textes arabes qui utilise les réseaux de neurones et la méthode de décomposition en valeur singulière (SVD). Les techniques d'apprentissage utilisées sont basées sur les réseaux de neurones MLP et RBF. On montre que l'introduction de la méthode de décomposition en valeur singulière améliore les performances de classification. Comme résultat tangible, la dimension réduite des vecteurs a aussi diminué le temps de calcul de la phase d'apprentissage pour les deux réseaux de neurones MLP et RBF. Un de leurs avantages est qu'ils exigent un minimum de calcul et de ressources de stockage, ce qui les rend idéaux pour la classification des textes arabe. Les expériences sur le corpus arabe "Hadith" ont montré que le modèle MLP est efficace pour la représentation et la classification des documents arabes. Cette étude conclut que le MLP surclasse le RBF pour les deux modèles (original et à base de SVD). Les résultats indiquent aussi que le même modèle avec la méthode SVD est plus capable de capturer les rapports non-linéaires entre les vecteurs des documents en entrée et les catégories des documents en sortie.

D'autres méthodes de sélection et de réduction d'attributs seront considérées dans nos futurs travaux. Il serait aussi utile d'utiliser une plus grande collection de données pour améliorer la capacité d'apprentissage de langage pour les modèles utilisés. Finalement, nous prévoyons de mener plus de comparaisons avec d'autres algorithmes d'apprentissage employés dans la littérature pour la catégorisation du texte tel que SVM, KNN, Réseaux Bayésien et algorithme de Boosting.