Annexe A : Système de fouille de textes dans les corpus traditions prophétiques

1. Les Interfaces du logiciel de fouille des textes prophétiques

Ces interfaces concernent les processus suivants : la formulation des requêtes de l'utilisateur, le traitement des requêtes, la recherche des documents, et la présentation des résultats à l'utilisateur.

1.1. L'interface principale

Ce composant fournira une interface multi-modale pour la recherche de l'information permettant une interaction à travers les différents modules du logiciel. Cette Interface est employée pour la formulation des requêtes, la présentation des résultats, et l'enrichissement des ces requêtes dans la lumière de ces résultats.

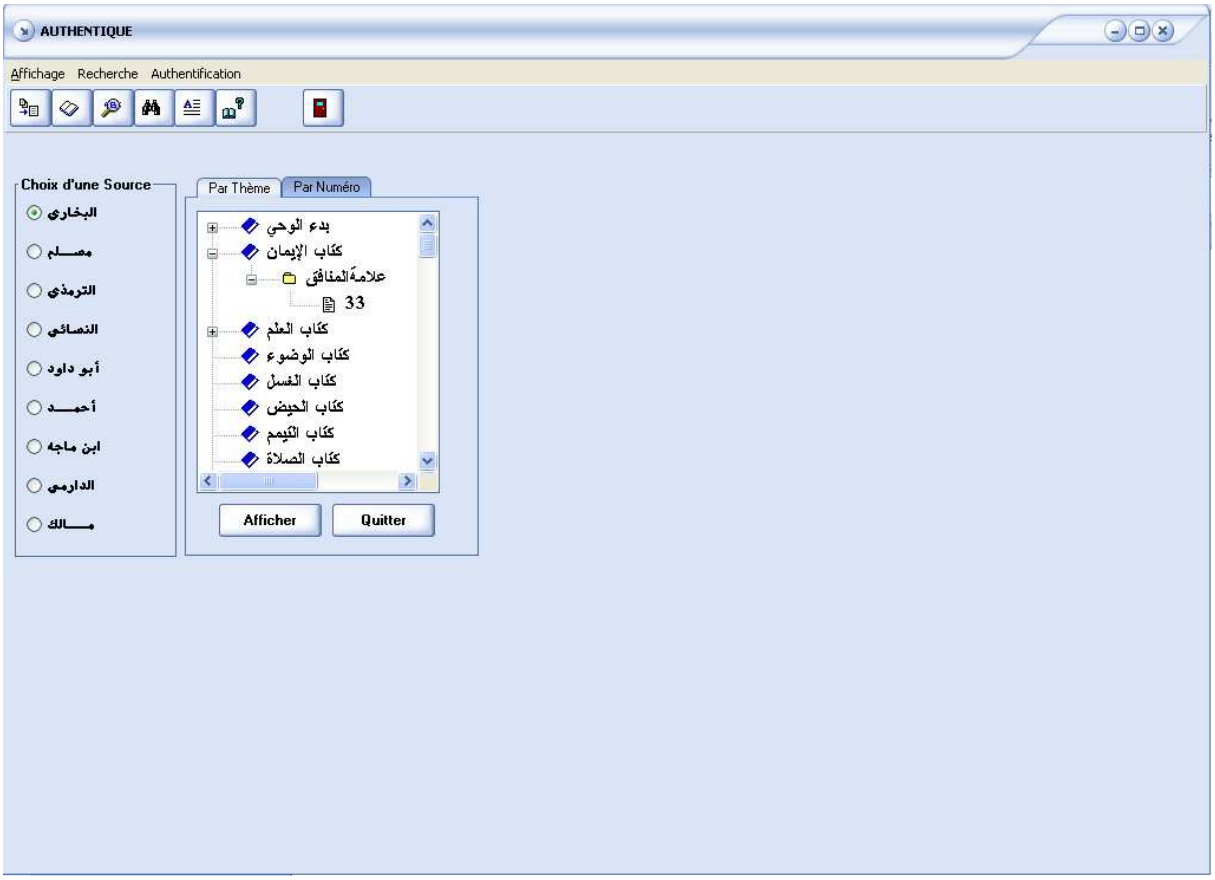

Fig. A.1. Interface principale du système.

1.2. Interface d'indexation

Cette interface permet de créer la représentation (Matrice *Doc-Term ou index*) pour les documents. Elle consiste à analyser les documents afin d'extraire un ensemble de mots clés servant comme descripteurs des documents, qui sont facilement manipulable par un système de recherche d'information.

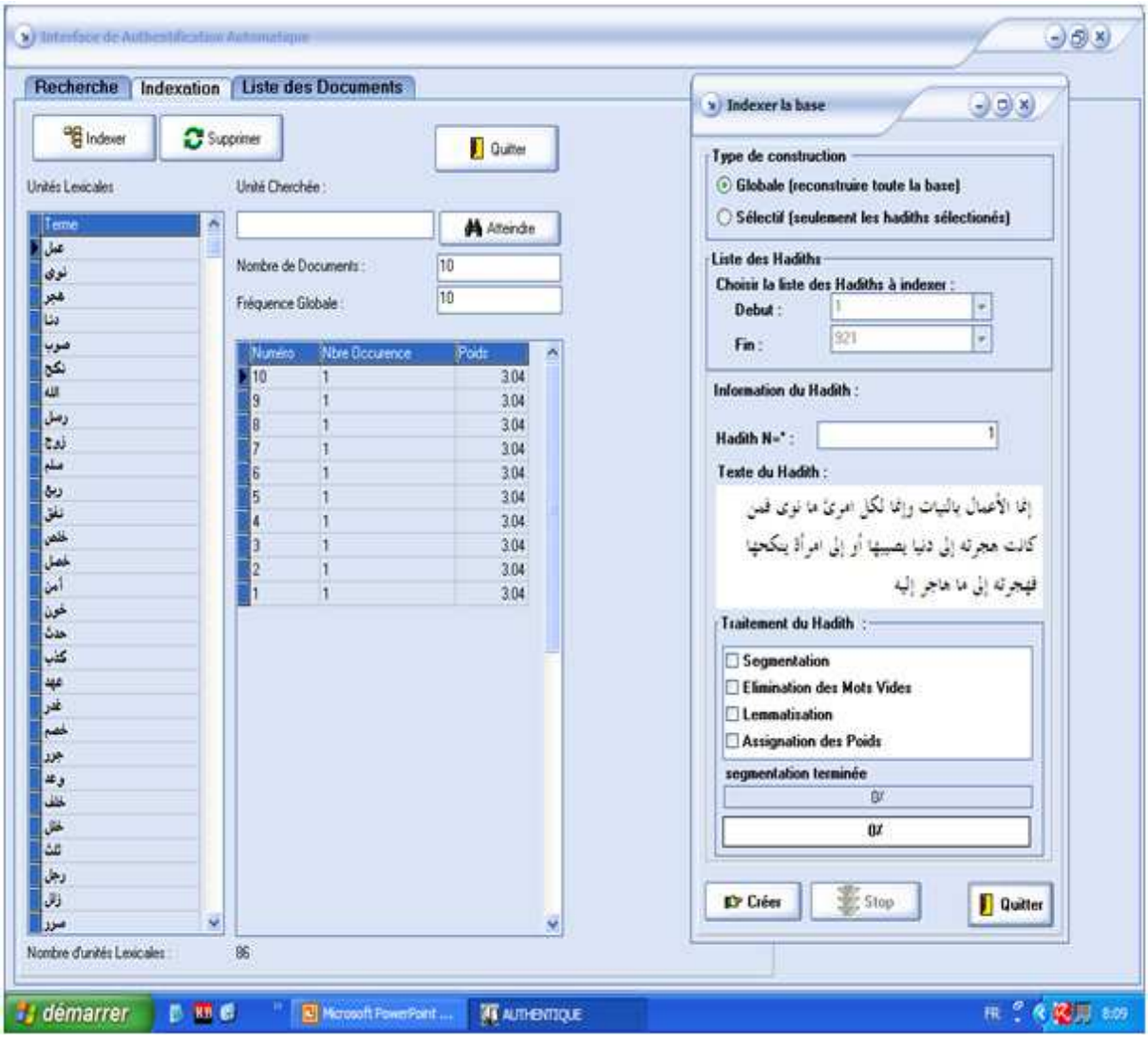

Fig. A.2. Interface d'indexation.

1.3. Interface de recherche

Cette interface fournit le cœur des fonctionnalités du système de recherche d'information. A partir de cette interface, on peut accéder à l'index prétraité pour obtenir des documents similaires la requête de l'utilisateur. Cette fonctionnalité sera basée sur un modèle de recherche vectoriel.

Fig. A.3. Interface de recherche.

1.4. Retour de pertinence:

Cette interface est responsable de la classification de l'ensemble des résultats (identification des documents) reçus du composant moteur de recherche sur la base de leur pertinence à la requête initiale. Les documents jugés pertinents seront utilisés pour reformuler une nouvelle requête en modifiant les poids des termes qui apparaissent dans ces documents. La liste des documents pertinents sera donc enrichie et l'ensemble des résultats sera envoyé à l'interface utilisateur pour les représenter à l'utilisateur.

Fig. A.4. Interface de représentation des résultats.

Annexe B : Système de classification des textes prophétiques par arbres de décision

Cette annexe présente un aperçu général du fonctionnement de notre système de classification des textes prophétiques par arbres de décision. Ce système est basé sur l'implémentation de la version ID3 de l'algorithme Arbre de Décisions.

1. Construction de l'arbre de décision

La première étape d'implémentation de l'algorithme arbres de décision consiste à la construction du l'arbre lui-même. Les nœuds internes de l'arbre sont les tests, les feuilles sont les catégories, comme il est montré dans la figure B.1.

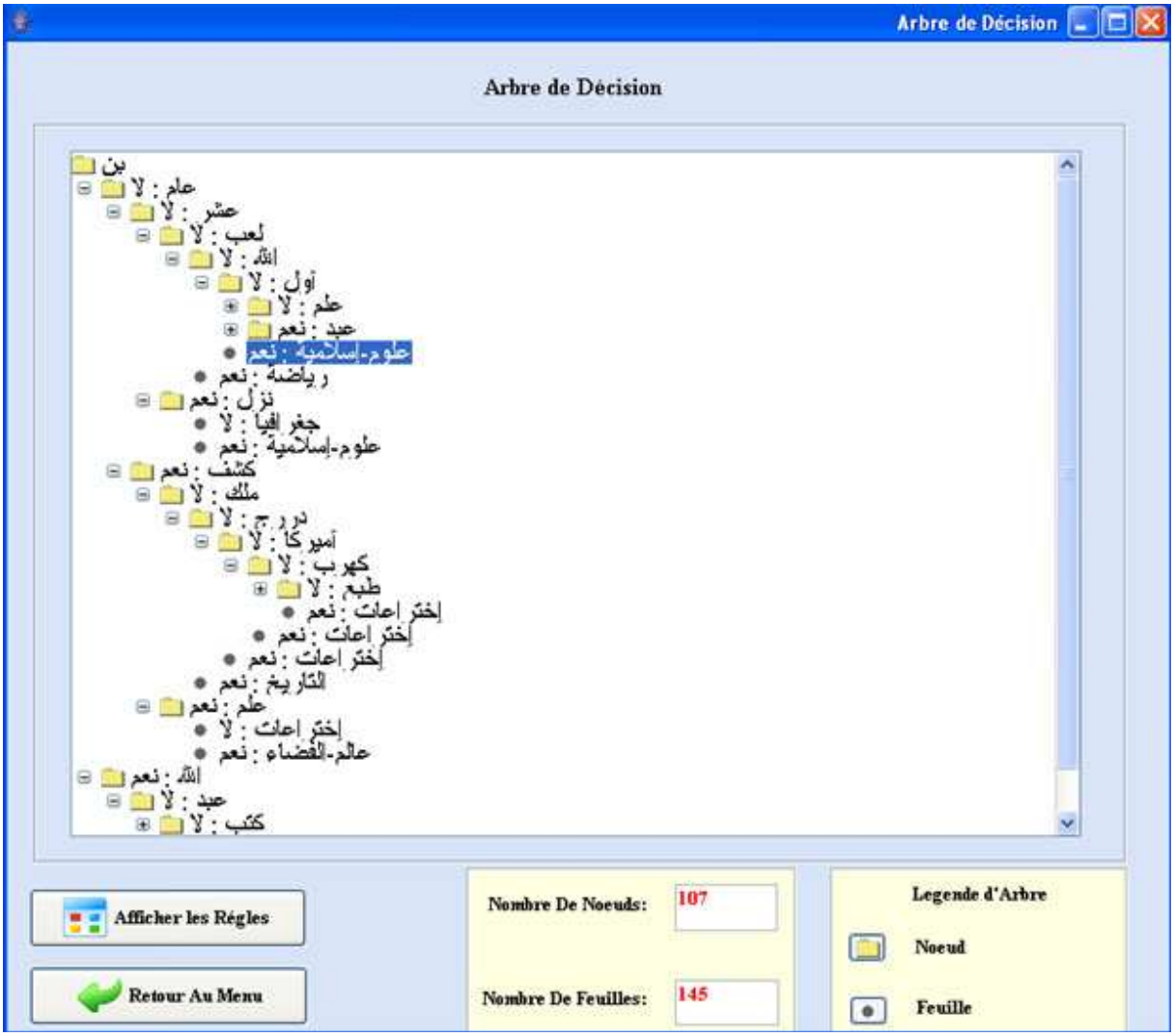

Fig. B.1. Construction de l'arbre de décision.

2. Génération des règles de classification

L'arbre de décision est une méthode plus claire de représentation des données. Pour rendre l'arbre plus lisible, un ensemble de règles de la forme " Si Expression Alors Conclusion" est alors généré à partir de cet arbre comme il est montré dans la figure B.2.

Fig. B.2. Les règles générées à partir de l'arbre de décision.

3. Classement des documents de test

L'ensemble des règles extraites sera utilisé pour le classement d'un ensemble de test et les résultats obtenues seront utilisés pour l'évaluation et la validation de notre. L'interface qui permet cette fonctionnalité est montrée dans la figure B.3 ci-dessous.

Fig. B.3. Le classement de corpus de test à partir des règles

Annexe C : Le Système *ArabTiling* pour la segmentation thématique des textes prophétiques

Cette annexe présente un aperçut général du fonctionnement de notre système de segmentation thématique *ArabTiling*. Ce système est basé sur l'implémentation et l'intégration de l'algorithme *TextTiling* dans un moteur de recherche d'information en langue arabe dans le but d'améliorer les performances de ce dernier. Ce système est développé en utilisant le langage de programmation *Java* sous la plate forme de développement (*NetBeans 5.5*).

1. Présentation des interfaces du système (*ArabTiling*)

1.1. Interface principale:

La première interface du système donne à l'utilisateur la possibilité de lancer une recherche, en introduisant sa requête en langage naturel comme il est mentionné dans la figure C.1.

Fig. C.1. Interface de recherche du système *ArabTiling*.

1.2. Interface de présentation des résultats de recherche

La deuxième interface permet de visualiser les résultats de la recherche ordonnés en fonction de leurs scores de similarité comme il est mentionné dans la figure C.2 ci-dessous.

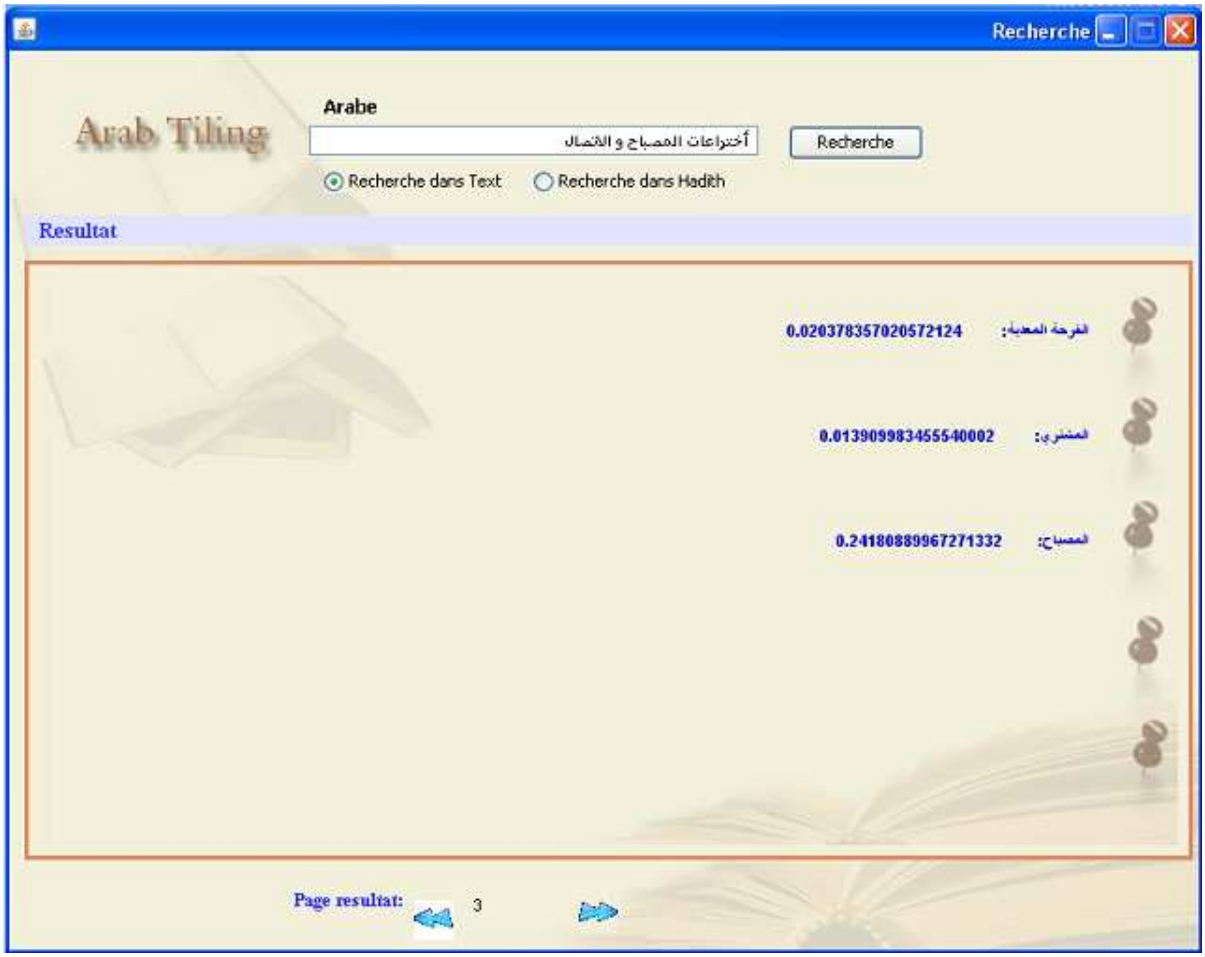

Fig. C.2. Présentation des résultats de recherche.

1.3. Interface de consultation des documents

La troisième interface est réservée pour la consultation d'un document. Cette interface nous donne aussi la possibilité de lancer le processus de segmentation thématique pour ce document. Ceci permettra de mieux raffiner la recherche en ne présentons à l'utilisateur que le ou les segments pertinents par rapport à la requête initiale.

Fig. C.3. Consultation des documents

1.4. Interface des segments pertinents

Cette interface permet de visualiser la liste des segments pertinents. A partir de cette interface, l'utilisateur peut consulter les parties pertinentes du document sans qu'il soit obligé de faire le passage de tout le document. Le bouton synthèse permettra de lancer l'interface de synthèse pour le processus de segmentation thématique. Cette fonctionnalité donne plus du détail pour le processus de segmentation.

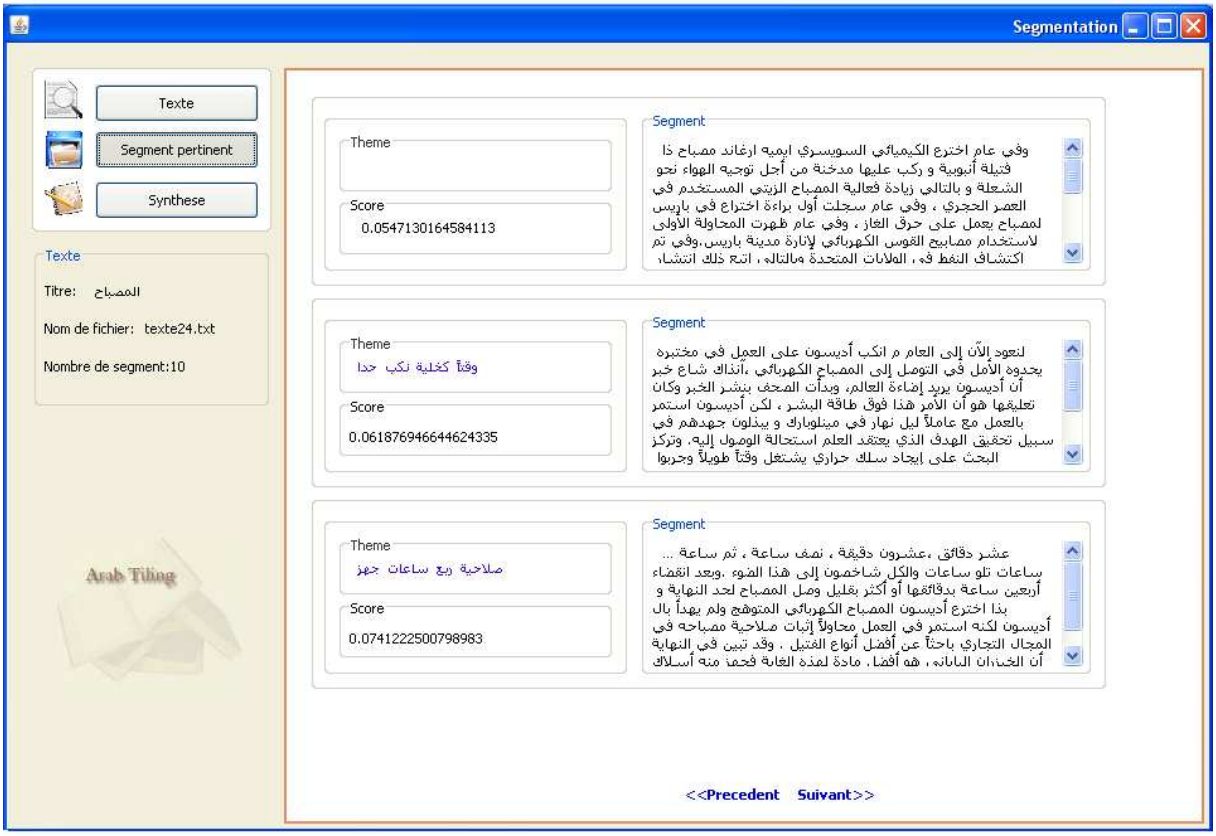

Fig. C.4. Interface des segments pertinents.

1.5. Présentation des interfaces de segmentation

Le processus de segmentation thématique est le noyau de base du système *ArabTiling*. On peut lancer la segmentation de deux manières différentes. La première a été déjà discutée dans la section 1.3. Elle permet à l'utilisateur une fois qu'il a visualisé les résultats de sa recherche, de lancer ce processus pour chaque document figurant dans la liste des résultats. La deuxième manière donne à l'utilisateur la possibilité de segmenter un texte en dehors du processus de recherche.

1.5.1. Interface de calcul de similarité

Le calcul de similarité consiste à faire une comparaison entre chaque deux bloc consécutif du texte. On utilise la fonction de similarité *Cosine* dans ce calcul, le résultat est un graphe de similarité utilisé dans l'opération de calcul de cohésion.

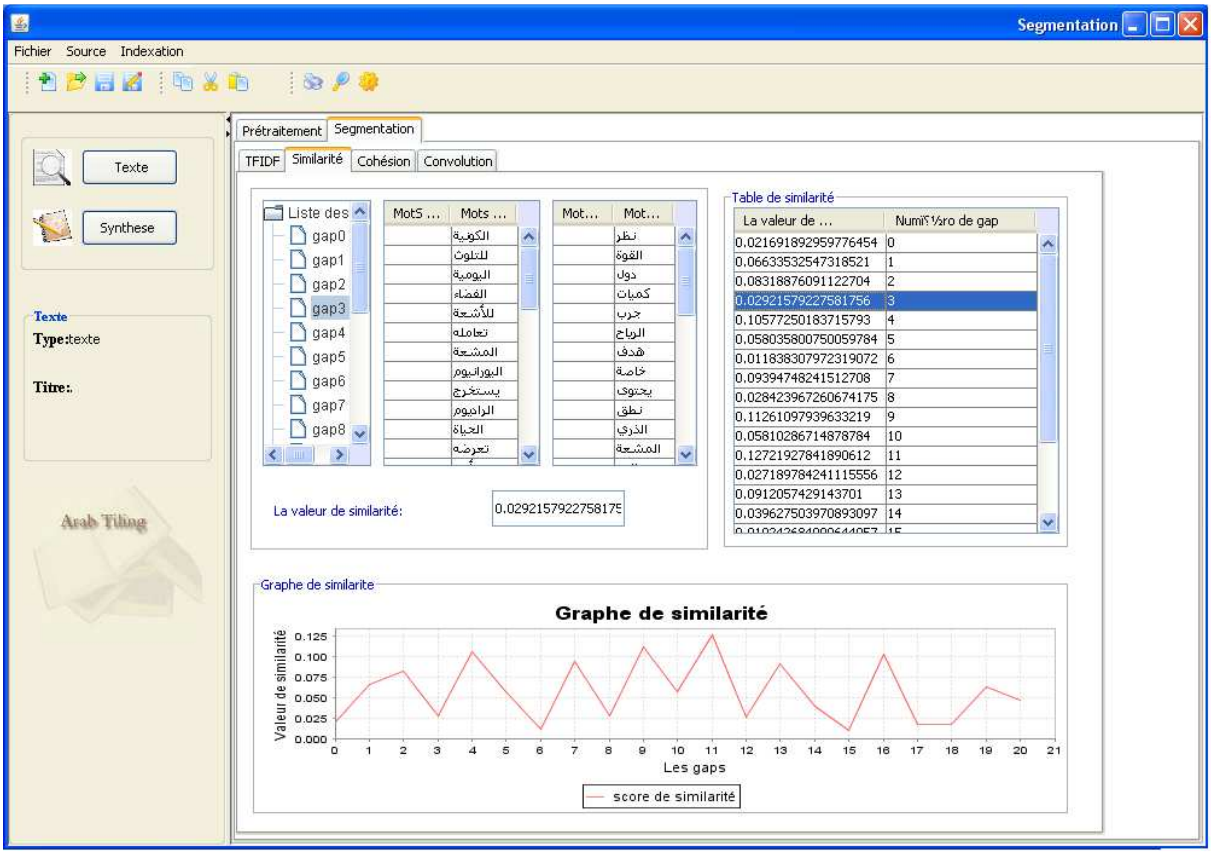

Fig. C.5. Interface de calcul de similarité.

1.5.2. Interface de calcul des scores de cohésion

A partir des scores de similarité, un score de cohésion est assigné à chaque gap (creux en fonction de similarité des gaps voisins). Les scores de cohésion sont représentés sous forme d'un graphe de cohésion.

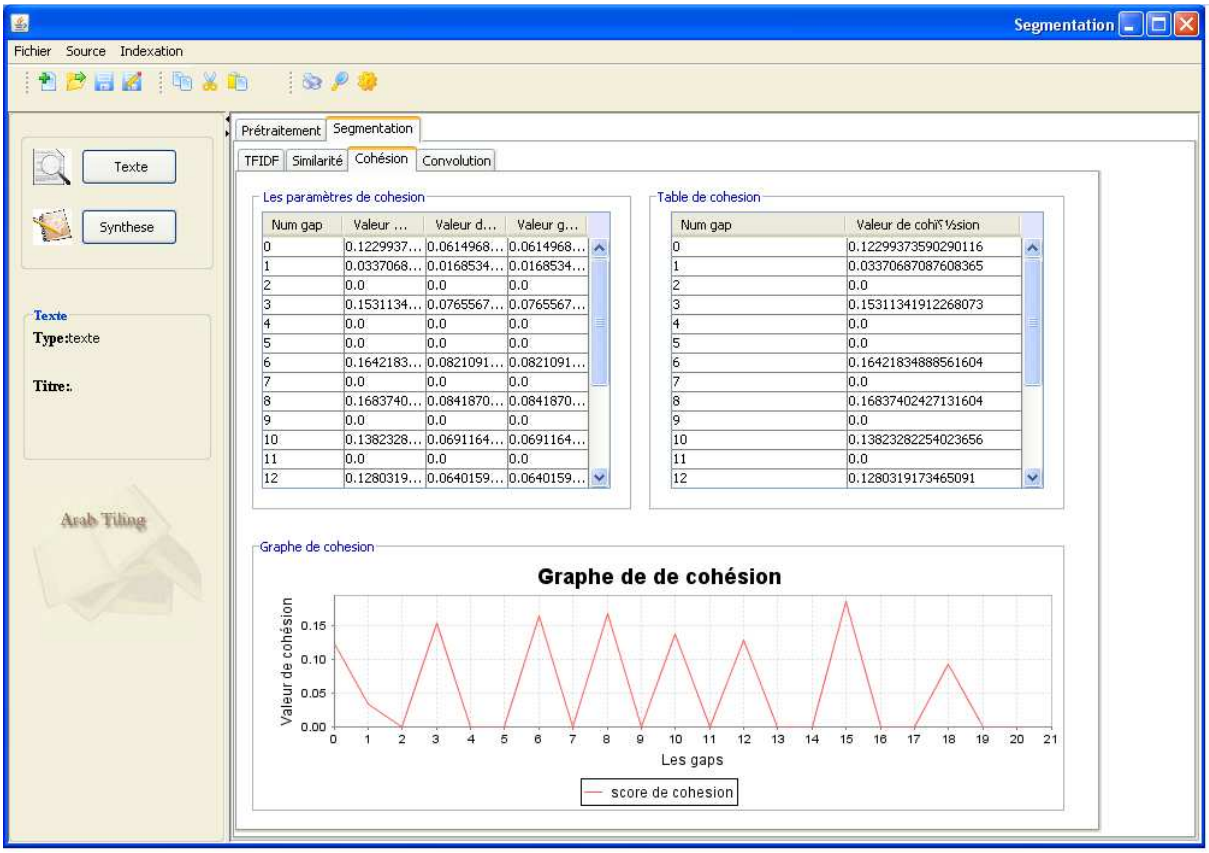

Fig. C.6. Interface de calcul des scores de cohésion

1.5.3. Interface de convolution

La convolution consiste à corriger quelques problèmes dans le graphe de cohésion. La sélection de frontières des segments est basée sur le score de cohésion après la convolution du graphe.

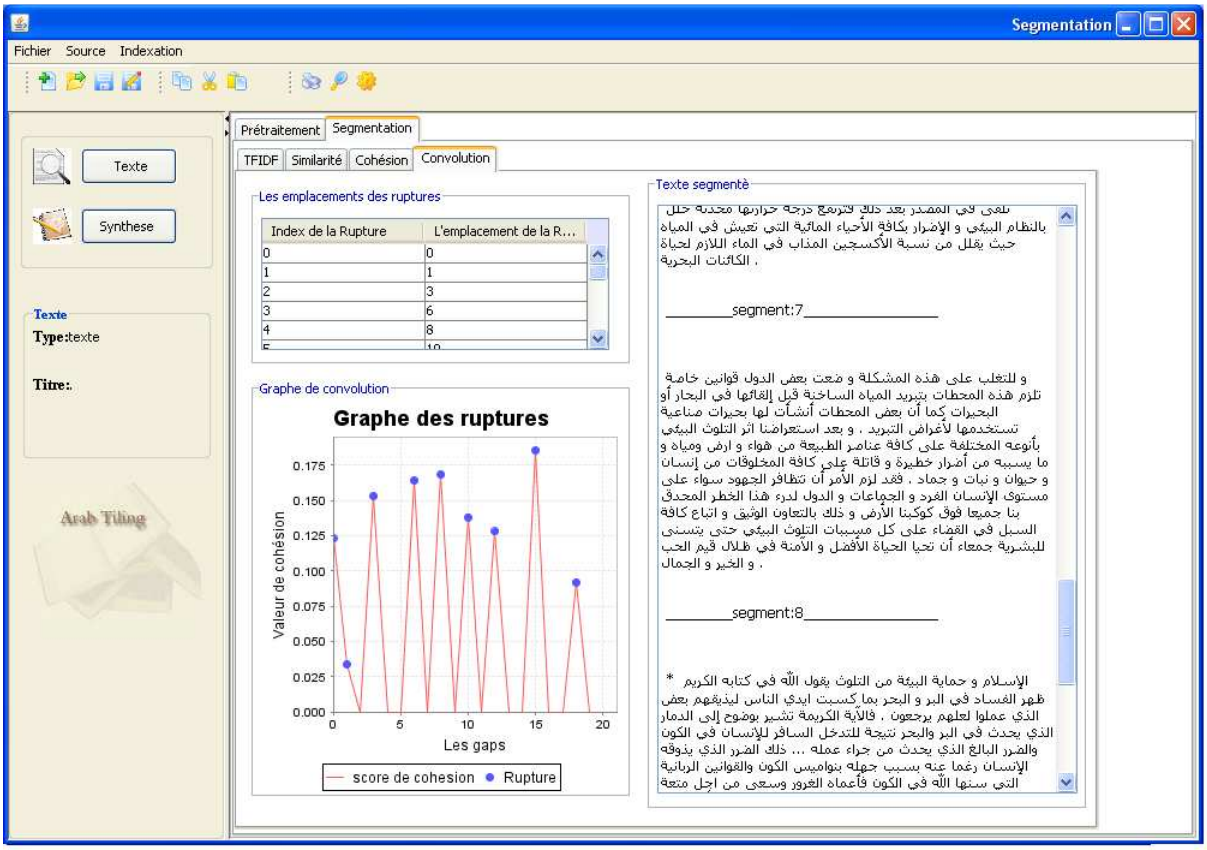

Fig. C.7. Interface de convolution.

Annexe D : Le Système *TopsegArab* pour la segmentation thématique des textes prophétiques

Cette annexe présente le détail du système de segmentation thématique *TOPsegArab* basé sur l'implémentation de l'algorithme C99. Ce système a été développé sous l'environnement de développement *Delphi7*.

1. La fenêtre principale du système « *TOPsegArab* »

Un des objectifs de réalisation du logiciel TOPsegArab est d'offrir une interface ergonomique, l'ergonomie dans TOPsegArab a été assurée par l'utilisation d'un menu principal avec des sous menus qui renferment un ensemble de fonctionnalités, d'une barre d'outils qui facilite l'accès aux fonctionnalités de l'application et d'un ensemble de fenêtres qui résume la démarche du système.

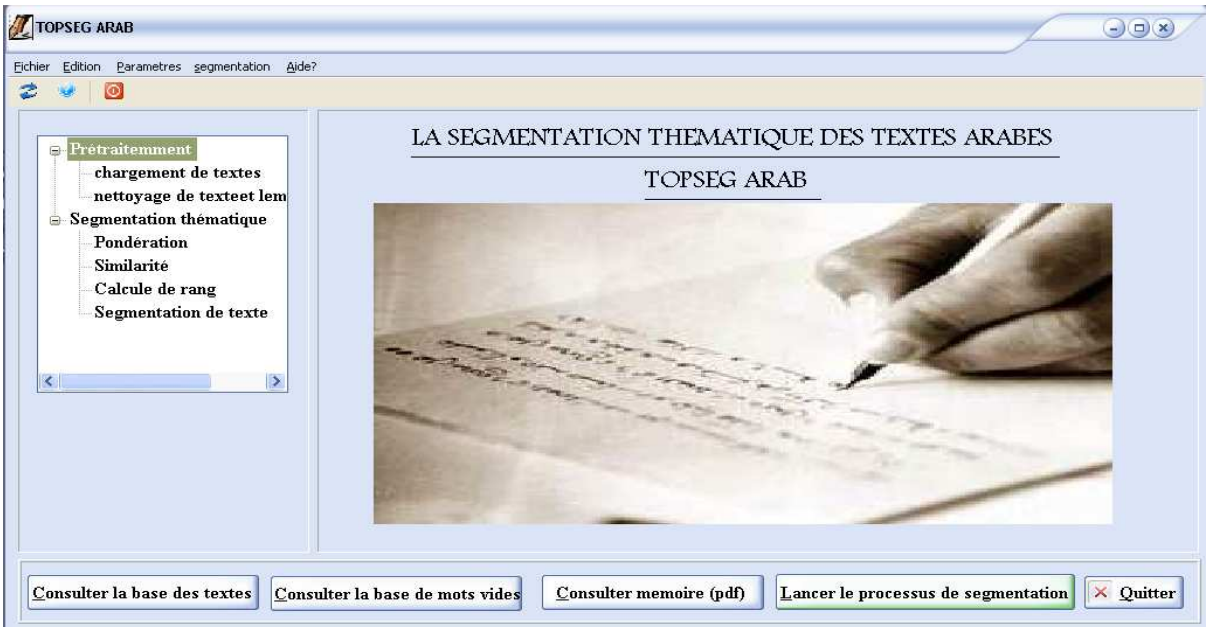

Fig. D.1. La fenêtre principale de TOPsegArab.

2. La segmentation thématique par la méthode C99

La phase de segmentation thématique consiste aussi à une série d'opérations: pondération, calcul de la matrice de similarité entre les phrases, calcul de la matrice de rang, calcul de la matrice des sommes et extraction des zones thématiques.

2.1. Interface de pondération

Les phrases du texte sont représentées dans l'espace vectoriel comme des vecteurs dont les composantes sont les lemmes pondérés par leurs fréquences TF qui prend en compte le nombre d'occurrences d'un lemme dans la même phrase.

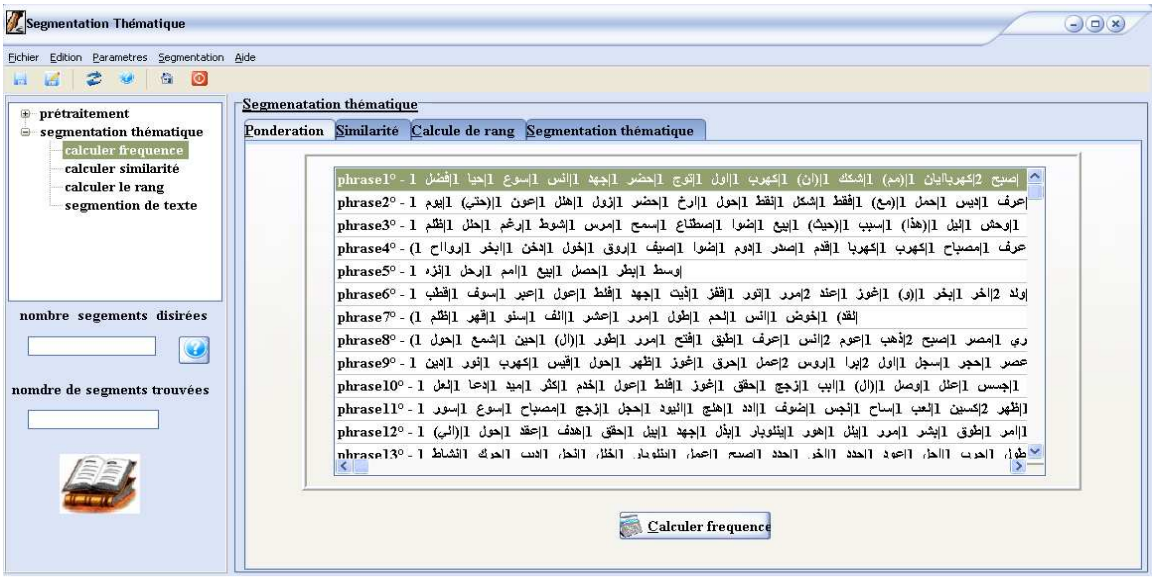

Fig. D.2. Pndération des lemmes.

2.2. Interfaces de calcul des matrices similarité et rang

Le calcul de similarité consiste à faire une comparaison entre les phrases du texte en utilisant la fonction de similarité *Cosine*.

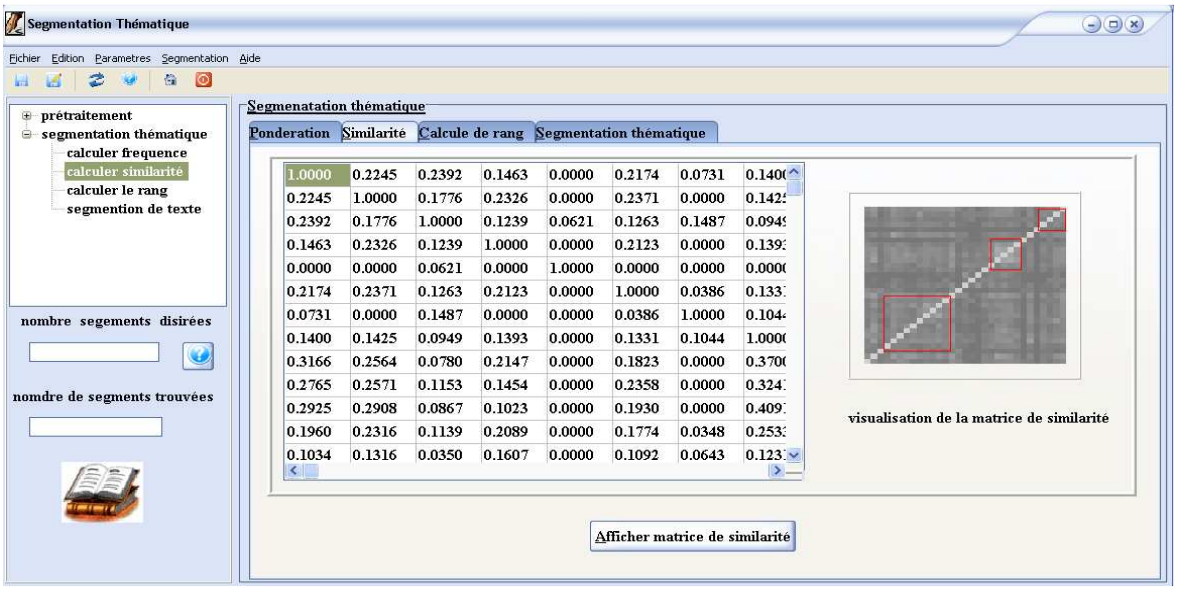

Fig. D.3. Calcul de la matrice de similarité.

Les scores de similarités sont utilisés pour obtenir la matrice de rang. Le système offre à l'utilisateur la possibilité de sélectionner une valeur de masque pour le rang dans une plage de valeurs : (3, 5, 7, 9 et 11).

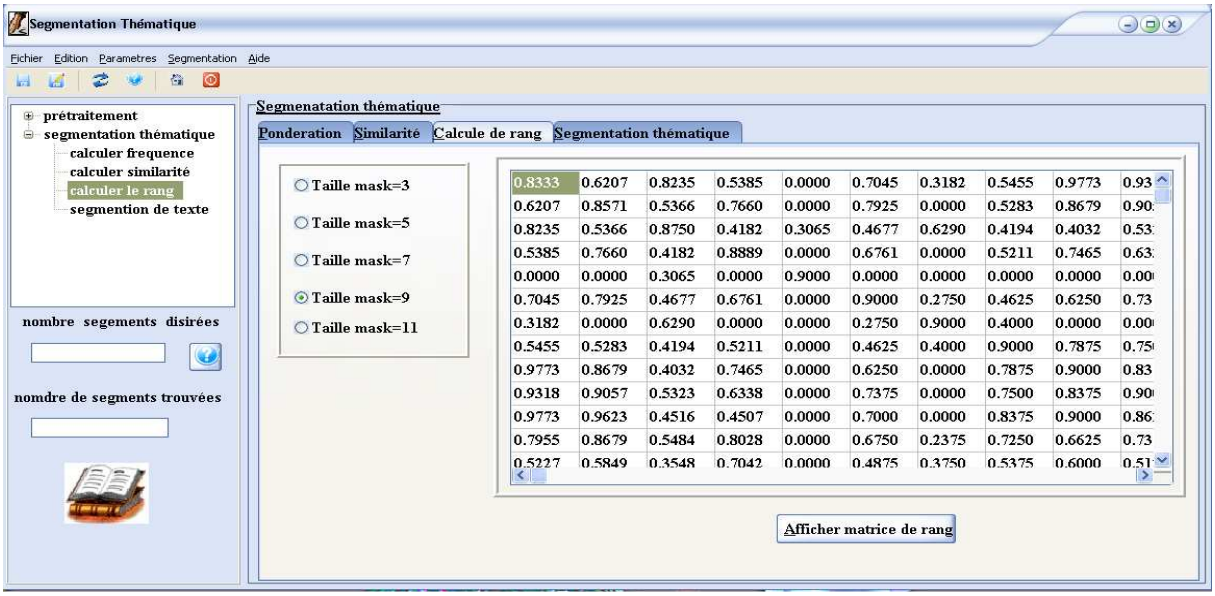

Fig. D.4. Calcul de la matrice de rang.

2.3. Interface d'extraction des zones thématiques

La dernière interface permet de visualiser les ruptures thématiques dans le texte après l'application du processus de segmentation. Dans cette interface, l'utilisateur peut définir le nombre de segments à trouver, il peut aussi par sauvegarder et imprimer le résultat segmentation.

Fig. D.5. Extraction des zones thématiques.