Approche basée sur les arbres sémantiques pour la désambiguïsation lexicale de la langue arabe en utilisant une procédure de vote

Laroussi Merhbene[1] Anis zouaghi[2] Mounir zrigui[3]
(1) LATICE, Faculté des Sciences Juridiques, Economiques et de Gestion de Jendouba
(2) LATICE, ISSAT Sousse, Université de Sousse, Tunis
(3) LATICE, Faculté des sciences de Monastir, Monastir, Tunis
aroussi.merhben@hotmail.com, anis.zouaghi@gmail.com, mounir.zrigui@fsm.rnu.tn

Résumé. Le problème de désambiguïsation lexicale du sens des mots est l'un des plus vieux problèmes de traitement du langage naturel. Dans cet article, nous proposons une approche semi-supervisé pour la désambiguïsation lexicale des mots arabes. La partie supervisée de notre méthode utilise le corpus et le dictionnaire comme ressources pour classifier les contextes du mot ambigu selon le sens. Le regroupement de ces contextes est représenté sous forme d'arbre sémantique. Par la suite nous allons faire la correspondance entre l'arbre sémantique (de chaque sens) et l'arbre de la phrase à désambiguïser pour obtenir un graphe acyclique pondéré. Nous avons défini une nouvelle mesure de score (en utilisant trois mesures de collocation) pour trouver l'arbre sémantique la plus proche. La partie non supervisé de ce travail est basé sur une procédure de vote permettant de classifier les mesures de collocations et de choisir le sens correct du mot ambigu.

Abstract. The problem of word sense disambiguation is one of the oldest problems of natural language processing. In this paper, we propose a semi-supervised approach to word sense disambiguation. The Supervised part of our method uses the corpus and the dictionary as a resource to classify the contexts of the ambiguous word by sense. The combination of these contexts is represented as semantic tree. Thereafter we will make the correspondence between the semantic tree (of each sense) and the tree of the sentence to be disambiguated to obtain a weighted directed acyclic graph. We have defined a new measure score (using three measures of collocation) to find the nearest semantic tree. The unsupervised part of this work is based on a voting procedure for classifying measures collocations and chooses the correct meaning of the ambiguous word.

Mots-clés : Gloses, Extraction de racines, Correspondance de mots, groupement de contextes, arbre sémantique, mesure de collocation, procédure de vote.
Keywords: Glosses, Stemming, string-matching, Context clustering, semantic tree, collocation measures, voting procedure.

1 Introduction

La Désambiguïsation lexicale dans sa définition la plus large est rien de moins que de déterminer le sens de chaque mot dans son contexte, ce qui semble être un processus largement inconscient des gens. Comme un problème de calcul, il est décrit comme «AI-complet" (Ide et Véronis 1998).

L'importance du WSD a été largement reconnue en informatique linguistique ; plusieurs centaines d'articles publiés dans l'ACL Anthology mentionnent le terme «Word Sense Disambiguation". Le WSD est considéré comme un catalyseur pour d'autres tâches et les applications de traitement du langage naturel (TALN), telles que l'analyse, l'interprétation sémantique, la traduction automatique, la recherche d'information, la recherche de texte et l'acquisition d'information lexicale.

Plusieurs systèmes de désambiguïsation lexicale qui se basent soit sur des approches supervisées, non supervisées, à base de connaissances ou hybrides, retournent des taux de précision au niveau de 90% ou plus (Agirre et al., 2006). Ces travaux portent généralement sur un nombre limité de mots et le plus souvent sur des noms dont il y a une large variance de sens entre eux.

Le non supervision signifie qu'il n'y a pas une intervention de l'humain lors du processus de désambiguïsation, ceci est un avantage. Tandis que l'intervention de l'humain pour faire l'apprentissage peut augmenter les performances de notre méthode.

En accord avec cette idée, nous présentons dans ce papier une méthode semi-supervisée pour la désambiguïsation lexicale des mots arabe. La partie innovante dans ce travail est la construction d'un arbre sémantique pour chaque sens du mot ambigu. En outre, nous définissons une procédure de vote qui donne un poids pour les mesures de collocation (utilisés pour mesurer la correspondance entre l'arbre sémantique de chaque sens et l'arbre de la phrase originelle).

Ce papier contient quatre sections, la deuxième section décrit la méthode de désambiguïsation des mots arabes. Les résultats expérimentaux sont décrits dans la section trois. Enfin, la quatrième section constitue la conclusion.

2 Description du système proposé pour la désambiguïsation lexicale des mots arabe

Les méthodes semi-supervisées de désambiguïsation lexicale sont une combinaison entre les méthodes supervisées et non supervisées. En s'inspirant des méthodes de représentation des clusters telles que l'arbre et le réseau lexical (Mihalcea, 2004) et (Navigili et al, 2005), nous avons développé une structure appelée arbre sémantique. Cette dernière est représentée sous forme d'arbre ou les mots clés sont classés selon leur influence sur le sens du mot ambigu. Ce traitement est basé sur l'extraction des racines (des mots appartenant aux phrases contenant le mot ambigu) et l'utilisation de l'algorithme de recherche d'une sous chaine approchée dans une chaine pour trouver les occurrences de ces racines et générer les contextes d'utilisation des mots ambigus.

Ensuite, pour déterminer le sens exact, nous avons défini une nouvelle mesure de similarité basée sur un graphe (obtenu en faisant la correspondance entre l'arbre sémantique et l'arbre de la phrase à désambiguïser) pour trouver l'arbre sémantique la plus proche de l'arbre généré pour la phrase originelle contenant le mot à désambiguïser. Cette dernière peut proposer plus qu'un sens, c'est la raison pour laquelle, nous avons défini une procédure de vote.

Dans ce qui suit, nous décrivons avec plus de détails chaque étape citées ci-dessus.

2.1 Inventaire de sens

L'inventaire de sens est l'une des problématiques majeures des travaux de désambiguïsation lexicale. Nous avons défini une méthode permettant de générer automatiquement pour chaque sens possible du mot ambigu des clusters (Mots clés appartenant aux paragraphes des mots ambigus) permettant de le définir. Certaines étapes de prétraitement seront appliquées à ces groupes et sont détaillées dans la partie suivante.

2.1.1 Prétraitements

En utilisant le corpus, nous allons collecter les phrases contenant les racines des mots à désambigüiser (exp: le mot «العيـــن" "Alayn" nous devons chercher la racine "عيـن" "ayn"). La segmentation de ces phrases est basée sur la ponctuation (., ;, !; ?, etc) et sur le nombre de mots contenus dans une phrase qui doivent être plus que trois.

Ensuite, on élimine les mots vides qui apparaissent fréquemment dans le corpus et n'ont pas une influence sur le sens du mot. La plupart des techniques proposées pour cette tâche (Zou et al., 2006) (Alajmi et al., 2012) sont fondées sur l'idée que les mots vides se produisent avec une fréquence beaucoup plus grande que les mots. Dans l'étude comparative (El-Khair, 2006) trois listes de mots vides ont été utilisées. La première est une liste générale, la seconde a été établie en utilisant une statistique de corpus et le troisième est la combinaison des deux listes. Pour la tâche de recherche d'information, il a été conclu que la première liste a donné les meilleurs résultats que les deux autres listes.

Pour cela, dans ce travail, nous avons utilisé une liste générale contenant 29,985 mots vides. Cette liste a été élaborée par des linguistiques arabe et considérée comme suffisante pour la tâche de désambiguïsation du sens des mots. Plus de détails seront donnés dans les résultats expérimentaux.

2.1.2 Extraction des racines

Chaque mot arabe, nom ou verbe, est généralement basé sur trois lettres et quelques fois sur quatre ou deux lettres. Dans le but d'extraire les racines des mots arabes, nous avons utilisé l'algorithme de «Al Shalabi Kanaan et Al serhan» (Al-Shalabi et al., 2003) qui n'utilise pas de ressources.

Cet algorithme, permet l'extraction de la racine en assignant des poids et des rangs aux lettres constituant un mot. Les poids sont des nombres réels entre 0 et 5. Il divise l'alphabet arabe en 6 groupes. Ces poids affiliés aux lettres ont été déterminés à travers des expériences sur des textes arabes. Le rang de l'ordre des lettres dans un mot dépend de la longueur de ce mot, et si le mot contient un nombre pair ou impair de lettres.

Suite à la détermination du poids et du rang de chaque lettre dans un mot, les poids des lettres sont multipliés par le rang de la lettre. Les trois lettres ayant la plus petite valeur du produit constituent la racine (lire de droite à gauche). Cet algorithme obtient un taux de 90% (Al-shalabi et al., 2003).

La sortie de cet étape est une liste de racines des mots qui constituent les mots appartenant aux gloses $R(g_i) = \{ R_1, R_2, \ldots, R_n\}$, ou g_i est la $i^{éme}$ glose et R_n est la $n^{iéme}$ racine obtenu.

2.1.3 Groupement des sens

L'idée de regroupement des sens est que les phrases extraites du corpus seront classées en groupes en utilisant les racines des mots appartenant aux gloses. Nous utilisons la liste des racines obtenues par la dernière étape et l'algorithme de recherche d'une sous-chaine approchée dans une chaine (Elloumi, 1998) pour trouver les occurrences possibles des racines.Cet algorithme est composé de deux parties essentielles. À l'aide de l'algorithme de remplissage (voir figure 2), nous arrivons à remplir la matrice contenant les deux mots à comparer.

```
Début
(i) (i.a) Construire une matrice M de taille (|x|+1)*(|t|+1); //
Remplissage
  (i.b) pour i:=1 à |x| faire M[i,0]:=i*δ ffaire;
          pour j:=0 à |t| faire M[0,j]:= 0 ffaire;
   (ii)   pour i:=1 à |x| faire
            pour j:=1 à |t| faire
               M[i,j]:= min{ M[i-1,j-1]+1,
                             M[i,j-1]+1,
                             M[i-1,j]+δ }
            ffaire
          ffaire
```

FIGURE 1 : Première partie « Remplissage de la matrice » de l'algorithme recherche d'une sous-chaine approchée dans une chaine.

Soient t et x deux chaînes telles que |x|<|t| et δ = cout de substitution. Par la suite on utilise l'algorithme de traçage arrière (voir figure 3) pour trouver la plus courte sous séquence commune. Soient γ le coût d'insertion et $\sigma_{i,j}$ le coût de surpression. Les mots contenant cette sous séquence commune seront considérés comme des occurrences de la racine. Une liste $L(R_i)$ d'occurrences sera générée des racines obtenues à partir de la dernière étape.

```
   (iii)    (iii.a)   Choisir   q,   1≤q≤|t|,   telle   que
M[|x|,q]=min1≤j≤|t|{M[|x|,j]};      // Traçage-Arrière
             i:=|x|; j:=q;
        (iii.b) tant que i≠0 et j≠0 faire
   si M[i,j]=M[i,j-1]+γ alors j:=j-1
   sinon
                si M[i,j]=M[i-1,j-1]+σi,j alors
               j:=j-1; i:=i-1
        sinon i:=i-1
              fsi
        fsi
   ffaire;
   (iv) p:=j+1;
       x':=tp,q
Fin
```

FIGURE 2: Deuxième partie « Traçage arrière » de l'algorithme recherche d'une sous-chaine approchée dans une chaine.

Vu que cet algorithme prend beaucoup de temps lors de son exécution, nous avons pensé pour faciliter cette tâche de générer une base de connaissances dans laquelle on enregistre les occurrences de chaque racine. Ainsi avant d'exécuter

cet algorithme on commence par parcourir la base de connaissances, si on ne trouve pas la racine on exécute cet algorithme de correspondance des mots.

2.2 Modélisation des groupes de sens

Plusieurs types de représentations textuelles structurés ont été élaborés, à savoir les graphes de cooccurrences (Véronis, 2004) et les graphes sémantiques pour l'analyse des chemins et des liens (Mihalcea, 2004) et (Navigili et al, 2005).

Dans ce travail nous avons choisi de représenter le texte (les groupes de sens) avec des arbres binaires. Ce choix est dù aux besoins de notre approche, à la rapidité de recherche pour les arbres, la compacité de la représentation et simplicité des algorithmes de calcul.

Le mot ambigu est représenté comme racine de l'arbre binaire et les mots qui l'entourent sont représentés comme des descendants. Les niveaux des descendants dans l'arbre binaire dépendent de la position de ces descendants par rapport au mot ambigu, c'est-à-dire, en premier niveau de l'arbre on trouve les mots qui sont situés juste à droite et à gauche du mot ambigu dans le corpus. Les différents phrases contenus dans les groupes de sens ou clusters obtenus seront transformés en arbre binaire dont la structure est la suivante T = (N, E, R, RC, LC, L), ou :

- N est un ensemble de nœuds, N = {n_1… n_n}. Chaque nœud correspond à un concept dans l'arbre binaire.
- E est un ensemble d'arêtes qui représente la relation entre le nœud N_i et le nœud N_j.
- R est la racine de l'arbre qui est le mot ambigu.
- RC est l'ensemble des fils droits qui sont les mots apparaissant à droite du mot ambigu.
- LC est l'ensemble des fils gauche, qui sont les mots apparaissant à gauche du mot ambigu.
- L est une fonction qui détermine le niveau des nœuds, il correspond à leur position par rapport au mot ambigu.

Si on excepte la racine R, chaque nœud de l'arbre possède exactement un seul fils. On appelle <R, RC, LC> un arbre binaire schématisé dans la figure 3 (b) suivante.

Un arbre sémantique permet d'arranger les mots contenus dans les différents clusters de sens. Cette tâche dépend du nombre d'occurrences des mots (contenue dans le même contexte du mot ambigu), aussi de la position par rapport au mot ambigu dans son contexte. Les mots les plus proches du mot ambigu ont généralement une influence sur sa signification.

Le choix de ces facteurs a été fait à partir des travaux de Yarowsky (Yarowsky, 1993) montrant que la performance d'un système de désambiguïsation lexicale diminue lorsque la distance par rapport au mot ambigu augmente.

La création d'arbre sémantique se fait à partir de la fusion de plusieurs arbres binaires obtenus. Nous obtenons un graphe acyclique dirigé ou arbre n-aire que nous appelons arbre sémantique, ST = (N, E, R, C, L, Nb, H), où :

- C'est l'ensemble des nœuds fusionnés, C = {c_1, ... c_n}. Les fils gauche et droite de chaque arbre binaire seront liés à la racine de l'arbre sémantique.
- Nb est une fonction qui retourne le nombre de nœuds dans l'arbre sémantique.
- H est une fonction qui retourne la hauteur de l'arbre sémantique.

Pour chaque nœud de l'arbre sémantique, nous définissons une structure qui contient les données suivantes :

- W est un ensemble de mots qui représente les étiquettes des nœuds. Ces mots sont les mots clés qui caractérisent un sens spécifique (obtenue à partir de l'étape de création de contexte d'utilisation).
- Enfant (N) est une fonction qui renvoie le nombre de fils d'un nœud N.
- Freq (N) est une fonction qui retourne le nombre de fréquences d'un nœud N.

Nous avons utilisé comme notation l'arbre sémantique, arbre en raison de sa structure basée sur la position des mots par rapport au mot ambigu et sémantique car les nœuds sont les mots des groupes de sens.

Dans le cas où l'on trouve un nœud qui existe déjà dans l'arbre sémantique, nous devons le fusionner. Lors de l'étape de fusion des arbres, nous utilisons un algorithme de parcours en largeur pour trouver le nœud répété soit dans un niveau plus élevé, même niveau ou un niveau inférieur.

2.3 Procédure de désambiguïsation

La procédure de désambiguïsation détaillée dans cette section est basée sur 3 étapes. Nous commençons par la création de liens pondérés par des mesures de collocation entre l'arbre binaire de la phrase à désambiguïser et l'arbre

sémantique, cette étape est appelée correspondance. Par la suite nous mesurons la similarité sémantique entre l'arbre de la phrase originale et l'arbre sémantique de chaque sens appelé ST_{S_k}, ou S_k correspond au $k^{ième}$ sens). Cette mesure peut proposer plus qu'un seul sens, dans ce cas, nous utilisons la procédure de vote.

2.3.1 Correspondance des arbres : graphe acyclique pondéré

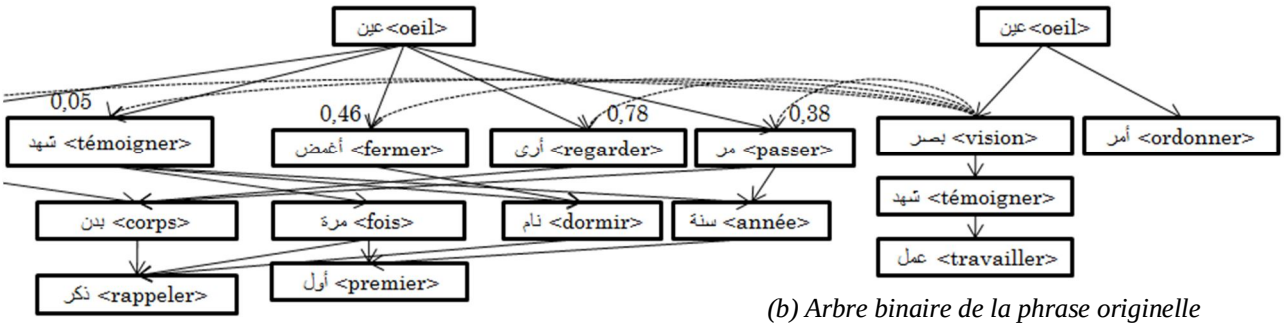
Les nœuds de l'arbre correspondant à la phrase originelle (contenant le mot à désambiguïser) sont liés aux nœuds de même niveau dans l'arbre sémantique de chaque sens. Les liens sont pondérés à l'aide de trois mesures différentes de collocation (détaillées dans ce qui suit).

Cette étape est appelée correspondance des arbres, on obtient un graphe orienté pondérée avec l'une des trois mesures de collocation (détaillés dans ce qui suit) comme un poids d'une arête (noté wc_{ij}). Nous ajoutons des liens pondérés par les mesures de collocation entre les nœuds N_i de l'arbre de la phrase à désambiguïser T_{os} et les nœuds N_j de l'arbre sémantique de chaque sens.

La figure 3 montre un exemple de correspondance entre l'arbre de la phrase d'origine et l'arbre sémantique obtenue précédemment. La phrase originale contenant le mot ambigu est la suivante:

"و كيف لا يكون الأمر كذلك و العين تبصر و تشاهد مثل هذه الأعمال ."

"Et comment ne pas être le cas et la perspicacité des yeux et de voir de tels actes."



(a) Fragment de l'arbre sémantique de la première glose du mot "عين" "ayn".

(b) Arbre binaire de la phrase originelle

FIGURE 3 : Exemple de correspondance entre l'arbre binaire de la phrase originelle (b) et un fragment de l'arbre sémantique de la première glose du mot "**عين**" "ayn" (a).

Pour cette phrase, nous devons éliminer les mots vides à l'aide de la liste prédéfinie de mots vides. Les mots vides qui seront éliminés sont (و, كيف, لا, يكون, كذلك, مثل, هذه) (et, comment, pas, être, bien, comme, ça). Par la suite nous allons extraire les racines des mots contenus dans la phrase originelle, ces racines sont les nœuds de l'arbre et selon leur position par rapport au mot ambigu nous allons affilier le niveau dans l'arbre. Les liens entre les nœuds sont pondérés par w_p (= 1 / niveau des nœuds).

Par exemple le mot "أمر" "AMR" est dans la deuxième position, le nœud qui lui correspond est situé dans le deuxième niveau de l'arbre. Le poids affiliée à la liaison entre les mots "عين" "ayn" et "أمر" "amr" est 1/2 = 0,5.

Chaque nœud de l'arbre extrait de la phrase originale, est lié avec les nœuds du même niveau dans l'arbre sémantique d'un sens particulier. Les liens utilisés pour l'étape de correspondance apparaissent en bleu en pointillés. Ils sont pondérés en utilisant des mesures de collocations (définit dans ce qui suit) normalisés entre 0 et 1 et appelé wc_{ij} qui correspond aux arcs qui lient les nœuds w_i et w_j.

Par exemple, le mot "مر" "marra" occurre 7 fois dans le corpus avec le mot "بصر" "bsr", en normalisant le poids de l'information mutuelle, on obtient wc_{ij} = 0,38. Pour la correspondance entre l'arbre de la phrase originelle et l'arbre sémantique d'un sens particulier du mot ambigu, nous utilisons trois mesures de collocation cités dans ce qui suit.

2.3.1.1 Le T-test

L'un des tests les plus connues dans le domaine de recherche de collocations (Manning et al., 1999), le t-test (voir équation 1) est calculé de la façon suivante :

$$wc_{ij} = T = (\bar{x} - \mu) / (\sqrt{\frac{s^2}{N}}) \quad \text{(Equation 1)}$$

Où $\bar{x}$ (la moyenne d'échantillon) est égale à s^2 (variance d'échantillon) est égale au nombre d'occurrence des deux mots divisé par le nombre total de mots dans le corpus ; N la taille d'échantillon; L'hypothèse nulle μ est mesuré en multipliant $P(w_i)$ par $P(w_j)$, ou $P(w_i)$ = Nombre d'occurrence de w_i dans le corpus divisé par le nombre total de mots dans le corpus.

2.3.1.2 Le Khi Carré

Pour le calcul du khi carré χ^2, nous mesurons $C_{1,1}$ qui est le nombre de fois ou w_1 et w_2 occurrent ensemble, $C_{1,2}$ correspond au nombre d'occurrence de w_1 sans prendre en considération w_2, $C_{2,1}$ correspond au nombre d'occurrence de w_2 sans prendre en considération w_1 et enfin $C_{2,2}$ le nombre de couples dans le corpus sans considérer le couple w_1,w_2. Dans ce qui, nous allons donner l'équation 2 utilisé pour le calcul du Khi Carré en utilisant le tableau 2 par 2.

$$\chi 2 = \frac{N \times (C_{1,1} \times C_{2,2} - C_{1,2} \times C_{2,1})^2}{(C_{1,1} + C_{1,2}) \times (C_{1,1} + C_{2,1}) \times (C_{1,2} + C_{2,2}) \times (C_{2,1} + C_{2,2})} \quad \text{(Equation 2)}$$

D'après (Maning et al., 1999), le $\chi 2$ est approprié pour les probabilités larges, pour lesquelles le t test ne donne pas des résultats satisfaisantes. Pour cela le $\chi 2$ est utilisé dans la plupart des problèmes de découverte de collocation.

2.3.1.3 Information Mutuelle

Cette mesure détermine combien un mot peut nous indiquer un autre mot (Manning et al., 1999), Elle est définit de la manière suivante (voir équation 3).

$$IM(w_i, w_j) = \log_2 \frac{P(wi,wj)}{P(wi)\,P(wj)} \quad \text{(Equation 3)}$$

Les bi-grammes ayant un nombre de fréquence qui n'est pas important, auront un score élevé que ceux qui ont un nombre de fréquence élevé.

2.3.2 Mesure de similarité sémantique

Pour la définition de la mesure de similarité, nous partons de la logique que pour mesurer la similarité entre deux phrases (la phrase à désambiguïser et le contexte d'utilisation générée pour le sens i du mot ambigu), nous devons mesurer la similarité entre les mots de chaque phrase.

D'autre part dans cette mesure, nous intégrons la position des différents mots de la phrase à désambiguïser par rapport au mot ambigu. Pour cela, nous utilisons le niveau des mots dans l'arbre sémantique, celui-ci dépond la position du terme correspondant à gauche ou à droite du mot ambigu.

La mesure de score définit dans ce qui suit (voir équation 4) nous permet de trouver l'arbre sémantique T_{st} la plus proche à l'arbre de la phrase originelle T_{os}.

$$Score = \sum_{N_i \in T_{os}} (\sum_{N_j \in ST_{S_k}} (wc_{ij} / ST_{S_k}(L(N_j)) / Nb(ST_{S_k})) / Nb(T_{OS})) \quad \text{(Equation 4)}$$

Cette mesure est la moyenne du produit de w_p et wc_{ij} entre les nœuds de T_{os} et ST_{sk}. Où $Nb(T_{OS})$ Est le nombre total de nœuds dans l'arbre T_{os} et $Nb(ST_{S_k})$ le nombre total des nœuds liés aux nœuds de l'arbre ST_{S_k}. $ST_{S_k}(L(N_j))$ Correspond au niveau des nœuds N_j contenu dans l'arbre sémantique ST_{S_k}.

2.3.3 Classification des mesures de collocation

La procédure de vote est utilisée pour un ensemble d'algorithmes. Chaque algorithme va donner un sens au mot ambigu et le sens qui a la majorité des votes sera choisi comme le sens correct (Navigili, 2009). Nous distinguons la majorité de vote, la combinaison de probabilités et la combinaison basée sur le rang, ces méthodes de vote ont été différenciées en variant le poids utilisé pour le vote.

Notre contribution par rapport à ce qui est existant est que nous donnons un poids pour les mesures de collocation utilisées par la mesure de score, non pas pour les sens proposés par les différents méthodes. La procédure de vote est une nouvelle approche supervisée, l'idée est que lors de l'étude expérimentale, nous avons classé les mesures de collocation selon l'attribution du sens. Un rang sera donné pour chaque mesure permettant de les classer selon l'attribution correcte des sens. Nous distinguons trois cas.

Dans le cas où les mesures de collocations donnent des résultats différents, alors la procédure de vote sera appliquée. Outre, lorsque les trois mesures de collocation donnent le même résultat, alors le sens donné sera attribué au mot ambigu et les rangs ne seront pas modifiés.

Les mesures de collocation peuvent donner des résultats différents, dans le cas où plus d'une mesure est en accord sur l'attribution d'un sens au mot ambigu, nous devons choisir le sens ayant la majorité des votes. Les rangs des mesures qui ont votés pour le sens attribué seront incrémentés et les rangs des mesures qui n'ont pas votés pour le sens attribué seront décrémentés.

Lorsque chacune des mesures de collocation donne un sens différent. Dans ce cas, le résultat donné par la mesure ayant le rang le plus élevé (attribué lors du dernier test de classification N) sera utilisé pour attribuer le sens du mot ambigu. Dans ce qui suit, nous détaillons les résultats donnés par la méthode décrite.

3 Résultats Expérimentaux

Avant d'entamer la partie ou nous donnons les résultats de notre méthode, nous allons décrire les données testés et les ressources utilisées.

3.1 Ressources utilisés

3.1.1 Dictionnaire

Pour la désambiguïsation de la langue arabe nous avons besoin d'un dictionnaire arabe-arabe qui contient les différents sens du mot ambigu, le problème dans les dictionnaires classiques est qu'ils contiennent des sens qui ne sont plus utilisés de nos jours. Nous utilisons le dictionnaire « Alwassit » (Muṣṭafá et al., 2008) qui est très connu pour la langue arabe et contient les anciens et nouveaux sens.

La plupart des travaux de désambiguïsation lexicale de la langue arabe et les autres langues, utilisent des sens ayant une granularité grosse, cela signifie que les sens des mots ne sont pas nombreux d'une part et ne sont pas très détaillés d'autre part.

Vu le nombre important de sens donné par le dictionnaire, nous devons travailler avec les sens de granularité fine. Ce choix rend notre travail plus ardu et complexe puisqu'il augmente le nombre de sens à considérer.

3.1.2 Corpus

Le corpus utilisé dans ce travail est l'ensemble de plusieurs corpus collectés. Les textes contenus dans ces corpus sont des articles de presse, des articles, des livres, des magazines et des articles de blogs téléchargés du net qui ont été enregistrés sans restriction. Le nombre total de mots dans le corpus est 123,8554,642 mots. Ce nombre important de mots dans le corpus, nous a aidés à trouver les occurrences pour la plupart des sens des mots ambigus testés.

3.2 Données expérimentées et problèmes rencontrés

Nous avons testés 127 mots ambigus qui ont été choisis par leur sens hors contexte. Pour chacun de ces mots ambigus, nous avons évalué 60 exemples par sens et 20 exemples lors de la partie de classification des mesures de collocations. De nombreux problèmes ont été rencontrés lors du processus de désambiguïsation cité dans ce qui suit:

– Nous avons trouvé des exemples pour les tests qui peuvent être jugés comme satisfaisants pour le processus de désambiguïsation. Nous avons eu recours à quatre annotateurs qui nous ont aidés à choisir des phrases qui permettent de donner plusieurs possibilités pour le choix du sens du mot ambigu. Le taux d'accord entre les annotateurs est de 73%.

– Pour certains mots considérés, nous avons trouvé des sens qui apparaissent dans le corpus et n'existent pas dans le dictionnaire. Pour le mot "ayn" on extrait une dizaine de phrases du corpus où il signifie un nom d'une ville au Liban. Un échantillon est donné dans ce qui suit:

"تستقبلنا مدينة العين ببهاء يختلف تماما عما ألفناه في أبوظبي" → "La ville d'Ayn nous reçoit brillamment complètement différente avec ce qu'on s'est habitué à Abu-Dhabi".

– Nous avons utilisé la granularité fine des sens et certaines sens sont presque inexistantes dans les textes du corpus, pour cela le nombre d'exemples testés varie d'un sens à un autre pour un seul mot. Comme solution nous avons essayé de générer du net le maximum de phrases qui correspond au sens ayant un nombre d'occurrence faible dans notre corpus.

3.3 Résultats obtenus

Nous détaillons dans le tableau 1 suivant, les résultats obtenus par notre méthode. En divisant le nombre de mots désambiguïsés correctement par le nombre de mots testé nous mesurons la précision. En plus de ces données, nous utilisons le nombre de mots ambigus pour mesurer le rappel et la couverture. On trouve aussi le F-score qui détermine la moyenne harmonique pondérée de la précision et du rappel (Navigili, 2009). Ces taux sont détaillés pour chaque mesure de collocation ainsi que pour la procédure de vote.

wc_{ij}	Rappel	Précision	F-Score	Couverture
T	0,739	0,754	0,747	0,9798
MI	0,70382	0,7182	0,7109	0,9798
χ 2	0,7590	0,7746	0,7668	0,9798
Procédure de vote	0,8305	0,8305	0,8305	1

TABLEAU 1: Les performances de notre méthode.

On remarque que le F-score obtenu en appliquant la procédure de vote est supérieur à celui obtenu par quelconque mesure de collocations. Il n'y a pas une grande différence entre la précision et le rappel obtenu par les mesures de collocations. Ceci peut s'expliquer par le fait que la majorité des mots testés ont été désambiguïsés, ce qui est mesuré par la couverture (proche de 100%). Cependant, la meilleure mesure de collocation est le χ 2, sinon la procédure de vote augmente le F-score de 6%.

L'avantage de l'arbre sémantique peut être démontré si on mesure la performance de notre méthode en variant le nombre de nœuds utilisés lors de l'étude expérimentale (voir la figure 4 ci-dessus).

Nous trouvons que plus l'arbre sémantique est enrichi par les nœuds, plus le F-score augmente. La diminution du F-score est principalement due à l'insuffisance du nombre de nœuds, ce qui conduit à l'échec de répondre à tous les événements possibles.

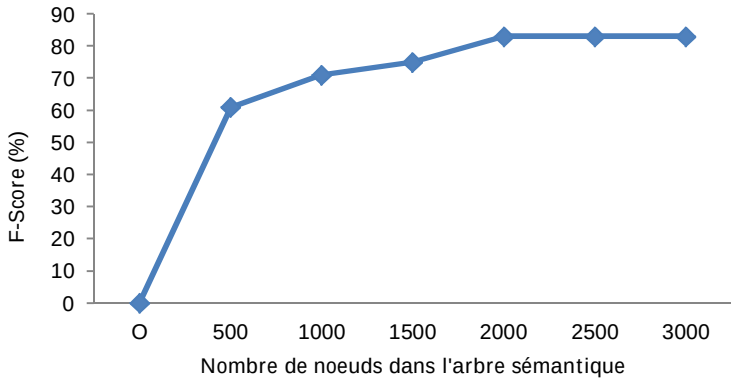


FIGURE 4. Performance de notre méthode et le nombre de nœuds correspondant dans l'arbre sémantique.

Nos résultats indiquent que pour les arbres sémantiques ayant au moins 500 nœuds, les performances de notre méthode augmente constamment. Outre, le F-score atteint le maximum et devient stable pour les tailles d'arbres sémantiques entre 2.000 et 3.000 nœuds.

Nous allons maintenant discuter les performances de notre méthode pour quelques mots ambigus. Nous mentionnons dans le tableau 2 ci-dessous, certains mots ambigus et le nombre de sens. Aussi pour le sens le plus fréquent nous détaillons le F-score obtenu et le rang des mesures de collocation obtenu après la phase de classification de la procédure de vote.

Mots	**Vocalisation**	**Nombre de sens**	**F-Score**	**Rang** T_{test}	**Rang** $_{MI}$	**Rang** $_{\chi 2}$
عين	Ain	16	0,7421	+14	+4	+12
حسب	Hsb	14	0,7532	+12	-3	+10
شعر	Chaar	8	0,8926	+14	+0	+19
فجر	Fjr	6	0,8420	+10	+18	+17
نور	nr	4	0,9605	+9	+3	+19

TABLEAU 2 : Résultats d'évaluation individuelle pour quelques mots ambigus.

D'après le tableau 2, nous pouvons noter que le plus faible F-score est obtenu par les mots ambigus ayant le plus grand nombre de sens. Dans les trois dernières colonnes du tableau 2, nous détaillons le rang des mesures de collocations pour le sens le plus fréquent. Pour les mots ambigus ayant plus de 10 sens, les rangs des mesures de collocation est inférieur à 15. En revanche, les mots ayant moins de 10 sens donnent le meilleur F-Score. Les mesures de collocations correspondantes à ces mots sont les plus classées (plus de 16).

En résumé, nos résultats indiquent que le $\chi 2$ est la mesure de collocation ayant le rang le plus élevé pour la majorité des données testées. Les mots ambigus ayant le nombre de sens le moins élevé donnent les meilleures performances. Ceci s'explique par le fait qu'elles facilitent le choix du sens correct.

La plupart des travaux qui ont été réalisés dans le domaine de désambiguïsation lexicale des autres langues, ont l'avantage d'être évalués tout au cours des conférences Senseval et SemEval. Ces travaux ont été testés en utilisant les mêmes ressources, les mêmes échantillons et la même granularité des sens. Pour les travaux de la langue arabe, les ressources et les échantillons testés sont inexistantes et non disponibles pour pouvoir comparer les travaux.

4 Conclusion

Cet article décrit une approche basée sur les arbres sémantiques pour la désambiguïsation semi-supervisée de la langue arabe. Le principal inconvénient de la langue arabe semble être le grand nombre de sens hors contexte pour les mots ambigus. L'étape de regroupement de sens était très bénéfique pour atteindre les performances obtenues par notre méthode. D'autre part, la représentation de l'arbre sémantique pour chaque sens était très pratique.

La mesure de score proposée (pour mesurer la correspondance entre l'arbre sémantique et l'arbre de la phrase originelle) utilise trois mesures de collocations qui seront classés en utilisant une procédure de vote supervisé.

Lors l'étude expérimentale nous avons testé des mots arabes ambigus choisis par leur nombre de sens hors de contexte. Les résultats montrent que notre méthode permet d'obtenir un taux de rappel et de précision élevé (83%).

Nous proposons dans les futurs travaux d'utiliser d'autres ressources utiles pour la langue arabe afin d'augmenter les performances de notre méthode.

Remerciements

Nous adressons nos plus vifs remerciements aux linguistes de notre université pour leur aide qu'ils nous ont apportés. Nous tenons aussi à exprimer notre gratitude envers les membres de notre unité de recherche LATICE pour leur soutien.

Références

AGIRRE E. AND EDMOND P. (2006). Word Sense Disambiguation: Algorithms and Applications. Springer.

AL-SHALABI R., KANAAN G., AL-SERHAN H. (2003). New approach for extracting Arabic roots. Papier présenté à ACIT, the International Arab Conference on Information Technology. Egypt, p. 42-59.

ALAJMI A., SAAD E.M., DARWISH R.R.(2012). Toward an Arabic Stop-words List Generation. International Journal of Computer Applications (0975-8887), vol.46, n°8, p. 9-13.

EL-KHAIR I. A. (2006). Effect of Stop Words Elimination for Arabic Information Retrieval: A comparative Study. International journal of Computing & Information Sciences, vol.4, n°3, p.119-133.

ELLOUMI M. (1998). Comparison of Strings Belonging to the Same Family. Information Sciences, An International Journal, Elsevier Publishing Co., Amsterdam, North-Holland (Publisher), vol. 111, n°(1-4), p. 49-63.

IDE I. AND VERONIS J. (1998). Word Sense Disambiguation : The State of the Art. Computational Linguistics, vol. 24 (1), p. 1-41.

MANNING C., SCHUTZE H. (1999). Foundations of Statistical Natural Language Processing. MIT Press, Cambridge, MA.

MERHBENE L., ZOUAGHI A. AND ZRIGUI M. (2012). Lexical Disambiguation of Arabic Language: An Experimental Study ». In proceeding of 11th Mexican International Conference on Artificial Intelligence, San Luis Potosí, SLP, México.

MUṢṬAFA M., SAYED AHMED N., DARWICH M., ABDALLAH A. (2008). Mu'jam al-Wasīṭ. Published in Bayrūt : Dār Iḥyā' al-Turāth al-'Arabī lil-Ṭibā'ah wa-al-Nashr wa-al-Tawzī'.

MIHALCEA, R., TARAU, P., AND FIGA E. (2004). Pagerank on semantic networks, with application to word sense disambiguation. In Proceedings of the 20th International Conference on Computational Linguistics (COLING, Geneva, Switzerland), p. 1126–1132.

NAVIGLI, R. (2005). Semi-automatic extension of large-scale linguistic knowledge bases. In Proceedings of the 18th Florida Artificial Intelligence Research Society Conference (FLAIRS, Clearwater Beach, FL), p.p: 548–553.

NAVIGILI R. (2009). Word Sense Disambiguation: A Survey. ACM Computing Surveys, Vol. 41, No. 2, Article 10, Publication date: February, p. 1-69.

VERONIS, J. (2004). Hyperlex: Lexical cartography for information retrieval. Comput. Speech Lang. 18, 3, p.p: 223–252.

YAROWSKY, D. (1993). ONE SENSE PER COLLOCATION. In Proceedings, ARPA Human Language Technology Workshop. Princeton, pp. 266-271.

ZOU F., WANG L., DENG X., HAN S. AND WANG L. S. (2006). Automatic Construction of Chinese Stop Word List », in proceeding of the 5th WSEAS International Conference on Applied Computer Science, Hangzhou, China, p. 1010-1015.