Évaluation des performances d'un modèle de langage stochastique pour la compréhension de la parole arabe spontanée

Anis ZOUAGHI[1], Mounir ZRIGUI[1], Mohamed BEN AHMED[2]

[1] Labo RIADI (Unité de Monastir)
Université de Monastir, Faculté des sciences de Monastir
[2] Labo RIADI – Université de la Mannouba,
École nationale des sciences de l'informatique
Anis.Zouaghi@riadi.rnu.tn, Mounir.Zrigui@fsm.rnu.tn,
Mohamed.Benahmed@riadi.rnu.tn

Résumé. Les modèles de Markov cachés (HMM : Hidden Markov Models) (Baum et al., 1970), sont très utilisés en reconnaissance de la parole et depuis quelques années en compréhension de la parole spontanée latine telle que le français ou l'anglais. Dans cet article, nous proposons d'utiliser et d'évaluer la performance de ce type de modèle pour l'interprétation sémantique de la parole arabe spontanée. Les résultats obtenus sont satisfaisants, nous avons atteint un taux d'erreur de l'ordre de 9,9% en employant un HMM à un seul niveau, avec des probabilités tri_grammes de transitions.

Abstract. The HMM (Hidden Markov Models) (Baum et al., 1970), are frequently used in speech recognition and in the comprehension of foreign spontaneous speech such us the french or the english. In this article, we propose using and evaluating the performance of this model type for the semantic interpretation of the spontaneous arabic speech. The obtained results are satisfying; we have achieved an error score equal to 9.9%, by using HMM with tri-grams probabilities transitions.

Mots-clefs : analyse sémantique, modèle de langage stochastique, contexte pertinent, information mutuelle moyenne, parole arabe spontanée.

Keywords: semantic analysis, stochastic language model, pertinent context, overage mutual information, spontaneous arabic speech.

1 Introduction

On distingue deux grands courants d'approches pour la compréhension de la parole : les approches symboliques linguistiques (ou par règles), et les approches stochastiques. Le premier type d'approches se base sur une représentation préalable de la grammaire. Pour décrire cette grammaire, on utilise généralement l'un des formalismes existants tels que : HPSG, les grammaires lexicales fonctionnelles (LFG), etc. Quand au deuxième type

d'approche, les règles sont déduites directement à partir d'un corpus d'apprentissage. Depuis quelques années, la tendance est vers l'utilisation des modèles de langages stochastiques dans le domaine de la compréhension de la parole spontanée (Schwartz et al., 1996), (Minker, 1999), (Bousquet, 2002), etc. Cette tendance s'explique par le faite que les approches stochastiques offrent une alternative efficace aux approches par règles, concernant le coût global de développement du modèle, et la portabilité vers d'autres domaines. De plus, du fait que le locuteur parle d'une manière spontanée, les fautes de syntaxe ou de grammaire sont beaucoup plus fréquentes à l'oral qu'à l'écrit. C'est pour cela, qu'une analyse portant uniquement sur la syntaxe n'est souvent pas efficace. Ainsi, certains proposent pour faire face à ce problème, une analyse plus fine des phénomènes linguistiques de l'oral tels que (Van Noord et al., 1999) et (Antoine et al., 2003), ou une combinaison d'une analyse syntaxique et sémantique tels que (Villaneau et al., 2001), (Seneff, 1992), etc. Contrairement à la langue latine, la compréhension automatique de la parole arabe spontanée reste encore très peu abordée au niveau de la recherche scientifique. Durant les deux dernières décennies les efforts ont été plutôt concentrés sur la réalisation des analyseurs morphologiques et syntaxiques pour l'arabe tel que (Ouersighni, 2001). Malgré l'importance de la représentation et de l'analyse sémantique pour la réalisation de n'importe quel système de compréhension, il n'existe que quelques travaux qui s'intéressent à ce domaine en vue du traitement automatique de la langue arabe écrite et non pas parlée tels que (Haddad et al., 2005), (Meftouh et al., 2001), etc. Dans cet article, nous présentons le modèle de langage stochastique employé pour l'analyse sémantique de la parole arabe spontanée dans le cadre d'une application finalisée, ainsi que les résultats d'évaluation obtenus.

2 L'application finalisée considérée

2.1 Le domaine de l'application

Pour tester et estimer les paramètres du modèle de langage stochastique, nous avons utilisé un corpus représentant le domaine des renseignements ferroviaires. La principale raison de ce choix est la taille statistiquement représentative du corpus d'apprentissage dont nous disposons (voir tables 1 et 2). Ce corpus a été collecté en demandant à cent personnes différentes de formuler des énoncés relatifs aux renseignements ferroviaires. Donc c'est un corpus simulé et non pas réel.

Domaine	Taille (Mo)	Nombre d'énoncés	Nombre de mots	Nombre de locuteurs
Renseignements ferroviaires	*3,4*	*10000*	*85900*	*1000*

Table 1 : Caractéristiques du corpus de point de vue volume.

Nature de la tâche	Renseignements sur les:				Réservations	autres
Taux de sa représentation	horaires	trajets	tarifs	durées	*10,41 %*	*40,64%*
	28,7 %	*9,37 %*	*16,66 %*	*3,12 %*		

Table 2 : Caractéristiques du corpus de point de vue contenu.

2.2 Le corpus d'apprentissage

Le modèle de langage va servir à attribuer à chaque mot de l'énoncé transcrit par le module de reconnaissance de la parole un couple de traits sémantiques noté TS. Chaque couple TS est constitué de deux traits élémentaires : TS = (classe sémantique TSC, trait micro sémantique TSM). Le premier trait sert à déterminer la classe sémantique à laquelle appartient le mot à interpréter. Par exemple, toutes les villes du réseau ferroviaire sont représentées par la classe sémantique مدينة "medina "(ville). Pour l'application considérée, nous avons utilisé en tout 12 classes sémantiques différentes (voir table 3 ci-dessous).

Classes sémantiques TCS	Exemples d'instanciations
طلب (demande)	متى (quand)- بكم (combien)- أحب (je veux)- يوجد (existe)- etc.
حركة (mouvement)	الذاهب (qui va)- اذهب (je vais)- يصل (arrive)- etc.
مؤشر_حركة (Indice_mouvement)	إلى (à)- نحو (vers)- عبر (à travers)- من (de)- etc.
مؤشر_توقيت (Indice_horaire)	بتاريخ (à la date)- الساعة (l'heure)- etc.
رمز (référence)	هذا (ce)- هاته (cette)- etc.
مدينة (ville)	سوسة (sousse)- تونس (Tunis)- etc.
ربط (liason)	و (et)- etc.
عدد_تذاكر (nombre_billets)	تذكرة (biellet)- مكان (place)– تذكرتين (deux billets)– etc.
حس (bruit)	أن (que)- نهاركم (journée)- etc.
نوع_التذكرة (type_billet)	مسترس - ل- ذهاب_ و_إياب - للصغار - للطلبة - etc.
شرط (condition)	لا_ تتجاوز_أعمارهم (qui_ne_dépassent_pas_l'age)- etc.
عدد (nombre)	2 – 1 etc.

Table 3 : Les classes sémantiques considérées.

La méthode d'identification ou d'extraction de ces classes est présentée dans le paragraphe suivant (*2.3*). En ce qui concerne le deuxième trait du couple TS, c'est un trait micro sémantique qui permet de différencier le sens des mots appartenant à une même classe sémantique. Par exemple, ce trait permet de distinguer une ville de départ d'une ville de destination dans un énoncé donné. Nous signalons que les mots synonymiques ou possédant un même rôle sémantique possèdent le même couple de traits TS. Le nombre total des traits micro sémantiques TMS utilisés est 20 traits, soit presque le double des TCS. Ces traits sont les suivants : طلب _ توقيت (demande_prix) – طلب _ ثمن (demande_générale) - عام _ لحظة (correspondance) – عبور (départ) – انطلاق (destination) – وجهة (demande_horaire) – درجة (classe) – etc. Ainsi, pour estimer (moment) – ساعة (heure) – تاريخ (date) – يوم (jour) – les paramètres du modèle de langage stochastique, nous avons crée un corpus d'apprentissage (voir figure 1). Ce corpus a été obtenu en étiquetant au début manuellement une quantité (500) des énoncés du corpus collecté par un expert humain. Le principe d'étiquetage est d'attribuer à chaque mot significatif pour l'application un couple TS tel que défini ci haut. Les mots non significatifs ou vides sont éliminés lors de la phase du prétraitement du corpus

initial, et certains mots sont regroupés en une seule entrée. L'élimination des mots vides nous a permis de simplifier la complexité et réduire la taille du modèle. Ensuite, nous avons appliqué ce modèle pour l'étiquetage sémantique des 9000 énoncés restants, et ce par groupes de 500. Entre chaque étape d'étiquetage automatique, nous avons procédé à une vérification des résultats obtenus et une correction des paramètres a été établie chaque fois qu'il y a une détection d'erreurs. Enfin, les 500 énoncés restants nous ont servi pour l'évaluation de la performance du modèle. Ainsi, 95% du corpus a été consacré à l'apprentissage et 5% aux tests.

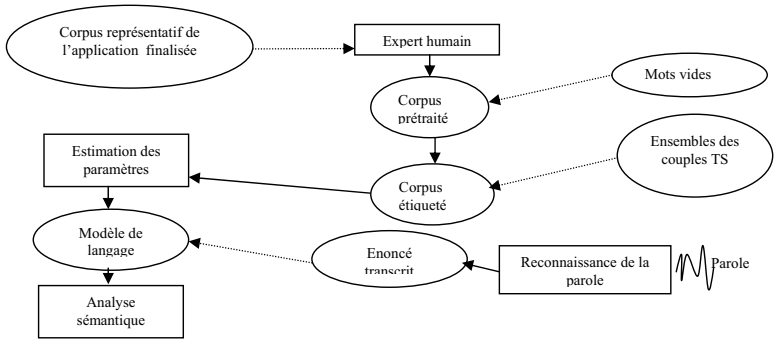

Figure 1 : Principe de l'estimation des paramètres du modèle de langage.

Les flèches en pointillés dans la figure 1, correspondent aux informations qui dépendent du domaine de l'application à modéliser.

2.3 L'extraction des classes sémantiques

Pour extraire les classes sémantiques de l'application, nous avons appliqué l'algorithme des K-means proposé par (McQueen, 1967), en utilisant l'information mutuelle moyenne IMm de (Rosenfeld, 1994) au lieu de la distance euclidienne pour mesurer la distance sémantique entre les différents mots du vocabulaire de l'application finalisée. Ceci, nous a amené à remplacer dans l'algorithme le critère d'évaluation $arg\ min_{j=1,\ldots,k}\ d^2(m_i, cgj)$ par $arg\ max_{j=1,\ldots,k}\ d(m_i, cgj)$ (voir figure 2). A part que cet algorithme, permet de faciliter la tâche d'identification des classes sémantiques, il a l'avantage d'être :

- Rapide face à des données de taille importante, puisqu'il converge à une vitesse linéaire de l'ordre de O(n.k.t) ; où n, k et t désignent respectivement le nombre des mots à classer, le nombre des classes sémantiques et le nombre d'itérations maximales.
- Et simple à implémenter.

Présentation de l'algorithme des k-means : -
Choisir d'une manière arbitraire les centres de gravité (cg1, cg2, cg3, …, cgk) des k classes sémantiques (cs1, cs2, cs3, …, csk).
Début
- *Etiquette* :
  Pour tout mot mi de m_1 à m_n faire
    Chercher la classe csk du mot mi en question :
      csk = arg $max_{j=1,\ldots,k}$ d(m_i, cgj) ;

où, d (m_i, cgj) = IMm(mi, cgj) = P(mi, $\overline{cgj}$)× Log [P(mi / cgj) / P(mi).P(cgj)] + P($\overline{mi}$, $\overline{cgj}$)× Log [P($\overline{mi}$ / $\overline{cgj}$) / P($\overline{mi}$).P($\overline{cgj}$)] + P(mi, $\overline{cgj}$)× Log [P(mi / $\overline{cgj}$) / P(mi).P($\overline{cgj}$)] + P($\overline{mi}$,

cgj)× Log [P(mi / cgj) / P(mi).P(cgj)]
Recalculer le centre de gravité de la classe csk :
cgk = $1/N_k \sum_{mi \in csk}$ mi ; où N_k désigne dans cet algorithme le nombre de mots dans la classe csk.
Fin Pour.
- Arrêt du traitement si les centres de gravité sont inchangés.
- Retourner à *Etiquette* sinon.
Fin

Figure 2 : l'algorithme des k-means en utilisant l'IMm comme métrique.

Cependant, le problème principal de cette méthode est la dépendance du résultat du classement final des informations données en entrée (les k centres de gravité des k classes sémantiques à déterminer sont choisis d'une manière totalement arbitraire). Cette limite ne pose pas de problèmes pour nous, puisque nous avons utilisé cette méthode rien que pour aider et donner une idée à l'utilisateur (surtout si cet utilisateur n'est pas un expert du domaine) sur la classification possible des mots de l'application d'un point de vue sémantique. Cependant les cartes auto organisatrices de (kohonen, 1989) offrent une alternative efficace, pour ceux qui cherchent des meilleurs résultats de partitionnement (Jamoussi, 2004).

3 Modélisation stochastique

3.1 Description du système de compréhension

Le système de compréhension conçu permet de construire la représentation sémantique d'un énoncé, sous la forme d'un ensemble d'associations attributs/valeurs (ou formulaire), comme le montre l'exemple suivant : Enoncé transcrit : أريد حجز مكان بالقطار الذاهب إلى تونس. ″ouridou hajza makan bilqitar athaheb ila tuwnis″ → Je veux réserver une place dans le train allant à Tunis.

Représentation sémantique :

نوعية (Type) = طلب حجز (demande de réservation)
مدينة_انطلاق (ville_ départ) = &Villecourante
يوم_انطلاق (jour_ départ) = ?
ساعة_انطلاق (heure_ départ) = ?
مدينة_وجهة (ville_ destination) = تونس (Tunis)
عدد_مقاعد (nombre_places) = 1

La figure 3 ci-dessous, présente l'architecture générale du système de compréhension. On remarque bien que la déduction du sens d'un énoncé par ce système est le résultat de l'accomplissement des traitements successifs suivants :
- La segmentation de l'énoncé transcrit par le module de reconnaissance de la parole : ce traitement permet d'identifier les mots ainsi que les différentes phrases du message du locuteur. Un même message peut être constitué d'un ou plusieurs requêtes à la fois. D'où, il est nécessaire que le système puisse identifier les différentes requêtes du message, afin qu'il puisse interpréter la demande de l'utilisateur dans toute son intégralité.
- Le prétraitement de l'énoncé : ce prétraitement consiste comme pour le prétraitement du corpus collecté à éliminer par exemple les mots vides, à regrouper certains mots en une seule entrée, etc. Ce modèle permet de simplifier la complexité de la tâche de compréhension.
- Le décodage sémantique de l'énoncé : c'est-à-dire l'étiquetage de chaque mot de l'énoncé prétraité avec les couples TS correspondants.
- La construction du sens de l'énoncé, cette étape correspond à la phase de génération de

l'ensemble des paires attribut/valeur (ou formulaire).
Le décodage sémantique des énoncés prétraités repose sur un modèle de langage stochastique qui permet d'encoder les règles de la grammaire (*voir paragraphe suivant*) et sur un lexique sémantique décrit dans un fichier et contient tous les mots du vocabulaire de l'application. Ce lexique est un ensemble d'associations de la forme : Mot M / TS décrivant le sens du mot + P(W / TSC, TSM) qui est la probabilité d'utilisation de TS = (TSC, TSM) pour la description du sens du mot M.

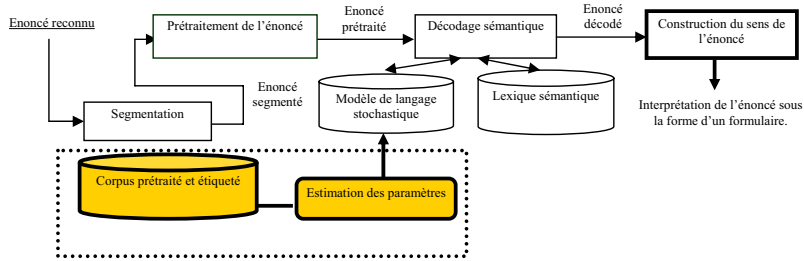

Figure 3 : Architecture du système de compréhension

3.2 Le modèle de langage

3.2.1 Le principe du décodage

Le modèle de langage que nous présentons ici, permet d'attribuer à chaque mot significatif un couple TS permettant de décrire sons sens. Comme nous l'avons signalé auparavant, nous avons choisi de représenter ce modèle à l'aide d'un modèle de Markov caché. Le principe du décodage sémantique est le suivant :
Nous considérons un énoncé constitué d'une suite de n mots : W = w1 w2 … wn
Cette suite de n mots est réduite à une suite de m mots après la phase du prétraitement de l'énoncé (élimination et regroupement de certains mots), où m ≤ n : W = w1 w2 … wm
Supposons que cette suite a été décodée via la suite de m couples de traits sémantiques suivante:TS=TS1TS2…TSm, ou encore TS= (TSC1,TSM1)(TSC2,TSM2)…(TSCm, TSMm).
Le but est alors de trouver les meilleures suites TS' connaissant W. Cette probabilité est calculée grâce au critère du maximum a posteriori : $P(TS' / W') = \text{Max}_{TS} P(TS / W) = \text{Max}_{TSC \times TSM} P(TSC, TSM / W)$
Ce qui donne en appliquant la formule de Bayes : $P(TSC, TSM / W) = P(W / TSC, TSM) \times P(TSC, TSM) / P(W)$
Nous avons ensuite utilisé l'algorithme de Viterbi (Rabiner et al., 1986), pour réaliser ce décodage.

3.2.2 La topologie du modèle

Nous avons considéré un modèle de Markov caché (HMM) à un seul niveau pour réaliser notre décodeur (voir figure 4). Chaque état du modèle markovien représente un couple TS et

les probabilités de transitions représentent les probabilités de passage d'un TS vers un autre. L'interprétation d'un mot dépend du contexte de l'énoncé, c'est-à-dire des relations de dépendances qu'il entretient avec les autres mots de l'énoncé. Comme ll montre la figure 4 suivante, nous avons considéré un HMM avec des probabilités tri-grammes de transitions entre les couples de traits sémantiques TSi des mots. Ce modèle contribue ainsi à la prédiction d'un couple de traits sémantiques TSi à partir des deux couples précédents TSi-1 et TSi-2.

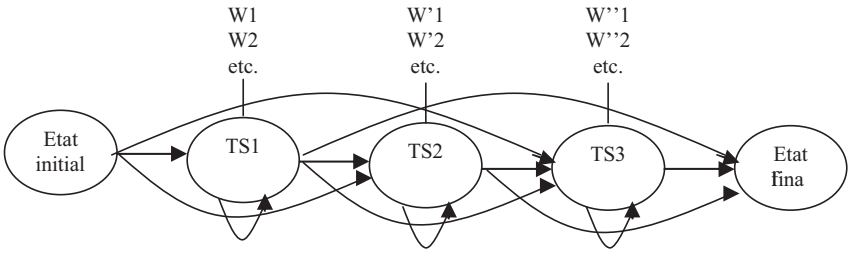

Figure 4 : Exemple de modélisation à l'aide d'un modèle de Markov caché à un niveau avec des probabilités tri-grammes de transitions entre les TSi

La réalisation de ce modèle nécessite principalement deux types d'informations :
- La manière d'agencement des couples TSi entre eux, sous la forme de probabilités tri-grammes de transitions entre les TSi :
P(TSi / TSi-1, TSi-2) = N (TSi, TSi-1, TSi-2) / N (TSi-1, TSi-2) ; où N (TSi, TSi-1,TSi-2) (resp. N (TSi-1, TSi-2)) est le nombre d'occurrence de TSi, TSi-1 et TSi-2 (resp. TSi-1 et TSi-2) ensemble.
- Et la probabilité d'émission de chaque mot du vocabulaire de l'application par chacun des couples TS définis. Un mot peut être décrit sémantiquement par plusieurs couples TS.
P (W/ TS) = N (W, TS) / N (TS) ; où N (W, TS) est le nombre fois de description de W par TS et N (TS) est le nombre total d'utilisation de TS.

3.2.3 Amélioration du modèle

En remarquant que ce n'est pas obligatoirement les mots précédant immédiatement le mot à interpréter qui ont une influence sémantique sur ce dernier, nous avons décidé d'employer lors de la phase de décodage du sens d'un mot que les TS des deux mots possédant la plus grande affinité sémantique avec celui-ci. Pour atteindre cet objectif, nous nous sommes basés sur la notion d'information mutuelle moyenne (Rosenfeld, 1994) qui permet de calculer le degré de corrélation ou de co-occurrence de deux mots donnés. Cette méthode nous a permis de ne plus utiliser systématiquement les TS des deux mots qui précédent immédiatement le mot à décoder.

4 Application du modèle et résultats

Pour tester la performance du modèle stochastique défini, nous avons utilisé les 500 énoncés du corpus collecté qui n'ont pas été employés lors de la phase d'estimation des paramètres du modèle de langage stochastique (*voir paragraphes 2.1 et 2.2*). Nous avons utilisé comme mesures de performances : - -

Le nombre total de mauvaises interprétations N_f défini comme suit : $N_f = N_{CS} + N_{MS}$, où N_{CS} et N_{MS} sont respectivement le nombre de TSC et le nombre de TSM incorrectement attribués par le système aux mots de l'énoncé.

- Le taux d'erreur du décodage sémantique : $Taux_{erreur} = N_f / N$; où N est le nombre total de traits TSC et TSM attribués à l'énoncé à interpréter.
- Le taux de précision est : $Taux_{précision} = N_C / N$; où N_C est le nombre des traits TSC et TSM correctement attribués.

La figure 5 suivante, présente les taux d'erreur et de précision trouvés. Ces taux sont répartis selon le type de renseignement demandé par l'utilisateur : demande de réservation (DR), ou de renseignements sur le trajet (DT), l'horaire (DH), le prix (DP), ou la durée du voyage (DD). On peut toujours aussi relever le taux d'erreurs des énoncés incorrectement décodés sémantiquement, en considérant le rapport entre les énoncés mal interprétés et le nombres total d'énoncés considérés dans le test (ici 500).

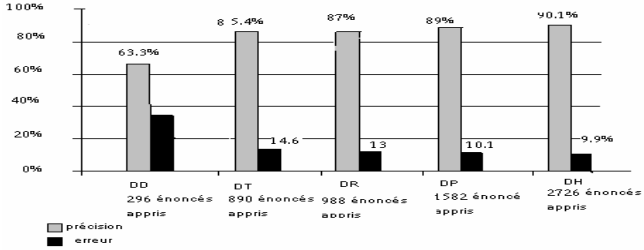

Figure 5 : Taux d'erreur et de précision selon le type de la demande de l'utilisateur et le nombre d'énoncés appris.

Le taux d'erreur réellement trouvé lors de la mesure de la performance de notre système est de l'ordre de 21,1%. En analysant davantage les résultats, nous avons conclu qu'un mauvais décodage est obtenu chaque fois qu'il y a un manque de données d'apprentissage. La figure 5 ci-dessus illustre bien ceci. En effet, d'après cette figure, nous remarquons que les résultats de décodage sont bons dans presque tous les types de renseignements demandés par l'utilisateur (DT, DR, DP et DH). Le plus mauvais décodage correspond aux énoncés de type DD. Ceci est dû au fait, que le nombre des énoncés DD considérés (3,12% du corpus) lors de la phase d'apprentissage du modèle de langage est insuffisant. En effet, nous avons constaté que seulement à partir de 1000 énoncés appris que notre système devienne performant. A partir de ce seuil, le taux d'erreur est inférieur à 11%. Au dessous de la barre de 500 énoncés, les résultats deviennent inacceptables. Le taux d'erreurs atteint 36,7% pour 296 énoncés appris, alors qu'il se restreint à 9,9% pour 2726 énoncés appris (voir figure 5). Donc, une mauvaise interprétation par notre système est due essentiellement à un manque de données d'apprentissage, et non pas au type ou à la topologie du modèle de langage utilisé. Nous avons aussi comparé ce modèle de langage employé par rapport à un modèle de langage avec des probabilités bi-grammes de transitions entre les TSi (1) et un modèle de langage avec des probabilités tri-grammes de transitions sans amélioration (2) (c-à-d sans considération des TS

des 2 mots influant sémantiquement sur le mot à interpréter). Nous avons trouvé que modèle (1) est efficace seulement lorsque le corpus d'apprentissage n'est pas assez volumineux (voir figure 6). En effet, plus l'ordre n d'un modèle n-grammes est petit, moins on a besoin de données d'apprentissage. Donc le modèle (1) peut être une alternative efficace au modèle utilisé (avec tri-grammes de transitions amélioré), dans le cas où on ne dispose pas de corpus assez volumineux représentatif du domaine de l'application à modéliser. Mais nous avons constaté que dés qu'il y a occurrence d'hésitations ou de mots inconnus précédant le mot à interpréter les modèles (1) et (2) deviennent aussi inefficaces.

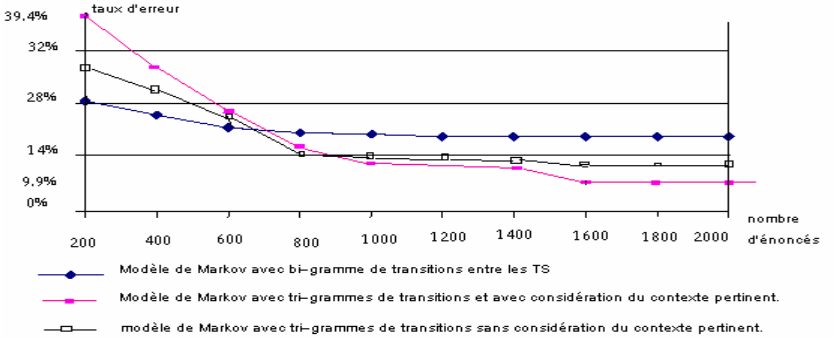

Figure 6 : Résultats de décodage obtenus en utilisant des modèles de Markov avec bi-grammes et tri-grammes de transitions avec et sans considération du contexte pertinent.

5 Conclusion

Nous avons présenté dans cet article le modèle de langage stochastique que nous avons employé pour le décodage sémantique de la parole arabe spontanée. Pour cela, nous avons utilisé un modèle de Markov caché à un seul niveau, avec des probabilités tri-grammes de transitions entre les couples de traits sémantiques TS. L'évaluation du modèle, en l'appliquant dans le domaine des renseignements ferroviaires a montré son efficacité. Nous avons atteint un taux de précision de l'ordre de 90,1% avec 2726 énoncés appris de type demandes d'horaires. Nous avons montré qu'en cas de manque de données d'apprentissage, un modèle de Markov caché à un seul niveau, avec des probabilités bi-grammes de transitions entre les TS est plus puissant. Ceci est vrai malheureusement que dans le cas d'énoncés non spontanés, c'est-à-dire ne contenant ni des hésitations ni des mots inconnus. Pour identifier les couples TS à employer pour l'interprétation des mots de l'énoncé, nous avons employé l'information mutuelle moyenne IMm de (Rosenfeld, 1994). Pour faciliter la tâche d'extraction des traits TSC d'une application, nous avons utilisé l'algorithme de partitionnement des K-means proposé par (McQueen, 1967). Cependant comme nous l'avons déjà signalé, nous avons utilisé cette méthode rien que pour aider et donner une idée à l'utilisateur sur la classification possible des mots de l'application d'un point de vue sémantique. Cependant les cartes auto organisatrices de (kohonen, 1989) offrent une alternative efficace, pour ceux qui cherchent des meilleurs résultats de partitionnement (Jamoussi, 2004).

Références

ANTOINE J-Y., GOULIAN J., VILLANEAU J. (2003), Quand le TAL robuste s'attaque au langage parlé: analyse incrémentale pour la compréhension de la parole spontanée, Actes de *TALN*.

Baum L.E., Petrie T., Soules G., Weiss N. (1970), A maximisation technique occurring in statistical analysis of probabilistic functions in Markov chains, *The Annals of Mathematical Statistics*.

BOUSQUET-VERNHETTES C. (2002), Compréhension robuste de la parole spontanée dans le dialogue oral homme-machine – Décodage conceptuel stochastique, Thèse de doctorat de *l'université de Toulouse III*.

HADDAD B., YASEEN M. (2005), A Compositional Approach Towards Semantic Representation and Construction of ARABIC, Actes de *LACL*.

JAMOUSSI S. (2004), Méthodes statistiques pour la compréhension automatique de la parole, Thèse de doctorat de *l'université Henri Poincaré*.

Kohonen T. (1998), Self-organisation and associative memory. Berlin, Spring-Verlag.

McQueen J. (1967), Some methods for classification and analysis of multivariate observations, Actes de *the Berkeley Symposium on Mathematical Statistics and Probability*.

MEFTOUH K., LASKRI M.T. (2001), Generation of the Sense of a Sentence in Arabic Language with a Connectionist Approach, Actes de *AICCSA*.

MINKER W. (1999), *Compréhension automatique de la parole spontanée*, Paris, L'Harmattan.

OUERSIGHNI R. (2001), A major offshoot of the Dinar-MBC project: AraParse, a morphosyntactic analyzer for unvowelled Arabic texts, Actes de *ACL/EACL*.

Rabiner L.R., Juang B.H. (1986), Introduction to Hidden Markov Models, *IEEE Transactions on Acoustics, Speech and Signal processing*.

ROSENFELD R. (1994), Adaptive statistical language modelling: A maximum entropy approach., Thèse de doctorat de *l'université de Carnegie Mellon*.

Schwartz R., Miller S., Stallard D., Makhoul J. (1996), Language Understanding Using Hidden Understanding Models, Actes de *ICSLP*.

SENEFF S. (1992), Robust parsing for spoken language systems, Actes de *ICASSP*, 189-192.

Van Noord G., Bouma G., Koeling R., Nederhof M.J. (1999), Robust grammatical analysis for spoken dialogue systems, *Natural Language Engineering 5(1)*.

Villaneau J., Antoine J.Y., Ridoux O. (2001), Combining Syntax and Pragmatic Knowledge for the Understanding of Spontaneous Spoken Sentences, Actes de *LACL'01*.