Stratégie de décision pour l'accès à des annuaires électroniques de grande taille

Dominique MASSONIÉ, Frédéric BÉCHET

LIA-CNRS (Laboratoire Informatique d'Avignon)
84911 AVIGNON Cedex 9, France
Tél. : ++33 (0)4 90 84 35 77 - Fax : ++33 (0)4 90 84 35 01
Mél : {dominique.massonie, frederic.bechet}@lia.univ-avignon.fr

ABSTRACT

This paper deals with the difficult task of recognition of a large vocabulary of proper names in a directory assistance application. After a presentation of the related work, it introduces a methodology for rescoring the N-best hypotheses generated by a first step recognition. Rather than augmenting the lexicon with alternate pronunciations, which is unsuitable for very large vocabularies of proper names, distortions of the canonical form are scheduled and performed according to decisions made by a rejection strategy. This strategy leads to a very significant improvement over the results obtained by a standard rejection method based on acoustic confidence scores only.

1. INTRODUCTION

La Reconnaissance Automatique de la Parole (RAP) constitue un outil de plus en plus important de l'interface homme-machine. La performance des systèmes actuels permet leur intégration à des services automatisés, en reposant sur un dialogue restreint ou adapté au domaine de la tâche. Le système doit être assuré de la qualité des informations qu'il collecte pour éviter de générer des réponses fantaisistes. Les conséquences d'une erreur sont plus frustrantes pour l'utilisateur que le doute émis lors d'une demande de confirmation, qui habilement employée permet d'augmenter la confiance du système [12].

Notre travail se place dans le cadre d'une application de renseignements téléphoniques, ou par extension, tout système d'assistance automatique avec accès oral par une entrée unique dans un très large vocabulaire (de type annuaire). En pratique, il s'agit d'obtenir le numéro de téléphone d'un correspondant après avoir indiqué ses coordonnées et surtout son nom-prénom. Le système de RAP est soumis à une grande perplexité en terme de choix du couple nom-prénom énoncé dans la requête, par rapport à la quantité de noms valides disponibles dans le lexique de l'application ou sinon inconnus. À l'échelle d'une ville ou d'un pays, le lexique possède une très grande taille avec une multitude de prononciations possibles et difficiles à envisager.

La variation du taux d'erreurs/mot perturbe la garantie d'un fonctionnement correct du dialogue, basé sur une forte confiance dans les paramètres du processus de décision. Des recherches menées dans le cadre du projet européen SMADA [2] ont établi la nécessité d'une stratégie pour accepter ou rejeter une hypothèse (le meilleur couple nom-prénom candidat). Des processus complémentaires basés sur différentes méthodes ou modèles peuvent être combinés et leurs résultats confrontés.

Notre contribution consiste d'une part à la mise en place d'une stratégie de décision empruntée au domaine de l'intelligence artificielle [4] et d'autre part aux types de moyens intégrés dans le processus de décision. Des décodeurs complémentaires basés sur des variations phonologiques sans ajout de prononciation alternative au lexique sont introduits. Chaque nouveau décodeur correspond à un type de variation phonologique (insertion, suppression, substitution) et s'applique sans connaissance linguistique *a priori*. Intégrés à la stratégie, ils modifient ou confirment les résultats établis, permettant de valider ou rejeter les candidats envisagés. Les décisions s'effectuent de manière incrémentale, en fonction de chaque source d'information, de contraintes et d'une base de règles. Ces données sont centralisées dans un 'tableau noir' qui assure la mise en œuvre des décisions. La stratégie est construite initialement à partir d'un corpus de développement. Son objectif consiste à limiter le nombre d'erreurs de faux rejets (le candidat correct n'est pas accepté) ou de fausses acceptations (un mauvais candidat n'est pas rejeté).

Dans les sections suivantes, nous présentons notre modèle de stratégie et ses méthodes d'apprentissage (section 2), ensuite nous explicitons son application à nos travaux (section 3) et enfin, les résultats obtenus sont comparés aux méthodes classiques (section 4).

2. STRATÉGIE ET DÉCISION

2.1. Définitions et objectifs

Les performances d'un système de RAP se mesurent au taux moyen d'erreurs d'insertion, substitution et suppression de mot, présents dans sa transcription écrite d'un ensemble de phrases prononcées. Toutes les erreurs n'ont pas le même impact sur la compréhension d'un message. Pour une application basée sur des couples de noms et prénoms, chaque erreur de reconnaissance est fatale. Ces erreurs se traduisent directement par le choix d'un mauvais candidat. Le taux d'erreurs du système de RAP correspond au taux de fausses acceptations d'un système sans procédure de décision. L'introduction de critères de décision permet de s'abstenir lorsque le système n'a pas confiance en sa réponse. Cette méthode ajoute un risque de faux rejets, lorsque l'application refuse d'accepter une hypothèse de candidat valide. Le taux d'erreur de référence devient la somme des faux rejets et des fausses acceptations. Le compromis entre faux rejet et fausse acceptation est intrinsèque au processus de décision. Faciliter les accepta-

tions réduit le risque de faux rejet au dépend des fausses acceptations et contraindre fortement les acceptations augmente le risque de faux rejet en limitant celui des fausses acceptations. Chaque application doit établir un point de fonctionnement acceptable en fonction de ses risques ou de ses besoins.

La mise en place d'une stratégie de décision nécessite des moyens et des méthodes pour contrôler ces moyens. Les informations collectées, mesures de confiances, confrontations de plusieurs décodeurs, permettent de définir des situations caractéristiques à partir des exemples d'un corpus de développement. La stratégie permet de faire appel aux ressources de manière sélective, en fonction des situations, jusqu'à retrouver un état caractéristique permettant une prise de décision. La décision définitive est prise en accord avec les objectifs, en terme de risques définis pour l'application.

Dans les sous-sections suivantes nous présentons les problèmes et contraintes de l'application (2.2), les méthodes de la stratégie (2.3) et son apprentissage (2.4).

2.2. Description des problèmes

Notre application d'accès oral à un annuaire cherche à minimiser le taux de fausses acceptations. Dans le cas où le système ne parvient pas à prendre une décision, l'utilisateur peut être pris en charge par un opérateur.

La stratégie doit permettre de classer chaque hypothèse avec la situation caractéristique qui minimise son risque de fausse acceptation. L'application décide ensuite de prendre ou non ce risque. Les moyens mis en œuvre pour établir les meilleures caractéristiques d'un candidat peuvent se décomposer en étapes. Les situations évoluent de manière itérative pour ne pas faire appel à toutes les ressources au même instant ou les consommer inutilement. Dans une application pratique, l'augmentation des moyens se traduit par une augmentation de la durée d'attente pour l'utilisateur. Notre stratégie doit s'adapter aux contraintes du système et limiter sa consommation, en favorisant au plus tôt les décisions les plus sûres. Les décisions repoussées sont plus indéterminées, elles nécessitent plus de ressources et leur risque de fausse acceptation augmente. Le déclenchement d'un processus très discriminant, mais ne s'appliquant qu'à un nombre limité de cas mérite d'être repoussé au profit d'un autre à la plus large couverture. L'objectif de minimisation du risque reste toujours le même pour l'application, mais la stratégie doit s'adapter et ordonnancer ses processus en fonction de leur utilité.

2.3. Architecture du processus de décision

L'architecture proposée pour évaluer progressivement les hypothèses est basée sur un modèle de 'tableau noir' développé en intelligence artificielle et appliqué en RAP [4]. Le 'tableau noir' centralise les informations (figure 1) et génère les décisions à partir de règles. Les types d'action possibles dans notre application sont :

– produire un résultat en sortie,
– rejeter l'entrée,
– exécuter un processus complémentaire,
– évaluer la situation du 'tableau noir'.

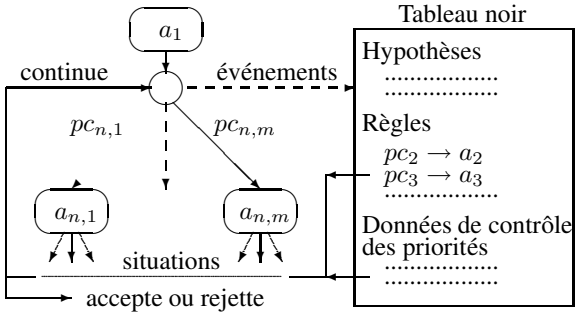

FIG. 1: Le 'tableau noir' centralise des règles (la stratégie sous une forme précondition → action), des hypothèses (les résultats des décodages et événements envisageables) et des contraintes (les priorités et pondérations de contrôle propres à l'application). Le fonctionnement du 'tableau noir' génère un arbre où chaque nœud n ajoute un traitement complémentaire au processus de décision, satisfaisant au mieux les contraintes parmi m actions possibles. Le résultat final consiste à accepter ou rejeter le meilleur candidat issu du décodage initial de l'action a_1.

Soit $A = \{a_i\}$ l'ensemble des actions, a_i étant exécutée lorsque la précondition pc_i est satisfaite. Une précondition est vraie, lorsque les résultats (hypothèses) correspondent à sa description logique de la situation. On note chaque règle $pc_i \rightarrow a_i$ (figure 1). Les événements possibles résultant de l'action a_i sont notés $\{E_{ij}\}$ et leur conséquence associée $\{c_{ij}\}$. Chaque conséquence c_{ij} correspond à un coût pour l'action, tel que

$$E_{ij} = \begin{cases} \text{acceptation correcte} & \text{coût } 0, \\ \text{fausse acceptation} & \text{coût } \alpha, \\ \text{faux rejet} & \text{coût } \beta, \\ \text{rejet correct} & \text{coût } 0, \\ \text{poursuit traitement} & \text{coût } 0. \end{cases} \quad (1)$$

La probabilité d'apparition de l'événement E_{ij} à la suite de l'action a_i est définie par $P(E_{ij}|a_i)$. Le choix de la meilleure action a^* à entreprendre est calculé en fonction de ses perspectives et de son coût. On a,

$$a^* = \arg\min_i \mu(a_i) \quad (2)$$

avec

$$\mu(a_i) = \sum_{j=1}^{J} c_{ij} \cdot P(E_{ij}|a_i) \quad (3)$$

Les poids attribués aux coûts d'une action reposent en théorie uniquement sur la qualité de la décision finale du système. Les meilleures actions se distinguent par un plus petit coût, mais peuvent être pénalisées en fonction de leur couverture ou de la quantité de moyens leur étant nécessaire.

2.4. Apprentissage de la stratégie

Les probabilités des événements, les coûts, les risques de chaque action sont calculés à partir de données de développement. Une stratégie S correspond à un ensemble d'actions $S = a_1, a_2 \ldots a_I$, dont le coût total C_S se mesure au nombre d'erreurs de décision sur son corpus de

développement. D'après l'équation 2, soit,

$$C_S = \sum_{i=1}^{I} \mu(a_i)$$

À partir de l'équation 3 et en utilisant les pondérations en 1, on exprime $\mu(a_i)$ tel que

$$\mu(a_i) = \alpha \cdot P(FA_i|acceptation) + \beta \cdot P(FR_i|rejet) \quad (4)$$

avec N_i étant le nombre total de traitements par a_i,

$$P(FA_i|acceptation) = \frac{|FA_i|}{N_i}$$

et

$$P(FR_i|rejet) = \frac{|FR_i|}{N_i}$$

La couverture correspond au rapport entre N_i et le nombre total de données de développement. L'objectif global de la stratégie consiste à minimiser le risque. L'objectif local de chaque étape consiste à minimiser le taux de fausse acceptation et maximiser la couverture.

Les règles du processus de décision sont des couples précondition-action simples : soit des confirmations ou permutations de positions dans les listes de résultats (exemple table 1), soit des variations de score entre hypothèses ou par rapport à des seuils. La génération de la stratégie est basée sur la construction d'un arbre qui intègre à chaque nœud la meilleure règle possible parmi celles encore disponibles. Le 'tableau noir' est mis à jour dans une des branches en fonction de la nouvelle situation (figure 1). Le nombre de données disponibles et leur qualité diminue en fonction de la profondeur. Une stratégie satisfaisant les contraintes imposées est ainsi obtenue.

La recherche du meilleur ordonnancement ou de la stratégie optimale dépasse le cadre de ce travail. La section suivante présente les moyens utilisés dans notre stratégie, basés sur la robustesse de la reconnaissance entre différents décodeurs confrontés à des distorsions.

3. GÉNÉRATION DYNAMIQUE DE VARIANTES DE PRONONCIATION

3.1. Prononciations alternatives

La modélisation du vocabulaire des systèmes de RAP, avec l'apprentissage des variations possibles et leur phonétisation est un problème largement abordé dans la littérature [13, 3, 10, 14]. L'augmentation du nombre de prononciations alternatives génère une explosion combinatoire de l'espace de recherche, sans pour autant garantir la présence de toutes les formes valides de chaque mot. Généralement dans le cas des annuaires, une forme phonétique canonique de chaque nom-prénom est produite à partir des modèles utilisés dans les systèmes de synthèse vocale. Une série de déviations plausibles est ensuite appliquée lors du décodage pour trouver un alignement avec la séquence effectivement prononcée. Une ou plusieurs passes permettent d'extraire une liste ordonnée de N candidats voisins, les *N-meilleurs*.

Une méthode de lexique dynamique utilisant une représentation des mots en fonction de leur contexte, puis sélectionnée selon un arbre de décision, est présentée en [8]. De nouveaux paramètres peuvent être pris en compte, comme la vitesse d'élocution [9]. Différentes unités acoustiques, syllabiques ou sub-phonétiques [6], sont applicables à la modélisation. L'apprentissage des différents modèles acoustiques nécessite des corpus audio transcrits, avec la forme canonique des mots alignée sur le signal de parole. Des modèles de phonèmes contextuels entraînés à partir d'une grande quantité de données d'apprentissage intègrent des phénomènes locaux tels que l'élision de voyelle ou la substitution de phonème [11]. Par contre, la suppression d'une syllabe doit être prise en compte par une prononciation alternative.

3.2. Décodeurs multiples

Un même signal de parole peut subir différentes formes de traitements, au niveau de la paramétrisation des observations acoustiques ou du paradigme de la transcription (Modèles de Markov Cachés, Réseaux de Neurones) [5]. Ainsi, plusieurs connaissances peuvent être modélisées et utilisées avec différentes méthodes et enfin combinées pour obtenir l'hypothèse finale. La difficulté consiste à exploiter les caractéristiques de chaque décodage de manière complémentaire, en utilisant des paramètres tels que les scores, la position des *N-meilleurs* ou la qualité moyenne des résultats d'un décodeur. Les campagnes d'évaluations de systèmes de RAP organisées par NIST ont montré que la fusion des résultats entre plusieurs systèmes obtient de meilleurs résultats que le meilleur d'entre eux [7].

Dans notre application d'assistance d'accès à un annuaire, la modélisation du lexique constitue la partie la plus sensible du décodage. Reconnaître la suite de phonèmes d'un nom-prénom dépend de sa prononciation et des confusions du système, ces deux types d'imprécisions étant difficilement distinguables. Pour cette raison, un décodeur principal et plusieurs variantes sont utilisés dans notre application. Le décodeur D_1 produit un treillis de phonèmes et une liste de *N-meilleurs* à partir d'un modèle de mots et de déviations. Le décodeur D_2 recherche et rescore les *N-meilleurs* de D_1 en alignant leur forme canonique dans le treillis de phonèmes. Ensuite, trois décodeurs (basés sur l'algorithme $A*$ [1]) recherchent le meilleur alignement en s'autorisant une distance de Levenshtein de 1 avec la forme canonique. Le décodeur D_I est celui autorisant l'apparition d'une insertion quelconque dans la prononciation, D_D celui pour une suppression et D_S celui pour une substitution. Chacun des décodeurs produit des hypothèses suffisamment différentes pour que la fusion des réponses minimise leurs erreurs (table 1).

TAB. 1: Résultat et couverture de chaque décodage alternatif confirmant le décodage initial (même meilleur).

Confirme D_1	D_2	D_I	D_D	D_S
Taux d'erreur (%)	16.15	15.49	16.33	16.62
Couverture (%)	65.77	65.08	62.78	60.4

4. EXPÉRIENCES ET RÉSULTATS

Les expériences présentées ici concernent une stratégie de rejet, consistant à confirmer le résultat de D_1 ou à rejeter son entrée. Seul le meilleur candidat initial étant considéré, la performance de la stratégie est limitée au

taux d'erreurs du décodage initial D_1.

Les expériences sont menées sur un corpus de test indépendant des données d'apprentissage. Le lexique de l'application provient d'un annuaire interne à France Télécom contenant plus de 120k entités nom-prénom. Les résultats du décodeur principal D_1 donnent 70% de correct. Ceci implique, avec acceptation de tous les cas, un taux de faux rejets nul et un taux de fausses acceptations de 30%. Dans le cas de l'accès à un annuaire électronique, les fausses acceptations ont un très fort coût pour l'application. Les rejets par le système automatique peuvent être transmis à un opérateur humain.

Nos résultats en terme de fausse acceptation et de faux rejet sont présentés sur la figure 2. La courbe permet de décider un point de fonctionnement pour minimiser les fausses acceptations à un seuil acceptable pour l'application. Les gains observés sont significatifs par rapport aux autres méthodes de rejet classiques, telles que l'écart de score entre les deux premiers candidats ou le calcul de la probabilité à postériori. Notre méthode permet de conserver un taux de fausse acceptation inférieur à 10% pour plus de la moitié des données. Ceci à comparer avec les 30% d'erreurs de base obtenues avec le décodeur D_1 sans stratégie de rejet. Par conséquent en fonction d'une application attribuant un coût dix fois supérieur aux fausses acceptations qu'aux faux rejets, à partir de l'équation 4 en fixant $\alpha = 10$ et $\beta = 1$, le coût du système sans stratégie est de $C_{\overline{S}} = 30 \cdot 10 + 0 \cdot 1 = 300$ et $C_S = 10 \cdot 10 + 50 \cdot 1 = 150$. Soit un risque d'erreur réduit de moitier pour l'utilisateur.

5. CONCLUSION ET PERSPECTIVES

Nous avons présenté dans ce papier une méthode de stratégie de rejet des candidats issus d'un premier décodage, dans le cadre d'une application de reconnaissance de la parole appliquée à un vaste annuaire. Une première utilisation de cette stratégie, basée sur une même classe de décodeurs complémentaires a permis d'obtenir des résultats très encourageants.

Différents systèmes de reconnaissance, modélisant d'autres caractéristiques que les variantes phonologiques restent à envisager. L'amélioration et l'introduction de nouvelles règles doit permettre d'effectuer des choix plus complexes que le rejet, comme le rattrapage de candidats non premier ou la détection de listes sans candidat valide.

REMERCIEMENTS

Les travaux décrits dans ce papier ont été menés dans le cadre du projet européen SMADA. Nous remercions Alexandre Ferrieux et Denis Jouvet de France Télécom R&D pour la mise à disposition des données.

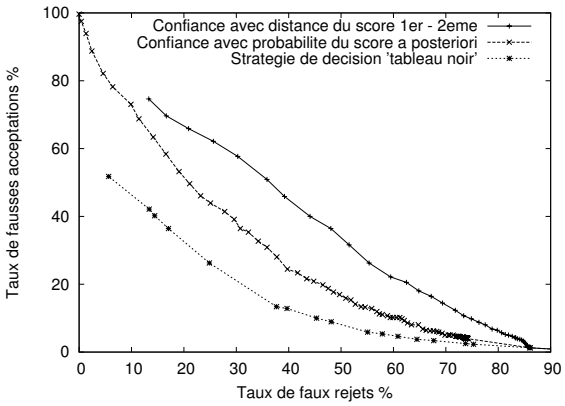

FIG. 2: Les résultats de notre stratégie de rejet par rapport aux méthodes classiques comme la probabilité à postériori ou l'écart entre les deux premiers.

RÉFÉRENCES

[1] F. Béchet, R. de Mori, and G. Subsol. Dynamic generation of proper name pronunciations for directory assistance. In *ICASSP'02*, Orlando, 2002.

[2] F. Béchet, E. Den Os, L. Boves, and J. Sienel. Introduction to the IST-HLT project speech-driven multimodal automatic directory assistance. In *ICSLP'00*, Beijing, 2000.

[3] N. Cremelie and J.P. Martens. In search of better pronunciations models for speech recognition. *Speech Communication*, 29(2-4) :115–136, 1999.

[4] R. de Mori, L. Lam, and M. Gilloux. Learning and plan refinement in a knowledge-based system for automatic speech recognition. *IEEE Trans. On Pattern Analysis and Machine Intelligence*, 9(2), 1987.

[5] Renato de Mori. *Spoken dialogue with computers*. Academic Press, 1998.

[6] S. Deligne, B. Maison, and R. Gopinath. Automatic generation and selection of multiple pronunciations for dynamic vocabularies. In *ICASSP'01*, Salt Lake City, 2001.

[7] J.G. Fiscus. A post-processing system to yield reduced word error rates : Rover. In *ASRU'97*, 1997.

[8] E. Fosler-Lussier. Contextual word and syllable pronunciation models. In *ASRU'99*, Keystone, Colorado, 1999.

[9] E. Fosler-Lussier and N. Morgan. Effects of speakingrate and word frequency on pronunciations in conversational speech. *Speech Communication*, 29(2-4) :137–158, 1999.

[10] Y. Gao, B. Ramabhadran, J. Chen, H. Erdogan, and M. Picheny. Innovative approaches for large vocabulary name recognition. In *ICASSP'01*, Salt Lake City, 2001.

[11] D. Jurafsky, W. Jianping, Z. Ward, K. Herold, Y. Xiuyang, and Z. Sen. What kind of pronunciation variation is hard for triphone to model ? In *ICASSP'01*, Salt Lake City, 2001.

[12] P. Natarajan, R. Prasad, R.M. Schwartz, and J. Makhoul. A scalable architecture for directory assistance automation. In *ICASSP'02*, Orlando, 2002.

[13] D. Neeraj, M. Weber, and J. Picone. Automated generation of N-best pronunciations of proper nouns. In *ICSLP'96*, Seattle, 1996.

[14] H. Strik and C. Cucchiarini. Modeling pronunciation variation for ASR : A survey of the literature. *Speech Communication*, 29(2-4) :225–246, 1999.