Correction interactive de transcriptions de parole par fusion de phrases

Mickael Rouvier, Benoit Favre, Frédéric Béchet *
Aix-Marseille Université, CNRS, LIF UMR 7279, 13000, Marseille, France
prenom.nom@lif.univ-mrs.fr

RÉSUMÉ

La clarification sous forme de dialogue permet d'aider à corriger les erreurs de RAP dans un système de traduction de parole automatique ainsi que dans d'autres applications interactives. Nous proposons d'utiliser des variantes de Levenshtein pour fusionner une phrase contenant une erreur et une phrase de clarification. Les erreurs de RAP qui pourraient nuire à l'alignement sont traitées par correspondance phonétique et une distance de plongement de mots est utilisée pour prendre en compte les synonymes en dehors des segments d'erreurs. Ces paramètres permettent une amélioration relative de 30 % du taux d'erreur de mots sur une sortie de RAP comparé l'absence de clarification. De plus, nous générons un ensemble de fusions potentielles et entraînons un réseau de neurones afin de sélectionner la meilleure fusion, permettant de sélectionner correctement 24 % de plus d'instance. Le système est utilisé dans le cadre du projet BOLT.

ABSTRACT

Interactive correction of speech transcripts

Clarification dialogs can help address ASR errors in speech-to-speech translation systems and other interactive applications. We propose to use variants of Levenshtein alignment for merging an errorful utterance with a targeted rephrase of an error segment. ASR errors that might harm the alignment are addressed through phonetic matching, and a word embedding distance is used to account for the use of synonyms outside targeted segments. These features lead to a relative improvement of 30 % of word error rate on ASR output compared to not performing the clarification. Twice as many utterance are completely corrected compared to using basic word alignment. Furthermore, we generate a set of potential merges and train a neural network on crowd-sourced rephrases in order to select the best merger, leading to 24 % more instances completely corrected. The system is deployed in the framework of the BOLT project.

MOTS-CLÉS : Correction d'erreur, Systèmes de dialogue, Détection d'erreurs, Transcription de la parole, Reranking, Fusion de phrases.

KEYWORDS: Error correction, Dialog systems, ASR error detection, Reranking, Levenshtein alignment.

Les systèmes de transcription automatique de la parole génèrent des transcriptions imparfaites à cause des conditions acoustiques, des mots hors vocabulaire ou des ambiguïtés venant d'un contexte lointain. Malgré les progrès récents sur la robustesse (Maas et al, 2012; Weninger et al, 2012), la reconnaissance en vocabulaire ouvert (Gerosa et Federico, 2009; Parada et al, 2011), la modélisation du contexte à long terme (Jonson, 2006; Mitchell et Lapata, 2009; Liu *et al.*, 2012), les systèmes de transcriptions font toujours des erreurs ayant un impact sur les tâches en aval.

Les systèmes interactifs offrent l'opportunité de corriger ces erreurs à l'aide de dialogues de clarifi-

*. Ce travail est partiellement financé par le projet DARPA HR0011-12-C-0016 sous la forme d'un contrat entre AMU et SRI International.

cation, comme dans les systèmes de dialogues en domaine restreint qui utilisent des confirmations implicites ou explicites (Shin et al, 2002; López-Cózar *et al.*, 2010). Néanmoins, ce n'est pas possible en domaine ouvert car il faudrait une bien meilleure modélisation sémantique du contenu des messages à transcrire.

Les efforts récents d'estimation de mesures de confiance (Yu et Deng, 2010; Seigel *et al.*, 2011), de détection de mots hors vocabulaire (Marin et al, 2012; Kombrink *et al.*, 2012; Parada *et al.*, 2010), de détection et de caractérisation automatique des erreurs (Bechet et Favre, 2013; Dufour *et al.*, 2012) offrent l'opportunité de localiser les segments d'erreurs et d'initier un dialogue de clarification de manière à améliorer la transcription. Ces systèmes de clarification peuvent demander à l'utilisateur de désambiguïser des homophones, épeler des mots hors vocabulaire ou reformuler une partie de leur phrase d'origine afin de corriger les erreurs (Prasad et al, 2012; Stoyanchev *et al.*, 2012). Au lieu de demander une reformulation complète, ces stratégies ciblent les erreurs à corriger de la même manière que le font les humains, générant des interactions naturelles et intuitives. En particulier, il est possible d'obtenir une transcription parfaite à l'aide de plusieurs questions de clarification.

Dans cet article, nous nous intéressons à la fusion d'une phrase où une erreur a été détectée et de la réponse de l'utilisateur à une question de clarification ciblant cette erreur. Nos contributions sont les suivantes :

— nous proposons une modification de l'alignement de Levenshtein en termes de topologie et de fonction de coût (phonétique, sémantique) pour prendre en compte les difficultés de la tâche de fusion ;
— nous générons plusieurs variantes de fusion dont la meilleure est sélectionnée par un perceptron multicouches tirant parti de caractéristiques riches ;
— le système est évalué sur un jeu de dialogues de clarification collecté dans le cadre du projet BOLT de traduction simultanée automatique.

1 Travaux connexes

Améliorer les transcriptions automatiques à l'aide d'interactions est principalement traité par des confirmations dans les systèmes de dialogue et des commandes d'édition dans les systèmes multimodaux comme la dictée vocale.

Il existe de nombreux travaux sur la confirmation dans les systèmes de dialogue (Shin et al, 2002). Dans sa plus simple forme, le résultat de la transcription automatique est resynthétisé et il est demandé à l'utilisateur si ce dernier est correct ou non. Les systèmes de compréhension à base de concepts reposent sur des scores de confiance sémantique pour ne confirmer que les concepts mal compris (Jung *et al.*, 2008). Si la modélisation sémantique est plus profonde, il est possible d'évaluer la cohérence des croyances du système pour en déduire des questions ciblées (Bohus et Rudnicky, 2009). Ces approches n'étant pas envisageables en domaine ouvert, certains travaux tirent parti des spécificités de la tâche pour cibler les confirmations, comme ceux de (Misu *et al.*, 2004) pour la recherche d'information.

Lorsque des interactions multimodales sont possibles, on peut tirer parti des autres modalités pour améliorer la transcription. Par exemple les systèmes de dictée vocale comme *Dragon Naturally Speaking* permettent de voir les mots de la transcription et de les corriger par des commandes spécifiques. De plus, des candidats pour la correction peuvent être générés à partir des réseaux de confusion pour faciliter leur sélection (Ogata et Goto, 2005; Huggins-Daines et Rudnicky, 2008; Laurent et al, 2011). Dans (Hoste *et al.*, 2012), une estimation des mots regardés par l'utilisateur permet de sélectionner ceux qui sont incorrects.

Dans la communauté du traitement automatique du langage naturel, la fusion de phrase a été explorée dans le cadre du résumé automatique, pour créer une version plus courte d'un ensemble de phrases se chevauchant en termes de contenu. La tâche est effectuée par fusion d'arbres syntaxiques (Filippova et Strube, 2008; Barzilay et McKeown, 2005) ou à l'aide d'une distance d'édition apprise automatiquement sur des corrections manuelles (Elsner et Santhanam, 2011).

2 Tâche

Nos travaux s'intéressent à la tâche de transcription automatique pour la traduction simultanée. Cette tâche induit les points suivants : le domaine est ouvert, les interactions doivent être orales pour permettre à l'utilisateur de garder un contact visuel, les interactions doivent être intuitives. Formellement, étant donné la transcription erronée d'une phrase d'**origine** et de la **réponse** à une question de clarification relative à un **segment d'erreur** de l'original, la tâche consiste à construire une nouvelle transcription capturant au mieux l'intention du locuteur. Les mots à l'intérieur du segment d'erreur sont considérés comme erronés et doivent être remplacés ; toutefois les mots hors du segment d'erreur doivent parfois aussi être modifiés par exemple pour satisfaire un nouveau contexte syntaxique. Voici un exemple d'interaction :

Parole d'origine	Où est la chambre **hyperbare** ?
ASR	où est *l'a* chambre **hyper barre**
Erreur détectée	hyper barre
Question	Pouvez-vous reformuler AUDIO(hyperbare) ?
Réponse	la chambre de décompression
Phrase générée	où est la chambre de décompression

Dans cet exemple, les entrées du système de fusion sont : la phrase originale où l'erreur détectée est masquée *"où est l'a chambre* $\langle err \rangle$" et la réponse *"la chambre de décompression*". La fusion de référence est *"où est la chambre de décompression*". Notez qu'ici une erreur supplémentaire sur *"l'a*" n'a pas été détectée par le système et doit donc être corrigée lors de la fusion.

Nous avons étudié le comportement des utilisateurs face aux dialogues de clarification ciblés du système de traduction simultanée créé dans le contexte du projet BOLT (Bechet et Favre, 2013). Les comportements les plus courants sont :

- la réponse remplace exactement le segment d'erreur ;
- la réponse contient des mots supplémentaires qui contextualisent l'opération d'édition ;
- la réponse est une reformulation complète de la phrase ;
- le contexte syntaxique de l'original doit être modifié pour y insérer la réponse (relative au lieu d'adjectif, par exemple) ;
- des mots du contexte sont reformulés par concision (anaphore pronominale, par exemple) ;
- des expressions permettent d'introduire la reformulation ("j'ai dit ...").

Les sections suivantes proposent un système pour effectuer la tâche de fusion étant donné ces comportements.

3 Système

L'idée générale de notre système est d'aligner les mots de la réponse à ceux de l'original et de ne remplacer dans l'original que ceux qui sont alignés avec le segment d'erreur. Pour traiter les

différents comportements de l'utilisateur, nous créons des variantes de systèmes fondées sur cette stratégie puis nous utilisons un classifieur pour sélectionner la meilleure variante comme fusion finale.

Les deux premiers systèmes triviaux sont le remplacement de l'original par la réponse (ciblant une reformulation complète) et l'insertion de la réponse à la place du segment d'erreur. Les autres variantes sont basées sur l'alignement de Levenenshtein appliqué dans le cadre d'automates transducteurs pondérés. Dans ce cadre, l'original et la réponse sont représentés par des accepteurs linéaires reconnaissant leur séquence de mots. Dans la phrase originale, le segment d'erreur est remplacé par une boucle sur un token spécial $\langle err \rangle$ de manière à pouvoir y aligner plusieurs mots de la réponse. En introduisant un troisième automate effectuant toutes les opérations d'édition possibles (insertion, suppression, substitution), il est possible par composition des trois automates d'obtenir l'alignement de plus faible coût entre l'original et la réponse. Soit A_o, l'original, A_c, la réponse et T_e le transducteur d'édition, le plus court chemin dans la composition $A_o \circ T_e \circ A_c$ donne l'alignement. La figure 1 détaille la structure de ces automates.

La fonction de coût d'édition est structurée de manière à regrouper les insertions/suppressions au sein de segments grâce à un coût supplémentaire α pour commencer un segment, puis un coût β pour continuer ce segment (Altschul et Erickson, 1986). De plus nous positionnons le coût d'alignement d'un mot avec le segment d'erreur à zéro afin de pousser les mots à y être alignés en priorité. En dehors du segment d'erreur, le coût de substitution γ est calculé à partir de la similarité entre les mots alignés :

— similarité wordnet : $\gamma = 0.5$ si les deux mots partagent un synset dans Wordnet, sinon $\gamma = 1$;
— similarité dans un plongement : γ est la similarité cosine entre les vecteurs représentant les mots dans un plongements à 300 dimensions. Ce plongement a été entrainé sur un corpus textuel du domaine en prenant la couche cachée d'un réseau de neurones entraîné à prédire chaque mot sachant son contexte (Mikolov et al, 2013) ;
— similarité phonétique : γ est la distance d'édition minimale entre toutes les paires de phonétisations des deux mots dans un lexique de transcription automatique (Lenzo, 2007).

Les variantes de systèmes sont créées en activant ou non ces différents coûts afin d'obtenir différentes propriétés de fusion.

4 *Reranker*

Le but du reranker est de sélectionner parmi l'ensemble des hypothèses (données par les systèmes de fusion) celle qui va générer la meilleure transcription selon les intentions de l'utilisateur. Nous arborons le problème de reranker comme un classifieur binaire : étant donné un ensemble de paramètres qui caractérisent la sortie d'un système de fusion, cette sortie correspond-elle bien aux intentions de l'utilisateur ?

Nous proposons d'extraire 14 paramètres pour caractériser la sortie d'un système de fusion :

— Identité du système de fusion
— Similarité du cosine entre l'hypothèse de fusion et la phrase originale
— Similarité du cosine entre l'hypothèse de fusion et la phrase de clarification
— Minimum, maximum et moyenne du score au mot donné par un réseau de neurones récurrents (*Recurrent Neural Network*, RNN)
— Longueur de la phrase du système de fusion
— Nombre de systèmes de fusion d'accord avec l'hypothèse de fusion
— Distance de Levenshtein entre l'hypothèse de fusion et la phrase originale
— Distance de Levenshtein entre l'hypothèse de fusion et la phrase de clarification

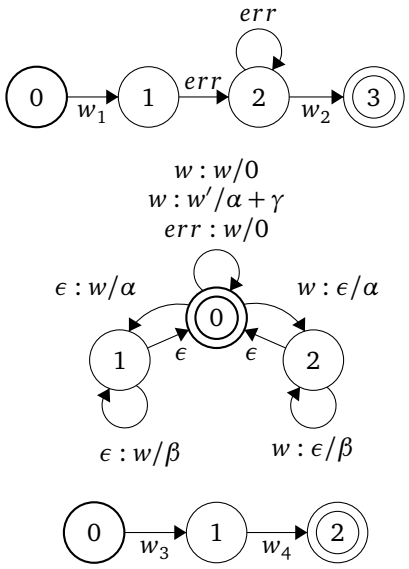

FIGURE 1 – Alignement par composition de transducteurs. L'accepteur du haut, A_o représente la phrase d'origine avec une boucle de symboles d'erreur à la place du segment d'erreur ; le transducteur du milieu T_e est un patron pour le transducteur d'édition pour une paire de mots (w, w') ; A_c est l'accepteur représentant la réponse de à la question de clarification. L'alignement est obtenu par le plus court chemin de la composition $A_o \circ T_e \circ A_c$. α est le coût de début d'une séquence d'insertions/supressions, β est le coût souffert pour ajouter un mot à cette séquence, et γ est le coût de substitution entre ces séquences.

— Différence de longueur entre l'hypothèse de fusion et la phrase originale
— Différence de longueur entre l'hypothèse de fusion et la phrase de clarification
— La phrase est-elle incluse dans l'originale ? Dans la clarification ?

Le reranker est un *Multi-Layer Perceptron* (MLP) doté de 6 couches (1 couche d'entrée, 4 couches cachées et 1 couche de sortie), entraîné via l'algorithme de *backpropagation*. Les couches d'entrée, cachées et de sortie sont composées respectivement de 14, 10 et 1 neurones. Dans la phase d'entraînement, le neurone de la couche de sortie est mis : à +1 si l'hypothèse donnée par le système de fusion correspond à la référence et à −1 dans l'autre cas. Nous utilisons un grand nombre de dialogues de clarification obtenus par *crowd-sourcing* (décrit dans la Section 5) comme corpus d'apprentissage, le RNN est entraîné sur le corpus TRANSTAC (corpus *in-domain*). Finalement, le système sélectionné parmi l'ensemble de systèmes de fusion donnés est celui qui obtient le meilleur score [1].

5 Expériences et résultats

L'approche développée dans ce papier est testée dans le cadre du projet BOLT Tâche b/c, conversation bilingue humaine-humaine par l'intermédiaire d'une machine. Le système joue le rôle d'un interprète qui peut prendre l'initiative de clarifier la saisie de l'utilisateur avant de la traduire.

1. Pour le MLP nous utilisons le toolkit FANN (http ://leenissen.dk/fann/wp) et pour le RNN nous utilisons le toolkit RNNLM (http ://www.fit.vutbr.cz/ imikolov/rnnlm).

Les systèmes de détection d'erreur de RAP (Reconnaissance Automatique de la Parole) et de TA (Transcription Automatique) sont gérés en amont pour détecter les segments d'erreur. Le module de dialogue demande alors à les reformuler. Le système de fusion, décrit dans ce papier, prend en compte la phrase originale et la phrase de clarification, dans le but de produire une meilleure transcription avant la traduction. Dans ce qui suit, seule la langue anglaise est traitée mais le traitement serait le même pour une autre langue.

Afin d'évaluer la qualité de notre système, nous avons produit deux corpus : un corpus de texte de reformulation collecté sur *Amazon Mechanical Turk* (AMT), principalement utilisé pour l'entraînement ; et un corpus de parole reproduisant le dialogue de clarification dans BOLT. Le premier corpus est composé d'un jeu de 900 phrases *in-domain* pour lesquelles nous demandons aux *Turkers* de reformuler aléatoirement une partie de la phrase (3 *Turkers* × 3 segments aléatoires × 100 phrases en entrée). Le jeu de données est ensuite sous-échantillonné de manière aléatoire en jouant sur les limites de clarification et d'erreur pour obtenir 11 175 phrases uniques. Le corpus de parole est composé de 70 phrases de dialogue pour lesquelles la phrase originale contient au moins un segment d'erreur lié au système de RAP. Ce corpus est très difficile pour le système de RAP car il contient un grand nombre de mots hors vocabulaire. La transcription automatique a été obtenue en utilisant le système de RAP du SRI basé sur les DNN. Ce système a été utilisé dans le cadre du projet BOLT et a obtenu sur les phrases originales un TEM de 30.6 % et 14.7 % sur les phrases de clarification. Dans la suite du papier, nous appelons *AMT* le corpus collecté par *crowd-sourcing* et *Parole* le corpus de parole audio.

Dans nos expériences, nous proposons 9 systèmes de fusions : *replace* et *insert* des systèmes *baseline*, un alignement de Levenshtein sans boucle d'erreur (*Align no-err-loop*), un alignement avec une boucle d'erreur (*Err-loop*), un alignement avec une pénalité affine d'insertion/suppression et une boucle d'erreur (*Err loop + affine gap* ; $\alpha = \beta = \gamma = 1$), le même système mais celui-ci aligne une séquence phonétique au lieu d'une séquence de mots (*Phonetic*), une pénalité affine d'insertion/suppression et une boucle d'erreur avec γ correspondant à la similarité phonétique entre les mots (*Phonetic + words*), le même système mais γ correspondant au minimum entre la similarité *Wordnet* et la similarité phonétique (*Phonetic + Wordnet*), le même système mais γ correspondant au minimum entre la similarité de plongement et la similarité phonétique (*Phonetic + embedding*).

Méthode	TCC	TEM
Replace (baseline)	45.94	20.24
Insert (baseline)	25.41	32.56
Align no err-loop	61.84	13.00
Err loop	69.76	07.30
Err loop + affine gap	75.02	05.37
Phonemes	51.86	10.51
Phonetic + words	77.60	04.74
Phonetic + Wordnet	80.02	03.99
Phonetic + embedding	78.70	04.42
Reranker	**88.36** ±0.72	**02.72** ±0.23
Oracle	96.18	00.56
(Bechet et Favre, 2013)	63.70	12.44

TABLE 1 – Les résultats des systèmes de fusion et du reranker sont reportés en termes de Taux de Classification Correcte (TCC) et Taux d'Erreur Mots (TEM) sur le corpus *AMT*. L'oracle est calculé systématiquement en sélectionnant la meilleure sortie des systèmes de fusion. Pour comparaison, nous donnons aussi les résultats de la méthode proposée dans (Bechet et Favre, 2013).

Les performances des systèmes de fusion sont évaluées selon deux métriques : (1) le Taux de

Classification Correcte (TCC) qui correspond au nombre de bonnes réponses données par le système de fusion sur le nombre de cas testés et (2) le taux d'erreur de mots (TEM) qui est le taux de mots incorrectement reconnus de l'hypothèse par rapport à la référence.

Le Tableau 1 donne les résultats obtenus par les différents systèmes de fusion sur le corpus *AMT*. On peut constater que tous les systèmes à base d'alignements améliorent de manière significative la *baseline*, conduisant à une réduction importante du TEM. Le système de fusion ayant obtenu les meilleurs résultats est : *Phonetic + Wordnet*. Il obtient un taux de classification correcte de 80.02 %. L'alignement sans aucune connaissance *a priori* sur la langue (sans lexique phonétique, ni représentation sémantique) obtient un taux de classification correcte de 75.02 %. Le reranker offre une amélioration supplémentaire de 10.4 % de précision et de 31.8 % de TEM par rapport au meilleur système de fusion. Nous notons que sur ce corpus, la méthode proposée dans (Bechet et Favre, 2013) permet d'améliorer uniquement la *baseline* et le système *Align no err-loop* (qui est un alignement basique).

Dans le Tableau 2, nous présentons les mêmes résultats sur le corpus *Parole*. Le comportement général des systèmes de fusion reste le même que sur le corpus de texte, excepté que le taux de classification correcte est plus bas sur les sorties de RAP. Pourtant, le TEM est grandement amélioré en utilisant la phrase de clarification puisqu'on observe un gain d'environ 10 points absolus. Sur le corpus *Parole*, le système *Phonetic + Wordnet* obtient de moins bons résultats que le système *Phonetic + embedding*, probablement parce que l'espace continu est plus robuste aux erreurs commises par le système de RAP. Les méthodes proposées dans (Bechet et Favre, 2013) donnent de bons résultats sur le texte de référence mais elles ne parviennent pas à améliorer le TEM sur la sortie de RAP, un facteur de motivation pour les futurs travaux.

Concernant le reranker, il obtient un taux de classification correcte proche de l'oracle mais le TEM n'est pas aussi élevé que sur le corpus *AMT*. Ceci suggère que (1) plusieurs stratégies de système de fusion doivent être explorées pour améliorer l'oracle, et (2) le reranker devrait être entraîné pour minimiser le TEM en plus de maximiser le taux de classification correcte.

	Ref.		RAP	
Méthode	TCC	TEM	TCC	TEM
Replace (baseline)	25.71	49.08	12.86	55.18
Insert (baseline)	37.14	28.09	08.57	46.52
Align no-err-loop	35.71	21.56	10.00	32.34
Err-loop	70.00	08.09	17.14	22.27
Err loop + affine gap	84.29	02.13	21.43	21.56
Phonemes only	74.29	04.68	15.71	21.56
Phonetic + words	85.71	**01.84**	21.43	21.13
Phonetic + Wordnet	82.86	02.70	18.57	21.28
Phonetic + embed.	85.71	**01.84**	21.43	20.99
Reranker	**87.57** ±1.36	01.93 ±0.58	**26.67** ±0.78	**20.97** ±0.58
Oracle	90.00	1.42	28.57	16.17
(Bechet et Favre, 2013)	84.29	02.27	15.71	30.64
No clarification	0.0	15.14	0.0	30.60

TABLE 2 – Les résultats des systèmes de fusion, du reranker, de l'oracle, de (Bechet et Favre, 2013) sont reportés en termes de Taux de Classification Correcte (TCC) et Taux d'Erreur Mots (TEM) sur le corpus *Parole*.

Le Tableau 3 donne le taux de classification correcte obtenu par le reranker sur le corpus *AMT* et *Parole*. Les résultats donnés sur le corpus *AMT* sont réalisés en utilisant une méthode d'évaluation croisée *2-fold* (la première partie du corpus est utilisée pour entraîner les modèles et l'autre partie pour les tester). Les résultats sur le corpus *Parole* sont donnés en utilisant un modèle entraîné sur toutes les données *AMT*. Comme les poids du MLP sont initialisés de manière aléatoire, nous

	AMT	Parole	
	Texte	Ref.	RAP
TCC	90.65 ±1.68	90.83 ±1.43	58.74 ±2.00

TABLE 3 – Taux de classification correcte sur les corpus *AMT* et *Parole*.

lançons 100 fois l'expérience et rapportons la moyenne et l'écart-type du taux de classification correcte.

Nous observons que sur le texte de référence le reranker obtient un taux de classification de 90.65 % et 90.83 % respectivement pour les transcriptions de référence des corpus *AMT* et *Parole*, tandis que sur la transcription automatique le taux de classification chute à 58 %. Cette différence peut être expliquée par le fait que le reranker est entraîné sur du texte de référence (*AMT*) et il est donc moins robuste aux erreurs commises par le système de RAP.

Le Tableau 4 donne le nombre de fois où chaque système de fusion a été classifié à +1 sur le corpus *Parole* par le reranker (le neurone de la couche cachée doit être > 0), ainsi que le ratio de bonnes réponses parmi eux. Nous notons que le nombre de bonnes réponses est lié à l'oracle, ce qui explique pourquoi les résultats sont plus bas sur la sortie de la RAP. Le reranker est entraîné sur du texte propre *AMT*, les systèmes ont tendance à être choisis moins souvent sur la sortie de la RAP en raison d'erreurs de mots qui réduisent le nombre de mots appariés entre la phrase originale et la phrase de clarification. Nous notons que même si plusieurs systèmes de fusion peuvent être classés à +1 pour une instance donnée, nous utilisons l'*argmax* comme sortie pour le reranker.

	Ref.		RAP	
Méthode	Sel.	TCC	Sel.	TCC
Replace (baseline)	27.14	94.74	22.86	50.00
Insert (baseline)	47.14	78.79	45.71	18.75
Align no-err-loop	35.71	100.00	22.86	43.75
Err-loop	74.29	94.23	51.43	33.33
Err-loop + affine-gap	94.29	87.88	80.00	26.79
Phonemes only	80.00	89.29	58.57	24.39
Phonetic + words	92.86	89.23	77.14	27.78
Phonetic + Wordnet	87.14	91.80	70.00	26.53
Phonetic + embedding	92.86	89.23	81.43	26.32

TABLE 4 – Les systèmes de fusion classés avec +1 par le reranker sur le corpus *Parole* en pourcentage d'instance et taux de classification correcte.

6 Conclusion

Dans ce papier, nous proposons un système pour résoudre le problème de fusion entre une phrase originale contenant une erreur et une phrase de clarification. Nous proposons plusieurs alignements qui sont des variantes de Levenshtein et un reranker pour sélectionner la meilleure hypothèse. Le système proposé permet d'améliorer le TEM d'environ 30 % et le reranker aide à éliminer complètement les erreurs dans 26 % des instances, atteignant ainsi les scores de l'oracle.

Dans les futurs travaux, nous allons étudier l'utilisation de l'alignement non-monotone qui sont des approches utilisées en traduction automatique, et faire usage de treillis de mots dans l'opération de fusion.

Références

ALTSCHUL, S. F. et ERICKSON, B. W. (1986). Optimal sequence alignment using affine gap costs. *Bulletin of mathematical biology*, 48(5):603–616.

BARZILAY, R. et MCKEOWN, K. (2005). Sentence fusion for multidocument news summarization. *Computational Linguistics*, 31(3):297–328.

BECHET, F. et FAVRE, B. (2013). Asr Error Segment Localization for Spoken Recovery Strategy. *In ICASSP*.

BOHUS, D. et RUDNICKY, A. I. (2009). The ravenclaw dialog management framework : Architecture and systems. *Computer Speech & Language*, 23(3):332–361.

DUFOUR, R., DAMNATI, G. et CHARLET, D. (2012). Automatic error region detection and characterization in lvcsr transcriptions of tv news shows. *In ICASSP*, pages 4445–4448. IEEE.

ELSNER, M. et SANTHANAM, D. (2011). Learning to fuse disparate sentences. *In Monolingual Text-To-Text Generation*, pages 54–63. Association for Computational Linguistics.

FILIPPOVA, K. et STRUBE, M. (2008). Sentence fusion via dependency graph compression. *In EMNLP*, pages 177–185. Association for Computational Linguistics.

GEROSA, M. et FEDERICO, M. (2009). Coping with out-of-vocabulary words : open versus huge vocabulary asr. *In ICASSP*, pages 4313–4316. IEEE.

HOSTE, L., DUMAS, B. et SIGNER, B. (2012). Speeg : a multimodal speech-and gesture-based text input solution. *In International Working Conference on Advanced Visual Interfaces*, pages 156–163. ACM.

HUGGINS-DAINES, D. et RUDNICKY, A. (2008). Interactive asr error correction for touchscreen devices. *In HLT*, pages 17–19. Association for Computational Linguistics.

JONSON, R. (2006). Dialogue context-based re-ranking of asr hypotheses. *In SLT*, pages 174–177. IEEE.

JUNG, S., LEE, C. et LEE, G. G. (2008). Using utterance and semantic level confidence for interactive spoken dialog clarification. *JCSE*, 2(1):1–25.

KOMBRINK, S., HANNEMANN, M. et BURGET, L. (2012). Out-of-vocabulary word detection and beyond. *In Detection and Identification of Rare Audiovisual Cues*, pages 57–65. Springer.

LAURENT, A. et AL (2011). Computer-assisted transcription of speech based on confusion network reordering. *In ICASSP*, pages 4884–4887. IEEE.

LENZO, K. (2007). The cmu pronouncing dictionary.

LIU, X., GALES, M. J. et WOODLAND, P. C. (2012). Use of contexts in language model interpolation and adaptation. *Computer Speech & Language*.

LÓPEZ-CÓZAR, R., GRIOL, D. et QUESADA, J. F. (2010). New technique to enhance the performance of spoken dialogue systems by means of implicit recovery of asr errors. *In Spoken Dialogue Systems for Ambient Environments*, pages 96–109. Springer.

MAAS, A. L. et AL (2012). Recurrent neural networks for noise reduction in robust asr. *In Interspeech*.

MARIN, A. et AL (2012). Using syntactic and confusion network structure for out-of-vocabulary word detection. *In SLT*, pages 159–164. IEEE.

MIKOLOV, T. et AL (2013). Efficient estimation of word representations in vector space. *In ICLR*.

MISU, T., KAWAHARA, T. et KOMATANI, K. (2004). Confirmation strategy for document retrieval systems with spoken dialog interface. *In Interspeech*.

MITCHELL, J. et LAPATA, M. (2009). Language models based on semantic composition. *In EMNLP*, pages 430–439. Association for Computational Linguistics.

OGATA, J. et GOTO, M. (2005). Speech repair : quick error correction just by using selection operation for speech input interfaces. *In Interspeech*, pages 133–136. Citeseer.

PARADA, C. et AL (2011). Learning sub-word units for open vocabulary speech recognition. *In ACL*, pages 712–721.

PARADA, C., DREDZE, M., FILIMONOV, D. et JELINEK, F. (2010). Contextual information improves oov detection in speech. *In NAACL*.

PRASAD, R. et AL (2012). Active error detection and resolution for speech-to-speech translation. *IWSLT*.

SEIGEL, M. S., WOODLAND, P. C. et AL (2011). Combining information sources for confidence estimation with crf models. *In Interspeech*, pages 905–908.

SHIN, J. et AL (2002). Analysis of user behavior under error conditions in spoken dialogs. *In ICSLP*, volume 2.

STOYANCHEV, S., LIU, A. et HIRSCHBERG, J. (2012). Clarification questions with feedback. *In Feedback Behaviors in Dialog*.

WENINGER, F. et AL (2012). Non-negative matrix factorization for highly noise-robust asr : To enhance or to recognize ? *In ICASSP*, pages 4681–4684. IEEE.

YU, D. et DENG, L. (2010). Semantic confidence calibration for spoken dialog applications. *In ICASSP*, pages 4450–4453. IEEE.