Analyse des erreurs d'une stratégie de sondage automatique d'opinions

Nathalie Camelin[1]*, Frederic Bechet*[1]*, Renato De Mori*[1] *, Geraldine Damnati*[2]*, **

[1] LIA - Université d'Avignon, BP1228 84911 Avignon cedex 09 France
[2] France Télécom R&D - TECH/SSTP/RVA 2 av. Pierre Marzin 22307 Lannion Cedex 07, France
{nathalie.camelin,frederic.bechet,renato.demori}@univ-avignon.fr
geraldine.damnati@orange-ftgroup.com

ABSTRACT

In this paper, we present an automatic opinion detection system which extracts distributions of opinions from real users of a telephone service. It enables to select from large corpora spoken messages that are likely to be reliably processed by the Automatic Speech Recognition system and the Automatic Opinion Classification one. For this reason, it is important to verify the representativness of the subcorpus extracted by the rejection strategy. Several measures, based on the Kullback-Leibler divergence, are proposed in order to evaluate the validness of our opinion extraction strategy analysing serveral types of errors it implies.

1. INTRODUCTION

Malgré le progrès réalisé par les systèmes de Reconnaissance Automatique de la Parole (RAP), de fort taux d'erreur mot (Word Error Rate - *WER*) sont obtenus sur certains messages *difficiles*, tels que ceux que l'on peut observer dans les conversations humain-humain collectés dans les centres d'appels ou encore les messages laissés sur des répondeurs téléphoniques comme présentés dans cette étude. Ceci s'explique notamment par les mauvaises conditions d'enregistrement et la parole spontanée constatés sur les messages collectés en milieu réel, comme remarqué dans la récente campagne *NIST Rich Transcription Meeting* mais également sur les corpus *MALACH*. Les enregistrements effectués par des centres d'appels ainsi que les corpus de sondages téléphoniques contiennent une grande variété de locuteurs, une mauvaise qualité audio due au téléphone portable ou/et au bruit ambiant, de la parole non contrainte, des messages de longueur variable et de nombreuses disfluences (hésitations, répétitions, reprises, ...). Par conséquent, l'Extraction d'Information (IE) est une tâche très difficile sur ce type de corpus. Cependant, le potentiel applicatif d'une telle tâche est important, *e.g* : extraction d'informatique décisionnelle à partir des messages de centres d'appels ou extraction d'opinions à partir de sondages téléphoniques.

L'extraction d'opinions a notamment donné lieu à de nombreuses publications [7, 6, 4] qui focalisaient sur des aspects différents de cette tâche : différenciation entre assertions subjectives ou objectives, résumé d'opinions, recherche de qui s'exprime à propos de quel sujet. En ce qui concerne la détection de thèmes, la redondance des idées ou d'occurrences de différents mots directement liés au thème permet de limiter l'impact des erreurs de reconnaissance. En revanche, les performances d'une détection plus fine sont grandement affectées par un fort WER. Dans ce cas, il est essentiel de prendre en compte uniquement les messages fiables selon un ensemble de mesures de confiance et de rejeter ceux équivoques. Il est important de noter que le rejet de ces derniers implique que le système extrait un échantillon du corpus.

Les performances des systèmes IE sont souvent évaluées par les mesures de précision et de rappel. Dans notre cas, le rappel n'est pas une mesure très pertinente car nous admettons qu'il est possible qu'une portion significative du corpus soit rejetée à cause des erreurs de reconnaissance. Dès lors, notre stratégie sera évaluée comme suit : la mesure de la précision sera appliquée afin d'évaluer le processus d'IE et une mesure de divergence sera utilisée pour évaluer la représentativité de l'échantillon sélectionné.

La section 2 décrit le cadre applicatif à partir duquel est testée la méthode d'analyse de sondage que nous proposons. La section 3 présente le système d'extraction d'opinion selon deux étapes : extraction des messages fiables puis classification des opinions. La section 4 présente les résultats obtenus et analyse le comportement du système.

2. CADRE APPLICATIF ET ANALYSE DE SONDAGE

2.1. Corpus de sondage téléphonique

Un premier recueil de sondage téléphonique, décrit en détail dans [2] et noté ici *CORPUS1*, a été collecté par France Télécom. Ce premier corpus a permis de définir une stratégie d'extraction d'opinions présentée dans [1].

Une seconde campagne de sondage d'opinions a été entreprise par France Télécom, le corpus correspondant est noté *CORPUS2*. L'appel à répondre au sondage est identique à celui de la précédente campagne, un court S.M.S. invite à s'exprimer sur la satisfaction vis à vis du service client récemment contacté. En revanche, le questionnaire est différent. Dans un premier temps, une série de quatre questions fermées permet de recueillir l'opinion globale puis l'opinion sur des dimensions particulières. Une question ouverte est ensuite proposée : "*Je vous remercie de m'avoir rappelé et d'avoir participé à l'étude. Si vous voulez me faire part d'un commentaire, laissez-moi un message ! C'est à vous !*".

Pendant un mois, environ 1 300 utilisateurs ont participé au sondage avec 700 d'entre eux ayant répondu à la question ouverte. Les 700 messages ainsi obtenus ont été transcrits manuellement et annotés selon les dimensions suivantes : la qualité de l'accueil (notée *accueil*), la rapidité d'accès au service (notée *attente*) et enfin l'efficacité du service (notée *efficacité*). Cette dernière dimension est la plus représentée, elle concerne à la fois l'évaluation des réponses aux attentes des utilisateurs (est ce que le problème a été réglé ?) mais aussi la qualité des informations données. Chaque dimension est associée à une polarité : *posi-*

*Travaux réalisés en collaboration avec France Télécom's R&D - contrat 021B178

tive ou *négative*. Nous avons donc un total de 6 étiquettes pour caractériser les expressions subjectives du corpus.

Dans la transcription manuelle, au sein de chaque message, ces expressions sont indiquées par des balises. Nous disposons ainsi d'un corpus de segments, chacun porteur d'une opinion particulière. Le but du traitement automatique est de retrouver ces segments et de les étiqueter avec l'une des 6 étiquettes. Voici un exemple de message avec les balises manuelles :

oui c'est monsieur NOMS PRENOMS j'avais appelé donc le service client ouais ***<seg etiq.=accueil,+>*** *j'ai été très bien accueilli* ***</seg>*** *des* ***<seg etiq.=efficacité,+>*** *bons renseignements* ***</seg>*** *sauf que* ***<seg etiq.=efficacité,->*** *ça ne fonctionne toujours pas* ***</seg>*** *donc je sais pas si j'ai fait une mauvaise manipulation ou y a un problème enfin voilà sinon* ***<seg etiq.=efficacité,+ label=accueil,+>*** *l'accueil était et les conseils très judicieux* ***</seg>*** *même si* ***<seg etiq.=efficacité,->*** *le résultat n'est pas n'est pas là* ***</seg>*** *merci au revoir*

La précision est calculée sur ces étiquettes. Notons que l'expression subjective *j'ai été très bien accueilli* est appelée *support* (sup) de l'étiquette `accueil+` .

CORPUS2 a été découpé en deux sous-corpus de taille équivalente, notés *APP* et *TEST*.

2.2. Analyse de sondage

L'analyse de sondage d'opinions est définie comme suit :

Soit C un corpus de n messages audio $m_1, m_2, \ldots, m_n$ exprimant une opinion à propos d'un service. Soit $C' \in C$ un sous-corpus de n' messages sélectionnés par un système automatique. On définit la proportion de messages sélectionnés par le système par :

$$cover = \frac{n'}{n} \tag{1}$$

Une opinion exprimée dans un message sera classée selon une dimension x et une valeur v. La valeur v d'une dimension x dans un message m est définie par $\vartheta(m, x)$ ainsi :

$$\vartheta(m,x) = \begin{cases} \textit{sans opinion} & \text{if } \forall\ sup_i \in m : sup_i \neq x \\ \textit{satisfait} & \text{if } \forall\ sup_i \in m : sup_i = x+ \\ \textit{insatisfait} & \text{if } \forall\ sup_i \in m : sup_i = x- \\ \textit{mitigé} & \text{otherwise} \end{cases}$$

Dans ce qui suit, les annotations manuelles seront notées *ref* et les annotations automatiques générées par la stratégie seront notées *hyp*.

Le principal objectif de l'analyse d'opinions est le calcul des proportions de messages contenant une opinion $O(x, v)$. $O_{ref}(x, v)$ correspond à l'opinion de référence d'un message tandis que $O_{hyp}(x, v)$ correspond à l'opinion attribuée automatiquement par le système.

Les proportions $p_{ref}(x, v)$ de l'opinion de dimension x et valeur v obtenues à partir de l'annotation manuelle est définie ainsi :

$$p_{ref}(x, v) = \frac{|O_{ref}(x, v)|}{n} \tag{2}$$

avec $|O_{ref}(x, v)|$ correspondant au nombre de messages $m \in C$ tel que $\exists sup_i \in m$ qui vérifie $\vartheta(m, x) = v$ et (x, v) une annotation manuelle.

En revanche, si l'on considère les opinions émises par le système, les proportions $p_{hyp}(x, v)$ sont définies par :

$$p_{hyp}(x, v) = \frac{|O_{hyp}(x, v)|}{n'} \tag{3}$$

avec $|O_{hyp}(x, v)|$ correspondant au nombre de messages $m \in C'$ tel que $\exists sup_i \in m$ qui vérifie $\vartheta(m, x) = v$ et (x, v) une annotation automatique.

Soit RP la distribution des proportions de référence $p_{ref}(x, v)$ et HP la distribution des proportions $p_{hyp}(x, v)$ obtenues de manière automatique. La stratégie peut être évaluée par la divergence entre la distribution HP générée par le système et la distribution RP de référence. L'évaluation se fait en fonction des divergences de Kullback-Leibler selon les différentes dimensions x :

$$D_{KL}\Big(RP(x)||HP(x)\Big) = \sum_v p_{ref}(x, v) \cdot \log \frac{p_{ref}(x, v)}{p_{hyp}(x, v)} \tag{4}$$

La divergence D_{KL} est alors obtenue par moyenne comme suit :

$$D_{KL}(RP||HP) = \sum_x \gamma_x D_{KL}\Big(RP(x)||HP(x)\Big) \tag{5}$$

où γ_x est un coefficient proportionnel à l'entropie de la dimension x dans C.

3. SYSTÈME DE DÉTECTION DES OPINIONS

L'extraction d'opinions à partir de parole spontanée émise par de vrais utilisateurs et collectée en milieu réel est une tâche très difficile. Typiquement, les expressions d'opinions sont mêlées dans le discours à l'énumération de faits (problème rencontré, détails personnels des situations, autre digressions). De plus, divers utilisateurs expriment leurs opinions de manière très différentes.

Afin de prendre en compte ces problèmes, nous proposons un système composé de deux étapes : transcription ciblée avec segmentation du message puis classification. L'articulation entre ces deux étapes consiste en un processus de rejet afin de ne garder que les segments jugés fiables dans un message. La figure 1 illustre le système.

3.1. Processus de segmentation

Il a été montré dans [2] que l'intégration du processus de segmentation directement dans le système RAP permettaient d'accroître les performances du système.

Les Modèles de Langage (ML) ont été appris à partir des messages de *CORPUS1* qui contient environ 4.4k mots différents pour un total de 130K occurrences. Presque la moitié de ces mots n'apparaissent qu'une seule fois dans le corpus et la restriction du lexique utilisé par le module de RAP aux mots apparaissant au moins deux fois limite ce lexique à 2.5k mots. Le taux de mots hors-vocabulaire résultant sur *CORPUS2* est de 2.9%.

Le modèle spécifique est un Modèle de Markov Caché à deux niveaux, un premier niveau sur les transitions entre type de segments (expression subjective ou segment factuel - considéré vide) et un deuxième niveau contenant un modèle de langage bigramme par dimension pour ceux contenant des expressions subjectives, ou une boucle de phonèmes non contrainte pour absorber les segments

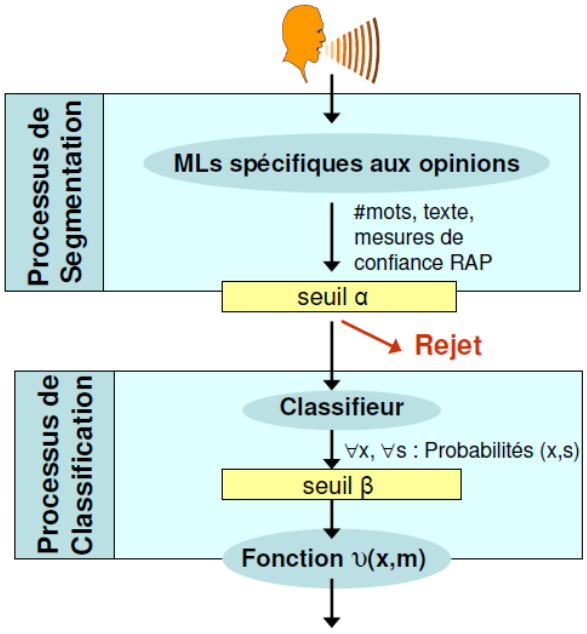

FIG. 1: Système d'extraction automatique d'opinions

vides. Un niveau supplémentaire a été ajouté dans le système RAP utilisé ici : il prend en compte la réponse obtenue à la question fermée suivante : "*Globalement, êtes-vous satisfait de la façon dont votre demande a été traitée ? Tout à fait, pas complètement, ou pas du tout ?*". Un ML spécifique a été appris pour chaque réponse possible. Ainsi, lors de la phase de reconnaissance, le ML utilisé lors du décodage de la réponse à la question ouverte dépend de la réponse obtenue à la question fermée précédente.

Le résultat de ce type de décodage pour un message est une liste d'hypothèses d'expressions subjectives, auxquelles est associé un ensemble de mesures de confiance (acoustiques et linguistiques). Ainsi, le principal avantage de ce modèle est de segmenter directement le flux audio en segments candidats pour l'extraction d'opinions.

Comme présenté dans la figure 1, ces mesures de confiance permettent de filtrer les segments peu fiables (seuil α). Par conséquent, si aucun segment d'un message n'est jugé fiable, alors le message est rejeté. Ceci correspond à échantillonner l'univers du sondage. Cet échantillonnage est particulier car il vise à éliminer les messages qui soit n'expriment aucune opinion soit l'exprime d'une manière complexe et difficilement reconnaissable automatiquement.

Soit $O'_{ref}(x,v)$ l'opinion de référence d'un message accepté par le système. Soit $p'_{ref}(x,v) = \frac{|O'_{ref}(x,v)|}{n'}$ la proportion des opinions de référence des messages retenus par le processus de segmentation. La proportion $p'_{ref}(x,v)$ diffère de la proportion réelle $p_{ref}(x,v)$ du sondage car elle est évaluée sur moins de messages.

L'erreur due à l'échantillonnage pour l'opinion $O(x,v)$ est donnée par :

$$e^{\text{échant.}}(O(x,v)) = p_{ref}(x,v) - p'_{ref}(x,v) \qquad (6)$$

Soit RP' la distribution des opinions de référence sur le sous-corpus traité par le système. La divergence $D_{KL}(RP||RP')$ due à l'échantillonnage est évaluée en remplaçant $p_{hyp}(x,v)$ par $p'_{ref}(x,v)$ dans l'équation (4) puis (5).

3.2. *Processus de classification*

Une fois la segmentation terminée, le modèle de classification, basé sur une méthode de boosting, est appliqué aux segments sélectionnés. Chaque segment est étiqueté avec une paire (x,s) où x est une dimension et s sa polarité (*positive* ou *négative*). Ce modèle est entraîné avec BoosTexter [5] sur l'ensemble des transcriptions manuelles des expressions subjectives de *CORPUS1* et *APP*.

Chaque segment est représenté par trois niveaux d'abstraction : étiquettes morpho-syntaxiques, lemmes et *seeds*. La construction à la fois manuelle et automatique du lexique de *seeds* (mot *a priori* polarisé et/ou relatif à notre application) ainsi que l'intérêt de son utilisation sont décrits dans [2]. Le score attribué par le classifieur à chaque paire (x,s) sur un segment est converti en probabilité par régression logistique [3]. Un seuil de rejet β est appliqué sur cette probabilité afin de filtrer les annotations : chaque paire (x,s) est attribuée à un segment si et seulement si sa probabilité est supérieure au seuil β.

Le processus se termine par l'application de la fonction $\vartheta(x,m)$ à l'ensemble des étiquettes attribuées aux segments retenus dans un message afin d'en obtenir l'opinion.

Afin de prendre en compte les erreurs d'interprétation, les opinions $O'_{ref}(x,v)$ et $O_{hyp}(x,v)$ sont considérées et l'erreur calculée ainsi :

$$e^{\text{interp.}}(O(x,v)) = p'_{ref}(x,v) - p_{hyp}(x,v) \qquad (7)$$

La divergence $D_{KL}(RP'||HP)$ due à l'interprétation est évaluée en remplaçant $p_{ref}(x,v)$ par $p'_{ref}(x,v)$ dans l'équation (4) puis (5) (γ_x reste inchangé).

L'erreur globale du système de détection automatique d'opinions pour une opinion $O(x,v)$ donnée est :

$$e(O(x,v)) = e^{\text{échant.}}(O(x,v)) + e^{\text{interp.}}(O(x,v)) = \qquad (8)$$

4. PARAMÉTRAGE DU SYSTÈME ET ANALYSE DES ERREURS

La stratégie développée sur *CORPUS1* et présentée dans [1] est ici appliquée sur *CORPUS2* avec une analyse détaillée des types d'erreurs en résultant.

Le corpus *APP* est utilisé afin d'entrainer le classifieur du processus de classification et de choisir le paramétrage du système, c'est à dire les valeurs de α et β. La précision et la divergence de Kullback-Liebler sont évaluées à la fois sur les corpus *APP* et *TEST*, et illustrées par la figure 2. Les différentes valeurs sont obtenues par variation des seuils α et β

On remarque que les deux corpus observent un comportement similaire et que les valeurs minimales de $D_{KL}(RP||HP)$ sont obtenues pour une précision de 70%[1]. La valeur minimale de $D_{KL}(RP||HP)$, 0.01 bit, est très bonne et bien inférieure à celle observée avec l'ancien système sur *CORPUS1* (0.1 bit). L'augmentation de la taille du corpus pour l'apprentissage est indéniablement un atout.

[1] À titre d'information, une précision de 70% correspond à une valeur état de l'art dans le cadre de la détection d'opinions sur du texte

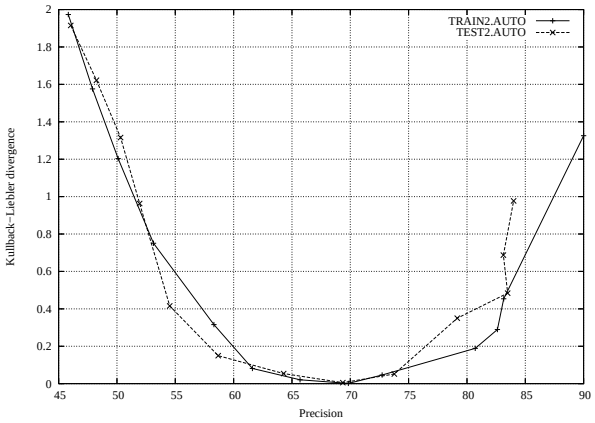

FIG. 2: Évaluation globale du système.

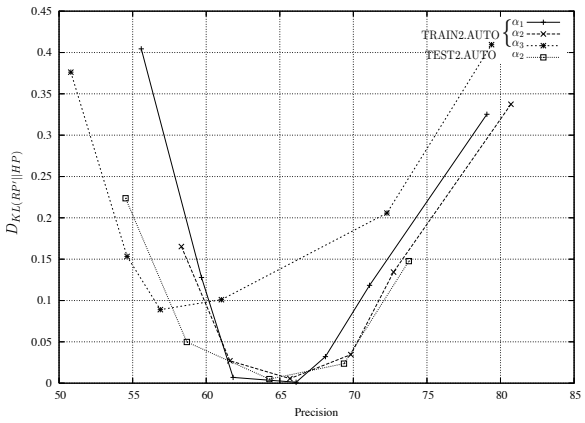

FIG. 4: Évaluation de l'erreur due à l'interprétation.

Afin de distinguer les erreurs dues à l'échantillonnage de celles dues à l'interprétation, les divergences $D_{KL}(RP||RP')$ et $D_{KL}(RP'||HP)$ sont calculées.

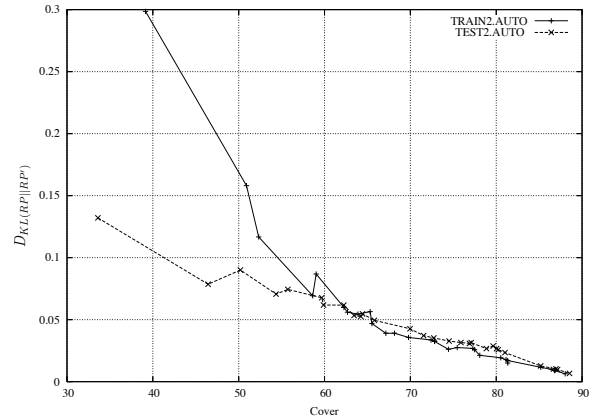

FIG. 3: Évaluation de l'erreur due à l'échantillonnage.

Dans la figure 3, $D_{KL}(RP||RP')$ est représentée en fonction de la couverture du système (*c.f.* : eq.1) et dont différentes valeurs sont obtenues en faisant varier α. On observe que plus la couverture est grande, plus la précision observée est faible sur le sous-corpus retenu. On remarque que la divergence est similaire pour *APP* et *TEST* lorsque la couverture est supérieure à 65%.

Dans la figure 4, $D_{KL}(RP'||HP)$ est représentée en fonction de la précision dont différentes valeurs sont obtenues par variation de β. La figure montre différentes courbes correspondant à des valeurs différentes de α. Soit α_1,α_2 et α_3 les valeurs correspondant respectivement à des couvertures de 65%, 75% et 85%. La figure 4 montre que l'erreur d'interprétation a un impact moindre pour les courbes correspondant à α_1 et α_2, ce qui détermine ainsi un intervalle de couverture correcte. En effet, si la couverture est trop grande, les erreurs d'interprétations causent une trop grande divergence $D_{KL}(RP'||HP)$ tandis qu'une faible valeur de divergence $D_{KL}(RP||RP')$ correspond à une faible couverture. Le choix de α_3, soit 85% de couverture, conduit à une divergence $D_{KL}(RP'||HP)$ de 0.0889 tandis que le choix α_1, soit 65% de couverture, conduit à une divergence $D_{KL}(RP'||HP)$ de 0.0011.

5. CONCLUSION

Les expériences présentées dans cet article valident la stratégie de sondage d'opinions présentée dans [1]. Elle permet de travailler sur des messages *difficiles*, observant un fort WER.

Nous avons introduit différentes mesures d'évaluations qui permettent de vérifier la représentativité et la précision des résultats obtenus. Ceux-ci montrent la robustesse de la stratégie proposée et confirme que malgré des transcriptions automatiques bruitées, il est possible d'effectuer une recherche d'information pertinente si certains messages ou portions de messages sont rejetés par une évaluation fiable.

Le paramétrage du système, qui correspond au choix de α et β, doit être effectué en fonction du but de l'application. D'un côté, si la précision est une priorité, alors α doit être élevé. D'un autre côté, si l'évaluation des proportions est une priorité, alors α et β doivent être choisis conjointement afin d'assurer une divergence minimale.

RÉFÉRENCES

[1] Nathalie Camelin, Frederic Bechet, Geraldine Damnati, and Renato De Mori. Speech mining in noisy audio message corpus. In *Interspeech*, Belgium, 2007.

[2] Nathalie Camelin, Geraldine Damnati, Frederic Bechet, and Renato De Mori. Détection automatique d'opinions dans des corpus de messages oraux. In *Journées d'Etude de la Parole*, 2006.

[3] Robert E.Schapire, Marie Rochery, Mazin Rahim, and Narendra Gupta. Boosting with prior knowledge for call classification. *IEEE*, 13(1) :174–181, march 2005.

[4] Soo-Min Kim and Eduard Hovy. Extracting opinions, opinion holders, and topics expressed in online news media text. In *Proceedings of the Workshop on Sentiment and Subjectivity in Text*, pages 1–8, Sydney, Australia, July 2006. Association for Computational Linguistics.

[5] Robert E. Schapire and Yoram Singer. BoosTexter : A boosting-based system for text categorization. *Machine Learning*, 39 :135–168, 2000.

[6] Veselin Stoyanov and Claire Cardie. Toward opinion summarization : Linking the sources. In *Proceedings of the Workshop on Sentiment and Subjectivity in Text*, pages 9–14, Sydney, Australia, July 2006. Association for Computational Linguistics.

[7] Jayce Wiebe and Ellen Riloff. Creating subjective and objective sentence classifiers from unannotated texts. In *CICLing*, 2005.