Some recent research works at LIUM based on the use of CMU Sphinx

Yannick Estève, Paul Deléglise, Sylvain Meignier, Holger Schwenk,
Loic Barrault, Fethi Bougares, Richard Dufour, Vincent Jousse, Antoine Laurent, Anthony Rousseau

LIUM, University of Le Mans, France
firstname.lastname@lium.univ-lemans.fr

Abstract

This paper presents an overview of the recent research works developed at LIUM using the CMU Sphinx tools. First, it describes the LIUM ASR system which reached very competitive results on French evaluation campaigns.

Then, different research works using the LIUM ASR system are described: detection and characterization of spontaneous speech in large audio database, language modeling to detect and correct errors in automatic transcripts or system combination in the framework of statistical machine translation.

Last, we discuss about the benefit of the availability of CMU Sphinx under a permissive open source license and, as we would like provide to the CMU Sphinx community some parts of our work, we discuss about the difficulties we encounter to participate in the development of CMU Sphinx.

1. Introduction

The LIUM automatic speech recognition system is based on the CMU Sphinx system. The tools distributed in the CMU Sphinx open-source package, although already reaching a high level of quality, can be supplemented or improved to integrate some state-of-art technologies. It is the solution LIUM has adopted to develop its own ASR system, by building on this base and gradually extending it to bring it to new performance levels, officially evaluated during the two ESTER evaluation campaigns on French broadcast news.

2. The ESTER evaluation campaigns

2.1. ESTER 1 and ESTER 2

The ESTER 1 evaluation campaign was organized within the framework of the TECHNOLANGUE project funded by the French government under the scientific supervision of the AFCP[1] with the DGA[2] and ELDA.

About 100 hours of transcribed data make up the corpus, recorded between 1998 and 2004 from six radio stations: *France Inter, France Info, RFI, RTM, France Culture* and *Radio Classique*. Shows last from 10 minutes up to 60 minutes. They consist mostly in prepared speech such as news reports, and a little conversational speech (such as interviews).

The corpus of articles from the French newspaper "Le Monde" from 1987 to 2003 can be used in addition to the transcription of the broadcast news to train the language model.

The ESTER 2 campaign falls under the continuity of ESTER 1. It was organized by the DGA and the AFCP during 2007 to 2008. The new campaign builds on the previous edition by reusing its corpus and extending it to cover new types of data. In particular, it includes more programs with foreign accents, as well as more programs spontaneous speech: in addition to French national broadcast news, ESTER 2 includes talk-shows and African programs (from the station *Radio Africa No 1*).

ESTER 2 supplements the ESTER 1 corpus with about 100 hours of transcribed broadcast news recorded from 1998 to 2004, with additional 6 hours for development and 6 hours for test from 2007-2008. Fast transcriptions of 40 hours of African broadcast news are also available. Textual resources are extended by articles from the newspaper "Le Monde" from 2004 to 2006.

[1] AFCP: Association Francophone de la Communication Parlée
[2] DGA: Délégation Générale de l'Armement

3. LIUM ASR system

The system described below was the best open source system during the ESTER 2 evaluation campaign (an older version of this system was also the best open source system during ESTER 1).

3.1. Diarization

The diarization system uses of the Sphinx toolkit to compute the feature vectors. This diarization system is composed of an acoustic BIC-based segmentation followed by a BIC-based hierarchical clustering. Each cluster represents a speaker and is modeled with a full covariance Gaussian. Viterbi decoding is used to adjust the segment boundaries using GMMs for each cluster.

Music and jingle regions are removed using Viterbi decoding with 8 GMMs, for music, jingle, silence, and speech (with wide/narrow band variants for the latter two, and clean/noised/musical background variants for wide-band speech).

Gender and bandwidth are then detected using 4 gender- and bandwidth-dependent GMMs.

Speech segments are then limited to 20 s by splitting overlong segments using a GMM-based silence detector.

This system, completed by a CLR-based clustering phase, obtained the best diarization error rate during the ESTER 2 campaign.

3.2. Speech recognition system

3.2.1. Features

The transcription decoding process is based on multi-pass decoding using 39 dimensional features (PLP with energy, delta, and double-delta). Two sets of features are computed for each show, corresponding to broadband and narrowband analysis.

3.2.2. Decoding

After speaker diarization, a first decoding pass permits to compute a CMLLR transformation for each speaker [1]. The decoding strategy involves 5 passes. The other passes are as follows:

- # 2 The best hypotheses generated by pass # 1 permit to compute a CMLLR transformation for each speaker. Decoding # 2, using SAT and Minimum Phone Error (MPE) acoustic models and CMLLR transformations, generates word-graphs.
- # 3 In the third pass, the word-graphs are used to drive a graph-decoding with full 3-phone context with a better acoustic precision, particularly in inter-word areas. This pass generates new word-graphs.
- # 4 The fourth pass consists in recomputing with a quadrigram language model the linguistic scores of the updated word-graphs of the third pass.
- # 5 The last pass generates a confusion network from the word-graphs and applies the consensus method to extract the final one-best hypothesis [2].

3.2.3. Acoustic models

Acoustic models for 35 phonemes and 5 kinds of fillers are trained using a set of 280 hours from ESTER 1 & 2. Models for pass # 1 are composed of 6500 tied states. Models for passes # 2 to # 5 are composed of 7500 and are trained in a MPE [3, 4] framework applied over the SAT-CMLLR models.

Both decoding passes employ tied-state word-position 3-phone acoustic models which are made gender- and bandwidth-dependent through MAP adaptation of means, covariances and weights.

The CMLLR technique for SAT in the second decoding pass generates a full 39x39 matrix for each speaker.

3.2.4. Vocaburary and Language models

Data used to build the linguistic models are of three kinds:

1. Manual transcriptions of broadcast news. They correspond to the transcription of the data used to train the acoustic models. We have also used manual transcriptions of conversations from the PFC corpus[5];
2. Newspaper articles: in addition to 19 years of "Le Monde" newspaper corpus, we also use articles from another French newspaper, "L'Humanité", from 1990 to 2007, and the French Giga Word Corpus;
3. Web resources drawn from "L'Internaute", "Libération", "Rue89", and "Afrik.com".

To build the vocabulary, we generate a unigram model as a linear interpolation of unigram models trained on the various training data sources listed above. The linear interpolation was optimized on the ESTER 2 development corpus in order to minimize the perplexity of the interpolated unigram model. Then, we extract the 122k most probable words from this language model.

Phonetic transcriptions for the vocabulary are taken from the BDLEX database, or generated by the rule-based, grapheme-to-phoneme tool LIA_PHON[6] for words not in the database.

Using this vocabulary, all the textual data of the training corpus is used to train trigram and quadrigram language models. To estimate and interpolate these models, the SRILM is employed using the modified Kneser-Ney discounting method. No cut-off is applied on unigrams, bigrams, trigrams and quadrigrams. The models are composed of 121k unigrams, 29M bigrams, 162M trigrams, and 376M quadrigrams.

4. LIUM system and CMU Sphinx tools

We have added large extensions to the SphinxTrain toolkit: MAP adaptation of means, but also weights and covariances of the models, as well as SAT based on CMLLR and MPE, are the most remarkable.

Passes # 1 and 2 use version 3.7 of the Sphinx decoder, slightly modified to employ the CMLLR transformation applied to the features. Pass # 4 is based on *sphinx3_astar* which we extended to handle quadrigram LMs. Passes # 3 and 5 are based on Sphinx version 4, which we heavily modified to develop the acoustic graph decoder and the confusion network generation.

Other parts, such as computation of PLP features and the diarization system, do not rely on Sphinx and are entirely in-house developments.

5. Experiments on the LIUM system

The experiments are carried out using the official test corpus of the ESTER 2 campaign. This corpus consists in 6 hours (26 shows) recorded between December 2007 and February 2008.

5.1. Global results

The WER over the test data for the LIUM ASR baseline system is 19.2 %.

Our system is based on a multi-pass architecture: table 1 shows the WER after each pass of the decoding process.

Table 1: Word error rates for each pass of LIUM'08

Pass	Word error rate
# 1 (general acoustic models, trigram)	27.1 %
# 2 (acoustic adaptation)	22.5 %
# 3 (word-graph acoustic rescoring)	20.4 %
# 4 (word-graph quadrigram rescoring)	19.4 %
# 5 (consensus)	**19.2** %
+ pronunciation variant probability	18.8 %
+ specialization of linguistic models	**18.1** %

We can observe that adaptation of the acoustic models allows a large gain in pass # 2, as does the better acoustic precision given by the full 3-phone search algorithm used to rescore a word-graph in pass # 3 (the acoustic models used theses two passes were trained using the MPE method). Rescoring this word-graph with a quadrigram model in pass # 4 allows to lower the WER by one extra point. The last pass does not have a significant impact on WER, but it allows the ASR system to provide confidence measures. Two improvements were integrated into our ASR system. The first one consists in assigning a score to each pronunciation variant in the dictionary. The score is computed by observing the frequency of the variant in the training corpus. Table 1 shows that this allows a gain of 0.4 point in terms of WER.

The last improvement is based on the presence of two kinds of francophone radio stations in the ESTER 2 campaign: French and African ones. We have decided to build two sets of linguistic knowledgde (lexicon and n-gram models), specific to each of these two kinds of stations. The MPE method to train acoustic

models was also adapted for the African radio stations. Table 1 shows that this allows to obtain the best word error rate of all our experiments: 18.1%.

5.2. Confidence measures

In order to provide additional information for applications which could use it, the LIUM system uses the *a posteriori* probabilities computed during the generation of the confusion networks to provide confidence measures [7].

However, as seen in table 2 which presents an evaluation of these confidence measures in terms of normalized cross entropy (NCE), with no specific treatment these *a posteriori* probabilities are not very good predictors of the word error rate.

So, a mapping method is applied which consists in splitting the *a posteriori* probabilities into 15 classes of values: each confidence measure is linearly transformed using the coefficient associated with its class. These coefficients have been optimized on the ESTER 2 development corpus to maximize NCE. Such mapping approach was presented in [8]. Table 2 shows that this method makes the confidence measures provided by the LIUM ASR system very competitive, with a NCE of 0.329 on the ESTER 2 test corpus.

Table 2: Contribution of the mapping method applied to the confidence measures of our ASR system

Confidence measures	NCE
without mapping	0.064
with mapping	**0.329**

6. Acoustic-based phonetic transcription method for proper nouns

One of our recent research work focuses on an approach to enhancing automatic phonetic transcription of proper nouns.

Proper nouns constitute a special case when it comes to phonetic transcription (at least in French, which was the language used for this study). Indeed, there is much less predictability in how proper nouns may be pronounced than for regular words. This is partly due to the fact that, in French, pronunciation rules are much less normalized for proper nouns than for other categories of words: a given sequence of letters is not guaranted to be pronounced the same way in two different proper nouns.

Common approaches to the problem of automatic grapheme to phoneme (G2P) conversion were proposed in the literature, the most popular are: the dictionary look-up strategy [9], the rule-based approach [10], and the knowledge-based approach [11].

In order to enrich the set of phonetic transcriptions of proper nouns with some less predictable variants, we used an Acoustic Phonetic Decoding (APD) system on speech segments that correspond to utterances of proper nouns.

In the manual transcription utterances, start and end times of individual words were not available. Therefore, the boundaries of each word of the transcription had to be determined by aligning the words with the signal, using a speech recognition system. In order to do the first forced alignement, we used three different grapheme to phoneme conversion method :

- A method we proposed in [12], based on the use of a statistical machine translation (SMT) system,
- A data-driven conversion system proposed in [11], based on the use of joint-sequence models (JSM),
- A rule-based G2P method, LIA_PHON ([6]) that relies on the spelling of words to generate the possible corresponding chains of phones.

When start and end times of segments that contains proper nouns are determined, they are then fed to the APD system to obtain their phonetic transcription. Thus, proper nouns which are present several times in the corpus potentially get associated with several phonetic transcriptions each. The filtering is used to remove phonetic variants of proper nouns that are the most likely to generate confusion with other words. The hardest step consists on detecting those phonetic variants. We propose to decode our training corpus using the proper noun phonetic dictionary that we want to filter, completed by a separate phonetic dictionary for non proper noun words. Every phonetic transcription that is never used to decode the corresponding proper noun in the right place gets removed from the dictionary, since it either caused an error or was not used at all. The process then gets repeated: the corpus is decoded again using the modified dictionary, which then gets filtered according to the results of this decoding. The whole decoding and filtering process is repeated until no more phonetic transcriptions get removed from the dictionary. When filtering process is over, the filtered dictionary is used instead of the G2P dictionary used at the beginning, for the forced alignment step. The full process (alignment/APD/filtering) is repeated until having two times the same filtered dictionary.

The metrics used are based on the Word Error Rate (WER) and on the Proper Noun Error Rate (PNER). The PNER is computed the same way as the WER but it is computed only for proper nouns and not for every word: $PNER = (I + S + E)/(N)$ where I is the number of wrong insertions of proper nouns, S the number of substitutions of proper nouns with other words, E the number of elisions of proper nouns, and N the total number of proper nouns.

On the ESTER 1 test corpus, the best results were obtained by using SMT to initialize the process. By using the ASR system developed for the ESTER 1 campaign, the WER decreased from 24.7% to 23.9% on segments that contains proper nouns. The PNER decreased from 26.2% to 20.5%.

7. Improving French ASR by targeting specific errors

Another of our recent research works focuses on the correction of specific errors. Some errors, which do not prevent understanding, are often neglected because they are not critical for the correct operation of such applications. For example, in French, errors of agreement in number or gender. For some other applications, such as subtitling for hearing-impaired people or assisted transcription [13], these errors are more important: in the former case, repetition of errors, even if they do not modify the meaning of a sentence, is very exhausting for the final user; in the latter case, where the goal is to produce an entirely correct transcription, these errors reduce the gain of productivity provided by the use of ASR system. Thus, the final user might limit his use of such transcription systems, feeling that ASR systems are not reliable enough because some errors would be easily corrected by a human user. The correspondence of gender, number (and/or person) is one of the most difficult aspects of the French language. French is an inflected language, containing a lot of homophonous inflected forms.

In this work, we wanted to repair some errors by post-processing the ASR output obtained with the Sphinx decoder. We proposed an approach [14] consisting in building a specific correction solution for each specific error. Indeed, some complex grammatical rules can not be modeled with a n-gram language model. Method must correct homophonous errors and must tend to be generic and reusable: no specific domain and no specific ASR system, and use of large vocabulary ASR. We particularly focused on the errors caused by the homophonous inflected forms of past participles, as well as the errors concerning the words *'vingt/vingts'* (twenty) and *'cent/cents'* (hundred). These errors are some of the most frequent errors produced by our ASR system, according to the analysis of confusion pairs.

To repair errors, we sought to use formal rules whenever possible. But this approach could not be the only one. In particular, formal rules are not very robust to errors existing in the lexical context of a targeted word. Thus, when it was possible to establish a formal rule, we did. If not, we tried to use a statistical method based on the use of a statistical classifier in order to correct a hypothesis word being a past participle. The statistical method, presented in [14], used various knowledge bases: lexical information, acoustic information, part-of-speech (POS), syntactic chunk categories, or other information levels. Moreover, acoustic information, given by the ASR system, is used to filter some potential corrections: a correction is validated only if one of its pronunciation variant matches with the pronunciation variant of the targeted word.

Method using formal rules, presented in [14], allowed to reduce the error rate for homophonous errors on words "cent" and "vingt" by 86.4% on our test corpus. Stochastic method which must repair some errors due to the homophonous inflected forms of past participles, allowed to reduce the error rate for this kind of errors by 11%.

8. Spontaneous speech characterization and detection in large audio database

We were also interested to the detection of spontaneous speech in a large audio database. Spontaneous speech, in opposition to prepared speech, occurs in Broadcast News (BN) data under several forms: interviews, debates, dialogues, etc. The main evidences characterizing spontaneous speech are disfluencies (filled pause, repetition, repair and false start), ungrammaticality and a language register different from the one that can be found in written texts. Depending on the speaker, the emotional state and the context, the language used can be very different. Processing spontaneous speech is one of the many challenges that ASR systems have to deal with. Indeed, automatically transcribing spontaneous speech is a more difficult task than automatically transcribing prepared speech (WER is higher).

In [15], we proposed a set of features for characterizing *spontaneous speech*. The relevance of these features was estimated on an 11 hours corpus (French Broadcast News) manually labelled by two human judges according to a level of spontaneity in a scale from 1 (clean, prepared speech) to 10 (highly disfluent speech, almost not understandable). The corpus was cut into segments thanks to automatic segmentation and diarization process, and transcripted by Sphinx decoder, before being annotated. Later, segments were labeled into three classes (*prepared*, *low spontaneity* and *high spontaneity*) depending of its level of spontaneity. In this work, we particularly focused on the detection of the *high spontaneity* class of speech.

In parallel to the subjective annotation of the corpus, we introduced the features used to describe speech segments. We chose speech segments that are relevant to characterize the spontaneity of those, and on which an automatic classification process can be trained on our annotated corpus. Three sets of features were used [15, 16]: acoustic features related to prosody, linguistic features related to the lexical and syntactic content of the segments, and confidence measures made by ASR system. The features were evaluated on our labeled corpus with a classification task: labeling speech segments according to the three classes of spontaneity.

Intuitively, we can feel that it should be rare to observe a *high spontaneous* speech segment surrounded by two prepared speech segments. Our previous approach, presented in [15], takes only into consideration the descriptors which are extracted from within the targeted segment, without taking into consideration information about surrounding segments. In order to improve our approach, we proposed in [16] to take into account the nature of the contiguous neighboring speech segments. It implies that the categorization of each speech segment from an audio file has an impact on the categorization of the other segments: the decision process becomes a global process. We chose to use a statistical classical approach by using a maximum likelihood method. With all these improvements, our method allows to achieve a 69.3% precision for *high spontaneous* speech detection with a 74.6% recall measure.

9. Using CMU Sphinx tools for Statistical Machine Translation

The LIUM laboratory is working on speech processing and machine translation. The speech team uses Sphinx library for several years. Since last year, the machine translation team has developed a system combination based on the decoding of lattices made of several confusion networks provided by different statistical machine translation (SMT) system. In order to decode these lattices, a token pass decoder has been developed. This decoder uses the Sphinx 4 library which is in Java. The following sections describe the approach for system combination, the alignment of hypotheses and the token pass decoder.

10. SMT System combination

The system combination approach is based on confusion network decoding as described in [17, 18] and shown in Figure 1. The protocol can be decomposed into three steps :

1. 1-best hypotheses from all M systems are aligned and confusion networks are built.
2. All confusion networks are connected into a single lattice with empirically estimated *prior* probabilities on the first arcs.
3. The resulting lattice is decoded and the 1-best hypothesis and/or n-best list of hypotheses are generated.

10.1. Hypotheses alignment and confusion network generation

For each segment, the best hypotheses of $M-1$ systems are aligned against the last one used as backbone. A modified version of the TERp tool [19] is used to generate a confusion network. This is done by incrementally adding the hypotheses to the CN. The hypotheses are added to the backbone beginning with the nearest (in terms of TERp) and ending with the more

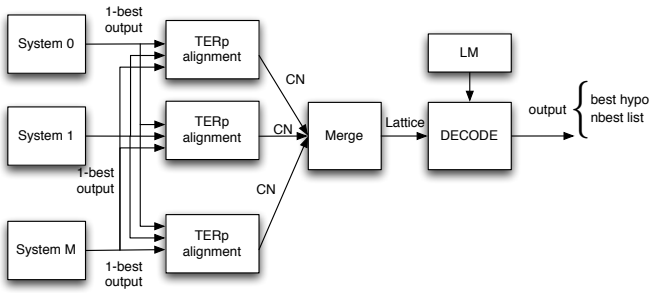

Figure 1: MT system combination.

distant one. This differs from the result of [18] where the nearest hypothesis is computed at each step. M confusion networks are generated in this way. Then all the confusion networks are connected into a single lattice by adding a first and last node. The probability of the first arcs (named *priors*) must reflect the capacity of such system to provide a well structured hypothesis.

10.2. Decoding

The decoder is based on the token pass decoding algorithm. The principle of this decoder is to propagate tokens over the lattice and accumulate various scores into a global score for each hypotheses.

The scores used to evaluate the hypotheses are the following :

- the system score : this replace the score of the translation model. Until now, the words given by all systems have the same probability which are equal to their priors, but any confidence measure can be used at this step.
- the language model (LM) probability.
- a fudge factor to balance the probabilities provided in the lattice with regard to those given by the language model.
- a *null-arc* penalty : this penalty avoids to always go through *null-arcs* encountered in the lattice.
- a length penalty : this score helps to generate well sized hypotheses.

The probabilities computed in the decoder can be expressed as follow :

$$log(P_W) = \sum_{n=0}^{Len(W)} [log(P_{ws}(n)) + \alpha P_{lm}(n)] + Len_{pen}(W) + Null_{pen}(W) \quad (1)$$

where $Len(W)$ is the length of the hypothesis, $P_{ws}(n)$ is the score of the n^{th} word in the lattice, $P_{lm}(n)$ is its LM probability, α is the fudge factor, $Len_{pen}(W)$ is the length penalty of the word sequence and $Null_{pen}(W)$ is the penalty associated with the number of null-arcs crossed to obtain the hypothesis.

At the beginning, one token is created at the first node of the lattice. Then this token spread over the consecutive nodes, accumulating the score on the arc it crosses, the language model probability of the word sequence generated so far and null or length penalty when applicable. The number of tokens can increase really quickly to cover the whole lattice, and, in order to keep it tractable, only the N_{max} best tokens are kept (the others are discarded), where N_{max} can be configured in the configuration file.

10.2.1. Technical details about the token pass decoder

This software is based on the Sphinx4 library and is highly configurable. The maximum number of tokens being considered during decoding, the fudge factor, the null-arc penalty and the length penalty can all be set within the xml configuration file. This is really useful for tuning.

This decoder uses a language model (LM) which is described in section 10.2.2.

10.2.2. Language Model

There are two ways of loading a LM with this software.

The first solution is to use the LargeTrigramModel class, but as its name tells us, only a 3-gram model can be loaded with this class.

The second and easiest way is to use a language model hosted on a lm-server. This kind of LM can be accessed via the LanguageModelOnServer class which is based on the generic LanguageModel class from the Sphinx4 library. This allow us to load a n-gram LM with *n* higher than 3, which is not possible with a standard LM class in Sphinx4 ... at this time.

In addition, the Dictionnary interface has been extended in order to be able to load a simple dictionary containing all the words known by the LM (no need to know the different pronunciations of each words in this case).

11. Experimental Evaluation of SMT System Combination

This MTSyscomb software was used for the IWSLT'09 evaluation campaign. Table 3 presents the results obtained with this approach. The SMT system is based on MOSES, the SPE system correspond to a rule-based system from SYSTRAN whose outputs have been corrected by a SMT system and the $Hierarchical$ is based on Joshua.

Systems	Arabic/English		Chinese/English	
	Dev7	Test09	Dev7	Test09
SMT	54.75	50.35	41.71	36.04
SPE	48.13	-	41.23	38.53
Hierarchical	54.00	49.06	39.78	31.89
SMT + SPE			42.55	40.14
+ tuning			43.06	39.46
SMT + Hier.	55.89	50.86		
+ tuning	57.01	51.74		

Table 3: Results of system combination on Dev7 (development) corpus and Test09, the official test corpus of IWSLT'09 evaluation campaign.

In these task, the system combination approach yielded +1.39 BLEU on Ar/En and +1.7 BLEU on Zh/En. One observation is that tuning parameters did not provided better results for Zh/En.

12. Conclusion

This paper presents some recent research works made at LIUM. We have started using CMU Sphinx tools in 2004 in order to develop an entire ASR system in French language. We have added some improvements and have made our system the best open source system participating to French evaluation campaigns.

This system is now at the center of the research works of the LIUM Speech Team. In the framework of speech processing, these works include grapheme-to-phoneme conversion, special strategy of correction to process frequent specific errors, detection and characterization of spontaneous speech in large audio database. We used also CMU Sphinx tools in our research work on statistical machine translation to combine SMT system. Other research works, not presented in this paper, are made by using CMU Sphinx, like speaker named identification which exploits the outputs of our speaker diarization system and the output of our ASR system.

We already share some our resources (French acoustic and language models for example) and we try to integrate our addons into the canonical source code of the CMU Sphinx project. This last thing is very hard because it needs a lot of development time, and because CMU Sphinx progresses and its source code frequently changes. In the future, we will try to integrate our work into the canonical source code as soon as possible in order to make easier this integration.

13. References

[1] M. J. F. Gales, "Maximum likelihood linear transfor- mations for hmm-based speech recognition," *Computer Speech and Language*, vol. 12, p. 7598, 1998.

[2] H. Mangu, E. Brill, and S. A., "Finding consensus in speech recognition: Word error minimization and other applications of confusion networks," *Computer Speech and Language*, vol. 14, no. 4, pp. 373–400, 2000.

[3] D. Povey and P. Woodland, "Minimum phone error and i-smoothing for improved discriminative training," in *ICASSP*, Florida, USA, 2002, pp. 105–108.

[4] D. Povey, "Discriminative training for large vocabulary speech recognition," Ph.D. Dissertation, Department of Engineering, University of Cambridge, United Kingdom, 2004.

[5] J. Durand, B. Laks, and C. Lyche, "La phonologie du français contemporain : usages, variétés et structure," *Romance Corpus Linguistics - Corpora and Spoken Language*, pp. 93–106, 2002.

[6] F. Béchet, "LIA_PHON un système complet de phonétisation de texte," in *Traitement Automatique Des Langues*. Hermès, 2001, vol. 42, pp. 47–68.

[7] H. Jiang, "Confidence measures for speech recognition: a survey," *Speech Communication Journal*, vol. 45, pp. 455–470, 2005.

[8] G. Evermann and P. Woodland, "Large vocabulary decoding and confidence estimation using word posterior probabilities," in *ICASSP*, Istanbul, Turkey, June 2000.

[9] M. De Calmes and G. Perennou, "BDLEX: a lexicon for spoken and written French," in *Proc. of LREC, International Conference on Language Resources and Evaluation*, 1998, pp. 1129–1136.

[10] F. Béchet, "LIA_PHON : un système complet de phonétisation de textes," in *TAL, Traitement Automatique des Langues*, 2001, pp. 47–67.

[11] M. Bisani and H. Ney, "Joint-sequence models for grapheme-to-phoneme conversion," *Speech Comm.*, vol. 50, no. 5, pp. 434–451, 2008.

[12] A. Laurent, P. Deléglise, and S. Meignier, "Grapheme to phoneme conversion using an smt system," in *In Proceedings of Interspeech-2009*, 2009.

[13] T. Bazillon, Y. Estève, and D. Luzzati, "Manual vs assisted transcription of prepared and spontaneous speech," in *LREC 2008*, Marrakech, Morocco, May 2008.

[14] R. Dufour and Y. Estève, "Correcting ASR outputs: specific solutions to specific errors in French," in *SLT 2008*, Goa, India, December 2008.

[15] R. Dufour, V. Jousse, Y. Estève, F. Béchet, and G. Linarès, "Spontaneous speech characterization and detection in large audio database," in *SPECOM 2009*, St Petersburg, Russia, June 2009.

[16] R. Dufour, Y. Estève, P. Deléglise, and F. Béchet, "Local and global models for spontaneous speech segment detection and characterization," in *ASRU 2009*, Merano, Italy, December 2009.

[17] W. Shen, B. Delaney, T. Anderson, and R. Slyh, "The MIT-LL/AFRL IWSLT-2008 MT system," in *IWSLT*, Hawaii, U.S.A, 2008, pp. 69–76.

[18] A.-V. Rosti, S. Matsoukas, and R. Schwartz, "Improved word-level system combination for machine translation," in *ACL*, 2007, pp. 312–319.

[19] M. Snover, B. Dorr, R. Schwartz, L. Micciulla, and J. Makhoul, "A study of translation edit rate with targeted human annotation," in *ACL*, 2006.