Real-Time Correction of Closed-Captions

P. Cardinal, G. Boulianne, M. Comeau, M. Boisvert
Centre de recherche Informatique de Montreal (CRIM)
Montreal, Canada
patrick.cardinal@crim.ca

Abstract

Live closed-captions for deaf and hard of hearing audiences are currently produced by stenographers, or by voice writers using speech recognition. Both techniques can produce captions with errors. We are currently developing a correction module that allows a user to intercept the real-time caption stream and correct it before it is broadcast. We report results of preliminary experiments on correction rate and actual user performance using a prototype correction module connected to the output of a speech recognition captioning system.

1 Introduction

CRIM's automatic speech recognition system has been applied to live closed-captioning of french-canadian television programs (Boulianne et al., 2006). The low error rate of our approach depends notably on the integration of the re-speak method (Imai et al., 2002) for a controlled acoustic environment, automatic speaker adaptation and dynamic updates of language models and vocabularies, and was deemed acceptable by several Canadian broadcasters (RDS,CPAC,GTVA and TQS) who have adopted it over the past few years for captioning sports, public affairs and newscasts.

However, for sensitive applications where error rates must practically be zero, or other situations where speech recognition error rates are too high, we are currently developing a real-time correction interface. In essence, this interface allows a user to correct the word stream from speech recognition before it arrives at the closed-caption encoder.

2 Background

Real-time correction must be done within difficult constraints : with typical captioning rates of 130 words per minute, and 5 to 10% word error rate, the user must correct between 6 and 13 errors per minute. In addition, the process should not introduce more than a few seconds of additional delay over the 3 seconds already needed by speech recognition.

In a previous work, (Wald et al., 2006) explored how different input modalities, such as mouse/keyboard combination, keyboard only or function keys to select words for editing, could reduce the amount of time required for correction. In (Bateman et al., 2000), the correction interface consisted in a scrolling window which can be edited by the user using a text editor style interface. They introduced the idea of a controllable delay during which the text can be edited.

Our approach combines characteristics of the two previous systems. We use a delay parameter, which can be modified online, for controlling the output rate. We also use the standard mouse/keyboard combination for selecting and editing words. However we added, for each word, a list of alternate words that can be selected by a simple mouse click; this simplifies the edition process and speeds up the correction time. However, manual word edition is still available.

Another distinctive feature of our approach is the fixed word position. When a word appears on screen, it will remain in its position until it is sent

Proceedings of the ACL 2007 Demo and Poster Sessions, pages 113–116,
Prague, June 2007.

out. This allows the user to focus on the words and not be distracted by word-scrolling or any other word movement.

3 Correction Software

The correction software allows edition of the closed-captions by intercepting them while they are being sent to the encoder. Both assisted and manual corrections can be applied to the word stream.

Assisted correction reduces the number of operations by presenting a list of alternate words, so that a correction can be done with a simple mouse click. Manual correction requires editing the word to be changed and is more expensive in terms of delay. As a consequence, the number of these operations should be reduced to a strict minimum.

The user interface shown in figure 1 has been designed with this consideration in mind. The principal characteristic of the interface is that there is no scrolling. Words never move; instead the matrix is filled from left to right, top to bottom, with words coming from the speech recognition, in synchronisation with the audio. When the bottom right of the matrix is reached, filling in starts from the upper left corner again. Words appear in blue while they are editable, and in red once they have been sent to the caption encoder. Thus a blue "window", corresponding to the interval during which words can be edited, moves across the word matrix, while the words themselves remain fixed.

For assisted correction, the list of available alternatives is presented in a list box under each word. These lists are always present, instead of being presented only upon selection of a word. In this way the user has the opportunity of scanning the lists in advance whenever his time budget allows.

The selected word can also be deleted with a single click. Different shortcut corrections, as suggested in (Wald et al., 2006) can also be applied depending on the mouse button used to select the word: a left button click changes the gender (masculin or feminin) of the word while a right button click changes the plurality (singular or plural) of the word. These available choices are in principle excluded from the list box choices.

To apply a manual correction, the user simply clicks the word with the middle button to make it editable; modifications are done using the keyboard.

Two users can run two correction interfaces in parallel, on alternating sentences. This configuration avoids the accumulation of delays. This functionality may prove useful if the word rate is so high that it becomes too difficult to keep track of the word flow. In this mode, the second user can begin the correction of a new sentence even if the first has not yet completed the correction of his/her sentence. Only one out of two sentences is editable by each user. The synchronisation is on a sentence basis.

3.1 Alternate word lists

As described in the previous section, the gender/plurality forms of the word are implicitly included and accessible through a simple left/right mouse click. Other available forms explicitly appear in a list box. This approach has two major benefits. First, when a gender/plurality error is detected by the user, no delay is incurred from scanning the choices in the list box. Second, since the gender/plurality forms are not included in the list box, their place becomes available for additional alternate words.

The main problem is to establish word lists short enough to reduce scanning time, but long enough to contain the correct form. For a given word output by the speech recognition system, the alternate words should be those that are most likely to be confused by the recognizer.

We experimented with two pre-computed sources of alternate word lists:

1. A list of frequently confused words was computed from all the available closed-captions of our speech recognition system for which corresponding exact transcriptions exist. The training and development sets were made up of 1.37M words and 0.17M words, respectively.

2. A phoneme based confusion matrix was used for scoring the alignment of each word of the vocabulary with every other word of the same vocabulary. The alignment program was an implementation of the standard dynamic programming technique for string alignment (Cormen et al., 2001).

Each of these techniques yields a list of alternate words with probabilities based on substitution like-

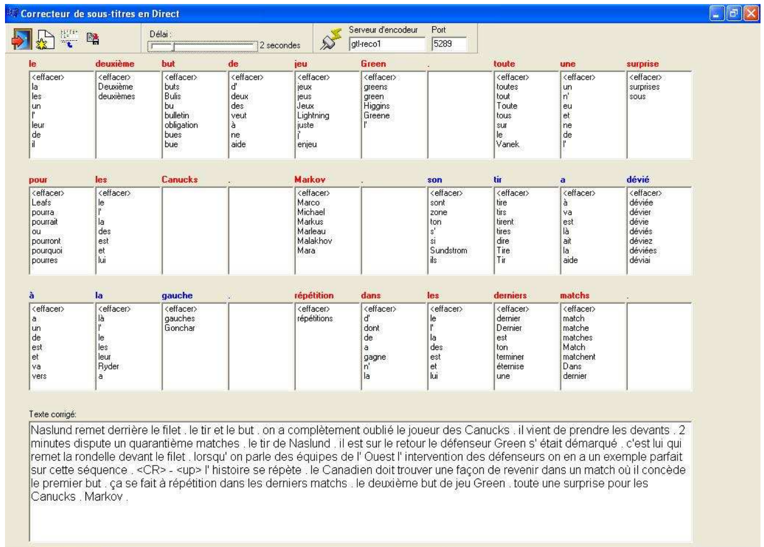

Figure 1: Real-time corrector software.

Source of alternates	coverage (%)
Word confusion matrix	52%
Phoneme confusion matrix	37%
Combined	60%

Table 1: *Coverage of substitutions (dev set).*

lihoods. Table 1 shows how many times substitutions in the development set could be corrected with a word in the list, for each list and their combination.

To combine both lists, we take this coverage into consideration and the fact that 48% of the words were common to both lists. On this basis, we have constructed an alternate list of 10 words comprised of the most likely 7 words of case 1; the remaining 3 words are the most probable substitutions from the remaining words of both lists.

3.2 Real-time List Update

The previous technique can only handle simple substitutions: a word that is replaced by another one. Another frequent error in speech recognition is the replacement of a single word by several smaller ones. In this case, the sequence of errors contains one substitution and one or more insertions. From the interface point of view, the user must delete some words before editing the last word in the sequence.

To assist the user in this case, we have implemented the following procedure. When a word is deleted by the user, the phonemes of this word are concatenated with those of the following words. The resulting sequence of phonemes is used to search the dictionary for the most likely words according to the pronunciation. These words are dynamically added to the list appearing under the preceding word. The search technique used is the same alignment procedure implemented for computing the confusion matrix based on phoneme confusion.

4 Results

In this section we present the results of two preliminary experiments. In the first one, we simulated a perfect correction, as if the user had an infinite amount of time, to determine the best possible results that can be expected from the alternate word lists. In the second experiment, we submitted a prototype to users and collected performance measurements.

4.1 Simulation Results

The simulation is applied to a test set consisting of a 30 minute hockey game description for which closed-captions and exact transcripts are available. We aligned the produced closed-captions with their corrected transcripts and replaced any incorrect word by its correct counterpart if it appeared in the alternate list. In addition, all insertion errors were deleted. Table 2 shows the word error rate (WER)

Source of alternates	WER
Original closed-captions	5.8%
Phoneme confusion matrix	4.4%
Word confusion matrix	3.1%
Combined	2.9%

Table 2: *Error rate for perfect correction.*

	Delay	
	2 seconds	15 seconds
test duration	30 minutes	8 minutes
# of words	4631	1303
# of editions	21	28
WER before	6.8%	6.2%
WER after	6.1%	2.5%
Gain (relative %)	8.1%	58.7%

Table 3: *Error rate after user correction.*

obtained for different alternate word lists.

The word confusion matrix captures most of the substitutions. This behavior was expected since the matrix has been trained explicitely for that purpose. The performance should increase in the future as the amount of training data grows. In comparison, the contribution of words from the phoneme confusion matrix is clearly limited.

The corrected word was the first in the list 35% of the time, while it was in the first three 59% of the time. We also simulated the effect of collapsing words in insertion-substitution sequences to allow corrections of insertions : the increase in performance was less than 0.5%.

4.2 User Tests

Experiments were performed by 3 unacquainted users of the system on hockey game descriptions. In one case, we allowed a delay of 15 seconds; the second case allowed a 2 second delay to give a preliminary assessment of user behavior in the case of minimum-delay real-time closed-captioning. Table 3 shows the error rate before and after correction.

The results show that a significant WER decrease is achieved by correcting using a delay of 15 seconds. The reduction with a 2 second delay is minor; with appropriate training, however, we can expect the users to outperform these preliminary results.

5 Conclusion and Future Work

We are currently developing a user interface for correcting live closed-captions in real-time. The interface presents a list of alternatives for each automatically generated word. The theoretical results that assumes the user always chooses the correct suggested word shows the potential for large error reductions, with a minimum of interaction. When larger delays are allowed, manual edition of words for which there is no acceptable suggested alternative can yield further improvements.

We tested the application for real-time text correction produced in a real-world application. With users having no prior experience and with only a 15 second delay, the WER dropped from 6.1% to 2.5%.

In the future, users will be trained on the system and we expect an important improvement in both accuracy and required delay. We will also experiment the effect of running 2 corrections in parallel for more difficult tasks. Future work also includes the integration of an automatic correction tool for improving or highlighting the alternate word list.

References

A. Bateman, J. Hewitt, A. Ariyaeeinia, P. Sivakumaran, and A. Lambourne. 2000. *The Quest for The Last 5%: Interfaces for Correcting Real-Time Speech-Generated Subtitles* Proceedings of the 2000 Conference on Human Factors in Computing Systems (CHI 2000), April 1-6, The Hague, Netherlands.

T.H. Cormen, C.E. Leiserson, R.L. Rivest, and C. Stein 2001. *Introduction to Algorithms* second edition, MIT Press, Cambridge, MA.

G. Boulianne, J.-F. Beaumont, M. Boisvert, J. Brousseau, P. Cardinal, C. Chapdelaine, M.Comeau, P. Ouellet, and F. Osterrath. 2006. *Computer-assisted closed-captioning of live TV broadcasts in French* Proceedings of the 2006 Interspeech - ICSLP, September 17-21, Pittsburg, US.

T. Imai, A. Matsui, S. Homma, T. Kobayakawa, O. Kazuo, S. Sato, and A. Ando 2002. *Speech Recognition with a respeak method for subtiling live broadcast* Proceedings of the 2002 ICSLP, September 16-20, Orlando, US.

Wald, M. 2006 *Creating Accessible Educational Multimedia through Editing Automatic Speech Recognition Captioning in Real Time.* International Journal of Interactive Technology and Smart Education : Smarter Use of Technology in Education 3(2) pp. 131-142