Some Experiments in Evaluating ASR Systems Applied to Multimedia Retrieval

Julián Moreno[1], Marta Garrote[1], Paloma Martínez[1], and José L. Martínez-Fernández[2]

[1] Computer Science Department, Universidad Carlos III de Madrid, Avda. Universidad n 30, 28911, Leganés, Madrid, Spain
{jmschnei,mgarrote,pmf}@inf.uc3m.es
[2] DAEDALUS – Data, Decisions and Language S.A.
Avda. de la Albufera, 321
28031 Madrid, Spain
jmartinez@daedalus.es

Abstract. This paper describes some tests performed on different types of voice/audio input applying three commercial speech recognition tools. Three multimedia retrieval scenarios are considered: a question answering system, an automatic transcription of audio from video files and a real-time captioning sys-tem used in the classroom for deaf students. A software tool, RET (Recognition Evaluation Tool), has been developed to test the output of commercial ASR systems.

Keywords: Automatic Speech Recognition (ASR), Evaluation Measurements, audio transcription, voice interaction.

1 Introduction

There is a growing demand for services that improve access to information available on the web. The current trend in developing Information Retrieval (IR) systems focuses on dealing with any format (audio, video, images). These different formats not only appear in the objects collection to be searched but also in the user's queries. The existence of a huge amount of multimedia resources in the web requires powerful tools that allow the users to find them. These solutions exploit metadata related to the video, image or audio, using text based retrieval techniques. Although these techniques are advanced enough and show accurate results, other data formats, as video or audio, still need research. Techniques allowing a content based approach for these formats are still under development. The main goal is to retrieve a video, audio or image without using metadata or any other text related to the content.

For image contents, there are research efforts to make content based analysis, for example, the ImageCLEF track at the Cross Language Evaluation Forum[1]. Some of the image processing techniques exploit colour, brightness and other features to classify images, but some of them try to recognize shapes appearing in the image.

[1] http://www.clef-campaign.org

Unfortunately, performance measures for this kind of analysis are still poor to allow some kind of widely used commercial application.

For audio contents, ASR techniques can be applied to produce textual transcriptions. In this way, conversion to text format is performed, in order to apply well known text retrieval techniques. This is the case of applications like Google Audio Indexer from Google Labs[2], which takes profit from audio transcription of videos using ASR technology, thus allowing to locate the point in a video or videos where the keyword written in the search box is mentioned. For the moment, this application only works for the English language and using videos of a specific domain (newscasts or politicians' talks). Nevertheless, there is a growing interest in the field of video and audio indexing. Google is not the only company developing products; other vendors in the market, such as Autonomy Virage[3], include tools to perform audio and video indexing.

In order to improve search and retrieval of audiovisual contents using speech recognition, it is necessary to evaluate the accuracy of ASR technology before using it for information retrieval applications.

The objective of this paper is twofold: firstly, evaluating the efficiency of speech recognition technologies in transcribing audio recordings from videos or audios to be indexed by an information retrieval system, such as a question answering system or a live subtitling application in a classroom; secondly, showing an evaluation tool, RET, developed to assist in testing the ASR technology in different application scenarios.

RET tool has been used in the evaluation of three ASR commercial products in (a) 160 short voice queries as input of a Question answering system in order to test a multimodal access (b) transcription of audio recordings from video resources with the aim of indexing them for further information extraction or information access and (c) live subtitling in an educational environment to help impaired students.

The paper is organized as follows: section 2 presents the related work; section 3 is devoted to explain the RET architecture and functionality, as well as the evaluation measures used; section 4 shows the experiments that have been performed in the three scenarios; an analysis of the results is shown in section 5; and finally, some enhancements are shown in section 6.

2 Related Work

The initial motivation which leads us to design and implement RET was the lack of a product covering all our needs. We were searching for a software tool that could provide us with measurements obtained from text comparison, to evaluate the efficiency of three speech recognition systems. There are several applications that have served as inspiration to solve our problem with the evaluation. One of these applications is DiffDoc [2], a texts comparison program which does not require a previous alignment: it does a direct comparison between files. The comparison process is similar to

[2] http://labs.google.com/gaudi
[3] http://www.virage.com/rich-media/technology/index.htm

RET's; both programs compare complex words and not simple characters, as most applications do (Winmerge [9], Altova Diffdog [1], Ultracompare [4]). An important advantage of DiffDoc [2] is the graphical interface, which shows the input files, allowing a visual comparison. The lack of numeric results after the comparison is the main disadvantage of this tool.

The *SCTK Scoring Toolkit* from National Institute of Standards and Technology of United States (NIST) [7] is a text comparison software specifically oriented to text-to-speech systems. The main differences between SCTK and RET are:

- RET interface displays original text (OT) and ASR output text (AOT), where the words that do not match are highlighted using colors. In this way, it is possible to carry out a qualitative study by linguists, apart from the quantitative one, in different application scenarios. It makes the application use easier.
- RET software supports several input formats such as XML (with or without temporal marks), TXT (plain text format, sentences format or TIME-TEXT format) and .SRT (**S**ub**R**ip Sub**T**itle files, with text and temporal marks). *NIST SCTK Scoring Toolkit* supports trn (transcript), txt (text), stm (segment time mark) and ctm (time marked conversation scoring) as input formats.
- The functionality of the algorithms used by both software tools is very similar. Regarding the input supported by the application, an adaptation of the algorithms functionality has been required. The algorithms of SCTK NIST and RET have not been compared as the input file formats from both systems are different and it was an unfeasible task.

3 Description of RET Tool

The RET architecture (Figure 1) is divided into three main modules, Graphical User Interface (GUI), IO File Manager and Text Comparator process.

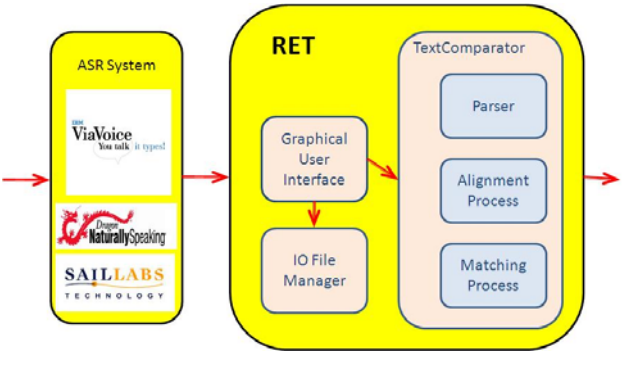

Fig. 1. Complete RET Tool Architecture

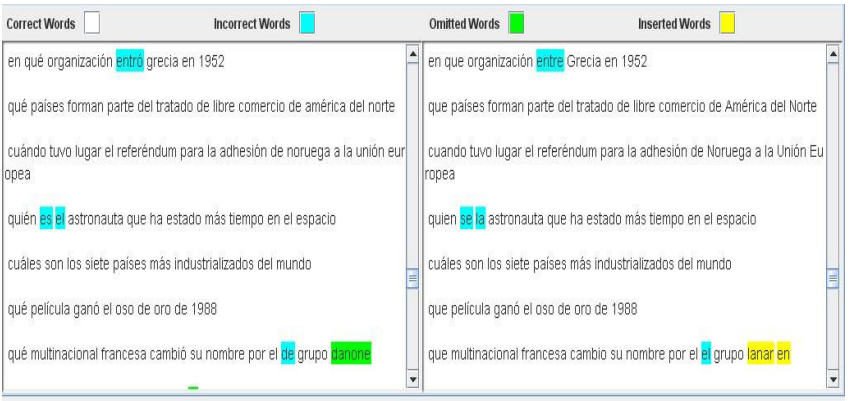

Fig. 2. RET Visual Results Image

Figure 2 shows the results of an evaluation example (on the left, the original text (OT) and on the right, the ASR output text (AOT)). A colours code is used to display the different types of errors: *white* for correct words, *blue* for incorrect words, *yellow* for inserted words and *green* for deleted words. Moreover, a tab with the numeric results, the numbers and a graphic bar chart are also shown.

TextComparator module compares OT and AOT files. It is made up of three different submodules: the parser, the alignment and the matching modules. Firstly, both files (OT and AOT) are parsed, obtaining two readable representations for the program to manage them. The procedure to obtain these objects is different depending on the format of the input file.

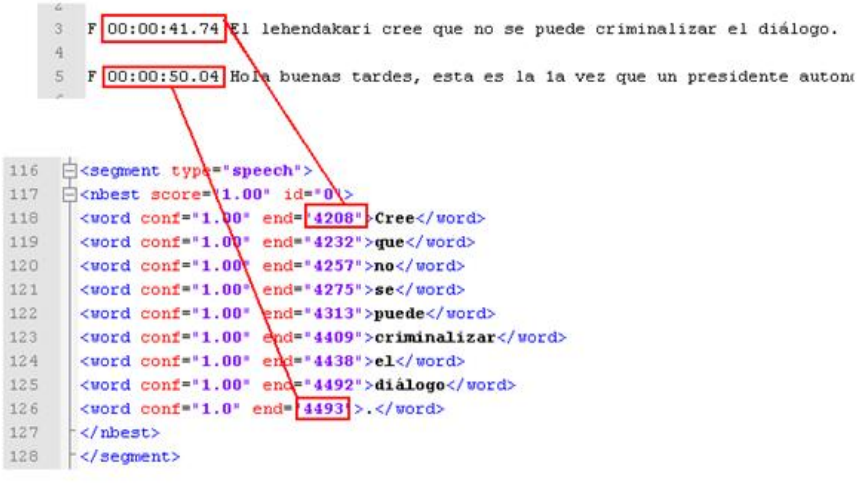
3 F 00:00:41.74 El lehendakari cree que no se puede criminalizar el diálogo.
4
5 F 00:00:50.04 Hola buenas tardes, esta es la 1a vez que un presidente auton
116
117 <nbest score="1.00" id="0">
118 <word conf="1.00" end="4208">Cree</word>
119 <word conf="1.00" end="4232">que</word>
120 <word conf="1.00" end="4257">no</word>
121 <word conf="1.00" end="4275">se</word>
122 <word conf="1.00" end="4313">puede</word>
123 <word conf="1.00" end="4409">criminalizar</word>
124 <word conf="1.00" end="4438">el</word>
125 <word conf="1.00" end="4492">diálogo</word>
126 <word conf="1.0" end="4493">.</word>
127 </nbest>
128

Fig. 3. Temporal alignment example

Once both files are parsed, the next step is aligning them. This process involves sentence matching of both texts, as, due to the different formats of the OT file (TXT) and AOT file (XML), the sequential order of elements of each sentence is not the same. Moreover, we must take into account those words that do not match up (because of any kind of recognizing error). This module aligns the texts obtained after parsing, and for this task, two different strategies are used: the first one aligns both texts by means of the temporal marks in the input file (an example is given in Figure 3); if there are no temporal marks, a positional alignment strategy is applied, to align by sentence or to obtain a plain text from the structured one.

The temporal alignment process takes the temporal marks from every sentence in OT and the temporal marks of every sentence in AOT. To align both texts, it is found a sentence in the AOT whose initial time is greater or equal to the initial time of OT sentence and whose final time is lower or equal to the final time of the OT sentence.

In the case of positional alignment, texts are aligned sentence by sentence. Figure 4 shows part of OT file and part of AOT file and their alignment. This means that the first sentence of OT is aligned with the first sentence of AOT, and so on.

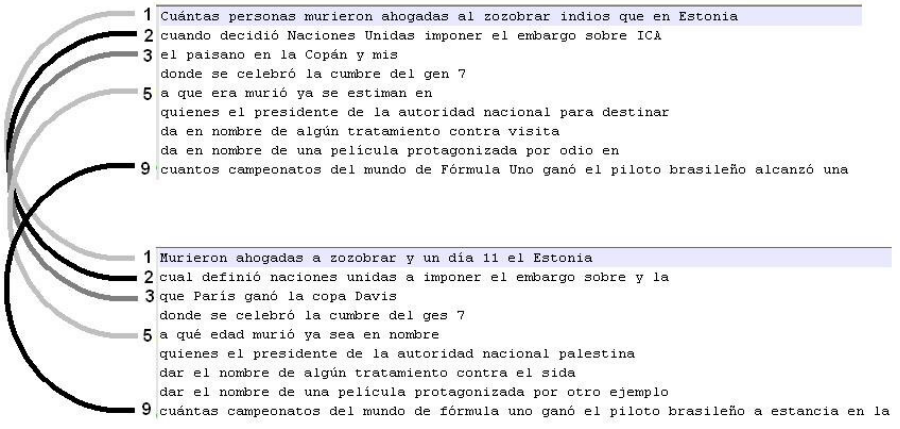

Fig. 4. Positional Alignment (Alignment by sentences)

After the alignment, the comparison algorithm is applied. Both pre-processed texts (parsed and aligned) are compared in the matching module. The algorithm takes one word from the OT and compares it to the words from the AOT. When it finds a matching word, the algorithm makes a complementary search of matching words along both texts, which is necessary because of the specific domain (speech recognition). This algorithm avoids a matching between two words that should not match. The complementary search algorithm is explained in the next pseudo-code:

Algorithm 1. Complementary Matching Algorithm

Input: S1 list of words of OT, S2 list of words of AOT,
S1i word i from text S1, S2j word j from text S2,
A1 position of matched word in OT,
A2 position of matched word in AOT,
D1 distance in OT, D2 distance in AOT
Output: B Boolean indicating if the two compared words are the ones that must be compared, that is, if the result is positive (true) both words are correctly matched and if it is negative (false) the words must not be matched.

```
i = A1 {position of word from OT to use}
j = A2 {position of word from AOT to use}
repeat
    {find new matching word in OT}
    if (S1i equals S2j) then
        k = j
        repeat
            {find new matching word in AOT}
            if (S1i equals S2k) then
                    restart method with A1=i and A2=k
            end if
            k = k + 1
        while (k is minor than length of S2 AND k is minor than D2)
        j = k
    end if
    i = i + 1
while (i is minor than length of S1 AND i is minor than D1)
```

Figure 5 shows the application of the complementary search algorithm over two sentences: the first one is the OT sentence and the second one is the AOT sentence. The first step of the algorithm takes the word 'Cuando' from the OT sentence, which is compared to the first word in the AOT sentence, 'Cuando'. Both words are equal, so the algorithm counts that there is a correct word. Then, the next word in the OT sentence, 'estábamos', is taken. This word does not appear in the AOT sentence, so the counter for incorrect words is increased. The algorithm continues until the word 'de' in the OT sentence is reached (marked 'Word 1' in Figure 5). The following word in the AOT sentence is 'en', so the rest of the sentence is searched until the word 'de' is found (labeled Matching Word in figure 5). At this moment, the algorithm would indicate that the Matching Word is the transcription of Word 1, an incorrect matching. But the complementary matching algorithm continues checking if the word 'de' appears again in the OT sentence, to ensure that the matching is correct. It finds another word 'de' (labeled Word 2 in Figure 5) and it has to decide if it should be related to the Matching Word ('de' in the AOT sentence) instead of Word 1. In this situation, the algorithm searches for another 'de' word in the AOT sentence. It fails, so the

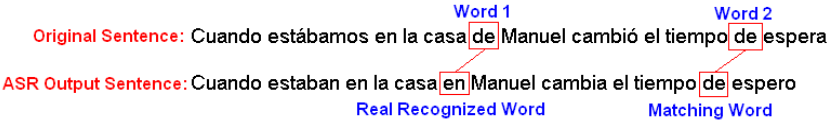

Fig. 5. Example of Complementary Search Process Profits

algorithm concludes that Word 2 should be linked to the Matching Word and not to Word 1. Thus, a transcription error for Word 1 should be counted. Of course, the algorithm can be parameterized to avoid dead ends while searching for matching words.

The standard measurements in speech recognition used to evaluate the ASR system are:

- *Correct* words rate = correct words (AOT) / words (OT) * 100.
- *Incorrect* words rate = incorrect words (AOT) / words (OT) * 100.
- *Omitted* words rate = not recognized words (wrong or right) (OT) / words (OT) * 100.
- *Inserted* words rate = inserted words (AOT) / words (OT) * 100.
- *Complete Correct Sentence Percentage* = number of sentences correctly recognized / number of sentences (OT).

SCTK NIST includes some other measurements besides these ones. For future versions, we planned to implement more measurements depending on specific user needs. Some examples of new measurements could be rates of specific words such as named entities, verbs, acronyms, etc.

4 Experiments

We introduce the experiments carried out with RET tool. All of them have been applied to the three different voice recognizers and with all the possible input formats.

The voice recognizers used for the experiments are IBM ViaVoice [3], Dragon Naturally Speaking [8] and Sail Labs' Media Mining Indexer (MMI) [6]. They are all commercial voice recognizers, and our aim was to compare them in order to choose the most appropriate one to accomplish tasks for other projects. ViaVoice 10.5 is a commercial speech recognizer that needs previous training by the user. It supports MS operating systems and incorporates keyboard shortcuts, specialized themes and the possibility of adding vocabulary. The output is word by word. We could not introduce audio files in ViaVoice, which meant a disadvantage to use it in some of the experimental scenarios. Dragon Naturally Speaking 9.0 shares many features with ViaVoice, but it does not accept specialized themes nor keyboard shortcuts. The Dragon Naturally Speaking output is phrase by phrase, needing several context words. ViaVoice and Dragon are speaker-oriented speech recognizers and they need a previous training process. However, we did not make a conventional training, but a multiuser one, using 10 different trainers, each one reading sentences from the basic text training provided by both programs. Finally, the main advantage of MMI version 5.0 is that it does not need previous training. The output is also phrase by phrase.

The scenarios where the program has been tested are two: a Question Answering System and an Audio-video Transcription System (divided in two sub scenarios), both in Spanish language (Castilian variety).

4.1 Question Answering System Scenario

As part of a biggest project on question answering, we tested the recognizers using as input 163 audio files containing questions read by 10 individuals (both sexes, different ages). They were short questions, asking information about important figures, celebrities, places, dates, etc. Some examples are: *Qué es BMW?(What is BMW?), Quién recibió el Premio Nobel de la Paz en 1989? (Who did win the Nobel Peace Prize in 1989?), Quién es la viuda de John Lennon? (Who is John Lennon's widow?), Cuándo se creó la reserva de ballenas de la Antártida? (When was the Antarctic whale reserve created?*. The recognizers were used to convert speech to text and later to send it to the question answering system.

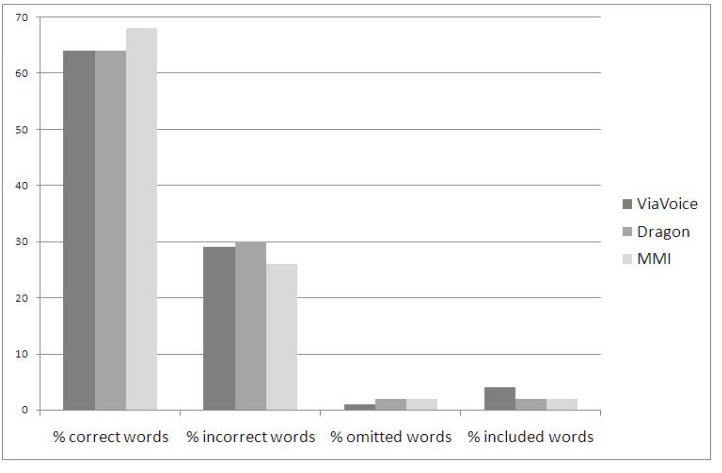

Fig. 6. Accuracy of the three speech recognizers in question answering scenario

The evaluation result of the recognition rate is shown in Figure 6. All systems are performing over a 60% of correct words rate. Structured texts have been used for testing[4].

The results of the evaluation provide numeric figures for the recognition rate, but not any accuracy value of the comparison between both texts. This can be seen in the graphical user interface. If we compare transcriptions with the OT using the visual results box, we can see that all speech recognizers are quite accurate due to the type of text (structured text).

[4] A structured text has a well-formed structure where sentences are systematically separated and can be easily parsed. An unstructured text has no defined structure or, even having a structure, the resulting separated sentences are too long to be considered.

4.2 Audio-Video Transcription

Video transcription for Information Retrieval

This work is focused on the use of a speech recognizer for making automatic transcriptions of audio and, subsequently, retrieving information from the resulting texts. For this task, the MMI was the chosen recognizer due to problems with ViaVoice to integrate audio files as input. As input, two newscasts video files were used; both of them last half an hour, and the difference between them is that while the first one is a national newscast, the 24h newscast addresses to an international audience. We made the comparison between the OT and the AOT from MMI to obtain measurements and assess the performance of the speech recognizer. The results are presented in Table 1.

Table 1. Results obtained with newscasts video files

	Newscast	Newscast 24h
% correct words	55	32
% incorrect words	32	48
% omitted words	7	9
% inserted words	3	9

In this case, the comparison process is carried out on structured texts, but these are formed by sentences long enough to be considered a mid-way point between structured and non-structured text scenarios. The comparison results are better than for the non-structured texts, but still worse than for the complete structured-texts. The difference between both results relies on the audio files used for the second test, which presented a higher noise level and this is reflected in the numeric results.

Real-time captioning system in a classroom

Another important scenario was a subtitling application for students with hearing impairment that transcribes the teacher's speech with the help of an ASR system, converting the spoken lesson into a digital resource. This content is available in real time for deaf students in form of captioning or as plain text, in paragraphs, where the user can navigate the whole transcription. A secondary task, apart from live subtitling, is the possibility of retrieving learning objects using subtitles to index video recorded in classrooms and helping students with disabilities in the learning process [5]. The evaluation has been carried out at the Carlos III University of Madrid during a 3th year subject of Computer Science degree called "Database Design". The teacher previously trained Dragon Naturally Speaking version 9 (DNS). Training duration was 30 minutes approximately, reading specific texts given by both ASR products. Additionally, specific vocabulary of "Database Design" subject was independently introduced and trained.

Four experiments were performed: (1) speech recognizer's basic model, (2) basic model and training, (3) basic model and specific vocabulary and (4) basic model, training and specific vocabulary. Figure 10 shows the figures provided by RET for the different tests.

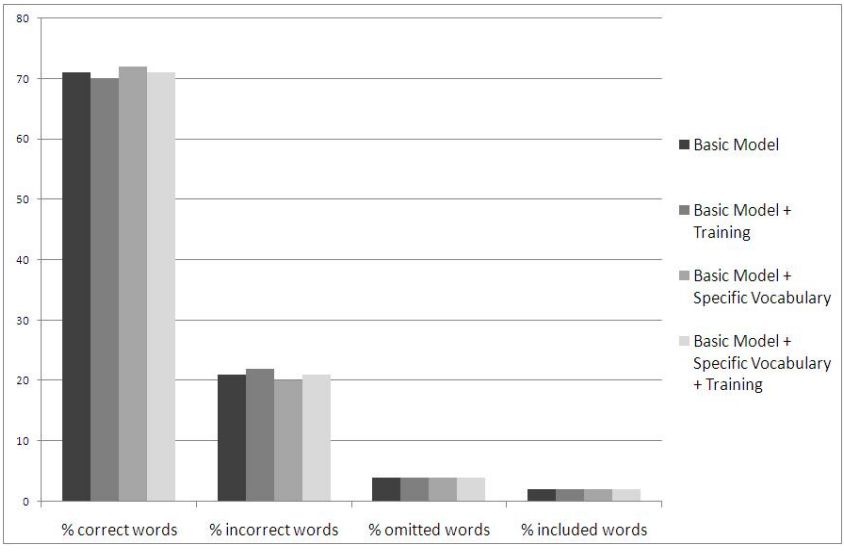

Fig. 7. Comparison of four tests in the real-time captioning scenario

The results obtained after the comparison show a high degree of accuracy for non-structured text, although it is usually poorer as the comparison process was not designed to work with this kind of texts.

As the algorithm does not work properly with non-structured texts, these results are due to a manual pre-processing of the texts, dividing them in two parts. Besides, the distances used in the 'complementary matching' algorithm were also adjusted to obtain the optimum value of the comparison results.

The scenario for this task (a classroom) involves dealing with spontaneous speech, even though the discourse is previously planned. This means the existence of typical elements of spontaneous speech as disfluences, self-interruptions, false starts, hesitations, all of which make the recognition process difficult. Owing to this fact, there is not much variation between the four tests, as training and vocabulary insertion do not provide better results. Moreover, keywords are not distinguished from stopwords, so, even introducing specific vocabulary, the total percentage does not improve as it is made up including stopwords.

5 Conclusions

Historically, the evaluation of ASR systems has been a quantitative evaluation, but also qualitative output is necessary and makes easier the task of testing an ASR system. Currently, SCTK software performs a quantitative evaluation, but it did not fit specific needs such as to work with XML file formats or a simple user interface to analyze transcription errors.

There are some features that distinguish RET software from SCTK. Firstly, the GUI is intuitive and friendly and makes the tool easier to use. The displayed results provide useful information, facilitating the interpretation task. RET supports different input file formats that fit our needs. And finally, measurements calculated by both systems are the same, but in our case we can easily increase the number of numeric results depending on specific needs.

The experiments are representative of the different text types which the software can deal with. For every experiment one of them has been used: (1) Structured text; (2) Unstructured text; and (3) Midway point text[5].

The numeric results from the ASR recognition rates, which are those given by the RET software, do not depend on the type of text. Texts features affects the quality of the comparison, being higher for structured texts, lower for midway-point texts and presenting the worst results for unstructured texts. This is consistent with the fact that the algorithm was design to work with structured texts and later adapted to deal with unstructured texts.

6 Future Work

As future work, one of the main improvements planned for RET is the increase of the number of evaluation measurements. Furthermore, several improvements are: (a) adding PoS (Part-of-speech) tagging to transcriptions to analyze which are the most problematic kind of words, for instance, named entities, verbs, acronyms, etc; (b) taking into account the length of words and sentences and (c) dividing the sentences into long and short, establishing a threshold to delimit them. Another important enhancement will be the creation of an output report with the comparison and the evaluation results.

Regarding the algorithms, future work aims the following:

- Allowing the user to keep a results history and establishing a fixed storing protocol for text comparison.
- Improving the comparison algorithm to manage continuous text (unstructured-text) or at least structured long texts. The use of punctuation marks could be useful for both the alignment and the comparison algorithm.
- Polishing the alignment algorithm, since increasing the accuracy of aligned texts, the comparison results will improve noticeably. Also, we must solve the problem of alignment for plain texts without temporal marks.
- Improving the 'Complementary Matching' algorithm to develop an automatic way to obtain the optimum values for the algorithm.

Acknowledgments. This research work has been supported by the Regional Government of Madrid under the Research Network MA2VICMR (S2009/TIC-1542) and by the Spanish Ministry of Education under the project BRAVO (TIN2007-67407-C03-01).

[5] Midway point text: structured text in which the text is long enough to be considered as unstructured one.

References

1. Altova, Altova DiffDog, http://www.altova.com/products/diffdog/diff_merge_tool.html (viewed July 2010)
2. DiffDoc, Softinterface Inc., http://www.softinterface.com/MD/Document-Comparison-Software.htm (viewed July 2010)
3. IBM ViaVoice, http://www-01.ibm.com/software/pervasive/embedded_viavoice (viewed July 2010)
4. IDM Computer Solutions, Inc., UltraCompare, http:// www.ultraedit.com/loc/es/ultracompare_es.html (viewed June 2010)
5. Iglesias, A., Moreno, L., Revuelta, P., Jimenez, J.: APEINTA: a Spanish educational project aiming for inclusive education In and Out of classroom. In: 14th ACM–SIGCSE Annual Conference on Innovation and Technology in Computer Science Education (ITiCSE 2009), Paris, July 3-8, vol. 8 (2009)
6. Media Mining Indexer, Sail Labs Technology, http://www.sail-technology.com/products/commercial-products/media-mining-indexer.html (viewed June 2010)
7. NIST, Speech recognition scoring toolkit (SCTK) version 1.2c. (2000), http://www.nist.gov/speech/tools (viewed May 2010)
8. Nuance Dragon Naturally Speaking, http://www.nuance.com/naturallyspeaking/products/whatsnew10-1.asp (viewed March 2010)
9. WinMerge, http://winmerge.org/ (viewed June 2009)