Machine-Aided Spoken Language Evaluation: The Current Approach

Brian Teaman

コンピューターによる会話の分析：新しいアプローチ

ティーマン　ブライアン

Abstract

This paper is a report on the current state of the Machine-Aided Spoken Language Evaluation（MASLE）system. MASLE is a computer system being devised to aid in the evaluation of second language speech. This system is used to first elicit and collect spoken language and secondly to aid in the evaluation of language by presenting the recorded speech to a human rater or a computer which uses automatic speech recognition. This paper will describe the project in its current state and then outline some details of the web-based recorder, the jukebox for human raters and a machine rating system.

Key words : Computer Assisted Language Learning, Speaking Test,
Automatic Speech Recognition

（Received September 30, 2005）

抄　　録

この論は MASLE システム（コンピュータによる会話の分析）の現状を考察するものである。MASLE とは第二言語の習得度測定を助けるために開発されたものである。このシステムは当初は会話を引き出して収録するための役にたてるものであった。次いでは言語使用を評価するために役立てるものとして、人間の評価者もしくはオートマティック・スピーチ・レコグニションを使用するコンピューターにデータとしての会話の録音を提供するものであった。本論はこのプロジェクトの最近の様相を述べた後、ウエッブを用いた記録、すなわち人間の評価者およびコンピューターによる評価システムのためのジュークボックス、について説明するものである。

キーワード：CALL、スピーキングテスト、オートマティック・スピーチ・レコグニション

（2005年 9 月30日　受理）

Overview of the MASLE System

The Machine-Aided Spoken Language Evaluation (MASLE) project, was originally conceived of a few years ago and first reported in Teaman (2004). In the Spring of 2005, this project was awarded a grant by Japan's Ministry of Science and Education. These funds will be used for software, hardware, project assistance and research related travel. This current report documents an updated overview of the system.

MASLE (pronounced "MAZZly") is a computer system being created to aid in the learning and evaluation of second language speech. MASLE will be a sophisticated network voice recorder and rater jukebox, as well as a device used to evaluate speech. In recent years, automatic speech recognition (ASR) has become common in many contexts, such as calling a large company. ASR is also slowly appearing in software for computer assisted language learning, but is still fairly uncommon and limited in scope.

This project is aiming to build a set of tools so that many different types of language data can be recorded and then evaluated. Currently, the only system available that is comparable to MASLE is the PhonePass testing system (Phonepass, 2005). PhonePass is a telephone-based system used for testing spoken English as a second language. The biggest advantage of PhonePass is that it is the only working system available for testing. In contrast, MASLE is a work in progress that will have the following advantages over PhonePass:

1. MASLE's ASR component will be developed specifically for Japanese speakers of English.
2. MASLE can be targeted specifically to a specific demographic group. The first version will be prepared for students of Osaka Jogakuin College. This allows for a test for a specific target audience which should increase the ability of the test to give appropriate feedback.
3. MASLE allows for human as well machine-based evaluations of speech. This allows for greater flexibility in testing situations.
4. MASLE can be done on a stand-alone computer, over a LAN or the Internet.
5. MASLE can be used for applications other than just testing.
6. MASLE's human evaluation component can be quickly applied to any language and not only those for which ASR models exist.

It can be seen from these features that MASLE is much more flexible and offers many advantages over a static phone-based test. MASLE does not specify any one test, method or language, but collects spoken data for any number of kinds of tests or exercises for any

spoken language.

In the following sections I will describe the modules that are currently being worked on: the MASLE recorder, the grader jukebox and the automatic grader.

The MASLE Recording Module

The first module of MASLE is the recorder or the data collection component. The goal of the recording module is to provide a flexible interface for collecting spoken language from the language learner over a network. It is an interface created for the purpose of recording speech data from learners. I purposely avoid calling this a testing program because it could be used in ways that are not strictly testing such as peer evaluation, self evaluation, or providing additional speaking practice. It is being created to present stimuli or prompts such as audio, image, text or any combination of the three. The user is expected to then respond orally to these prompts as directed by the system. This oral response is then recorded for analysis in one of the next two modules to be described. The recording module will have the following components:

2-1 A program that will run over a network
2-2 A database designating the parameters of the test
2-3 Recording software
2-4 A database for collecting the output of the test

The core program (2-1) will be written in some variety of HTML/XHTML with PHP (v. 4.3) as a control language. The database that the program will access could be merely a text file for text data and a directory to record the audio data. However a database language such as MySQL could also be used as a more secure way to handle these database features. The program will run through a web browser and will access the database (2-2) and display a prompt and then record the speech. The database will have a structure that will contain information such as the following:

3-1 The data that will be used for a prompt
3-2 The name of the audio file that will be recorded
3-3 The time limit for the recording (if any)

With this information, the core program will know what to display (3-1) as a prompt for each item (whether it be text, audio, some visual object or some combination of the three). It will also know what to name the audio file (3-2) that will be recorded. Furthermore it will limit the time of the recording. The recording software that is currently

being used is Listenup (2005) which allows for user controlled completion of recording or automatic completion by setting a time limit. Finally the data will be written to a file (2–3) and will include information such as the following:

4–1 The speaker ID
4–2 The item ID
4–3 The audio data

The audio recording and the database of the recording session will be used later by the human and/or the machine rater.

A hypothetical session will then run as in the following description. A program will display the first prompt while instructing the user to read or describe the item so it can be recorded and uploaded to the server. After this, the program goes on to the next prompt and then more speech will be recorded. This repeats as many times as is necessary to finish recording the speech for the current task.

Currently, there are limitations with the interface, so that the user has to push the record button, stop the recording and then push a button to upload. Ideally, one could create a test that will do all of these things automatically. It is not currently possible to use the ListenUp software this way, but it will be a goal of the project to work this out so the interface can run as smooth as possible without requiring so many actions by the test-taker and thus reducing chances for errors.

After the speech databases are created by the recording program, the audio is now ready for either the jukebox or the machine grader. In the following sections I will describe the features of these modules.

The MASLE Grader Jukebox Module

In The second module, the grader jukebox, MASLE provides raters with an interface which plays back recorded material and prompts the rater for a rating. The Jukebox will consist of the following five parts:

5–1 A program that will run over the web and prompt the grader
5–2 A database designating the parameters of the test
5–3 Playback software
5–4 A database of the output audio recorded by the recorder
5–5 A database for the output of the grader judgment

The grader jukebox will work in a similar fashion to the recorder component (5–1), but

with some differences, most notably that it does not record but only plays the audio previously recorded by the recorder module. It will also run on a combination of XHTML and PHP with a text or MySQL database. The parameters of the test mentioned in 5–2, will be the number of items on the test and the number of speakers and their IDs. Playback software (5–3) could be handled by the Listenup software which also handles the recording. However any number of free players are available for playing audio files over a network. The audio database mentioned in 5–4 is created by the recorder module described above. Finally, the rater judgment provided by the listener can be passed to a database (5–5).

A hypothetical session would consist of the program playing audio produced by the recorder program and prompting the rater for a rating. The program would then continuously prompt the rater with new data and collect the response iteratively until all of the speech was rated. The program could simply go through the speech in a linear fashion or randomly present data from any number of speakers. Furthermore, the jukebox program could be programmed to play each recording any number of times to further test the consistency of the rater herself or the consistency between different raters.

The MASLE Automatic Grader Module

The third and final component to be described here is the automatic grader module. In this module MASLE will allow for automatic grading of speakers rather than rating by a human as described above about the module. This section will describe the steps needed to be taken in order to make the ASR work. The automatic grader is being built on the HAPI (Odell et. al., 1999) interface to the HTK speech recognition program (Young et. al., 2002). The steps of writing the grader are:

6–1 Write a grammar of the expected speech.
6–2 Write a dictionary of all the words and their forms in the expected speech.
6–3 Run the speech through the grammar to get the recognition results.

The first step (6–1) is to write a grammar. I will explain the case of the English sentence “Hi, how are you?”. The grammar looks like the following:

(SIL HI [SIL] HOW [SIL] ARE [SIL] YOU SIL)

Where SIL is silence and brackets demarcate optional material. In this case there are optional silences between words in the target sentence and required silences at the beginning and end of the entire utterance. Including these silences is a standard procedure

used when writing such grammars.

The next step is to create an ARPABET dictionary (represented in Table 2) with the grammar's words that are found in the grammar. Here is what the dictionary looks like:

Table 1: Words recognized as "Hi How are You" in two test sentences.

WORD FROM GRAMMAR	ARPABET representation
ARE	aa er
HI	hh ay
HOW	hh aw
SIL	sil
YOU	y uw

This simply describes the phonology of each word using ARPABET symbols. The column on the left lists the words in the grammar and the character sequences on the right represent the arpabet sounds that correspond to the words of the grammar. For example, the word "are" is composed of the sound /a/ as in *bottle* (ARPABET "aa") with the final sound of /r/ as in "far" represented by "er" in the ARPABET.

Table 2 shows sample results of running two recorded phrases through the recognizer using the above grammar and dictionary. The first phrase was the correct phrase "Hi how are you", while the second, one was an utterance containing none of the correct words: "Why oh why, Ohio?" This was used to test the system.

Table 2: Words recognized as "Hi How are You" in two test sentences.

Test sentence	HI	HOW	ARE	YOU	Total confidence
"Hi how are you"	Y	Y	Y	Y	100%
"Why oh why, Ohio"	N	Y	N	N	25%

In the first case, the four words were recognized properly. In the second case, the phrase "Why oh why, Ohio?" was forced through the recognizer to see if any false positives were generated. A false positive was generated for one out of four words. Somewhere in the phrase, the system thought there was a close enough match to some portion to the word "HOW." This shows that the system is not perfect but that it will probably yield enough appropriate results to get a meaningful final score. It showed that the correct phrase got a much higher score than the incorrect phrase, which is a step in the right direction. This was the first attempt to use the recognizer, so more testing will be needed to create test sentences which will allow the recognizer to make appropriate judgments of non-native speakers at various levels. It does not have to be perfect, however, in a testing situation. It

only needs to reliably produce a higher score for well-spoken English than it does for English that does not match the model.

Conclusion

This paper has outlined the current state of the MASLE project. There is yet a lot to be done to get a working product together, however all the pieces are in place. What is needed is to further elaborate the simple prototypes that have been created for the recorder and jukebox. The machine rater will be improved by running extensive testing on native and non-native speakers. Furthermore, programs will be written to convert the HAPI recognizer output to useable ratings of non-native speakers.

Bibliography

ListenUp (version 1.54) (2005) [Computer software]. Softsynth.com.

Odell, J. et. al. (1999) *The Hapi Book: A Description of the HTK Application Program Interface.* Entropic, Speech Technology Inc.

PhonePass (2005). A phone-based test of spoken English. Ordinate Corporation.

Teaman, B. (2004). *An Introduction to MASLE: Machine Aided Spoken Language Evaluation.* Hiroshima University Journal of Language Teaching and Research, Vol. 7:39–49.

Young, S. et. al. (2002) *HTK: Hidden Markov Model Testing Kit,* Cambridge University.

Acknowledgement

This research was partly supported by the Ministry of Education, Science and Culture of Japan under Grants-in-Aid No. 17520406.