Automatic feedback on phonemic transcription
M.L. García Lecumberri, University of the Basque Country
M. Cooke, University of Sheffield
J.A. Maidment, UCL

1. Introduction and motivation
This paper describes a tool which provides feedback on transcription attempts. Phonemic transcription is one of the requirements of many phonetics and phonology and even general linguistics courses at university level. Transcription of one’s native language is a useful mechanism for raising awareness of the sound system and its behaviour in connected speech, making the speaker’s competence explicit.

Among language learners, transcription may not only raise awareness of their existing knowledge of the Target Language (TL) sound system, as in the case of native speakers, but it also makes learners aware of their pronunciation errors. Additionally, it can be used as a teaching tool by providing students with visual images of the TL’s pronunciation and also as a diagnostic technique for the tutor’s assessment of pronunciation. Accordingly, transcription is also used in language courses for non-linguistics students (Koet 1990).

It is well known that English is one of the languages in which spelling is particularly unhelpful as a representation of words’ phonemic shapes: "In the case of English, the use of a phonemic transcription system is specially important because the language has no simple sound-symbol correspondence system" (Celce-Murcia et al. 1996, p. 38). This is probably one of the reasons why transcription is particularly frequent in English language courses.

Nevertheless, transcription is not just a reflection of speakers’ pronunciation, it is also a skill, and as such it requires regular practice to improve it (García-Lecumberri & Maidment 2000). Regular practice should also, ideally, be complemented with specific feedback. This is one of the areas where instructors find difficulties, since providing frequent personalized and detailed feedback to a group of students is very time consuming. Therefore, student self-monitoring becomes a necessity, particularly in overcrowded teaching environments.

However, autonomous transcription learning can often be frustrating. Resources available to the student, such as pronunciation dictionaries (e.g. Wells 2000) and phonemic descriptions of the language cannot reflect the numerous variations which words may experience in context, or, may give general directions for contextual changes but cannot possibly apply them to each particular situation and item. It is then up to the students to combine citation forms with contextual rules to produce the appropriate result, but without feedback as to whether their deductions are correct. Transcription books which include comments and explanations on connected speech phenomena (e.g. García-Lecumberri & Maidment 2000) help fill the gap between citation form consultation and personalized feedback from instructors.

Some simple web-based applications have already been developed by members of the SIPhTrA (System for Interactive Phonetic Training and Assessment) at UCL. Single word transcription is catered for by a JavaScript program called Transcriber (Maidment, 1999), which presents English targets as orthography, pr-recorded sound clips, or both. A somewhat more sophisticated tool is the JavaScript program called Photon (Maidment et al, 2000). This presents the user with an on-screen phonetic keyboard and allows the user to hear a synthetic speech version of their input. Both of these programs suffer from the absence of cross-browser compatibility and both need the user to have access to an up-to-date version (at least version 5.1) of Microsoft Internet Explorer.
These considerations, and a fortuitous coincidence of people with varied skills, were the trigger for the current project: the development of means to provide automatic and personalized feedback on student transcription attempts. This initial aim has taken shape in the form of a tool which is described in this paper. The tool is in the initial stages of development and testing with ‘live’ subjects. In the following sections we expand on the design goals for the transcription tool, then go on to describe what the transcription tool can already do and how this was achieved technically. Finally, we mention some of the tasks we hope a future version will be able to perform.

2 Design
The main design goals for the tool were to develop a platform-independent, easy-to-use facility for the input and analysis of phonemic transcriptions for learners. Particular emphasis was placed on the requirement for effective feedback, which must be rapid, correct and relevant. Multiple pronunciation alternatives and optional pronunciations should be supported. Additionally, the tool should be easy to adapt to other languages. Further, it should support the storage of a base of material for L1 vs L2 research. Although the initial goal is to provide a tool for phonemic transcription, the tool’s design should allow the later incorporation of extensions required for a narrow analysis, such as a greater range of diacritics and support for audio playback.

3 Description of the initial version
The tool in use is depicted in figure 1. The student initially selects a transcription task, which results in the loading of the orthography in the upper panel. The transcription attempt is entered in the lower panel using the phonemic keyboard. Additionally, customisable key-bindings can be used to allow keyboard entry of symbols. The phonemic keyboard is itself fully customisable and contains the symbols and grouping of symbols suitable for the accent and processes defined by the tutor. For instance, it would be possible to limit the choice of symbols to reflect those introduced in an accompanying lecture course. The only pronunciation model developed at present is RP, so the tool requires student transcriptions according to this accent. However, there is no accent bias in the tool itself.

Behind the scenes, the tool has access to a model transcription provided by an expert. A ‘tutor’ version of the tool can be used to input such transcriptions and orthographies. The model transcription is not merely a linear sequence of symbols, but is marked up to denote alternative and optional pronunciations. The tutor interface is specialized to contain addition symbols required for marking up alternatives.

Once the student is satisfied with the transcription, s/he submits it, at which point the tool computes the optimal alignment with the model. Errors are

\[\text{Figure 1: the transcription tool}\]

---

1 The tool is not designed for large-scale manual transcription of corpora in the style, for instance, of Transcriber (Barras et al, 2001).

2 The word "error" is used to indicate divergences between the model and student versions. At a later date, we hope that all divergences will correspond to actual errors, but at the moment there is the possibility that a difference may be due to alternative pronunciations which have not been contemplated and thus are not present in the model. These alternatives may refer to some connected speech processes which have not been included yet (such as assimilations, some segment...
categorized as *confusions*, indicating the substitution of one symbol for another, *missing*, denoting the absence of a required symbol, and *extra*, suggesting an additional symbol. Feedback is currently provided by highlighting words or symbols, and by listing errors.

3.1 Highlighting of *words*: each word containing an error is highlighted. The student then looks at these cases, makes the changes deemed necessary and resubmits. This process may be repeated several times, until the student decides to give up and display the correct model.

3.2 Highlighting of *symbols*: here, only the actual divergent symbols are highlighted. The correction task is easier for the student than the previous one, since s/he does not need to guess where in the word the mistake occurs. Corrections and resubmissions follow the same steps as before. It should be noted that missing symbols are not marked as such in the current version since to do so would have the potential of confusing the aligner on resubmission.

3.3 Error listing: The other approach is to list errors ordered by frequency of occurrence, as shown in figure 2. Feedback text on specific errors is also displayed in the list. In this way, specific advice on common errors (such as /s/-/z/ confusion) can be given. Here, the student needs to scan the transcription to try to find error locations, correct them and resubmit.

3.4 Task-specific control of feedback: although different feedback options are available, it may be desirable to control their availability in a task-specific manner. The tutor can customize the tool for each particular student or group of students, so that some feedback options are excluded altogether, or only presented after a certain number of attempts have been made. Ideally, the tool would moderate the use of the different feedback methods according to the student’s progress, or so that they are alternatives to make the use of the tool more varied and interesting.

In the current system, the tutor is able to specify which feedback options are available and when.

![Figure 2: Listing of errors](image)

Typically, this would be used in a sequence from less to more informative (e.g., divergent words, then error list, then divergent symbols). The possibility of several re-submissions coupled with the ability to define a feedback policy is an important pedagogical feature of the tool, since it allows the student to pitch his/her abilities against the tool without disclosing the correct transcription after the first attempt. However, since the feedback policy is currently in the hands of the tutor, care is required to facilitate student-centred choice of task-feedback based on personal learning style or requirements. In the initial stages of a transcription course, this policy might be quite liberal and involve no more than refusing to show the correct transcription until a few attempts have been made, but preventing the undue frustration a requirement of too many trials might cause. This degree of autonomy is important to allow the student take charge of his learning process and also adapt the tool to his own needs/characteristics. In the case of students learning transcription within a formal course unit, their instructor may want to make the choice depending on course goals and student groups characteristics. Students active participation in

---
deletions) or, of course, to accent variants which are outside the scope of RP itself.
their learning process (for example in the design of activities as is the case here) has been proved to be very beneficial not only as immediate progress/results are concerned but also as conductive to a profound significant learning which may have more long term permanence (Hill 1994, Newble & Cannon 1995).

4 Implementation
4.1 Alignment: At the core of the tool is a string aligner based on a modified dynamic programming algorithm. The aligner finds the optimal sequence of string edits required to match the student’s transcription attempt with that provided by the tutor. The algorithm accommodates optional symbols and single symbol alternatives but currently employs no linguistic constraints.

4.2 XML: To facilitate portability, customization and future localisation, all data used by the tool is marked up using the extensible markup language, XML. XML not only allows the definition of custom markup tags, but, when used with a suitable parser and toolkit, combines data portability with ease of programmatic access. Currently, four XML document types have been defined for the transcription tool to handle phonetic symbols, orthography and associated transcription, student transcription attempts and feedback text for common errors.

4.3 UNICODE?: Ideally, a modern transcription tool would utilise UNICODE phonetic symbol designations. Unfortunately, the lack of universal support for the phonetic subset of UNICODE in browsers forces an alternative solution. Since the tool is aimed at learners, ASCII schemes such as SAM were ruled out in favour of the use of images. Image composition requires more computation than character-set display, but the time required to display a typical paragraph on is acceptable.

4.4 Java: The tool is implemented in Java and consequently runs on all common platforms. At present, it is a standalone application, but will be available on the web as a Java applet in the near future.

5 Future work
The basic functionality of the tool (eg feedback) has been tested by analyzing the responses to a corpus of 15 L2 transcription attempts (containing roughly 250-300 symbols each), derived from previous exam papers. In the near future, responses from both L1 (UCL) and L2 (UPV) transcription students using the tool with be collected. Other planned developments include, migration to the web; localization to different accents/languages; incorporation of audio; support for narrow transcription; and better support for alternative pronunciations.

6 References


