

適性化線上英語聽寫測驗系統之研究

Toward Computer Assisted Learning for English Dictation

黃上銘 劉昭麟
政治大學 資訊科學系
{g1753006, chaolin}@nccu.edu.tw

高照明
台灣大學 外國語文學系
zmgao@ntu.edu.tw

摘要

適性化是許多現代資訊系統所需具備的重要特徵。好的電腦輔助教學系統更必須能夠因材施教，適時提供適合使用者程度和需求的教材，才能夠輔助學生在比較低的壓力之下作到有效率的學習。本論文提出能夠偵測學生聽寫測驗的錯誤的若干方法，藉由這一些偵測機制，我們可以分析學生部分文法層次，單字層次，聽音層次的錯誤。最後我們依據所觀測到的錯誤型態，提供給學生合適的教材或是練習題，以輔助學生增進其英語聽寫能力。

關鍵字：電腦輔助英語教學、電腦化適性測驗、minimum edit distance

1. 引言

在電腦化適性化測驗 (computerized adaptive testing, 以下簡稱 CAT) 裡，呈現給考生的試題順序，是依據受測者在前一個試題上的表現好壞來決定。使用者透過電腦接受測驗，每回答一個問題，系統會根據受測者的作答，評估其能力，然後依照評估的結果，決定後續的出題。相較於紙筆測驗，若有良好的題庫，並且能精確估計受測者的能力，CAT 不僅可以做到因材施教，還可以節省大量的成本，以及冗長的施測時間 [14]。

而隨著全球資訊網的普及，透過網路來進行教學的互動方式，也逐漸流行起來。以 Web 為基礎的測驗系統 (Web-based testing systems, 以下簡稱 WBT)，管理方便、作業平台獨立、操作介面簡單，這些優點都是傳統的電腦測驗系統所欠缺的 [10, 14]。本研究提出一個線上英語聽寫測驗系統，其特色為自動出題，並能針對受測者弱點，提供適性化的練習題目，在此基礎架構上，可以用於輔助線上英語教學，或與其他英語測驗系統相整合。

在目前的英語教學系統之研究中，Michaud 等人的 ICICLE (Interactive Computer Identification and Correction of Language Errors)，能夠提供聾人學習書面英語的指導，以輔助課堂上的教學 [10]。該系統能夠找出使用者所寫的英文作文的文法錯誤，然後針對使用者的表現，給予詳細的教學回饋。一開始，使用者先傳送一篇作文給 ICICLE，系統分析出文法上的錯誤後，會回饋出教學性的指導，以幫助學生自我訂正。當使用者把錯誤的地方更正好之後，可以再餵給系統，進行分析。

Tsiriga 與 Virvou 的 Web-PVT (Web Passive Voice Tutor)，目標在於教授英語的過去式時態 [16]。Web-PVT 整合了智慧型教學系統 (Intelligent Tutoring Systems) 以及適性化超媒體 (Adaptive

Hypermedia) 的技術。他們兩人認為，Web-PVT 能夠為各別的使用者，提供適性化的教材與回饋。Web-PVT 的系統中，具備了使用者建模 (user modeling) 的功能，可以將使用者目前的知識程度，以及可能發生的觀念誤解，加以追蹤記錄。當使用者從網頁上瀏覽教材的時候，系統可以根據這些資訊，適時的支援。

而在英語測驗系統方面，已經有多位研究者，針對字彙測驗的半自動化出題，提出研究報告。Stevens 從一般用途的語料庫 (corpus) 中，利用用語索引 (concordancer)，產生字彙試題 [15]。Wilson 從電子字典和語料庫中，自動產生練習試題，以用於電腦輔助語言教學 (computer adaptive language learning) [17]。Coniam 則用語料庫中 word frequency 的資訊，來自動產生克漏字測驗 [3]。Gao 發展了 AWETS (Automatic Web-based English Testing System) [5]，該系統利用語料庫及字頻資訊，促進字彙題的產生。AWETS 由四個獨立的模組所構成：試題產生模組、測驗發送模組、成績記錄模組以及試題分析模組。在目前最新的研究中，AWETS 已經不僅止於字彙出題，還能針對聽說讀寫四項來測驗學生的能力。

本研究以 AWETS 為基礎，針對其聽寫測驗系統的部分，加入適性化測驗的機制。接下來，在第 2 節中，我們概述本系統的功能。在第 3 節中，則說明分析聽寫錯誤的方法。第 4 節，說明如何定義錯誤類型，並提供適性化的練習題目。最後，第 5 節總結，並討論本系統未來的展望。

2. 系統概述

AWETS 英語線上測驗系統，可以針對聽說讀寫來出題 [5]。其中，聽寫測驗的操作流程如下：首先，施測者在施測前，先透過系統提供的網頁介面，設定聽寫測驗的題目。系統會從題庫中隨機選題，播出一段話語。話語的內容，是我們請人讀稿所錄下的聲音檔，每段稿一句話，大約有 6 到 13 個單字。受測的學生聽完之後，將答案填寫在網頁中央的文字框中。假如學生聽寫完，覺得滿意的話，就按確定鈕，結束該題的作答。倘若學生對於自己的作答不滿意，或者有些地方還聽不清楚的話，可以選擇重填。系統會重播題目的內容，讓學生聽寫。前後總共可以作答三次。學生作答完後，系統會批改學生聽寫的答案，然後把批改的結果，輸出在網頁上，報告學生這題所得的分數。假如學生的作答中，有漏寫單字的話，系統會將該單字，用紅色的中括號標記出來。假如學生多寫了其他的單字，系統則會用紅線，把該單字劃掉。整個聽寫

測驗作完之後，系統會把學生作答的結果，記錄下來，並把學生聽寫錯誤的單字統計起來。

本系統在這樣的介面基礎上，再加入適性化練習的功能。學生作答完之後，系統會將標準答案，與學生的作答相比對，分析其中的錯誤。假如學生的聽寫有錯誤的話，系統會提供跟這個錯誤相近的題目，幫助學生自我評量。舉例來說，系統出一題：“Be sure to take the right road.”，而學生的作答則是：“Be sure to take the right load.”。受測者將“road”聽寫成“load”，系統會分析出，受測者對於“l”跟“r”的發音，可能分辨不清楚，於是會從題庫中，找出有“r”或“l”發音的題目，提供給使用者練習。

3. 聽寫錯誤之分析

當受測者聽寫發生錯誤時，原因可能很多。我們觀察學生的作答，認為聽寫出錯的原因，至少有以下幾種可能：

- 受測者對於該單字的發音，聽不清楚。
- 受測者對於該單字的發音，雖然能聽清楚，但因為以前沒學過，或是一時想不起來，而無法確定這是什麼單字，甚至判斷錯誤。
- 受測者對於單字的發音，雖然能聽清楚，但因為語言文化的差異，而聽成其他音標。
- 受測者對於單字的音義，雖然能聽懂，但難以完全掌握整句話的結構。
- 受測者對於單字的音義，能夠聽懂，對於整句的結構，也能掌握。卻因為筆誤而出錯。

我們在這裡列了五種受測者聽寫犯錯的可能原因，但實際上，聽寫發生錯誤的原因非常複雜，我們難以一一列舉出來。而為了讓系統能夠因材施教，達到適性化的目標，我們必須要能從受測者的作答中，推測出受測者犯錯的原因，而我們解謎的線索，也僅僅只有受測者所提供的作答而已。儘管如此，我們還是可以從中抽絲剝繭，找出一些端倪來。

依據我們目前有限的真實資料，把受測者的作答，與標準答案相比對的話，可以觀察到三種不同層次的錯誤：

- 文法上的錯誤：包括名詞單複數（有沒有加s、es的差異）、動詞時態（過去式寫成未來式）等問題。例如原單字為“civilizations”，而受測者聽寫成“civilization”，漏寫了s。這個層次的錯誤，代表受測者能夠掌握整個字的音義，不過有些細微處沒注意到。
- 拼字上的錯誤：例如原單字為“bridges”，而受測者聽寫成“reading”，差異太大，表示受測者對整個字可能沒聽清楚。
- 拼音上的錯誤：例如原單字為“human”，而受測者聽寫成“homan”。這可能有二種情況。其中一種可能，受測者對整個字的發音結構，能夠掌握，但對於“hu”這個音節比較不能分辨清楚。另一種可能，受測者對整個字的發音結構，已經聽清楚了，但不會拼而拼錯了。

這三種層次的錯誤，若對應到前面五種聽寫錯誤的原因，則：文法上的錯誤，涵蓋了第四種、第五種原因（筆誤，以及受測者沒有完全掌握整句話的結構）；而拼字上的錯誤，涵蓋了第一跟第二種原因（聽不清楚發音，或者不確定單字的涵義）；

拼音上的錯誤，則涵蓋了第二種及第三種原因（雖然有聽清楚，但是判斷錯誤）。

因此，我們可以按照這三個層次，依序來分析受測者的聽寫錯誤。在與標準答案比對時，我們先從整個句子，分析文法上的錯誤，接著再從

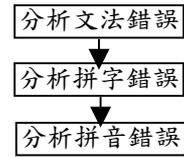


圖 1. 聽寫錯誤分析之流程

每個單字上，逐一比對拼字上的錯誤，最後從單字的音節上，找出拼音上的錯誤。整個流程如圖 1 所示。下面就各層次，分別說明。

3.1. 分析文法錯誤

在分析文法錯誤的階段，其重點在於：受測者的作答中，有些單字僅僅因為單複數不一致、或者時態不同，而與標準答案沒有完全契合。我們希望能把這些意義相同，但文法形式不一致的單字找出來。

當我們把受測者的作答，與標準答案相比對時，首先遇到的問題是，受測者答案的字數，往往不會與標準答案的字數相同。大多數情況下，受測者難以一下子把整句話全部記下來，所以受測者答案的字數，通常會比較少。但即使字數相同，受測者聽寫的順序，也不一定跟標準答案一致。有些人作聽寫測驗時，會先把自己聽懂的部分寫下來，而不會按照先後順序來寫。因此，在比對的過程中，為了方便後續的錯誤分析，我們必須將受測者答案中，與標準答案相契合的單字，先定錨下來。

標準答案: Bridges are really interesting things.

受測者作答: These are very interesting things.

圖 2. 標準答案與受測者答案的對應關係

舉例來說，假如標準答案是“Bridges are really interesting things.”，而受測者作答為“These are very interesting things.”，兩相比對之後，可以發現，受測者答案中的“are”跟“interesting things”，與標準答案完全一致，如圖 2 所示。我們把二者相契合的部分，也就是“are”跟“interesting things”，經過定錨之後，標準答案與受測者答案中，不相契合處的對應關係就突顯出來了。在此例中，標準答案中的“Bridges”，跟受測者答案中的“These”，似乎有著對應關係，雖然這二個字差異很大，但由受測者的聽寫順序來看，“These”應該是把“Bridges”聽錯的結果。同樣的，(really, very)應該也是類似的對應關係。

此處的演算法，如圖 3 所示。首先，輸入是二串單字序列，分別是標準答案與受測者作答。而結果則會輸出這二串單字序列中，不契合部分的對應關係 C。以前一段所舉的例子來說，輸入 S = (Bridges, are, really, interesting, things)，輸入 A = (These, are, very, interesting, things)，我們希望結果能輸出 C = {(Bridges; These), (really; very)}。為了找出不契合的部分，我們先比對 S 跟 A 二串序列

Input: 受測者答案 $S=s_1s_2\dots s_m$ 。
標準答案 $A=a_1a_2\dots a_n$ 。
 s_i 表 S 中的各別單字, a_j 表 A 中的各別單字。

Output: 不契合處的對應關係 $C=\{c_1, c_2, c_3, \dots\}$,
其中 $c_k=(s_w, \dots, s_x; a_y, \dots, a_z)$,
 $1 \leq w \leq x \leq m, 1 \leq y \leq z \leq n$ 。

J_k^S 表在 S 中第 k 個定錨點,

J_k^A 表在 A 中第 k 個定錨點。

Step 0: $J_0^S \leftarrow 0; J_0^A \leftarrow 0; k \leftarrow 1$

Step 1: for $i \leftarrow 0$ to m do
for $j \leftarrow 0$ to n do
if $s_i = a_j$ then begin
 $J_k^S \leftarrow i; J_k^A \leftarrow j; k \leftarrow k+1$
end

Step 2: $J_k^S \leftarrow m; J_k^A \leftarrow n$

Step 3: for $i \leftarrow 1$ to k do begin
 $w \leftarrow J_{i-1}^S+1; x \leftarrow J_{i-1}^S-1$
 $y \leftarrow J_{i-1}^A+1; z \leftarrow J_{i-1}^A-1$
 $c_i \leftarrow (s_w\dots s_x, a_y\dots a_z)$
end

圖 3. 找不契合處之對應關係的演算法

中, 哪些單字是一致的。若 S 跟 A 二串序列中, 有單字一致的話, 我們就作標記, 把此處定錨下來。在圖 3 的演算法中, 定錨點 J_k^S, J_k^A , 表示在 S 跟 A 中找到第 k 個相同單字時, 該單字在序列 S 跟 A 中的位置。在步驟 1 的時候, 我們逐一比對 S 跟 A 中的每個單字, 假如任二個單字相同的話, 就增設一個定錨點, 把該單字在 S 跟 A 中的位置, 記錄下來。而在步驟 3 的時候, 我們把每個定錨點之間不契合的單字, 逐一加入 C 中。由於句首、句尾, 通常也會有不契合的地方, 為了避免遺漏這些單字, 在步驟 0 跟步驟 2 的時候, 我們分別在句首跟句尾設一個定錨點, 這樣到了步驟 3, 才不會把開頭跟結尾的不契合處給遺漏了。

利用圖 3 的演算法, 我們可以把標準答案與受測者答案之間的契合處、不契合處, 分別都找出來。一旦我們把這些對應關係篩檢出來後, 我們就可以針對這些對應關係, 去做進一步的錯誤分析。

為了找出受測者在文法上所犯的錯誤, 我們把受測者的答案與標準答案, 用圖 3 的演算法, 先後比對二次。其中, 在第二次比對前, 我們把標準答案跟受測者的作答, 分別作 stemming 的處理, 把每個單字的變化形式 (如過去式、未來式、複數格), 轉回原本的詞幹。例如說, 把“lie”的現在式“lying”轉回原本的“lie”。在此, 我們用 Porter 的 Stemmer 來處理 [12]。

我們比較前後二次的結果; 第一次沒有作 stemming 的處理, 第二次則經過了 stemming 的處理。兩相比較之後, 假如第二次的比對結果, 比第一次多找出了一些相契合的單字, 就表示在受測者的答案中, 這些單字有著單複數或是時態的錯誤。

3.2. 分析拼字錯誤

在前面分析文法錯誤的階段, 受測者答案中, 哪些單字與標準答案不契合的, 我們已經把它過濾出來了。而接下來, 在分析拼字錯誤的階段, 我們希望能決定, 這些單字中, 哪些字的發音, 受測者

大致上已經能夠掌握清楚, 而哪些字的發音, 受測者仍無法聽清楚。

在此, 我們假設: 假如受測者對於這個單字的發音結構, 大致上已經有所掌握的話, 那麼受測者作答的時候, 儘管有些字母沒有拼對, 但該單字大部分, 應該會與標準答案中相對應的單字一致。相反地, 假如受測者對於這個單字的發音, 還聽不清楚的話, 那受測者所拼的單字, 必然會與標準答案的單字, 差異很大。

根據這個假設, 我們可以從每組單字中, 拼字的差異程度, 來猜測受測者對於這個單字, 是否聽清楚了。

我們用二個方法來分析拼字的差異程度: 最大相同字母數 (maximum common characters, 以下簡稱 MCC), 以及 minimum edit distance (以下簡稱 MED)。此處, MCC 是指二個單字之間, 最大的相同字母數。而 MED 則是指二個單字之間, 最短的編輯距離, 或者說, 從前一個字, 變成後一個字, 最少的修改動作 [9]。

如 (interwoven, interwoved) 這組字, 前面是標準答案的單字, 後面是受測者的作答。其中, 相同的字母為 “i” “n” “t” “e” “r” “w” “o” “v” “e”, 總共有 9 個字母相同, 所以其 MCC 值為 9。

而依照 minimum edit distance 的演算法 [9], 假如二個字母相同的話, 不須修改, 成本值為 0, 而增加或刪減一個字母的成本值為 1, 從這個字母替換成另一個字母, 其成本值則為 2。而在這組單字中, 僅有最後一個字母 “n” 跟 “d” 不同。我們直接將 “d” 替換成 “n” 的話, 需要花費的修改成本, 其值為 2。所以, 這組單字的 MED 值為 2。

以 (with, intercatly) 這組字來說, 相同的字母為 “i” 跟 “t”, 只有 2 個字母相同, 所以其 MCC 值為 2。至於 MED, 假如我們要把 “with” 修改成 “intercatly” 的話, 至少得作 11 次更動: 刪除 “w”、加入 “n”、加入 “e”、加入 “r”、加入 “c”、加入 “a”、加入 “l”、加入 “e”、加入 “l”、加入 “y”、刪除 “h”。所以其 MED 值為 11。

在前一階段中, 我們已經把標準答案跟受測者作答中, 不契合的部分找出來了。現在, 我們把不契合處的每個單字, 任二個為一組, 計算它們的 MCC 值以及 MED 值。

舉例而言, 假如標準答案是 “Because they are intricately interwoven with human civilizations.”, 而學生聽完後, 作答為 “Because they are intercatly interwoved.”。經過第一階段的分析後, 系統會把這二句中不契合的部分找出來, 換言之, 輸出 $C=\{(intricately, interwoven, with, human, civilizations; intercatly, interwoved)\}$ 。

系統會把屬於標準答案的單字 (也就是前面小括號內的五個單字), 與屬於受測者答案的單字 (也就是後面小括號內的二個單字), 任二個為一組, 逐一計算每組單字的 MCC 值與 MED 值。換句話說, 針對這份作答中與標準答案不契合的部分, 系統總共會計算 $5*2=10$ 組的單字。

從這個例子中, 可以觀察到, 即使是不契合的單字, 在聽寫順序上, 多多少少都會與標準答案相隨順。例如, “intercatly” 對應 “intricately”, 而 “interwoved” 對應 “interwoven”。假如不計算所有單

字的組合，而考慮按照單字的順序來篩選組合的話，似乎可以減少一些不必要的計算。例如(human, intercatly) 這組單字，從單字的順序，很明顯的，應該不可能會是相對應的字。然而，我們無法保證，每位受測者的聽寫順序，都會跟標準答案一致。有鑑於此，我們還是選擇計算所有單字的組合。

(標準答案, 受測者答案)	MCC	MED
(intricately, intercatly)	10	2
(interwoven, interwoved)	9	2
(with, intercatly)	2	11
(civilizations, interwoved)	2	10

表 1. 四組單字的 LCS 值與 MED 值

表 1 是這十組單字中，其中四組的 MCC 值與 MED 值。從表中，可以很明顯的發現，當二個字的 MCC 值很大，而 MED 值很小的時候，這二個字在拼字上的差異，比較小。相反地，當二個字的 MCC 值很小，而 MED 值很大的時候，這二個字在拼字上的差異，則比較大。根據這些數值，我們就可以猜測哪幾組單字的發音比較相近。

經過我們一些小型實驗結果，我們發現，用下面二項條件，來篩檢出發音相近的單字，效果非常好：

- 該組單字的 MCC 值，大於標準答案單字長的二分之一；且
- 該組單字的 MED 值，小於標準答案單字的長度。

因此，表 1 的四組單字中，(intricately, intercatly) 跟 (interwoven, interwoved) 會被判為發音相近的單字，而 (with, intercatly) 與 (civilizations, interwoved) 則會被判為發音不相近的單字。而這也合於我們平常的語言經驗。要注意的是，篩檢的結果，並不一定都是一對一的關係，有時候也會有多對一，或一對多的情況出現。目前系統對於這樣的情況，並沒有作特別的處理。

進一步，我們把發音相近的單字，丟到下一階段，以進行拼音上的錯誤分析。而對於發音不相近的單字，我們則認為，受測者對這個單字，可能還沒聽清楚。

3.3. 分析拼音錯誤

在分析拼字錯誤的階段中，我們已經把拼字不完全一致，但發音相近的單字篩檢出來了。而在這一階段中，針對這些發音相近的單字，我們希望能分析出受測者辨音錯誤的地方到底在哪裡。

為了比對二個單字的拼音，我們先把每個單字的音節剖析出來，然後逐一去比對每個子音母音。

在此，我們利用 JLex (lex 的 JAVA 版 [2]) 所做成的字彙分析器 (lexical analyzer, 以下簡稱 lexer) 來協助我們剖析單字的音節。換句話說，要輸入 lexer 的資料是一個單字，而切割出來的 token，其單位則是一個子音或一個母音 (或者稱為「音素」, phoneme [4])。而當系統呼叫 lexer、輸入單字的時候，透過 lexer，我們希望能輸出一

串音素的序列。因此，我們首先要寫規格檔，定義 token 的內容，然後把規格檔餵給 JLex, JLex 就會做出一個 lexer 的函式，之後系統就呼叫這個函式來剖析音節。

這份規格檔中，要定義以下三件事：lexer 能切割出哪些音素；這些音素可以有哪幾種字母的組合；lexer 切出這些音素時，要怎麼處理。其中第二項的定義，關係到系統分析拼音錯誤的準確程度。然而，英文的 26 個字母，雖然有標音的作用，但隨著語言的流變，每個字母不一定只對應到單一音素，而有些字母則會對應到多個音素，例如字母“a”就有“æ”、“ɑ:”、“er”等不同的發音，而有些，則是二個字母對應到一個音素，例如“ch”的發音為“tʃ”。由此可見，要定義字母與音素之間的對應關係，是一件很麻煩的事。

而字母與音素之間的對應關係 (grapheme-phoneme correspondence, 簡稱 GPC), 是語言學中的研究題目之一。我們參考語言學的相關成果，來解決 token 定義的問題。其中，Lange 根據語料庫的資料，對於英文的單音節與雙音節單字，完整的分析了 GPC 的關係 [8]。表 2 列舉了其中幾條 GPC 規則，其中的標音符號，為國際音標 (International Phonetic Alphabet, 簡稱 IPA) [6, 7]。

母音	子音
a → /æ/	b → /b/
a → /ɑ:/	bb → /b/
e → /ɛ/	ch → /tʃ/
ee → /i/	ch → /k/
i → /aɪ/	ph → /f/

表 2. 字母與音素之間的對應規則

我們參考其中部分的規則，寫成了 JLex 的規格檔。在定義好規格檔之後，再利用 JLex 產生出一個可以剖析音節的函式，然後用這個函式 (也就是 lexer) 來剖析每個單字的音節。以“interwoven”這個字為例，當系統要分析這個單字的發音結構時，系統就呼叫 lexer 來處理，而 lexer 會剖析出“i”“n”“t”“er”“w”“o”“v”“e”“n”等 9 個 token。要注意的是，規格檔中的轉換規則，還沒有完全符合於音韻學所說的音韻規律 (the rules of phonology)。我們希望未來能在 lexer 中，加入這方面的判斷。

當我們把單字的音節剖析出來之後，接下來的問題是，我們要如何比較音節的差異？例如說，“la”、“ra”、“da” 這三個音節，就我們主觀感覺來說，“la”跟“ra”的發音比較相近，而“la”跟“da”的發音，差異比較大。但電腦要怎麼去判斷哪些音節比較相近？哪些音節比較不相近？又假如說，標準答案中，有一個單字是“room”，現在有二位受測者，一位聽寫成“loom”，另一位聽寫成“doom”，系統要怎麼去區別這二位受測者的錯誤程度呢？再者，為了要達到適性化測驗的效果，我們希望能根據受測者聽寫的弱點，提供發音相近的練習題目，問題是，哪些是跟受測者弱點發音相近的單字呢？

為了方便比較每個音素發音的差異程度，我們有必要為其下一個量化的定義。我們把這樣量化後的差異程度，稱作「發音差距」(distance among phonemes)。

在本系統中，我們參考語言學的相關知識[4]，來定義每個音素之間的發音差距。

首先，就子音而言，從發音部位來分，有舌根音、舌面音、舌尖音、雙唇音、喉音等五種差別，而從發音方法來分，有塞音、塞擦音、擦音、鼻音等四種差別。所以，我們定義：

- 假如任二個子音，音素一致的話，差距為 0；
- 假如其發音部位跟發音方法都相同的話，差距為 1；
- 假如發音部位或發音方法，其中有一個相同的話，差距為 2；
- 假如發音部位以及發音方法都不同的話，差距為 3。

舉例來說，“ma”、“da”這二個音節，其中 m 是雙唇鼻音，d 是舌尖塞音，二者的發音部位跟發音方法都不同，所以“ma”跟“da”的發音差距為 3。又，“ma”、“na”這二個音節，其中 n 是齒槽鼻音，二者的發音方法相同，都是鼻音，但發音部位不同，所以“ma”跟“na”的發音差距為 2。

至於母音的話，傳統上，則是根據舌頭高度、舌頭位置、嘴唇位置這三個屬性來分類。依照 IPA 所定義的母音符號，從舌頭高度來分，有高、半高、上中、中、下中、低等六種差別，而從舌頭位置來分，有前、中、後等三種差別，而從嘴唇位置來分，則有圓唇、非圓唇等二種差別。類似前面定義子音的方式，我們依據舌頭高度、舌頭位置、嘴唇位置三個因素，定義母音的發音差距如下：

- 假如任二個母音，音素一致，差距為 0；
- 假如三個因素都相同的話，差距為 5；
- 假如如有任一個不同，差距為 10；
- 假如如有二個不同，差距為 15；
- 假如三者都不相同，差距為 20。

舉例來說，“mea”跟“mi”這二個音節，其中“mea”的母音為/i/，其發音屬性為舌頭高、舌位前、非圓唇，而“mi”的母音為/I/，其發音屬性為舌頭半高、舌位前、非圓唇，二者比較，只有一個屬性不同，所以其發音差距為 10。又，“mea”跟“moo”這二個音節，其中“moo”的母音為/u/，其發音屬性為舌頭高、舌位後、圓唇，相較之下，有二個屬性不同，所以其發音差距為 15。

我們定義母音之間的發音差距，很明顯的，比子音之間的發音差距來得大。這是考量到：人類發母音的時候所產生的氣流，相對於發子音來說，比較能自由地逸出口腔或鼻腔，聲量也就因此而比較高。換句話說，在聽寫時，母音的差異，會比子音的差異，來得顯著。為了突顯這項特性，我們增加母音之間的差距，以拉大子音跟母音的差距。

此外，假如一個是子音，另一個是母音的話，二者的發音差距定為 50。這是基於假設說，母音跟子音，二者是不能比較的。因此，我們將二者之間的發音差距，設為 50，以突顯母音跟子音的不可比較性。以圖 5 中的 (r, er) 來說，“r”為子音，“er”為母音，基於子音母音不能比較的假設，系統會把 (r, er) 的發音差距設為 50。

y	9	8	58	107	107	25	75	43	93	32	17
el	8	17	57	57	57	25	75	43	67	17	32
t	7	55	7	7	10	60	28	67	17	67	117
a	6	5	55	71	73	26	67	17	67	82	74
c	5	54	21	23	26	67	17	67	74	74	74
er	4	18	52	50	50	17	59	24	74	24	49
t	3	52	2	0	2	52	9	57	24	74	109
n	2	50	0	2	4	54	7	56	24	74	59
i	1	0	50	52	53	4	54	6	56	23	9
#	0	1	2	3	4	5	6	7	8	9	10
#	i	n	t	r	i	c	a	t	e	l	y

圖 5. (intricately, intercatly) 的發音差距

在把二個字剖析成子音母音的序列之後，我們根據上述的定義，逐一去計算每個音素的發音差距。計算的方法，類似於 minimum edit distance 演算法的概念，只是比對的單位，從單一字母改成單一音素，而加入、刪除、替換的成本值，則為二個音素的發音差距。

舉例來說，我們要比對 (intricately, intercatly) 這組字的發音差距，其中前一個字為標準答案，後一個字為受測者的作答。經過 MED 的計算之後，其矩陣如圖 5 所示。矩陣中，右上角的值為 17，這代表 (intricately, intercatly) 這組字的發音差距。而粗體字的數值，則是整個計算過程中的最小值。

為了找出這組單字中，發音不一致的音素，我們從右上角開始，沿著粗體字的路徑回溯。在回溯時，這二個字後面都是“catly”，差距值都是 17，沒有變化。而在“intercatly”的“er”，跟“intricately”的“i”這二個音素的時候，差距從 17 降為 2，表示這二個音素有差異。換句話說，在回溯的過程中，假如二個音素的差距為 0，表示這二個音素相同；假如差距不為 0，表示音素不一致。我們把差距不為 0 的部分記錄下來。這樣我們就把二個字發音不同的地方找出來了。

4. 適性化之練習回饋

把發音不一致的地方找出來之後，我們希望能從這樣的錯誤訊息中，推測受測者是犯了哪一類型的錯誤，然後從題庫中，隨機選出這一類型的題目，提供受測者練習。

在 [1] 中，列舉了在聽力上容易混淆的幾種發音，我們參考這份資料，將發音的錯誤，分為以下九種類型：

- 子音“l”、“r”容易混淆。
- 子音“f”、“v”、“h”、“b”容易混淆。
- 子音“s”、“z”、“d”、“tj”、“j”容易混淆。
- 母音“i”跟“æ”容易混淆。
- 母音“e”跟“a:”容易混淆。
- 母音“æ”跟“a:”容易混淆。
- 母音“a:”跟“u”容易混淆。
- 母音“i”跟“I”容易混淆。
- 母音“e”跟“æ”容易混淆。

於是，我們依照這九種錯誤類型，蒐集例句，將題庫的題目加以分類。當我們把受測者作答中，發音不一致的地方找出來之後，系統就根據上面九

種的錯誤類型，判斷受測者是犯了哪一類的錯，然後提供適性化的練習。要注意的是，我們在分析拼音錯誤的時候，所找的是發音差距不為0的音素組合，但二個音素發音有差距，並不代表它們容易混淆。所以，對於發音不一致的地方，假如不屬於上述九種類型的話，系統將會忽略不計。然而，容易混淆的發音，並不只這九種，未來我們還要再加入更多的錯誤類型，以加強系統的偵錯能力。

而在分析拼字錯誤的階段，在篩檢過不契合的部分之後，假如屬於標準答案的單字裡，在受測者作答中，並沒有找到發音相近的單字，這顯示受測者對該單字可能沒有聽清楚，因此，系統會搜尋題庫中有該單字的題目，提供給使用者練習。

至於文法上的錯誤，我們則按照單數格、複數格、過去式、現在式、未來式這五種類型，將題庫的題目加以分類。假如受測者的聽寫中，犯了這方面的錯誤，系統就根據標準答案中，名詞的單複數格、或動詞的時態，從題庫挑選類別相對應的題目，供使用者練習。

選題的順序，以聽音混淆的錯誤為優先，其次是沒聽清楚的單字，然後是文法上的錯誤。

5. 結論及未來展望

本論文提出了一個適性化線上英語聽寫測驗系統。受測者作答之後，系統會按文法、拼字、拼音三個層次，來分析受測者答案的錯誤。其中，我們從經過 stemming 處理前後的差異，來過濾出動詞時態、名詞單複數的錯誤。而在分析拼字錯誤的階段，我們計算每組單字的最大相同字母數以及 minimum edit distance，來決定受測者是否大致上已經能夠掌握了這個單字發音的結構。而在分析拼音錯誤的階段，我們定義了子音母音的發音差距。我們先用 lexer，把每個單字剖析成子音母音，然後運用 minimum edit distance 演算法的概念，找出受測者發音錯誤的地方。我們將發音的錯誤，分成九種類型，而文法的錯誤，則分成五種類型。系統分析受測者的作答，猜測受測者犯錯的層次與類型，然後提供相近的練習題目。

本系統目前正持續發展中，未來將著重於以下幾個方向：

- 使文法錯誤的分析更詳細。
- 使字母和音標之間的對應關係更精確。
- 定義更多的錯誤類型。
- 使用者建模 (user modeling)，加強系統適性化的能力。

本系統未來將用於輔助大一英文的課堂教學，希望能藉由學生的參與和回饋，以評估系統的適性化能力。

致謝

本研究承蒙國科會研究計畫 NSC-92-2213-E-004-004 之部分補助。

參考文獻

- [1] 經典傳訊編輯部，“再造英語大革命 聽力”，經典傳訊，7-46 頁，2001。
- [2] E. J. Berk, C. S. Ananian, “JLex: A Lexical Analyzer Generator for JAVA™,”

<http://www.cs.princeton.edu/~appel/modern/java/JLex/>, 2003.

- [3] D. Coniam, “A Preliminary Inquiry into Using Corpus Word Frequency Data in the Automatic Generation of English Cloze Tests,” *CALICO Journal*, No. 2-4, pp. 15-33, 1997.
- [4] V. Fromkin, R. Rodman, *An Introduction to Language*, Thomson Learning, 7th Edition, 2002.
- [5] Z.-M. Gao, C.-L. Liu, “A Web-Based Assessment and Profiling System for College English,” *Proceedings of the 11th International Conference on Computer Assisted Instruction*, 2003.
- [6] International Phonetic Association, <http://www2.arts.gla.ac.uk/IPA/ipa.html>
- [7] International Phonetic Association, *Handbook of the International Phonetic Association: A guide to the use of the International Phonetic Alphabet*, Cambridge University Press, 1999.
- [8] M. Lange, “From orthography to phonology: Nature of grapheme-phoneme conversion processes in written words recognition,” <http://www.psy.unsw.edu.au/Users/mlange/GPC/>, 2000.
- [9] V. I. Levenshtein, “Binary codes capable of correcting deletions, insertions, and reversals,” *Cybernetics and Control Theory*, Vol. 10, No. 8, pp. 707-710, 1966.
- [10] C. McCormack, D. Jones, “Building a Web-Based Education System,” *John Wiley*, 1998.
- [11] L. N. Michaud, K. F. McCoy, L. A. Stark, “Modeling the Acquisition of English: An Intelligent CALL Approach,” *User Modeling 2001*, LNAI 2109, pp. 14-23, 2001.
- [12] M. F. Porter, “An algorithm for suffix stripping,” *Program*, Vol. 14, No. 3, pp. 130-137, 1980.
- [13] C. Roever, “Web-Based Language Testing,” *Language Learning & Technology*, Vol. 5, No. 2, pp. 84-94, 2001. <http://llt.msu.edu/vol5num2/roever/default.html>
- [14] L.M. Rudner, “An on-line, interactive, computer adaptive testing tutorial,” <http://ericae.net/scripts/cat/>, 1998.
- [15] V. Stevens, “Classroom concordancing: Vocabulary materials derived from relevant authentic text,” *English for Specific Purposes Journal*, Vol. 10, pp. 35-36, 1991.
- [16] V. Tsiriga, M. Virvou, “Diagnosing Language Transfer in a Web-based ICALL that Self-Improves its Student Modeler,” *Proceedings of the 2nd IEEE International Conference on Advanced Learning Technologies: Media and the Culture of Learning*, pp. 349-543, 2002.
- [17] E. Wilson, “The Automatic Generation of CALL Exercises from General Corpora,” *Wichman et al. (eds.) Teaching and Language Corpora*, pp. 116-130, Longman, 1997.