字形相似別字之自動校正方法

Automatic Correction for Graphemic Chinese Misspelled Words

張道行　Tao-Hsing Chang
蘇守彥　Shou-Yen Su
國立高雄應用科技大學 資訊工程系
Department of Computer Science and Information Engineering
National Kaohsiung University of Applied Sciences
changth@kuas.edu.tw
shouyen@gmail.com

陳學志　Hsueh-Chih Chen
國立台灣師範大學 教育心理與輔導學系
Department of Educational Psychology and Counseling
National Taiwan Normal University
chcjyh@ntnu.edu.tw

摘要

不論華語為母語或外語的學習，錯別字是相當重要的議題。許多研究對於正在求學階段的學生提出矯正錯別字的建議，以及對教師提出教學正字的策略建議。儘管學生在求學時對錯別字的產生作了許多的防範和矯正，但有時候在撰寫文件時，還是會有錯別字產生而不自覺，因此除了在教學上強調錯別字辨認外，如何在使用文字過程中提示錯別字發生成為重要的問題。利用部件組字與形構資料庫，可以得知字的形體結構和組成的部件元素，探討字形相似性的混淆，進而找出造成錯誤的別字。然而，如何由程式自動又正確地找出文件中的別字並不是容易的事情。現階段在字形的錯別字偵測皆有研究者在各領域進行研究和應用，然而正確率距離實際需要仍有一段距離。若是能仔細分析別字的型態、機率以及發生時的語境，應該能夠更精確且快速的偵測出別字並有效的更正。本文利用 bi-gram 字詞比值、bi-gram 詞性比值和候選詞相似度三種特徵，嘗試利用分類模型：SVM、Neural Network 和線性迴歸法對別字偵查與校正。

Abstract

No matter that learning Chinese as a first or second language, a quite important issue, misspelled words, needs to be addressed. Many studies proposed that there was a suggestion of correcting misspelled words for students who are still schooling as well as a suggestion of teaching and learning strategies of Chinese characters for teachers. Although in schooling, it does to prevent students who do lots of precautions and corrections from generating misspelled words; students sometimes are unconscious of their misspelled words while writing. As a result, in addition to emphasize the recognition of misspelled words in teaching, mentioning how to prevent from generating misspelled words during the process of using words becomes a critical issue. Nevertheless, it is not an easy matter to find misspelled words automatically and correctly within documents by using formula. Currently, there are researchers conducting research on graphemic misspelled words detection and applying it to

different fields. But the accuracy is still far from the real demand. If it can analyze the model, probability and context of misspelled words in detail, it could be detecting the misspelled words more quickly and precisely as well as correcting those words effectively. We had been already accumulated quite research experiences on graphemic misspelled words. This project will combine with resources provided by the mainline project to process the problem of graphemic misspelled words. If it can achieve a breakthrough, it will not only offer a quite effective auxiliary tool for teaching Chinese misspelled words, but assist in establishing a learning tool of Chinese character errors corpus more quickly.

關鍵詞：別字偵測、別字校正、字形相似

Keywords: Misspelled Words Detection, Misspelled Words Correction, Graphemic Similarit

一、緒論

不論華語為母語或外語的學習，避免錯別字的發生都是相當重要的課題。中、小學甚至高中職課程中都會不斷的要求學生認識、更正錯別字，因為長期寫錯別字在一般人的既定印象中認為是語文水準低落，而實際上在國中基本學力測驗和高中學力測驗的中文作文寫作測驗，錯別字也一直是評分重點項目。而在工作文件中出現錯別字，可能會造成讀者對句子誤解，甚至不了解所要表達的意思，徒增事後再溝通的時間，影響整體的工作效率，甚至造成工作單位的損失。由於以上的問題皆是錯別字所引起，所以更需要正視這個問題。

錯別字分為字體不存在的錯字、以及字體本身為正字但使用錯誤的別字。在目前電腦如此普及和網路通訊發達的環境下，文件的產生多由鍵盤以輸入法輸入，因此只會有別字而不會有錯字產生。若系統可以在使用者書寫文章到一個段落後，自動偵測別字和校正，將可以提升文件的整體品質和效率，減少許多事後的訂正或誤會，也可以讓使用者察覺到別字的發生，進而減少再使用同樣別字的機會。然而，如何由程式自動又正確地找出文件中的別字並不是容易的事情。在英文句子中，字詞之間有空白間隔，因此很容易就可以檢測出是否有錯字和未知字；而中文句子是以連續的單字組成，而單字也可能是一個詞，當發生別字時很難確認是正確字或是別字。

早在 1995 年 Chang[1]已提出中文別字自動偵錯技術，雖然能夠自動找出別字，但還是存在需要改善的缺點。例如：False Alert 過多、偵錯時間過長或無法參考前後文…等。Ren 等人[4]使用規則式加上語言模型的混合方法偵測錯誤，雖然其效能並不理想，但也在此領域提出新概念。而 2002 年 Lin[6]針對倉頡輸入法造成別字的情況進行研究，並提出偵錯的系統。Huang 等人[2]對拼音錯誤的別字設計了一套檢測校正系統，對於每個字建立以拼音相似或相近的混淆字集，利用 bi-gram 語言模型找出別字的位置，並以可能性最高的字替代之。之後陸陸續續還有其他人針對不同情況提出偵錯方法，例如洪大弘等人[7]對國中生作文進行別字的偵查，提供了別字的更正建議，並說明別字與正字的資訊和其關係，用來提升學生的學習能力。而陳勇志等人[8]則以前者為基礎，改良了偵錯模板自動產生正反面知識語料庫，利用 Template 和 Translate 模組進行句子校正，最後以 POS Language Model 作最後校正，提升對別字校正的正確率。

一般而言別字類型分為字音相似與字形相似。現階段在字形的錯別字偵測皆有研究者在各領域作過研究和應用，然而正確率距離實際需要仍有一段距離。若是能仔細分析別字的型態、機率以及發生時的語境，計算出正字詞彙的判別參數，利用分類器訓練出高精確度的模型，或許能夠更精確且快速的偵查出別字並有效的更正。本文的目的是提

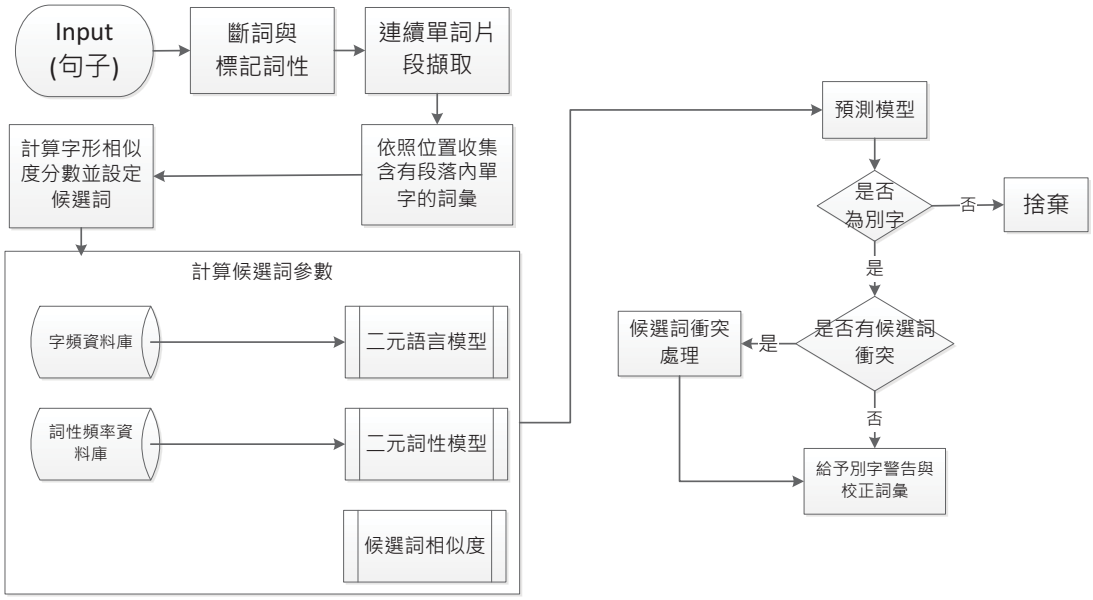
出一套方法能自動且準確的識別句子中是否有別字，且能精準地偵測句子中別字的位置，並校正為正確字。我們利用字形部件資料庫以字形相似方法辨識疑似含有別字的詞彙，結合字形相似度、Bi-gram 字頻機率和 Bi-gram 詞性機率，將這三個判別特徵分別嘗試以三種分類器模型：SVM、Neural Network、Linear Regression 判別，進而找出正字，最後再提供正字與別字在字形的相似差異，以利使用者比較，避免再次犯錯。

二、方法架構與流程

本文利用詞彙集(Lexicon-based)斷詞法的特性偵測別字。假設句子中的詞彙沒有別字，理想的斷詞系統會將句子分割成正確的詞彙組合，若是詞彙中存在別字，系統則會將絕大部分含有別字的詞彙以單字詞的形式斷開。例如：句子「我們都喜歡學校」，經過斷詞方法處理後會被切割成「我們 都 喜歡 學校」四個詞彙；假設撰寫者寫成「我們都喜歡學佼」，經過斷詞方法處理則會被切割成「我們 都 喜歡 學 佼」，因為詞典無法找到「學佼」這個詞彙，所以會將「學」與「佼」分別斷開。利用此特性，可以假設在連續單字詞片段中可能存在含有別字的詞彙，因此對包含兩個以上單字詞的句子片段，可以對每個單字詞逐一從詞典中搜尋出所有含有該單字詞的詞彙，稱為「候選詞」。再利用預測模型判斷被找出的詞彙是否為原句子片段中的正確詞彙。而句子中疑似含有別字的原字串組合稱為「疑似字組」。

在辨識字形易混淆的單字時，可以利用部件組合判斷字體空間結構，找出字形相似度高的混淆字。並將字形所計算的相似度作為候選詞的參數之一。接著，利用候選詞和疑似字組在原本句中的前後字詞分別計算 bi-gram 字頻機率，再計算兩者的比值做為候選詞的第二個參數。最後，以候選詞和疑似字組分別在原句子中前後詞性文法組成的合理性做為候選詞的第三個參數。

訓練資料中每個候選詞的三個參數以及是否真的具有別字的結果，可以用以訓練預測模型。已訓練之模型可以藉由輸入的三個參數值，輸出該疑似字組是否有別字。若預測模型判斷候選詞為疑似字組的正字詞彙，則可對使用者發出警告和校正詞彙，否則就捨棄。若同時有多個候選詞皆被預測模型判斷為疑似字組的正字詞彙，則以候選詞衝突情形來判別兩候選詞的分數，以分數高者為最後校正對象，如圖一流程圖所示。

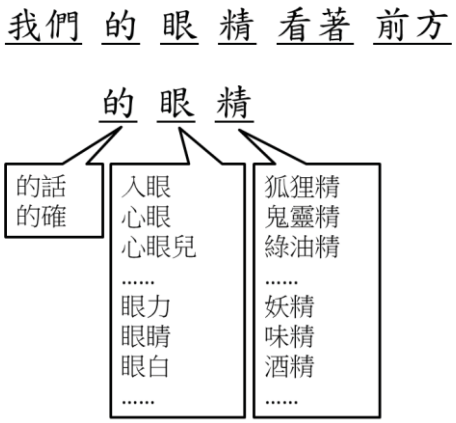


圖一、方法架構與流程圖

三、偵測與校正方法

（一） 偵測可能別字及產生候選詞

所有待處理句子首先經過斷詞及詞性標記處理。由於含有別字的詞彙會造成連續單字詞片段，因此從詞典中找出由片段中任一字所產生的所有詞彙。例如圖二所示，句子「我們的眼精看著前方」經過斷詞處理成「我們_的_眼_精_看著_前方」，其中句子裡「精」為「睛」的別字，因此「眼」和「精」被各自斷開。依照連續單字詞片段「的眼精」，從詞典中搜尋含有「的」為字首且詞彙長度小於等於三的詞彙，例如：「的話」、「的確」；以「眼」為第二字的詞彙且長度小於等於三的詞彙，例如：、「入眼」「心眼」…等，和以「眼」為第一個字且詞彙長度小於等於二的詞彙，例如：「眼力」、「眼睛」…等；以「精」為尾字且詞彙長度小於三的詞彙，例如：「狐狸精」、「綠油精」、「酒精」…等。這些詞彙即為「候選詞」。



圖二、透過斷詞搜尋出可能含有別字的詞彙

（二） 字形相似度

對連續單字詞片段收集候選詞後，計算每一候選詞與所對應疑似字組的字形相似度，並做為後續判斷候選詞是否為正字詞彙的參數之一。

1. 部件和字形結構關係介紹

字形相似比較方法是使用陳學志等人[9]所提供的中文部件組字資料庫.i 拆解字進行比較。漢字的部件是漢字組成的基本單位。資料庫中有 439 個基礎中文部件，而部件的複雜程度也有所不同，筆劃數從 1 劃到最多 17 劃，例如：「一」、「丶」、「丨」「ノ」…「齒」、「龍」、「龜」、「龠」。字形的空間結構歸納出 11 個空間結構關係，例如：垂直組合、水平組合、封閉組合…等。本文比較兩字部件和字形結構的組合，累加其相同部件的筆劃數，計算出兩字之相似度。

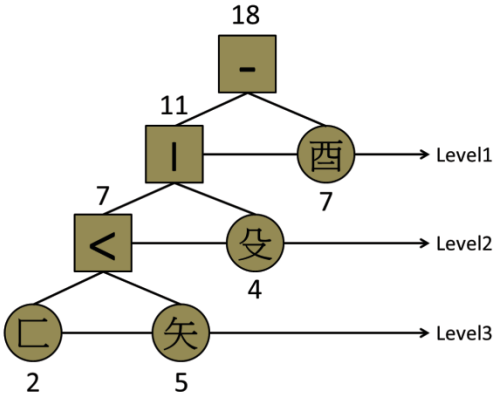
部件分成兩大類，若部件本身是個字則稱為「成字部件」，例如：「女」、「火」；若不是成字則稱為「非成字部件」，例如：「丶」、「丨」。但有些非成字部件無法在 Windows 系統的 Unicode 呈現，所以需要配合結構關係符號，並以中括號 “[]”表示。例如：「即」的右邊非成字部件為「卩」；但左邊的非成字部件是無法呈現，所以以「[|即]」來表示左邊的非成字部件。在 439 個基礎中文部件中，包括 246 個成字部件和 193 個非成字部件。陳學志等人[9]歸納出 11 種不同的結構關係和符號表示，分別為：

(1) 單獨存在(X)：單獨的部件，為字組最基本的組成元素，不可再進行拆解。例如：「木」、「心」。
(2) 垂直組合(-)：其結構關係表示某字的組成型態為部件或部件組以上下垂直相鄰的方式組成。而部件組成順序從左至右，為由上至下的組成方式。例如：「員= -(口,貝)」、「豈= -(山,豆)」。
(3) 水平組合(|)：其結構關係表示某字的組成型態為部件或部件組以左右水平相鄰的方式組成。而部件組成順序從左至右，為由左至右的組成方式。例如：「明= |(日,月)」、「辦= |(辛,力,辛)」。
(4) 封閉組合(0)：其結構關係表示某字的組成型態為部件四面包圍其他部件或部件組的方式組成。其第一個部件表示為包圍的部件，其他為被包圍的部件或部件組。例如：「圍= 0(口,韋)」、「困= 0(口,木)」。
(5) 左上包圍(/)：其結構關係表示某字的組成型態為部件從左方和上方覆蓋其他部件或部件組的方式組成。其第一個部件表示為覆蓋的部件，其他為被覆蓋的部件或部件組。例如：「彥= /(-(文,厂),彡)」、「屏= /(尸,-(丷,一,廾))」。
(6) 右上包圍(\)：其結構關係表示某字的組成型態為部件從右方和上方覆蓋其他部件或部件組的方式組成。其第一個部件表示為覆蓋的部件，其他為被覆蓋的部件或部件組。例如：「或= /(戈,-(口,一))」、「氣= \(气,米)」。
(7) 左下包圍(L)：其結構關係表示某字的組成型態為部件從左方和下方包覆其他部件或部件組的方式組成。其第一個部件表示為包覆的部件，其他為被包覆的部件或部件組。例如：「超= L(走,-(刀,口))」、「近= L(辶,斤)」。
(8) 上方三面包圍( )：其結構關係表示某字的組成型態為部件從左方、右方和上方覆蓋其他部件或部件組的方式組成。其第一個部件表示為覆蓋的部件，其他為被覆蓋的部件或部件組。例如：「同= (冂,-(一,口))」、「咸= (戊,-(一,口))」。
(9) 下方三面包圍(V)：其結構關係表示某字的組成型態為部件從左方、右方和下方包覆其他部件或部件組的方式組成。其第一個部件表示為包覆的部件，其他為被包覆的部件或部件組。例如：「凶= V(凵,乂)」、「幽= V(山,幺,幺)」。
(10) 左方三面包圍(<)：其結構關係表示某字的組成型態為部件從左方、上方和下方包覆其他部件或部件組的方式組成。其第一個部件表示為包覆的部件，其他為被包覆的部件或部件組。例如：「匪= <(匚,非)」、「區= <(匚,-(口,|(口,口)))」。
(11) 左右夾擊(T)：其結構關係表示某字的組成型態為部件位局中央，兩旁為其他部件左右夾擊。其第一個部件表示為位居中央或被夾擊的部件，第二部件為左邊夾擊的部件、第三部件為右邊夾擊的部件。例如：「夾= T（大,人,人）」、「巫= T（工,人,人）」。另一種情況，若左右各有兩個夾擊部件時，其他第二至第五的部件分別表示左上、左下、右上、右下之夾擊部件。例如：「噩= T(王,口,口,口,口)」

2. 部件和字形結構關係介紹

字形可由一連串部件結構表達，每一個部件結構由若干個部件或部件子結構透過一個結構關係連結，而部件子結構也是一個部件結構，例如：「醫」的部件結構為「-(|(<(匚,矢),殳),酉)」，由部件子結構「|(<(匚,矢),殳)」和部件「酉」透過結構關係「-」所連結組成，因此可以將部件結構轉換成樹狀結構的形式進行討論。其中，部件結構的最外層結構關係可以視為樹狀結構的根節點(Root)，例如：「醫」部件結構中最外層的結構關係「-」可視為樹狀結構中的根節點，如圖三所示；部件結構內的部件則表示成樹狀結構的葉結點(Leaf Node)，在此稱為部件節點；部件子結構則視為樹狀結構的子樹(Subtree)，而部件子結構的結構關係則為分支節點(Branch Node)，依照部件和部件子結

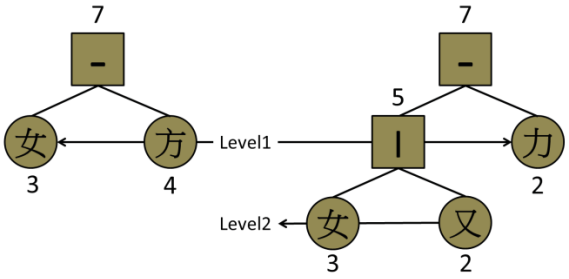
構級數設定為樹狀結構中的階度(Level)，並將部件的筆劃視為葉的權重(Weight)，而分支節點的權重為其子節點向上累加的權重，所以根節點所表示的權重為整個字的筆劃數。例如：「醫」其部件結構為「-(|(<(匚,矢),殳),酉)」，可以發現「醫」是由「|(<(匚,矢),殳))」和 「酉」上下相鄰的的空間結構關係「-」所組成，依照由左至右的順序，以結構關係「-」為根部向下建構左子樹「|(<(匚,矢),殳))」和右子樹「酉」，左子樹以遞迴方式繼續向下建構，右子樹則為一個節點「酉」。部件「酉」在部件結構為第一級部件相對於樹狀結構的階度 1，「殳」第二級部件為樹狀結構的階度 2，「匚」和「矢」第三級部件為樹狀結構的階度 3，如圖三所示。



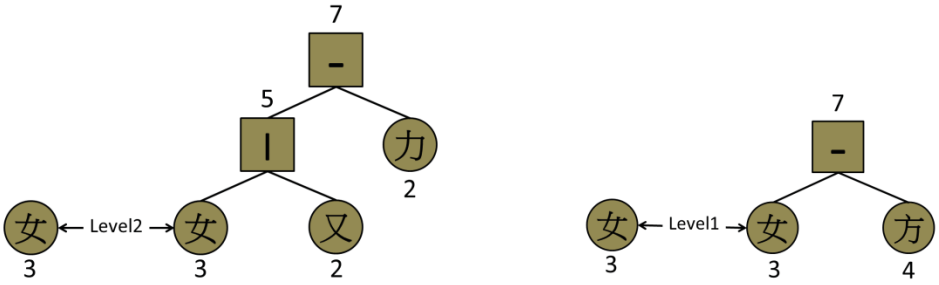
圖三、字形「醫」樹狀結構

藉由上述的樹狀結構，我們可以比較兩個字形的相似度。比較過程中會出現三種型態，分別是單節點和單節點、單節點和樹狀結構、樹狀結構和樹狀結構。第一種型態為單節點和單節點比較，單節點表示為一個部件，而部件為字組成的基本單位，因此兩個單節點比較相似度，若相同則相似度為 1，否則為 0，例如：「女」和「火」兩者皆為成字部件，其樹狀結構皆為單節點，而兩部件不同，因此直接給予相似度 0。

第二種型態為單節點和樹狀結構比較。此型態又分成兩種情形，第一種若單節點不屬於樹狀結構中的節點，則兩者的相似度為 0，例如：「火」和「努」比較，「火」的結構為單節點，而單節點「火」沒有出現在「努」的樹狀結構中，因此直接判別兩者相似度為 0。第二種情形為單節點為樹狀結構中的一個節點，則以單節點的權重值除以樹狀結構根節點的權重值，作為兩者的相似度，例如：「女」和「努」的比較，單節點「女」為「努」的樹狀結構中的一個節點，則以「女」的筆劃數除以「努」的筆劃數為兩者的相似度。但以字形結構觀察，階層值越大之部件，所占整個字體比例越小，例如圖四所示，「妨」和「努」兩者筆劃數相同且樹狀結構皆包含節點「女」，而節點「女」在兩樹狀結構的階層不同，因此在兩字字形上所佔的比例也明顯不同。因此將相似度再乘上一個階層權重值(1-(x-1)*0.05)，x 為單節點在樹狀結構中所在之階層，作為兩者的相似度。如圖五所示，「女」分別與「妨」和「努」比較相似度，「女」的筆劃數為 3，「妨」和「努」的筆劃數皆為 7，節點「女」為「努」的子節點且階層為 2，因此其相似度計算為 3/7*0.95=0.407；節點「女」為「妨」的子節點且階層為 1，因此其相似度計算為 3/7*1=0.429。

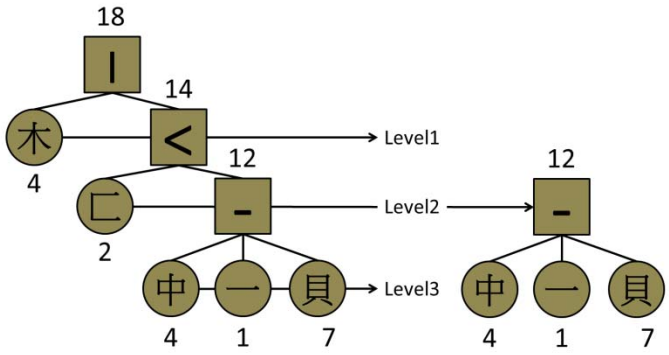


圖四、「妨」和「努」的樹狀結構



圖五、樹狀結構「努」和「妨」與部件「女」比較

第三種型態為樹狀結構和樹狀結構比較，共分下列 3 種情形。第一種情形為其中一方樹狀結構為另一方樹狀結構的子樹，則以子樹的根節點權重除以父樹的根節點權重，乘上階層權重值作為兩者的相似度。如圖六所示：「櫃」和「貴」比較，「貴」的筆劃數為 12，「櫃」的筆劃數為 18，而樹狀結構「貴」為樹狀結構「櫃」的子樹且階層為 2，因此其相似度計算為 12/18*0.95=0.666。



圖六、「櫃」與「貴」相似度比較

第二種情形為兩樹狀結構的根節點相同。兩個根節點相同表示兩字字形屬於相同的結構關係，因此以階層 1 的子樹交叉比較，累加相同部件節點的權重值，以累加的權重值分別除以兩樹根節點權重值並平均作為兩字的相似度。如表一所示，「噪」和「橾」的比較，「噪」階層 1 的子樹為「口」和「桑」;「橾」階層 1 的子樹為「木」和「桑」，交叉比較相同的部件並累加其部件節點權重，選取最大權重值的組合「木」跟「口」比和「桑」和「桑」比為，其累加樹重為 0+13=13，分別除上兩樹根節點的權重並平均為((13/16)+(13/16))/2=0.8125。

子樹在比對到相同的部件節點，若此部件節點的父節點不同，表示其部件的結構關係組成不同，則不累加其部件結點權重值。以表二為例，「估」和「伽」比較，「估」階

層 1 的子樹為「亻」和「古」,「伽」階層 1 的子樹為「亻」和「加」,其中子樹「古」和子樹「加」比較,發現有相的部件「口」,但其父節點不同,「加」的結構關係為「|」,「古」的結構關係「-」,因此不累加部件「口」的筆劃數,因此兩字相似度為((2/7)+(2/7))/2=0.2857。

表一、「橾」和「噪」階層 1 子樹比較表

噪 / 橾	口	桑
木	0	4
桑	3	13

表二、「估」和「伽」階層 1 子樹比較表

估 / 伽	亻	古
亻	2	0
加	0	0

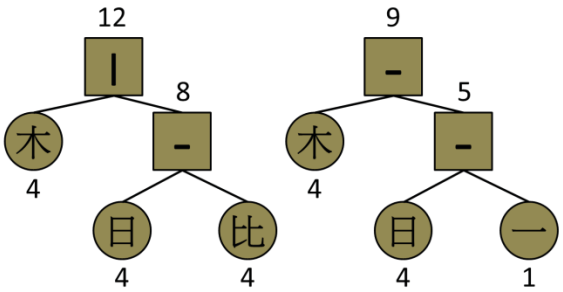
我們發現筆劃數較少的字和筆劃數較多的字在比較時,相同部件權重值固定下,除以字筆劃數較少的筆劃數,分數會較高,平均後會影響整體相似度。如表三所示,「叭」和「嘿」比較,其相同部件權重值為 3,以原來的計算方式,其相似度為((3/5)+(3/15))/2=0.4。我們期望遇到此類形時,相似度不應該這麼高,因此則另外計算其相似度。以 6097 字的筆劃數由少至多排列,設定前 30%屬於筆畫數較少的一類,即筆劃數低於 10 的字。判斷兩樹狀結構中任一方的根節點權重小於 10,則以相同部件權重值除以兩者間較大的根節點權重值為相似度。如表 5 所示,「叭」和「嘿」比較,因為「叭」的筆劃數為 5,屬於筆劃數較少的一類,因此將原本相似度更改為 3/15=0.2。

表三、「叭」和「嘿」階層 1 子樹比較表

叭 / 嘿	口	八
口	3	0
黑	0	0

第三種情形為兩樹狀結構的根節點不相同。從部件資料庫中 6097 字來看,可以觀察幾種結構關係是較特殊的字形所有,例如包圍字形結構(0)共 28 字,大部份以「口」部件為包圍型態,例如:「國」、「圓」;下方三面包圍字形結構(V)共 4 字,分別為「凶」、「函」、「幽」、「匭」,其包圍部件為「凵」和「山」;左方三面包圍字形結構(<)共 15 字,階為「匚」部件所包圍;左右夾擊字形結構(T)共 35 字,例如:「衝」、「夾」。上述四種字形結構較為特殊,當兩樹比較相似度時,若其中一樹根節點為上述其中之一,則另一樹根節點必須是相同,才可利用第二情形進行比較,否則不比較並設定相似度為 0。

觀察垂直組合結構關係(-)和水平組合結構關係(|),這兩種字形結構上較有衝突如圖七所示,「棍」和「查」兩字擁有相同部件「木」、「日」,且樹狀結構也相似,但觀察兩字的字形結構較不相似。



圖七、「棍」和「查」樹狀結構比較

因此兩字的字形結構為垂直組合和水平組合時，則利用第二種情形計算完相似度後再乘上 0.8 最為最後的相似度，如表四所示，「棍」和「查」比較，其相同部件相似度為 8，其中「查」的筆劃數為 9，屬於筆劃數較少的一類，而兩樹的根節點又分別屬於垂直組合和水平組合，因此相似度為(8/12)*0.8=0.5333。排除上述幾個例子，兩字根節點不相同的情況下，則以計算字形相似度後，乘上 0.9 作為最後的相似度。

表四、「棍」和「查」階層 1 子樹比較表

查 / 棍	木	旦
木	4	0
昆	0	4

（三） bi-gram 字頻機率

在語料中可以觀察到某一些字會與另外特定的字頻繁的相鄰出現，稱為共現關係，而共現的次數稱為共現頻率，利用共現頻率計算出現機率稱為共現機率。若句子中含有別字，可以猜測別字與前後字的共現頻率低於正字與前後字的共現頻率，因為語料庫中所收集的資料為一般寫作常使用的正字組合。因此若候選詞的字與前後字的共現頻率比疑似字組高時，可以合理的懷疑句子中出現別字。本文利用 bi-gram 字頻機率計算字與字之間的共現機率，以候選詞的共現機率和疑似字組的共現機率比較，經過正規化計算，設定為候選詞的參數。本文整理聯合報 2003 全年度的報紙內容，分析後記錄 7205 個單字所出現的頻率和兩個單字相鄰出現的頻率。bi-gram 字頻機率為連續條件機率公式，計算字與字的共現機率。假設由 n 個字元組成字串 S: $X_1X_2X_3 \dots X_{n-1}X_n$，其 bi-gram 字頻機率值由公式(1)所示。

$$P(S) = \prod_{i}^{n} \frac{f(X_{i+1}|X_i)}{f(X_i)} \tag{1}$$

其中$P(S)$表示在字串中字元的連續機率；$f(X_{i+1}|X_i)$表示字元X_i、X_{i+1}連續出現的頻率；$f(X_i)$表示X_i單獨出現的頻率。分別計算完候選詞和疑似字組與句子中前後字的 bi-gram 字頻機率，將候選詞機率值除以兩者相加之值做為候選詞的第二個參數，稱為 bi-gram 字頻機率比值，。

（四） bi-gram 詞性機率

當別字取代正字時，句子中的詞性組合必會有所改變，因為正確句子其標記的詞性應該較合乎語法，因此正確句子的詞性組合的共現機率會比含有別字句子的詞性組合的共現機率來得高。因此本文利用 bi-gram 詞性機率來判別別字的依據之一。

將原句子和候選詞替換疑似字組的句子作斷詞處理和標記詞性，利用連續條件機率公式計算詞性與詞性間的共發機率，我們稱為 bi-gram 詞性機率。假設連續 n 個詞性組合 A:$Y_1Y_2Y_3 \ \ldots Y_{n-1}Y_n$，bi-gram 詞性機率如公式 2 所示。

$$P(A) = \prod_{i=1}^{n} \frac{f(Y_{i+1}|Y_i)}{sqrt(f(Y_i) \times f(Y_{i+1}))} \tag{2}$$

其中$P(A)$表示連續 n 個詞性組合的機率；$f(Y_{i+1}|Y_i)$表示詞性Y_i 、Y_{i+1}連續出現的頻率；$f(Y_i)$表示Y_i單獨出現的頻率。含有別字的句子會比正確句子多出一個以上的詞彙，因此詞性數量也會較正確句子多。以圖八為例，句子「晚上在工司吃」，經過斷詞和詞性標記處理為「晚上(Nd)_在(P)_工(Na)_司(Nb)_吃(Vc)」五個詞彙和詞性組合；而由候選詞「公司」替換疑似字組，經過斷詞和詞性標記處理為「晚上(Nd)_在(P)_公司(Nc)_吃(Vc)」四個詞彙及詞性。

詞彙	晚上	在	公司		吃
詞性	Nd	P	Nc		Vc
詞彙	晚上	在	工	司	吃
詞性	Nd	P	Na	Nb	Vc

圖八、正確句子和含有別字的句子其斷詞和詞性組合

因為兩句子的詞性數量不同，無法皆以公式 2 直接計算比較。我們對含有別字的句子其計算公式稍作修正，假設正確句子的詞性數為 n 個，而含有別字的句子詞性數為 n+k 個，k 為疑似字組比原來句子多出的詞性數量，而疑似字組拆解的位置為 x，將位置 x 之前的詞性組合設為 A1，位置 x+k 之後的詞性組合設為 A2，則含有疑似字組的句子其 bi-gram 詞性機率如公式 3 所示。

$$P(A1)P(A2) = \prod_{i=1}^{x} \frac{f(Y_{i+1}|Y_i)}{sqrt(f(Y_i) \times f(Y_{i+1}))} \times \prod_{i=x+k}^{n+k} \frac{f(Y_{i+1}|Y_i)}{sqrt(f(Y_i) \times f(Y_{i+1}))} \tag{3}$$

我們另外計算「詞性共現強度權重值」用以調整公式(3)。首先對於 36 個詞性共現機率作排序，可以發現以「Df-Vh」的共現機率為最高：P(Vh|Df)=0.721。在詞性共現機率排序中間的組合為「De-Ve」，其共現機率為 P(De|Ve)=0.01206。假設 P(X)為疑似字組內部的詞性共現機率，其權重值計算方式為公式 4 所示。

$$W = (1 + (\frac{P(X)}{(P(X) + 0.01206)}) \quad (4)$$

對一個疑似字組而言，其 bi-gram 詞性機率為公式(3)之值乘以公式(4)之值。將候選詞機率值除以兩者相加之值做為候選詞的第三個參數，稱為 bi-gram 詞性機率比值。

（五） 候選詞衝突

候選詞可以分成兩類，第一類為候選詞是疑似字組的替換對象，我們稱為應替換詞；另一類為候選詞不是疑似字組的替換對象，我們稱為不替換詞。有時一個疑似字組會同時出現多個候選詞都被視為應替換詞，因此我們以下列規則解決這個問題。當候選詞長度為相同時，則以三個參數總和比較；當候選詞長度為不同時，則比較候選詞與疑似字組的相同字數，若相同字數一樣則比較候選詞的相似度，若前兩者皆相同則比較兩者字數。

四、預測模型

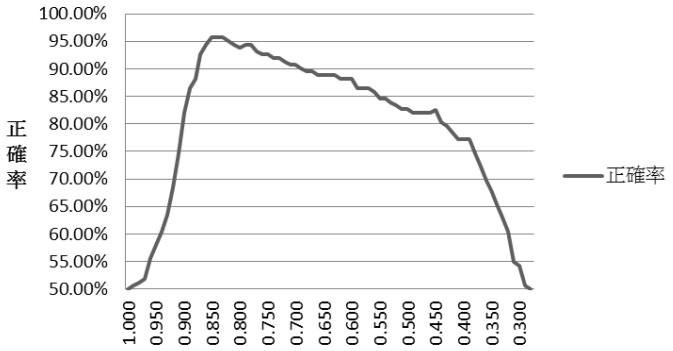
將每個候選詞經過運算可以得到字形相似度、bi-gram 字頻機率比值和 bi-gram 詞性機率比值。當候選詞字形越像、字頻機率越高、詞性機率越高，則為應替換詞機率也越高。本文利用訓練資料中的三個參數訓練幾種監督式預測模型，並在第五節的實驗比較其預測應替換詞的正確性。

(一) 線性回歸法

候選詞的三個判別參數可以作為線性迴歸方程的三個參數項，利用已知的資料計算出迴歸系數β_i，如公式 5 所示。

$$y = \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 \quad (5)$$

對一個測試資料，其 y 值越大越可能為應替換詞，因此我們對 y 須設一閾值，若 y 高於閾值則判別為應替換詞， 反之則判別為不替換詞。圖九為應替換與不替換資料各半組成之訓練資料對不同閾值的預測正確率。在閾值設定為 1 時，全部資料皆判別為不替換詞，因此正確率為 50%。隨著閾值向下調整，正確率會漸漸上升，達到一個高峰後，正確率就開始下降。因此可以利用閾值對正確率的變化，取正確率最高的閾值為設定值。



圖九、門檻值對於訓練資料正確率變化

(二) SVM 與 NN

本文也嘗試以支援向量機(Support Vector Machine)和類神經網路(Neural Network)模型預測應替換詞，以比較線性與非線性預測模型是否有差異。類神經網路使用倒傳遞類神經網路(Back Propagation Neural Network)演算法，而支援向量機所使用的核心函數為 RBF。

五、實驗

（一） 實驗環境

本實驗所採用訓練資料和測試資料是由「國立台灣師範大學心理與教育測驗研究發展中心」所提供九年級 1187 位學生根據題目為「用餐時刻」所撰寫之寫作所產生。每篇寫作經由二至三位受過訓練的閱卷者評分，評分等級為 1 至 6 分。因為分數較低的文章通常誤用別字的情形會比分數高的文章來得多，故選取 1 至 3 分共 258 篇文章，合計 6015 句，以人工檢查含有別字 81 個。產生應替換詞數量有 81 筆。由表五可知，字形相似度低於 0.625 即皆為不替換詞，故僅保留高於此值之不替換詞 647 筆。應替換詞與不替換詞比例約 1：8。

表五、各字形相似度區間之應替換詞和不替換詞數量

相似度 / 類別	1~0.9	0.9~0.8	0.8~0.7	0.7~0.6	總數
應替換詞	10 (12.35%)	49 (60.49%)	20 (23.69%)	2 (2.47%)	81
不替換詞	1 (0.15%)	20 (3.09%)	106 (16.38%)	520 (80.37%)	647

由於本文在預測模型需要訓練資料，然資料內容中應替換詞和不替換詞的數量懸殊，可能導致訓練時產生過度學習的情形，因此不替換詞資料透過隨機的方式擷取與應替換詞相同數量作為訓練資料。

（二） 實驗結果與討論

本實驗以 4-fold cross validation 檢驗各模型的精確度。將不替換詞 81 筆與應替換詞 81 筆隨機分成 4 個資料集，其中，資料集二、資料集三、資料集四，兩類詞各為 20 筆，共 40 筆，資料集一兩類詞各為 21 筆，共 42 筆。

本文使用預測正確率、recall rate 和 precision rate 三個指標評估三種預測模型的效能。假設 *Tr* 為可能出現別字而確實有別字的句子數；*Tf* 為可能出現別字但沒有別字的句子數。三項評估指標計算公式如下:

$$\text{預測正確率} = \frac{Pr+Nr}{Pr+Pw+Nr+Nw} \text{ ; Recall} = \frac{Pr}{Pr+Pw} \text{ ; Precision} = \frac{Pr}{Pr+Nw}$$

其中 *Pr* 為 *Tr* 中被正確校正別字的句子數；*Pw* 為 *Tr* 中被預測不須校正的句子數；*Nr* 為 *Tf* 中被預測不須校正的句子數；*Nw* 為 *Tf* 中被錯誤校正的句子數。

1. 使用 SVM 預測

本實驗以 LibSVM[3]為本實驗的訓練和測試工具。依照 LibSVM 所預設的參數對訓練資料進行訓練，模型對於測試資料的判別正確率不高，因此必須找出最佳的參數重新訓練模型。因為無法直接得知最佳的參數值，只能以訓練的方式找出最佳參數設定。利用 LibSVM 裡的 grid 工具找出最佳參數值：kernel 為 RBF、cost 設定為 32、gamma 設定為 0.5，依此參數進行交叉驗證，如表六所示。由實驗結果可知，平均預測正確率為 95.63%、recall rate 平均為 97.50%、precision rate 平均為 94.10%。

表六、SVM 預測結果

訓練集	測試集	測試筆數	正確預測數	預測正確率	Recall Rate	Precision Rate
234	1	42	42	100%	100%	100%
134	2	40	37	92.50%	95.00%	90.48%
124	3	40	38	95.00%	100%	90.90%
123	4	40	38	95.00%	95.00%	95.00%
平均				95.63%	97.50%	94.10%

2. 使用 Neural Network 預測

本文使用 Qnet2000 設定 Neural Network 並運算結果。網路結構設定經最佳化測試為 3 層，包含輸入層 3 個運算元、隱藏層設定為 10 個運算元，最後的輸出層為 1 個運算元。轉換函數設定為一般倒傳遞網路所使用的雙曲函數(Sigmoid function)，學習率(Learn Rate)設定為 0.01，動量(Momentum)設定為 0.8，最大訓練次數(Max Iterations)設定為 10000 次。表七為 Neural Network 的預測結果。字形判別平均正確率為 96.25%、recall rate 平均為 96.25%、precision rate 平均為 96.37%。

表七、Neural Network 的預測結果

訓練集	測試集	測試筆數	正確預測數	預測正確率	Recall Rate	Precision Rate
234	1	42	42	100%	100%	100%
134	2	40	39	97.50%	95.00%	100%
124	3	40	37	92.50%	95.00%	90.48%
123	4	40	38	95.00%	95.00%	95.00%
平均				96.25%	96.25%	96.37%

3. 使用線性迴歸法預測

表八表示交叉驗證的各訓練資料集透過線性迴歸找出的參數權重值和閾值。表八顯示字形相似度和 bi-gram 字頻機率比值的權重值較高，bi-gram 詞性機率比值之權重值較低。表九表示以表八閾值進行預測之結果。平均預測正確率為 97.50%、recall rate 平均

為 96.25%、precision rate 平均為 98.75%。

表八、不同訓練資料集產生之迴歸參數權重值和閾值

訓練集	bi-gram 字頻機率比值權重值	bi-gram 詞性機率比值權重值	候選詞相似度權重值	最佳閾值
234	0.476108	0.047314	0.495112	0.82
134	0.533153	0.041051	0.426089	0.83
124	0.515488	0.065720	0.446411	0.85
123	0.527719	0.038126	0.446539	0.85

表九、線性迴歸法預測結果

訓練集	測試集	測試筆數	正確預測數	預測正確率	Recall Rate	Precision Rate
234	1	42	42	100%	100%	100%
134	2	40	39	97.50%	95.00%	100%
124	3	40	38	95.00%	95.00%	95.00%
123	4	40	39	97.50%	95.00%	100%
平均				97.50%	96.25%	98.75%

4. 結果討論

本實驗透過 SVM、Neural Network 和以線性迴歸三種預測方法，以交叉驗證的方式對資料集進行預測。透過 SVM 平均預測正確率為 95.63%，透過 Neural Network 平均預測正確率為 96.25%，透過線性迴歸法平均預測正確率為 97.5%，如表十所示。

表十、各模型的評估指標平均值

模型	預測正確率	Recall Rate	Precision Rate
SVM	95.63%	97.50%	94.10%
NN	96.25%	96.25%	96.37%
LR	97.50%	96.25%	98.75%

實驗結果顯示預測正確率和 precision rate 最高者為線性迴歸法，recall rate 最高者為 SVM。其中，線性迴歸法其 precision rate 較其他模型來得高，表示找到的候選詞較正確，不容易有錯誤警告產生。若在真實應用環境下，應替換詞和不替換詞的比例為 1:8，因此實驗 precision rate 的些微差異會使得在真實環境下使用者的感受有相當明顯的不同，因此線性迴歸法應較適合在真實環境下使用。

六、結論

本文提出一個偵測字形相似別字及校正的方法，透過陳學志等人[9]部件結構來計算字形間的相似度，利用相似度對別字搜尋候選詞，再計算候選詞的 bi-gram 字頻機率比值、bi-gram 詞性機率比值，最後利用預測模型判斷候選詞是否為應替換詞。在先前

研究中多依靠混淆字集所收集的資料，雖然多為相似度較高且易混淆的對應字，但若別字不屬於混淆字集，則無法偵測校正。本文所提方法能不被混淆字集侷限，此方法若遇到應替換詞相似度低時，可依靠其他參數得到正確預測。經由實驗結果顯示，線性迴歸法的正確率及 precision rate 較其他模型來得高，表示找到的別字較正確，產生錯誤警告的機率也相對較低，因此較符合真實環境下的使用需要。

本研究仍有許多限制待未來進一步研究。首先，本研究只針對二字詞以上的詞彙別字進行探討，對於單字詞的別字無法偵測與校正。第二、對於被偵測之別字是否可能為未知詞並未考慮，若語料中含有大量未知詞則可能造成未知詞的片段被誤認成詞彙，導致預測正確率降低。第三、本研究只針對字形部分探討，字音相似之別字是否可用本文所提架構與模型加以偵測與校正，值得進一步探討。

誌謝

本文作者感謝教育部及國立台灣師範大學「邁向頂尖大學計畫」編號 101J1A0301 以及編號 101J1A0701 計畫支持，同時感謝國立台灣師範大學心理與教育測驗研究發展中心提供語料。

參考文獻

[1] Chao-Huang Chang. A New Approach for Automatic Chinese Spelling Correction. In Proceedings of Natural Language Processing Pacific Rim Symposium’95, Seoul, Korea. pp: 278-283,1995.

[2] Chuen-Min Huang, Mei-Chen Wu, and Ching-Che Chang. Error Detection and Correction Based on Chinese Phonemic Alphabet in Chinese Text. World scientific publishing company, International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, Vol.16, Suppl. 1 pp.: 89-105, 2008.

[3] Chih-Jen Lin, Chih-Chung Chang. LIBSVM -- A Library for Support Vector Machines. Take from http://www.csie.ntu.edu.tw/~cjlin/libsvm/

[4] Fuji Ren, Hongchi Shi, and Qiang Zhou. A hybrid approach to automatic Chinese text checking and error correction, in Proceedings of 2001 IEEE International Conference on Systems, Man, and Cybernetics. pp: 1693-1698,2001.

[5] J.A.K. Suykens and J. Vandewalle. Least Squares Support Vector Machine Classifiers. Neural Processing Letters. Volume 9, Number 3 , 293-300, 1999.

[6] Yih-Jeng Lin, Feng-Long Huang and Ming-Shing Yu. A Chinese Spelling Error Correction System. Proceedings of the Seventh Conference on Artificial Intelligence and Applications, 2002.

[7] 洪大弘，2009，”基於語言模型及正反陙語料知識庫之中文錯別字自動偵錯系統”，朝陽科技大學，碩士論文。

[8] 陳勇志，2010，”利用雜訊通道模型與自動產生偵錯模板改良學生中文作文錯別字偵測與改正”，朝陽科技大學，碩士論文。

[9] 陳學志、張璨勻、邱郁秀、宋曜廷、張國恩，2011，”中文部件組字與形構資料庫之建立及其在識字教學的應用”，教育心理學報。