

ỨNG DỤNG MACRO VBA TRONG THAO TÁC VĂN BẢN TIẾNG VIỆT

✍ PGS.TS. Phan Huy Khánh

Ban Tư vấn Đại học Đông Á

TÓM TẮT

Sử dụng công cụ vĩ lệnh (macro) VBA (Visual Basic for Application) trong hệ thống MSOffice, chúng tôi đã viết các trình tiện ích giúp xây dựng từ điển các cụm từ gõ tắt giải quyết nhanh chóng và hiệu quả nhiều vấn đề đặt ra trong quá trình thao tác các văn bản tiếng Việt, chủ yếu là gõ nhanh gõ tắt các cụm từ tiếng Việt.

Keywords - Mã, chuyên mã, trình tiện ích, vĩ lệnh (macro), xử lý văn bản.

ABSTRACT

Using of the macro system VBA (Visual Basic for Application) in MS-Office, we write the utilities to help build a dictionary of Vietnamese short-cut typing in order to resolve quickly and efficiently many problems in the Vietnamese word processing, mainly fast typing Vietnamese documents.

1. Đặt vấn đề

Soạn thảo văn bản (STVB), hay xử lý văn bản (Word/Text Processing) là một trong những ứng dụng quan trọng trong lĩnh vực CNTT [1][2][3]. Hầu như mọi công việc diễn ra trên máy tính đều có liên quan ít nhiều đến STVB:

- Giao tiếp người – máy thông qua các trình ứng dụng, tiện ích hay duyệt web.

- Lập trình, tạo ra mã nguồn, trong một ngôn ngữ lập trình nào đó.

- STVB để phục vụ cho nhiều mục đích khác nhau: giao lưu giữa các cá nhân trong xã hội, công việc hành chính và văn phòng, lưu trữ và tìm kiếm tài liệu, xây dựng các hệ thống thông tin, phục vụ dịch thuật các ngôn ngữ tự nhiên, các lĩnh vực thuộc trí tuệ nhân tạo, ứng dụng trong các ngành in ấn, chế bản điện tử.

Người ta phân biệt văn bản, hay văn kiện, tài liệu, tư liệu (thuật ngữ tiếng Anh thông dụng là Document) thành nhiều thể loại khác nhau, như: văn bản không định dạng (Non-format Text), văn bản có định dạng, có cấu trúc, văn bản thuần túy chỉ chứa ký tự (Text), hay văn bản không thuần túy, chứa các thành phần đồ họa, bảng biểu, hình vẽ..., văn bản đơn ngữ hay đa ngữ, văn bản thuần nhất (chỉ sử dụng một bộ mã) hay văn bản không thuần nhất (sử dụng nhiều bộ mã).

Có nhiều hình thức tạo ra văn bản: hoặc được soạn thảo trực tiếp kiểu WYSIWYG (What You See Is What You Get) như Winword, NotePad..., soạn thảo gián tiếp để chỉ có thể đưa ra được văn bản cuối cùng nhờ một trình biên dịch hay trình duyệt (Browser) đối với các loại siêu văn bản HTML... hay chuyển đổi văn bản sang các dạng phổ dụng không phụ thuộc vào các hệ thống Tin học như PDF

của Adobe Acrobat.

Có thể hiểu đơn giản mỗi văn bản là một tập hợp các ký tự và được lưu cất trên đĩa dưới dạng một tệp văn bản. Về mặt vật lý, các ký tự của văn bản được tổ chức thành từ (Word), dòng (Line), đoạn (Paragraph) và trang (Page). Tất cả các đơn vị này đều không có độ lớn (số lượng các ký tự) xác định trước và do đó phải quy ước hoặc có các ký tự đặc biệt để đánh dấu. Thông thường, người ta quan niệm mỗi từ là một chuỗi ký tự đặt giữa hai ký tự phân cách như khoảng trống, các dấu chấm câu (phẩy, chấm, hai chấm, chấm phẩy...), mỗi dòng được kết thúc bởi dấu hết dòng là những ký tự đặc biệt.

Về mặt logic, mỗi văn bản đều có cấu trúc phân cấp theo các thành phần khác nhau như chương, mục, đầu văn bản, cuối văn bản, phần tham khảo, chỉ mục, minh hoạ, v.v... Mỗi thành phần có cách trình bày vật lý khác nhau (trên màn hình, trên giấy) tùy theo nội dung, thẩm mỹ và ý đồ người sử dụng (NSD). Để phân biệt cách trình bày các thành phần, cần phải có các dấu định dạng (Markup). Như vậy trong một văn bản, có hai loại ký tự: các ký tự in được (nhìn thấy được) và các ký tự điều khiển định dạng là không nhìn thấy được.

Các thao tác STVB rất đa dạng, tùy theo phần mềm soạn thảo và máy in sử dụng mà cách thức thực hiện có khác nhau. Ngoài các thao tác cơ bản là gõ nội dung văn bản vào máy, chọn phong chữ, định dạng..., những thao tác khác hay gặp là:

- Tạo mới một văn bản theo khuôn mẫu (Template), hay mô hình cho trước nào đó.
- Mở một văn bản đã có trên một hệ thống lưu trữ vào bộ nhớ để tiếp tục soạn thảo.

- Lưu cất văn bản đã/đang cập nhật lên một hệ thống lưu trữ, hoặc với tên cũ, hoặc với một tên khác.

- Đóng một văn bản để kết thúc soạn thảo.
- Chuyển đổi hay trao đổi văn bản...

Tùy theo ngôn ngữ tự nhiên đang được xử lý mà cách thức gõ nội dung văn bản có khác nhau. Việc gõ vào máy một văn bản Anh ngữ thuần túy thật đơn giản vì bàn phím máy tính chỉ toàn các phím tiếng Anh. Tuy nhiên vấn đề trở nên rắc rối khi và chỉ khi thao tác với các văn bản không phải Anh ngữ, như Hán ngữ, Nhật ngữ, Ấn ngữ, tiếng Ả-Rập, v.v...

Chẳng hạn, để có thể xử lý được các chữ Việt có dấu trên những hệ soạn thảo Anh ngữ, cần phải xây dựng các phần mềm, hay bộ gõ, kèm theo và không phải lúc nào cũng có thể dễ dàng in ra được một văn bản tiếng Việt trên bất kỳ một hệ thống Tin học nào. Các bộ gõ chữ Việt có dấu thường dùng một trong các phương pháp như telex, VNI, VIQR.

Hiện nay ở Việt Nam, để sử dụng hệ thống MSOffice, nhiều bộ gõ như ABC, VietKey, VNI, Vietware, BK HCM, VISCII, VIQR..., gần đây là UniKey, với trên 30 bộ mã tiếng Việt đã được xây dựng. Chính sự phong phú bộ gõ và bộ mã này dẫn đến vấn đề rắc rối là phải chuyển đổi văn bản từ một bộ mã này sang một bộ mã khác, nếu không sẽ không thể đọc được. Có một thời, các văn bản được soạn thảo ở phía Bắc thường chỉ sử dụng nhất quán bộ mã ABC, sau đó là TCVN, khi gửi vào phía Nam bắt buộc người ta phải chuyển đổi sang bộ mã VNI, hay Vietware, mới đọc được. Ngược lại, khi gửi ra phía Bắc, người ta lại phải chuyển đổi sang bộ mã ABC!

Từ mấy năm nay, việc sử dụng phổ biến

bộ mã chuẩn quốc tế Unicode đã đem lại rất nhiều lợi ích cho NSD. Tuy nhiên vẫn còn nhiều vấn đề bức xúc, nổi cộm trong quá trình STVB, đặc biệt trong xử lý tiếng Việt, là làm thế nào để tăng tốc độ thao tác, tăng tốc độ gõ vào từ bàn phím các loại văn bản, hay dữ liệu nói chung, làm thế nào để kiểm sửa lỗi chính tả, tóm tắt hay phân tích văn bản, v.v...

Trong bài báo này, chúng tôi trình bày phương pháp lập trình vĩ lệnh VBA (Visual BASIC for Application) xây dựng từ điển các cụm từ gõ tắt, sử dụng các công cụ AutoCorrect và AutoText có sẵn trong MSOffice phối hợp với bộ gõ UniKey, cho phép gõ nhanh, gõ tắt các văn bản tiếng Việt và quản lý được cơ sở dữ liệu gõ tắt. Để đi đến giải pháp, chúng tôi giới thiệu tóm tắt các công cụ MSOffice, bộ gõ UniKey và phương pháp lập trình VBA.

2. Giới thiệu các công cụ xử lý

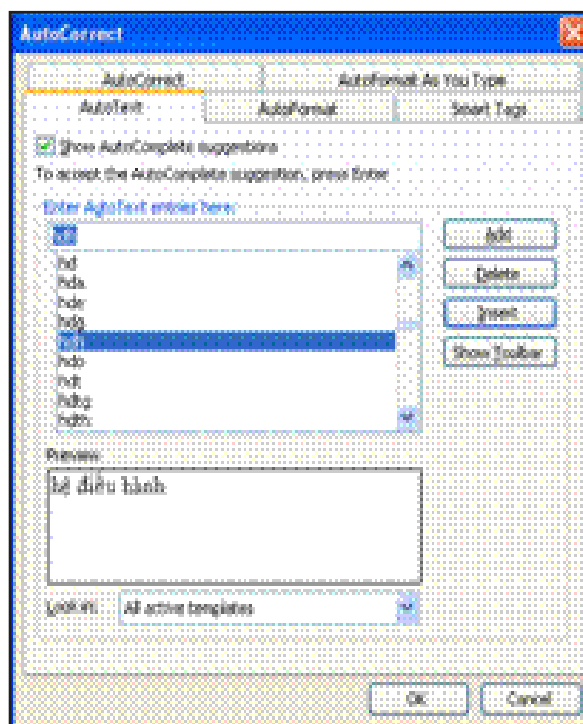
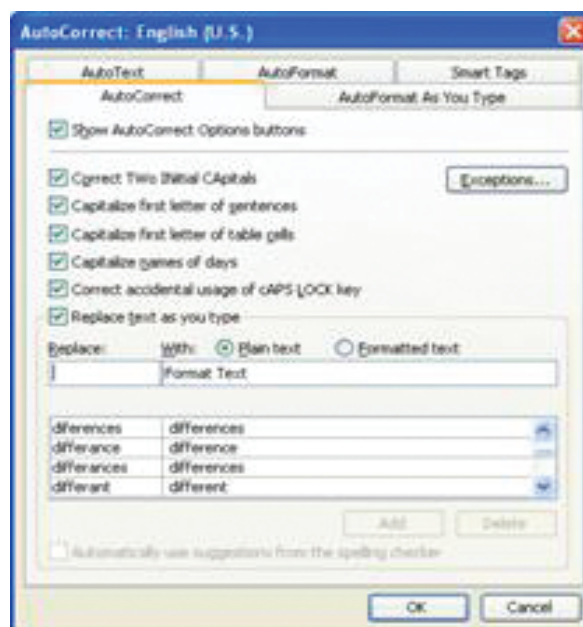
2.1. Các công cụ sửa lỗi văn bản

a. Công cụ tự động sửa lỗi/gõ tắt AutoCorrect

Chức năng chủ yếu của AutoCorrect (xem hình 1) trong MSOffice là dùng sửa các lỗi NSD gõ sai trên bàn phím. Ví dụ nếu NSD gõ từ school thì hệ thống sẽ tự động sửa lại thành school, teh sửa lại thành the... Bằng cách đó, người ta có thể bổ sung vào AutoCorrect các chữ viết tắt tương ứng với các cụm từ thường gặp để hệ thống tự động thay thế khi gõ. Chẳng hạn gõ vn để thay bằng Việt Nam, gõ pl để thay bằng pháp luật...

Cách sử dụng AutoCorrect rất đơn giản. NSD gọi lệnh Tools/Autocorrect Option... để định nghĩa các cụm từ viết tắt tùy ý. NSD có thể dễ dàng sao lưu nội dung AutoCorrect hay có thể chia sẻ cho các máy tính khác, bằng

cách sử dụng tệp MSO1033.acf tại thư mục C:\Documents and Settings\Application Data\Microsoft\Office theo tài khoản người dùng tương ứng.



Hình 1. AutoCorrect và AutoText

Một khi AutoCorrect được định nghĩa, mọi ứng dụng MSOffice như soạn thảo văn bản với Word, sử dụng bảng tính Excel, hay thao tác với PowerPoint đều có thể sử dụng chức năng gõ tắt với AutoCorrect. Một nhược điểm dễ nhận ra khi sử dụng AutoCorrect là việc thay thế ngoài ý muốn các cụm viết tắt. Chẳng hạn, nếu trong từ điển AutoCorrect có các định nghĩa hi=hiều, múr=mức chi=chiều..., thì khi gõ hiều sẽ cho kết quả hiềuều, gõ muộן sẽ cho kết quả múcộן, gõ chiều sẽ cho kết quả chiềuều, v.v... Lúc này, để trả về nội dung cần gõ, NSD phải gõ tổ hợp phím Ctrl-Z (lệnh Edit-Undo AutoCorrect).

b. Công cụ gõ tắt Autotext

Khác với AutoCorrect, chức năng AutoText cho phép NSD định nghĩa trước các từ thông dụng khi STVB. Khi NSD sử dụng chức năng này bằng cách gõ các ký tự đại diện và kích hoạt nhờ phím F3, để nội dung tương ứng sẽ được tự động thay thế vào vị trí hiện hành của dấu chèn (Inert Point) trong văn bản soạn thảo.

Để cập nhật AutoText, NSD gọi lệnh Insert/AutoText để gõ cụm từ định nghĩa cho đoạn văn bản tương ứng. NSD có thể kèm theo phần định dạng bằng cách chọn Format Text, hoặc chỉ lấy phần nội dung mà không cần phần định dạng bằng cách chọn Plain Text. Nhược điểm của AutoText là chỉ sử dụng soạn thảo văn bản với Word, hay nói đúng hơn, Excel và PowerPoint đều không có chức năng này.

c. Chức năng gõ tắt trong các bộ gõ tiếng Việt thông dụng 2

Hầu hết các bộ gõ tiếng Việt thông dụng hiện nay, như ABC, VietWare, VietKey, Uni-Key... đều có chức năng gõ tắt văn bản. NSD

có thể định nghĩa các từ (cụm từ) trong quá trình gõ vào nội dung văn bản. Nguyên tắc hoạt động của những phần mềm này khá đơn giản. NSD cũng có thể quản lý được nội dung của các chữ viết tắt nhờ lưu trữ trong một tệp văn bản gõ tắt. Tuy nhiên nhược điểm của các bộ gõ này là chỉ cho phép lưu giữ một số lượng rất hạn chế các cụm từ viết tắt. Hơn nữa, khi sử dụng, các kết quả thay thế sẽ pha trộn lẫn lộn với AutoCorrect, gây khó khăn cho NSD.

2.2. Tìm hiểu công cụ lập trình macro VBA

VBA và VBScript là hai dạng sử dụng hay gặp khi lập trình với BASIC (viết tắt của Beginner's All-purpose Symbolic Instruction Code), một ngôn ngữ lập trình bậc cao do J. Kemeny và Th. Kurtz đề xuất từ những năm 1960 tại trường Đại học Dartmouth,

Hoa Kỳ [7]. Ngôn ngữ BASIC có tính năng dễ học, dễ lập trình, nên nhanh chóng được cải biên, phát triển và cài đặt thành nhiều phiên bản trên các dòng máy tính khác nhau như TinyBASIC, MicrosoftBASIC, CBASIC, BASICA, GW-BASIC, TrueBASICA... Năm 1988, A. Cooper là người đầu tiên đưa ra ý tưởng phát triển BASIC thành ngôn ngữ lập trình trực quan (Visual Programming Language), gọi tắt là VB [4]. Hiện nay, VB là một trong những ngôn ngữ được sử dụng chủ yếu trong môi trường Windows và internet. VB rất phong phú về câu lệnh và từ khóa. Thư viện VB chứa hàng trăm hàm cho phép NSD phát triển dễ dàng ứng dụng trong các giao diện đồ họa GUI (Graphical User Interface).

Trong MSOffice (Word, Excel, Access, Project, v.v...), người ta sử dụng VBA để tạo ra các vĩ lệnh (macro) tự động hóa [5][6]. Có

thể nói phương pháp lập trình vĩ lệnh đã nâng cao hiệu quả sử dụng của các trình MSOffice.

2.3. Tạo tự động vĩ lệnh VBA trong một ứng dụng MSOffice

Khi soạn thảo văn bản với Winword, hoặc thao tác bảng tính với Excel, hoặc tạo ra các trang trình diễn với PowerPoint..., NSD có thể tạo ra các vĩ lệnh một cách tự động và/hoặc kết hợp với cách lập trình thông thường. Mỗi vĩ lệnh VBA có dạng một chương trình con (sub-program) được khai báo như sau :

Sub macro-name(*arg-list*)

' Dòng chú thích (*comment line*)

...

Các câu lệnh VBA (macro commands/instructions)

...

End Sub 'macro-name

NSD tự do đặt tên của vĩ lệnh để đầu từ khoá Sub (chữ đậm). Mỗi vĩ lệnh có thể có hoặc vắng mặt các tham đối arg-list. Phần thân của vĩ lệnh gồm các dòng lệnh VBA. Kết thúc một vĩ lệnh là cặp từ khoá End Sub. NSD có thể đặt tùy ý các dòng chú thích trong vĩ lệnh. Mỗi dòng chú thích được đặt giữa một dấu nháy đơn ' (quote) và một dấu hết đoạn Enter.

Để tạo mới tự động một vĩ lệnh, NSD gọi lệnh Tool-Macro-Record New Macro... Đặt lại tên vĩ lệnh trong ô Macro Name từ hộp thoại Record Macro. Chọn chế độ lưu cất vĩ lệnh (hộp thoại Store Macro in) rồi OK. Từ lúc này, NSD có thể tiến hành công việc và hệ thống sẽ tự động ghi chép lại bằng các lệnh VBA tương ứng. Sau khi thực hiện xong, NSD

nhấp chuột tại nút Stop Recording (hoặc gọi lệnh Tool-Macro-Stop Recording) để kết thúc ghi vĩ lệnh. Thông thường, NSD tiến hành qua hai bước như sau: bait lure entice

1. Tạo tự động một vĩ lệnh thực hiện một chức năng nào đó có liên quan đến công việc theo cách được gọi vui là tạo “vốn lập trình” (Programming Funds).

2. Lập trình trực tiếp trong cửa sổ soạn thảo của VBA bằng cách gọi lệnh Tool-Macro-Macros... (hoặc tổ hợp phím Alt+F8) để làm xuất hiện hộp thoại Macros. Chọn tên vĩ lệnh rồi gọi Edit, hoặc gọi Create để bắt đầu làm việc với cửa sổ NewMacros (code).

VBA có hệ thống trợ giúp lập trình rất thuận tiện. NSD có thể gõ một dấu chấm sau tên một đối tượng nào đó để chọn một thuộc tính (mục con thừa kế) hoặc gọi lệnh Help (phím F1). Ví dụ để đặt độ dẫn giữa các dòng văn bản là 16, ta thực hiện lần lượt như sau: gõ Selection, dấu chấm, chọn ParagraphFormat, dấu chấm rồi chọn LineSpacing, gõ tiếp giá trị 16, ta nhận được:

Selection.ParagraphFormat.LineSpacing = 16

Ví dụ sau đây tạo ra một hộp đối thoại để gõ vào tên tệp trước khi mở tệp:

Sub EnterFileName()

Dim FileNam **As Variant**

ChangeFileOpenDirectory “C:\MyDocs”

FileNam = InputBox(“Cho biết tên tệp?”,
20)

If Len(FileNam) = 0 **Then**

MsgBox “Tên tệp chưa hợp lý.”

Else

MsgBox “Bắt đầu mở tệp “ & FileNam
Documents.Open FileName:=FileNam,
ReadOnly:=True

End If

End Sub

Cách tạo vĩ lệnh trong Excel hay Power-
Point cũng được thực hiện một cách tương tự.

*a. Xây dựng kho từ điển xử lý gõ nhanh gõ
tắt*

Trước đây, khi STVB, NSD thường sử dụng
các phong chữ (font) tiếng Việt theo thói quen
hoặc theo cấu hình cài đặt sẵn của hệ thống.
Các phong chữ hay gặp là ABC/TCVN, VNI,
VietWare, BK HCM... Khi cần trao đổi văn
bản, NSD có thể sử dụng công cụ trợ giúp
chuyển mã văn bản (Text Conversion) thường
có trong các trình gõ tiếng Việt như UniKey,
Vietkey... để chuyển đổi văn bản sang phong
chữ yêu cầu. Chính vì sự rắc rối về việc chuyển
đổi văn bản không cùng bộ mã tiếng Việt (hay
không cùng phong chữ sử dụng) mà hiện nay,
xu thế sử dụng thống nhất Unicode đã trở nên
phổ biến và từng bước dẫn đến bắt buộc trong
mọi hoàn cảnh.

Để có thể sử dụng lại từ điển gõ tắt văn
bản AutoText, AutoCorect trong MSOffice
đã được xây dựng trong một bộ mã tiếng Việt
thông dụng trước đây sang phong chữ thống
nhất Unicode, chúng tôi viết các vĩ lệnh như
sau :

Các vĩ lệnh sao chép AutoText

Vĩ lệnh CopyAutoText() sao chép nội
dung của từ điển AutoText lên cửa sổ văn bản
AllMyAutoText.doc hiện hành.

Sub CopyAutoText()

‘ CopyAutoText Macro

‘ Macro recorded 16/03/2009 by Phan Huy
Khánh

Windows(“AllMyAutoText.doc”).Activate

N = NormalTemplate.AutoTextEntries.
Count

For I = 1 **To** N

With Selection

.TypeText Text:= NormalTemplate.Auto-
TextEntries.Item(I).Name

.TypeText Text:=vbTab

.TypeText Text:= NormalTemplate.Auto-
TextEntries.Item(I).Value

.TypeParagraph

End With

Next I

ActiveDocument.Save

End Sub ‘ CopyAutoText

Sau khi lưu cất nội dung tệp AllMyAuto-
Text.doc lên hệ thống lưu trữ, có thể sử dụng
công cụ chuyển mã của UniKey để chuyển
nội dung từ điển AutoText sang Unicode. Tiếp
theo là bước tạo mới từ điển AutoText. Vĩ lệnh
dưới đây cho phép xây dựng mới từ điển Au-
toText.

Sub CreateAutotext()

‘ CreateAutotext Macro

‘ Macro recorded 16/03/2009 by Phan Huy
Khánh

Windows(“AllMyAutoText.doc”).Activate

I = 0 : N = NormalTemplate.AutoTextEn-
tries.Count

```

While I < N
    I = I + 1: Selection.HomeKey
Unit:=wdLine
    TxtName = Selection.Paragraphs.First
    TxtName = Left(TxtName, Len(TxtName)
- 1)
With Selection
    .MoveDown: .HomeKey Unit:=wdLine
    .MoveDown Unit:=wdParagraph,
Count:=1, Extend:=wdExtend
    .MoveLeft Unit:=wdCharacter, Count:=1,
Extend:=wdExtend
    ActiveDocument.AttachedTemplate.Auto-
toTextEntries.AddRange:=Selection.Range,
Name:=TxtName
    .MoveDown
End With
Wend
End Sub ' CreateAutotext

```

Các vĩ lệnh sao chép AutoCorrect

Một cách tương tự, xây dựng vĩ lệnh Copy-
AutoCorrect() sao chép nội dung của từ điển
AutoCorrect lên cửa sổ văn bản AllMyAuto-
Correct.doc hiện hành.

```

Sub CopyAutoCorrect()
    ' CopyAutoCorrect Macro
    ' Macro recorded 16/03/2009 by Phan Huy
Khánh
    N = AutoCorrect.Entries.Count
    Windows("AllMyAutoCorrect.doc ").Ac-
tivate
For i = 1 To N

```

```

    x = AutoCorrect.Entries.Item(i).Name
    y = AutoCorrect.Entries.Item(i).Value
    Selection.TypeText Text:=x & "=" & y &
Chr(13)
Next i
    ActiveDocument.Save
End Sub ' CopyAutoCorrect Macro

```

Sau khi lưu cất nội dung tệp AllMyAuto-
Correct.doc lên hệ thống lưu trữ, có thể sử dụng
công cụ chuyển mã của UniKey để chuyển nội
dung từ điển AutoCorrect sang Unicode. Tiếp
theo là bước tạo mới từ điển AutoCorrect. Vĩ
lệnh dưới đây cho phép xây dựng mới từ điển
AutoCorrect.

```

Sub NewAutoTextCreate()
    ' NewAutoTextCreate Macro
    ' Macro recorded 18/03/2009 by Phan Huy
Khánh
    N = "x"
While N <> "zzz"
    t = Selection.Paragraphs.First
    N = Left(t, Len(t) - 1)
    Selection.MoveDown Unit:=wdParagraph,
Count:=1
    t = Selection.Paragraphs.First
    V = Left(t, Len(t) - 1)
    Selection.MoveDown Unit:=wdParagraph,
Count:=1
    ' Update
    AutoCorrect.Entries.Add Name:=N,
Value:=V
With AutoCorrect

```

```
.CorrectInitialCaps = True
.CorrectSentenceCaps = True
.CorrectDays = True
.CorrectCapsLock = True
.ReplaceText = True
.ReplaceTextFromSpellingChecker = True
.CorrectKeyboardSetting = False
.DisplayAutoCorrectOptions = True
.CorrectTableCells = True
```

End With

Wend

End Sub ' NewAutoTextCreate Macro

b. Đánh giá kết quả và hướng phát triển

Từ điển cụm từ gõ tắt AutoCorrect cho phép sử dụng thuận tiện đồng thời trong các ứng dụng Word, Excel và PowerPoint của MSOffice. Bằng cách trao đổi qua lại những cụm từ gõ tắt đã được xây dựng từ trước đến nay trong MSOffice, trong UniKey, nhờ kỹ thuật lập trình vĩ lệnh VBA trên đây, chúng tôi đã có được từ điển cụm từ gõ tắt với số lượng xấp xỉ 1500 từ.

Trên cơ sở đó, chúng tôi cũng có thể xây dựng từ điển chuyên dụng cho AutoText để có thể thao tác với các chữ Hán, bằng cách gõ từ Hán-Việt để nhận được một chữ Hán, chẳng hạn, NSD gõ nhân chỉ sơ tính bản thiện để nhận được câu chữ Hán人之初性本善. Tuy nhiên, trong trường hợp này, cần giải quyết vấn đề đồng nghĩa, chẳng hạn cùng một từ Hán-Việt nhân, có tới 13 chữ Hán 人, 仁, 儿, 因, 姻, 氮, 裡, 紉, 網 tương ứng.

Một trong những vấn đề hay gặp là chuyển

đổi giữa văn bản tiếng Việt không dấu (thường dùng trong E-Mail, tin nhắn...) thành có dấu. Dùng AutoText có thể dễ dàng thay thế từng từ đơn không dấu thành có dấu, chẳng hạn nhan thành nhân (!), nhưng không thể thay thế chỉ thành chi (giữ nguyên !), tinh thành tính, v.v... Giải pháp của chúng tôi là xây dựng từ điển các cụm từ không dấu có độ dài tối đa theo tiếp cận phân tách văn bản Maximum Matching [2] trong xử lý ngôn ngữ tự nhiên. Chẳng hạn, tinh ban thien / tính bản thiện.

Hệ thống các trình tiện ích viết bằng vĩ lệnh VBA được lưu giữ trong các mẫu văn bản Normal.dot cho phép NSD tùy ý cải biên, sửa đổi, bổ sung..., nhằm tạo ra các khả năng phát triển thuận tiện trong môi trường MSOffice ■



TÀI LIỆU THAM KHẢO

- [1] Phan Huy Khánh. «Good Spelling of Vietnamese Texts, one aspect of computational linguistics in Vietnam». Proceeding of ACL-2000, 38th Conference, HồngKông, 2000, t.593-594.
- [2] Phan Huy Khánh, Võ Trung Hùng. « Thiết kế cơ sở dữ liệu đa ngữ ngữ pháp tiếng Việt ». Tạp chí Khoa học Công nghệ các trường Đại học Bách khoa, No 36+37, 2002, t19-t24.
- [3] Phan Huy Khánh. «Xây dựng từ điển đa ngữ sử dụng dạng thức văn bản RTF Winword». Kỷ yếu Hội thảo Khoa học Quốc gia ICT.rda'2003, trang 323-332.

Tài liệu lấy từ internet :

- [4] A. Cooper. «Why I am called “the Father of Visual Basic”». Cooper Interaction Design, 1996.
- [5] Microsoft Knowledge Base Article - 212682. «VBA macro examples to insert text into a document in Word 2000». Trang web : <http://support.microsoft.com/>
- [6] Microsoft Knowledge Base Article - 212710. «WD2000: Retrieving Settings from Word Dialog Boxes Using OLE». Trang web : <http://support.microsoft.com/>
- [7] Microsoft Knowledge Base Article - 226118. «List of resources that are available to help

