Sub WordPairAlyseForMac()

' Version 04.02.20

' Creates a file of all the adjacent word pairs

pausePercent = 100

stepPercent = 10

goFast = False

goFast = True

windowHeight = 300

topWindowHeight = 200

windowWidth = 600

' Number of exceptions list

mySet = 1

' mySet = 0 ' Use every single word, no exceptions

Select Case mySet

Case 1:

nonWords = "a,an,also,and,are,as,at,be,by,can,did,for,from,had,has,have"

nonWords = nonWords & ",however,if,in,is,it,its,my,no,of,on,or,our,shown,so"

nonWords = nonWords & ",than,that,the,their,there,these,they,this,to,upon"

nonWords = nonWords & ",very,was,we,what,when,where,which,will,with,you,your"

Case 2

' Set based on statistically most common words

nonWords = ",the,of,is,and,to,in,this,are,be,by,for,as,"

nonWords = nonWords & ",or,at,an,can,it,its,my,no,of,on,or"

Case 3

nonWords = ",blah,blah,their,there,these,they,this"

Case Else

nonWords = ",zcdummyzc"

End Select

Set FUT = ActiveDocument

doingSeveralMacros = (InStr(FUT.Name, "zzTestFile") > 0)

If doingSeveralMacros = False Then

myResponse = MsgBox(" WordPairAlyse" & vbCr & vbCr & \_

"Analyse this document?", vbQuestion \_

+ vbYesNoCancel, "WordPairAlyseForMac")

If myResponse <> vbYes Then Exit Sub

End If

stTime = Left(Time, 5)

Debug.Print vbCr & vbCr & "Started at: " & stTime & \_

" Preparing text for search" & vbCr

Set mainDoc = ActiveDocument

ActiveDocument.ActiveWindow.WindowState = wdWindowStateNormal

startTime = Timer

Set rng = ActiveDocument.Content

Documents.Add

ActiveDocument.ActiveWindow.WindowState = wdWindowStateNormal

Set tempDoc = ActiveDocument

Selection.Text = LCase(rng.Text)

Selection.HomeKey Unit:=wdStory

nonWords = "," & nonWords & ","

nonWords = Replace(nonWords, ",,", ",")

nonWords = Left(nonWords, Len(nonWords) - 1)

' Underline all non-words

wd = Split(nonWords, ",")

Set rng = ActiveDocument.Content

For i = 1 To UBound(wd)

With rng.Find

.ClearFormatting

.Replacement.ClearFormatting

.Text = wd(i)

.Wrap = wdFindContinue

.Replacement.Text = ""

.Replacement.Font.Underline = True

.MatchWildcards = False

.MatchWholeWord = True

.Execute Replace:=wdReplaceAll

End With

If goFast = False Then DoEvents

Next i

' Replace anything underlined with a full stop

Set rng = ActiveDocument.Content

With rng.Find

.ClearFormatting

.Replacement.ClearFormatting

.Font.Underline = True

.Text = ""

.Wrap = wdFindContinue

.Replacement.Text = "."

.MatchWildcards = False

.Execute Replace:=wdReplaceAll

If goFast = False Then DoEvents

End With

Set rng = ActiveDocument.Content

With rng.Find

.ClearFormatting

.Replacement.ClearFormatting

.Text = ChrW(8217)

.Wrap = wdFindContinue

.Replacement.Text = "."

.MatchWildcards = False

.Execute Replace:=wdReplaceAll

If goFast = False Then DoEvents

End With

Set rng = ActiveDocument.Content

Documents.Add

ActiveDocument.ActiveWindow.WindowState = wdWindowStateNormal

Set tempDoc2 = ActiveDocument

Selection.Text = rng.Text

Selection.HomeKey Unit:=wdStory

tempDoc.Close SaveChanges:=False

tempDoc2.Activate

FRtime = Timer

Set rng = tempDoc2.Content

lenRng = rng.End

StatusBar = "Started at: " & stTime

With rng.Find

.ClearFormatting

.Replacement.ClearFormatting

.Text = "[a-z]{2,} [a-z]{2,}"

.Wrap = wdFindStop

.Replacement.Text = ""

.Forward = True

.MatchWildcards = True

.Execute

End With

stopNow = False

stopAt = lenRng \* (pausePercent + 0.1) / 100

Do While rng.Find.Found = True And stopNow = False

startNow = rng.Start

spPos = InStr(rng, " ")

startNew = startNow + spPos

wdPair = rng.Text

schWd = Replace(wdPair, " ", "")

Set rng2 = tempDoc2.Content

myTot = rng2.End

With rng2.Find

.ClearFormatting

.Replacement.ClearFormatting

.Text = schWd

.Wrap = wdFindStop

.Replacement.Text = ""

.Forward = True

.MatchCase = False

.MatchWholeWord = True

.MatchWildcards = False

.Execute

End With

If goFast = False Then DoEvents

If rng2.Find.Found = True And stopNow = False Then

textLeft = Int((lenRng - startNew) / 100)

pCent = Int(startNew / lenRng \* 100)

pCentText = Trim(Str(pCent)) & "%"

If InStr(allWords, wdPair) = 0 Then

Beep

allWords = allWords & wdPair & ":" & schWd & ":"

timNow = Timer

timGone = timNow - startTime

m = Int(timGone / 60)

s = Int(timGone) - m \* 60

If Len(s) = 1 Then s = "0" & s

estTime = Int((timGone \* lenRng / startNew \_

+ FRtime - startTime) / 60)

hr = Val(Left(stTime, 2))

mn = Val(Right(stTime, 2))

mn = mn + estTime

carryH = Int(mn / 60)

mn = mn Mod 60

hr = hr + carryH

hrText = Trim(Str(hr))

mnText = Trim(Str(mn))

If Len(hrText) = 1 Then hrText = "0" & hrText

If Len(mnText) = 1 Then mnText = "0" & mnText

endTime = hrText & ":" & mnText

rng2.Start = 0

rng2.End = 0

With rng2.Find

.ClearFormatting

.Replacement.ClearFormatting

.Text = "<" & wdPair & ">"

.Replacement.Text = "^&!"

.MatchWildcards = True

.Execute Replace:=wdReplaceAll

End With

myCount = ActiveDocument.Content.End - myTot

myPData = Trim(Str(myCount))

If myCount > 0 Then

WordBasic.EditUndo

myResult = myResult & wdPair & " . . " & \_

myPData & vbCr

End If

With rng2.Find

.ClearFormatting

.Replacement.ClearFormatting

.Text = "<" & schWd & ">"

.Replacement.Text = "^&!"

.MatchWildcards = True

.Execute Replace:=wdReplaceAll

End With

myCount = ActiveDocument.Content.End - myTot

myWData = Trim(Str(myCount))

If myCount > 0 Then

WordBasic.EditUndo

myResult = myResult & schWd & " . . " & \_

myWData & vbCr & vbCr

End If

secs = Trim(Str(s))

If Len(secs) = 1 Then secs = " " & secs

mins = Trim(Str(m))

If m < 10 Then mins = " " & mins

dispTime = mins & " m " & secs & " s"

StatusBar = wdPair & " (" & myPData & ":" & myWData & \_

") " & pCentText & " " & dispTime \_

& " Finish time: " & endTime

If Len(wdPair) < 11 Then

closeBkt = ") "

Else

closeBkt = ")"

End If

Debug.Print wdPair & " (" & myPData & ":" & myWData & \_

closeBkt, pCentText, dispTime & " Finish time: " \_

& endTime

End If

End If

' Be sure you're past the previous occurrence

rng.Start = startNew

rng.End = startNew

rng.Find.Execute

If goFast = False Then DoEvents

If startNew > stopAt Then

Beep

myTime = Timer

Do

Loop Until Timer > myTime + 0.2

Beep

If doingSeveralMacros = False Then

stepPercent = Val(InputBox("Done: " & pausePercent \_

& "%" & vbCr & "Continue searching? (Cancel = Stop)", \_

"WordPairAlyse", Trim(Str(stepPercent))))

If stepPercent = 0 Then

stopNow = True

Else

stopNow = False

pausePercent = pausePercent + stepPercent

stopAt = lenRng \* (pausePercent + 0.1) / 100

End If

End If

End If

Loop

' Currently in tempDoc2

Selection.WholeStory

Selection.TypeText Text:=myResult

Set rng = tempDoc2.Content

With rng.Find

.ClearFormatting

.Replacement.ClearFormatting

.Text = "^p^p"

.Replacement.Text = "zczc"

.MatchWildcards = False

.Execute Replace:=wdReplaceAll

.Text = "^p"

.Replacement.Text = ":"

.Execute Replace:=wdReplaceAll

.Text = "zczc"

.Replacement.Text = "^p"

.Execute Replace:=wdReplaceAll

End With

Selection.WholeStory

Selection.Sort

With rng.Find

.Text = "^p"

.Replacement.Text = "^p^p"

.Execute Replace:=wdReplaceAll

.Text = ":"

.Replacement.Text = "^p"

.Execute Replace:=wdReplaceAll

End With

Selection.Start = 0

Selection.End = 4

Selection.Delete

Selection.TypeText Text:="Word pair inconsistencies" & vbCr

timNow = Timer

timGone = timNow - startTime

m = Int(timGone / 60)

s = Int(timGone) - m \* 60

Debug.Print , , , Trim(Str(m)) & " m " & Trim(Str(s)) & " s"

Debug.Print , "", "mySet = "; Str(mySet) & vbCr

Beep

myTime = Timer

Do

Loop Until Timer > myTime + 0.3

Beep

myTime = Timer

Do

Loop Until Timer > myTime + 0.3

If doingSeveralMacros = False Then

Beep

MsgBox "Total time:" & Str(m) & " m " & Str(s) & " s"

Else

FUT.Activate

End If

End Sub