{

"collab\_server" : "",

"contents" : "######préparation packages, wd, etc\nsetwd(\"C:/Users/lb3/OneDrive/sync/git/loubill/Domino/R/Domino/MTX\")\ninstall.packages(c('tm', 'SnowballC', 'wordcloud', 'topicmodels', \"koRpus\", \"mclustcomp\", \"devtools\", \"installr\", \"RNewsflow\", \"koRpus\",\"wordVectors\"))\n# installr::install.Rtools()\n# devtools::install\_github(\"lbilliet/loubill\")\n# slam\_url <- \"https://cran.r-project.org/src/contrib/Archive/slam/slam\_0.1-37.tar.gz\"\n# install\_url(slam\_url)\n# urlSlam <- \"https://cran.r-project.org/bin/windows/contrib/3.6/slam\_0.1-43.zip\"\n# install\_url(urlSlam)\n# install\_github(\"bmschmidt/wordVectors\")\n#devtools::install\_github(\"bmschmidt/wordVectors\")\nlibrary('devtools')\nlibrary(tm)\nlibrary(SnowballC)\nlibrary(wordcloud)\nlibrary(topicmodels)\nlibrary(\"koRpus\")\nlibrary(\"wordVectors\")\nlibrary(\"RNewsflow\")\nlibrary(\"mclustcomp\")\n\n######import données\ndf.mtx<-read.csv2(\"data/mtx.csv\", sep = \"\\n\" ,header=F, quote = \"\", stringsAsFactors = F, fileEncoding = \"UTF8\")\ndf.alt<-read.csv2(\"data/alt.csv\", sep = \"\\n\" ,header=F, quote = \"\", stringsAsFactors = F, fileEncoding = \"UTF8\")\n\n######conversion encodage suppression caractère speciaux\n#définition fonctio suppression char speciaux\nUnaccent <- function(text) {\n text <- gsub(\"['`^~\\\"]\", \" \", text)\n text <- iconv(text, to=\"ASCII//TRANSLIT//IGNORE\")\n text <- gsub(\"['`^~\\\"]\", \"\", text)\n return(text)\n}\n###application fonction suppr char speciaux\ndf.mtx.unaccent<-Unaccent(df.mtx)\ndf.alt.unaccent<-Unaccent(df.alt)\n\n#constitution corpus\ncorpus.mtx = Corpus(VectorSource(df.mtx.unaccent))\ncorpus.alt = Corpus(VectorSource(df.alt.unaccent))\n\n#####préparation texte\n#retirer ponctuation\ncorpus.mtx<-tm\_map(corpus.mtx, removePunctuation)\ncorpus.alt<-tm\_map(corpus.alt, removePunctuation)\n#retirer stopwords\ncorpus.mtx <- tm\_map(corpus.mtx, removeWords, stopwords(\"french\"))\ncorpus.alt <- tm\_map(corpus.alt, removeWords, stopwords(\"french\"))\n#stemmatiser\ncorpus.mtx <- tm\_map(corpus.mtx, stemDocument)\ncorpus.alt <- tm\_map(corpus.alt, stemDocument)\n\n#######consitution matrice documents\n#matrice frequence termes\ntdm.mtx <- TermDocumentMatrix(corpus.mtx)\ntdm.alt <- TermDocumentMatrix(corpus.alt)\ninspect(tdm.mtx)\ninspect(tdm.alt)\n#matrice frequence documents\ndtm.mtx <- DocumentTermMatrix(corpus.mtx, control = list(weighting = weightTfIdf, stopwords = TRUE))\ninspect(dtm.mtx)\ndtm.alt <- DocumentTermMatrix(corpus.alt, control = list(weighting = weightTfIdf, stopwords = TRUE))\ninspect(dtm.alt)\n#démo rch termes retrouvés à une fréq de 2\nfreq.tdm.mtx<-findFreqTerms(tdm.mtx,lowfreq = 5, highfreq = Inf)\nsummary(freq.tdm.mtx)\nfreq.tdm.alt<-findFreqTerms(tdm.alt,lowfreq = 5, highfreq = Inf)\nsummary(freq.tdm.alt)\n\n################################\n#boucle constitution vecteur concaténation corpus mtx\ntxt.mtx<-NULL\nfor (i in 1 : nrow(df.mtx)){\n txt.mtx<-paste0(result, df.mtx[i,])\n}\n#normalisation contenu textuel du vecteur\n#suppression ponctuation\ntxt.mtx<-gsub(\"[[:punct:]]\", \" \", txt.mtx)###remove punct atxt cette méthode pour remplacer par des whitespaces plutôt que les suppr et risque de concaténer des mots ensembles et former de nouveaux mots\ntxt.mtx<-Unaccent(txt.mtx)\ntxt.mtx<-tolower(txt.mtx)\nvec.mtx<-unlist(strsplit(txt.mtx, split = \" \"))\nvec.mtx<-gsub(\" \",\"\",vec.mtx)\nvec.mtx[vec.mtx != \"\"]\nstem.vec.mtx<-wordStem(vec.mtx)\n\n#boucle constitution vecteur concaténation corpus alt\ntxt.alt<-NULL\nfor (i in 1 : nrow(df.alt)){\n txt.alt<-paste0(result, df.alt[i,])\n}\n#normalisation contenu textuel du txtteur\n#suppression ponctuation\ntxt.alt<-gsub(\"[[:punct:]]\", \" \", txt.alt)###remove punct atxt cette méthode pour remplacer par des whitespaces plutôt que les suppr et risque de concaténer des mots ensembles et former de nouveaux mots\ntxt.alt<-Unaccent(txt.alt)\ntxt.alt<-tolower(txt.alt)\nvec.alt<-unlist(strsplit(txt.alt, split = \" \"))\nvec.alt<-gsub(\" \",\"\",vec.alt)\nvec.alt[vec.alt != \"\"]\nstem.vec.alt<-wordStem(vec.alt)\n#mettre vecteurs même longueur\nstem.vec.alt<-c(stem.vec.alt, rep(NA, length(stem.vec.mtx)-length(stem.vec.alt)))\nwrite.table(stem.vec.alt, file = \"data/stemVecAlt.txt\", sep = \"\\t\", row.names = F, quote = F)\nwrite.table(stem.vec.mtx, file = \"data/stemVecMtx.txt\", sep = \"\\t\", row.names = F, quote = F)\n\n#comparaison via mclustcomp\nmclustcomp(stem.vec.mtx,stem.vec.alt, types = \"all\")#pb ici il faut une boucle pour parcourir chaque vecteur de la df\n",

"created" : 1529328095287.000,

"dirty" : false,

"encoding" : "UTF-8",

"folds" : "",

"hash" : "2515890357",

"id" : "286F9C1B",

"lastKnownWriteTime" : 1529480966,

"last\_content\_update" : 1529480966,

"path" : "C:/Users/lb3/OneDrive/sync/git/loubill/Domino/R/Domino/MTX/casConcretMTX.R",

"project\_path" : "casConcretMTX.R",

"properties" : {

"tempName" : "Untitled1"

},

"relative\_order" : 3,

"source\_on\_save" : false,

"source\_window" : "",

"type" : "r\_source"

}