Problem2\_final

Reshms Sekar

August 19, 2015

library('tm') ## Loading required package: NLP library('randomForest') ## randomForest 4.6-10 ## Type rfNews() to see new features/changes/bug fixes. library('e1071') library('rpart') library('ggplot2') ## ## Attaching package: 'ggplot2' ## ## The following object is masked from 'package:NLP': ## ## annotate library('caret') ## Loading required package: lattice setwd("~/Desktop") #reader function readerPlain = function(fname){ readPlain(elem=list(content=readLines(fname)), id=fname, language='en') }

author\_dirs = Sys.glob('~/Desktop/ReutersC50/C50train/*') file\_list = NULL train\_labels = NULL for(author in author\_dirs) { author\_name = substring(author, first=23) files\_to\_add = Sys.glob(paste0(author, '/*.txt')) file\_list = append(file\_list, files\_to\_add) train\_labels = append(train\_labels, rep(author\_name, length(files\_to\_add))) }

# Named conversion & cleanup

all\_docs = lapply(file\_list, readerPlain) #names(all\_docs) = file\_list #names(all\_docs) = sub('.txt', '', names(all\_docs))

# Initialize Training Corpus

train\_corpus = Corpus(VectorSource(all\_docs)) names(train\_corpus) = file\_list

# Tokenization of training Corpus

train\_corpus = tm\_map(train\_corpus, content\_transformer(tolower)) train\_corpus = tm\_map(train\_corpus, content\_transformer(removeNumbers)) train\_corpus = tm\_map(train\_corpus, content\_transformer(removePunctuation)) train\_corpus = tm\_map(train\_corpus, content\_transformer(stripWhitespace)) train\_corpus = tm\_map(train\_corpus, content\_transformer(removeWords), stopwords("SMART"))

# Create training DTM & dense matrix

DTM\_train = DocumentTermMatrix(train\_corpus) DTM\_train = removeSparseTerms(DTM\_train, 0.975)

author\_dirs = Sys.glob('~/Desktop/ReutersC50/C50test/*') file\_list = NULL test\_labels = NULL for(author in author\_dirs) { author\_name = substring(author, first=22) files\_to\_add = Sys.glob(paste0(author, '/*.txt')) file\_list = append(file\_list, files\_to\_add) test\_labels = append(test\_labels, rep(author\_name, length(files\_to\_add))) }

# Named conversion & cleanup

all\_docs = lapply(file\_list, readerPlain) #names(all\_docs) = file\_list #names(all\_docs) = sub('.txt', '', names(all\_docs))

# Initialize Testing Corpus

test\_corpus = Corpus(VectorSource(all\_docs)) names(test\_corpus) = file\_list

# Tokenization of Testing Corpus

test\_corpus = tm\_map(test\_corpus, content\_transformer(tolower)) test\_corpus = tm\_map(test\_corpus, content\_transformer(removeNumbers)) test\_corpus = tm\_map(test\_corpus, content\_transformer(removePunctuation)) test\_corpus = tm\_map(test\_corpus,content\_transformer(stripWhitespace)) test\_corpus = tm\_map(test\_corpus, content\_transformer(removeWords), stopwords("SMART"))

reuters\_dict = NULL reuters\_dict = dimnames(DTM\_train)[[2]]

# Create testing DTM & matrix using dictionary words only

DTM\_test = DocumentTermMatrix(test\_corpus, list(dictionary=reuters\_dict)) DTM\_test = removeSparseTerms(DTM\_test, 0.975) #DTM\_test = as.matrix(DTM\_test) DTM\_train\_df = as.data.frame(inspect(DTM\_train))