SLP Assignment F October 8, 2020

**Word Embeddings**

In this assignment you will explore the use of vector representations of lexical identity.

Part 1. For your Information Retrieval assignments, briefly describe how you could use word vectors and what benefit you would expect. Please answer concisely, with a short paragraph for each.

Part 2. Get familiar with the use of some set of pretrained dimensions. Links to word2vec, glove, and Ratnaparkhi’s bitstrings are on the course homepage, but for this I recommend the glove 50-dimension embeddings. Specifically, determine whether *rabbit* or *bunny* is closest to *food*, and determine whether *texas-el* is closer to *juarez,* to *chicago* or to *rabbit.*

# Nigel Ward, October 2018

# Code Fragments SLP Assignment F: Word Vectors

import numpy as np

import math

def loadGloveModel(gloveFile): # from Karishma Malkan on stackoverflow

print "Loading Glove Model"

f = open(gloveFile,'r')

model = {}

for line in f:

splitLine = line.split()

word = splitLine[0]

embedding = np.array([float(val) for val in splitLine[1:]])

model[word] = embedding

print "Done.",len(model)," words loaded!"

return model

def distance(vec1, vec2):

return math.sqrt(sum([(x-y)\*(x-y) for x,y in zip( vec1, vec2)]))

def mostSimilar(query):

distances = [distance(model[x], model[query]) for x in model.keys()[:200000]]

# distances = [distance(model[x], model[query]) for x in model.keys()]

minval = min(distances)

print 'for %s: min distance of %.2f found for %s' % (query, minval, model.keys()[distances.index(minval)])

#---------------------------------

model = loadGloveModel('glove.6B.50d.txt') # 17 sec on my desktop

#print model['texas-el']

mostSimilar('texas')