ASSIGNMENT 5:-

import numpy as np

import keras.backend as K

from keras.models import Sequential

from keras.layers import Dense,Embedding,Lambda

from keras.utils import np\_utils

from keras.preprocessing import sequence

from keras.preprocessing.text import Tokenizer

import gensim

data = open("/content/corona.txt","r")

covid\_data= [text for text in data if text.count("")>=2]

vectorize=Tokenizer()

vectorize.fit\_on\_texts(covid\_data)

covid\_data=vectorize.texts\_to\_sequences(covid\_data)

total\_vocab=sum(len(s) for s in covid\_data)

word\_count=len(vectorize.word\_index)+1

window\_size=2

def cbow\_model(data,windows\_size, total\_vocab):

  total\_length=window\_size\*2

  for text in data:

    text\_len=len(text)

    for idx, word in enumerate(text):

      context\_word=[]

      target=[]

      begin=idx-window\_size

      end=idx+window\_size+1

      context\_word.append([text[i] for i in range(begin,end) if 0<- i< text\_len and i!=idx])

      target.append(word)

      contextual = sequence.pad\_sequences(context\_word, total\_length=total\_length)

      final\_target=np\_utils.to\_categorical(target, total\_vocab)

      yield(contextual, final\_target)

model=Sequential()

model.add(Embedding(input\_dim=total\_vocab,output\_dim=100,input\_length=window\_size\*2))

model.add(Lambda(lambda x:K.mean(x,axis=1), output\_shape=(100,)))

model.add(Dense(total\_vocab, activation="softmax"))

model.compile(loss="categorical\_crossentropy", optimizer="adam")

for i in range(10):

  cost=0

  for x, y in cbow\_model(data,window\_size, total\_vocab):

    cost+=model.train\_on\_batch(contextual, final\_target)

  print(i, cost)

dimensions = 100

vect\_file=open("/content/drive/MyDrive/vector.txt", "w")

vect\_file.write('{} {}\n'.format(total\_vocab, dimensions))

weight=model.get\_weights()[0]

for text, i in vectorize.word\_index.items():

  final\_vec="".join(map(str, list(weight[i,:])))

  vect\_file.write('{}{}\n'.format(text, final\_vec))

vect\_file.close()

cbow\_output=gensim.models.KeyedVectors.load\_word2vec\_format("/content/drive/MyDrive/vector.txt", binary=False)

cbow\_output.most\_similar(positive=["virus"])

OUTPUT:-

0 0 1 0 2 0 3 0 4 0 5 0 6 0 7 0 8 0 9 0

8