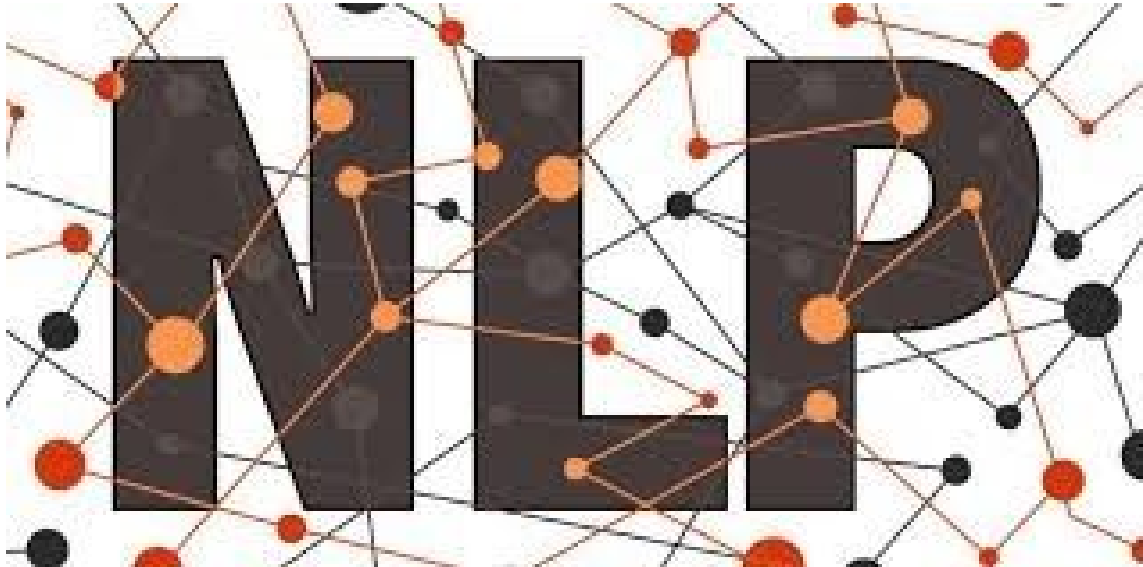


Assignment 4



Lakshya(MT19067)

M.tech(3rd Sem)

Question 1

Introduction

Multilingual extension of BingLiu lexicon

DataSet

Bing Liu lexicon CSV.

```
a+ positive
abound positive
abounds positive
abundance positive
abundant positive
accessible positive
accessible positive
acclaim positive
acclaimed positive
acclamation positive
accolade positive
accolades positive
accommodative positive
acomodative positive
accomplish positive
accomplished positive
accomplishment positive
accomplishments positive
accurate positive
```

CSV file contains English lexicons and their polarity.

English.txt

That place was awesome , , , sally !
Awsome Pizza especially the Margheritta slice .
Always busy but fast moving .
Great atmoshere and worth every bit .
Open late -LRB- well as late as I ever got there and I 'm a night person -RRB- .
Winnie and her staff are the best crew you can find serving you .
The food is reliable and the price is moderate .
What more can you ask for ?
For authentic Thai food , look no further than Toons .
Ive been to many Thai restaurants in Manhattan before , and Toons is by far the best Thai .
Try the Pad Thai , or sample anything on the appetizer menu ... they 're all delicious .
Everything about this restaurant was special .
The service was attentive , yet discreet .
The flavors robust and subtle .
The brioche and lollies as party favors is a cute and sweet touch to a most memorable meal .
I 'm saving up for my next visit .
I went there in late afternoon for some bite size food and refreshment with my date .
The place was quiet and delightful .
Service was good and food is wonderful .
I did not try the caviar but I tried their salmon and crab salad -LRB- they are all good

The english.txt file contains the English corpus. Each line is separated by a newline character.

Hindi.txt

फेसबुक का सिक््योरिटी चेकअप फीचर पॉपअप की तरह यूजर्स को दिखाई देगा ।
इस पॉपअप बॉक्स में पासवर्ड चेंज करने , लॉगइन अलर्ट्स चालू करने और मौजूदा फेसबुक
अब नए फीचर से इस तरह की गलतियां नहीं होंगी और अकाउंट को हैक करना बेहद मुश्किल
Twitter periscope App एंड्रॉयड 4.4.4 किटकैट या उससे ऊपर के वर्जन पर
पेरिस्कोप एप में रिज्यूम नोटिफिकेशन फीचर दिया गया है जो बहुत ही खास है ।
इस फीचर के तहत यूजर ब्रॉडकास्ट के जरिए देखे जाने वाले वीडियो रुकने पर फिर वहीं
इस फीचर के तहत ब्रॉडकास्टर का टाइम और मोबाइल डेटा दोनों की बचत होती है ।
आपके स्मार्टफोन में ज्यादातर एप ऐसे ही हैं जिनसे आपको अपनी जेब का पैसा खर्च कर ही
3जी पर मैसेज डिलीवरी व्हाट्सपप और आर्इमैसेज की तुलना में बहुत तेज है ।
व्हाट्सपप जैसा फीचर है ।
जिस भी व्यक्ति ने एक ओएस के रूप में ब्लैकबेरी 10 का अनुभव नहीं किया है , इस पर
हालांकि इसके अधिकतर भाग सही स्टैंडर्ड रूप में दिखते हैं , फिर भी पूरे एप्प में साइड स
आपकी रुचि बनाये रखने के लिए , इस एप्प में काले , आसमानी , नीले और सफेद रंग
कॉन्टैक्ट लिस्ट या तो छोटे कॉन्टैक्ट इमेज के साथ हो सकती है या आपके कॉन्टैक्ट में प
फिल्टर करने की सुविधा नहीं है ।

The hindi.txt file contains the Hindi corpus. Each line is separated by a newline character.

English-Hindi.txt

firmly ||| firmly
simplicity ||| सादगी बोलती है
2011 ||| 2011
cross ||| क्रॉस
i-pod ||| i-pod
lay ||| लेटो
suggestions ||| सुझाव
viewed ||| देखा गया
pace ||| के लिए की जाने वाली परार्थना
accurately ||| यथासंभव
mhz ||| mhz
disposable ||| डिस्पोजेबल
30gb ||| 30gb
mbps ||| mbps
circular ||| गोलाकार
naturally ||| स्वाभाविक रूप से
3000 ||| 3000
blast ||| विस्फोट
communicate ||| बातचीत

This file contains english - hindi word pair separated by "|||".

Methodology

English to Hindi Dictionary

```
the:['']
.:['']
,:['']
i:['मै', 'is']
to:['करने के लि']
and:['औ']
a:['ए', 'के रूप मे']
it:['य', 'इसक']
is:['ह']
of:['क']
this:['य']
for:['के लि']
that:['व']
in:['मे', 'in']
you:['आ']
with:['के सा']
my:['मेर']
on:['प']
have:['पा', 'अमी']
```

Length of the dictionary

```
Length of eng-hin dictionary 135786
```

```
def get_L1():
    f = open("english-hindi-dictionary.txt", "r")
    eng_to_hin = {}
    for x in f:
        words = x.strip().split("|||")
        eng = lemmatizer.lemmatize(words[0].strip())
        hin = words[1][:words[1].find("\n")].strip()
        if eng not in eng_to_hin:
            eng_to_hin[eng] = list()
        eng_to_hin[eng].append(hin)
```

For generating eng to hindi dictionary

L1 lexicon dictionary

```
('accomplishment', 'positive'):['उपलब्ध', 'उपलब्धियो']
('achievement', 'positive'):['उपलब्ध', 'उपलब्धिया']
('advantage', 'positive'):['ला', 'ला']
('appeal', 'positive'):['अपी', 'अपी']
('approval', 'positive'):['अनुमोद', 'स्वीकृत']
('assurance', 'positive'):['आश्वास', 'आश्वासन क']
('award', 'positive'):['पुरस्का', 'पुरस्का']
('bargain', 'positive'):['सौद', 'सस्ते दामो']
('beauty', 'positive'):['सौंदर्', 'हसीनाओं का जलव']
('benefit', 'positive'):['ला', 'ला']
('blessing', 'positive'):['आशीर्वा', 'बनें खुशहा']
('bonus', 'positive'):['बोन', 'बोन']
('boom', 'positive'):['बू', 'में उछाल का नेतृत्व कर रहे हैं']
('breeze', 'positive'):['मंद हव', 'हवाए']
('capability', 'positive'):['क्षमत', 'क्षमताओ']
('celebration', 'positive'):['जश्', 'जश्']
('champion', 'positive'):['चैंपिय', 'चैंपिय']
('charm', 'positive'):['आकर्ष', 'चार्म']
('cheer', 'positive'):['चीयर्स करे', 'खुश हो जा']
('cleaner', 'positive'):['क्लीन', 'क्लीन']
('comfort', 'positive'):['आरा', 'सुख सुविधाओ']
('complement', 'positive'):['पूर', 'पूर']
('compliment', 'positive'):['सराहनाए', 'प्रशंस']
('congratulation', 'positive'):['बधा', 'बधा']
('contribution', 'positive'):['योगदा', 'योगदानो']
('crisp', 'positive'):['कड़', 'के चिप्']
('enhancement', 'positive'):['वृद्ध', 'संवर्द्ध']
('enthusiast', 'positive'):['उत्साह', 'प्रेमियो']
('fair', 'positive'):['मैल', 'मैल']
('fan', 'positive'):['प्रशंस', 'प्रशंसकों को आवाज़ दी जाती ह']
('favorite', 'positive'):['पसंदीद', 'अपनी पसंदीदा तस्वीरों में शामिल किय']
```

This dictionary has key as a tuple of word and polarity and values as a list of hindi words mapped to that word. The dictionary has been prepared using the lemmatization and words with different hindi meanings maps to the same english word.

Initial length of L1 : 4759

Procedure

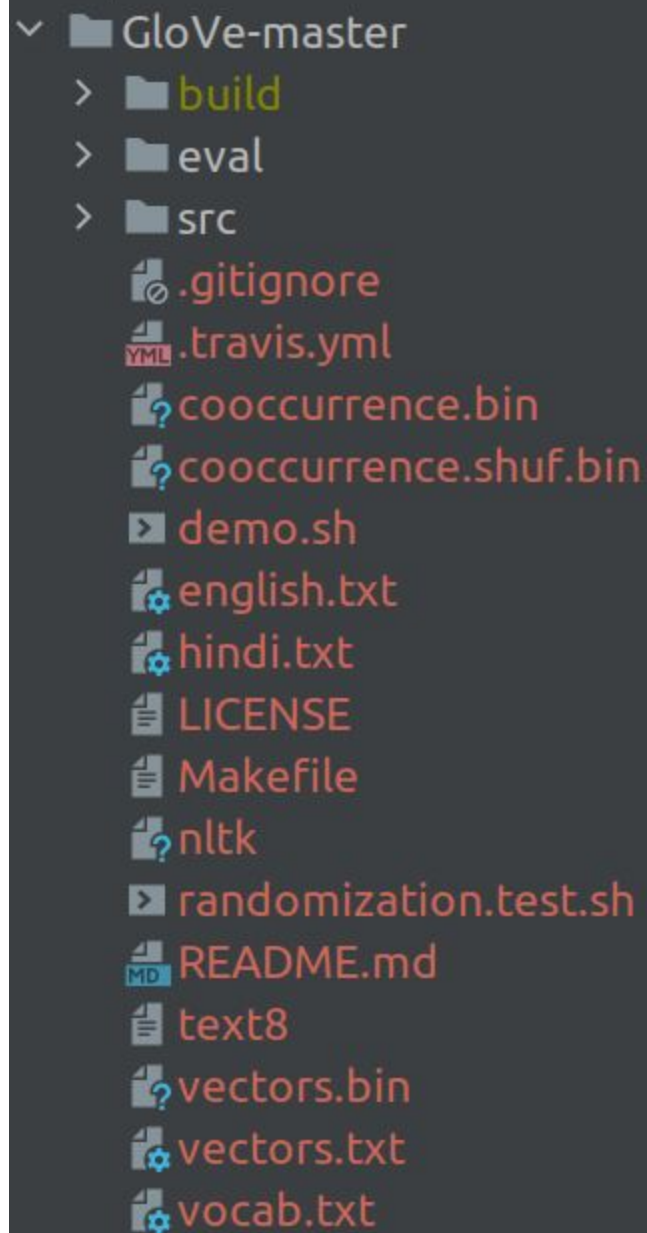
- Extract the english to hindi dictionary.
- Using that english-hindi dictionary, prepare L1.
- Prepare the w2v model and glove model.
- Iterate over L1 and pick key value pair and get the top similar words and prepare all combinations of similar words and check if it can be added in L1.

```
df = pd.read_csv("BingL10.csv")
ar = df.values
bing_eng_hin_l1 = {}
for row in ar:
    row = row[0].split("\t")
    wrd = row[0].strip()
    pol = row[1].strip()
    wrd_bing = lemmatizer.lemmatize(wrd)
    key = (wrd_bing, pol)
    if wrd_bing in eng_to_hin:
        if wrd_bing not in bing_eng_hin_l1:
            bing_eng_hin_l1[key] = list()
        for hin in eng_to_hin[wrd_bing]:
            if not str.isalpha(hin):
                bing_eng_hin_l1[key].append(hin)
return bing_eng_hin_l1, eng_to_hin
```

For generating L1

```
with open (input_file, 'r') as f:
    for i, line in enumerate (f):
        yield gensim.utils.simple_preprocess(line)
eng_documents = list (read_input_eng (data_file))
data_file = "hindi.txt"
def read_input_hin(input_file):
    with open (input_file, 'r') as f:
        for i, line in enumerate (f):
            if "\n" in line:
                line = line[:line.find("\n")].strip()
            yield line.split(" ")
hin_documents = list (read_input_hin (data_file))
model_w2v_en = gensim.models.Word2Vec(eng_documents)
model_w2v_en.train(eng_documents, total_examples=len(eng_documents), epochs=150)
model_w2v_hi = gensim.models.Word2Vec (hin_documents)
model_w2v_hi.train(hin_documents, total_examples=len(hin_documents), epochs=150)
```

Training word2vec model



A screenshot of a file explorer window showing the directory structure of the GloVe-master project. The root directory is 'GloVe-master', which contains subdirectories 'build', 'eval', and 'src'. It also contains several files: '.gitignore', '.travis.yml', 'cooccurrence.bin', 'cooccurrence.shuf.bin', 'demo.sh', 'english.txt', 'hindi.txt', 'LICENSE', 'Makefile', 'nlTK', 'randomization.test.sh', 'README.md', 'text8', 'vectors.bin', 'vectors.txt', and 'vocab.txt'. Each file is represented by an icon indicating its type (e.g., folder, text file, script, binary).

- ▼ GloVe-master
 - > build
 - > eval
 - > src
 - .gitignore
 - .travis.yml
 - cooccurrence.bin
 - cooccurrence.shuf.bin
 - demo.sh
 - english.txt
 - hindi.txt
 - LICENSE
 - Makefile
 - nlTK
 - randomization.test.sh
 - README.md
 - text8
 - vectors.bin
 - vectors.txt
 - vocab.txt

Glove model training


```

easy -0.454471 -0.151761 0.970919 0.132582 -0.880543 -0.746992 0.669263 -0.258077 0.07230
They 0.475927 0.032962 -0.067734 0.035756 -0.272083 -0.455429 -0.018330 -0.121602 0.27473
could 0.034408 -0.187512 -0.291208 0.160221 -0.267265 -0.117033 -0.241000 -0.302220 -0.20
don't 0.217278 0.202584 -0.102343 -0.103782 0.140038 -0.144155 -0.250551 -0.139050 -0.103
make -0.080436 0.405117 0.115509 -0.138915 0.405631 0.050610 0.598151 -0.428937 -0.092161
over -0.043461 0.213656 -0.231495 0.244622 -0.309054 -0.267342 0.089146 -0.200186 -0.1381
want 0.164360 0.116177 -0.169043 -0.070477 0.241634 0.114573 0.177119 -0.409117 -0.027798
first -0.494600 0.068170 -0.721846 0.670335 -0.192730 0.315314 -0.560384 -0.176650 -0.284
've 0.570337 -0.379502 -0.323695 1.152229 -0.987164 -0.512398 -1.035379 0.360080 -0.16494
I'm -0.150168 -0.314042 -0.186625 -0.237530 -0.560629 0.141153 0.236741 -0.244067 -0.2024
delicious 0.013553 -0.139487 -0.126597 0.590346 0.220705 -0.200711 0.656181 -0.044635 -0.
pizza 0.059147 -0.315184 0.044111 0.451950 0.345532 -0.072131 0.308632 0.191381 -0.387031
it's -0.190447 0.019275 -0.140906 0.155629 -0.493562 -0.278539 0.377236 0.123446 -0.19506
still -0.154837 -0.024535 -0.388245 0.202391 -0.677784 -0.338127 0.044102 -0.135537 -0.14

```

English word vector generated with glove

```

है -0.232452 1.912895 0.753697 0.495530 0.239394 0.933999 0.498415 0.033517 0.329663 0.540
। -0.682643 1.934180 -0.029189 -0.762072 0.857318 0.850910 -0.144045 0.723248 0.213364 -0.
के 0.160160 1.646950 -0.190368 -0.217296 -0.003334 0.356133 -0.679881 0.171709 1.045460 -0.
में -1.305571 1.534557 0.901106 -1.215680 -0.561907 0.084396 0.312404 0.008863 0.596095 1.5
, -1.833562 0.913561 1.543191 0.748956 0.119021 1.068017 0.045441 0.362027 1.382408 -0.73
की -0.346534 0.328063 0.857150 -1.237955 -0.678290 0.684419 -1.714488 0.853004 0.095722 1.
और -0.987208 1.006381 0.555520 -0.285530 0.209242 1.047381 -0.981725 -0.339956 -0.191996
से -0.907809 1.742498 -0.871577 0.291011 -0.003330 1.309694 -0.589070 0.286081 1.108029 1.
का 0.612497 0.368747 -0.168131 -0.453192 0.177382 1.544578 0.333691 0.294904 0.032513 1.1
हैं -1.126492 0.634897 0.057836 -0.870221 0.450968 1.464398 -0.637897 0.795912 1.293129 -0.
भी 0.001837 1.429492 0.323163 0.684712 1.024477 1.651641 -0.627731 -0.191116 0.474987 0.4
को -0.322569 1.327392 -0.538403 -0.590669 1.117866 1.498315 -0.264141 0.309368 0.425545 0.
यह -0.080291 1.005469 -0.309988 0.759094 -0.556550 0.415874 0.448611 0.717766 0.572580 0.
पर -0.452691 0.743918 -0.826992 -0.186376 0.140501 0.870834 -0.929388 0.138246 1.597898 0.
इस -0.310833 0.458318 0.113958 -0.904272 0.103353 1.109327 0.206875 0.446199 1.008507 0.5
नहीं -0.708235 0.280059 -0.184362 0.961111 0.481175 1.359840 -0.789781 0.320718 1.062574 0.

```

Hindi word vector generated with glove

HyperParameters Tuned

- Window size
- Vector size
- Epochs

```
('choice', 'positive') : लगाए  
( 'stay', 'positive') : रह  
( 'give', 'positive') : दे  
( 'country', 'positive') : दे  
( 'tell', 'positive') : बता  
( 'say', 'positive') : कह  
( 'transfer', 'positive') : अंतर  
( 'see', 'positive') : देखे  
( 'go', 'positive') : जा  
( 'get', 'positive') : पाए
```

10 New words generated in L1

```
('give', 'positive') : दे
('country', 'positive') : दे
('say', 'positive') : कह
('tell', 'positive') : बता
('come', 'positive') : आ
('see', 'positive') : देखे
('go', 'positive') : जा
('transfer', 'positive') : अंतर
('heard', 'positive') : सुन
('found', 'positive') : मिल
('get', 'positive') : पाए
('saw', 'positive') : देख
('seen', 'positive') : देख
('bought', 'positive') : खरीद
('arrived', 'positive') : पहुंच
('got', 'positive') : मिल
('purchased', 'positive') : खरीद
```

20 New words generated in L1

Assumptions

- Python file is required.
- L1 would look like

Word	Polarity	Hindi
------	----------	-------
- English to English word has to be skipped.
- Lemmatisation has to be performed for better performance.

References

- [Stackoverflow.com](https://stackoverflow.com)
- [Github.com](https://github.com)