**CSI 5386**
**Assignment 1**
libo long 300151908 (part 1) Wei Li 300113733 (part 2) report date: 2/3/2020

# 1 part 1

## 1.1 microblog2011_tokenized

we use SpaCy to tokenize each line of microblog2011.txt. Each url is treated as unique toke. Foreign language are filtered. Each token is bracket by [] in each line

```
1 [Save] [BBC] [World] [Service] [from] [Savage] [Cuts]
[http://www.petitionbuzz.com/petitions/savews]

2 [a] [lot] [of] [people] [always] [make] [fun] [about] [the] [end] [of] [the] [world]
[but] [the] [question] [is] [..] ["ARE] [U] [READY] [FOR] [IT] [?] [..]

3 [ReThink] [Group] [positive] [in] [outlook] [:] [Technology] [staffing] [specialist]
[the] [ReThink] [Group] [expects] [revenues] [to] [be] [] [marg] [...]
[http://bit.ly/hFjtmY]

4 [’] [Zombie] [’] [fund] [manager] [Phoenix] [appoints] [new] [CEO] [:] [Phoenix]
[buys] [up] [funds] [that] [have] [been] [closed] [to] [new] [business] [and] [...]
[http://bit.ly/dXrlH5]

5 [Latest] [:] [:] [Top] [World] [Releases]
[http://globalclassified.net/2011/02/top-world-releases-2/]

6 [CDT] [presents] [ALICE] [IN] [WONDERLAND] [-] [Catonsville] [Dinner] [has] [posted]
[’] [CDT] [presents] [ALICE] [IN] [WONDERLAND] [’] [to] [the] [...]
[http://fb.me/GMicayT3]

7 [Territory] [Manager] [:] [Location] [:] [Calgary] [,] [Alberta] [,] [CANADA]
[Job] [Category] [:] [bu] [...] [http://bit.ly/e3o7mt] [#] [jobs]

8 [I] [cud] [murder] [sum1] [today] [n] [not] [even] [flinch] [I] [’m] [tht] [fukin]
[angry] [today]

9 [BBC] [News] [-] [Today] [-] [Free] [school] [funding] [plans] [’] [lack] [transparency]
[’] [-] [http://news.bbc.co.uk] [/] [today] [/] [hi] [/] [today] [/]
[newsid_9389000/9389467.stm] [...]

10 [Manchester] [City] [Council] [details] [saving] [cuts] [plan] [:]
[http://bbc.in/fYPYPC] [...] [Depressing] [.] [Apparently] [we] [re] [4th] [most]
[deprived] [&] [top] [5] [hardest] [hit]

11 [http://bit.ly/e0ujdP] [,] [if] [you] [are] [interested] [in] [professional]
[global] [translation] [services]
```

```
12 [Fitness] [First] [to] [float] [but] [is] [n’t] [the] [full] [service] [model]
[dead] [?] [http://bit.ly/evflEb]

13 [David] [Cook] [!] [http://bit.ly/fkj2gk] [has] [the] [mostest] [beautiful] [smile]
[in] [the] [world] [!]

14 [Piss] [off] [.] [Cnt] [stand] [lick] [asses]

15 [BEWARE] [THE] [BLUE] [MEANIES] [:]
[http://bit.ly/hu8iJz] [#] [cuts] [#] [thebluemeanies]

16 [Como] [perde] [os] [dentes] [no] [World] [Of] [Warcraft] [-] [Via] [Alisson]
[http://ow.ly/1beBPo]

17 [How] [exciting] [!] [RT] [@BunchesUK] [:] [Hello] [!] [What] [’s] [happening]
[in] [your] [world] [?] [We] [’re] [all] [gearing] [up] [for] [#] [Valentines] [with]
[bouquets] [flying] [out] [the] [door] [.]

18 [I] [’d] [very] [much] [appreciate] [it] [if] [people] [would] [stop] [broadcasting]
[asking] [me] [to] [add] [people] [on] [BBM] [.]

19 [@samanthaprabu] [sam] [i] [knw] [u] [r] [a] [cricket] [fan] [r] [u] [watching]
[any] [of] [the] [world] [cup] [matches]

20 [John] [Baer] [:] [Who] [did] [n’t] [see] [this] [coming] [?] [:] [TO] [THOSE]
[who] [know] [Ed] [and] [Midge] [Rendell] [-] [heck] [,] [to] [the] [Philly] [world]
[at] [la] [...] [http://bit.ly/ii6WEO]
```

### 1.2 statistic

we read the microblog2011_tokenized.txt file in section 1.1, then we used nltk library to count the number of all element, and types of elements in the file.

numbers of tokens: 932280
types fo tokens: 97011
types/numbers ratio: 0.10405779379585532

### 1.3 frequency of token

we converted all letters to lower case, the we used nltk.FreqDist to count the frequency for each types of token. we listed top 100 tokens by frequency. it shows the higher frequency tokens are punctuation, symbols and short words.

list top 100

```
1 ('#', 30235)
2 (':', 23876)
3 ('the', 21126)
4 (',', 18741)
5 ('.', 17863)
6 ('to', 13877)
```

```
7 (’-’, 13506)
8 (’...’, 12636)
9 (’!’, 11896)
10 (’in’, 10839)
11 (’of’, 10574)
12 (’a’, 10459)
13 (’i’, 8456)
14 (’and’, 8360)
15 (’for’, 7112)
16 (’on’, 6286)
17 (’is’, 6251)
18 ("’s", 5845)
19 (’?’, 5506)
20 (’"’, 5395)
21 (’(’, 4612)
22 (’)’, 4544)
23 (’rt’, 4452)
24 (’you’, 4099)
25 (’at’, 3909)
26 (’it’, 3834)
27 (’egypt’, 3269)
28 ("’", 3208)
29 (’with’, 3145)
30 (’my’, 3122)
31 (’that’, 3023)
32 (’new’, 2978)
33 (’news’, 2717)
34 (’&’, 2546)
35 (’from’, 2472)
36 (’this’, 2439)
37 (’are’, 2429)
38 (’be’, 2271)
39 (’/’, 2183)
40 (’by’, 2052)
41 ("n’t", 2007)
42 (’do’, 1979)
43 (’will’, 1937)
44 (’have’, 1936)
45 (’not’, 1923)
46 (’egyptian’, 1896)
47 (’your’, 1822)
48 (’obama’, 1814)
49 (’state’, 1802)
50 (’me’, 1789)
51 (’we’, 1747)
52 (’just’, 1746)
53 (’as’, 1682)
54 (’us’, 1675)
55 (’out’, 1647)
```

```
56 (’has’, 1593)
57 (’all’, 1569)
58 (’what’, 1471)
59 (’no’, 1464)
60 (’up’, 1459)
61 (’now’, 1444)
62 (’|’, 1422)
63 (’super’, 1416)
64 (’world’, 1412)
65 (’..’, 1411)
66 (’was’, 1405)
67 (’’, 1389)
68 (’so’, 1388)
69 (’an’, 1372)
70 (’jan25’, 1337)
71 (’social’, 1332)
72 (’media’, 1330)
73 (’like’, 1315)
74 (’white’, 1307)
75 (’via’, 1295)
76 (’bowl’, 1289)
77 (’get’, 1287)
78 (’about’, 1280)
79 ("’m", 1274)
80 (’but’, 1269)
81 (’\xa0’, 1252)
82 (’2’, 1227)
83 (’if’, 1225)
84 (’they’, 1194)
85 (’can’, 1193)
86 (’2011’, 1148)
87 (’$’, 1143)
88 (’how’, 1125)
89 (’more’, 1117)
90 (’de’, 1097)
91 (’union’, 1057)
92 (’people’, 1048)
93 (’he’, 1025)
94 (’who’, 1020)
95 (’security’, 1014)
96 (’airport’, 1000)
97 (’love’, 994)
98 (’today’, 989)
99 (’or’, 986)
100 (’day’, 977)
```

## 1.4 token appeared only once

we read the Tokens.txt file in section 1.3, then counted the number of tokens appeared only once. the result shows we more than 70% token only appeared once.
tokens appeared only once: 68296 types fo tokens: 97011 tokens appeared only once/types fo tokens: 0.70400263887

## 1.5 frequency of words’ token

we use regular expression to filtered all punctuations and symbols, then listed all the words by frequency
listed top 100

```
1 (’the’, 21126)
2 (’to’, 13877)
3 (’in’, 10839)
4 (’of’, 10574)
5 (’a’, 10459)
6 (’i’, 8456)
7 (’and’, 8360)
8 (’for’, 7112)
9 (’on’, 6286)
10 (’is’, 6251)
11 (’rt’, 4452)
12 (’you’, 4099)
13 (’at’, 3909)
14 (’it’, 3834)
15 (’egypt’, 3269)
16 (’with’, 3145)
17 (’my’, 3122)
18 (’that’, 3023)
19 (’new’, 2978)
20 (’news’, 2717)
21 (’from’, 2472)
22 (’this’, 2439)
23 (’are’, 2429)
24 (’be’, 2271)
25 (’by’, 2052)
26 (’do’, 1979)
27 (’will’, 1937)
28 (’have’, 1936)
29 (’not’, 1923)
30 (’egyptian’, 1896)
31 (’your’, 1822)
32 (’obama’, 1814)
33 (’state’, 1802)
34 (’me’, 1789)
35 (’we’, 1747)
36 (’just’, 1746)
37 (’as’, 1682)
```

```
38 ('us', 1675)
39 ('out', 1647)
40 ('has', 1593)
41 ('all', 1569)
42 ('what', 1471)
43 ('no', 1464)
44 ('up', 1459)
45 ('now', 1444)
46 ('super', 1416)
47 ('world', 1412)
48 ('was', 1405)
49 ('so', 1388)
50 ('an', 1372)
51 ('social', 1332)
52 ('media', 1330)
53 ('like', 1315)
54 ('white', 1307)
55 ('via', 1295)
56 ('bowl', 1289)
57 ('get', 1287)
58 ('about', 1280)
59 ('but', 1269)
60 ('if', 1225)
61 ('they', 1194)
62 ('can', 1193)
63 ('how', 1125)
64 ('more', 1117)
65 ('de', 1097)
66 ('union', 1057)
67 ('people', 1048)
68 ('he', 1025)
69 ('who', 1020)
70 ('security', 1014)
71 ('airport', 1000)
72 ('love', 994)
73 ('today', 989)
74 ('or', 986)
75 ('day', 977)
76 ('president', 977)
77 ('u', 966)
78 ('release', 956)
79 ('law', 955)
80 ('one', 953)
81 ('time', 942)
82 ('his', 922)
83 ('good', 888)
84 ('video', 888)
85 ('house', 883)
86 ('mubarak', 873)
```

```
87 ('over', 863)
88 ('jobs', 857)
89 ('protests', 849)
90 ('when', 848)
91 ('show', 844)
92 ('service', 831)
93 ('our', 818)
94 ('cairo', 812)
95 ('got', 806)
96 ('go', 797)
97 ('job', 782)
98 ('lol', 776)
99 ('after', 757)
100 ('energy', 753)
```

### 1.6 frequency of words excluding stopwords

we read file in section 1.5, then filtered all stop words tokens.
listed top 100

```
1 ('rt', 4452)
2 ('egypt', 3269)
3 ('news', 2717)
4 ('egyptian', 1896)
5 ('obama', 1814)
6 ('state', 1802)
7 ('super', 1416)
8 ('world', 1412)
9 ('social', 1332)
10 ('media', 1330)
11 ('white', 1307)
12 ('bowl', 1289)
13 ('union', 1057)
14 ('people', 1048)
15 ('security', 1014)
16 ('airport', 1000)
17 ('love', 994)
18 ('today', 989)
19 ('president', 977)
20 ('release', 956)
21 ('law', 955)
22 ('video', 888)
23 ('house', 883)
24 ('mubarak', 873)
25 ('jobs', 857)
26 ('protests', 849)
27 ('service', 831)
28 ('cairo', 812)
29 ('job', 782)
30 ('lol', 776)
```

```
31 ('energy', 753)
32 ('says', 748)
33 ('phone', 747)
34 ('police', 726)
35 ('global', 713)
36 ('dog', 705)
37 ('free', 701)
38 ('back', 682)
39 ('bbc', 669)
40 ('taco', 667)
41 ('bell', 666)
42 ('protesters', 641)
43 ('return', 638)
44 ('live', 637)
45 ('rite', 626)
46 ('toyota', 624)
47 ('special', 614)
48 ('know', 601)
49 ('iran', 599)
50 ('ca', 564)
51 ('think', 562)
52 ('ap', 555)
53 ('health', 551)
54 ('court', 547)
55 ('twitter', 544)
56 ('man', 543)
57 ('crash', 534)
58 ('tv', 532)
59 ('cuts', 521)
60 ('post', 519)
61 ('budget', 517)
62 ('home', 514)
63 ('weather', 513)
64 ('watch', 510)
65 ('business', 501)
66 ('top', 493)
67 ('government', 481)
68 ('food', 476)
69 ('right', 472)
70 ('online', 470)
71 ('car', 461)
72 ('organic', 459)
73 ('tcot', 455)
74 ('blog', 451)
75 ('address', 447)
76 ('attack', 446)
77 ('peace', 445)
78 ('haiti', 434)
79 ('mexico', 428)
```

```
80 (’pakistan’, 426)
81 (’big’, 424)
82 (’help’, 422)
83 (’moscow’, 414)
84 (’museum’, 414)
85 (’protest’, 410)
86 (’check’, 404)
87 (’life’, 403)
88 (’date’, 396)
89 (’work’, 394)
90 (’jordan’, 394)
91 (’nt’, 392)
92 (’internet’, 387)
93 (’fifa’, 382)
94 (’recovery’, 381)
95 (’auto’, 381)
96 (’game’, 375)
97 (’call’, 374)
98 (’olbermann’, 374)
99 (’york’, 372)
100 (’tonight’, 372)
```

## 1.7 frequency of pairs

we stored all types of tokens in lower case (F.txt) of section 1.6, then we used nltk.bigrams to find all the pairs in the corpus, order by frequency, and filtered pairs if at least one element is not in the F.txt.
listed top 100

```
1 ((’super’, ’bowl’), 1199)
2 ((’social’, ’media’), 980)
3 ((’taco’, ’bell’), 579)
4 ((’white’, ’house’), 365)
5 ((’union’, ’address’), 354)
6 ((’global’, ’warming’), 311)
7 ((’keith’, ’olbermann’), 267)
8 ((’president’, ’obama’), 261)
9 ((’bowl’, ’xlv’), 209)
10 ((’world’, ’cup’), 202)
11 ((’white’, ’stripes’), 201)
12 ((’barack’, ’obama’), 191)
13 ((’rahm’, ’emanuel’), 188)
14 ((’moscow’, ’airport’), 187)
15 ((’bbc’, ’news’), 179)
16 ((’united’, ’states’), 174)
17 ((’health’, ’care’), 156)
18 ((’press’, ’release’), 144)
19 ((’budget’, ’cuts’), 143)
20 ((’julian’, ’assange’), 139)
21 ((’customer’, ’service’), 137)
```

```
22 ((’president’, ’barack’), 133)
23 ((’hosni’, ’mubarak’), 129)
24 ((’egypt’, ’protests’), 124)
25 ((’supreme’, ’court’), 122)
26 ((’tahrir’, ’square’), 121)
27 ((’youtube’, ’video’), 116)
28 ((’glenn’, ’beck’), 116)
29 ((’world’, ’news’), 112)
30 ((’egyptian’, ’protesters’), 109)
31 ((’birth’, ’certificate’), 109)
32 ((’egyptian’, ’museum’), 106)
33 ((’middle’, ’east’), 104)
34 ((’prime’, ’minister’), 103)
35 ((’release’, ’date’), 103)
36 ((’world’, ’service’), 102)
37 ((’blog’, ’post’), 102)
38 ((’hillary’, ’clinton’), 98)
39 ((’breaking’, ’news’), 95)
40 ((’egyptian’, ’protests’), 94)
41 ((’egyptian’, ’president’), 92)
42 ((’bbc’, ’world’), 90)
43 ((’media’, ’marketing’), 87)
44 ((’president’, ’hosni’), 87)
45 ((’federal’, ’judge’), 86)
46 ((’egyptian’, ’police’), 83)
47 ((’current’, ’tv’), 82)
48 ((’union’, ’speech’), 79)
49 ((’kate’, ’middleton’), 75)
50 ((’ca’, ’nt’), 71)
51 ((’security’, ’forces’), 71)
52 ((’egyptian’, ’people’), 71)
53 ((’egyptian’, ’government’), 70)
54 ((’airport’, ’security’), 69)
55 ((’lol’, ’rt’), 68)
56 ((’fox’, ’news’), 67)
57 ((’ai’, ’nt’), 66)
58 ((’domodedovo’, ’airport’), 66)
59 ((’phone’, ’hacking’), 66)
60 ((’egyptian’, ’embassy’), 66)
61 ((’care’, ’law’), 66)
62 ((’international’, ’airport’), 66)
63 ((’special’, ’olympics’), 65)
64 ((’egyptian’, ’army’), 64)
65 ((’tear’, ’gas’), 64)
66 ((’cowboys’, ’stadium’), 64)
67 ((’gabrielle’, ’giffords’), 64)
68 ((’global’, ’war’), 63)
69 ((’unemployment’, ’rate’), 63)
70 ((’state’, ’tv’), 61)
```

```
71 ((’anthony’, ’hopkins’), 61)
72 ((’cell’, ’phone’), 59)
73 ((’social’, ’networking’), 58)
74 ((’climate’, ’change’), 58)
75 ((’shorty’, ’award’), 58)
76 ((’weight’, ’loss’), 57)
77 ((’green’, ’bay’), 57)
78 ((’fifa’, ’soccer’), 57)
79 ((’judge’, ’rules’), 57)
80 ((’state’, ’hillary’), 57)
81 ((’oprah’, ’winfrey’), 56)
82 ((’olympic’, ’stadium’), 56)
83 ((’justin’, ’bieber’), 54)
84 ((’mayoral’, ’ballot’), 54)
85 ((’fifa’, ’world’), 53)
86 ((’louis’, ’vuitton’), 53)
87 ((’toyota’, ’recalls’), 52)
88 ((’chicago’, ’mayoral’), 52)
89 ((’illinois’, ’supreme’), 52)
90 ((’egyptian’, ’state’), 50)
91 ((’strings’, ’attached’), 50)
92 ((’car’, ’crash’), 50)
93 ((’dog’, ’training’), 50)
94 ((’muslim’, ’brotherhood’), 49)
95 ((’court’, ’rules’), 49)
96 ((’organic’, ’food’), 48)
97 ((’york’, ’city’), 47)
98 ((’law’, ’firm’), 47)
99 ((’egypt’, ’museum’), 47)
100 ((’detroit’, ’police’), 47)
```

## 2 Part 2: Evaluation word embeddings

In this section, we evaluate 8 different word embedding methods over 7 similarity task datasets and 4 analogy questions task datasets. We consider pre-trained CBOW, Skip-grams, GloVe, PDC, HDC, LexVec, ConceptNet Numberbatch Numberbatch, FastText. We use similarity test data set MTurk, MEN, WS353, Rubenstein and Goodenough, Rare Words, SimLex999, TR9856. We also use analogy questions task datasets MSR_WordRep, Google_analogy, MSR, SEMEVAL_2012_Task 2.

In the experiment, there are two different kind of tasks: semantic similarity test and Word analogy. In word semantic similarity test, given a pair of words, the distance between the two word vectors is calculated and the distance is compared with a human rated similarity. The goodness of fit is calculated by Spearman correlation. The word analogy test, on the other hand, is given a combination of word (a, a*, b), predicting the word ’b*’ that has the similar relationship to ’b’ as the relation of ’a*’ and ’a’. We use a SimpleAnalogySolver given by [word-embeddings-benchmarks] to predict the word b* for the task, and calculate the accuracy of the prediction. In both of the tasks, we calculate a score of the task, either correlation or accuracy. The higher the score, the better the embedding model is.

### 2.1 CBOW evaluation

CBOW is part of the unsupervised Word2Vec proposed by Google. The intuition behind CBOW(Continuous Bag of Words) is using the context to predict a target word. Given the context of a word as the input, it predict the word using a paramerized neural network without activation function. We use CBOW pretrained with embedding dimension 300, window size 15, minimum count 10, and iterations 20, where window size specify the size of the context, and minimum count specify the minimum times a word needs to be seen. The model is trained on etTenTen: Corpus of the Estonian Web. The result of evaluation on similarity test and analogy task is given in table 1 and table 2.

| Mturk | MEN | WS353 | RG65 |
|---|---|---|---|
| 0.236008 | 0.266661 | 0.22192 | 0.173106 |
| RW | SIMLEX999 | TR9856 | mean |
| 0.208761 | 0.08837 | 0.118343 | 0.187596 |

Table 1: Similarity result of CBOW

| | MSR WordRep | Google analogy | MSR | SEMEVAL 2012 Task 2 | mean |
|---|---|---|---|---|---|
| CBOW | 0.005009 | 0.039194 | 0.01 | 0.026911 | 0.020278 |

Table 2: Analogy tasks on CBOW

### 2.2 Skip-grams evaluation

Skip-gram is similar to the CBOW, with the difference that instead of predict target word from context, the model predict the context from the target word. We use pretrained Skip-grams from word2vec. The model is trained on Google News dataset with embedding dimension 300. The result of evaluation on similarity test and analogy task is given in table 3 and table 4.

| Mturk | MEN | WS353 | RG65 |
|---|---|---|---|
| 0.681506 | 0.758511 | 0.700017 | 0.760783 |
| RW | SIMLEX999 | TR9856 | mean |
| 0.497048 | 0.441966 | 0.178644 | 0.574068 |

Table 3: Similarity result of Skip-gram

| | MSR WordRep | Google analogy | MSR | SEMEVAL 2012 Task 2 | mean |
|---|---|---|---|---|---|
| CBOW | 0.19 | 0.401811 | 0.711875 | 0.20406 | 0.376937 |

Table 4: Analogy tasks on Skip-grams

### 2.3 GloVe evaluation

Unlike Word2vec, GloVe conbines both local context relation and global concurrence statistics by calculate a co-occurrence probability matrix. We use pretrained GloVe from GloVe. We also use 300 dimensional embedding, the word embedding was trained on Wikipedia 2014 +

Gigaword, and contains 400K words. The result of evaluation on similarity test and analogy task is given in table 5 and table 6.

| Mturk | MEN | WS353 | RG65 |
|---|---|---|---|
| 0.633182 | 0.737465 | 0.543426 | 0.769525 |
| RW | SIMLEX999 | TR9856 | mean |
| 0.366982 | 0.3705 | 0.098367 | 0.502778 |

Table 5: Similarity result of GloVe

| | MSR WordRep | Google analogy | MSR | SEMEVAL 2012 Task 2 | mean |
|---|---|---|---|---|---|
| CBOW | 0.22836 | 0.717356 | 0.61425 | 0.163963 | 0.430982 |

Table 6: Analogy tasks on GloVe

## 2.4 PDC evaluation

PDC and HDC are two distributional word embedding models that take both syntagmatic (words that co-occur in the same context) and paradigmatic relation(words that occurs with similar contexts, but not necessarily the same text) into consideration. They both learned from information of both text region and surrounding words. More specifically, PDC has a CBOW like structure, but adding an extra document branch to incorporate text region information. In the experiment, we use pretrained PDC and HDC from http://ofey.me/projects/wordrep. The model was trained on Wikipedia 2010, and is 300 dimensional. The result of evaluation on similarity test and analogy task is given in table 7 and table 8.

| Mturk | MEN | WS353 | RG65 |
|---|---|---|---|
| 0.672333 | 0.772648 | 0.733431 | 0.790069 |
| RW | SIMLEX999 | TR9856 | mean |
| 0.472393 | 0.426882 | 0.207014 | 0.58211 |

Table 7: Similarity result of PDC

| | MSR WordRep | Google analogy | MSR | SEMEVAL 2012 Task 2 | mean |
|---|---|---|---|---|---|
| CBOW | 0.252121 | 0.747595 | 0.596375 | 0.174074 | 0.442541 |

Table 8: Analogy tasks on PDC

## 2.5 HDC evaluation

Similar to PDC, HDC considers both syntagmatic and paradigmatic relations, but it has a structure that is similar to Skip-gram. The model was trained on Wikipedia 2010, and is 300 dimensional. The result of evaluation on similarity test and analogy task is given in table 9 and table 10.

| Mturk | MEN | WS353 | RG65 |
|---|---|---|---|
| 0.65767 | 0.760335 | 0.716873 | 0.805805 |
| RW | SIMLEX999 | TR9856 | mean |
| 0.463447 | 0.406832 | 0.207092 | 0.574008 |

Table 9: Similarity result of HDC

| | MSR WordRep | Google analogy | MSR | SEMEVAL 2012 Task 2 | mean |
|---|---|---|---|---|---|
| CBOW | 0.250847 | 0.731273 | 0.564375 | 0.184511 | 0.432752 |

Table 10: Analogy tasks on HDC

## 2.6 LexVec evaluation

LexVec embedding model is another model that is similar to GloVe which achieves state of the art results in multiple NLP tasks. The model use low-rank, weighted factorization of the Positive Point-wise Mutual Information matrix (PPMI). PPMI matrix is generally better in semantic tasks than PMI. Unlike SVD, factorizing the PPMI matrix using reconstruction loss function does not weight all errors equally, but penalizes errors of frequent co-occurrences more heavily. The embedding is download from LexVec. The model is trained on enwiki+newscrawl, and is 300 dimensional embedding. The result of evaluation on similarity test and analogy task is given in table 11 and table 12. LexVec different with SVD,

| Mturk | MEN | WS353 | RG65 |
|---|---|---|---|
| 0.655411 | 0.751388 | 0.621565 | 0.747058 |
| RW | SIMLEX999 | TR9856 | mean |
| 0.45623 | 0.385223 | 0.14688 | 0.537679 |

Table 11: Similarity result of LexVec

| | MSR WordRep | Google analogy | MSR | SEMEVAL 2012 Task 2 | mean |
|---|---|---|---|---|---|
| CBOW | 0.252822 | 0.728305 | 0.57375 | 0.198185 | 0.438265 |

Table 12: Analogy tasks on LexVec

## 2.7 ConceptNet Numberbatch evaluation

ConceptNet is a knowledge graph that connects word and phrases with weighted edges. It is originally used to parse corpus, but also represents links between knowledges, which is ideal to word embedding. The ConceptNet Numberbatch use a generalization of the retrofitting method to combine the ConceptNet and distributuional semantics. Numberbatch is built using an ensemble that combines data from ConceptNet, word2vec, GloVe, and OpenSubtitles 2016. Retrofitting infers new vectors with the objective of being close to their original values and also close to their neighbors in the graph with edges E. We use pretrained model from ConceptNet Numberbatch, the model is 300 dimensional. The result of evaluation on similarity test and analogy task is given in table 13 and table 14.

| Mturk | MEN | WS353 | RG65 |
|---|---|---|---|
| 0.719718 | 0.859638 | 0.754611 | 0.90988 |
| RW | SIMLEX999 | TR9856 | mean |
| 0.545442 | 0.650525 | 0.130053 | 0.652838 |

Table 13: Similarity result of ConceptNet Numberbatch

| | MSR WordRep | Google analogy | MSR | SEMEVAL 2012 Task 2 | mean |
|---|---|---|---|---|---|
| CBOW | 0.159396 | 0.381242 | 0.539375 | 0.238102 | 0.329529 |

Table 14: Analogy tasks on ConceptNet Numberbatch

### 2.8 fastText evaluation

fastText is a library for efficient learning of word representations and sentence classification. The English pretrained model is download from fastText, which contains 1M words and is 300 dimensional. The result of evaluation on similarity test and analogy task is given in table 15 and table 16.

| Mturk | MEN | WS353 | RG65 |
|---|---|---|---|
| 0.702549 | 0.790632 | 0.733276 | 0.846259 |
| RW | SIMLEX999 | TR9856 | mean |
| 0.513408 | 0.449965 | 0.157249 | 0.599048 |

Table 15: Similarity result of FastText

| | MSR WordRep | Google analogy | MSR | SEMEVAL 2012 Task 2 | mean |
|---|---|---|---|---|---|
| CBOW | 0.274761 | 0.592509 | 0.813 | 0.219978 | 0.475062 |

Table 16: Analogy tasks on FastText

### 2.9 Overall comparison

In this part, we give a table of all models and their results on all the evaluation datasets. We also calculate the average score over all the datasets for each of the model. Table 17 gives the results.

The result shows that comparing to baseline models like Word2Vec and GloVe, novel and latter methods like ConceptNet Numberbatch and FastText got higher score. Another observation is that while CBOW perform the worst in all task, its performance on words analogy task is far more worse than similarity task. Finally, although PDC and HDC has similar structure to Word2Vec, where PDC has similar structure to CBOW, their score are close, and better than Word2Vec methods.

|  | Mturk | MEN | WS353 | RG65 | RW | SIMLEX999 |
|---|---|---|---|---|---|---|
| CBOW | 0.236 | 0.267 | 0.222 | 0.173 | 0.209 | 0.088 |
| Skip-gram | 0.682 | 0.759 | 0.700 | 0.761 | 0.497 | 0.442 |
| GloVe | 0.633 | 0.737 | 0.543 | 0.770 | 0.367 | 0.371 |
| PDC | 0.672 | 0.773 | 0.733 | 0.790 | 0.472 | 0.427 |
| HDC | 0.658 | 0.760 | 0.717 | 0.806 | 0.463 | 0.407 |
| LexVec | 0.655 | 0.751 | 0.622 | 0.747 | 0.456 | 0.385 |
| ConceptNet Numberbatch | **0.720** | **0.860** | **0.755** | **0.910** | **0.545** | **0.651** |
| FastText | 0.703 | 0.791 | 0.733 | 0.846 | 0.513 | 0.450 |
|  | TR9856 | MSR WordRep | Google analogy | MSR | SEMEVAL 2012 Task 2 | mean |
| CBOW | 0.118 | 0.005 | 0.039 | 0.010 | 0.027 | 0.127 |
| Skip-gram | 0.179 | 0.190 | 0.402 | 0.712 | 0.204 | 0.502 |
| GloVe | 0.098 | 0.228 | 0.717 | 0.614 | 0.164 | 0.477 |
| PDC | **0.207** | 0.252 | **0.748** | 0.596 | 0.174 | 0.531 |
| HDC | **0.207** | 0.251 | 0.731 | 0.564 | 0.185 | 0.523 |
| LexVec | 0.147 | 0.253 | 0.728 | 0.574 | 0.198 | 0.502 |
| ConceptNet Numberbatch | 0.130 | 0.159 | 0.381 | 0.539 | **0.238** | 0.535 |
| FastText | 0.157 | **0.275** | 0.593 | **0.813** | 0.220 | **0.554** |

Table 17: Comparison of models