# Word Embedding by Using Word2Vec and BERT

Github link: https://github.com/Allen-ZKW/NLP\_HZAU/tree/task7

#### **Abstract**

In this task, we use two algorithms to complete Word Embedding. This two algorithms all try to transform words to vectors, these vectors will reflect the closeness and similarity between two different words

## **Principle**

Both of these two algorithms' target is translating word to type of information which can be understand by computers and programs. We can use 'translate data' to calculate distance between two different words. BERT algorithm takes word in different context into concern, so the result of BERT will include more information. This model is trained by forwards information and afterwards information.

### Measure

## **Data Preparation**

unzip data/litcovid-trainingdata.zip

## **Environmental Preparation**

```
conda create -n NLP_task7 python=3.8

conda activate NLP_task7

conda install pytorch torchvision torchaudio cpuonly -c pytorch

conda install --yes --file requirements.txt
```

#### Word2Vec

```
python3 Skip_Gram_basic.py
```

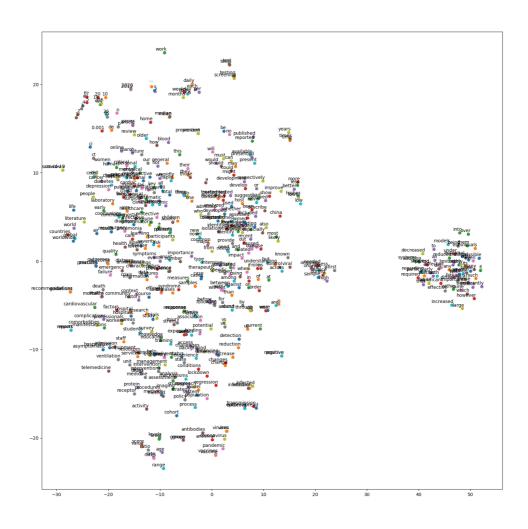
#### **BERT**

```
python3 Bert_4_Litcovid_WordEmbedding.py
```

## Result

#### **BERT Vectors**

#### **BERT Visualization**



#### Word2Vec Vectors

|UNK| -0.04881330579519272 0.07739069312810898 -0.05652253329753876 -0.09073057770729065 -0.11317484825849533 0.058399297297000885 0.022058576345443726 0.04351604357 41718006134 -0.011797694489359856 0.10310874134302139 0.011879590339958668 0.11605213582515717 -0.006655462551862001 -0.001093752682209015 -0.04330721125006676 -0.08714

0-14333058893680573 -0.1587565541267395 0.14329899847507477 -0.3877052068710327 -0.10853473842144012 -0.030758662149310112 0.04662489891052246 -0.024330385029 78235 -0.04327136278152466 -0.07235678285360336 -0.3090354800224304 0.12058231979608536 -0.020435530692338943 0.10384968668222427 0.28311672806739807 0.135265320539474

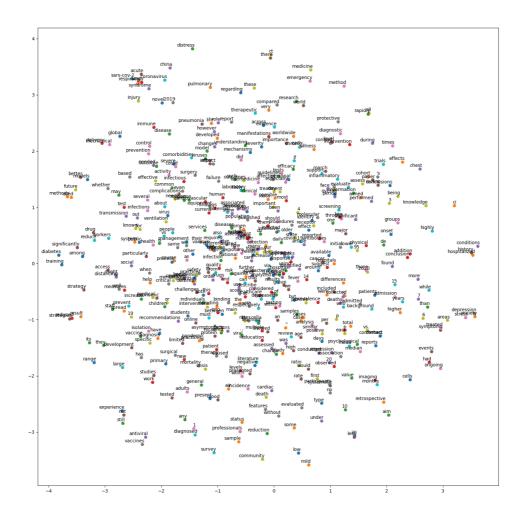
0.03092149645090103 0.06394444406032562 0.22052952647209167 0.12312621623277664 0.13519074022769928 -0.33134761452674866 -0.08193377405405045 0.17148441076278 -10996109992265701 0.08987542241811752 -0.06514536589384079 0.0856313705444336 -0.06625906378030777 0.11863439530134201 -0.1884736567735672 -0.07212437689304352 0.09053

and -0.022921787574887276 -0.27186140418052673 -0.13821113109588623 0.2388809472322464 -0.006997520104050636 0.015737395733594894 -0.2610098421573639 -0.0031904140 710476875305 0.14956840872764587 -0.26795533299446106 -0.29544875025749207 0.18868643045425415 -0.15007612109184265 0.13892631232738495 -0.015536155551671982 0.1629645 803672458976507

were 0.5449550747871399 0.12638379633426666 0.03735343739390373 0.08160431683063507 -0.11231604963541031 0.1520289033651352 -0.36640504002571106 0.1093474701046943 883780777454376 -0.6140636205673218 -0.09264766424894333 0.023603010922670364 -0.13203932344913483 0.09088529646396637 0.24826601147651672 0.026199063286185265 0.11704 2937

5 -0.2018497884273529 -0.014681762084364891 -0.27701535820961 -0.3487420082092285 0.5759435892105103 0.1453063189983368 0.08596295118331909 -0.42484018206596375 (2977199554443 0.04105306416749954 -0.10346832126379013 0.026820257306098938 -0.1885482668876648 0.4364898204803467 -0.06917911767959595 -0.06044343486428261 0.52001386

#### Word2Vec Visualization



In these two graphs, we can find that in BERT result word mainly separate in two parts while Word2Vec result stay in one part. Also, in BERT result, there exits some nodes which stay very close. Although, Word2Vec result's nodes all exist in one part, the distance between two nodes is longer than BERT model. BERT result is more obvious, which may be one reason of playing a better role in word embedding.