
lightnlp-cookbook

发布 0.3.3

2020 年 12 月 30 日

1	前言	1
2	声明	3
3	功能	5
3.1	序列标注, sl	5
3.2	结构分析, sp	5
3.3	句子关系, sr	5
3.4	文本分类, tc	6
3.5	文本生成, tg	6
3.6	词向量, we	6
4	安装	7
4.1	安装依赖	7
5	模型	9
6	数据	11
6.1	embedding	11
6.2	corpus	11
7	各模块使用教程	13
7.1	词向量, we	13
7.2	文本分类, tc	26
7.3	序列标注, sl	28
7.4	文本生成, tg	38
7.5	句子关系, sr	45
7.6	结构分析, sp	47
7.7	工具模块	51

8 架构	53
8.1 项目架构	53
8.2 架构说明	54
9 参考	57
9.1 Deep Learning	57
9.2 NLP	57
9.3 知识图谱	57
9.4 Pytorch 教程	58
9.5 torchtext 介绍	58
9.6 其他工具模块	58
9.7 词向量	58
9.8 数据集	59
9.9 序列标注	59
9.10 文本分类	59
9.11 命名实体识别	59
9.12 关系抽取	59
9.13 事件抽取	59
9.14 语言模型	60
9.15 文本生成	60
9.16 语句相似度	60
9.17 文本蕴含	60
9.18 中文分词	60
9.19 词性标注	60
9.20 指代消解	61
9.21 依存句法分析	61
9.22 自动摘要	61
9.23 阅读理解	61
9.24 聊天机器人	61
9.25 其他	62
10 Indices and tables	63

依据自然语言处理四大任务等，框架主要设计为有以下五大功能：

- 序列标注, Sequence Labeling
- 文本分类, Text Classification
- 句子关系, Sentence Relation
- 文本生成, Text Generation
- 结构分析, Structure Parsing

因此将有五个主要的功能模块：sl（序列标注）、tc（文本分类）、sr（句子关系）、tg（文本生成）、sp（结构分析）和其他功能模块如 we（词向量）。

声明

1. 本项目从定位上来说只是一次收集和尝试，目的并不作为企业级和生产级使用，目标群体主要是对自然语言处理各任务实践感兴趣的其他方向计算机开发人员以及初学者，更主要的是自娱自乐。
2. 若有真正场景需求的用户或开发人员们可以参考[Industry 中文 NLP 商业服务](#)寻求商业服务，当然本渣也乐意提供有偿服务。
3. 对 Pytorch 和自然语言处理有一定了解的童鞋们，如果想追求快速开发且自由定制 nlp 应用，可以考虑复旦大学 nlp 实验室开源的[fastNLP](#)，框架功能丰富，且简洁易用。
4. 本项目并未像其他一些框架一样为各任务提供了一些训练数据和训练好的模型从而可以直接下载使用。
5. 本项目的诸多模型许多都是参考 Github 上原有实现，然后在基础之上二次加工而成，在这里要向相关作者致以诚挚的谢意！
6. 未对各任务模型的各种参数进行精细微调，仅仅只是能跑通的程度。
7. 本项目仅在以下本人两种开发环境下能跑通，其他环境出现的各种问题本人概不负责。
 - Windows 10, Python 3.6, Pytorch 1.3
 - Manjaro, Python 3.7, Pytorch 1.3

3.1 序列标注, sl

- 中文分词, cws
- 命名实体识别, ner
- 词性标注, pos
- 语义角色标注, srl

3.2 结构分析, sp

- 基于图的依存句法分析, gdp
- 基于转移的依存句法分析, tdp

3.3 句子关系, sr

- 语句相似度, ss
- 文本蕴含, te

3.4 文本分类, tc

- 关系抽取, re
- 情感极性分析, sa

3.5 文本生成, tg

- 语言模型, lm
- 聊天机器人, cb
- 机器翻译, mt
- 文本摘要, ts

3.6 词向量, we

- 词袋模型, cbow
 - base
 - hierarchical_softmax
 - negative_sampling
- 跳字模型, skip_gram
 - base
 - hierarchical_softmax
 - negative_sampling

```
pip install lightNLP
```

建议使用国内源来安装，如使用以下命令：

```
pip install -i https://pypi.douban.com/simple/ lightNLP
```

4.1 安装依赖

由于有些库如 `pytorch`、`torchtext` 并不在 `pypi` 源中或者里面只有比较老旧的版本，我们需要单独安装一些库。

4.1.1 安装 `pytorch`

请使用最新版本的 Pytorch！

具体安装参见[pytorch 官网](#)来根据平台、安装方式、Python 版本、CUDA 版本来选择适合自己的版本。

4.1.2 安装 `torchtext`

使用以下命令安装最新版本 `torchtext`：

```
pip install https://github.com/pytorch/text/archive/master.zip
```

- ner: BiLstm-Crf
- cws: BiLstm-Crf
- pos: BiLstm-Crf
- srl:BiLstm-Crf
- sa: TextCnn
- re: TextCnn, 当前这里只是有监督关系抽取
- lm: Lstm, 基础的 LSTM, 没有使用 Seq2Seq 模型
- ss: 共享 LSTM + 曼哈顿距离
- te: 共享 LSTM + 全连接
- tdp: lstm + mlp + shift-reduce(移入规约)
- gdp: lstm + mlp + biaffine (双仿射)
- cbow: base、hierarchical_softmax、negative_sampling
- skip_gram: base、hierarchical_softmax、negative_sampling
- cb: Seq2Seq+Attention
- mt: Seq2Seq+Attention
- ts: Seq2Seq+Attention

6.1 embedding

6.1.1 1. 中文词向量

- Chinese-Word-Vectors
- fastText/Chinese

6.1.2 2. 中文字向量

- ChineseEmbedding

6.1.3 3. 英文字向量

- fastText/English
- GloVe

6.2 corpus

7.1 词向量, we

7.1.1 词袋模型, cbow

1. 训练数据

就普通的文本格式

训练数据示例如下:

第一章 陨落的天才

“斗之力，三段！”

望着测验魔石碑上面闪亮得甚至有些刺眼的五个大字，少年面无表情，唇角有着一抹自嘲，紧握手掌，因为大力，而导致略微尖锐的指甲深深的刺进了掌心之中，带来一阵阵钻心的疼痛……

“萧炎，斗之力，三段！级别：低级！”测验魔石碑之旁，一位中年男子，看了一眼碑上所显示出来的信息，语气漠然的将之公布了出来……

中年男子话刚刚脱口，便是不出意外的在人头汹涌的广场上带起了一阵嘲讽的骚动。

“三段？嘿嘿，果然不出我所料，这个“天才”这一年又是在原地踏步！”

“哎，这废物真是把家族的脸都给丢光了。”

“要不是族长是他的父亲，这种废物，早就被驱赶出家族，任其自生自灭了，哪还有机会待在家族中白吃白喝。”

“唉，昔年那名闻乌坦城的天才少年，如今怎么落魄成这般模样了啊？”

2. 使用示例

CBOW 共实现了三种模型，分别为基础 softmax 模型 (CBOWBaseModule)、基于负采样的优化模型 (CBOWNegativeSamplingModule)、基于层次 softmax 的优化模型 (CBOWHierarchicalSoftmaxModule)。

三种模型提供的接口一致，如下所示：

导入接口

```
from lightnlp.we import CBOWHierarchicalSoftmaxModule, CBOWNegativeSamplingModule,   
↳CBOWBaseModule # 分别导入 CBOW 的不同模型
```

训练

```
# cbow_model = CBOWHierarchicalSoftmaxModule()  
# cbow_model = CBOWBaseModule()  
cbow_model = CBOWNegativeSamplingModule()  
  
train_path = '/home/lightsmile/NLP/corpus/novel/test.txt'  
dev_path = '/home/lightsmile/NLP/corpus/novel/test.txt'  
  
cbow_model.train(train_path, dev_path=dev_path, save_path='./cbow_saves')
```

测试

```
cbow_model.load('./cbow_saves')  
cbow_model.test(dev_path)
```

预测

```
test_context = ['族长', '是', '的', '父亲']  
print(cbow_model.evaluate(test_context, '他'))  
print(cbow_model.evaluate(test_context, '提防'))
```

预测结果：

```
0.9992720484733582  
2.4813576079191313e-30
```

保存词向量

```
cbow_model.save_embeddings('./cbow_saves/cbow_ns.bin')
```

./cbow_saves/cbow_ns.bin 文件内容:

623 300

```
<unk> -0.69455165 -1.3275498 -1.1975913 -0.3417502 0.13073823 1.3608844 0.15316872 -2.
↪295731 0.45459792 0.09420798 -0.73944765 0.11755463 -1.6275359 0.6623806 0.8247673 1.
↪7149547 -0.49345177 -0.5932094 -1.3025115 0.40126365 1.8675354 0.46296182 0.81418717 -
↪0.51671696 -1.328723 -0.27371547 -1.5537426 1.0326972 0.11647574 0.1607528 0.5110576 -
↪1.2010366 -0.81535685 0.5231469 2.212553 0.43934354 -0.8626878 1.5049676 -0.8135034 -0.
↪8322859 0.068298176 0.7376674 0.6459309 0.07635216 -0.77374196 0.29933965 1.6596211 0.
↪46682465 -0.8282705 -0.22142725 1.7853647 1.4777366 -0.63895816 2.1443112 -2.2435715 0.
↪85962945 1.6643075 1.082537 -0.6922347 -2.2418396 -0.20210272 -1.2102528 -0.48685002 0.
↪65887684 -0.2534356 -1.0342008 -1.1101105 0.94670665 0.21063486 -0.2467249 0.16507177
↪0.61120677 0.27850544 -1.0511587 -0.9382702 -0.105773546 -1.2759126 0.77076215 1.
↪6730801 0.7634321 0.22365877 -1.7401465 -1.6434158 0.94023687 -1.3609751 -2.153141 0.
↪3826534 0.32158422 -2.4204254 -2.1351569 -0.7265906 1.2896249 -1.6444998 0.62701744 3.
↪9122646e-05 -1.348553 1.6431069 0.4589956 -1.8367497 0.81131816 0.13370599 0.9231004 -
↪0.2677846 0.22468318 0.10889411 -1.0416583 0.016111592 -0.36729148 0.24761267 -1.
↪143464 -0.6162608 -0.6412186 0.79434645 -0.11785016 1.8588868 -0.06067674 -1.1092484 -
↪0.039183926 -0.5137064 -0.15945728 -1.4222018 0.31517547 -0.81327593 0.0048671463 -0.
↪18886662 0.28870773 1.0241542 0.24846096 0.15484594 0.83580816 -0.59276813 0.12078259 -
↪0.2424585 -0.1992609 -1.7673252 -0.45719153 0.3185026 0.052791957 0.072982006 0.
↪27393457 0.24782388 -1.073425 0.2915962 -0.52252334 -0.0066470583 -0.4599936 0.
↪34365907 0.7187273 -0.7599531 -0.5792492 1.1238049 0.8469614 -0.078110866 0.20481071 -
↪0.015566204 0.39713895 0.27844605 -0.37874687 -0.32269904 0.18351592 -1.2942557 1.
↪0065168 2.6649168 -0.09024592 -0.115473986 -0.29874867 0.5950803 -0.6491804 0.9974534 -
↪1.0031152 -2.4024782 -0.11324192 0.3452371 -0.68466026 -0.7123374 -0.61712 -2.0060632
↪0.49333447 0.4248587 -0.05601518 0.099164896 1.8789287 -0.2811404 0.91072047 2.713236
↪1.3424015 -0.007254917 -1.2505476 -0.7478102 0.7299547 -0.089441456 -0.43519676 0.
↪45425606 0.49322376 -1.0130681 -0.56024987 -0.74189216 0.5030309 -1.023638 -1.7686493
↪0.638495 0.612898 0.5948498 2.5866709 0.1675552 -0.059030745 -0.3356758 0.66674125 1.
↪1920244 0.24162059 1.3198696 0.28690717 -2.68874 -0.48055518 -1.5761619 0.14664873 0.
↪83967185 -0.7924626 0.7860132 -0.7246394 1.0014578 0.14658897 -0.64450735 0.86360186 2.
↪015226 -0.06311106 0.54246426 -2.120671 0.60732156 -0.9577766 -0.962489 -0.13819228 -1.
↪9003396 1.477142 0.13473822 -1.3756094 0.21764572 0.71171355 0.03748321 -0.393383 0.
↪011907921 0.5097328 -0.710836 0.8421267 -0.89845014 -0.31148115 -0.12334009 -0.
↪58898896 0.35046947 0.26125875 1.1667713 -0.77842957 -0.5580311 0.7409664 -1.3743105 -
↪0.8576632 0.8552787 -0.70344007 -0.86729395 0.8507328 0.081006676 -0.36887273 0.
↪93737006 -0.8049869 -1.1607035 -1.4482615 -0.4097167 0.45684943 -0.71613914 0.41646683
↪2.408504 -0.29688725 -0.45588523 -2.1563365 0.6449749 0.06401941 -0.5306914 1.9065568 -
7.1.1.8词向量, 2w↪2589759 -0.11746242 -2.0128748 0.85062236 1.7910615 -0.23783809 0.22933501 0.8359954 -
↪0.16953708 0.711695 -0.13198276 1.3160635 0.48212075 -0.83564043
```

(续上页)

```

<pad> 0.8598764 -0.8392776 0.21543029 1.0473262 -0.35116714 0.92687714 0.19446017 0.
→43463743 -0.50851333 -1.5483292 0.4361628 -0.05452338 -0.26497063 0.66488725 -0.
→55493516 -0.2797728 1.2510214 -0.65309256 1.1241713 0.41626474 1.9894124 -0.51694274 1.
→5471387 1.0384578 0.2893607 0.8567941 -0.2927318 0.24968228 0.7357801 0.01763151 -0.
→46739513 -1.3317417 -0.36859253 -0.9243944 -0.35533777 -1.6850173 -0.23949681 1.
→8554561 0.68137765 0.7045612 1.2475091 -1.6330634 -0.052583996 -1.7476727 -0.692077 0.
→7417215 0.12882428 -1.0369412 -0.84594417 -0.2566721 0.34262887 -1.07697 -0.61600417 -
→0.15071104 -0.44881743 -0.7726476 1.7515095 0.20912598 0.70576566 -0.36712894 -0.
→31342962 0.47315833 -1.1460096 -0.70875674 -0.4837299 1.4506056 -0.9727428 0.39702946
→0.07864575 0.3648432 0.49154198 0.020293105 -0.7249207 0.97864133 1.4640467 0.5678606 -
→2.860407 -0.39765677 -0.4860878 0.8766392 0.84922194 0.41535607 0.87215734 0.28720066 -
→0.7825528 0.5715837 0.15444374 0.76095456 -1.0340949 1.3190961 0.34591895 1.2966202 -0.
→8545642 0.9938145 0.1409012 0.99152505 0.8077086 0.93903935 -0.6754034 -0.91347355 -1.
→8044235 -0.7238192 0.2459109 0.15390426 0.1533081 -1.2125725 -0.854381 0.49695554 -1.
→7440581 -0.64858806 -1.2289644 0.5474777 0.9272567 0.22399819 -0.034679767 2.3584945 0.
→07103437 0.81011516 0.0698216 0.3754226 -0.65767145 0.3823659 0.40215418 -1.707603 0.
→114939004 0.8273572 0.29516712 -0.6673007 -1.2765539 0.99865556 -1.2278188 0.03912367 -
→0.45458874 -1.0813018 -2.2441347 1.9152719 0.47215146 -0.12260598 -0.26454082 0.
→35173896 1.6129894 0.97668684 -1.8338121 -1.1014528 0.6723529 -0.45019576 0.6598951 -0.
→69084466 -0.10172084 -1.8603181 -1.6612647 -0.7758482 0.8601411 0.6049721 -0.29201725 -
→0.9079055 -0.34003752 0.66082954 -0.41279477 -0.33470514 -0.49652928 0.25946292 -1.
→3803854 0.65220726 -1.4215298 0.40058938 0.049067397 1.6812779 -0.27791974 1.7441406 -
→2.3301284 1.2588984 0.83174706 1.2724131 0.32724786 -1.653587 -0.79792064 1.051248 -0.
→58498347 0.28445363 -1.2115283 1.108874 0.52255243 0.9853287 1.4537731 0.904213 1.
→1746532 -1.1101269 -0.2703188 -0.6313266 0.69475996 -0.18485409 -0.57447076 -1.6579882
→1.2468975 -0.39891937 -1.4791157 0.8945784 0.33060122 1.0275787 -2.3348236 -0.90038484
→-1.3821996 0.5423107 -0.6897772 0.61041445 -0.574857 1.2986363 -1.5685147 -0.71202 -2.
→6498976 0.75422263 -0.37448043 -0.2572616 0.5239151 0.8996191 -0.33151335 1.7309458 -0.
→73092127 0.36491084 0.16062969 -0.23153275 0.24280524 -0.773348 1.0458037 -0.6981066 -
→1.5083469 -0.8071363 -0.1494729 0.3972236 -0.88379115 0.20430249 -1.1207113 -0.9375089
→-0.12876953 1.4187068 1.8777137 -1.999467 -1.9011496 0.4638691 -0.15722306 -1.509574 0.
→051803187 0.6853142 -1.0125363 -0.99807036 -0.86616534 -0.32387426 0.97010213 1.
→0255684 1.4593514 -0.36234704 -0.21524686 -1.7589426 0.66719395 0.70087874 0.95069945
→0.6235363 0.14841044 0.27994245 0.13287897 -0.44436157 0.7895685 1.2041568 -0.47667173
→-1.4123715 1.0322057 -1.709688 -1.225889 0.08815727 -0.6686178 -0.7308128 0.7389635 0.
→17666328 1.5924493 1.3784972 0.6649754 1.31653 0.9976657 -1.3411351 -0.05105546 -0.
→887594 0.67946136 1.041635 0.43628508 0.048369333 0.19013812 0.8495835 -0.08113135 -0.
→32964498 0.59289676 -0.11091884 1.1329387 1.3676411 1.5922078 0.09468127 1.1554819 1.
→0879983 -0.939253 0.72018343
, 0.8955515 0.17006782 -1.0863748 2.0142775 0.14233534 1.0502641 -1.9146186 1.5254054 0.
→41852686 -1.0021765 0.78738636 -1.1434265 -1.15919 1.3279808 -1.2685264 1.046601
→8198309 -0.37393337 0.5671053 -1.6003635 1.3942565 -0.37112692 -0.83049476 0.7837918 -
0.82138366 1.5960232 -0.5573124 -1.2436191 -1.428412 -1.8232468 0.6043092
→1.5128951 0.05398989 -0.7654913 -0.012385335 -0.48144546 1.1542314 -0.37977073 0.
→5381807 -0.25640526 -1.974048 1.2697856 -0.117085345 1.1256135 -1.0347183 1.5650568 0.

```

(下页继续)

(续上页)

的 -0.019410107 -0.24678797 -0.5141552 2.7299752 0.6342168 -0.110809356 0.2703856 0.
 ↪41705674 -0.76466995 -2.4204311 -0.59976536 -0.7159314 -0.8618017 1.0497526 0.54623944
 ↪-0.7981596 -0.67481875 1.0958283 -0.46740645 1.0951735 0.61883473 -1.0565901 -0.
 ↪32493624 -0.31894302 -1.8763341 -0.94696546 -0.56408083 0.7680552 -0.37237883 1.875175
 ↪1.5623778 0.16714819 1.5595838 0.0839203 -0.8165728 -1.2181876 -1.4141134 -2.221717 1.
 ↪0910231 0.39918897 -1.4147882 -1.9443827 2.6638284 -2.5849214 -0.3483093 -1.2768111 1.
 ↪2041935 0.41885737 -0.6264915 -1.2598635 -0.17101997 -0.09451551 0.5562106 1.8215355 -
 ↪1.3849229 -0.16678634 -1.3049109 1.3956747 -0.425332 -0.58320785 -0.62582475 -0.
 ↪16236432 0.8221694 0.20428674 -0.27942896 0.121347904 0.3831149 0.19451053 0.3466418 -
 ↪1.2984078 0.36676487 0.75776196 1.5233855 1.6458269 1.73043 -0.5802344 -0.48261273 -0.
 ↪6443515 -1.0062621 0.8157141 0.0649764 0.13610162 -0.33701542 -0.42747515 -0.
 ↪0011477228 -0.9921381 0.558996 0.48417446 0.42329437 0.54720676 0.57775104 -1.2895788
 ↪0.64017355 0.9923972 0.64543486 -2.407712 0.40264577 0.738344 1.1438419 0.6721332 0.
 ↪18367681 -0.5367812 1.710209 0.22282977 -0.37812966 2.1818678 0.61612314 1.6069653 1.
 ↪6151379 -1.0042768 0.8307863 0.085298695 -0.5351512 0.77987534 -1.1209589 -1.2757269 0.
 ↪19029789 0.09809208 0.30246544 -0.14954329 -0.66100293 1.0569872 0.28426272 0.9857154
 ↪0.75427866 1.4701519 -0.12504229 -0.87289083 -0.43871146 0.20166902 0.2271485 -0.
 ↪05514332 -0.720507 -0.4757063 0.8947587 0.36385572 1.272678 -0.35486463 1.2087017 -0.
 ↪4758017 -0.18907958 0.24432425 -1.2633739 -0.37864834 -1.0377893 -1.0432142 0.60313225
 ↪-0.4432806 0.597437 -0.5591857 0.28537536 0.039966587 -1.1142912 -0.7018597 -0.2819324
 ↪1.0536848 -0.040540628 0.16402985 0.70751774 -1.624833 -1.2773706 0.05926119 0.4667645
 ↪0.6903434 1.0204479 -1.7858443 -0.26309192 1.6994039 -1.0891271 -0.71158147 0.24580163
 ↪-0.07374777 -1.4286835 1.8534608 0.12186845 -1.1296402 -0.7697011 1.6788592 2.6152475
 ↪0.606213 0.3166484 0.30229023 -1.2840998 0.012669044 0.87669975 0.32712832 -0.4437163
 ↪0.53256166 -0.54276496 0.32467005 -0.9636277 -0.58549994 -0.1298496 0.67720413 -2.
 ↪3554142 1.3474101 -0.81879246 2.5617309 1.878895 0.49217474 -1.3570213 1.1938144 0.
 ↪3645778 -0.29008883 0.50031495 -1.5553544 -1.2081774 0.87830913 -1.1718067 1.7222011 -
 ↪0.13035145 -1.9812089 -1.8173308 -0.41010964 -0.26526994 -0.4790508 0.45257586 0.
 ↪80826676 2.0087717 -1.0434382 -2.4669588 0.54181504 0.054128893 -0.33712658 -2.437975
 ↪1.0693933 0.13688947 -0.60142255 -0.10989515 -1.1721189 1.1690396 0.98004854 1.7259405
 ↪-0.63115627 0.17960648 0.1349787 1.8558581 0.2962184 -0.47908902 -0.13066223 -0.
 ↪49583495 -0.80173033 1.1078131 -0.21119505 -0.8546662 0.6391783 -0.5089646 -0.96097887
 ↪0.038478117 -0.67008615 -0.54741406 -0.9072827 -0.06801312 1.3966236 -0.547623 0.
 ↪16072778 -1.3989493 -2.599672 0.2585235 0.25142732 0.1333462 -1.0716463 -1.0153651 -0.
 ↪6559947 0.51636326 -1.7126486 -0.073620744 -0.6133027 -0.74761003 0.09934151 1.0121211
 ↪-0.95096993 1.5341284 -1.079764 0.113598 0.29572484 -0.2686275 0.64157134 2.4731357 -1.
 ↪695656 0.55485827 -0.47317806 0.26248395 0.28782308 -0.53618616 -0.8938534 -0.5614469 -
 ↪0.16780692 -0.86070776 0.7112449 0.95629495 -0.4078699 0.73303235 0.22123657 0.
 ↪44072202 1.5468754 0.09615625 2.2312448 1.7467606 1.3082488
 了 0.124426864 1.8280954 0.9831009 0.14293717 -1.4974583 3.1034458 -0.7097836 0.20220008
 ↪1.4538946 -1.8817077 -0.22880717 -1.027875 -0.53895986 0.80745065 -1.0450182 -0.

(下页继续)

7.1.4 词向量 - 0.2843285 1.0525922 -0.40245408 -1.1113168 0.58638555 -0.86827195 0.4367374
 ↪0.59662205 0.7141082 -0.8070898 -0.96410495 0.35778406 -0.2732946 0.43445915 1.7109047
 ↪-0.41755947 0.810394 -1.0918777 1.1574733 -1.2285464 0.2751894 0.10051493 0.9152668 0.

(续上页)

```

... 1.7539198 -0.07875835 -0.51359785 0.5462624 1.0336319 0.33710518 0.7153517 -0.
→14696723 -0.4674709 0.585131 -0.09571628 -0.044367265 -0.43465808 -1.075802 -0.
→29818213 -0.7845866 1.1654521 -1.3100251 1.8042226 0.2514134 1.4274467 -1.0617328 -0.
→3200904 1.2856162 0.3420093 1.7161297 1.8614627 -0.20988376 -0.42488077 -0.7149864 0.
→41926503 -0.37290215 -0.118796825 -0.57392484 0.39521572 0.45619187 -0.24028234 0.
→4770612 0.04256915 -0.39457968 -0.008392483 -1.209323 0.430775 0.82605964 -0.004404845
→0.37295258 0.4512206 -0.2135426 -0.16859093 0.8448976 -0.31460437 -1.7188169 -0.
→5480035 0.44762316 -0.14954409 0.31225446 0.9399047 0.21786243 0.69624907 0.53500223 -
→2.7766602 0.3260321 -0.13577469 0.6590769 0.58879477 -0.62039936 -0.866531 -0.13919026
→-0.073862985 0.34415373 -2.1943939 -0.72885746 -2.5571342 -0.73328006 2.3266015 0.
→4431778 -0.10030712 1.3283393 -0.26529813 -0.33246863 0.81044066 0.66299045 1.3830155 -
→0.49563265 -1.7842948 2.4802263 -0.36092368 0.74590343 0.8457939 -0.1902837 1.0022603 -
→0.5104553 0.80944073 -0.3719534 0.7508766 -0.730415 -1.265106 -0.6364332 -1.685758 1.
→1658943 -0.064504445 -0.15554048 0.08889705 -0.09455234 -0.36020827 -0.44518313 -0.
→49773395 -1.8581092 0.3746055 -0.14251812 0.029269492 0.37341043 0.69249976 -0.4510986
→-0.6552884 -0.49757797 -0.9416513 -1.042354 0.21657246 -0.5294435 -0.12662728 0.
→3742792 -0.6304494 -0.3711382 -0.8409685 -0.55995417 0.5129402 -0.2115912 0.33800915 0.
→67653304 0.36557457 0.5908807 0.18838193 0.3303122 -0.26492664 -1.3028978 -1.9588792 0.
→13098347 1.2453116 -0.5137858 0.15241857 -0.49777454 0.5939944 1.2962011 -1.665363 -0.
→97219986 0.29830503 -0.43484548 -0.9646101 -2.1332662 1.064172 0.37780657 -0.5783379 0.
→6535722 0.9515499 0.2886058 -0.7116952 0.09929629 0.8267979 0.36100662 -0.32459423 0.
→35443765 -0.23248821 0.88938844 -0.039720625 -0.9524684 0.27245703 2.8707743 0.
→43341875 1.5878333 -0.52806544 1.6490899 -1.7025334 -0.5329122 -1.031357 0.7788266 1.
→6018186 -0.049502328 -0.029527912 -0.47482267 0.16400504 0.20526074 -0.09405405 1.
→0447553 1.0227536 -1.0295554 0.751836 1.3792868 1.2144673 0.5338277 -0.70540535 -0.
→33774805 0.113717 -0.1213611 0.6725416 0.18328986 -0.20078385 -1.1855491 -0.9250905 1.
→0585163 -0.40305907 0.36642185 0.101170816 -0.66567755 0.2951031 -0.6511099 -0.
→99900395 -0.21455282 0.81051373 -0.14177085 1.3635707 -1.7237631 0.51812005 -0.
→71558076 1.7924819 0.14843622 -0.29164916 1.126084 -0.20472099 1.6225713 -0.60215634 -
→0.23482214 -1.5326608 0.6890701 -1.2694215 -0.20689794 -1.0027355 0.7053792 -0.8321893
→1.176607 1.0103234 -1.3610929 0.16453268 -2.3285384 1.4695607 -0.022401335 -1.6919589 -
→0.61018145 -1.6643481 0.65750724 -0.15422283 -0.33395147 0.77055055 -0.2663506 -0.
→640906 -1.2953341 -1.2691419 -0.9496096 1.4021212 -0.29681277 -1.2956185 -0.81685257 -
→0.93699765 -0.10026271 0.4026852 0.17704841 0.14466256 -1.3512911 -0.9849602 1.561256
→1.6520786 0.2695429 -0.3704157 -0.66111404 1.3731217 -1.2292235 0.35934207 -1.1112843
→1.3329659 -0.4493885 -0.693006 -1.4414659 0.21878286 -2.2706199 1.1016893 -0.16959193 -
→0.13103354 -0.051698178 -0.8295336 0.46076056 -0.3791775 0.5837915 0.3287772 -0.
→1266879 0.29440388 3.2369833 0.22973283 0.39704415 -0.99494326 0.69763094 -0.075644396
→-0.031685505 0.6717069 0.6972548 -0.8750802 0.25193936 0.91673565 0.44680834 0.
→36706924 1.1802963

```

(下页继续)

(续上页)

少女 0.5887249 -0.78131676 -0.9086393 -1.1748865 -0.7446431 -0.33194453 -0.018740159 -0.
 ↪6819682 1.1373322 -0.2449827 0.38390064 -0.4037972 -0.42380548 1.8774717 -0.056615744 ↪
 ↪1.1482375 1.0340028 -0.57691437 -0.10536296 0.602655 0.7542164 -0.5638564 -0.71151686 -
 ↪0.08572001 0.29281658 0.52927524 1.5935234 0.09691928 -1.0369319 0.18286628 1.6077064 -
 ↪0.6484846 1.2906547 0.82070255 0.42539054 -0.46507382 0.32321668 -1.6392659 -0.264856 ↪
 ↪1.2421234 -0.20365983 -0.020171288 0.86471444 0.7232603 -0.9572046 1.6881616 -0.
 ↪5733427 0.34953114 -0.7623181 -0.1049821 -0.23901421 1.7843546 1.4431484 -1.0618613 0.
 ↪88080454 -0.42794758 -1.6699258 0.3234611 -0.35222912 -1.1160336 0.057735726 -0.
 ↪7693502 0.1561758 0.50093096 1.9453335 -0.812546 -2.8262587 -0.009005266 -0.09875295 -
 ↪1.333687 -0.14573775 0.46749806 0.755247 1.1295704 0.895495 -1.5277107 -1.5787225 0.
 ↪124769524 -1.6838331 1.7976208 0.86056334 1.5805879 0.4043093 0.86494225 1.6273291 0.
 ↪40853548 -1.7177533 1.3041753 -0.40075505 -1.908944 -0.35136628 -1.6667027 -0.3832609 ↪
 ↪1.4697397 -1.7034197 -0.7213212 -0.34379014 1.3429763 0.12348689 -1.4705572 -1.4270422 ↪
 ↪0.24953331 1.3322998 0.02141577 -0.04586138 -0.08307748 -0.9784215 0.04490414 1.383406 ↪
 ↪-0.57164764 0.18689618 -0.46882167 -0.05742165 -0.90621465 -1.7430568 0.64610285 0.
 ↪22093566 0.71984667 0.23604086 -2.0309274 0.18095501 0.79003716 0.7923131 2.2337909 0.
 ↪50145984 -0.20433225 0.24310149 1.6265295 -2.0527804 0.076875634 -0.19025083 -0.
 ↪51757085 0.22870481 0.027272848 1.1691102 0.4587316 0.43038988 -1.4018912 0.31812528 -
 ↪1.0155283 -0.6313369 -0.6585674 0.22004573 -0.6052359 1.5660753 0.4774539 2.1519923 0.
 ↪11055413 0.32297432 0.3056909 1.5830464 -0.14859697 0.49388915 1.1956668 -2.5543363 0.
 ↪22358978 1.3447273 1.3092629 -0.14362293 0.7085022 -1.7020465 0.09408313 -1.417123 0.
 ↪7645757 0.060660124 -0.36149168 0.7115275 1.7099825 -0.15572844 0.27442068 0.048999123 ↪
 ↪-0.19752415 -0.8670349 -0.26930657 -0.27720222 -0.17450356 1.3144078 -0.2786439 1.
 ↪4584504 0.5331807 -2.408406 -1.1464162 -0.7464278 -0.88895607 -0.5660856 -0.14826216 -
 ↪0.8454592 -0.41659743 0.73387223 1.8717443 1.2645547 0.5606523 -0.78016657 0.95922476 ↪
 ↪2.5326197 1.6011894 0.6156151 -0.4252702 0.3975298 -1.6362991 1.4911361 0.28891438 0.
 ↪87486833 0.7208409 0.5737307 -1.0389473 -1.3981676 -0.4815167 0.03707392 1.7858388 0.
 ↪59070474 -0.5626557 0.3910045 0.035984877 2.1952462 -0.9893836 0.62462777 -0.3701214 -
 ↪1.3561703 0.7157114 -1.0020103 1.1730001 -0.48587084 0.57544714 -0.7790919 0.52735734 -
 ↪0.3946973 -0.58449775 1.0182343 0.85085005 0.2953459 -1.9785928 -0.3930518 -0.72646505 ↪
 ↪0.9768115 0.17771009 -0.44179973 0.78593755 0.8447062 -0.005129957 0.5753596 0.6570053 ↪
 ↪0.70418715 -0.6634827 0.5337006 0.3853094 -0.28450736 -1.0903058 -0.14038745 1.3840564 ↪
 ↪0.7502709 -0.043994833 -1.3120382 1.4737962 -0.09856514 -0.053444806 1.3115609 -0.
 ↪9847638 2.2367926 -0.30558985 1.4043404 0.18040906 -0.36622265 -0.8305084 -1.085571 -0.
 ↪012008861 -0.89203405 -0.18426119 1.6373096 -1.3801707 0.3139381 -1.0484347 0.44056708 ↪
 ↪-0.14707406 0.5474443 0.2298568 -1.53983 2.0013795 -1.0588335 -0.009949998 1.066051 -2.
 ↪4138741 0.5206372 0.023850137 -0.62356704 0.34778613 -0.6537413 0.42022324 -0.12714641 ↪
 ↪-0.28691298 0.60363704 -0.3824652 0.60583377 0.24133673 -0.85732937 -0.27193385 -0.
 ↪535049 -2.1983075 2.1011653 -0.15304893

7.1.2 跳字模型, skip_gram

1. 训练数据

就普通的文本格式

训练数据示例如下：

第一章 陨落的天才

“斗之力，三段！”

望着测验魔石碑上面闪亮得甚至有些刺眼的五个大字，少年面无表情，唇角有着一抹自嘲，紧握的手掌，因为大力，而导致略微尖锐的指甲深深的刺进了掌心之中，带来一阵阵钻心的疼痛……

“萧炎，斗之力，三段！级别：低级！”测验魔石碑之旁，一位中年男子，看了一眼碑上所显示出来的信息，语气漠然的将之公布了出来……

中年男子话刚刚脱口，便是不出意外的在人头汹涌的广场上带起了一阵嘲讽的骚动。

“三段？嘿嘿，果然不出我所料，这个“天才”这一年又是在原地踏步！”

“哎，这废物真是把家族的脸都给丢光了。”

“要不是族长是他的父亲，这种废物，早就被驱赶出家族，任其自生自灭了，哪还有机会待在家族中白吃白喝。”

“唉，昔年那名闻乌坦城的天才少年，如今怎么落魄成这般模样了啊？”

2. 使用示例

skip_gram 共实现了三种模型，分别为基础 softmax 模型 (SkipGramBaseModule)、基于负采样的优化模型 (SkipGramNegativeSamplingModule)、基于层次 softmax 的优化模型 (SkipGramHierarchicalSoftmaxModule)。

三种模型提供的接口一致，如下所示：

导入接口

```
from lightnlp.we import SkipGramBaseModule, SkipGramNegativeSamplingModule, \
    SkipGramHierarchicalSoftmaxModule # 分别导入 skip_gram 不同模型
```

训练

```
# skip_gram_model = SkipGramHierarchicalSoftmaxModule()
skip_gram_model = SkipGramNegativeSamplingModule()
# skip_gram_model = SkipGramBaseModule()
```

(下页继续)

(续上页)

```
train_path = '/home/lightsmile/NLP/corpus/novel/test.txt'
dev_path = '/home/lightsmile/NLP/corpus/novel/test.txt'

skip_gram_model.train(train_path, dev_path=dev_path, save_path='./skip_gram_saves')
```

测试

```
skip_gram_model.load('./skip_gram_saves')

skip_gram_model.test(dev_path)
```

预测

```
test_target = '族长'
print(skip_gram_model.evaluate(test_target, '他'))
print(skip_gram_model.evaluate(test_target, '提防'))
```

预测结果：

```
1.0
0.002815224463120103
```

保存词向量

```
skip_gram_model.save_embeddings('./skip_gram_saves/skip_gram_ns.bin')
```

./skip_gram_saves/skip_gram_ns.bin 文件内容：

```
623 300
<unk> -0.69455165 -1.3275498 -1.1975913 -0.3417502 0.13073823 1.3608844 0.15316872 -2.
↪ 295731 0.45459792 0.09420798 -0.73944765 0.11755463 -1.6275359 0.6623806 0.8247673 1.
↪ 7149547 -0.49345177 -0.5932094 -1.3025115 0.40126365 1.8675354 0.46296182 0.81418717 -
↪ 0.51671696 -1.328723 -0.27371547 -1.5537426 1.0326972 0.11647574 0.1607528 0.5110576 -
↪ 1.2010366 -0.81535685 0.5231469 2.212553 0.43934354 -0.8626878 1.5049676 -0.8135034 -0.
↪ 8322859 0.068298176 0.7376674 0.6459309 0.07635216 -0.77374196 0.29933965 1.6596211 0.
↪ 46682465 -0.8282705 -0.22142725 1.7853647 1.4777366 -0.63895816 2.1443112 -2.2435715 0.
↪ 85962945 1.6643075 1.082537 -0.6922347 -2.2418396 -0.20210272 -1.2102528 -0.48685002 0.
↪ 65887684 -0.2534356 -1.0342008 -1.1101105 0.94670665 0.21063486 -0.2467249 0.16507177
↪ 0.61120677 0.27850544 -1.0511587 -0.9382702 -0.105773546 -1.2759126 0.77076215 1.
7.1. 词向量, we ↪ 6730801 0.7634321 0.22365877 -1.7401465 -1.6434158 0.94023687 -1.3609751 -2.153141 0.
↪ 3826534 0.32158422 -2.4204254 -2.1351569 -0.7265906 1.2896249 -1.6444998 0.62701744 3.
↪ 9122646e-05 -1.348553 1.6431069 0.4589956 -1.8367497 0.81131816 0.13370599 0.9231004 -
```

(下页继续)

(续上页)

```

<pad> 0.9462378 -1.0530922 0.26814827 0.75049055 -0.43643618 0.90060383 0.38048416 0.
↪3394666 -0.6542603 -1.2994871 0.2035602 0.13271607 -0.0821392 0.6386408 -0.53183955 -0.
↪21759015 1.3303281 -0.4926919 1.2892267 0.49860442 1.6501433 -0.5349831 1.7068337 1.
↪1600994 0.4631011 1.0019102 -0.080210954 0.35248953 0.88543874 0.08718851 -0.50338817 -
↪1.4847835 -0.5894625 -1.0142589 -0.37832302 -1.6291661 -0.12362847 1.8569889 0.
↪47709444 0.6944984 1.5645366 -1.643663 -0.4542581 -1.7151413 -0.8393249 0.9062153 -0.
↪047601987 -1.101938 -0.68224543 -0.39662254 0.5475226 -1.2819566 -0.86349916 -0.
↪07766274 -0.27872422 -0.8497833 1.7615329 0.2950122 0.68848085 -0.26785335 0.08160306
↪0.5527327 -1.1441914 -0.8601009 -0.2983682 1.4938309 -0.7786196 0.29549783 0.08286876
↪0.33651295 0.45808968 0.10132327 -0.94001776 1.0869813 1.7297467 0.6415491 -3.0990815 -
↪0.70891887 -0.62066174 0.8763827 0.75606215 0.18597008 0.782098 0.07622817 -0.55206585
↪0.72135127 -0.019433482 0.5038495 -0.94488984 1.4516689 0.18088494 1.3465247 -0.
↪74685186 0.99718165 0.065872364 0.98572636 0.8221382 0.768447 -0.4056811 -0.9117917 -2.
↪05203 -0.78518504 0.12391317 -0.033092286 0.46701878 -1.1559975 -0.89441043 0.36609322
↪-2.0792224 -0.57335913 -1.0121179 0.6026655 1.0777911 0.09417599 0.26320156 2.6018775 -
↪0.2755741 0.9520457 -0.04128785 0.32038128 -0.5574524 0.26191193 0.18591642 -1.9010495
↪-0.27394825 0.65679026 0.29634175 -0.60466653 -1.3784024 0.7435744 -1.4532461 -0.
↪037048157 -0.5559504 -1.1130326 -2.0174382 1.9073203 0.21787305 -0.14302431 -0.
↪29675826 0.33756196 1.4894477 0.7317302 -2.0191894 -1.1759464 0.8036417 -0.37761644 0.
↪9244614 -0.7413941 -0.08381902 -1.4885721 -1.6779492 -0.59202635 1.0431904 0.7708446 -
↪0.041408855 -1.2213532 -0.2857886 0.7738537 -0.7683973 -0.3201996 -0.4752588 0.
↪14970754 -1.5409429 0.4487029 -1.5121255 0.56920415 0.11346252 1.4692949 -0.0945662 2.
↪142825 -2.618194 1.4771916 0.6997561 1.0059751 0.24992754 -1.8951392 -0.8522846 0.
↪98763144 -0.8822291 0.11832724 -1.0928622 1.2359277 0.80170745 1.0475003 1.5270966 0.
↪95872986 0.8958471 -1.2497357 -0.31796277 -0.8195951 0.51742077 -0.22876325 -0.5562857
↪-1.924446 1.2476108 -0.35275942 -1.6121302 0.57726604 0.20068043 1.1353028 -2.4147425 -
↪0.8989361 -1.4968843 0.6448405 -0.8628415 0.88103485 -0.3248718 1.0207465 -1.3894114 -
↪0.90123475 -2.6463938 0.9470338 -0.11909193 -0.61639553 0.7213106 0.8824293 -0.
↪39685965 1.6633297 -1.107534 0.4709047 0.33735672 -0.056239445 0.35526997 -0.9191851 1.
↪2952671 -0.75040734 -1.7293545 -0.5775496 0.006652971 0.3147311 -0.85833013 0.09456847
↪-1.0624956 -0.9020722 -0.09103666 1.7845771 1.9998456 -1.727455 -2.2023408 0.3902349 -
↪0.24948567 -1.6048291 0.14066061 0.44590333 -0.93849236 -1.1319045 -0.62959474 -0.
↪12584576 0.91559213 1.2120887 1.6113585 -0.2791995 -0.11430749 -2.109812 0.7273863 0.
↪7348798 1.119425 0.89362687 0.25193694 0.07618663 0.07243939 -0.4955755 1.0170685 1.
↪5341507 -0.5218003 -1.6152122 0.9274748 -1.6640632 -1.3126534 0.11114946 -0.65346044 -
↪0.6130383 0.7909551 0.22126918 1.6984801 1.2792808 0.5046258 1.2279602 0.9770026 -1.
↪145929 -0.0426054 -0.94418496 0.5853211 1.007048 0.36722738 0.17046496 -0.041508738 0.
↪8590547 0.08046034 -0.60373837 0.64457446 -0.25976962 0.960138 1.0904832 1.8453016 0.
↪018720066 1.3756162 1.0828762 -1.249238 0.79106873
, 0.900907 0.07571198 -0.7531744 1.374081 -0.051039666 0.7277553 -1.5629473 1.4199361 0.
↪43262932 -0.68931437 0.38527122 -0.95629644 -0.67784256 1.0736706 -0.73837465 1.
↪0659839 1.3746638 0.13170229 0.44516808 -1.3651135 0.6797121 -0.41324878 -0.74141294 0.
↪6231089 -0.40646043 1.5950443 -0.4391045 -0.8985314 -1.1638266 -1.085841 0.6797121 -0.41324878 -0.74141294 0.
↪07254573 1.17701 0.14969468 -0.6091943 -0.0053135455 -0.24863426 0.8653415 -0.49431074
↪0.40305167 -0.019052468 -1.4530281 0.9524088 0.19623129 1.0812551 -0.7029672 0.

```

(续上页)

的 -0.20160268 -0.21241257 -0.4043321 2.1909928 0.49679247 -0.1325275 0.42584014 0.
 ↪24683614 -0.9702372 -1.8741518 -0.7252045 -0.49983644 -0.65247107 0.76959157 0.8259947
 ↪-0.6513283 -0.8979589 1.3167363 -0.2347999 1.0080328 0.04068194 -0.70863515 -0.3246833
 ↪-0.42160204 -1.4620594 -0.8168891 -0.28889376 0.96584153 -0.06742255 1.7455348 1.
 ↪4442544 0.24048196 1.2854279 0.14298542 -0.6515416 -0.7833069 -1.2629595 -2.1125658 1.
 ↪0249085 0.40845707 -1.2015674 -1.5291021 2.2098982 -2.1091754 -0.41207585 -1.0482799 0.
 ↪78599465 0.31762454 -0.5183959 -1.3728874 -0.15425354 0.0436417 0.25058135 1.4118088 -
 ↪1.0422605 -0.07541436 -1.1068802 0.9455314 -0.5711538 -0.6461189 -0.18952727 -0.
 ↪21206772 0.7762203 0.02701252 -0.20934224 0.22475724 0.83357966 0.2729621 0.44660333 -
 ↪1.3553737 0.52011245 0.805614 1.4149666 1.6287401 1.6888571 -0.48137692 -0.31397444 -0.
 ↪9113469 -0.84345376 0.6603501 -0.01110802 -0.07945263 -0.3899895 -0.5179783 0.16231695
 ↪-0.8189658 0.4745495 0.25934523 0.44624203 0.38669428 0.48463246 -1.2172782 0.81605846
 ↪0.8540907 0.44676793 -2.2456388 0.39226332 0.32102612 1.3043944 0.8576041 0.20862296 -
 ↪0.51472336 1.3458765 0.0749595 -0.09046895 2.0222366 0.66191554 1.2438471 1.487185 -0.
 ↪8989187 0.74124473 -0.023372192 -0.44223523 0.6265085 -0.77208 -1.2376994 -0.015103638
 ↪0.05522992 0.09492233 -0.23367663 -0.51066947 0.7461931 0.027357863 0.96861994 0.
 ↪32018813 1.3833076 -0.25898007 -0.78406125 -0.45310873 0.014744817 0.29670417 -0.
 ↪14763092 -0.69623804 -0.5450609 0.94851893 0.26137534 1.1350462 -0.37602738 1.2355423 -
 ↪0.5942697 -0.40257028 0.14945346 -1.1114887 -0.33638072 -0.8206219 -0.49885294 0.
 ↪48334897 -0.18454923 0.37010637 -0.1678072 0.37671828 0.4342044 -0.6350701 -0.45998573
 ↪-0.017748803 0.7512209 -0.077632286 -0.014426709 0.3794435 -1.424685 -0.9906908 -0.
 ↪18001547 0.5476539 0.5737612 1.0301657 -1.4586309 -0.13719666 1.2435308 -0.83659005 -0.
 ↪5306402 0.5618243 0.09081328 -1.3465264 1.4966936 0.09357808 -1.1012143 -0.8601246 1.
 ↪5156639 2.2984858 0.48021093 0.48690277 0.10015719 -1.1285332 0.033390086 0.75665116 0.
 ↪33436894 -0.76391685 0.38092253 -0.5958041 0.14064594 -0.97736394 -0.61810505 -0.
 ↪19199394 0.5802718 -2.0147579 1.2695452 -0.76885015 2.0117762 1.6867042 0.40166444 -1.
 ↪2177283 0.9590221 0.42674235 -0.19682969 0.5186299 -1.2642647 -0.8827122 0.6405918 -0.
 ↪75486416 1.4068292 0.08357187 -1.7767766 -1.399881 -0.48782966 -0.17888801 -0.2964377
 ↪0.46553472 0.5966398 1.6487685 -0.9368118 -2.1757812 0.3725775 0.17232625 -0.57584375 -
 ↪2.0905218 1.1426715 0.16465737 -0.21529475 0.04670203 -1.2801889 0.96900284 1.024924 1.
 ↪7386234 -0.42269483 -0.019390948 0.21670038 1.6658463 0.50636417 -0.5177907 -0.
 ↪028692413 -0.20513107 -0.5925592 0.9350939 -0.31975344 -0.77178174 0.64559805 -0.
 ↪19214787 -0.59565204 0.044145744 -0.67553836 -0.6401371 -0.7174468 -0.08718807 1.
 ↪4099891 -0.64017993 0.093481496 -1.288413 -2.2056794 0.29075697 0.068356246 0.14803462
 ↪-0.89601153 -0.86346716 -0.49652 0.27589476 -1.5443418 -0.377951 -0.351205 -0.7131136
 ↪0.14281324 1.0897856 -0.5939442 1.2551428 -0.9339728 0.016592525 0.04654862 -0.
 ↪40097335 0.62805176 2.1380942 -1.3083881 0.7333389 -0.36281568 0.07450257 0.018646415 -
 ↪0.7296821 -0.9214585 -0.5921474 -0.30300933 -0.6692921 0.43340573 0.78606945 -0.
 ↪34040892 0.2503277 0.037954286 0.40688336 1.2049704 0.10686994 1.8918518 1.416152 0.
 ↪9152621

(下页继续)

(续上页)

```

了 -0.11100567 1.8244103 0.99881077 -0.1217553 -1.2965926 2.5752037 -0.52939695 -0.
→024460727 1.481322 -1.6454383 -0.32412064 -0.812192 -0.6247802 0.5879142 -0.99333626 -
→0.09178917 1.2610664 0.17786814 -0.09755645 -1.381767 0.3328385 -0.28843904 -0.8976809
→0.51493317 -0.2354604 0.7267729 -0.2759223 -0.875182 0.80424625 -0.68562067 0.345338 -
→0.36612466 0.68599254 -0.69162357 -0.74803936 0.4015563 -0.19871116 0.64606947 1.
→5207865 -0.5436914 0.7913142 -0.8087305 0.8900298 -1.2114931 0.29318812 0.010917914 0.
→794381 0.22564566 0.45246655 0.035850927 0.90022266 0.30116358 -0.08802621 0.22181591 -
→1.3465856 0.45377666 -1.1994032 -1.1599623 -0.90402436 0.4413774 -0.60787785 -0.
→18609983 -0.28758097 0.86769056 1.070895 0.18053351 1.5689834 -0.825002 -0.1844389 0.
→8352371 -2.0554366 -1.1158409 -1.1282442 0.3164764 0.81705356 1.3285462 -1.1886504 0.
→34115657 1.4381564 0.18249284 -2.3909829 -0.7105743 -1.3165226 2.6451735 0.044927556 1.
→8649187 -0.7606436 1.3267876 -0.8695353 -0.82985395 0.53350765 -1.5758001 0.07124073 0.
→3113274 -0.31943765 0.19381034 -0.40180263 -1.2200341 -0.46648136 -0.9460392 1.56903 -
→0.023224937 -0.109002806 -0.122972146 -0.38673204 -0.34158748 0.8986403 -1.3978604 0.
→30148593 0.10867346 0.5560802 0.6666235 -0.9128663 -0.11801989 -1.6656537 0.470698 1.
→1878209 0.21174078 -1.3163888 -0.88419724 1.2939 1.0864538 -0.59195155 1.3402531 0.
→9949826 1.0850434 1.9927737 -0.05526331 -0.9249912 -0.99546725 0.19423905 0.34155425 0.
→116912715 0.46196705 -0.22920796 0.62712306 0.19182688 2.545757 0.0005199494 0.
→047344636 1.4041449 -0.4366798 -0.30238223 -0.8592163 -0.1450108 -1.0397685 -1.7507579
→0.35720333 -1.0890079 -1.0088387 -2.166981 -0.119711794 -0.894902 -0.34313017 -0.
→83818877 1.8535222 -0.35545102 1.1037723 -0.30759943 -0.073416024 -0.6099894 -0.
→082894325 0.5696355 0.02636172 -0.4934832 0.8380404 0.6522451 0.23057912 -0.97358364 -
→0.20894456 -0.6702974 -1.350783 -1.494697 0.6830102 -0.6583244 1.4899143 1.3386569 0.
→8388928 0.53304636 0.10571614 -1.6693112 0.34666675 -0.18932551 -1.1207005 -0.13619985
→-0.37456858 1.6312733 -1.0417099 0.18789463 0.76180553 0.3472593 0.746827 -0.59724313
→1.1076568 0.39835903 0.3250713 -0.890195 0.851977 -1.8276579 -0.5058599 0.038222674 -0.
→40975145 0.34742656 -0.656784 0.52408326 -0.24545604 0.6244086 1.5422938 -0.11375929 0.
→4439602 1.194646 0.108786985 -1.2637025 0.30540422 -0.6331694 -0.10961318 -0.269253 0.
→3094217 0.37967274 0.4198166 -0.7752912 0.93305284 -0.51205474 0.10966317 0.42575598 1.
→3847312 0.09545336 0.35880265 1.3952161 -0.50513256 0.6978953 -0.42724264 0.61427826 0.
→42278302 1.3981684 -0.21110083 -1.2695825 0.3271231 -0.18441366 -0.18477333 -1.5454911
→0.89642715 -1.0348141 0.9462476 -0.97564644 0.33486953 0.015225394 0.66027576 -0.
→016567787 0.38754386 0.33223125 -0.01711932 0.81019586 1.8340786 -0.8252963 0.53832585
→0.7248964 1.0357465 -1.7218229 0.7785235 -0.059777193 -1.6390696 -0.13061148 1.6316975
→0.9055566 0.31447434 -0.050311707 -0.14748432 -1.152082 -0.32038185 -0.20986848 -0.
→3397427 0.7406922 0.19800813 -0.37535998 0.8428346 -0.74992263 -1.8344332 1.7500505 -0.
→019035367 -0.17899665 -1.7635467 0.27003205 -0.07612512 -0.0870243 0.9975619 -0.
→7678579 -0.51703197 -1.0690825 1.4577312 1.6575297 -0.5548624 0.60410196 1.4686142 0.
→36586297 0.55307645 -0.72146875 -1.1415323 1.8086451 -1.4277877
… 1.5090982 -0.12030965 -0.49108917 0.553509 0.8591837 0.36316723 0.67062587 -0.
→35389656 -0.37601423 0.4347592 -0.21231161 -0.0190618 -0.092252515 -0.8948071 -0.
→35531363 -0.6883051 0.98320943 -1.3508446 1.3872787 0.038260926 1.3760954 -0.96063673 -
24→0.40200645 1.1017556 0.3531388 1.588081 1.8033375 -0.269804 -0.57227846 -0.57227846
→21478365 -0.3091859 -0.28627992 -0.44842216 0.28986707 0.50668955 -0.12714297 0.
→40329775 0.2786915 -0.15945554 -0.072181724 -1.1862227 0.43608934 0.5287543 -0.

```

(下页继续)

(续上页)

少女 0.43625292 -0.49615338 -0.80635625 -0.95218086 -0.7034901 -0.24982916 -0.19628817 -
 ↪0.6407552 1.1144902 -0.18383415 0.240521 -0.48174432 -0.43238354 1.5460479 0.17876568 ↪
 ↪1.1772094 0.7226882 -0.40163878 -0.27529833 0.6727029 0.6616228 -0.4649058 -0.82819325 ↪
 ↪0.077967666 0.22773252 0.47824302 1.263238 0.16044211 -0.8465441 -0.03908252 1.5012993 ↪
 ↪-0.6464531 1.2720991 0.7075523 0.2655532 -0.16548817 0.18449956 -1.57685 -0.3937183 1.
 ↪1220738 -0.024133356 0.061765857 0.9063283 0.63651776 -0.9984391 1.7219917 -0.16911109 ↪
 ↪0.3950257 -0.44229308 -0.042594746 -0.18409568 1.5873259 1.5174317 -0.7761925 0.
 ↪85540444 -0.25851408 -1.7153916 0.21526498 -0.30503598 -1.0740207 -0.26189303 -0.
 ↪7361771 0.3404945 0.6267011 1.6266515 -0.7249698 -2.5369713 -0.040242277 -0.1521137 -1.
 ↪2230408 0.105983995 0.24729799 0.66424936 0.7428047 0.8017003 -1.6448103 -1.3638207 0.
 ↪22201629 -1.6191676 1.7650356 0.8733405 1.2922136 0.15274213 0.56989926 1.3315401 0.
 ↪3242414 -1.5333288 0.9968748 -0.26946962 -1.7603902 -0.06785398 -1.4466665 -0.2051215 ↪
 ↪1.0579334 -1.6114388 -0.6311598 -0.42203426 1.2702805 0.049536392 -1.5223664 -1.
 ↪1988252 -0.005043749 1.2519345 -0.1971786 -0.15249117 -0.09027343 -0.68423694 0.
 ↪05290712 1.1976366 -0.55118096 0.43239573 -0.3921759 -0.10146248 -0.84204507 -1.
 ↪5519683 0.41450986 0.08121465 0.70506895 0.36502063 -1.8334937 0.0016303719 0.8337098 ↪
 ↪0.8315608 2.1455774 0.5289709 -0.15538014 0.18361539 1.5023578 -1.5350997 -0.34078425 -
 ↪0.17852718 -0.2980528 0.3791773 0.13815962 1.0244071 0.3535323 0.45246312 -1.0676222 0.
 ↪26734406 -0.7833891 -0.4830871 -0.81460315 0.40275338 -0.4218575 1.2936684 0.36921188 ↪
 ↪2.0835738 0.29924822 0.607953 0.13717254 1.3793758 -0.17895296 0.25180477 1.0935096 -2.
 ↪4911408 0.059239827 1.4227028 1.0269765 -0.012899189 0.667102 -1.654389 0.32084498 -1.
 ↪3945848 0.9240499 0.14125341 -0.40186697 0.46305555 1.4799619 -0.1761365 0.08784475 -0.
 ↪16064519 -0.32681388 -0.9726041 -0.46950138 -0.10297061 -0.1305801 1.2899957 -0.
 ↪20281254 1.5096477 0.579964 -2.1962826 -1.0500919 -0.75640684 -0.68146974 -0.61898047 -
 ↪0.038223892 -0.63422275 -0.40183118 0.71116716 1.7054831 1.2117803 0.5540193 -0.
 ↪7673277 0.8996408 2.1318727 1.3070806 0.3919676 -0.19198467 0.37812284 -1.5805892 1.
 ↪622781 0.4241692 0.9778187 0.5272743 0.6481839 -0.91835433 -1.2837874 -1.0056475 -0.
 ↪03336126 1.7704506 0.59232867 -0.6739266 0.4252755 -0.04943613 1.933676 -1.0441738 0.
 ↪4929349 -0.5993543 -0.97701305 0.6164371 -1.2127788 1.2599823 -0.5473247 0.93479854 -0.
 ↪6774386 0.59664416 -0.5358335 -0.6079708 0.7937759 0.5176709 0.2346288 -2.056608 -0.
 ↪35982183 -0.6090097 0.8602409 0.055992365 -0.6665505 0.6803273 0.8781159 -0.028397428 ↪
 ↪0.6073012 0.5945208 0.7166259 -0.48727062 0.25150546 0.06475472 -0.33076963 -0.9388699 ↪
 ↪-0.47334203 1.1939013 0.78029764 0.022830477 -1.198792 1.2806718 0.1218006 0.05305545 ↪
 ↪1.0819892 -0.8576298 1.796153 -0.05783273 1.4125075 0.22831114 -0.14899425 -0.5525253 -
 ↪0.8165426 0.043676265 -0.641531 -0.37138024 1.5661736 -1.2548814 0.12986626 -0.9875852 ↪
 ↪0.4007069 -0.23187949 0.4992489 0.2498534 -1.1453637 1.7015793 -0.91252553 0.07962135 ↪
 ↪0.7060094 -2.1144807 0.18295327 0.30965614 -0.36403483 0.39038125 -0.580957 0.33897293 ↪
 ↪-0.1780094 -0.03921564 0.55165535 -0.44981298 0.706237 0.13913499 -0.35856977 -0.
 ↪20512235 -0.3393937 -1.9689944 2.374302 0.087832846

7.2 文本分类, tc

7.2.1 情感极性分析, sa

1. 训练数据

tsv 文件格式

训练数据示例如下:

	label	text
0	0	备胎是硬伤!
1	0	要说不满意的话, 那就是动力了, 1.5 自然吸气发动机对这款车有种小马拉大车的感觉。如今天气这么热, 上路肯定得开空调, 开了后动力明显感觉有些不给力不过空调制冷效果还是不错的。
2	0	油耗显示 13 升还多一点, 希望慢慢下降。没有倒车雷达真可恨
3	0	空调不太凉, 应该是小问题。
4	0	1、后排座椅不能平放; 2、科技感不强, 还不如百万帝豪, 最希望增加车联网的车机。像你好博越一样。3、全景摄像头不清楚, 晚上基本上用处不大
5	1	车子外观好看, 车内空间大。
6	1	最满意的真的不只一点, 概括一下最满意的就是性价比了。ps: 虽然没有 s7 性价比高 (原厂记录仪, 绿净)
7	0	底盘调教的很低, 坐的感觉有些别扭, 视角不是很好。
8	0	开空调时, 一档起步动力不足。车子做工有点马虎。

2. 使用示例

训练

```
from lightnlp.tc import SA

# 创建 SA 对象
sa_model = SA()

train_path = '/home/lightsmile/Projects/NLP/chinese_text_cnn/data/train.sample.tsv'
dev_path = '/home/lightsmile/Projects/NLP/chinese_text_cnn/data/dev.sample.tsv'
vec_path = '/home/lightsmile/Downloads/1410356697_ 浅笑哥 fight/自然语言处理/词向量/sgns.
↪zhihu.bigram-char'
```

(下页继续)

(续上页)

```
# 只需指定训练数据路径, 预训练字向量可选, 开发集路径可选, 模型保存路径可选。
sa_model.train(train_path, vectors_path=vec_path, dev_path=dev_path, save_path='./sa_
↳ saves')
```

测试

```
# 加载模型, 默认当前目录下的 `saves` 目录
sa_model.load('./sa_saves')

# 对 train_path 下的测试集进行读取测试
sa_model.test(train_path)
```

预测

```
sa_model.load('./sa_saves')

from pprint import pprint

pprint(sa_model.predict('外观漂亮, 安全性佳, 动力够强, 油耗够低'))
```

预测结果:

```
(1.0, '1') # return 格式为 (预测概率, 预测标签)
```

7.2.2 关系抽取, re

1. 训练数据

训练数据示例如下, 其中各列分别为实体 1、实体 2、关系、句子

钱钟书	辛笛	同门	与辛笛京沪唱和聽钱钟书与钱钟书是清华校友, 钱钟书高辛笛两班。
元武	元华	unknown	于师傅在一次京剧表演中, 选了元龙 (洪金宝)、元楼 (元奎)、元彪、成龙、元华、元武、元泰 7 人担任七小福的主角。

2. 使用示例

训练

```
from lightnlp.tc import RE

re = RE()

train_path = '/home/lightsmile/Projects/NLP/ChineseNRE/data/people-relation/train.sample.
↪txt'
dev_path = '/home/lightsmile/Projects/NLP/ChineseNRE/data/people-relation/test.sample.txt
↪'
vec_path = '/home/lightsmile/NLP/embedding/word/sgns.zhihu.bigram-char'

re.train(train_path, dev_path=dev_path, vectors_path=vec_path, save_path='./re_saves')
```

测试

```
re.load('./re_saves')
re.test(dev_path)
```

预测

```
print(re.predict('钱钟书', '辛笛', '与辛笛京沪唱和聽钱钟书与钱钟书是清华校友，钱钟书高辛笛两
班。'))
```

预测结果：

```
(0.7306928038597107, '同门') # return 格式为（预测概率，预测标签）
```

7.3 序列标注，sl

7.3.1 中文分词（cws）

1. 训练数据

BIS

训练数据示例如下：

4 S
日 S
清 B
晨 I
, S
同 B
样 I
在 S
安 B
新 I
县 I
人 B
民 I
政 I
府 I
门 B
前 I
, S
不 B
时 I
有 S
民 B
众 I
专 B
程 I
来 I
此 S
拍 B
照 I
留 B
念 I
, S
有 S
的 S
甚 B
至 I
穿 B
着 I
统 B
一 I

(下页继续)

(续上页)

的 S
服 B
饰 I
拍 B
起 I
了 S
集 B
体 I
照 I
。 S

2. 使用示例

1. 训练

```
from lightnlp.sl import CWS

cws_model = CWS()

train_path = '/home/lightsmile/NLP/corpus/cws/train.sample.txt'
dev_path = '/home/lightsmile/NLP/corpus/cws/test.sample.txt'
vec_path = '/home/lightsmile/NLP/embedding/char/token_vec_300.bin'

cws_model.train(train_path, vectors_path=vec_path, dev_path=dev_path, save_path='./cws_
↪saves')
```

2. 测试

```
cws_model.load('./cws_saves')

cws_model.test(dev_path)
```

3. 预测

```
print(cws_model.predict('抗日战争时期，胡老在与侵华日军交战中四次负伤，是一位不折不扣的抗战老英雄'))
```

预测结果:

```
['抗日战争', '时期', ',', ' ', '胡老', '在', '与', '侵华日军', '交战', '中', '四次', '负伤', ',', ' ', '是', '一位', '不折不扣', '的', '抗战', '老', '英雄']
```

7.3.2 词性标注 (pos)

1. 训练数据

BIS

训练数据示例如下:

```
只 B-c
要 I-c
我 B-r
们 I-r
进 B-d
一 I-d
步 I-d
解 B-i
放 I-i
思 I-i
想 I-i
, S-w
实 B-i
事 I-i
求 I-i
是 I-i
, S-w
抓 B-v
住 I-v
机 B-n
遇 I-n
, S-w
开 B-l
拓 I-l
进 I-l
取 I-l
, S-w
建 B-v
```

(下页继续)

(续上页)

```
设 I-v  
有 S-v  
中 B-ns  
国 I-ns  
特 B-n  
色 I-n  
社 B-n  
会 I-n  
主 I-n  
义 I-n  
的 S-u  
道 B-n  
路 I-n  
就 S-c  
会 S-v  
越 S-d  
走 S-v  
越 S-d  
宽 B-a  
广 I-a  
。 S-w
```

2. 使用示例

1. 训练

```
from lightnlp.sl import POS

pos_model = POS()

train_path = '/home/lightsmile/NLP/corpus/pos/train.sample.txt'
dev_path = '/home/lightsmile/NLP/corpus/pos/test.sample.txt'
vec_path = '/home/lightsmile/NLP/embedding/char/token_vec_300.bin'

pos_model.train(train_path, vectors_path=vec_path, dev_path=dev_path, save_path='./pos_
↪saves')
```

2. 测试

```
pos_model.load('./pos_saves')

pos_model.test(dev_path)
```

3. 预测

```
print(pos_model.predict('向全国各族人民致以诚挚的问候！'))
```

预测结果：

```
[('向', 'p'), ('全国', 'n'), ('各族', 'r'), ('人民', 'n'), ('致以', 'v'), ('诚挚', 'a'), ('
→ 的', 'u'), ('问候', 'vn'), ('! ', 'w')]
```

7.3.3 命名实体识别 (ner)

1. 训练数据

BIO

训练数据示例如下：

```
清 B_Time
明 I_Time
是 O
人 B_Person
们 I_Person
祭 O
扫 O
先 B_Person
人 I_Person
, O
怀 O
念 O
追 O
思 O
的 O
日 B_Time
子 I_Time
```

(下页继续)

```
。 0  
  
正 0  
如 0  
宋 B_Time  
代 I_Time  
诗 B_Person  
人 I_Person
```

2. 使用示例

1. 训练

```
from lightnlp.sl import NER  
  
# 创建 NER 对象  
ner_model = NER()  
  
train_path = '/home/lightsmile/NLP/corpus/ner/train.sample.txt'  
dev_path = '/home/lightsmile/NLP/corpus/ner/test.sample.txt'  
vec_path = '/home/lightsmile/NLP/embedding/char/token_vec_300.bin'  
  
# 只需指定训练数据路径和 TensorBoard 日志文件路径，预训练字向量可选，开发集路径可选，模型保存  
# 路径可选（模型保存路径默认为 `xx_saves`，其中 xx 为模型简称，如 ner）。  
ner_model.train(train_path, vectors_path=vec_path, dev_path=dev_path, save_path='./ner_  
↪saves', log_dir='E:/Test/tensorboard/')
```

2. 测试

```
# 加载模型，默认当前目录下的 `ner_saves` 目录  
ner_model.load('./ner_saves')  
# 对 train_path 下的测试集进行读取测试  
ner_model.test(train_path)
```

3. 预测

```
from pprint import pprint

pprint(ner_model.predict('另一个很酷的事情是，通过框架我们可以停止并在稍后恢复训练。'))
```

预测结果：

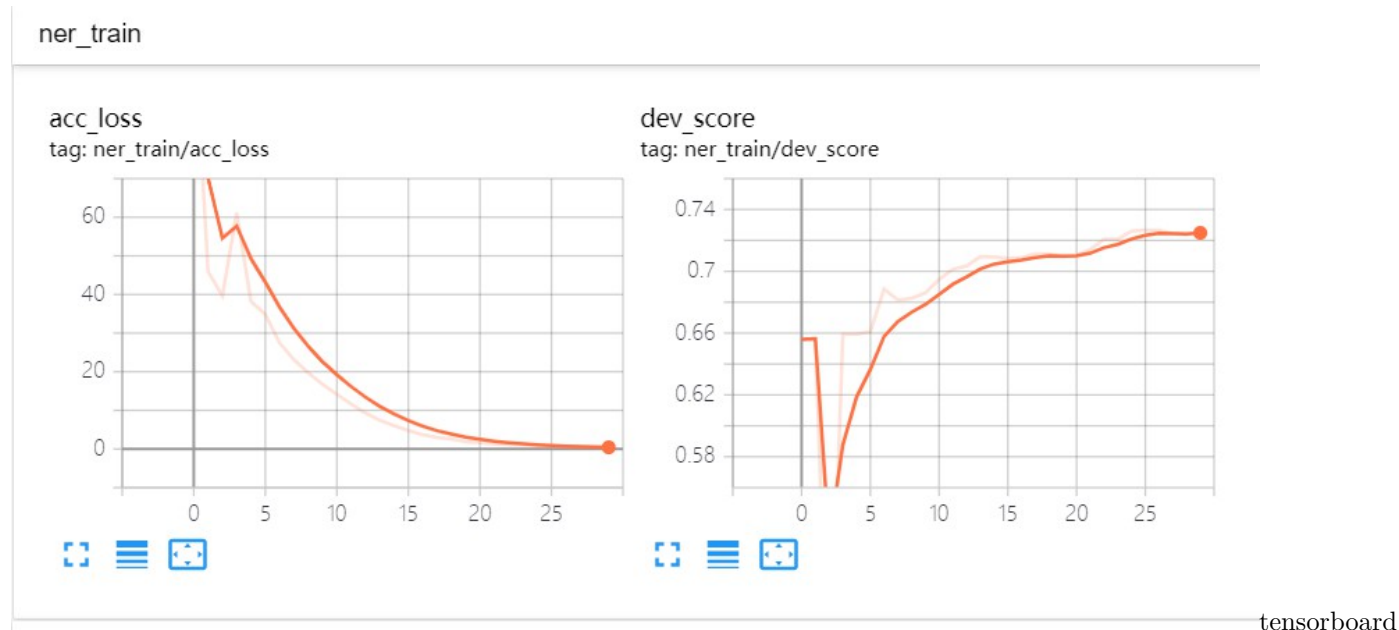
```
[{'end': 15, 'entity': '我们', 'start': 14, 'type': 'Person'}]
```

4. 查看训练效果

命令行中执行以下命令，其中 E:\Test\tensorBoard 修改为模型训练时日志存储路径，port 端口指定可选：

```
tensorboard --logdir=E:\Test\tensorBoard --port=2019
```

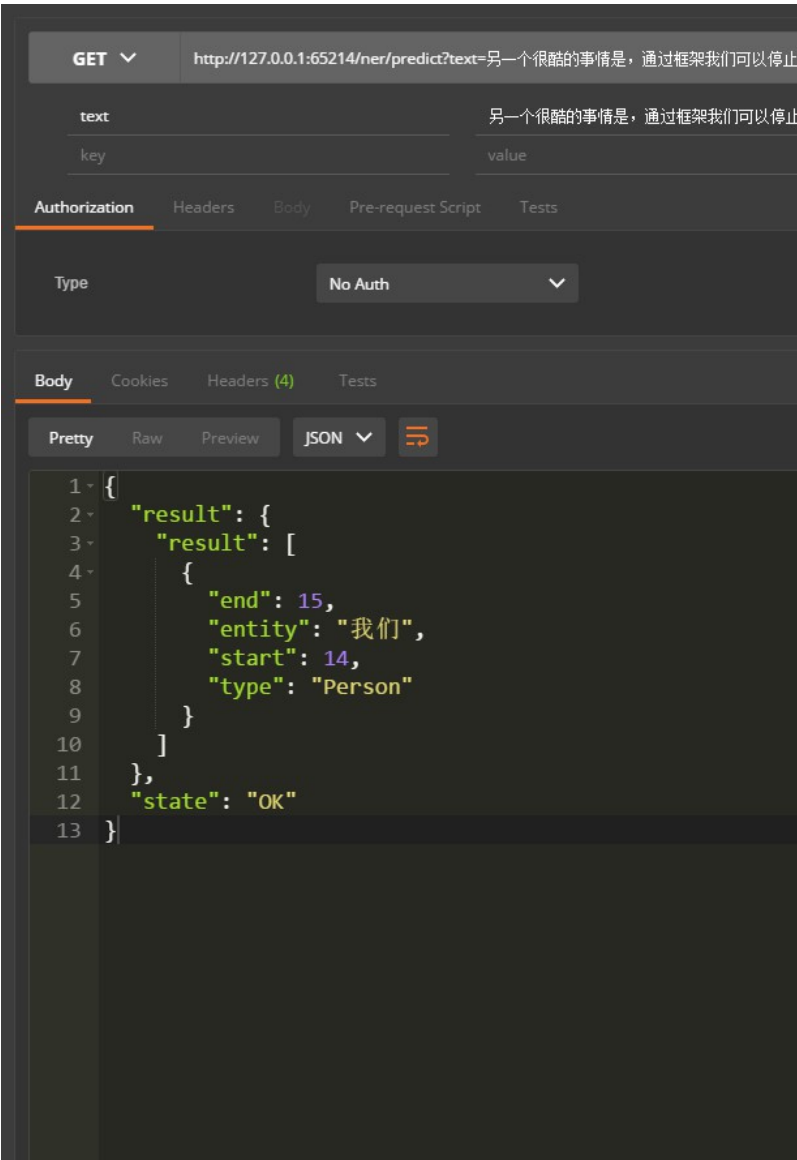
可以看到类似如下效果：



5. 部署服务

```
ner_model.deploy(host="localhost", port=2020, debug=False)
```

其中所有参数都可选，host 参数默认为 localhost，port 端口将由程序自动向系统申请空闲端口，默认不开启 debug 模式。



可以使用 Postman 或者编写程序进行调用测试,如下图:

```
In [9]: import requests
import json

In [22]: text = '另一个很酷的事情是，通过框架我们可以停止并在稍后恢复训练。'

In [23]: result = requests.get('http://127.0.0.1:65214/ner/predict', params={'text':text})

In [26]: from pprint import pprint
pprint(result.json())

{'result': {'result': [{'end': 15,
                        'entity': '我们',
                        'start': 14,
                        'type': 'Person'}]},
           'state': 'OK'}
```

jupyter-

notebook

7.3.4 语义角色标注 (srl)

1. 训练数据

CONLL

训练数据示例如下，其中各列分别为词、词性、是否语义谓词、角色，每句仅有一个谓语动词为语义谓词，即每句中第三列仅有一行取值为 1，其余都为 0。

宋浩京	NR	0	0
转达	VV	0	0
了	AS	0	0
朝鲜	NR	0	0
领导人	NN	0	0
对	P	0	0
中国	NR	0	0
领导人	NN	0	0
的	DEG	0	0
亲切	JJ	0	0
问候	NN	0	0
,	PU	0	0
代表	VV	0	0
朝方	NN	0	0
对	P	0	0
中国	NR	0	B-ARG0
党政	NN	0	I-ARG0
领导人	NN	0	I-ARG0
和	CC	0	I-ARG0
人民	NN	0	E-ARG0
哀悼	VV	1	rel
金日成	NR	0	B-ARG1
主席	NN	0	I-ARG1
逝世	VV	0	E-ARG1
表示	VV	0	0
深切	JJ	0	0
谢意	NN	0	0
。	PU	0	0

2. 使用示例

1. 训练

```
from lightnlp.sl import SRL

srl_model = SRL()

train_path = '/home/lightsmile/NLP/corpus/srl/train.sample.tsv'
dev_path = '/home/lightsmile/NLP/corpus/srl/test.sample.tsv'
vec_path = '/home/lightsmile/NLP/embedding/word/sgns.zhihu.bigram-char'

srl_model.train(train_path, vectors_path=vec_path, dev_path=dev_path, save_path='./srl_
↳ saves')
```

2. 测试

```
srl_model.load('./srl_saves')

srl_model.test(dev_path)
```

3. 预测

```
word_list = ['代表', '朝方', '对', '中国', '党政', '领导人', '和', '人民', '哀悼', '金日成
↳ ', '主席', '逝世', '表示', '深切', '谢意', '。']
pos_list = ['VV', 'NN', 'P', 'NR', 'NN', 'NN', 'CC', 'NN', 'VV', 'NR', 'NN', 'VV', 'VV',
↳ 'JJ', 'NN', 'PU']
rel_list = [0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0]

print(srl_model.predict(word_list, pos_list, rel_list))
```

预测结果：

```
{'ARG0': '中国党政领导人和人民', 'rel': '哀悼', 'ARG1': '金日成主席逝世'}
```

7.4 文本生成, tg

7.4.1 语言模型, lm

1. 训练数据

就普通的文本格式

训练数据示例如下：

第一章 陨落的天才

“斗之力，三段！”

望着测验魔石碑上面闪亮得甚至有些刺眼的五个大字，少年面无表情，唇角有着一抹自嘲，紧握的手掌，因为大力，而导致略微尖锐的指甲深深的刺进了掌心之中，带来一阵阵钻心的疼痛……

“萧炎，斗之力，三段！级别：低级！”测验魔石碑之旁，一位中年男子，看了一眼碑上所显示出来的信息，语气漠然的将之公布了出来……

中年男子话刚刚脱口，便是不出意外的在人头汹涌的广场上带起了一阵嘲讽的骚动。

“三段？嘿嘿，果然不出我所料，这个“天才”这一年又是在原地踏步！”

“哎，这废物真是把家族的脸都给丢光了。”

“要不是族长是他的父亲，这种废物，早就被驱赶出家族，任其自生自灭了，哪还有机会待在家族中白吃白喝。”

“唉，昔年那名闻乌坦城的天才少年，如今怎么落魄成这般模样了啊？”

2. 使用示例

训练

```
from lightnlp.tg import LM

lm_model = LM()

train_path = '/home/lightsmile/NLP/corpus/lm_test.txt'
dev_path = '/home/lightsmile/NLP/corpus/lm_test.txt'
vec_path = '/home/lightsmile/NLP/embedding/char/token_vec_300.bin'

lm_model.train(train_path, vectors_path=vec_path, dev_path=train_path, save_path='./lm_
↪saves')
```

测试

```
lm_model.load('./lm_saves')

lm_model.test(dev_path)
```

预测

文本生成

默认生成 30 个

```
print(lm_model.generate_sentence('少年面无表情，唇角有着一抹自嘲'))
```

结果：

```
少年面无表情，唇角有着一抹自嘲，紧握的手掌，因，无所谓的面上，那抹讥讽所莫下了脚步，当时的
```

得到给定上文下，下一个字的 topK 候选集及其概率

默认输出 top5 个

```
print(lm_model.next_word_topk('少年面无表情，唇角'))
```

结果：

```
[('有', 0.9791949987411499), ('一', 0.006628755945712328), ('不', 0.004853296559303999),  
→ ('出', 0.0026260288432240486), ('狠', 0.0017451468156650662)]
```

评估语句分数

结果为以 10 为底的对数，即 $\log_{10}(x)$

```
print(lm_model.sentence_score('少年面无表情，唇角有着一抹自嘲'))
```

结果：

```
-11.04862759023672
```

评估当前上文下，某一个字作为下一个字的可能性

```
print(lm_model.next_word('要不是', '萧'))
```

结果：

```
0.006356663070619106
```

7.4.2 机器翻译, mt

1. 训练数据

tsv 文件格式

训练数据示例如下:

```
Hi.      嗨。
Hi.      你好。
Run.     你用跑的。
Wait!    等等!
Hello!   你好。
I try.   让我来。
I won!   我赢了。
Oh no!   不会吧。
Cheers!  干杯!
He ran.  他跑了。
```

2. 使用示例

训练

```
from lightnlp.tg import MT

mt_model = MT()

train_path = '/home/lightsmile/NLP/corpus/translation/mt.train.sample.tsv'
dev_path = '/home/lightsmile/NLP/corpus/translation/mt.test.sample.tsv'
source_vec_path = '/home/lightsmile/NLP/embedding/english/glove.6B.100d.txt'
target_vec_path = '/home/lightsmile/NLP/embedding/word/sgns.zhihu.bigram-char'

mt_model.train(train_path, source_vectors_path=source_vec_path, target_vectors_
    ↪ path=target_vec_path,
               dev_path=train_path, save_path='./mt_saves')
```

测试

```
mt_model.load('./mt_saves')

mt_model.test(train_path)
```

预测

```
print(mt_model.predict('Hello!'))
print(mt_model.predict('Wait!'))
```

预测结果为：

```
('你好。', 0.6664615107892047)
('! ', 0.661789059638977)
```

7.4.3 聊天机器人, cb

1. 训练数据

tsv 文件格式

训练数据示例如下：

```
呵呵      是王若猫的。
不是      那是什么？
怎么了    我很难过，安慰我~
开心点哈，一切都会好起来 嗯 会的
我还喜欢她，怎么办      我帮你告诉她？发短信还是打电话？
短信      嗯嗯。我也相信
你知道谁么      肯定不是我，是阮德培
许兵是谁      吴院四班小帅哥
```

2. 使用示例

训练

```
from lightnlp.tg import CB

cb_model = CB()
```

(下页继续)

(续上页)

```

train_path = '/home/lightsmile/NLP/corpus/chatbot/chat.train.sample.tsv'
dev_path = '/home/lightsmile/NLP/corpus/chatbot/chat.test.sample.tsv'
vec_path = '/home/lightsmile/NLP/embedding/word/sgns.zhihu.bigram-char'

cb_model.train(train_path, vectors_path=vec_path, dev_path=train_path, save_path='./cb_
↪saves')

```

测试

```

cb_model.load('./cb_saves')

cb_model.test(train_path)

```

预测

```

print(cb_model.predict('我还喜欢她，怎么办'))
print(cb_model.predict('怎么了'))
print(cb_model.predict('开心一点'))

```

预测结果为：

```

('我你告诉她？发短信还是打电话？', 0.8164891742422521)
('我难过，安慰我', 0.5596837521537591)
('嗯会的', 0.595637918475396)

```

7.4.4 文本摘要, ts

1. 训练数据

tsv 文件格式

训练数据示例如下：

徐州 18 岁农家女孩宋爽，今年考入清华大学。除了自己一路闯关，年年拿奖，还帮妹妹、弟弟制定学习计划，姐弟仨齐头并进，妹妹也考上区里最好的中学。这个家里的收入，全靠父亲务农和打零工，但宋爽懂事得让人心疼，曾需要 200 元奥数竞赛的教材费，她羞于开口，愣是急哭了... 戳腾讯公益帮帮她们！# 助学圆梦# 江苏新闻的秒拍视频 徐州农家女孩考上清华，她的懂事让人心酸... 盖被子，摇摇篮，汪星人简直要把萌娃宠上天~细致周到有耐心，脾气还好，汪星人不愧是一届带娃好手 [笑而不语] 偶买噶视频的秒拍视频 带娃好手汪星人！把宝宝们宠上天 [憧憬]

(下页继续)

(续上页)

人们通常被社会赋予的“成功”所定义，“做什么工作”“赚多少钱”都用来评判一个人的全部价值，很多人出现身份焦虑。身份焦虑不仅影响幸福感，还会导致精神压力，甚至自杀。如果你也有身份焦虑，这个短片或许会有帮助。秒拍视频 感到压力大的同学看过来！如何缓解身份焦虑？[并不简单]

网友 @ 星蓝 seiran 教大家自制的捕捉器教程，简单方便，里面的洗洁精换成肥皂水或洗衣粉水都可以（用于溶解蟑螂腹部油脂防止爬出），白糖稍微多放点。怕蟑螂的童鞋，可以换成不透明的瓶子。转需~
 ↳这个厉害了！[good]

2. 使用示例

训练

```
from lightnlp.tg import TS

ts_model = TS()

train_path = '/home/lightsmile/NLP/corpus/text_summarization/ts.train.sample.tsv'
dev_path = '/home/lightsmile/NLP/corpus/text_summarization/ts.test.sample.tsv'
vec_path = '/home/lightsmile/NLP/embedding/word/sgns.zhihu.bigram-char'

ts_model.train(train_path, vectors_path=vec_path, dev_path=train_path, save_path='./ts_
↳saves')
```

测试

```
ts_model.load('./ts_saves')

ts_model.test(train_path)
```

预测

```
test_str = """
    近日，因天气太热，安徽一老太在买肉路上突然眼前一黑，摔倒在地。她怕别人不扶她，连忙
    说“快扶我起来，我不讹你，地上太热我要熟了！”这一喊周围人都笑了，老人随后被扶到路边休息。（颍州
    晚报）[话筒] 最近老人尽量避免出门！
    """

print(ts_model.predict(test_str))
```

预测结果为:

```
(', 我不讹你, 地上太热我要熟了! [允悲]', 0.03261186463844203)
```

7.5 句子关系, sr

7.5.1 句子关系, ss

1. 训练数据

tsv 文件类型

训练数据示例如下, 其中各列分别为语句 a, 语句 b, 相似关系, 包括 0, 不相似, 1, 相似:

1	怎么更改花呗手机号码	我的花呗是以前的手机号码, 怎么更改成现在的支付宝的号码手机号	1
2	也开不了花呗, 就这样了? 完事了	真的嘛? 就是花呗付款	0
3	花呗冻结以后还能开通吗	我的条件可以开通花呗借款吗	0
4	如何得知关闭借呗	想永久关闭借呗	0
5	花呗扫码付钱	二维码扫描可以用花呗吗	0
6	花呗逾期后不能分期吗	我这个 逾期后还完了 最低还款 后 能分期吗	0
7	花呗分期清空	花呗分期查询	0
8	借呗逾期短信通知	如何购买花呗短信通知	0
9	借呗即将到期要还的账单还能分期吗	借呗要分期还, 是吗	0
10	花呗为什么不能支付手机交易	花呗透支了为什么不可以继续用了	0

2. 使用示例

训练

```
from lightnlp.ss import SS

ss_model = SS()

train_path = '/home/lightsmile/Projects/NLP/sentence-similarity/input/atec/ss_train.tsv'
dev_path = '/home/lightsmile/Projects/NLP/sentence-similarity/input/atec/ss_dev.tsv'
vec_path = '/home/lightsmile/NLP/embedding/char/token_vec_300.bin'

ss_model.train(train_path, vectors_path=vec_path, dev_path=train_path, save_path='./ss_
→saves')
```

测试

```
ss_model.load('./ss_saves')
ss_model.test(dev_path)
```

预测

```
print(float(ss_model.predict('花呗更改绑定银行卡', '如何更换花呗绑定银行卡')))
```

预测结果：

```
0.9970847964286804
```

7.5.2 文本蕴涵, te

1. 训练数据

tsv 文件类型

训练数据示例如下，其中各列分别为前提、假设、关系，其中关系包括 **entailment**，蕴含、**neutral**，中立、**contradiction**，矛盾

是的，我想一个洞穴也会有这样的问题	我认为洞穴可能会有更严重的问题。	neutral
几周前我带他和一个朋友去看幼儿园警察	我还没看过幼儿园警察，但他看了。	contradiction
航空旅行的扩张开始了大众旅游的时代，希腊和爱琴海群岛成为北欧人逃离潮湿凉爽的夏天的令人兴奋的目的地。	航空旅行的扩大开始了许多旅游业的发展。	entailment
当两名工人待命时，一条大的白色管子正被放在拖车上。	这些人正在监督管道的装载。	↪ neutral
男人俩互相交换一个很大的金属环，骑着火车向相反的方向行驶。	婚礼正在教堂举行。	↪ contradiction
一个小男孩在秋千上玩。	小男孩玩秋千	entailment

2. 使用示例

训练

```
from lightnlp.sr import TE

te_model = TE()
```

(下页继续)

(续上页)

```

train_path = '/home/lightsmile/Projects/liuhuaiyong/ChineseTextualInference/data/te_
↳train.tsv'
dev_path = '/home/lightsmile/Projects/liuhuaiyong/ChineseTextualInference/data/te_dev.tsv
↳'
vec_path = '/home/lightsmile/NLP/embedding/char/token_vec_300.bin'

te_model.train(train_path, vectors_path=vec_path, dev_path=train_path, save_path='./te_
↳saves')

```

测试

```

te_model.load('./te_saves')
te_model.test(dev_path)

```

预测

```

print(te_model.predict('一个小男孩在秋千上玩。', '小男孩玩秋千'))
print(te_model.predict('两个年轻人用泡沫塑料杯子喝酒时做鬼脸。', '两个人在跳千斤顶。'))

```

预测结果为:

```

(0.4755808413028717, 'entailment')
(0.5721057653427124, 'contradiction')

```

7.6 结构分析, sp

7.6.1 基于图的依存句法分析, gdp

1. 训练数据

CONLL 格式, 其中各列含义如下:

1	ID	当前词在句子中的序号, 1 开始.
2	FORM	当前词语或标点
3	LEMMA	当前词语 (或标点) 的原型或词干, 在中文中, 此列与 FORM 相同
4	CPOSTAG	当前词语的词性 (粗粒度)

(下页继续)

(续上页)

5	POSTAG	当前词语的词性（细粒度）
6	FEATS	句法特征，在本次评测中，此列未被使用，全部以下划线代替。
7	HEAD	当前词语的中心词
8	DEPREL	当前词语与中心词的依存关系

在 CONLL 格式中，每个词语占一行，无值列用下划线' _ ' 代替，列的分隔符为制表符' \t '，行的分隔符为换行符' \n '；句子与句子之间用空行分隔。

示例如：

1	坚决	坚决	a	ad	-	2	方式
2	惩治	惩治	v	v	-	0	核心成分
3	贪污	贪污	v	v	-	7	限定
4	贿赂	贿赂	n	n	-	3	连接依存
5	等	等	u	udeng	-	3	连接依存
6	经济	经济	n	n	-	7	限定
7	犯罪	犯罪	v	vn	-	2	受事
1	最高	最高	n	nt	-	3	限定
2	人民	人民	n	nt	-	3	限定
3	检察院	检察院	n	nt	-	4	限定
4	检察长	检察长	n	n	-	0	核心成分
5	张思卿	张思卿	n	nr	-	4	同位语

2. 使用示例

训练

```
from lightnlp.sp import GDP

gdp_model = GDP()

train_path = '/home/lightsmile/NLP/corpus/dependency_parse/THU/train.sample.conll'
vec_path = '/home/lightsmile/NLP/embedding/word/sgns.zhihu.bigram-char'

gdp_model.train(train_path, dev_path=train_path, vectors_path=vec_path, save_path='./gdp_
↪saves')
```

测试

```
gdp_model.load('./gdp_saves')
gdp_model.test(train_path)
```

预测

```
word_list = ['最高', '人民', '检察院', '检察长', '张思卿']
pos_list = ['nt', 'nt', 'nt', 'n', 'nr']
heads, rels = gdp_model.predict(word_list, pos_list)
print(heads)
print(rels)
```

预测结果如下，其中程序会自动在语句和词性序列首部填充“”，因此返回的结果长度为 $\text{len}(\text{word_list}) + 1$ ：

```
[0, 3, 3, 4, 0, 4]
['<ROOT>', '限定', '限定', '限定', '核心成分', '同位语']
```

7.6.2 基于转移的依存句法分析，tdp

1. 训练数据

格式大致如下，其中每行代表一个 `sentence` 和对应的 `Actions`，两者用 `|||` 分隔，其中 `Actions` 包括三种：`Shift`、`REDUCE_R` 和 `REDUCE_L`，分别代表移入、右规约、左规约，其中 `sentence` 和 `Actions` 之间的序列长度对应关系为 $\text{len}(\text{Actions}) = 2 * \text{len}(\text{sentence}) - 1$ ：

```
Bell , based in Los Angeles , makes and distributes electronic , computer and building
→products . ||| SHIFT SHIFT REDUCE_R SHIFT SHIFT SHIFT SHIFT REDUCE_L REDUCE_R REDUCE_R
→REDUCE_R SHIFT REDUCE_R SHIFT REDUCE_L SHIFT REDUCE_R SHIFT REDUCE_R SHIFT SHIFT
→REDUCE_R SHIFT REDUCE_R SHIFT REDUCE_R SHIFT REDUCE_R SHIFT REDUCE_L REDUCE_R SHIFT
→REDUCE_R
`` Apparently the commission did not really believe in this ideal . ' ' ||| SHIFT SHIFT
→SHIFT SHIFT REDUCE_L SHIFT SHIFT SHIFT SHIFT REDUCE_L REDUCE_L REDUCE_L REDUCE_L
→REDUCE_L REDUCE_L SHIFT SHIFT SHIFT REDUCE_L REDUCE_R REDUCE_R SHIFT REDUCE_R SHIFT
→REDUCE_R
```

2. 使用示例

训练

```
from lightnlp.sp import TDP

tdp_model = TDP()

train_path = '/home/lightsmile/Projects/NLP/DeepDependencyParsingProblemSet/data/train.
↳sample.txt'
dev_path = '/home/lightsmile/Projects/NLP/DeepDependencyParsingProblemSet/data/dev.txt'
vec_path = '/home/lightsmile/NLP/embedding/english/glove.6B.100d.txt'

tdp_model.train(train_path, dev_path=dev_path, vectors_path=vec_path, save_path='./tdp_
↳saves')
```

测试

```
tdp_model.load('./tdp_saves')
tdp_model.test(dev_path)
```

预测

```
from pprint import pprint
pprint(tdp_model.predict('Investors who want to change the required timing should write_
↳their representatives '
                        'in Congress , he added . '))
```

预测结果如下：

```
{DepGraphEdge(head=(',', 14), modifier=('he', 15)),
 DepGraphEdge(head=('<ROOT>', -1), modifier=('Investors', 0)),
 DepGraphEdge(head=('Congress', 13), modifier=(',', 14)),
 DepGraphEdge(head=('Investors', 0), modifier=('who', 1)),
 DepGraphEdge(head=('he', 15), modifier=('added', 16)),
 DepGraphEdge(head=('in', 12), modifier=('Congress', 13)),
 DepGraphEdge(head=('representatives', 11), modifier=('in', 12)),
 DepGraphEdge(head=('required', 6), modifier=('timing', 7)),
 DepGraphEdge(head=('should', 8), modifier=('their', 10)),
 DepGraphEdge(head=('the', 5), modifier=('change', 4)),
 DepGraphEdge(head=('the', 5), modifier=('required', 6)),
```

(下页继续)

(续上页)

```

DepGraphEdge(head=('their', 10), modifier=('representatives', 11)),
DepGraphEdge(head=('their', 10), modifier=('write', 9)),
DepGraphEdge(head=('timing', 7), modifier=('should', 8)),
DepGraphEdge(head=('to', 3), modifier=('the', 5)),
DepGraphEdge(head=('want', 2), modifier=('to', 3)),
DepGraphEdge(head=('who', 1), modifier=('want', 2))}

```

返回的格式类型为 set，其中 DepGraphEdge 为命名元组，包含 head 和 modifier 两元素，这两元素都为 (word, position) 元组

7.7 工具模块

7.7.1 词向量

本框架也提供了类似 gensim 的加载词向量并得到相似词汇的功能，使用示例如下：

```

from lightnlp.utils.word_vector import WordVectors

vector_path = '/home/lightsmile/Projects/MyGithub/lightNLP/examples/cbow_saves/cbow_base.
↳bin'
word_vectors = WordVectors(vector_path)
print(word_vectors.get_similar_words('少女', dis_type='cos'))

```

输出结果为：

```

[('少女', 0.9999998807907104), ('嘲笑', 0.17718511819839478), ('同龄人', 0.
↳17244181036949158)]

```


8.1 项目架构

- base
 - config.py
 - model.py
 - module.py
 - tool.py
- sl, 序列标注
 - ner, 命名实体识别
 - cws, 中文分词
 - pos, 词性标注
 - srl, 语义角色标注
- sp, 结构分析
 - tdp, 基于转移的依存句法分析
 - gdp, 基于图的依存句法分析
- sr, 句子关系
 - ss, 句子相似度

- te, 文本蕴含
- tc, 文本分类
 - re, 关系抽取
 - sa, 情感分析
- tg, 文本生成
 - cb, 聊天机器人
 - lm, 语言模型
 - mt, 机器翻译
 - ts, 文本摘要
- utils
- we, 词向量
 - cbow, 词袋模型
 - skip_gram, 跳字模型

8.2 架构说明

8.2.1 base 目录

放一些基础的模块实现，其他的高层业务模型以及相关训练代码都从此 module 继承相应父类。

config

存放模型训练相关的超参数等配置信息

model

模型的实现抽象基类，包含 `base.model.BaseConfig` 和 `base.model.BaseModel`，包含 `load`、`save` 等方法

module

业务模块的训练验证测试等实现抽象基类，包含 `base.module.Module`，包含 `train`、`load`、`_validate`、`test` 等方法

tool

业务模块的数据处理抽象基类，包含 `base.tool.Tool`，包含 `get_dataset`、`get_vectors`、`get_vocab`、`get_iterator`、`get_score` 等方法

8.2.2 util 目录

放一些通用的方法

9.1 Deep Learning

- What's the difference between “hidden” and “output” in PyTorch LSTM?
- What's the difference between LSTM() and LSTMCell()?
- What is the difference between Luong Attention and Bahdanau Attention?
- 深度学习框架技术剖析 (转)
- Attention? Attention!

9.2 NLP

- 基于表示学习的信息抽取方法浅析
- 知识抽取-实体及关系抽取
- 知识抽取-事件抽取

9.3 知识图谱

- 翻译模型 (Trans 系列) 的知识表示学习
- 知识图谱向量化表示

9.4 Pytorch 教程

- PyTorch 常用方法总结 4: 张量维度操作 (拼接、维度扩展、压缩、转置、重复……)
- Pytorch 中的 RNN 之 `pack_padded_sequence()` 和 `pad_packed_sequence()`
- pytorch 学习笔记 (二): gradient
- `torch.multinomial()` 理解
- Pytorch 细节记录
- What does `flatten_parameters()` do?
- 关于 Pytorch 的二维 tensor 的 `gather` 和 `scatter_` 操作用法分析
- Pytorch `scatter_` 理解轴的含义
- ‘`model.eval()`’ vs ‘`with torch.no_grad()`’
- 到底什么是生成式对抗网络 GAN?

9.5 torchtext 介绍

- torchtext
- A Tutorial on Torchtext
- Torchtext 详细介绍
- torchtext 入门教程, 轻松玩转文本数据处理

9.6 其他工具模块

- python 的 Tqdm 模块
- pytorch-crf

9.7 词向量

- Chinese-Word-Vectors
- ChineseEmbedding
- 预训练好的词向量资源 - 简书
- pytorch_word2vec
- pytorch-word-embedding

9.8 数据集

- [Chinese-Literature-NER-RE-Dataset](#)
- [ChineseTextualInference](#)
- [中文短文本摘要数据集](#)
- [机器翻译语料库大全 \(免费下\)](#)
- [中文公开聊天语料库](#)

9.9 序列标注

- [a-PyTorch-Tutorial-to-Sequence-Labeling](#)
- [sequence_tagging](#)

9.10 文本分类

- [chinese_text_cnn](#)

9.11 命名实体识别

- [sequence_tagging](#)

9.12 关系抽取

- [ChineseNRE](#)
- [pytorch-pcnn](#)
- [关系抽取 \(分类\) 总结](#)

9.13 事件抽取

这里目前粗浅的将语义角色标注技术实现等同于事件抽取任务。

- [语义角色标注](#)
- [iobes_iob 与 iob_ranges 函数借鉴](#)
- [BiRNN-SRL](#)

- [chinese_semantic_role_labeling](#)

9.14 语言模型

- [char-rnn.pytorch](#)
- [Simple Word-based Language Model in PyTorch](#)
- [PyTorch 中级篇 \(5\): 语言模型 \(Language Model \(RNN-LM\)\)](#)

9.15 文本生成

- [好玩的文本生成](#)
- [基于深度学习的文本生成过程](#)

9.16 语句相似度

- [siamese_lstm](#)
- [sentence-similarity](#)

9.17 文本蕴含

- [ChineseTextualInference](#)

9.18 中文分词

- [中文自然语言处理中文分词训练语料](#)
- [中文分词、词性标注联合模型](#)
- [pytorch_Joint-Word-Segmentation-and-POS-Tagging](#)

9.19 词性标注

- [常见中文词性标注集整理](#)
- [分词：词性标注北大标准](#)
- [ICTCLAS 汉语词性标注集中科院](#)

- 中文文本语料库整理
- 中文分词、词性标注联合模型
- `pytorch_Joint-Word-Segmentation-and-POS-Tagging`

9.20 指代消解

- AllenNLP 系列文章之四：指代消解

9.21 依存句法分析

- 汉语树库
- Deep Biaffine Attention for Neural Dependency Parsing
- 中文句法结构
- 句法分析之依存句法
- Deep Biaffine Attention for Neural Dependency Parsing, hankcs 简要解读
- Simple and Accurate Dependency Parsing Using Bidirectional LSTM Feature Representations
- biaffine-parser
- DeepDependencyParsingProblemSet

9.22 自动摘要

- 干货 | 当深度学习遇见自动文本摘要, seq2seq+attention

9.23 阅读理解

- ASReader: 一个经典的机器阅读理解深度学习模型

9.24 聊天机器人

- ywk991112 / `pytorch-chatbot`
- keon / `seq2seq`
- bentrevett / `pytorch-seq2seq`
- IBM / `pytorch-seq2seq`

9.25 其他

- 基于距离的算法曼哈顿，欧氏等
- 在分类中如何处理训练集中不平衡问题
- Python-Pandas 如何 shuffle（打乱）数据？
- Python DataFrame 如何删除原来的索引，重新建立索引
- Pandas 在读取 csv 时如何设置列名-常用方法集锦
- Python 中 `__repr__` 和 `__str__` 区别
- Python3:ImportError: No module named 'compiler.ast'
- Automated Python 2 to 3 code translation
- git 拉取远程分支到本地
- OpenCC - 简体繁体转换
- Getting a random free tcp port in python using sockets
- 如何用 flask 部署 pytorch 模型 | Sherlock's Blog

CHAPTER 10

Indices and tables

- `genindex`
- `modindex`
- `search`