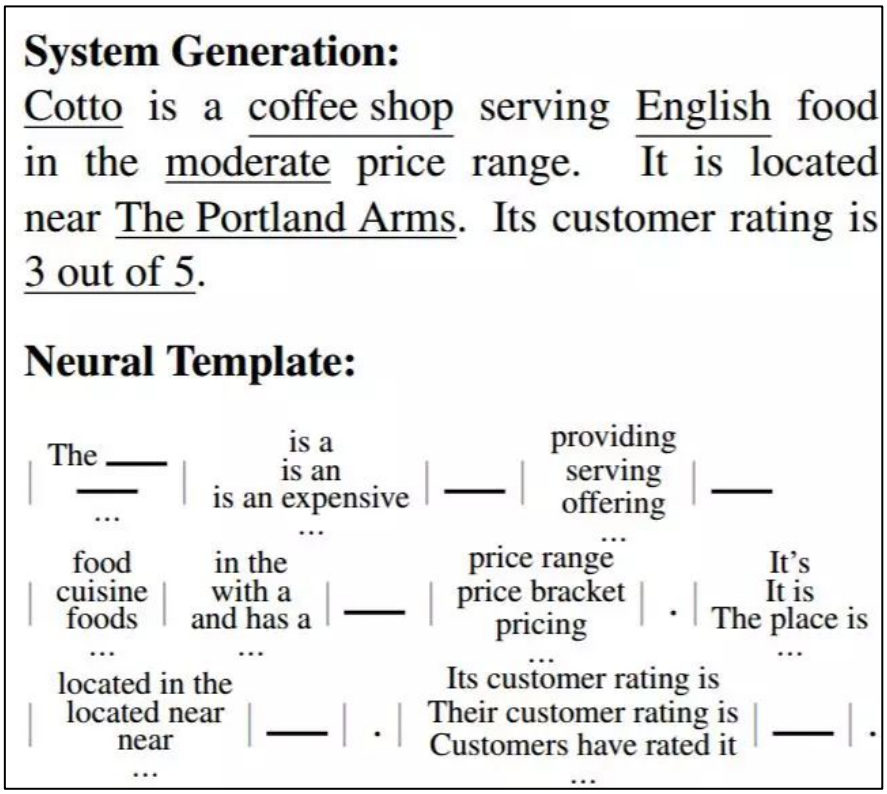
生成財報敘述-總結報告

- *演算法*

論文: Learning Neural Templates for Text Generation

摘要：編碼器-解碼器模型在很大程度上無法解釋文本，並且難以控制其措辭或內容。本文提出了一種使用隱藏的半馬爾可夫模型（HSMM）解碼器的神經生成系統，它學習潛在的、離散的模板並生成。我們展示了該模型學習有用模板的能力，並且這些模板讓生成變得更具解釋性和可控性。



- *數據集*

指標識別數據

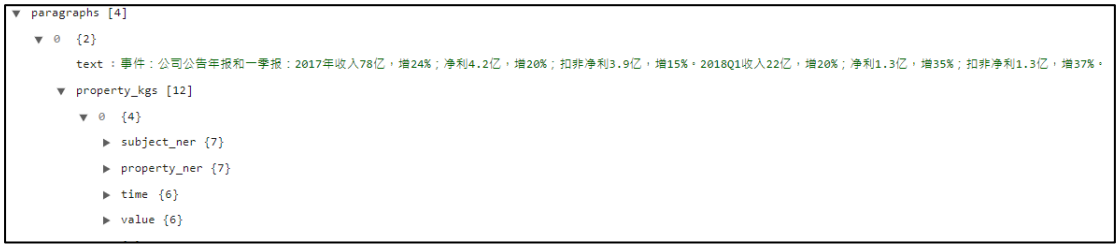


圖 1.

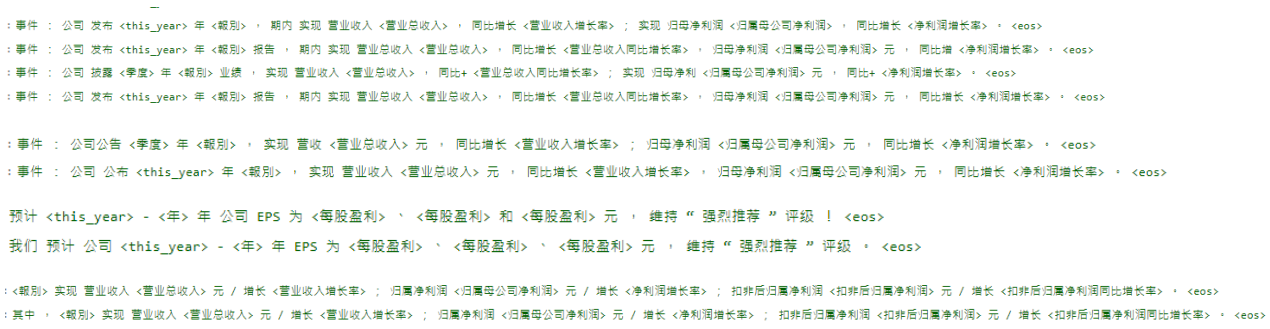


圖 2.

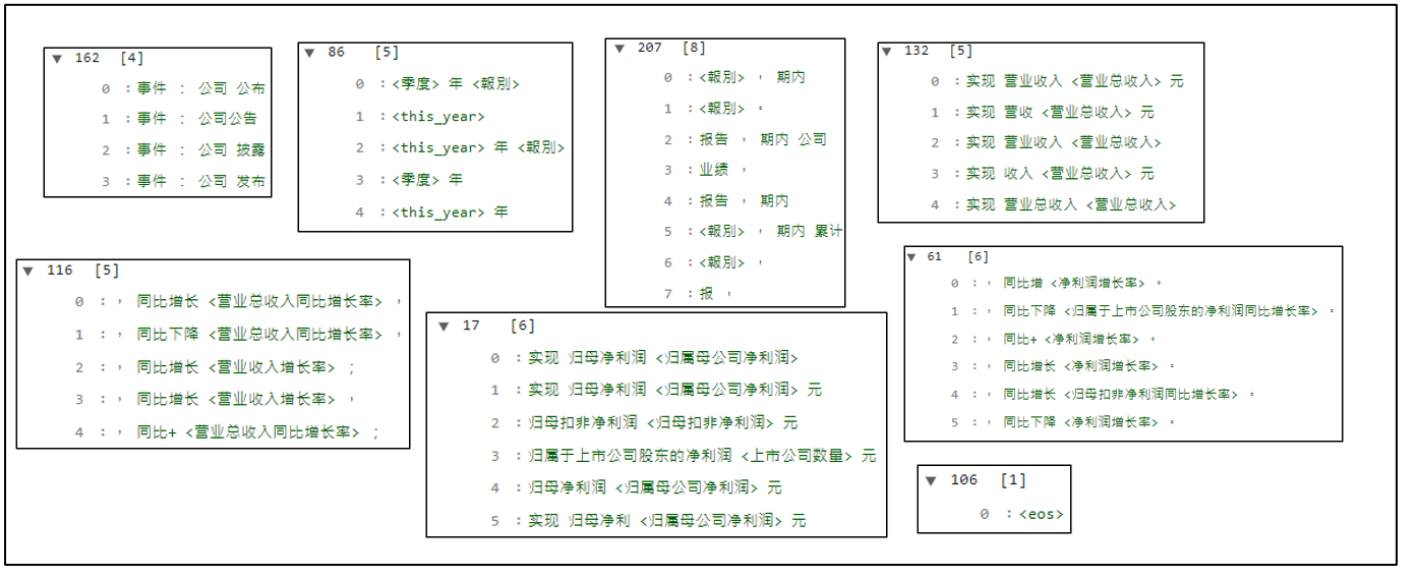
利用指標識別數據(如圖 1)，將每段的財報敘述對齊成如圖 3 的訓練資料。
(1 筆訓練資料為 1 句財報敘述)

句子結構分類為 報別敘述、其他敘述 這兩種。

總訓練資料為 8079 筆，依 9:1 切分 訓練集、驗證集做使用

- *實驗結果*

Template states:



Selected template state:

[事件 : 公司公告]162,[<this_year>]86,[报告 , 期内 公司]207,[实现 营业收入 <营业总收入> 元]132,[, 同比增长 <营业收入增长率> ,]116,[实现 归母净利 <归属母公司净利润> 元]17,[, 同比下降 <净利润增长率> 。]61,[<eos>]106,

Predictions result:

事件：公司公告<this_year>报告，期内公司实现营业收入<营业总收入>元，同比增长<营业收入增长率>，实现归母净利 <归属母公司净利润>元，同比下降<净利润增长率>。<eos>

其他敘述：

我们预计公司<this_year>-<年>年公司EPS为<归属母公司净利润>、<归属母公司净利润>和<归属母公司净利润>元，维持“审慎强烈推荐”评级！<eos>

维持“强烈推荐-A”。预测<年>-<年>年EPS分别元、<每股盈利>元/<每股盈利>元，目前股价相应对应<next_year>年PE为<股价>倍。<eos>

预计<last_year>-<next_year>年净利润分别约为<归属母公司净利润>元、<归属母公司净利润>元和<归属母公司净利润>元，PE分别为<市盈率>-<next_year>年对应<this_year>X、<市盈率>X、<市盈率>X，维持“强烈推荐-A”评级。<eos>

维持“审慎推荐-A”投资评级。我们预计公司<this_year>-<年>年实现收入<营业总收入>元、<营业总收入>元和<营业总收入>元，实现归母净利润<归属母公司净利润>元、<营业总收入>元和<归属母公司净利润>元，EPS分别为<每股盈利>元、<营业总收入>元和<每股盈利>元，PE分别为<市盈率>倍、<市盈率>倍和<市盈率>倍，维持“审慎推荐-A”投资评级。<eos>

報別敘述：

事件：<this_year>年<報別>公司实现营收<营业总收入>元，同比增长<营业收入增长率>；实现归母净利润<归属母公司净利润>元，同比增长<净利润增长率>；EPS为<每股盈利>元，同比增加<每股收益增长率>；加权平均净资产收益率<加权平均净资产收益率>，增加<加权平均净资产收益率同比增长率>。<eos>

事件：公司前<報別>实现营收<营业总收入>元，同比增长<营业收入增长率>；净利润<归属母公司净利润>元，同比增长<净利润增长率>，EPS为<每股盈利>元。<eos>

事件：公司公布了<報別>财务数据，，公司实现营业收入<营业总收入>元，同比+<销售收入同比增长率>；实现归属于上市公司股东净利润<归属母公司净利润>元，同比-<净利润增长率>；对应每股收益<每股盈利>元，同比上升<每股收益增长率>。<eos>

- *實際應用*

輸入:以.CSV 檔的形式，輸入結構化數據
輸出:以.TXT 檔的形式，輸出財報文本敘述

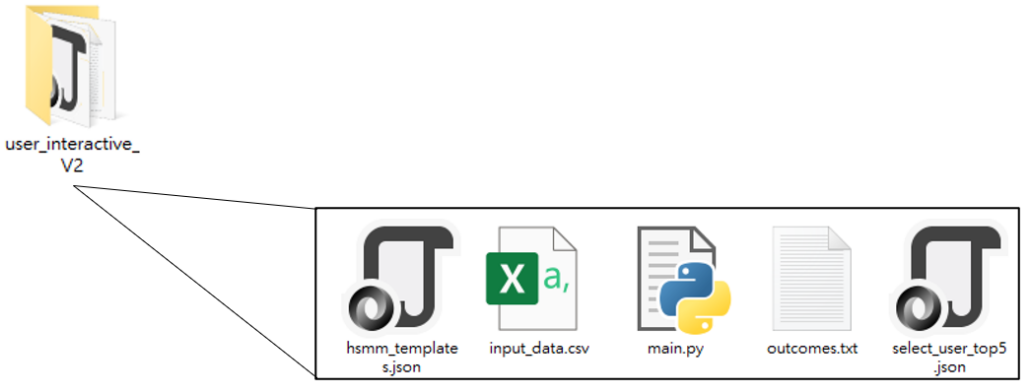
	A	B	C	D	E	F
1	this_year	報別	营业总收入	营业总收入	归属母公司	净利润增长率
2	2020	一季報	100000	20%	50000	10%

事件：公司发布2020年报告，期内实现营业总收入100000元，同比增长20%，归母净利润50000元，同比增10%。
事件描述：公司披露2020年一季报。2020一季报单季实现营收100000元，YoY-20%；实现归母净利50000元，YoY-10%。<this_year><報別>单季实现营收100000元
2020年前一季报公司实现营业收入100000元，同比微增20%，归属母公司净利润50000元，同比上升10%，每股收益<每股盈利>元，基本符合我们预期。考虑到国
事件：公司发布2020年一季報一季報，报告期实现营业总收入100000，同比增长20%，归母净利润50000元，同比大幅扭亏为盈。
疫情致使若按新会计准则追溯调整，公司业绩短期承压，事件：天虹发布2020年一季報，报告期内公司实现营业收入100000元，同比-20%，实现归母净利润50000

- *上傳至阿里雲-實際應用的程式*

路徑:./data/taiwan-cuda/user_interactive_V2

- hsmm_templates.json: HSMM模型產出的模板
- input_data.csv:輸入結構化數據資料
- main.py:使用者介面功能的主程式
- outcome.txt:輸出財報文本敘述
- select_user_top5.json:輸出財報文本敘述所參考的模板



- *上傳至阿里雲-實際應用的程式*

使用方法:

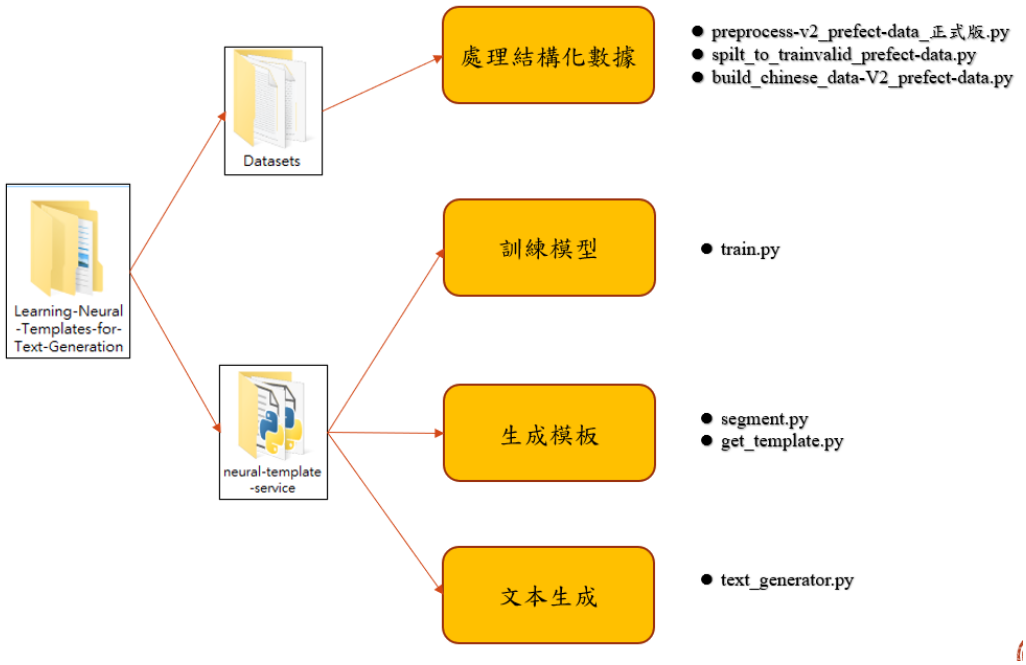
- ssh root@116.62.122.132
- docker exec -it taiwan-cuda /bin/bash
- cd data/taiwan-cuda/user_interactive_V2
- python3 main.py

```
[root@hczq-ios ~]# docker exec -it taiwan-cuda /bin/bash
root@cea4420bf09c:/# cd data/taiwan-cuda
root@cea4420bf09c:/data/taiwan-cuda# cd user_interactive_V2/
root@cea4420bf09c:/data/taiwan-cuda/user_interactive_V2# python3 main.py
```

```
root@cea4420bf09c:/data/taiwan-cuda/user_interactive_V2# python3 main.py
事件：<unk>、公司公布2020年一季報业绩快报，实现收入100000元，同比减少20%，归母净利润50000元，同比增长10%。
事件：公司发布2020年一季報报告，期内实现收入100000，同比减少20%，归母净利润-50000元，同比减少10%元。
事件：苏宁易购发布2020年一季報，报告期内公司实现营业收入100000元，同比下降20%，实现归母净利润为50000元，同比+10%。
事件：<unk>科技发布2020年一季報，一季報，期间实现营业收入100000元，同比增长<营业收入增长率>。归母净利50000元，同比增长10%。
疫情致使若按新会计准则追溯调整，公司业绩短期承压，事件：天虹发布2020年一季報，报告期内公司实现营业收入100000元，同比-20%，实现归母净利润50000元，同比下降10%。
```

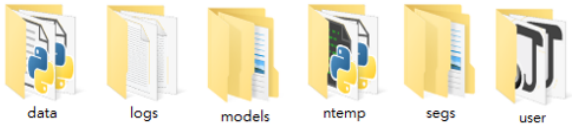
- *上傳至阿里雲-訓練模型的程式*

路徑:./data/taiwan-cuda/Learning-Neural-Templates-for-Text-Generation



- *上傳至阿里雲-訓練模型的程式*

執行環境:Python2.7、Pytorch0.3.1



- ./data，訓練資料
- ./logs,紀錄過程
- ./model，模型訓練後權重
- ./ntemp，HSMM模型
- ./segs，保存抽取後模板
- ./user，給用戶看.json

處理結構化數據

- preprocess-v2_prefect-data_正式版.py
- spilt_to_trainvalid_prefect-data.py
- build_chinese_data-V2_prefect-data.py

執行流程:

1.輸入原始資料(./Dataset/XXXX) 執行preprocess-v2 後生成 train_data.json
2. 輸入train_data.json執行split_to_trainvalid 後，會生成 train.json、vail.json(./dataset)
3. train.json、vail.json丟入執行build_Chinese_data-V2，會生成src_train.txt、train_tgt_lines.txt、train.txt、src_val.txt、val_tgt_lines.txt、val.txt(./Chinese_data) (src_*.txt文件是結構化的數據，tgt_*.txt文件是可讀的文本)

訓練模型

- train.py

生成模板

- segment.py
- get_template.py

文本生成

- text_generator.py

執行流程:

1. train.cfg、seg.cfg 調整參數
2.執行 train_and_seg.sh ,即可執行train 和 segment
3.執行write_all_result，生成結果