

HONG KONG UNIVERSITY OF

SCIENCE AND TECHNOLOGY

5014 REPORT

Chinese Word Correction

Author:

XU MIN

With Teammate:

LI LINRUI

Supervisor:

CHEN LEI

May 25, 2018

Contents

1	Abstract	2
2	Introduction	2
2.1	The Traditional Language Model for English	2
2.1.1	The First Part: Create Dictionary	2
2.1.2	The Second Part: Create Candidates	2
2.1.3	The Third Part: Give Result	3
2.2	The Basic Mathematics in Language Model	4
2.3	Differences Between English Correction and Chinese Correction .	5
3	Related Tools	5
3.1	Jieba	5
3.2	Cbow (Continuous Bag-of-Words)	6
3.3	Data	7
4	The Process of Chinese Word Correction	7
5	Results	9
5.1	Word Vector	9
5.2	Related Words	9
5.3	Sentence Correction	11
6	Conclusion and Future Forwarding	11

1 Abstract

Under the premise of statistics language model in English, our team takes use of the basis thought about language model and then analyzes the differences between English and Chinese, which could give hints to find suitable way in Chinese word correction. In this report, the data set is the Chinese corpus from Wikipedia and Sina Micro-Blog used as the Chinese language dictionary. Jieba package is used in data pre-processing. N-gram algorithm and Cbow algorithm help to judge the rationality of the sentence and finds out which word has the largest influence on the rationality. Then the tagged word derivatives candidates for model to choose the most suitable replacement word. By the way, LSTM model also be considered to predict the next word and enrich the candidate sets. The final result after correction will be decided by the rationality of the sentence changed by each candidate.

2 Introduction

At the beginning of natural language processing, the processing object is English and the scientist's thoughts were restricted in the method of how human learns language and they wanted to let computer to imitate human's brain. But it was unrealistic and there was no known algorithm suitable for computer to learn language by understanding the words' meaning. After about 20-year attempt on this kind of thoughts, the scientists hadn't got breakthrough and some people began to try another way, which used statistics model to do natural language processing. From rule to statistics, it was a great turning in the area of natural language processing.[3]

2.1 The Traditional Language Model for English

The process of early statistics language model in English is shown in Figure 1. The spelling mistake is the main problem which needs to be solved in English correction and the traditional model can solve the spelling mistake with one or two letters. The model can be divided into three parts:

2.1.1 The First Part: Create Dictionary

- 1 Collect abundant training texts and divide the sentence into words.
- 2 Count the frequency of each word.
- 3 Store the record as the model's dictionary

2.1.2 The Second Part: Create Candidates

- 1 Divide the sentence which needs to be corrected into words.

- 2 Divide the possible wrong word into a pair, which need to traverse all possibilities and store all pairs. Create the candidates by four ways for each pair. For example in Figure 1, where the pair is 'asb' and 'cefg', the first way is to delete the first letter in 'cefg'. The second way is to exchange the first letter with the second letter in 'cefg'. The third way is to change the first letter in 'cefg' into another letter in the alphabet, which needs to store all the changed forms as candidates. The forth way is to insert a letter after the first letter in 'cefg' and store all the possibilities. All the new forms above are stored as the pair's candidates.
- 3 Considering the situation that there are two letters wrong in the spelling, use the same four ways and iterate itself to produce the candidates, whose mistake distance is two.
- 4 Union the candidates created in 2 and 3.

2.1.3 The Third Part: Give Result

- 1 Compare the union set with dictionary and only hold the candidates which the dictionary has.
- 2 Find the frequency of each stayed candidate.
- 3 Output the candidate with highest frequency to replace the wrong word.

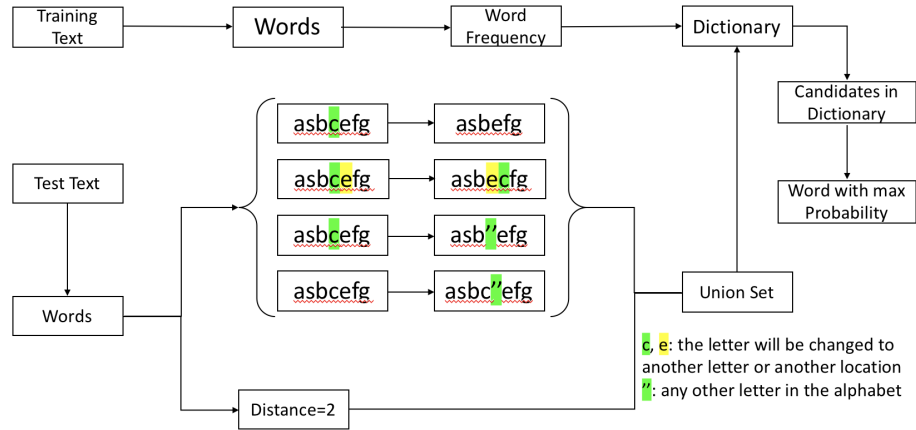


Figure 1: English Word Correction

2.2 The Basic Mathematics in Language Model

In order to judge a sentence's rationality, a reasonable way is to calculate the occurrence probability of the sentence. The related mathematics concept will be shown next.

Assume S is a sentence and is consist of a series of words $w_1, w_2, w_3 \dots, w_n$, which are in specific order and n is the length of the sentence. The probability of the sentence in the text is $P(S)$.

$$P(S) = P(w_1, w_2, w_3 \dots, w_n)$$

$$P(w_1, w_2, w_3 \dots, w_n) = P(w_1) * P(w_2|w_1) * P(w_3|w_1, w_2) \dots P(w_n|w_1, w_2, \dots, w_{n-1})$$

where $P(w_1)$ represent the occurrence probability of the word w_1 , $P(w_2|w_1)$ is the occurrence probability of the word w_2 with the premise that the w_1 is known, and so on. It is obvious that the occurrence probability of word w_n is dependent on all the words before w_n .

From the angle of calculation complexity, it is easy to calculate $P(w_1)$ and $P(w_2|w_1)$ but it begins to be hard to calculate after $P(w_3|w_1, w_2)$. In order to solve this problem, the scientist consider the Markov assumption which the word only have relationship with the last word and the formula becomes:

$$P(w_1, w_2, w_3 \dots, w_n) = P(w_1) * P(w_2|w_1) * P(w_3|w_2) \dots P(w_n|w_{n-1})$$

which is called as Bigram Model.

The next step is to calculate the conditional probability $P(w_i|w_{i-1})$. According to its definition:

$$P(w_i|w_{i-1}) = \frac{P(w_i, w_{i-1})}{P(w_{i-1})}$$

where $P(w_{i-1}, w_i)$ is the joint probability and $P(w_{i-1})$ is the edge probability. The two probabilities can be estimated from corpus. Assume w_{i-1}, w_i become a word pair and then count how many times the pair appears in the training text ($count(w_{i-1}, w_i)$). The frequency of w_{i-1} in the same text set is stored as $count(w_{i-1})$. The size of the corpus is called *textsiz*e and use the law of large numbers, we can get:

$$P(w_{i-1}, w_i) \approx \frac{count(w_{i-1}, w_i)}{textsiz}$$

$$P(w_{i-1}) \approx \frac{count(w_{i-1})}{textsiz}$$

As a result,

$$P(w_i|w_{i-1}) = \frac{count(w_{i-1}, w_i)}{count(w_{i-1})}$$

Above all, the most early statistics model is shown.

2.3 Differences Between English Correction and Chinese Correction

- 1 English only has 26 letters in the alphabet, which is easy to generate the word with similar form. However, Chinese has thousands of Chinese characters and will produce large candidate set.
- 2 Some English words have various pronunciation, but the pronunciation rarely have effect on the word meaning. In contrast, Chinese characters not only have two or more pronunciation, but also have different meanings with each pronunciation.
- 3 The boundary of Chinese is not clear. Speaking strictly, Chinese have no boundary. Because there is no clear delineation of words and the length of words is very short. For the factor, in order to do Chinese spelling detect, a part of or the whole context is considered as sample, such as a long phrase or a complete sentence, and a single word can also be focused on.
- 4 Under the social environment that the internet and media are rapidly developing, the internet users always create new collocations in Chinese and the dictionary is hard to catch up the language update.

So, the Chinese word correction could inherit the basic thought of English word correction, but in some areas, there needs some changes to cooperate with the differences.

3 Related Tools

3.1 Jieba

English words in a sentence is separated by natural boundary ‘ ’(space), but for Chinese, there is no boundary between two words. So, we need use segmentation algorithm to finish it. Jieba is a popular tool in Chinese word segmentation and support three types of segmentation mode (Accurate Mode, Full Mode and Search Engine Mode). In this report, accurate mode is adopted and it attempts to divide the sentence into the most accurate segmentation, which is suitable for text analysis. The Figure 2 is an example of Jieba word segmentation and the second line in the graph is the vocabulary of word judged by Jieba too.

因/M300/仪表/模块/存在/安全隐患/, /众泰/汽车/将/召回/部分/M300/车型
/c /eng /n /n /v /i /x /nz /n /d /v /n /eng /n

Figure 2: The Sample of Jieba

Jieba’s theory mainly be composed by directed acyclic graph (DAG) and dynamic programming. The DAG is built by all possible word combinations

in the sentence. In order to have efficient word graph scanning, Jieba also use the concept of prefix dictionary. After get all possible word combination, dynamic programming can help to find the most probable combination on the word frequency. What's more, for unknown words, a HMM-based model is used with the Viterbi algorithm. Thus Jieba can give out the most reasonable word segmentation of a sentence.[1]

3.2 Cbow (Continuous Bag-of-Words)

Cbow is a specific CSLM (Continuous space language model) and can be trained to produce word vectors. In this report, the Cbow model is trained by the Chinese corpus from Wikipedia and Sina Micro-blog, which size is about 1GB. Figure 3 gives the general architecture of the Cbow model.

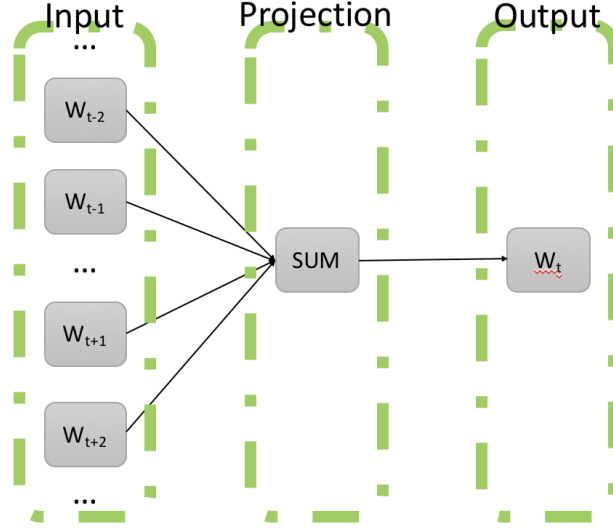


Figure 3: The Architecture of Cbow Model

In Cbow model, w_t is the forecast target and the premise is the known context of w_t . It is obvious that the Cbow model has three layers and doesn't have hidden layer, which is always in the ordinary neural network model. The three layers in the Cbow model is:

- **Input Layer:** Assume there are c words before w_t and c words after w_t , then the $2 * c$ words make up vectors as input parameters in this layer.
- **Projection Layer:** The vectors from input layer are summed up in this layer and the result will be sent to the output layer.
- **Output Layer:** There is a binary Huffman tree in this layer. Every leaf node in this tree is a word in training corpus and the occurrence frequency

The goal function which needs to be solved is :

In the function, there is

The parameters produced during the calculation can be modified during training process and the training way is stochastic gradient descent. The details of process to calculate the formulation will not be shown in this report.[2]

3.3 Data

达尔文 英语又译作 达尔温 北部的首府大城市人口约千余人属于热带气候 受到 雷暴 和龙卷风的 侵袭 最有 记录的 热带 气旋 发生在 英国 记者 对 达尔文 进行了 两次 采访 此后 在 第二次世界大战 中 达尔文 遭受 过 多次 轰炸 因此 达尔文 被 称为 是 澳大利亚 多元文化 的首府 由于 它 距离 澳洲 最近 所以 是 重要的 进出口 口主要 出口 活牲畜 和 矿物 历史 帕尔默 斯顿 议会 址地 正式 是 于年 被 英国 生物学家 詹姆斯 斯尼 有人 定居 在此 属于 南澳大利亚 州 以 当时 英国 的 命名 为 帕尔默 斯顿 年 会议 址地 正式 定名为 达尔文 市 被 当地 破坏了 城市 毁灭 人 炸死 以后 又 受到了 一系列 的 轰炸 同时 伴随 灾难 导致 数千人 死亡 的城市 建筑 被 摧毁 人们 被迫 空 搬离 以后 城市 几乎 全部 重建 年代 在南 方 又 建立了一个 卫星城 现在 在香港 的一个 组织 里 代表团 提出 租借 达尔文 作为 建设 新香港 的 建议 但 最终 不了了之 打通了 澳洲 南北 的交通 这里 有到达 达 尔东 帝力 地理 和 气候 都 比 达尔文 好 达尔文 距 澳洲 首都 堪培拉 千公里 距离 达尔东 帝力 只有 千公里 距 印度 尼西亚 首都 雅加达 千公里 即使 距离 菲律宾 首都 马尼拉 也只有 千公里 距 新加坡 千公里 达尔文 是 热带气候 一年之中 只分为 雨季 和旱季 旱季 为月 至月和 月 是最 长的 月份 温度为 雨季 时最高 有 热带 风暴 降水 最多 的是 月至月 这段时间 期间 雨 量 超过 常 发生 旱季 月份 平均 最高 温度 为 月份 平均 最低 温度 为 月份 平均 最高 温度为 月份 平均 最低 温度为 全年 平均 降水 雨量 最高 的 月份 平均 降水 为 全年 降 雨 日 政府 北部 地区 选举 的 议会 就是 在 达尔文 达尔文的 行政 机构 为 年后 立即 由 议会 执行 市长 由 市长 和 市民 组成 四个 选区 每个 选区 选 举 一名 市议员 现任 市长 是 达尔文 商业 业 采矿 业 每年 产值 为 亿美元 主要产品 为 黄金 铝土 铜中的 油气田 和 海底 锰矿 的 产地 居民 从事 旅游业 旅游 业 还有 很好的 扩展 前景 为 维护 达尔文 治安 澳大利亚 在 达尔文 的 驻军 在 年 已经 达到 使得 达尔文 的 港口 地位 日益 重要 有两个 重要 项目 启动一 个 是 一个 大型 的 计划 预计 年 报告 方案 是 修建 托托山 国际 机场 另一个 是 一个 城市 项目 经过 完工 建成 后 将 成为 整个 澳 洲 全 国 学 校 中小 学 学 生 学 习 的 中心 在 学生 中 其中 一大部分 是 来自 亚洲 主要是 是 墨尔本 悉尼 等 城市 本地 也有 许多 分校区 小 学校在 达尔文 市 学生 生活 文化 设施 很 美 观 是 通往 亚洲 的 大门 是 主要的 典型 全市 人口 包括 有 个 民族 米切路 族 是 总 会 社 和 餐 馆 集中 的 地方 有许多 露天 餐馆 每年 举办 达 尔文 节 包 括 放 烟花 音乐 会 等等 每年 月 月 啤酒 啤 酒 喝 做的 帆船 比赛 还有 达尔文 杯 赛 马 以及 月 举行 有一个 大型 水族馆 的 印度 洋 和 澳大利 亚 珍 珠 业 展览 店 也 坐落在 达尔文 友好 城市 希腊 卡 萨 利 诺 岛 美国 安 利 科 雷 斯 印度 尼 西 亚 安 汉 大 陆 和 韩国 海 口 东 京 达 尔 文 参 考 文 献 外部 链接 达尔文 网站 北 部 地 方 政府 网 站 达尔文 的 节日

There is 310,211 fields in the Wikipedia data set and 561,028 Weibo contents.

From the Figure 4, the process to realize the Chinese word correction has been shown. Something need to be mentioned that the Cbow model appeared in the graph has been trained by the training data from Wikipedia and Sina Micro-blog, so it can be used in the process directly.


```
{'author': 'asd紫衣', 'url': 'http://weibo.com/6276804601/FEq78gIv6', 'title': 'asd紫衣', 'pubSource': '新浪微博', 'content': '这天我差点出事哦，我去买烧烤，公交车，我看到个人像鬼，多少路我忘记了，后来跟着个杨洋的什么广告，我心定了//@张若昀：一杯不会止一杯', 'pubTime': '1515967706', 'last_update': '2018-01-15 09:29:57', 'id': '-9223363810167202147', 'copyrightSource': 'asd紫衣'}
```

```
{'author': '故事与酒缸', 'url': 'http://weibo.com/5759812287/FDLHA0DtI', 'title': '故事与酒缸', 'pubSource': '新浪微博', 'content': '【广菲克产销分离JEEP国产在即】在4月20日举办的上海国际车展上，广菲克展出了包括菲翔米兰典藏版、致悦Cross概念车、致悦Mefisto概念车、菲翔跃享版、致悦豪华运动版和菲亚特500等在内的8款车型。而这也是广菲克销售公司挂牌前，在“家门口”的第一次联合演出。', 'pubTime': '1515597336', 'last_update': '2018-01-11 01:31:56', 'id': '-9223357131622972986', 'copyrightSource': '故事与酒缸'}
```

```
{'author': '丘份碗66', 'url': 'http://weibo.com/6142925293/G0mQ3azgA', 'title': '丘份碗66', 'pubSource': '新浪微博', 'content': '【昌河汽车将推2款新车】与北汽集团整车事业本部副部长、江西昌河汽车有限责任公司副总经理廖雄辉沟通后获悉：昌河汽车在上海车展发布了全新福瑞达M50s并公布了首款SUV车型，两款车都将年内上市。', 'pubTime': '1517050815', 'last_update': '2018-01-29 22:48:53', 'id': '-9223352888553609071', 'copyrightSource': '丘份碗66'}
```

```
{'author': 'sweet-cake糯米小姐', 'url': 'http://weibo.com/1973006845/G04E28Irv', 'title': 'sweet-cake糯米小姐', 'pubSource': '新浪微博', 'content': '生日蛋糕目前还是延续先前的活动哦 有过交易记录的老顾客预订生日蛋糕可享立减50元优惠 新顾客预订可减20元 限预订8寸以上生日蛋糕，每人限定一个，不可与其它活动同享，如需外送，外送费按实际地址收取，武汉的亲可以来菱角湖万达这边自提 活动只接到2.13.', 'pubTime': '1516884072', 'last_update': '2018-01-26 03:19:06', 'id': '-9223352209235844772', 'copyrightSource': 'sweet-cake糯米小姐'}
```

```
{'author': 'Caroline_Nine', 'url': 'http://weibo.com/1875936727/FFdIjvCCL', 'title': 'Caroline Nine', 'pubSource': '新浪微博', 'content': '消防车🚒//@APP菌：哈哈哈哈哈哈哈这个恋爱还不够生气的//@一个阿呆仔：学猪叫那个真笑出猪声[允悲]//@梨园西池水：哈哈哈哈哈哈哈哈哈//@太皇太后您有喜啦：哈哈哈哈哈哈哈怎么这么好笑_>_<', 'pubTime': '1516422278', 'last_update': '2018-01-20 16:36:19', 'id': '-9223330358968196918', 'copyrightSource': 'Caroline_Nine'}
```

Figure 5: Sina Data Format

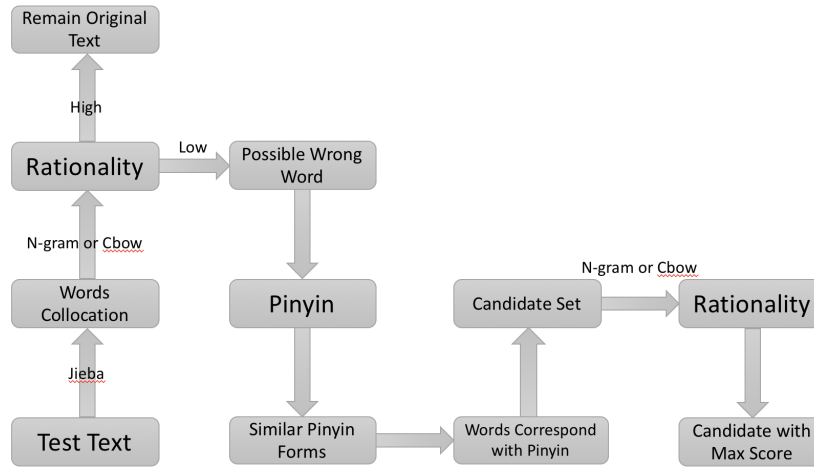


Figure 6: The Process of Chinese Word Correction

- Use Jieba to do Chinese word segmentation on the test text.
- Input the word segmentation result into the trained model and the model gives the rationality of the words collocation.
- Judge the rationality of each word. If a word is not in our dictionary built from our corpus, distribute the word into spelling error word. If a word is in our dictionary and only has one character, then check it whether in stop words list. Each stop word is a single character and can be used alone in a sentence and its frequency is very high. For these words, although they are correct words, still don't do anything to them. The last situation is that the word is in our dictionary and only has one character, not in stop word list, and one of its adjacent words is also a single character. Then combine the two words together.
- Translate the word which has got from the last step into Pinyin format.
- Create the similar Pinyin forms. Only consider the forms with one letter difference, so the methods which could be adopted include deleting a letter, adding a letter and changing a letter. Compared with English word correction, it is not feasible for Chinese word correction to exchange two letters in Pinyin because Pinyin requests strict rules and has fewer available candidates. After traverse all the change methods, the similar Pinyin set can be done.
- Based on the similar Pinyin set, find all the possible words which are corresponding with the Pinyin in the set and collect these words as candidate set of replacement words.
- Use the words in the candidate set to replace the word from test text one by one and get their rationality scores in the text. Choose the candidate with the highest score as the final result to correct the text.

5 Results

5.1 Word Vector

After input a Chinese word, using command `model['word']` will output the word's vector and the example is shown in Figure 7.

5.2 Related Words

After input a Chinese word, using command `model.similar_by_word('word')` will give all the words with high similarity and the example is shown in Figure 8.

```

In [718]: 1 model['中国']

/Users/llr/anaconda2/envs/python3/lib/python3.6/site-packages/ipykernel_launcher.
precreated `__getitem__` (Method will be removed in 4.0.0, use self.wv.__getitem__(
"""Entry point for launching an IPython kernel.

Out[718]: array([-1.7964096e-01, -1.3616471e+00, -1.3694850e-01,  4.1055191e-01,
-1.6951275e+00, -5.9313983e-01,  8.0688763e-01,  5.4034400e-01,
-3.4143579e-01,  4.1705149e-01, -9.4484448e-02, -1.2403933e+00,
 1.3862280e+00, -6.0323185e-01,  1.2341700e+00,  8.3426237e-01,
 1.6828422e-02, -1.8615024e+00, -1.7387542e-01,  2.5797661e-02,
 1.5448942e+00,  7.6141310e-01, -5.8893114e-01, -1.4503571e-01,
-7.3553777e-01, -3.7667838e-01,  6.5697658e-01, -1.7631556e-01,
 5.8166105e-01,  9.4590360e-01,  1.0090262e+00,  1.9395365e-01,
-8.2447678e-01,  7.8445345e-01,  7.4826592e-01, -5.5752254e-01,
 1.4695696e+00,  3.9412490e-01, -4.0946892e-01,  1.5772720e-01,
-3.6689785e-01, -5.9891373e-01,  2.2042398e+00,  3.8541451e-01,
 2.0035009e-01,  2.3759880e+00,  1.0337848e-01,  1.0145075e-01,
-7.8084487e-01, -4.6236676e-01,  7.8955799e-01, -6.7722082e-01,
 2.4268526e-01,  3.5147718e-01,  9.6108025e-01,  5.4061139e-01,
-3.7216380e-01,  9.5521593e-01,  9.0507829e-01,  7.5663686e-01,
-1.4013035e+00,  7.4981403e-01,  9.5363903e-01,  9.2508042e-01,
-1.3971719e+00, -7.6749462e-01, -3.8258302e-01, -9.8785585e-01,
 1.0476898e-03, -5.6448871e-01, -2.0408943e+00,  6.1312097e-01,
 1.4536954e-01, -3.0298781e-01,  1.3627682e+00,  2.0431620e-01,
-2.5009730e-01,  1.6404577e-02, -1.1580597e+00,  1.4962924e+00,
 1.1524370e-01,  7.8622109e-01,  1.7723812e+00, -5.0246876e-01,
-6.9727331e-01,  3.3248937e-01,  8.3543873e-01,  8.1210715e-01,
 4.2032608e-01, -6.0366189e-01,  4.9439505e-01, -6.7591392e-02,
 1.0581459e+00,  1.7908603e-01, -1.3195149e+00,  8.6309440e-02,
-6.6436511e-01,  5.8624017e-01, -1.1006351e+00, -3.9206645e-01],
dtype=float32)

```

Figure 7: Word Vector

```

In [719]: 1 model.similar_by_word('中国')

/Users/llr/anaconda2/envs/python3/lib/python3.6
precreated `similar_by_word` (Method will be rem
"""Entry point for launching an IPython kerne

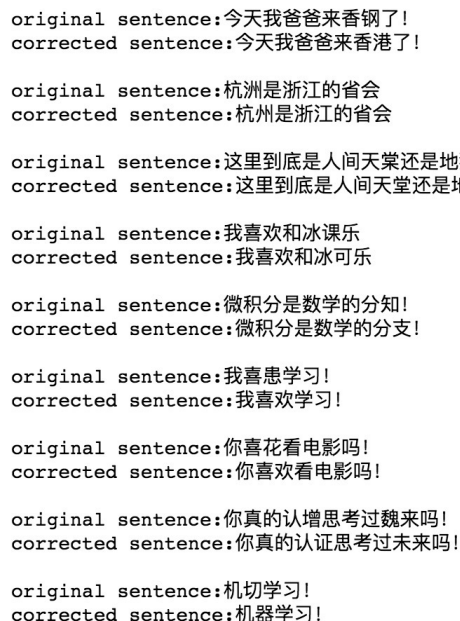
Out[719]: [('北京', 0.6465203762054443),
('上海', 0.6366360783576965),
('台湾', 0.5958433151245117),
('内地', 0.5902783274650574),
('越南', 0.5884591937065125),
('朝鲜半岛', 0.5845679044723511),
('亚洲', 0.5787181854248047),
('东亚', 0.5729079842567444),
('我国', 0.5721892714500427),
('日本', 0.5702469348907471)]

```

Figure 8: Similar Word

5.3 Sentence Correction

After the model finishes training and the candidate word is selected, the result of using candidate word to replace the original word is output and seems good in Figure 9. Some other test sentences have been tested but the model don't give the correct answer. The estimation of the success rate is about 65%.



```
original sentence:今天我爸爸来香钢了!  
corrected sentence:今天我爸爸来香港了!  
  
original sentence:杭州是浙江的省会  
corrected sentence:杭州是浙江的省会  
  
original sentence:这里到底是人间天堂还是地狱!  
corrected sentence:这里到底是人间天堂还是地狱!  
  
original sentence:我喜欢和冰课乐  
corrected sentence:我喜欢和冰可乐  
  
original sentence:微积分是数学的分知!  
corrected sentence:微积分是数学的分支!  
  
original sentence:我喜患学习!  
corrected sentence:我喜欢学习!  
  
original sentence:你喜花看电影吗!  
corrected sentence:你喜欢看电影吗!  
  
original sentence:你真的认增思考过魏来吗!  
corrected sentence:你真的认证思考过未来吗!  
  
original sentence:机切学习!  
corrected sentence:机器学习!
```

Figure 9: Final Result

6 Conclusion and Future Forwarding

As a conclusion, there still have large room for improvement in Chinese word correction. Although the common expression in Chinese could be recognized by the statistics language model well and the model can correct them in right way, the unbounded nature of Chinese terms causes infinite combinations in the language and the multiple meanings of singleton or words leads such a hard work for machine to learn the Chinese. By the way, during the period in doing the project, I find that the Chinese dictionary on the Internet is incomplete, which causes some defects in training stage. So we need to replenish the dictionary before training the model. What's more, we can not promise that the text used in training the model is formal and accurate. Because even the texts are got from Wikipedia, there still have many gibberish and some other languages. This can be explained that, to some extent, Korean and Japanese have common

points with Chinese and are not easy to clean from Chinese totally. One of the solutions of the situation is to let the training text set as large as possible, which means that the individual computer could be hard to train a good model in this area.

Despite the difficulties in the Chinese word correction, there still have bright future in this area. There are many developed models likes NNLM, N-gram and Cbow in this area and if choose suitable train set, the result could be well. On the other hand, Chinese word correction is very useful in daily life. Because it is not unusual that people type wrong Pinyin and send wrong sentences to others during social activity, which may cause some confusion or misunderstanding to relationship. At that time, people really want that there are some reliable tools to correct their words and avoid awkward. As consequences, Chinese word correction is being needed by people and have market in China, which will motivate the development of this technology.

Back to the model in our project, the initial thought not only consider the Cbow model to train the data set, but also want to use LSTM to predict the next word after some parts of texts and the predicted word will be added into the candidate set, which make the candidate set consider more factors and be more abundant so that give more accurate correction. However, restricted by the time, we didn't add the LSTM into our project but we think it is worth to have an attempt on combination between Cbow and LSTM.

References

- [1] fxsjy. Jieba. <https://github.com/fxsjy/jieba>. Last commit Aug 28, 2017.
- [2] Zhaoyi Guo, Xingyuan Chen, Peng jin, and Si-Yuan Jing. Chinese spelling errors detection based on cslm. *IEEE/WIC/ACM*, 2015.
- [3] WU Jun. *Beauty of Mathematics*. Posts and Telecommunications Press, 2012.06.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.