Guanlin Li (/)

Notes on Neubig's Tutorial 0~3: Part C

This post is about a natural language processing task or application titled **word segmentation (WS)**, which is very essential in Chinese and Japanese etc. language processing. Usually, WS will function as the *very beginning part* in the processing pipeline of raw text. The content of the post hopes to cover the following 3 parts: **1). the basic definition of WS**, to make strangers familiar with this task; **2). problem formulation**, which deals with how we could use the *probability score of a language model* and transfer the WS problem as a classic **dynamic programming** problem algorithmically; **3). Viterbi algorithm** to solve the dynamic programming problem.

1. What is word segmentation?

Actually, this question has been introduced in the tutorial slides. In the third page, Neubig says “**word segmentation adds spaces between words**”. This is a very straightforward explanation. However, why languages like Chinese and Japanese need word segmentation? The reason can bear both simple and complex explanations.

The *simple reason* is that languages with characters (象形文字) instead of letters will not use space to explicitly distinguish word meanings.

The *complex reason* is that languages have its own **writing systems** (书写系统), which have been evolved to incorporate some regularities and rules to make written language easy to read and understand. The writing systems of Chinese and other character-based languages does not require spaces to separate words with independent meanings. Here, you can know that: **every language phenomenon is evolvable according to social development and convention formulation**. So if you understand social culture better, you will be a good language user.

More formally, we can define the word segmentation problem as a **structured prediction** problem. The **input** of the prediction problem is a natural language sequence which is usually a sentence, $w_1, w_2, \dots, w_n$, and the **output** is the input sequence with augmented separators inserted between characters, $w_1, w_2, s, \dots, s, w_n$, where s denotes separator.

Comment.

Here to be more clearer, we regularize the usage of the word: **word** and **character**, and we use “word” to denote “phrase in Chinese” like “祖国”, “花朵”, and use “character” to denote each specific Chinese character like “祖”, “国” etc.

Let us use a more specific example, the input could be “山东省因居太行山以东而得名”, and the output of the WS system would be “山东省s因s居s太行山s以东s而s得名”, where “s” is the separator as well.

Structured prediction.

Structured prediction (https://en.wikipedia.org/wiki/Structured_prediction) (SP) is a prediction problem which predicts the structure in the input. More specifically, the input would be a combinatorial structure, i.e. a sequence, a graph structure etc. and the output would a combinatorial structure as well, which augments the input structure with latent structural informations, e.g. labels in POS tagging, parse trees in syntactic parsing, links in link prediction over graph etc. To be honest, SP problem can be much complex and does not have a sound definition because of the diversity of all kinds of SP problems. If you are interested in SP problems, you can read Hal Daume’s PhD thesis here (http://www.umiacs.umd.edu/~hal/docs/daume06thesis.pdf).

2. Word segmentation: problem formulation

This part first casts WS problem as a machine learning problem with certain training corpus. We still use a model $P_\theta(y|x)$ to describe the probabilistic relationship between the input sentence x and the output augmented sentence y.

Then, we regard the problem of learning as estimating the parameters of the model, θ; and prediction as finding the best y regarding to the model $y^* = argmax_y P_\theta(y|x)$ given x. Since y has combinatorial structure, we should use a search algorithm to smartly traverse the search space and effectively find the best y^*.

We will specifically discuss how to use a language model as P_θ to score the ys in output space, and an efficient dynamic programming algorithm Viterbi algorithm to search the best y^*.

Word segmentation and other structured prediction problem, e.g. named entity recognition, part-of-speech tagging etc. can be formulated as a two-stage process:

Caveat.

I want to insert a caveat here. Since the two-stage solution to structured prediction problems is not the only one we can embrace upon, however, the two-stage paradigm is the most famous and popular one since it has motivate many research development in probabilistic graphical models (https://en.wikipedia.org/wiki/Graphical_model) which is an approach with a group of algorithms and theories to do probabilistic machine learning. There is another way to do structured prediction which integrates learning and inference (search) within a same stage, it is titled SEARN (http://www.umiacs.umd.edu/~hal/searn/) as a nickname for "search and learn", which brings us with a sequential decision making (kind of reinforcement learning) view of structured prediction. (PS: the PhD thesis I mentioned above is on this topic, actually the author is the inventor of SEARN, bravo!)

- **Learning stage**: we assume a model with the form $P_\theta(x, y)$ or $P_\theta(y|x)$ with parameter θ, to learn a scoring rule (which is the probability of the conditional $P(y|x)$), so we end up learning a scoring rule over output space (search space) $\mathcal{Y}$.
- **Decoding stage**: given an estimated model P_θ, if we are provided with a new x which is not in the training corpus, we should now predict the corresponding y. To do this, we should solve the optimization problem $y^* = argmax_y P_\theta(y|x)$. This is sometimes called a **decoding** problem or search **problem**, where you need to search over exponentially possible combinatorial structures of y.

Note 1.

Here, someone may be curious about why $\boxed{P_\theta(x, y)}$ can form a scoring rule over the output space $\mathcal{Y}$, since it is a probability score over the combined space of both $\mathcal{X}$ and $\mathcal{Y}$. Since we can transform P_θ to be the conditional probability by dividing the marginal $P(x) = \sum_y P_\theta(x, y)$ to get our dreaming formula $P(y|x) = \frac{P_\theta(x,y)}{\sum_y P_\theta(x,y)}$. ✓

However, we can find that $argmax_y P(y|x) = argmax \frac{P_\theta(x,y)}{P(x)} = \underset{y}{argmax} P_\theta(x, y)$ since x is given, thus a constant here with respect to the $argmax_y$ operator.

I think the above discussion has made it clear how to model a structured prediction problem, or specifically, a WS problem. Next in this section, we are going to consider **Chinese** word segmentation and use a model $P_\theta(y|x)$ to learn the scoring rule and introduce the decoding algorithm - Viterbi algorithm - in next section.

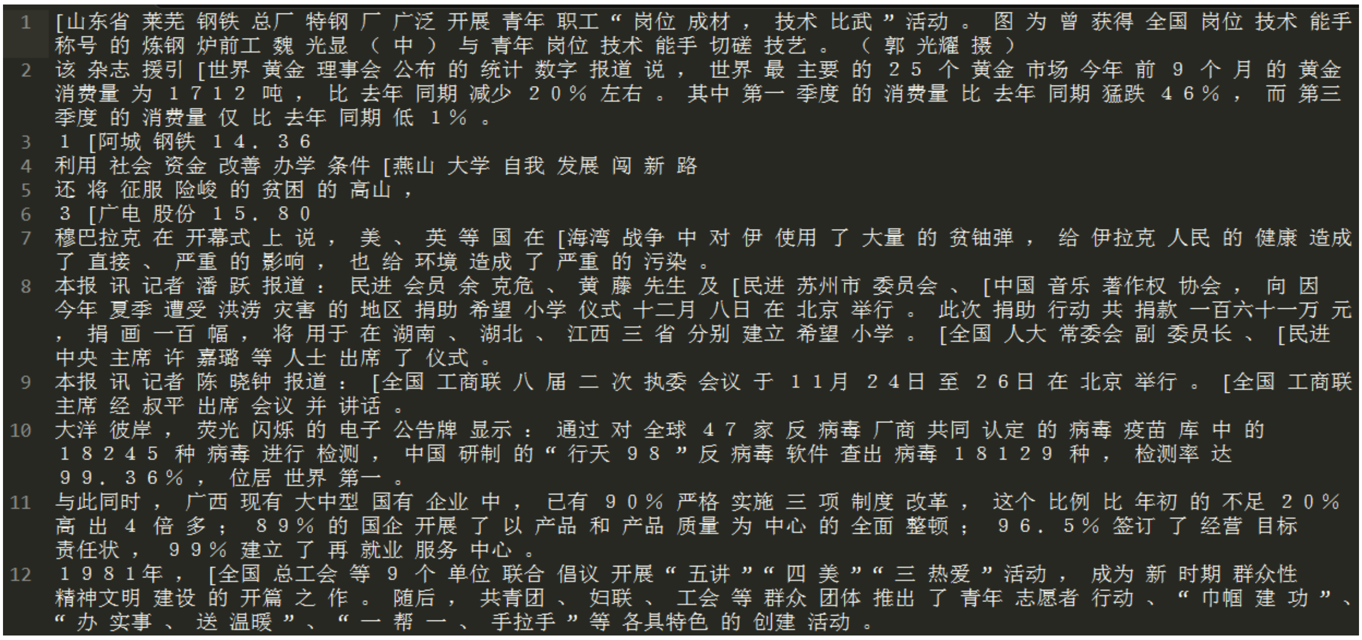
Note 2. 判别模型与生成模型

The two probability forms introduced in the **Learning stage** consist of two paradigms of predictive modeling: the **discriminative model** and the **generative model**. This two words "discriminative" and "generative" has some ambiguity in some claims or statements. Considering the statement, "Discriminative model is not just probabilistic classifiers and can still have generative power", how you understand it? My division of discriminative/generative model is that when you are doing predictive modeling (maybe you can sometimes call it supervised learning) - that is you are given an input x and an output y, you learn a model across all such training samples, and when new x is given, the model and its related prediction algorithm is asked to predict y - if you model a joint distribution over both input and output space $P(x, y)$ you are using a generative model, or if you model a conditional distribution with examples in input space as condition $P(y|x)$ you are doing discriminative modeling.

So the generative models can generate samples in input space as well whereas the discriminative models can only generate sample in output space, which could be class labels or a combinatorial structure, I say in the latter case, the discriminative model has generative power.

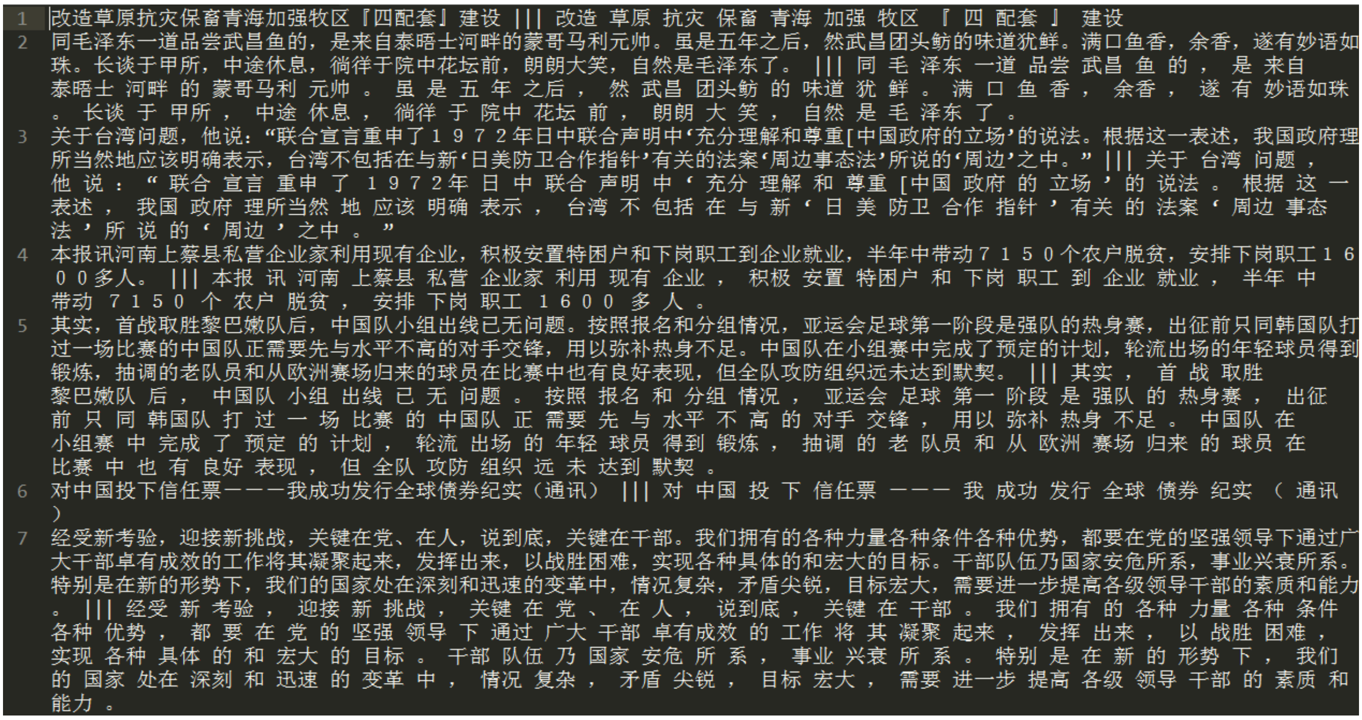
Firstly, let us be familiar with our training data, how they looked?

- The training data is just Chinese sentences which have been segmented.



```
[山东省 莱芜 钢铁 总厂 特钢 厂 广泛 开展 青年 职工 “ 岗位 成材 ， 技术 比武 ” 活动 。 图 为 曾 获得 全国 岗位 技术 能手 称号 的 炼钢 炉前工 魏 光显 （ 中 ） 与 青年 岗位 技术 能手 切磋 技艺 。 （ 郭 光耀 摄 ）
该 杂志 援引 [世界 黄金 理事会 公布 的 统计 数字 报道 说 ， 世界 最 主要 的 ２５ 个 黄金 市场 今年 前 ９ 个 月 的 黄金 消费量 为 １７１２ 吨 ， 比 去年 同期 减少 ２０％ 左右 。 其中 第一 季度 的 消费量 比 去年 同期 猛跌 ４６％ ， 而 第三 季度 的 消费量 仅 比 去年 同期 低 １％ 。
１ [阿城 钢铁 １４．３６
利用 社会 资金 改善 办学 条件 [燕山 大学 自我 发展 闯 新 路
还 将 征服 险峻 的 贫困 的 高山 ，
３ [广电 股份 １５．８０
穆巴拉克 在 开幕式 上 说 ， 美 、 英 等 国 在 [海湾 战争 中 对 伊 使用 了 大量 的 贫铀弹 ， 给 伊拉克 人民 的 健康 造成 了 直接 、 严重 的 影响 ， 也 给 环境 造成 了 严重 的 污染 。
本报 讯 记者 潘 跃 报道 ： 民进 会员 余 克危 、 黄 藤 先生 及 [民进 苏州市 委员会 、 [中国 音乐 著作权 协会 ， 向 因 今年 夏季 遭受 洪涝 灾害 的 地区 捐助 希望 小学 仪式 十二月 八日 在 北京 举行 。 此次 捐助 行动 共 捐款 一百六十一万 元 ， 捐 画 一百 幅 ， 将 用于 在 湖南 、 湖北 、 江西 三 省 分别 建立 希望 小学 。 [全国 人大 常委会 副 委员长 、 [民进 中央 主席 许 嘉璐 等 人士 出席 了 仪式 。
本报 讯 记者 陈 晓钟 报道 ： [全国 工商联 八 届 二 次 执委 会议 于 １１月 ２４日 至 ２６日 在 北京 举行 。 [全国 工商联 主席 经 叔平 出席 会议 并 讲话 。
大洋 彼岸 ， 荧光 闪烁 的 电子 公告牌 显示 ： 通过 对 全球 ４７ 家 反 病毒 厂商 共同 认定 的 病毒 疫苗 库 中 的 １８２４５ 种 病毒 进行 检测 ， 中国 研制 的 “ 行天 ９８ ” 反 病毒 软件 查出 病毒 １８１２９ 种 ， 检测率 达 ９９．３６％ ， 位居 世界 第一 。
与此同时 ， 广西 现有 大中型 国有 企业 中 ， 已有 ９０％ 严格 实施 三 项 制度 改革 ， 这个 比例 比 年初 的 不足 ２０％ 高 出 ４ 倍 多 ； ８９％ 的 国企 开展 了 以 产品 和 产品 质量 为 中心 的 全面 整顿 ； ９６．５％ 签订 了 经营 目标 责任状 ， ９９％ 建立 了 再 就业 服务 中心 。
１９８１年 ， [全国 总工会 等 ９ 个 单位 联合 倡议 开展 “ 五讲 ” “ 四 美 ” “ 三 热爱 ” 活动 ， 成为 新 时期 群众性 精神文明 建设 的 开篇 之 作 。 随后 ， 共青团 、 妇联 、 工会 等 群众 团体 推出 了 青年 志愿者 行动 、 “ 巾帼 建 功 ” 、 “ 办 实事 、 送 温暖 ” 、 “ 一 帮 一 、 手拉手 ” 等 各具特色 的 创建 活动 。
```

- The test data consists of lines, each with an test example and its gold reference separated by `|||`.



```
改造草原抗灾保畜青海加强牧区『四配套』建设 ||| 改造 草原 抗灾 保畜 青海 加强 牧区 『 四 配套 』 建设
同毛泽东一道品尝武昌鱼的，是来自泰晤士河畔的蒙哥马利元帅。虽是五年之后，然武昌团头鲂的味道犹鲜。满口鱼香，余香，遂有妙语如珠。长谈于甲所，中途休息，徜徉于院中花坛前，朗朗大笑，自然是毛泽东了。 ||| 同 毛 泽东 一道 品尝 武昌 鱼 的 ， 是 来自 泰晤士 河畔 的 蒙哥马利 元帅 。 虽 是 五 年 之后 ， 然 武昌 团头鲂 的 味道 犹 鲜 。 满 口 鱼 香 ， 余香 ， 遂 有 妙语如珠 。 长谈 于 甲所 ， 中途 休息 ， 徜徉 于 院中 花坛 前 ， 朗朗 大 笑 ， 自然 是 毛 泽东 了 。
关于台湾问题，他说：“联合宣言重申了１９７２年日中联合声明中‘充分理解和尊重[中国政府的立场’的说法。根据这一表述，我国政府理所当然地应该明确表示，台湾不包括在与新‘日美防卫合作指针’有关的法案‘周边事态法’所说的‘周边’之中。” ||| 关于 台湾 问题 ， 他 说 ： “ 联合 宣言 重申 了 １９７２年 日 中 联合 声明 中 ‘ 充分 理解 和 尊重 [中国 政府 的 立场 ’ 的 说法 。 根据 这 一 表述 ， 我国 政府 理所当然 地 应该 明确 表示 ， 台湾 不 包括 在 与 新 ‘ 日 美 防卫 合作 指针 ’ 有关 的 法案 ‘ 周边 事态 法 ’ 所 说 的 ‘ 周边 ’ 之中 。 ”
本报讯河南上蔡县私营企业家利用现有企业，积极安置特困户和下岗职工到企业就业，半年中带动７１５０个农户脱贫，安排下岗职工１６００多人。 ||| 本报 讯 河南 上蔡县 私营 企业家 利用 现有 企业 ， 积极 安置 特困户 和 下岗 职工 到 企业 就业 ， 半年 中 带动 ７１５０ 个 农户 脱贫 ， 安排 下岗 职工 １６００ 多 人 。
其实，首战取胜黎巴嫩队后，中国队小组出线已无问题。按照报名和分组情况，亚运会足球第一阶段是强队的热身赛，出征前只同韩国队打过一场比赛的中国队正需要先与水平不高的对手交锋，用以弥补热身不足。中国队在小组赛中完成了预定的计划，轮流出场的年轻球员得到锻炼，抽调的老队员和从欧洲赛场归来的球员在比赛中也有良好表现，但全队攻防组织远未达到默契。 ||| 其实 ， 首 战 取胜 黎巴嫩队 后 ， 中国队 小组 出线 已 无 问题 。 按照 报名 和 分组 情况 ， 亚运会 足球 第一 阶段 是 强队 的 热身赛 ， 出征 前 只 同 韩国队 打 过 一 场 比赛 的 中国队 正 需要 先 与 水平 不 高 的 对手 交锋 ， 用以 弥补 热身 不足 。 中国队 在 小组赛 中 完成 了 预定 的 计划 ， 轮流 出场 的 年轻 球员 得到 锻炼 ， 抽调 的 老 队员 和 从 欧洲 赛场 归来 的 球员 在 比赛 中 也 有 良好 表现 ， 但 全队 攻防 组织 远 未 达到 默契 。
对中国投下信任票———我成功发行全球债券纪实（通讯） ||| 对 中国 投 下 信任票 ——— 我 成功 发行 全球 债券 纪实 （ 通讯 ）
经受新考验，迎接新挑战，关键在党、在人，说到底，关键在干部。我们拥有的各种力量各种条件各种优势，都要在党的坚强领导下通过广大干部卓有成效的工作将其凝聚起来，发挥出来，以战胜困难，实现各种具体的和宏大的目标。干部队伍乃国家安危所系，事业兴衰所系。特别是在新的形势下，我们的国家处在深刻和迅速的变革中，情况复杂，矛盾尖锐，目标宏大，需要进一步提高各级领导干部的素质和能力。 ||| 经受 新 考验 ， 迎接 新 挑战 ， 关键 在 党 、 在 人 ， 说到底 ， 关键 在 干部 。 我们 拥有 的 各种 力量 各种 条件 各种 优势 ， 都 要 在 党 的 坚强 领导 下 通过 广大 干部 卓有成效 的 工作 将 其 凝聚 起来 ， 发挥 出来 ， 以 战胜 困难 ， 实现 各种 具体 的 和 宏大 的 目标 。 干部 队伍 乃 国家 安危 所 系 ， 事业 兴衰 所 系 。 特别 是 在 新 的 形势 下 ， 我们 的 国家 处在 深刻 和 迅速 的 变革 中 ， 情况 复杂 ， 矛盾 尖锐 ， 目标 宏大 ， 需要 进一步 提高 各级 领导 干部 的 素质 和 能力 。
```

Comment.

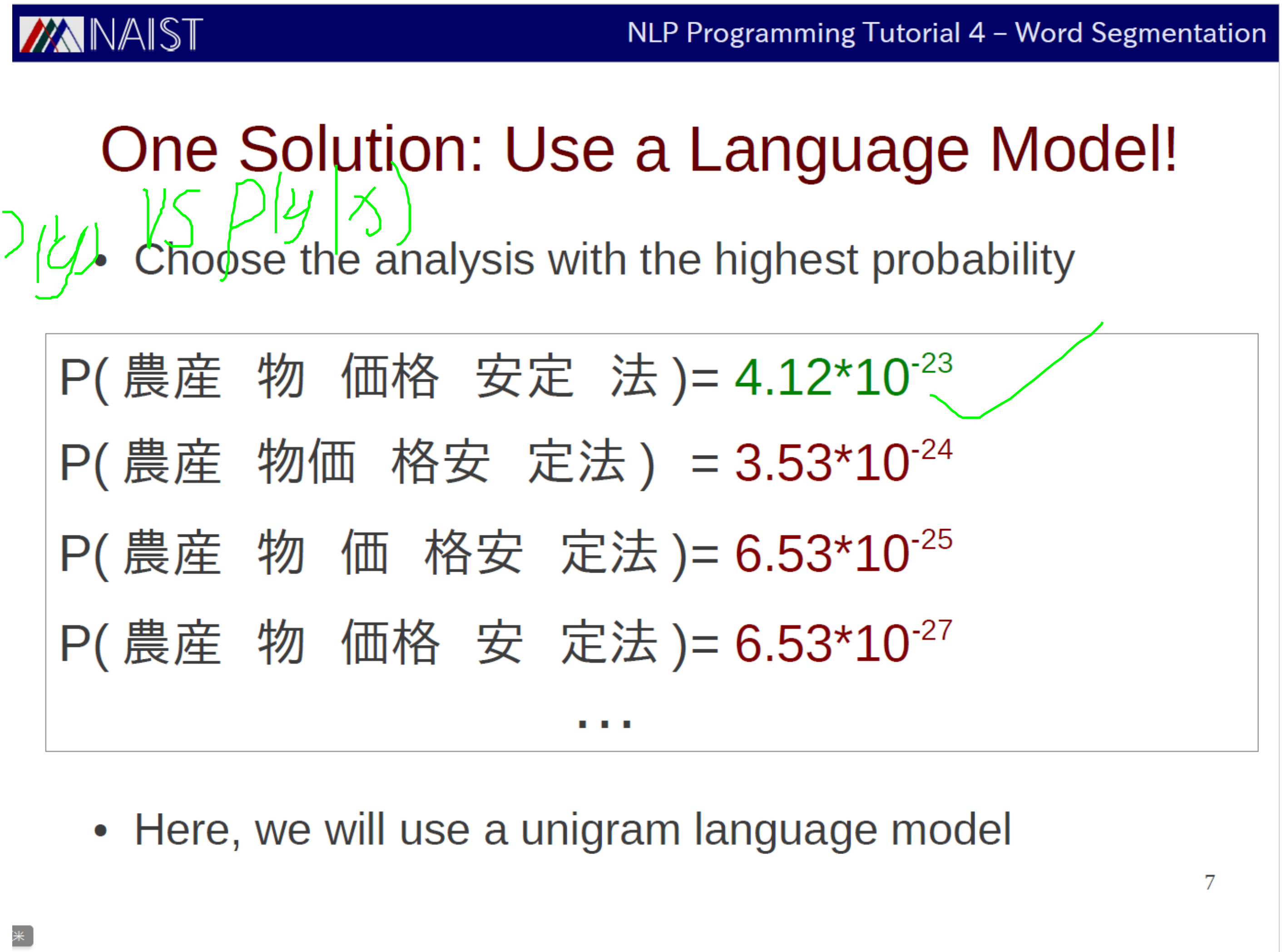
Here I want make us know that we are going to use the Chinese word segmentation dataset which I cleaned from 人民日报 corpus. In my github repo (https://github.com/Epsilon-Lee/nlptutorial) of our NLP tutorial, I have added file `rmrb=199812.raw` in the `/data` folder, and have preprocessed the raw data into train and test files: `rmrb-train.tok` and `rmrb-test.tok`. In the experiment, I would like to use the Chinese corpus instead of the original Japanese one.

2.1 Why a language model as a scoring rule?

1 Let us see how a language model can be used as a scoring rule for WS problem. In the previous discussion, we know that we need to learn a model $P_\theta(y|x)$, so that given the input unsegmented sentence, we can have a scoring rule - the probability - of possible segmentations y s over x.

However, since a language model is not a conditional distribution of the form $P(y|x)$ as we have discussed above, we cannot directly use a language model. But if we change the form of conditional to be: $P(y|x) = \frac{P(y)P(x|y)}{P(x)}$, we can find that since we constrain y within the space of inserting separators into x, so for the y, the probability of getting the x is actually 1, that is $P(x|y) = 1$ in the above equation. So we can get a new scoring rule only determined by $P(y)$ (Note that here we ignore the denominator because it is the same with respect to the same x).

2 Another view of using a language model as a scoring rule is very intuitive, which is demonstrated in Neubig's slide below:



NAIST NLP Programming Tutorial 4 – Word Segmentation

One Solution: Use a Language Model!

P(y) VS P(y|x)

- Choose the analysis with the highest probability

P(農産 物 価格 安定 法)= $4.12*10^{-23}$

P(農産 物価 格安 定法) = $3.53*10^{-24}$

P(農産 物 価 格安 定法)= $6.53*10^{-25}$

P(農産 物 価格 安 定法)= $6.53*10^{-27}$

…

- Here, we will use a unigram language model

7

Comment.

Here I would like to emphasize a mistake I made when I am trying to understand why language model here could help. The mistake is that I am confused with the **minimum granularity** of certain natural language the LM wants to model. For example, for English, the smallest language phenomenon we are modeling is the words/tokens of English, i.e. cat, dog, ‘s, ‘ll, etc. However, for Chinese, the smallest language phenomenon might be characters like “疼”, “晓” etc. Or if we focus on dealing with “词组” in Chinese, the smallest phenomenon we are modeling should be phrase like “饺子”, “海鲜” etc. Here, we are going to use **phrase-level language model** which is trained over segmented Chinese sentences for us to score different segmentations of a given sentence, so that the LM can recognize good segmentations from bad ones thus acts as a scoring rule.

2.2 How to model the decoding problem using LM score?

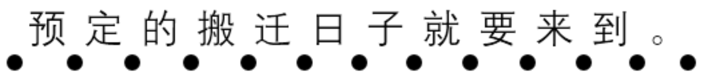
So after using the method learned from Tutorial 1, we can train our phrase-level language model over the `rmrb-train.tok` file. And now, we get the trained unigram language model $P(\cdot)$ where the $\cdot$ can be any phrase in Chinese. If the phrase exists in the training corpus, we can get its probability estimation, or otherwise, we will get a smoothed estimation for this unknown word.

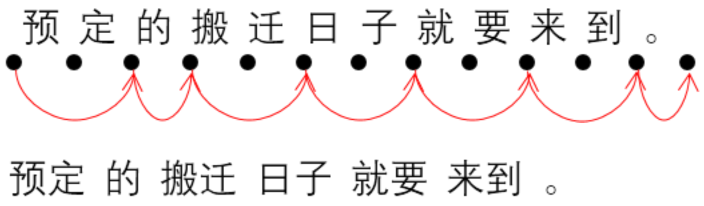
Now, let us embrace the decoding stage - to deal with the decoding problem (actually, it is a search problem). The **first stupid method** we can use to solve the decoding problem is to use **exhaustive search**. That is, given an unsegmented sentence x, we can enumerate all possible segmentations ys and use the scoring rule to judge $P(y)$, and choose y^* with the biggest $P(y)$. This is a very stupid method, how can we improve it?

Comment.

Actually, this is a search algorithm design problem. Since I am poor in algorithm design and analysis, the following story maybe weird for you to read with intuition. Forgive me about that, and I promise I am doing the best of myself.

We can formulate the problem of finding the best segmentation as finding a path that connects those black nodes in the following graph.




预 定 的 搬 迁 日 子 就 要 来 到 。
预定 的 搬迁 日子 就要 来到 。

The above 3 figures represent:

- Nodes separate each character in the sentence.
- An optimal path which achieves the golden segmentation. (*Golden segmentation means the best segmentation or the reference segmentation.*)
- The reference segmentation.

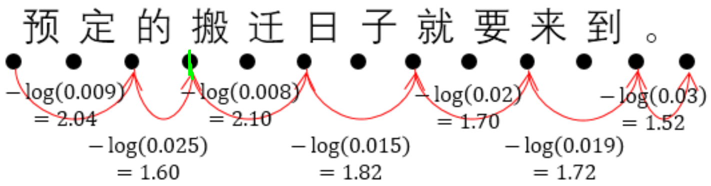
In terms of the path, I would claim that:

- Each path represents a certain segmentation of the original sentence. (**This is obvious!**)
- **All** possible paths equal to **all** possible segmentations of the sentence. (**This is not that obvious, but intuitively you can accept this, right?**)

There are constraints for the path. That is:

- Each edge of that path should not cross with others;
- those edges that consist a path are adjacent.

After we abstract the segmentation problem as a path finding problem, we should now find the usage of the trained language model $P(\cdot)$. It is very obvious to say that the language model can give weights between any two nodes, that is the edge with weight, like below:



The sum of the path is the negative log likelihood of the sentence "预订 的 搬迁 日 就要 来到 。". That is:

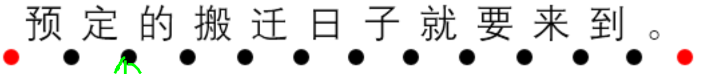
$$\begin{aligned}
&-\log P(\text{预订, 的, 搬迁, 日子, 就要, 来到, 。})\\
&= -\log P(\text{预订}) - \log P(\text{的}) - \log P(\text{搬迁}) - \log P(\text{日子}) - \log P(\text{就要}) - \log P(\text{来到}) - \log P(\text{。})\\
&= 2.04 + 1.60 + 2.10 + 1.82 + 1.70 + 1.72 + 1.52\\
&= 10.78
\end{aligned}$$

According to the above example, we **define** the weight of each edge equals to the *negative log likelihood of the phrase covered by the edge*. So to find the best segmentation, we are supposed to find the smallest sum path.

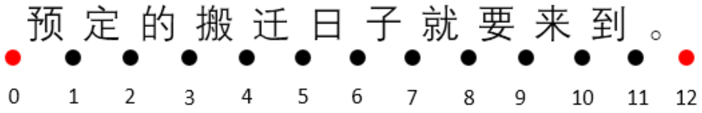
2.3 Solve the path finding problem by Viterbi algorithm

The nature of Viterbi algorithm is to take advantage of the optimal substructure of the problem, and use **Dynamic Programming** to efficiently compute optimal sub-solutions; and find the best solution by backward tracking.

Given the sentence to be segmented, we can draw the nodes to separate each character. Moreover, we can find that the path should start and end with the red nodes in the following figure, that is the beginning and ending of the node sequence.



We index the nodes as following:



Suppose that for each node with index i, we **1).** save the smallest possible sum s_i to that node and **2).** record the edge with the node index n_i that brings with this sum (which is the edge (n_i, i)). Since there are $i-1$ possible edges that could be linked from previous $i-1$ nodes, so we can have a recursive formula to compute s_i:

$$s_i = min_{1 \le j \le i-1}[s_j + P(j, i)], n_i = argmin_{1 \le j \le i-1} s_j$$

The above formula assumes that when computing s_i for node i, previous nodes have saved the optimal sub-solution. Here $P(j, i)$ means the substring composed by characters from j to $i - 1$.

After computing from **left-to-right**, we can get the optimal sum at each node and the best adjacent edge that can lead to the sum. So if we start backward tracking from node 12, we can find a path all the way down to node 0, and this is the smallest sum path of the all sentence. The `python` psudocode of the algorithm might be the following.

```
# const to denote +inf
INF = 10000
# 1. Train the language model, and get a dictionary:
  unigram_prob = {'string' : probability}
# 2. Given an unsegmented sentence encoded by unicode (follow the instruction on the 5th slide of Neubig)
  str_utf8 # you can access each char by indexing str_utf8[i] to get the i-th character
# 3. Initialize a list to hold sub-optimal sums s_i and another list to hold previous edge n_i
  length = len(str_utf8)
  s = [INF for i in range(length)]
  s[0] = 0.
  n = [0 for i in range(length)]
# 4. loop to compute s_i and n_i
  for end_node_idx in range(1, len):
    for start_node_idx in range(0, end_node_idx):
      # find the best previous adjacent edge
      unigram = str_utf8[start_node_idx:end_node_idx]
      tmp_sum = s[start_node_idx] + unigram_prob[unigram]
      # compare which is sum is smaller
      if tmp_sum < s[end_node_idx]:
        s[end_node_idx] = tmp_sum
        n[end_node_idx] = start_node_idx
# 5. backward tracking to find the path
  start_node_idx = end_node_idx = length - 1
  str_list = []
  while n[start_node_idx] != 0:
    start_node_idx = n[end_node_idx]
    word = str_utf8[start_node_idx : end_node_idx + 1]
    str_list.append( )
    end_node_idx = start_node_idx
  segmentation = list(reversed(str_list))
```

Appendix

The discussion paper has not been decided yet. Following are some papers which have some impact on the community. To be honest, for every task like word segmentation, many methods have been tried out, Bayesian, log-linear, neural networks etc. So the **key point** is to grasp each of those techniques behind specific tasks, so that you can have more dimensions when you are solving a new task in your job or research career.

Reference

- Chinese Word Segmentation without Using Lexicon and Hand-crafted Training Data (http://aclweb.org/anthology/C98-2201), Maosong Sun et al. 1998.
- Chinese Word Segmentation and Named Entity Recognition: A Pragmatic Approach (http://www.mitpressjournals.org/doi/pdf/10.1162/089120105775299177), 2006. Jianfeng Gao, et al. Computational Linguistics.
- Optimizing Chinese Word Segmentation for Machine Translation Performance (https://nlp.stanford.edu/pubs/acl-wmt08-cws.pdf), WMT 2008. Manning's group.
- Max-Margin Tensor Neural Network for Chinese Word Segmentation (http://www.aclweb.org/anthology/P14-1028), ACL 2014. Baobao Chang's group at Peking Univ.

0 Comments **epsilon-lee** 1 **Login**

♡ **Recommend** **Share** **Sort by Best**

Start the discussion…

LOG IN WITH

OR SIGN UP WITH DISQUS ?

Name

Be the first to comment.

✉ **Subscribe** **Add Disqus to your siteAdd DisqusAdd** **Privacy**