阶段性学习总结——中文分词及数据挖掘

张佳敏

目录

一．中文分词

- 李纪为博士的论文在讲到分词对于中文深度学习是否具有必要性时，提到中文分词相对于英文直接利用空格分词而言是比较麻烦的，其中两种中文分词的模型，word-based models 由于词分布的稀疏性会过度拟合导致大量OOV的产生，分词后语料库太大，使其分词效果并不如char-based models。
- 作者在论文中提到了很多用来得出此结论而做的实验：
    1. 用Jieba 对CTB进行分词
    2. 语言建模
    3. 机器翻译
    4. 文本分类
    5. 设置Frequency Bar来探究词语稀疏性对于两种模型的影响
    6. OOV对模型效果的影响。
- 下面对这6个实验进行理解和结果分析。

一．中文分词

1. 用Jieba 对CTB进行分词

bar	# distinct	prop of vocab	prop of corpus
∞	50,266	100%	100%
4	38,889	77.4%	10.1%
1	24,458	48.7%	4.0%

作者的实验结果显示，使用Jieba分词对CTB进行切词，共得到50266个不同词汇，其中仅出现一次的词语有24458个，占词语比例的48.7%，占总语料库的4.0%，由此可见，很多词出现的频率是很低的，但是这些词却占据了总词数很大的比例，在语料库的占比又特别小。数据稀疏易导致过拟合问题，很多词语会被处理为OOV，进而会影响模型训练效果。

一．中文分词

2.语言建模

在语言建模这一任务中，是通过给定一定的信息预测后续的词语。作者使用的是CTB6 来对比两个模型的效果的。作者对比了不同纬度下word、char和混合模型的模型效果，结果显示，维度在512时差距并不明显，但在2048的时候，ppl达到最优的结果差距明显。与此同时，作者在 CWS 包和 LTP 包下也进行了试验，结果相同。

作者还对混合模型的效果进行了比较，对比混合模型与char only，结果发现在嵌入词向量之后，效果反而不如char only 模型。

model	dimension	ppl
word	512	199.9
char	512	193.0
word	2048	182.1
char	2048	170.9
hybrid (word+char)	1024+1024	175.7
hybrid (word+char)	2048+1024	177.1
hybrid (word+char)	2048+2048	176.2
hybrid (char only)	2048	171.6

一．中文分词

3. 机器翻译

作者对中译英和英译中都进行了实验来对比不同模型下的机器翻译效果。

表四的中译英和表五的英译中结果显示，无论是中译英还是英译中，char模型都比word模型的效果好。

一. 中文分词

3. 机器翻译

TestSet	Mixed RNN	Bi-Tree-LSTM	PKI	Seq2Seq +Attn (word)	Seq2Seq +Attn (char)	Seq2Seq (word) +Attn+BOW	Seq2Seq (char) +Attn+BOW
MT-02	36.57	36.10	39.77	35.67	36.82 (+1.15)	37.70	40.14 (+0.37)
MT-03	34.90	35.64	33.64	35.30	36.27 (+0.97)	38.91	40.29 (+1.38)
MT-04	38.60	36.63	36.48	37.23	37.93 (+0.70)	40.02	40.45 (+0.43)
MT-05	35.50	34.35	33.08	33.54	34.69 (+1.15)	36.82	36.96 (+0.14)
MT-06	35.60	30.57	32.90	35.04	35.22 (+0.18)	35.93	36.79 (+0.86)
MT-08	–	–	24.63	26.89	27.27 (+0.38)	27.61	28.23 (+0.62)
Average	–	–	32.51	33.94	34.77 (+0.83)	36.51	37.14 (+0.63)

Table 4: Results of different models on the Ch-En machine translation task. Results of Mixed RNN (Li et al., 2017), Bi-Tree-LSTM (Chen et al., 2017a) and PKI (Zhang et al., 2018) are copied from the original papers.

TestSet	Seq2Seq +Attn (word)	Seq2Seq +Attn (char)	Seq2Seq +Attn+BOW	Seq2Seq (char) +Attn+BOW
MT-02	42.57	44.09 (+1.52)	43.42	46.78 (+3.36)
MT-03	40.88	44.57 (+3.69)	43.92	47.44 (+3.52)
MT-04	40.98	44.73 (+3.75)	43.35	47.29 (+3.94)
MT-05	40.87	42.50 (+1.63)	42.63	44.73 (+2.10)
MT-06	39.33	42.88 (+3.55)	43.31	46.66 (+3.35)
MT-08	33.52	35.36 (+1.84)	35.65	38.12 (+2.47)
Average	39.69	42.36 (+2.67)	42.04	45.17 (+3.13)

Table 5: Results on the En-Ch machine translation task.

ers are the other way around. For LCQMC (Liu sentence. We conclude that the char-based model

一．中文分词

4. 文本分类

作者利用不同的benchmarks 来对基于char和基于word的模型进行实验。

Table 6: Results on the LCQMC and BQ corpus.

Dataset	description	char valid	word valid	char test	word test
chinanews	1260K/140K/112K	91.81	91.82	91.80	**91.85** (+0.05)
dianping	1800K/200K/500K	78.80	78.47	**78.76** (+0.36)	78.40
ifeng	720K/80K/50K	86.04	84.89	**85.95** (+1.09)	84.86
jd_binary	3600K/400K/360K	92.07	91.82	**92.05** (+0.16)	91.89
jd_full	2700K/300K/250K	54.29	53.60	**54.18** (+0.81)	53.37

Table 7: Results on the validation and the test set for text classification.

作者使用双向 LSTM 模型对基于word 和基于char 的模型分别进行训练用于评测，表七结果显示除了China news 以外，基于char的模型得出的结果都要优于基于word的模型。

一．中文分词

4. 文本分类

表八展示了模型基于对已有数据分布（源领域）的训练，学习新数据分布（目标领域）的能力。作者基于不同的情感分析数据库对两种模型进行了评测，结果发现，基于char的模型在领域适应能力更强且具有更好的表现。

train_dianping_test_jd		
model	acc	proportion of sen containing OOV
word-based	81.28%	11.79%
char-based	83.33%	0.56%

train_jd_test_dianping		
model	acc	proportion of sen containing OOV
word-based	67.32%	7.10%
char-based	67.93%	46.85%

一．中文分词

5. 设置Frequency Bar来探究词语稀疏性对于两种模型的影响

在Analysis这一部分当中，作者通过实验利用设置Frequency Bar来探究词语稀疏性对于两种模型的影响。结果显示，两种模型表现得最好的时候，词规模是差不多的，但是Frequency Bar却相差不小，对于word的模型来说，低频词的学习难度是比较高的，相应的准确度也远不如基于model的模型。

一. 中文分词

5. 设置Frequency Bar来探究词语稀疏性对于两种模型的影响

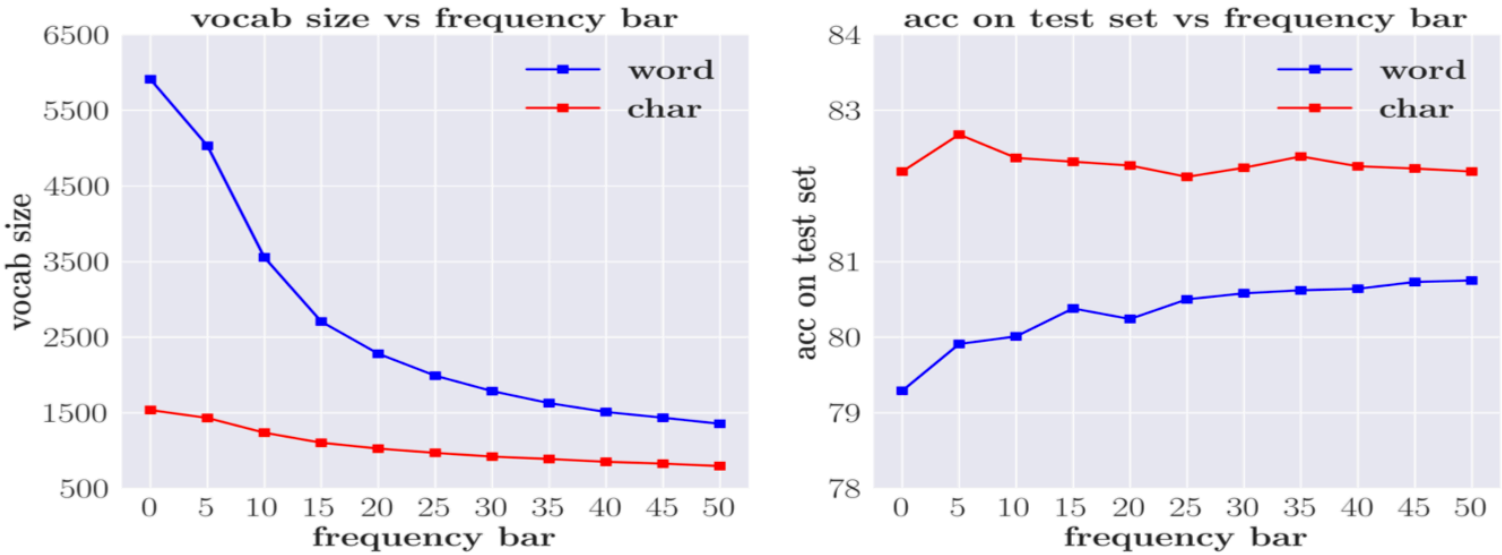

Figure 2: Effects of data sparsity on the char-based model and the word-based model.

一．中文分词

6.OOV对模型效果的影响。

作者在论文中阐述影响word model的另一大因素是OOV，但降低Frequency bar虽然会减少OOV,但是会出现数据稀疏问题。作者在实验中基于不同的Frequency bar 分别移除对应包含OOV的句子，结果显示，随着frequency bar的增加，两种模型的效果差距在减小，不过char-based model始终优于word-based model.

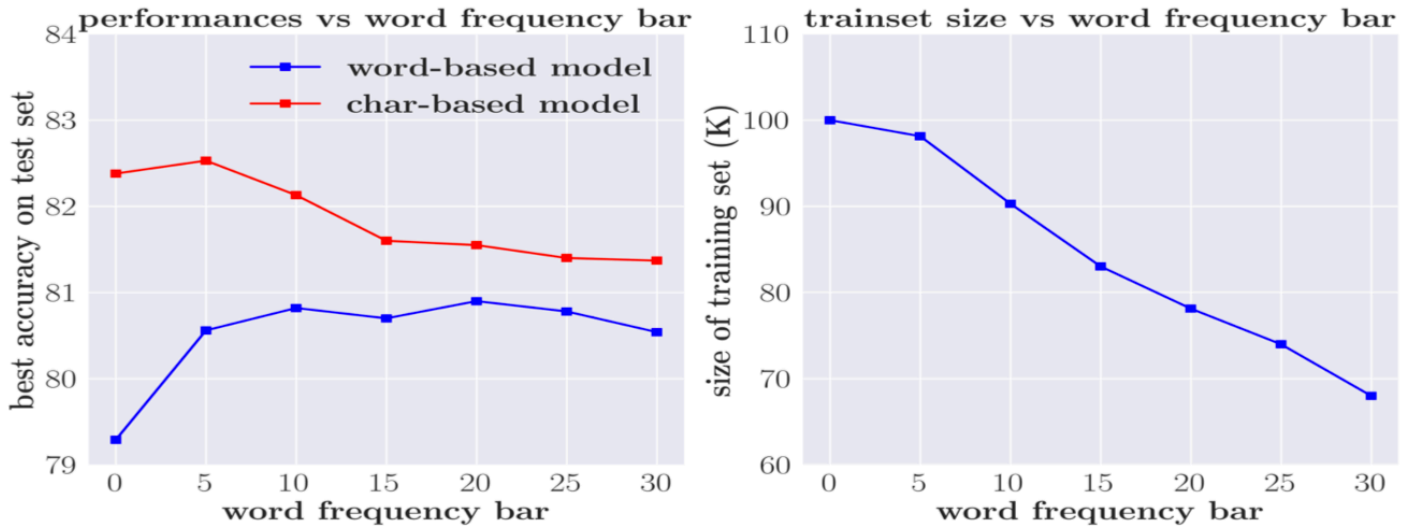

Figure 4: Effects of removing training instances containing OOV words.

一．中文分词

李纪为博士在这项研究探究了中文分词必要性的问题，并从四类NLP任务的实验中得出char-based model是优于word-based model的这个结论。

作者将word模型效果不佳的原因归结于词分布的稀疏性导致更多OOV的产生、过拟合以及领域转化能力差。

二．数据挖掘

1.介绍

在Introduction 这一章节里，作者依次对万维网和网络和互联网的历史做了简要的介绍。在介绍了网络信息的几个特点之后，提出网络数据挖掘和传统的数据挖掘的不同，之后介绍了数据挖掘的一些任务以及步骤：预处理、数据挖掘、后处理。紧接着将网络数据挖掘分成三类：Web结构挖掘，Web内容挖掘和Web使用挖掘。

整篇论文分为两大部分，第一部分主要涉及的是数据挖掘的主要内容，分别是第二章的关联规则和序列模式，第三章的监督学习，第四章的无监督学习以及第五章的半监督学习。

二. 数据挖掘

2.关联规则和序列模式

2.1 关联规则的基本概念

在这一小节中，介绍了关联规则的概念并阐释了支持度和信任度的定义。文中用超市购物篮为例子：

t_1: Beef, Chicken, Milk
t_2: Beef, Cheese
t_3: Cheese, Boots
t_4: Beef, Chicken, Cheese
t_5: Beef, Chicken, Clothes, Cheese, Milk
t_6: Chicken, Clothes, Milk
t_7: Chicken, Milk, Clothes

买完chicken和clothes 再买milk的交易有三个，总交易有七个，故同时含有三者的概率为3/7，即为支持度，再已买chicken和clothes 的情况下再买milk的概率为3/3，即为信任度。
两者的定义分别为

$$support = \frac{(X \cup Y).count}{n}. \qquad confidence = \frac{(X \cup Y).count}{X.count}.$$

(count为总事件数量)

文中还提到了支持度和信任度用来约束关联关系。

二. 数据挖掘

2.关联规则和序列模式

2.1 先验算法（Apriori Algorithm）

先验算法的两个步骤：1、生成所有频繁项目集 2、从频繁项目集生成所有确定的关联规则

先验算法时依赖于先验或向下闭包属性来有效地生成所有频繁项集的。

文中介绍了先验算法时怎么产生频繁项目集：

t_1: Beef, Chicken, Milk
t_2: Beef, Cheese
t_3: Cheese, Boots
t_4: Beef, Chicken, Cheese
t_5: Beef, Chicken, Clothes, Cheese, Milk
t_6: Chicken, Clothes, Milk
t_7: Chicken, Milk, Clothes

仍以超市购物篮为例，为产生频繁项集，先分别得出minsup>30%的项集：
{{Beef}:4, {Cheese}:4, {Chicken}:5, {Clothes}:3, {Milk}:4}
由此产生二维项集{{Beef, Cheese}, {Beef, Chicken}, {Beef, Clothes}, {Beef, Milk}, {Cheese, Chicken}, {Cheese, Clothes}, {Cheese, Milk}, {Chicken, Clothes}, {Chicken, Milk}, {Clothes, Milk}}
发现其中大于最小支持度的只有{{Beef, Chicken}:3, {Beef, Cheese}:3, {Chicken, Clothes}:3, {Chicken, Milk}:4, {Clothes, Milk}:3} 组成2维最大项集

二. 数据挖掘

2.关联规则和序列模式

2.2先验算法（Apriori Algorithm）

t_1: Beef, Chicken, Milk
t_2: Beef, Cheese
t_3: Cheese, Boots
t_4: Beef, Chicken, Cheese
t_5: Beef, Chicken, Clothes, Cheese, Milk
t_6: Chicken, Clothes, Milk
t_7: Chicken, Milk, Clothes

再由此产生三维候选项集：{{Chicken, Clothes, Milk}，{Beef, Cheese, Chicken}}
由于{Beef, Cheese, Chicken}中的{Cheese, Chicken}并不在二维最大项集中，故最大项集只有{{Chicken, Clothes, Milk}}

二. 数据挖掘

2.关联规则和序列模式

2.2 先验算法（Apriori Algorithm）

文中也具体提及了先验算法的内容：-»

先验算法采用自底向上的处理方法，可以生成最后的频繁项目集，在这个例子中，可决定出最频繁购买的物品集，除开先验算法只需要扫描k次（k为最大项目集的个数），先验算法也是具有缺点的（产生大量项目，很难分析其中有用的信息。）

```
Algorithm Apriori(T)
1   C1 ← init-pass(T);                          // the first pass over T
2   F1 ← {f | f ∈ C1, f.count/n ≥ minsup};      // n is the no. of transactions in T
3   for (k = 2; Fk−1 ≠ ∅; k++) do               // subsequent passes over T
4     Ck ← candidate-gen(Fk−1);
5     for each transaction t ∈ T do             // scan the data once
6       for each candidate c ∈ Ck do
7         if c is contained in t then
8           c.count++;
9       endfor
10    endfor
11    Fk ← {c ∈ Ck | c.count/n ≥ minsup}
12  endfor
13  return F ← ⋃k Fk;
```

Fig. 2.2. The Apriori algorithm for generating frequent itemsets

```
Function candidate-gen(Fk−1)
1   Ck ← ∅;                                     // initialize the set of candidates
2   forall f1, f2 ∈ Fk−1                        // find all pairs of frequent itemsets
3     with f1 = {i1, … , ik−2, ik−1}            // that differ only in the last item
4     and f2 = {i1, … , ik−2, i'k−1}
5     and ik−1 < i'k−1 do                       // according to the lexicographic order
6       c ← {i1, …, ik−1, i'k−1};               // join the two itemsets f1 and f2
7       Ck ← Ck ∪ {c};                          // add the new itemset c to the candidates
8       for each (k−1)-subset s of c do
9         if (s ∉ Fk−1) then
10          delete c from Ck;                   // delete c from the candidates
11      endfor
12  endfor
13  return Ck;                                  // return the generated candidates
```

Fig. 2.3. The candidate-gen function

二. 数据挖掘

2.关联规则和序列模式

2.2 关联规则的基本概念

2.2.2 生成关联规则

算法 -»

文中还是举用了之前的例子：
当已产生最大项集时：{{Chicken, Clothes, Milk}:3}
对应产生的关联规则为：

Rule 1: Chicken, Clothes → Milk [sup = 3/7, conf = 3/3]
Rule 2: Chicken, Milk → Clothes [sup = 3/7, conf = 3/4]
Rule 3: Clothes, Milk → Chicken [sup = 3/7, conf = 3/3].

由于最小相信度的限制，只有1和3 符合，再根据算法产生Rule 4。

Algorithm genRules(F) // F is the set of all frequent itemsets
1 **for** each frequent k-itemset f_k in F, $k \geq 2$ **do**
2 output every 1-item consequent rule of f_k with confidence ≥ *minconf* and
support ← $f_k.count / n$ // n is the total number of transactions in T
3 $H_1 \leftarrow$ {consequents of all 1-item consequent rules derived from f_k above};
4 ap-genRules(f_k, H_1);
5 **endfor**

Procedure ap-genRules(f_k, H_m) // H_m is the set of m-item consequents
1 **if** ($k > m + 1$) AND ($H_m \neq \emptyset$) **then**
2 $H_{m+1} \leftarrow$ candidate-gen(H_m);
3 **for** each h_{m+1} in H_{m+1} **do**
4 $conf \leftarrow f_k.count / (f_k - h_{m+1}).count$;
5 **if** ($conf \geq minconf$) **then**
6 output the rule $(f_k - h_{m+1}) \rightarrow h_{m+1}$ with confidence = *conf* and
support = $f_k.count / n$; // n is the total number of transactions in T
7 **else**
8 delete h_{m+1} from H_{m+1};
endfor
ap-genRules(f_k, H_{m+1});
endif

Fig. 2.4. The association rule generation algorithm

二．数据挖掘

2.关联规则和序列模式

2.3 关联规则挖掘的数据格式&2.4 多最小支持度关联规则挖掘

2.3中讲到的数据格式提到表格数据集转换为事务数据集，也可使用二进制表示将事务数据集转换为表数据集。

2.4提到项目的频率不同会带来两个不同的问题，一个是minsup设置的太高会找不到频率低的项目规则，另一个是如果想要得到频率低的项目规则，那么minsup会被设置的很低而导致组合情况特别多以至于数据挖掘变得不可能。作者仍然用超市购物作为例子具体讲了问题，作者对此提出了两个解决方法，一个是将数据划分为几个较小的块（子集），因为无法找不到涉及跨不同块的项目的项目集或规则而别舍弃；另一个更好的方法是允许用户指定多个最小支持来进行数据挖掘。

二. 数据挖掘

2.关联规则和序列模式

2.4 多最小支持度关联规则挖掘

在Example 9中，作者分别设置最小支持度

MIS(1) = 10% MIS(2) = 20% MIS(3) = 5% MIS(4) = 6%

发现{1,2}在级别2上具有9%的支持，则它不满足MIS（1）或MIS（2），根据算法会被丢弃，从而不会为级别3生成潜在频繁项集{1,2,3}和{1,2,4}，但是项集{1,2,3}和{1,2,4}可能是频繁的，故丢弃{1,2}是错误的。

故又提出一种算法：按照其MIS值按升序对项目进行排序，以避免出现问题。

三．总结

最近主要在看李纪为博士的论文，其中穿插学习理解了web data mining 的前两章的理论知识，李纪为博士的论文已经通读过几遍了，后期会尽量去复现其中提到的实验，并继续数据挖掘的论文阅读与学习理解。

文献参考

- Agarwal, D. Statistical Challenges in Online Advertising. In Tutorial given at ACM KDD-2009 conference, 2009.
- Brusilovsky, P., A. Kobsa, and W. Nejdl. Adaptive Web: Methods and Strategies of Web Personalization. 2007, Berlin: Springer.
- Castillo, C., D. Donato, A. Gionis, V. Murdock, and F. Silvestri. Know your neighbors: Web spam detection using the web topology. In Proceedings of ACM SIGIR Conf. on Research and Development in Information Retrieval (SIGIR-2007), 2007: ACM.