

1. Text representation

1.1 word - one hot

向量长度=词典大小

词典: [我们, 去, 爬山, 今天, 你们, 昨天, 跑步]
0 ③ ② ① ④

每个单词的表示:

- 我们: $(1, 0, 0, 0, 0, 0, 0) \rightarrow 7维 = 1词典 |$
- 爬山: $(0, 0, 1, 0, 0, 0, 0) \rightarrow 7维 = 1词典 |$
- 跑步: $(0, 0, 0, 0, 0, 0, 1) \rightarrow 7维 = 1词典 |$
- 昨天: $(0, 0, 0, 0, 0, 1, 0) \rightarrow 7维 = ..$

1.2 sentence

1.2.1 boolean

词典: [我们, 又, 去, 爬山, 今天, 你们, 昨天, 跑步] 8个单词

每个句子的表示

- { 我们^V 今天^V 去^V 爬山: $(1, 0, 1, 1, 1, 0, 0, 0) \rightarrow 8维 = 1词典 |$
- { 你们^V 昨天^V 跑步: $(0, 0, 0, 0, 0, 1, 1, 1) \rightarrow 8维 = 1词典 |$
- { 你们^V 又^V 去^V 爬山^V 又^V 去^V 跑步: $(0, 1, 1, 1, 0, 1, 0, 1) \rightarrow 8维 = 1词典 |$

不管单词出现在句子里，出现多少次，都为 1.

1.2.2 count

词典: [我们, 又, 去, 爬山, 今天, 你们, 昨天, 跑步]

representation

每个句子的表示

我们 今天 去 爬山: $(1, 0, 1, 1, 1, 0, 0, 0) \Rightarrow 8维 = 1词典 |$

你们 昨天 跑步: $(0, 0, 0, 0, 0, 1, 1, 1) \Rightarrow 8维 = 1词典 |$

你们 又 去 爬山 又 去 跑步: $(0, 2, 2, 1, 0, 1, 0, 1) \Rightarrow 8维 = 1词典 |$

是需要考虑单词在句子中出现的次数。

2. sentence similarity

2.1 计算距离，欧式距离

计算距离（欧式距离）： $d = |s1 - s2|$

S1: “我们今天去爬山” = (1,0,1,1,0,0,0,0)

S2: “你们昨天跑步” = (0,0,0,0,0,1,1,1)

S3: “你们又去爬山又去跑步” = (0,2,2,1,0,1,0,1)

D 越大，相似度越小。

$$d(s_1, s_2) = \sqrt{1^2 + 1^2 + 1^2 + 1^2 + 1^2 + 1^2} = \sqrt{6}$$

$$d(s_1, s_3) = \sqrt{1^2 + 2^2 + 1^2 + 1^2 + 1^2} = \sqrt{8}$$

$$d(s_2, s_3) = \sqrt{2^2 + 2^2 + 1^2 + 1^2} = \sqrt{10}$$

$$\text{sim}(s_1, s_2) > \text{sim}(s_2, s_3); \quad \text{sim}(s_1, s_3) > \text{sim}(s_2, s_3)$$

缺点：不考虑向量的方向。只考虑大小。

2.2 余弦相似度

既考虑方向，也考虑大小。

计算相似度（余弦相似度）： $d = s1 \cdot s2 / (|s1| * |s2|)$

S1: “我们今天去爬山” = (1,0,1,1,0,0,0,0)

S2: “你们昨天跑步” = (0,0,0,0,0,1,1,1)

S3: “你们又去爬山又去跑步” = (0,2,2,1,0,1,0,1)

$$d(s_1, s_2) = 0/4 = 0$$

$$d(s_1, s_3) = (2+1)/(\sqrt{3} \cdot \sqrt{11}) = \frac{3}{\sqrt{33}}$$

$$d(s_2, s_3) = 2/(\sqrt{3} \cdot \sqrt{11}) = \frac{2}{\sqrt{33}}$$

$$\text{sim}(s_1, s_3) > \text{sim}(s_2, s_3) > \text{sim}(s_1, s_2)$$

cosine similarity

分子：内积。

分母：可以等同于一个 normalization

2.3 example - count base representation

句子1: He is going from Beijing to Shanghai

句子2: He denied my request, but he actually lied.

句子3: Mike lost the phone, and phone was in the car

句子1: (0, 0, 1, 0, 0, 0, 1, 1, 1, 0, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0)

句子2: (1, 0, 0, 1, 0, 1, 0, 0, 2, 0, 0, 1, 0, 0, 1, 0, 1, 0, 0, 0, 0)

句子3: (0, 1, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 1, 1, 0, 2, 0, 0, 2, 0, 1)

denied

he

缺点：

Denied 重要性很高，但在句子中只出现了一次我们设置为 1。但是 he 重要性没有那么高，却出现了两次被设置为 2。所以，仅用次数并不能判断词的重要性。

→ 所以有了下面的 tf-idf

3. Tf-idf Representation

$$tfidf(w) = tf(d, w) * idf(w)$$

↓ 文档 d 中 w 的词频 ↓ $\log \frac{N}{N(w)}$

N: 语料库中的文档总数

$N(w)$: 词语 w 出现在多少个文档？

其中红色的那项：考虑了单词的重要性。

绿色的：与上文的 countbase 一样。

$$tfidf(w) = tf(d, w) * idf(w)$$

① 词典: [今天, 上, NLP, 课程, 的, 有, 意思, 数据, 也]
|词典| = 9

句 1 今天 上 NLP 课程
 句 2 今天 的 课程 有 意思
 句 3 数据 课程 也 有 意思

\Rightarrow 句 1
 $= (1 \cdot \log \frac{3}{2}, 1 \cdot \log \frac{3}{1}, 1 \cdot \log \frac{3}{1}, 1 \cdot \log \frac{3}{3}, 0, 0, 0, 0, 0)$
 $= (\log \frac{3}{2}, \log 3, \log 3, \log 1, 0, 0, 0, 0, 0) \Rightarrow 9 维 = |词典|$

\Rightarrow 句 2
 $= (1 \cdot \log \frac{3}{2}, 0, 0, 1 \cdot \log \frac{3}{3}, 1 \cdot \log \frac{3}{1}, 1 \cdot \log \frac{3}{2}, 1 \cdot \log \frac{3}{2}, 0, 0)$
 $= (\log \frac{3}{2}, 0, 0, \log 1, \log 3, \log \frac{3}{2}, \log \frac{3}{2}, 0, 0) \Rightarrow 9 维 = |词典|$

\Rightarrow 句 3
 $= (0, 0, 0, 1 \cdot \log \frac{3}{3}, 0, 1 \cdot \log \frac{3}{2}, 1 \cdot \log \frac{3}{2}, 1 \cdot \log \frac{3}{1}, 1 \cdot \log \frac{3}{1})$
 $= (0, 0, 0, \log 1, 0, \log \frac{3}{2}, \log \frac{3}{2}, \log 3, \log 3) \Rightarrow 9 维 = |词典|$

Reference: <https://www.cnblogs.com/pinard/p/6693230.html>

4. Summary

One-hot:

Boolean-based

Count-based

Tf-idf based

4.1 one-hot 问题

1) 单词之间的语义相似度

下面哪些单词之间语义相似度更高？

我们，爬山，运动，昨天

我们: $(0, 1, 0, 0, 0, 0)$
爬山: $(0, 0, 1, 0, 0, 0)$
运动: $(1, 0, 0, 0, 0, 0)$
昨天: $(0, 0, 0, 1, 0, 0)$

① Euclidean Distance

$$d(\text{我们}, \text{爬山}) = \sqrt{2}$$

$$d(\text{我们}, \text{运动}) = \sqrt{2}$$

$$d(\text{运动}, \text{爬山}) = \sqrt{2}$$

$$d(\text{昨天}, \text{爬山}) = \sqrt{2}$$

② Cosine Similarity

$$\text{sim}(\text{我们}, \text{爬山}) = \frac{0}{\sqrt{2}} = 0$$

$$\text{sim}(\text{我们}, \text{运动}) = 0$$

$$\text{sim}(\text{运动}, \text{爬山}) = 0$$

$$\text{sim}(\text{昨天}, \text{爬山}) = 0$$

$$X \quad \text{① One-hot representation}$$

⊗ 有没有可能用半表示语义相似度？

One-hot 并不能表示单词之间的相似度。

2) sparsity

5. Distributed Representation

One-Hot Representation

我们: $[1, 0, 0, 0, 0, 0, 0]$
爬山: $[0, 0, 1, 0, 0, 0, 0]$
运动: $[0, 0, 0, 0, 0, 0, 1]$
昨天: $[0, 0, 0, 0, 0, 1, 0]$

Distributed Representation

我们: $[0.1, 0.2, 0.4, 0.2]$
爬山: $[0.2, 0.3, 0.7, 0.1]$
运动: $[0.2, 0.3, 0.6, 0.2]$
昨天: $[0.5, 0.9, 0.1, 0.3]$

5.1 similarity

Distributed Representation

我们: $[0.1, 0.2, 0.4, 0.2]$
爬山: $[0.2, 0.3, 0.7, 0.1]$
运动: $[0.2, 0.3, 0.6, 0.2]$
昨天: $[0.5, 0.9, 0.1, 0.3]$

四维向量

① Euclidean Distance

$$d(\text{我们}, \text{爬山}) = \sqrt{0.1^2 + 0.1^2 + 0.3^2 + 0.1^2} \\ = \sqrt{0.01 + 0.01 + 0.09 + 0.01} = \sqrt{0.12}$$

$$d(\text{运动}, \text{爬山}) = \sqrt{0.2^2 + 0.1^2} = \sqrt{0.05}$$

$$(d(\text{运动}, \text{爬山}) < d(\text{我们}, \text{爬山})) \\ \Rightarrow \text{sim}(\text{运动}, \text{爬山}) > \text{sim}(\text{我们}, \text{爬山})$$

$$\textcircled{2} \quad d(\text{爬山}, \text{运动}) < d(\text{昨天}, \text{运动})$$

$$\textcircled{3} \quad \text{sim}(\text{爬山}, \text{运动}) > \text{sim}(\text{昨天}, \text{运动})$$

分布式表示方法(单词)

词向量(word vectors)

词向量 word vector 是分布式表示方法的一种。

5.2 compare

Comparing the Capacities

容量空间

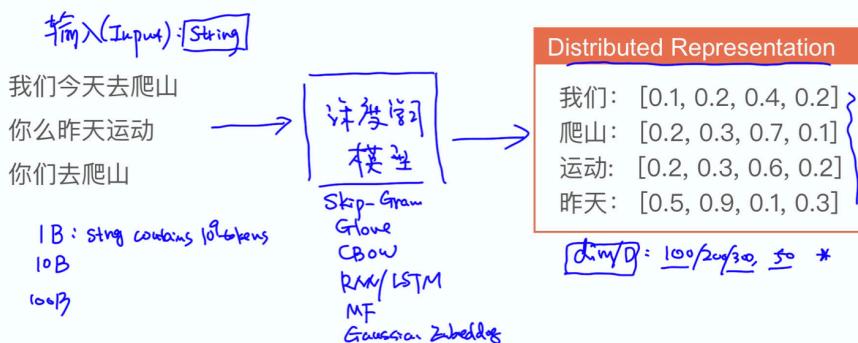
Q: 100 维的 One-Hot 表示法最多可以表达多少个不同的单词?

$$\text{向量大小} = 100 \quad \text{我们} = (1, 0, 0, \dots, 0) \quad 100 \text{ 个单词!}$$
$$\text{运动} = (0, 0, \dots, 1, 0, \dots, 0)$$

Q: 100 维的 分布式 表示法最多可以表达多少个不同的单词?

$$\text{我们: } (0.1, 0.2, 0.3, \dots, 0.1) \xrightarrow[100 \text{ 维}]{\text{Binary}} \text{ 二进制: } (1, 0, 1, 0, \dots, 0, 1) \quad 2^{100} \text{ 不同的单词!}$$
$$+ \infty$$
$$\downarrow \downarrow \downarrow \downarrow \quad 100 \text{ 维}$$
$$0/1 \quad 0/1$$

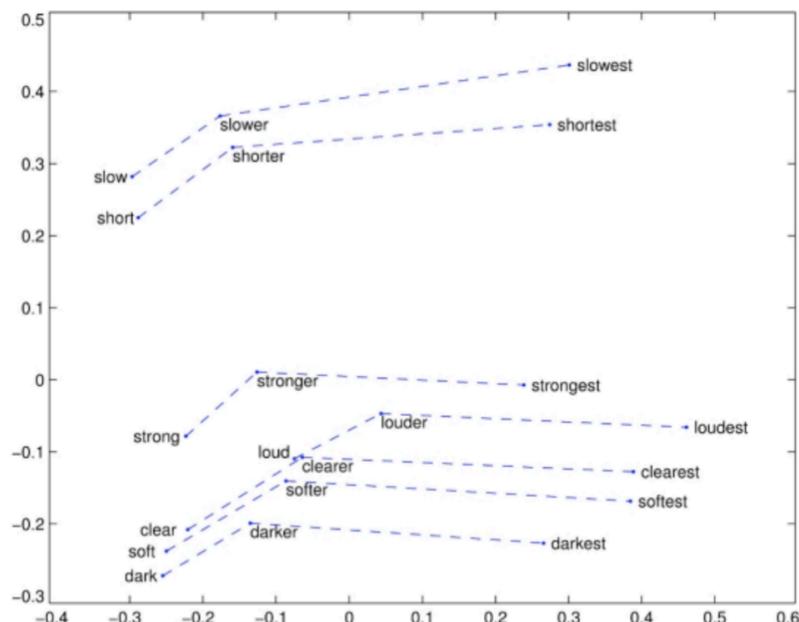
6. learn word embeddings



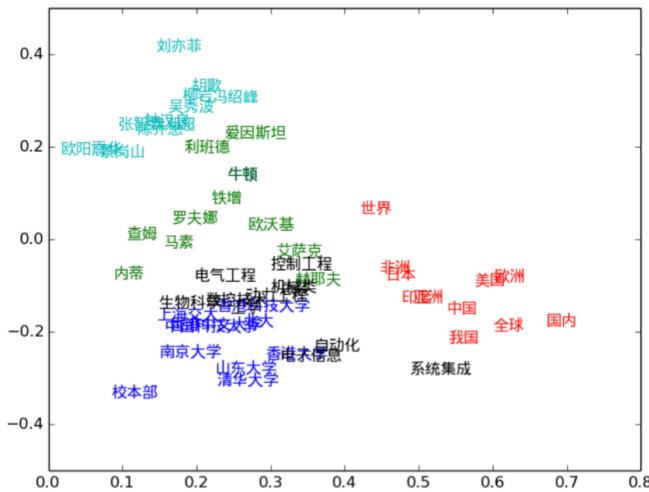
6.1 Essence of Word Embedding

词向量代表单词的意思。

Word embedding (word2vec) → meaning



类似的单词会聚在一起。



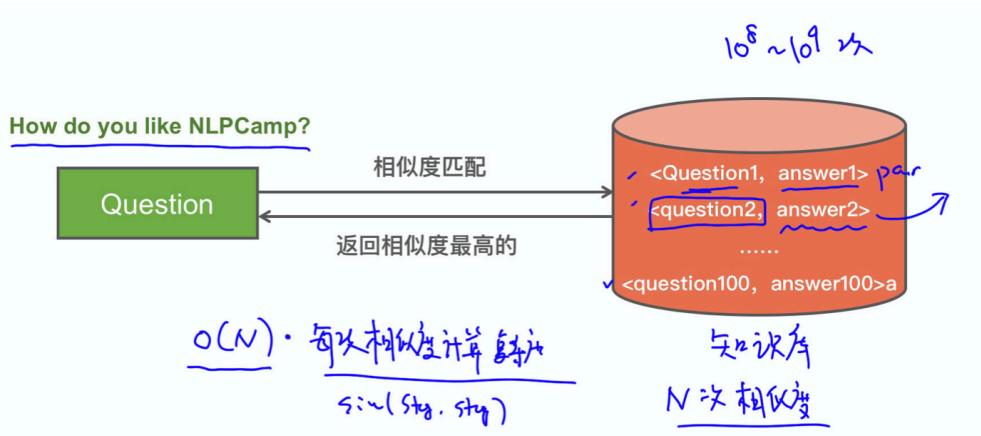
6.2 From Word Embedding to Sentence Embedding

$$\begin{array}{l}
 \text{我们: } (0.1, 0.2, 0.1, 0.3) \\
 \text{去: } (0.3, 0.2, 0.15, 0.2) \\
 + \text{ 还: } (0.2, 0.15, 0.4, 0.7) \\
 \hline
 & 0.6 & 0.55 & 0.65 & 1.2 \\
 \hline
 & (0.2 & 0.18 & 0.22 & 0.4)
 \end{array}$$

A: “我们 + 还”
 ① Averaging if 2)

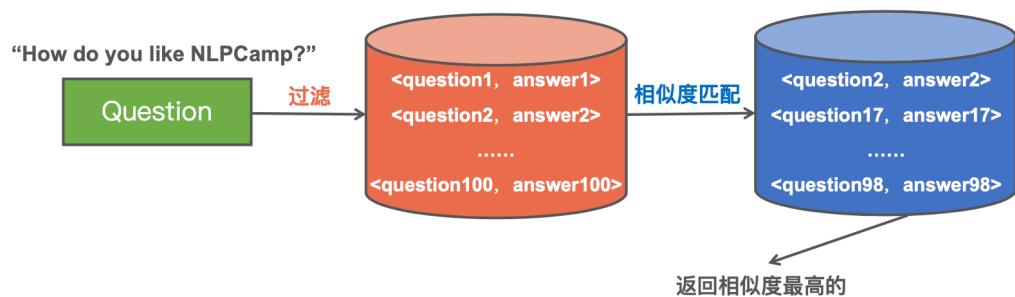
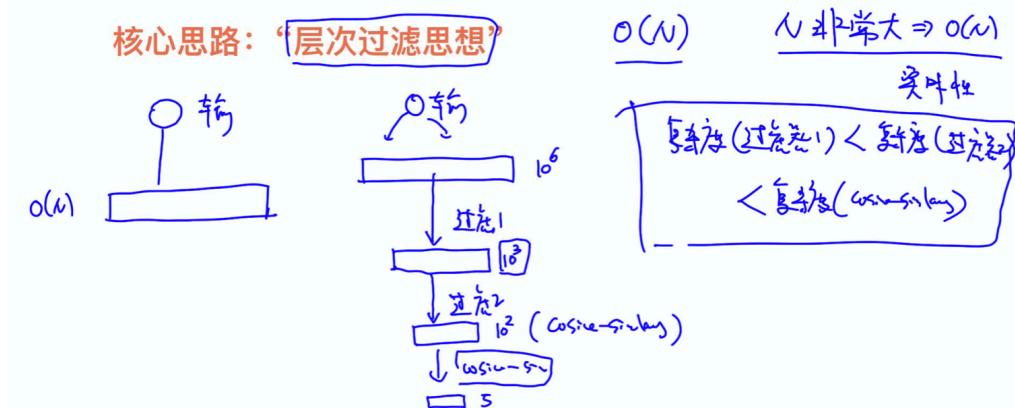
B: “我们 + 去” = $(0.2, 0.2, 0.25, 0.4)$ ✓
 ② LSTM/RNN

7. Recap: Retrieval-based QA System



复杂度太高

How to Reduce Time Complexity?



8. inverted index

Introducing Inverted Index

