文本表示1：TF-IDF、SVD（Fold in）

覃雄派

文本表示1：TF-IDF、SVD（Fold in）

提纲

- 为什么要Fold in
- Fold in实例

文本表示1：TF-IDF、SVD（Fold in）

- 为什么要Fold in
  - 假设我们有一个文集，已经进行SVD分解
    - 得到文档的降维表示
    - 也得到单词的降维表示
  - 现在有一个新的查询，可以把该查询看作一个新文档
    - 如何对这个查询进行降维?
    - 如何计算和这个查询相似的文档?
  - 办法有
    - （1）新查询加入文档集，重新进行SVD，重新对所有文档、单词进行降维
    - （2）Fold in这个新查询

文本表示1：TF-IDF、SVD（Fold in）

文本表示1：TF-IDF、SVD（Fold in）

- 假设有5个文档
  - d1 : Romeo and Juliet.
  - d2 : Juliet: O happy dagger!
  - d3 : Romeo died by dagger.
  - d4 : “Live free or die”, that’s the New-Hampshire’s motto.
  - d5 : Did you know, New-Hampshire is in New-England.

  - 有一个新的查询search query：dies, dagger

参考文献https://www.engr.uvic.ca/~seng474/svd.pdf
https://manuel.midoriparadise.com/public_html/svd-lsi-tutorial.pdf

文本表示1：TF-IDF、SVD（Fold in）

- 假设有5个文档
  - 构造词项-文档矩阵
  - 注意，已经去掉停用词
  - 红色为词汇表的词汇

d1 : Romeo and Juliet.
d2 : Juliet: O happy dagger!
d3 : Romeo died by dagger.
d4 : “Live free or die”, that’s the New-Hampshire’s motto.
d5 : Did you know, New-Hampshire is in New-England.

	d_1	d_2	d_3	d_4	d_5
romeo	1	0	1	0	0
juliet	1	1	0	0	0
happy	0	1	0	0	0
dagger	0	1	1	0	0
live	0	0	0	1	0
die	0	0	1	1	0
free	0	0	0	1	0
new-hampshire	0	0	0	1	1

文本表示1：TF-IDF、SVD（Fold in）

- 假设有5个文档
  - 进行奇异值分解
  - 请用jupyter打开如下python notebook进行实验

名称	修改日期	类型	大小
test_svd_romeo_juliet.ipynb	2021/11/29 20:15	IPYNB 文件	78 KB

文本表示1：TF-IDF、SVD（Fold in）

- 假设有5个文档
  - 进行奇异值分解

```
u [[-0.396  0.280 -0.571  0.450 -0.102 -0.078  0.280  0.376]
 [-0.314  0.450  0.411  0.513  0.204  0.078 -0.280 -0.376]
 [-0.178  0.269  0.497 -0.257  0.043  0.414  0.243  0.591]
 [-0.438  0.369  0.013 -0.577 -0.220 -0.493  0.037 -0.215]
 [-0.264 -0.346  0.146  0.047  0.417 -0.491 -0.404  0.456]
 [-0.524 -0.246 -0.339 -0.273  0.155  0.571 -0.317 -0.161]
 [-0.264 -0.346  0.146  0.047  0.417 -0.079  0.721 -0.295]
 [-0.326 -0.460  0.317  0.237 -0.725 -0.000  0.000  0.000]]
```

```
s [[ 2.285  0.000  0.000  0.000  0.000]
 [ 0.000  2.010  0.000  0.000  0.000]
 [ 0.000  0.000  1.361  0.000  0.000]
 [ 0.000  0.000  0.000  1.118  0.000]
 [ 0.000  0.000  0.000  0.000  0.797]
 [ 0.000  0.000  0.000  0.000  0.000]
 [ 0.000  0.000  0.000  0.000  0.000]
 [ 0.000  0.000  0.000  0.000  0.000]]
```

```
vh [[-0.311 -0.407 -0.594 -0.603 -0.143]
 [ 0.363  0.541  0.200 -0.695 -0.229]
 [-0.118  0.677 -0.659  0.198  0.233]
 [ 0.861 -0.287 -0.358  0.053  0.212]
 [ 0.128  0.034 -0.209  0.333 -0.910]]
```

文本表示1：TF-IDF、SVD（Fold in）

- 假设有5个文档
  - 进行奇异值分解，选择K=2进行降维

```
u [[-0.396  0.280 -0.571  0.450 -0.102 -0.078  0.280  0.376]
 [-0.314  0.450  0.411  0.513  0.204  0.078 -0.280 -0.376]
 [-0.178  0.269  0.497 -0.257  0.043  0.414  0.243  0.591]
 [-0.438  0.369  0.013 -0.577 -0.220 -0.493  0.037 -0.215]
 [-0.264 -0.346  0.146  0.047  0.417 -0.491 -0.404  0.456]
 [-0.524 -0.246 -0.339 -0.273  0.155  0.571 -0.317 -0.161]
 [-0.264 -0.346  0.146  0.047  0.417 -0.079  0.721 -0.295]
 [-0.326 -0.460  0.317  0.237 -0.725 -0.000  0.000  0.000]]
```

```
s [[ 2.285  0.000  0.000  0.000  0.000]
 [ 0.000  2.010  0.000  0.000  0.000]
 [ 0.000  0.000  1.361  0.000  0.000]
 [ 0.000  0.000  0.000  1.118  0.000]
 [ 0.000  0.000  0.000  0.000  0.797]
 [ 0.000  0.000  0.000  0.000  0.000]
 [ 0.000  0.000  0.000  0.000  0.000]
 [ 0.000  0.000  0.000  0.000  0.000]]
```

```
vh [[-0.311 -0.407 -0.594 -0.603 -0.143]
 [ 0.363  0.541  0.200 -0.695 -0.229]
 [-0.118  0.677 -0.659  0.198  0.233]
 [ 0.861 -0.287 -0.358  0.053  0.212]
 [ 0.128  0.034 -0.209  0.333 -0.910]]
```

文本表示1：TF-IDF、SVD（Fold in）

- 假设有5个文档
  - 进行奇异值分解，选择K=2进行降维

```
u2 [[-0.396  0.280]
 [-0.314  0.450]
 [-0.178  0.269]
 [-0.438  0.369]
 [-0.264 -0.346]
 [-0.524 -0.246]
 [-0.264 -0.346]
 [-0.326 -0.460]]
```

```
s2 [[ 2.285  0.000]
 [ 0.000  2.010]]
```

```
vh2 [[-0.311 -0.407 -0.594 -0.603 -0.143]
 [ 0.363  0.541  0.200 -0.695 -0.229]]
```

文本表示1：TF-IDF、SVD（Fold in）

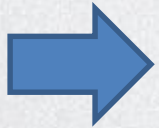
- 假设有5个文档
  - 进行奇异值分解，选择K=2进行降维
  - 单词的低维表示

概念空间的坐标

```
u2 [[-0.396   0.280]
 [-0.314   0.450]
 [-0.178   0.269]
 [-0.438   0.369]
 [-0.264  -0.346]
 [-0.524  -0.246]
 [-0.264  -0.346]
 [-0.326  -0.460]]
```

概念空间的强度

```
s2 [[ 2.285   0.000]
 [ 0.000   2.010]]
```



```
          words [[-0.905   0.563]
romeo      [-0.718   0.904]
juliet     [-0.407   0.541]
happy      [-1.002   0.741]
dagger     [-0.603  -0.695]
live       [-1.198  -0.495]
die        [-0.603  -0.695]
free       [-0.746  -0.924]]
new-hampshire
```

当$D = U\Sigma V^T$

1，词项表示：$U\Sigma$的各个行向量

2，文档表示：ΣV^T的各个列向量

文本表示1：TF-IDF、SVD（Fold in）

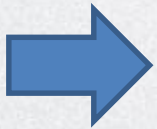
- 假设有5个文档
  - 进行奇异值分解，选择K=2进行降维
  - 单词的低维表示，文档的低维表示

概念空间的强度

```
s2 [[ 2.285   0.000]
 [ 0.000   2.010]]
```

概念空间的坐标

```
vh2 [[-0.311 -0.407 -0.594 -0.603 -0.143]
 [ 0.363   0.541   0.200 -0.695 -0.229]]
```



d1 d2 d3 d4 d5

```
docs [[-0.710 -0.931 -1.359 -1.378 -0.326]
 [ 0.730   1.087   0.402 -1.398 -0.460]]
```

当$D = U\Sigma V^T$

1，词项表示：$U\Sigma$的各个行向量

2，文档表示：ΣV^T的各个列向量

文本表示1：TF-IDF、SVD（Fold in）

文本表示1：TF-IDF、SVD（Fold in）

- 假设有5个文档
  - 现在，新来一个查询，包含两个单词dagger，die
  - 如何对其进行降维?
    - Fold In
  - 推导
    - $\mathrm{A} \approx \mathrm{U_k} S_k V_K^T$
    - $[d_1 d_2 d_3 d_4 d_5] \approx \mathrm{U_k} S_k [\hat{d}_1 \hat{d}_2 \hat{d}_3 \hat{d}_4 \hat{d}_5]$
    - $d_1 \approx \mathrm{U_k} S_k \hat{d}_1$
    - $S_k^{-1} U_k^T d_1 \approx \hat{d}_1$
    - $S_k^{-1} U_k^T q \approx \hat{\mathrm{q}}$

$\hat{\mathrm{q}}$ 低维空间的坐标

    - $S_k \hat{\mathrm{q}} \approx S_k S_k^{-1} U_k^T q = \mathrm{U}_k^T q$

$S_k \hat{\mathrm{q}}$为低维空间向量表示

文本表示1：TF-IDF、SVD（Fold in）

- 假设有5个文档
  - 新来一个查询，包含两个单词dagger，die
  - 如何对其进行降维？
    - Fold In $U_k^T q$

- 新来的查询，可以看作一个文档，用此公式进行Folding in
- 与原有文档进行相似度计算，找出相似文档

```
u2 [[-0.396   0.280]
 [-0.314   0.450]
 [-0.178   0.269]
 [-0.438   0.369]
 [-0.264  -0.346]
 [-0.524  -0.246]
 [-0.264  -0.346]
 [-0.326  -0.460]]
```

T

```
q = [[0],
     [0],
     [0],
     [1], #dagger
     [0],
     [1], #die
     [0],
     [0]]
```

```
q2 [[-0.962]
 [ 0.122]]
```

文本表示1：TF-IDF、SVD（Fold in）

- 假设有5个文档
  - 计算q的低维表示，和d1,d2,d3,d4,d5的低维表示的
    - 夹角余弦
    - 夹角

```
cos_list [ 0.782   0.741   0.987   0.607   0.472]
```

```
degree_list [ 38.532   42.194   9.259   52.639   61.856]
```

```
from most similar to least [3 1 2 4 5]
```

文本表示1：TF-IDF、SVD（Fold in）

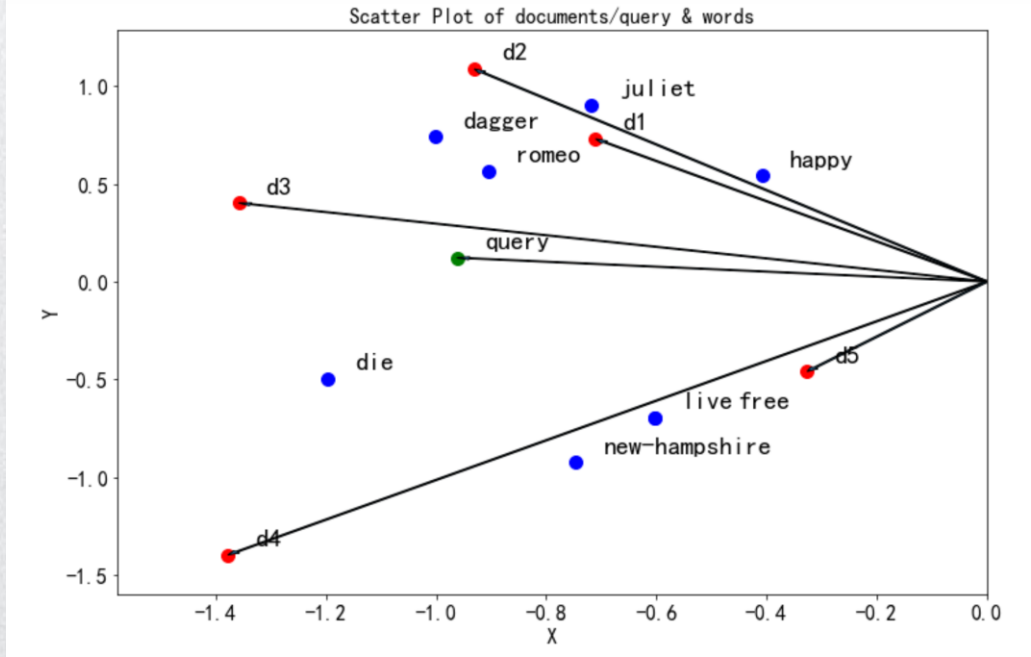
- 假设有5个文档
  - 进行可视化
    - q的低维表示
    - d1,d2,d3,d4,d5的低维表示
    - 各个单词的低维表示

```
from most similar to least [3 1 2 4 5]
```

与query最相似的文档依次为d3,d1,d2,d4,d5

d1 : Romeo and Juliet.
d2 : Juliet: O happy dagger!
d3 : Romeo died by dagger.
d4 : “Live free or die”, that’s the New-Hampshire’s motto.
d5 : Did you know, New-Hampshire is in New-England.

有一个新的查询search query：dies, dagger



文本表示1：TF-IDF、SVD（Fold in）

- 假设有5个文档
  - 进行可视化
    - q的低维表示
    - d1,d2,d3,d4,d5的低维表示
    - 各个单词的低维表示

d1 : Romeo and Juliet.
d2 : Juliet: O happy dagger!
d3 : Romeo died by dagger.
d4 : “Live free or die”, that’s the New-Hampshire’s motto.
d5 : Did you know, New-Hampshire is in New-England.
有一个新的查询search query：dies, dagger

- d3之外，d1为什么和query也很相似呢？甚至比d2还要相似？
- 1.这里我们看到的降维把语义相关的文档和单词聚拢在一起
- 2.可以针对每个文档，看看文档由哪些单词构成，对照右图进行理解

```
from most similar to least [3 1 2 4 5]
```

与query最相似的文档依次为d3,d1,d2,d4,d5