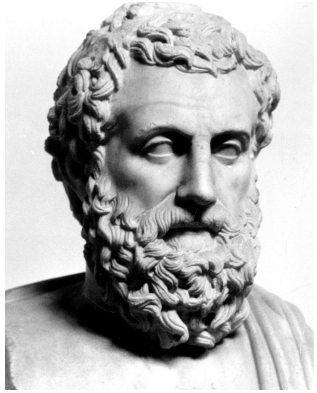
Artificial Intelligence For NLP Lesson-06



人工智能与自然语言处理课程组

2019.May. 04

Outline

1. Keywords
2. TFIDF, TFIDF Vectorized
3. WordCloud
4. Boolean Search
5. PageRank
6. Build Your first Search Engine

Previous Remain

- 1. How to get related words by word2vec?

1. Keywords

Which Words are important?

金正男遇害案成悬案?最后一名嫌犯越南籍女子获释

2019-05-03 11:04:10 来源: 东方网 举报

1113

易信

微信

QQ空间

微博

（原标题：马来西亚释放“谋杀金姓男子”越南女嫌犯）

历经了三年的曲折，世界上最引人注目的谋杀谜团之一却匆匆收场。

据韩联社报道，今天（5月3日）上午7时20分左右，被指控杀害朝鲜最高领导人金正恩同父异母兄弟金正男的第二名女性——越南公民段氏香从马来西亚一所女子监狱出狱。

2. TF-IDF

- Term Frequency – Inverse Document Frequency
  - The Simplest approach is to assign the weight to be equal to the number of occurrences of term *t* in document *d.* *Term Frequency (tf)*
  - It is more commonplace to use *document frequency df,* defined to be the number of documents in the collection that contain term *t.*
  - Denoting as usual the total number of documents in a collection by *N,* we define the *inverse document frequency* (idf) of a term t as follows.

$$\text{idf}_t = \log \frac{N}{\text{df}_t}.$$

$$\text{tf-idf}_{t,d} = \text{tf}_{t,d} \times \text{idf}_t.$$

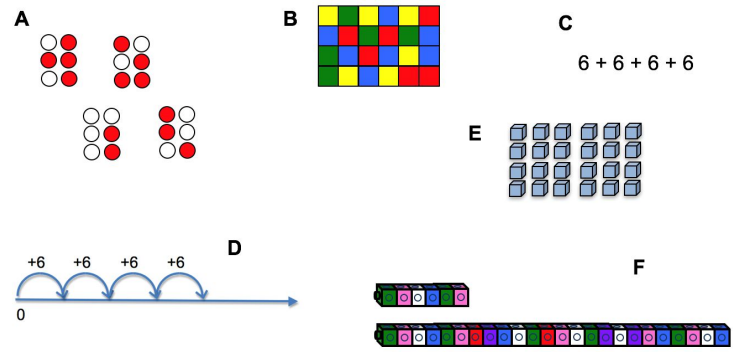
In other words, tf–idf*t,d* assigns to term *t* a weight in document *d* that is

- highest when *t* occurs many times within a small number of documents (thus lending high discriminating power to those documents);
- lower when the term occurs fewer times in a document, or occurs in many documents (thus offering a less pronounced relevance signal);
- lowest when the term occurs in virtually all documents.

- (online-coding for *tf-idf* and *word cloud*)

The Vector space model for scoring

- As we mentioned in Lesson-5, Word2Vec, and in this lesson TFIDF. The representation of a set of documents as vectors in a common vector space is known as the *vector space model* and is fundamental to a host of information retrieval (IR) operations including scoring documents on *a query, document classification, and document clustering.* We first develop the basic ideas underlying vector space scoring; a pivotal step in this development is the view of queries as vector.



The importance of Representation

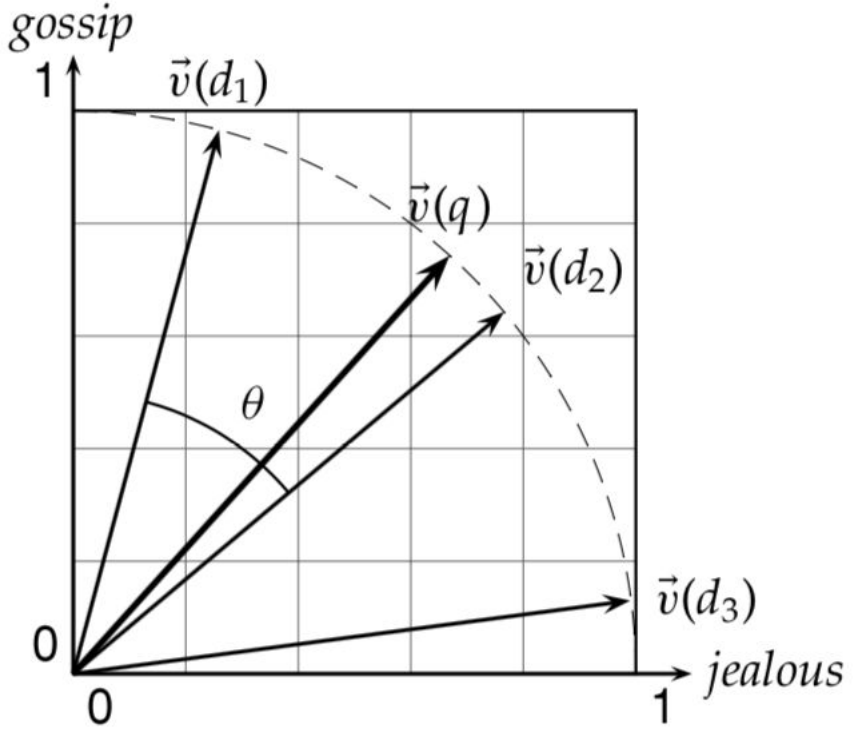
- Representation + Policy

Scikit-Learning TFIDF and Simplest Classification Model

- (on-line coding using scikit learning)

$$\text{sim}(d_1, d_2) = \frac{\vec{V}(d_1) \cdot \vec{V}(d_2)}{|\vec{V}(d_1)||\vec{V}(d_2)|},$$

Cosine similarity illustrated: $\text{sim}(d_1, d_2) = \cos\theta$.



Boolean Search

1. To Process large document collections quickly.

2. To allow more flexible matching operations.

3. To allow ranked retrieval.

	Antony and Cleopatra	Julius Caesar	The Tempest	Hamlet	Othello	Macbeth
Antony	1	1	0	0	0	1
Brutus	1	1	0	1	0	0
Caesar	1	1	0	1	1	1
Calpurnia	0	1	0	0	0	0
Cleopatra	1	0	0	0	0	0
mercy	1	0	1	1	1	1
worser	1	0	1	1	1	0
…						

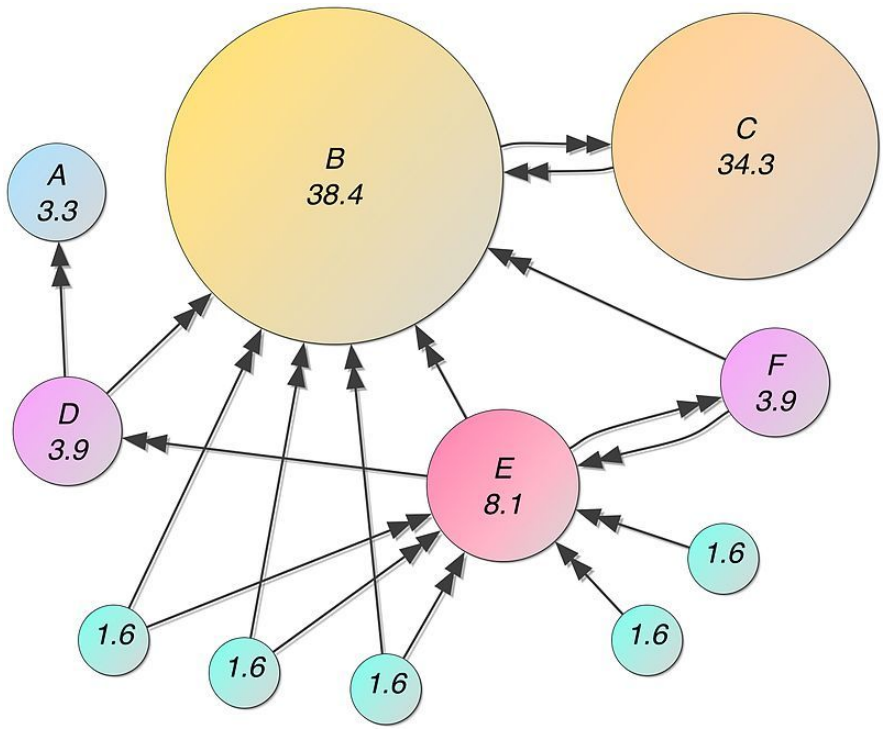
.

$$110100 \text{ AND } 110111 \text{ AND } 101111 = 100100$$

Ranking Using TFIDF

- With TFIDF we could build a search engine.

PageRank



Iterative [edit]

At $t = 0$, an initial probability distribution is assumed, usually

$$PR(p_i; 0) = \frac{1}{N}.$$

where N is the total number of pages, and $p_i; 0$ is page i at time 0.

At each time step, the computation, as detailed above, yields

$$PR(p_i; t+1) = \frac{1-d}{N} + d \sum_{p_j \in M(p_i)} \frac{PR(p_j; t)}{L(p_j)}$$