Vector space reduction

As already said in the text-based ranking's notes, computing the similarity between two docs (cosine) in vector space is costly due to the high-dimensionality, therefore we would like to "pack" our vectors in fewer dimensions while still preserving the similarities

More formally, we would like to compute $\cos(d, q)$ for all n docs in $O(km + kn)$ ($k \ll m$), instead of $O(nm)$

WE WILL SEE TWO ALGEBRA METHODS

LSI: Latent Semantic Indexing

It's a data-dependent method that creates a k-dim subspace by eliminating redundant axes: this is done by "merging" related axes, like synonym and polysemy terms: therefore the axes don't represent a term anymore, but a "concept".

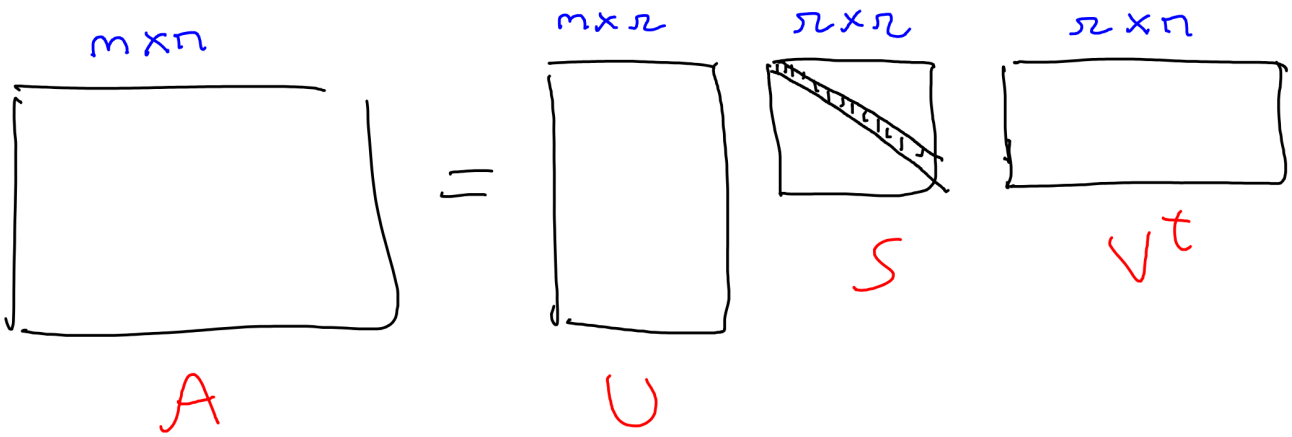
IMO?

In practice, it pre-processes the docs using the Singular Value Decomposition technique, and from there it creates a new smaller vector space for faster querying.

SVD

$$A = USV^t$$

- A = $m \times n$ matrix with rank $r \leq m, n$, where row = Term and column = doc
- U = $m \times r$ matrix, where row = Term and column = eigenvector of T. T is the term-Term correlation matrix, a symmetrical $m \times m$ matrix ($T = AA^t$)
- S = $r \times r$ diagonal matrix, that stores the eigenvalues of T in decreasing order
- V = $n \times r$ matrix, where row = doc and column = eigenvector of D. D is the doc-doc correlation matrix, a symmetrical $n \times n$ matrix ($D = A^t A$)

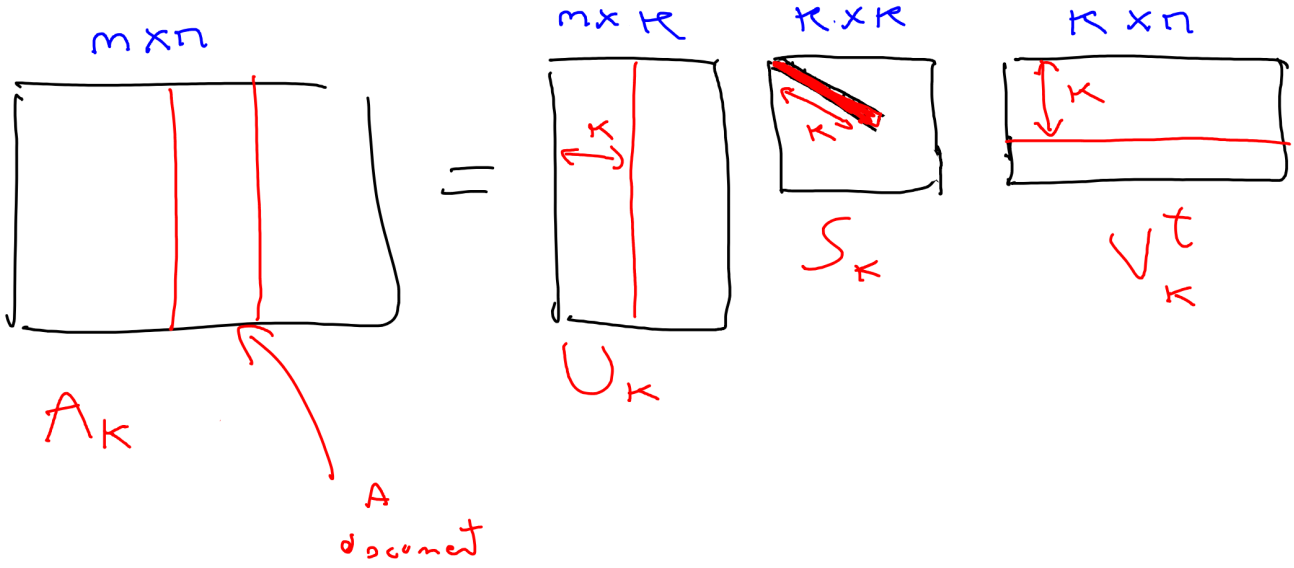


Now let's take a $k << n$ and zero out all but the k biggest eigenvalues in S and call S_k this new version of S.

The rank of S_k is k, which usually is $\simeq 100$, while the rank of A is usually above 10 000.

We then compute the product with U and V^T, which consider only the first k column of U (U_k) and the first k rows of V^T (V_k^T)

$$A_k = U_k S_k V_k^t$$



A_k, of all the matrices $m \times n$ of rank k, is the best approximation of A, because it preserves all the relative distances approximately

$$\|A - A_k\| = \min_{B \in \mathbb{R}^{m \times n}} \|A - B\|$$

BUT IS AS BIG AS A

Since we are interested in doc-doc similarity, Let's see how to write D now:

$$D = A^t A = (USV^t)^t (USV^t) =$$

$$= (SV^t)^t (SV^t) \quad n \times n$$

Take $X = SV^t$ and take its rank k approximation

$$X_k = S_k V^t \quad k \times n$$

This is a good approx of D, because

$$A^t A \simeq X_k^t X_k \quad \text{(both } n \times n\text{)}$$

We can use X_K to define how to project A, because

$$X_K = S_K V_K^t \Rightarrow X_K \simeq U_K^t A$$

But X_K has way less rows, because K<<n

therefore we've compressed the terms in "concepts"

and the columns represent the projection of the docs

Now we have to also project the query to the new space: Since we can see the query vector as a new column of A, $Q' = U_K^T Q$ is the projected query (O(rm) complexity)

Which are the concepts?

c-th concept = c-th col of U_k $^{m \times k}$

- $U_k[i][c]$ = strength of association between c-th concept and i-th term
- $V_k^t[c][j]$ = strength of association between c-th concept and j-th document
- projected document $d'_j = U_k^t d_j$

  $d'_j[c]$ = strength of concept c in d_j
- projected query $q' = U_k^t q$

  $q'[c]$ = strength of concept c in q

RANDOM PROJECTION

It is a data-independent reduction method that choose randomly a k-dim vector subspace that guarantees good stretching properties between any pair of points, with high probability.

It is based on the Johnson-Lindenstrauss lemma that states that exist a function, called JL-embedding which project a vector into a smaller space preserving (almost) the distances between points.

- Let P be a set of n distinct points in m dimension ($k \ll m$)
- Given $\varepsilon > 0$, exists a projection function $f: \mathbb{R}^m \to \mathbb{R}^k$ such that:

$$\forall (u, v) \in P:$$

$$(1-\varepsilon) \cdot \|u-v\|^2 \leq \|f(u) - f(v)\|^2 \leq (1+\varepsilon) \|u-v\|^2$$

And $k = O(\varepsilon^{-2} \cdot \log n)$

The lemma applies for the euclidean distance, but we can arrange it for the cosine similarity:

$$u, v \in P. \quad 2u \cdot v = \|u\|^2 + \|v\|^2 - \|u-v\|^2$$

$$2 \cdot f(u) \cdot f(v) = \underbrace{\|f(u)\|^2 + \|f(v)\|^2}_{\wedge|} - \underbrace{\|f(u) - f(v)\|^2}_{\wedge| \text{ LEMMA}}$$

$$2 f(u) \cdot f(v) \leq (\|u\|^2 + \|v\|^2)(1+\varepsilon) - \|u-v\|^2 (1-\varepsilon)$$

Since $\|u-v\|^2 = \|u\|^2 + \|v\|^2 - 2u \cdot v$

$$2 \cdot f(u) \cdot f(v) \leq (\|u\|^2 + \|v\|^2)(1+\varepsilon) - (\|u\|^2 + \|v\|^2 - 2uv)(1-\varepsilon) =$$

$$= (\|u\|^2 + \|v\|^2)(1+\varepsilon) + (2uv - \|u\|^2 - \|v\|^2)(1-\varepsilon) =$$

$$= (\|u\|^2 + \|v\|^2)(1+\varepsilon) + 2uv - 2\varepsilon uv - \|u\|^2 +$$

$$+ \varepsilon \|u\|^2 - \|v\|^2 + \varepsilon \|v\|^2 =$$

$$= \cancel{\|u\|^2} + \varepsilon \|u\|^2 + \cancel{\|v\|^2} + \varepsilon \|v\|^2 + 2uv -$$

$$- 2\varepsilon uv - \cancel{\|u\|^2} + \varepsilon \|u\|^2 - \cancel{\|v\|^2} + \varepsilon \|v\|^2 =$$

$$= 2\varepsilon \|u\|^2 + 2\varepsilon \|v\|^2 + uv(2 - 2\varepsilon) =$$

$$= \varepsilon \|u\|^2 + \varepsilon \|v\|^2 + uv(1-\varepsilon)$$

$$f(u) \cdot f(v) \leq \varepsilon \left(\|u\|^2 + \|v\|^2 \right) + u \cdot v \, (1 - \varepsilon)$$

if u, v are normalized, $\|u\|, \|v\| = 1$, therefore $\cos(u,v)$ changes by at most 2ε

How to construct f?

Any $m \times k$ projection matrix P using any random distribution with mean $\mu = 0$ and variance $\sigma = 1$

EXAMPLE

$$p_{ij} = \begin{cases} 1 & \text{prob } 1/2 \\ -1 & \text{prob } 1/2 \end{cases}$$

$$p_{ij} = \begin{cases} 1 & \text{prob } 1/3 \\ -1 & \text{prob } 1/3 \\ 0 & \text{prob } 1/3 \end{cases}$$