string kernel

- similarities between two documents

$\sum$=alphabet, $\sum^n$=set of all strings of length n

for a given index sequence $\mathbf{i} = (1 \le i_1 < i_2 < ... < i_r \le |s|)$

define $s(\mathbf{i}) := s(i_1)s(i_2)....s(i_r)$ and $l_s(\mathbf{i}) = i_r - i_1 + 1 \ge r$

example $s = \texttt{fast food}$,$\mathbf{i} = (2,3,9) \Rightarrow s(\mathbf{i}) = \texttt{asd}, l_s(\mathbf{i}) = 9 - 2 + 1 = 8$

$0 < \lambda \le 1$ parameter, define $[\Phi_n(s)]$ a map with $|\sum^n|$ components

$$[\Phi_n(s)]_u = \sum_{\mathbf{i}:s(\mathbf{i})=u} \lambda^{l_s(\mathbf{i})}$$

example $[\Phi_3(\texttt{Nasdaq})]_{\texttt{asd}} = \lambda^3$, $[\Phi_3(\texttt{lass das})]_{\texttt{asd}} = 2\lambda^5$

the kernel induced

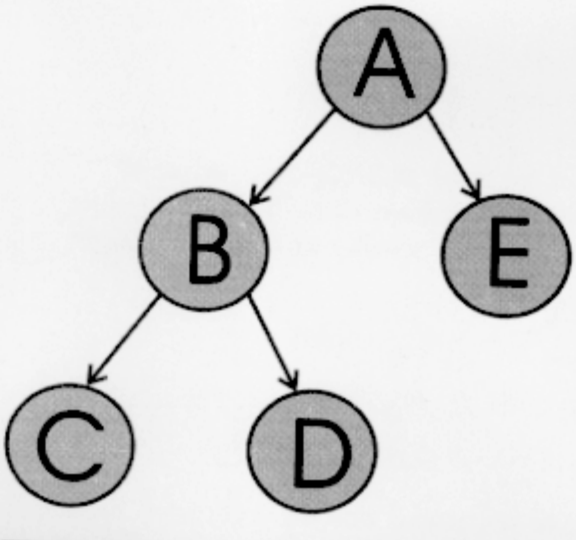
$$k_n(s,t) = \sum_{u \in \sum^n} [\Phi_n(s)]_u [\Phi_n(t)]_u = \sum_{u \in \sum^n} \sum_{(\mathbf{i},\mathbf{j}):s(\mathbf{i})=t(\mathbf{j})=u} \lambda^{l_s(\mathbf{i})} \lambda^{l_t(\mathbf{j})}$$

$k := \sum_n c_n k_n$ linear combination of kernels on different substring-lengths

tree kernel

- encode a tree as a string by traversing in preorder and parenthesing

- substrings correspond to subset trees



- tag can be computed in loglinear time

- then use a string kernel

`tag(T)=(A(B(C)(D))(E))`