Machine learning 2
Exercise sheet 6

FLEISCHMANN Kay, Matrnr: 352247
ROHRMANN Till, Matrnr: 343756

May 22, 2013

1 String Kernels

Given the kernel function

$$\begin{aligned} k(\cdot,\cdot) : \mathcal{A}^\star \times \mathcal{A}^\star &\to \mathbb{R} \\ (x,y) &\mapsto \sum_{s\in\mathcal{A}^\star} \#(s \subseteq x)\cdot\#(s\subseteq y)\cdot w(s) \end{aligned}$$

1.1 Unweighted bag-of-words

$w(s) = 1$ iff s is a word of the target language L.

Prove that the unweighted bag-of-words is a kernel function. Let $K = (k(x_i, x_j)_{i,j}$ be the kernel matrix produced by the strings x_i with $i \in [1, N]$. Let $v \in \mathbb{R}^N$ be an arbitrary vector.

$$\begin{aligned} \sum_{i,j=1}^{N} v_i k(x_i,x_j) v_j &= \sum_{s\in\mathcal{A}^\star} w(s) \sum_{i,j=1}^{N} v_i \#(s\subseteq x_i)\cdot\#(s\subseteq x_j) v_j \\ &= \sum_{s\in\mathcal{A}^\star} w(s) \underbrace{\left(\sum_{i=1}^{N} v_i\cdot\#(s\subseteq x_i)\right)^2}_{\geq 0} \\ &\geq 0 \end{aligned}$$

Thus the function $k(\cdot,\cdot)$ is positive semi-definite.

□

1.2 Inverse document frequency

$w(s) = IDF(s) = \log N - \log DF(s)$ with $DF(s) = \#\{k : 1 \leq k \leq N, s \subseteq D_k\}$.

Prove that the function $k(\cdot,\cdot)$ with the inverse document frequency is a kernel function. Let $K = (k(x_i, x_j)_{i,j}$ be the kernel matrix produced by the strings x_i with $i \in [1, N]$. Let $v \in \mathbb{R}^N$ be an arbitrary vector.

$$\begin{aligned} \sum_{i,j=1}^{N} v_i k(x_i,x_j) v_j &= \sum_{s\in\mathcal{A}^\star} w(s) \sum_{i,j=1}^{N} v_i \#(s\subseteq x_i)\cdot\#(s\subseteq x_j) v_j \\ &= \sum_{s\in\mathcal{A}^\star} \underbrace{\log\left(\frac{N}{DF(s)}\right)}_{\geq 0} \underbrace{\left(\sum_{i=1}^{N} v_i\cdot\#(s\subseteq x_i)\right)^2}_{\geq 0} \\ &\geq 0 \end{aligned}$$

Thus the function $k(\cdot,\cdot)$ is positive semi-definite.

□

1.3 n-spectrum kernel

$w(s) = 1$ iff $|s| = n$.

Prove that the n-spectrum kernel is indeed a kernel. Defining our target language as $L = A^n$ and using the results from subsection 1.1 leads directly to the assumption.

$\square$

1.4 Blended n-spectrum kernel

$w(s) = 1$ iff $|s| \leq n$.

Prove that the blended n-spectrum kernel is indeed a kernel. Defining our target language $L = \bigcup_{i=0}^{n} A^i$ and using the results from subsection 1.1 leads directly to the assumption.

$\square$

1.5 Kernel matrices

Let $n = 3$ and the data set is

ananas, anna, natter, otter, otto

1.5.1 n-spectrum kernel

$$K_{spec} = \begin{array}{c} \\ \text{ananas} \\ \text{anna} \\ \text{natter} \\ \text{otter} \\ \text{otto} \end{array} \begin{array}{c} \begin{array}{ccccc} \text{ananas} & \text{anna} & \text{natter} & \text{otter} & \text{otto} \end{array} \\ \begin{pmatrix} 6 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 4 & 2 & 0 \\ 0 & 0 & 2 & 3 & 1 \\ 0 & 0 & 0 & 1 & 2 \end{pmatrix} \end{array}$$

1.5.2 Blended n-spectrum kernel

$$K_{bspec} = \begin{array}{c} \\ \text{ananas} \\ \text{anna} \\ \text{natter} \\ \text{otter} \\ \text{otto} \end{array} \begin{array}{c} \begin{array}{ccccc} \text{ananas} & \text{anna} & \text{natter} & \text{otter} & \text{otto} \end{array} \\ \begin{pmatrix} 29 & 14 & 7 & 0 & 0 \\ 14 & 12 & 5 & 0 & 0 \\ 7 & 5 & 17 & 11 & 5 \\ 0 & 0 & 11 & 14 & 9 \\ 0 & 0 & 5 & 9 & 13 \end{pmatrix} \end{array}$$

2 Tree kernels

Prove: A string x contains $\mathcal{O}(|x|^2)$ substrings. The number of all substrings of a string x is the sum of all substrings with length 1, length 2, ..., length $|x|$. Thus

$$\begin{aligned} \#_{\text{substrings}}(x) &= \sum_{l=1}^{|x|} \#_{\text{substrings of length } l}(x) \\ &\leq \sum_{l=1}^{|x|} |x| - l + 1 \end{aligned} \tag{1}$$

The inequality (1) holds because there at most $|x| - l + 1$ substrings of length l in a string of length $|x|$. There might also be fewer, if substrings occur mutliple times.

$$\begin{aligned}(1) &= |x|^2 - \frac{(|x|+1)|x|}{2} + |x| \\ &= \frac{|x|^2 + |x|}{2} \\ &\in \mathcal{O}(|x|^2)\end{aligned}$$

□

Prove: A substring w of x can be reached in $\mathcal{O}(|w|)$ in the suffix tree of x. If the string x contains the substring w then there is also a suffix starting with w. Thus, by traversing the suffix tree, following the edges with the corresponding letter, we'll find the substring w at the depth $|w| \in \mathcal{O}(|w|)$. If the substring w is not contained in x, then at latest after $|w| - 1$ steps there is no edge anymore to follow, indicating that there is no substring w.

□

Prove: The suffix tree of x can be stored in $\mathcal{O}(|x|)$ space. In a suffix tree each leaf denotes exactly one suffix. Since there are exactly $|x|$ suffixes in a string of length $|x|$, each suffix tree has $|x|$ leaves. Furthermore, each interior node, except for the root which can also have only one child, has at least 2 children. Thus, there can only be a maximum of $|x| - 1$ interior nodes, because every interior nodes adds at least two leaves to the tree while consuming one open connection of another interior node. Consequently, the maximum number of nodes is $|x| - 1 + |x| + 1 = 2|x| \in \mathcal{O}(|x|)$ and therefore we can store the suffix tree in linear space. □