Übung zur Vorlesung

Maschinelles Lernen 2

Sommersemester 2009

Abteilung Maschinelles Lernen
Institut für Softwaretechnik und
theoretische Informatik
Fakultät IV, Technische Universität Berlin
Prof. Dr. Klaus-Robert Müller
Email: krm@cs.tu-berlin.de

Blatt 10

Abgabe bis *Mittwoch*, 1. Juli 2009 bis 13 Uhr, Ausarbeitung im Sekretariat FR6052, oder bei Mikio Braun, FR6058, Lösung des praktischen Teils per Email an mikio@cs.tu-berlin.de.

Aufgaben

Auf dem letzten Übungsblatt wurde der Weighted-Degree-Kernel per Email an mikio@cs.tu-berlin.devorgestellt. Zur Erinnerung, wenn $\boldsymbol{u}_{k,l}(x)$ der Teilstring der Länge k an der Position l des Strings $\boldsymbol{x}$ ist, so ist der WDK definiert durch

$$\mathrm{k}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \sum_{k=1}^{K} \beta_k \sum_{l=1}^{L-k+1} \mathbf{I}(\boldsymbol{u}_{k,l}(\boldsymbol{x}_i) = \boldsymbol{u}_{k,l}(\boldsymbol{x}_j)).$$

Auf diesem Übungsblatt soll der WDK zur Detektierung von Splicestellen verwendet werden. Wir verwenden hierzu die SVM Implementierung SVM^{light} (http://svmlight.joachims.org/).

SVM^{light} unterstützt von Hause aus nicht den WDK, daher werde wir explizit die Merkmale

$$\phi_{w,k,l} = \sqrt{\beta_k}\mathbf{I}(\boldsymbol{u}_{k,l}(\boldsymbol{x}) = w).$$

für alle $w \in \{A, C, G, T\}^k$, $k \in \{1, \ldots, K\}$, $l \in \{1, \ldots |\boldsymbol{x}| - k\}$ berechnen und dann einen linearen Kern verwenden.

1. **15 Punkte** Vervollständige im Programmskelett die Funktion `write_wdk_features`, die die WDK Merkmale wie oben beschrieben in eine SVM^{light}-Datei schreibt.

2. **15 Punkte** Trainiere SVM^{light} auf dem Splice-Datensatz für $K = \{1, 2, 3\}$ mit Regularisierungskonstante $C = \{0.001, 0.01, 0.1, 1, 10\}$ und messe die Vorhersagegenauigkeit auf dem Trainings- und Testdatensatz. Erzeuge für jedes k einen Plot, der diese Genauigkeiten für Training und Test gegen C plotte.

   Gib die Ergebnisse als Tabelle zusammen mit diesen drei Plots ab.

```
function sheet10

X = textread('splice-train-data.txt', '%s');
Y = load('splice-train-label.txt');
XE = textread('splice-test-data.txt', '%s');
YE = load('splice-test-label.txt');

tic
fprintf('Writing training features for k = 1\n');
write_wdk_features('wdk1-train.txt', 1, X, Y);
fprintf('Writing test features for k = 1\n');
write_wdk_features('wdk1-test.txt', 1, XE, YE);
toc

tic
fprintf('Writing training features for k = 2\n');
write_wdk_features('wdk2-train.txt', 2, X, Y);
fprintf('Writing test features for k = 2\n');
write_wdk_features('wdk2-test.txt', 2, XE, YE);
toc

tic
```

```
fprintf('Writing training features for k = 3\n');
write_wdk_features('wdk3-train.txt', 3, X, Y);
fprintf('Writing test features for k = 3\n');
write_wdk_features('wdk3-test.txt', 3, XE, YE);
toc

function beta = beta(K, k)
beta = 2 * (K - k + 1) / K / (K + 1);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
% Your solution below
%

% 1. write out weighted degree kernel features
% out into file FN up to degree K. (You should accept values of K = 1, 2,
% 3
function write_wdk_features(FN, K, X, Y)
% ...
```