Übung zur Vorlesung

Maschinelles Lernen 2

Sommersemester 2008

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 13

Abgabe: Montag 14. Juli 2008, bis 12 Uhr bei Mikio Braun (FR6058) oder im Sekretariat (FR6052). Praktische Übungsaufgaben über PASS abgeben (`https://ml01.zrz.tu-berlin.de/~mikio/pass.pl?conf=blatt13.conf`). Verwende matlab (`/home/ml/ml/bin`), oder octave (`http://www.octave.org` frei verfügbar).

Aufgaben

1. **Eigennamenerkennung mit HMMs (20 Punkte)** Unter *www.cs.tu-berlin.de/∼brefeld/data* findest Du die Datei *esp.train*. Sie enthält Sätze aus einem spanischen Nachrichtenarchiv, in dem Eigennamen annotiert sind. Jedes Token trägt eines der folgenden Label (in BIO Notation): *B-PER*, *I-PER*, *B-LOC*, *I-LOC*, *B-ORG*, *B-ORG*, *B-MISC*, *I-MISC* und *O(utside)* (hierbei steht *B-* für "Begin", *I-* für "Inside"). Trainiere auf den Daten ein Hidden Markov Modell welches die annotierten Eigennamen erkennt. Gehe dabei wie folgt vor:

   (a) Schreibe ein Skript `estimate_hmm` (ggf. auch in einer Skriptspraceh, jedoch **kein** Java), welches aus den ersten 5000 Sätze die Transitions- und Emissionswahrscheinlichkeiten schätzt. Dieses Skript muß zur weiteren Verarbeitung in matlab auch wieder die Wörter und Zustände auf entsprechende Zahlen abbilden. Das Skript gibt daher neben den Übergangswahrscheinlichkeiten auch zwei Felder aus, die die Wörter und Zustände auflisten.

   (b) Schreibe ein Skript `predict_names.m` welches unter Benutzung des Viterbi-Algorithmus vom letzten Übungsblatt zur Vorhersage der Eigennamen (bzw. eine logarithmierte Version falls numerische Probleme auftreten sollten). Berichte sowohl den 0/1 Verlust ($\Delta = [[y_i \neq \hat{y}]]$) als auch den Sequenzverlust ($\Delta = \sum_{k=1}^{|x_k|} [[y_{i,k} \neq \hat{y}_k]]/|x_k|$).

   (c) **Freiwillige Zusatzaufgabe** Lade die HMM-SVM^{struct} herunter (*http://www.cs. cornell.edu/People/tj/svm_light/svm_hmm.html*). Benutze die Histogramme der Emissionswahrscheinlickheiten als Eingabemerkmale (siehe auch mitgeliefertes Beispiel). Wiederhole den Versuchsaufbau mit der HMM-SVM (benutze einen linearen Kern mit dem default C). Wie verhält sich die Performance im Gegensatz zum HMM?

2. **Duales Optimierungsproblem (20 Punkte)** Gegeben sei eine Trainingsmenge (x_1, y_1), …, (x_n, y_n) mit strukturierten Eingabevariablen $x_i \in \mathcal{X}$ und Ausgabevariablen $y_i \in \mathcal{Y}$. Desweiteren sei $\Delta : \mathcal{Y} \times \mathcal{Y} \to \Re_0^+$ ein beliebiger strukturierter Verlust. Das primale Optimierungsproblem der strukturierten SVM mit Margin-Rescaling ist jetzt gegeben durch

$$\begin{aligned} \min_{w,\xi} \quad & \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n} \xi_i \\ \text{s.t.} \quad & \forall_{i=1}^{n} \forall_{\bar{y} \neq y_i} : \ \langle w, \Phi(x_i, y_i) - \Phi(x_i, \bar{y}) \rangle \geq \Delta(y_i, \bar{y}) - \xi_i \\ & \forall_{i=1}^{n} : \ \xi_i \geq 0. \end{aligned}$$

   Zeige, dass das äquivalente duale Optimierungskriterium gegeben ist durch

$$\begin{aligned} \max_{\alpha} \quad & \sum_{i=1}^{n} \sum_{\bar{y} \neq y_i} \alpha_i(\bar{y}) \Delta(y_i, \bar{y}) - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{\bar{y} \neq y_i} \sum_{\bar{y}' \neq y_j} \alpha_i(\bar{y}) \alpha_j(\bar{y}') \langle \delta\Phi(i, \bar{y}), \delta\Phi(j, \bar{y}') \rangle \\ \text{s.t.} \quad & \forall_{i=1}^{n} \forall_{\bar{y} \neq y_i} : \ \alpha_i(\bar{y}) \geq 0. \end{aligned}$$

   wobei $\delta\Phi(i, \bar{y}) := \Phi(x_i, y_i) - \Phi(x_i, \bar{y})$.