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
http://ml.cs.tu-berlin.de

Blatt 5

Abgabe: 20. Mai 2008, bis 12.15 h bei Mikio Braun (mikio@cs.tu-berlin.de)

Für praktische Aufgaben bitte ebenfalls Code und Ausgabe abgeben. Verwende Matlab oder Octave, welches unter www.octave.org frei verfügbar ist. **Da auf diesem Blatt viel mit Text gearbeitet wird, sind zur Vorverarbeitung auch andere Sprachen als Matlab zulässig, versuche jedoch, Dich auf perl, python oder ruby zu beschränken.**

Aufgaben

1. **Wörterbuch (25 Punkte)** Unter www.cs.tu-berlin.de/∼brefeld/data befinden sich Nachrichtentexte, die dem Reuters Korpus entnommen wurden. Entsprechend den Dateinamen handelt es sich entweder um Beispieldokumente der Klasse *interest* oder um Beispiele einer anderen Klasse, hier der Einfachheit halber mit *not-interest* bezeichnet.

   - Schreibe ein Skript `make_dictionary`, dass ein Wörterbuch aller in den Dokumenten enthaltenen Terme erzeugt. Schreibe das Ergebnis in eine von Matlab lesbare Datei, die drei Arrays enthält, `word_counts` für die Anzahl der Auftreten insgesamt, `doc_counts` für die Anzahl der Auftreten in Dokumenten, und ein Cell-Array `words`, welches die Wörter enthält. Der Zusammenhang soll jeweils so sein, dass derselbe Index dasselbe Wort bezeichnet.
   - Welches sind die 20 häufigsten Terme, welche haben die 20 niedrigste Frequenz?
   - Plotte die Termfrequenz gegen die Frequenz der Termfrequenzen und das Zipfsche Gesetz zum Vergleich. Folgt die empirische Verteilung dem Zipfschen Gesetz?
   - Wie erklären sich die Funde?

2. **Klassenbasierte Termverteilung (5 Punkte)** Wende die Wörterbuchfunktion nun auf jede Klasse einzeln an.

   - Plotte die entstehenden Verteilungen. Was geschieht mit den Verteilungen?
   - Schreibe ein Skript (basierend auf der Ausgabe von Aufgabe 1, am besten im Matlab) `find_discriminative_terms`, welches versucht, besonders diskriminative Terme (d.h., die es erlauben, die beiden Klassen gut zu trennen) zu ermitteln. Berichte von dem Ergebnis.

3. **TF.IDF Repräsentation (15 Punkte)** In dieser Aufgabe sollen die TF.IDF Repräsentation auf den Datensatz angewendet werden.

   - Schreibe ein Skript `tf_idf`, welches zusätzlich noch die TF.IDF-Scores ausrechnet.
   - Liste die 20 diskriminativsten Merkmale und die 20 am wenigsten diskriminativen Merkmale auf.
   - Vergleiche die Ergebnisse mit denen der vorherigen Aufgabe. Welche Gemeinsamkeiten/Unterschiede gibt es?