Praktikum Data Mining

Dokument Klassifikation / Spam Filter

Oliver Fesseler Maria Florusß Stefan Seibert

Daniel Grießhaber

26. Juni 2014

Durchführung

Implementierung: Dokument Klassifikation

Feststellungen Wenn nur zwischen zwei Kategorien unterschieden wird (in diesem Fall 'Tech' und 'Non-Tech'), dann werden zwar die meisten tatsächlich Technik bezogenen Artikel der Technik zugeordnet, jedoch gilt selbes auch für viele nicht technische Artikel. Dies liegt zum Teil daran, dass in Technik bezogenen Artikeln auch viele Wörter vorkommen, die in nicht Technik bezogenen Artikeln vorkommen. Anders herum ist dies eher weniger der Fall, da 'Non-Tech' ein sehr weiter Bereich ist. So können dort Artikel aus Politik, Sport, Wirtschaft und Kunst auftauchen. Im Non-Tech-Bereich sind sehr viel mehr verschiedene Worte aufgefürt, von denen jedoch jedes ein relativ kleines Gewicht trägt. Kommt nun in einem nicht Technik bezogenen Artikel das Wort 'Computer' vor, jedoch auch die Worte 'DAX', 'Bank' und 'Eurorettungsschirm', wird Computer möglicherweise mit einem Wert von 0.9 für Tech bewertet, während die drei anderen Worte jeweils nur mit 0.1 für Non-Tech bewertet werden. So würde dieses Dokument Tech zugeordnet werden, obwohl es viel mehr Worte aus anderen Bereichen enthält, die jedoch nur gering gewertet werden. Es sollte also darauf geachtet werden, dass die beiden Zielgruppen eine etwa gleich mächtige Wortmenge aufweisen.

Zu einem weiteren Teil liegt die falsche Einteilung der Non-Tech Artikel daran, dass in einem eigentlich Technik bezogenen RSS-Feed wie etwa Heise, durchaus auch nicht technische Artikel vorkommen können, beispielsweise Artikel /über Edward Snowden oder die NSA-Affäre, die für die technisch interessierte Zielgruppe zwar interessant sind, jedoch eher Politik zugeordnet werden würden. Dies kann dann dazu führen, dass andere Politik bezogene Artikel ebenfalls Technik zugeordnet werden, obwohl sie damit nur sehr wenig oder nichts zu tun haben. Man sollte also sehr genau darauf achten, welche Artikel man zum Training verwendet.

Um diesen Effekt des falsch Klassifizierens muss sich jedoch nur Gedanken gemacht werden, wenn es wichtig ist, dass beide Klassen richtig kategorisiert werden. Wenn man beispielsweise aus einer Masse an Dokumenten nur Technik bezogene Artikel filtern will,

dann ist es nicht schlimm, dass zusätzlich einige nicht Technik bezogene Artikel in der resultierenden Auswahl vorhanden sind, solange alle Technik bezogenen Artikel auf jeden Fall darin vorkommen.

Nur wenn beide auf jeden Fall richtig klassifiziert werden sollen, beispielsweise wenn eine Masse an Dokumenten in traurige und fröhliche Texte klassifiziert werden soll, dann ist ein solcher Effekt nicht wünschenswert.

Der Effekt des falschen Klassifizierens bei nur einer genau spezifizierten Klasse (Tech vs. Non-Tech) ist sehr deutlich an einem Experiment erkennbar, das wir zusätzlich durchgeführt haben:

Neben Technik haben wir noch weitere Klassen eingeführt, wie Sport, Politik, Wirtschaft und Wissenschaft. Die Technik bezogenen Artikel wurden weiterhin weitestgehend korrekt als Tech-Artikel klassifiziert, von den Non-Tech-Artikeln, die zuvor Tech zugeordnet wurden, wurden nun jedoch viele in ihre jeweils korrekte Kategorie eingeordnet.

Weiterhin haben wir festgestellt, dass es extrem ausschlaggebend ist, mit wie vielen Testdokumenten man trainiert. Während bei nur 3 RSS-Feeds, beziehungsweise etwa 75 Dokumente pro Kategorie noch die meisten Dokumente falsch klassifiziert wurden, zeigte sich bereits bei 5 RSS-Feeds, beziehungsweise etwa 150 Dokumente pro Kategorie ein deutlicher Anstieg der richtig klassifizierten Dokumente.

Dennoch werden weiterhin etliche Dokumente falsch klassifiziert. Dies könnte folgende Ursachen haben: Beispielsweise ist die Kategorie Wissenschaft sehr viel breit gefächerter, was vorkommende Worte angeht, als beispielsweise Sport. In die Wissenschaftskategorie fallen verschiedene Artikel aus den Bereichen Neurowissenschaft, Astronomie, Anthropologie oder Physik, die alle ein sehr unterschiedliches Feld an Begriffen aufweisen, während sich in der Kategorie Sport erstens die meisten Artikel um Fußball drehten und selbst wenn nicht, viele der Worte trotzdem auch in anderen Sportarten vorkommen.

So stellten wir fest, dass die meisten Sport bezogenen Artikel richtig klassifiziert wurden, während kaum ein Wissenschaftsartikel richtig eingeordnet wurde. (Allerdings könnte das auch zu einem gewissen Teil daran gelegen haben, dass in den allgemeinen RSS-Feeds, die wir als Test nutzten, kaum wissenschaftsbezogene Artikel vorkamen, jedoch aufgrund der Fußballweltmeisterschaft entsprechend viele Sportartikel.)

Als weitere Kategorie, die eingeführt werden könnte, bietet sich eine Art Feuilleton-Kategorie an, in die Artikel über Prominente und ähnliches fallen. Allerdings ist es schwierig hier geeignete Feeds zu finden, da diese Kategorie wiederum sehr breit gefächert ist. Artikel die dieser Klasse zugeordnet werden würden, kommen häufig nur in allgemeinen Nachrichten-Feeds vor, die sich für das Training nicht eignen. Allerdings gibt es in den Test-Feeds einige Artikel, die in diese Kategorie fallen würden, und aufgrund der fehlenden Kategorie falsch Klassifiziert werden.

Abschließend kan also festgehalten werden, dass bei der Dokumenten-Klassifikation auf folgende Dinge zu achten ist:

- Es sollten für jede Kategorie genügend Testdaten zur Verfügung stehen
- Die Kategorien sollten falls möglich von ihrem Wortumfang (also die Anzahl der signifikanten Worte für diese Kategorie) in etwa vergleichbar sein.
- Für die Variable `initprob` sollten verschiedene Werte getestet werden, bis die Klassifizierung ein bestmögliches Ergebnis liefert.

Beantwortung der Fragen zum Versuch

Was wird mit Evidenz bezeichnet und warum muss diese für die Klassifikation nicht berücksichtigt werden?

Die Evidenz p(x) ist im Beispiel die Wahrscheinlichkeit dafür, dass das Wort x überhaupt vorkommt unabhängig davon, welcher Klasse die Dokumente, in denen es vorkommt, zugeordnet werden. Deshalb ist die Evidenz für alle Klassen gleich und muss für die Entscheidung nicht miteinbezogen werden.

Der eigentliche Wahrscheinlichkeitswert, mit der ein Dokument einer Klasse zugeordnet wird ist nicht relevant. Es ist nur relevant, für welche Klasse der Wert am größten ist. Deshalb ist es nicht wichtig, alle diese Werte noch einmal durch den selben Wert (die Evidenz) zu teilen.

Wann würden Sie in der Formel für die gewichtete Wahrscheinlichkeit den Wert von `initprob` kleiner, wann größer als 0.5 wählen? (Falls Sie die Möglichkeit haben diesen Wert für jedes Feature und jede Kategorie individuell zu konfigurieren)

- Wenn viele Dokumente eingelesen werden, die mit hoher Wahrscheinlichkeit viele unbekannte Worte enthalten sollte der Wert von `initprob` möglichst klein gewählt werden. Ansonsten würde ein Dokument eher einer Klasse zugeordnet, die viele seiner Worte nicht kennt, da diese einen relativ großen Wert zugeschrieben bekommen.
- Der Wert sollte auch dann niedirg gewählt werden, wenn die bekannten Worte relativ kleine Wahrscheinlichkeitswerte aufweisen. Wenn alle Worte im Durchschnitt nur mit einer Wahrscheinlichkeit von 0.1 vorkommen und für nicht bekannte Worte das `initprob` auf 0.5 gesetzt ist, kann dies das Ergebnis erheblich verfälschen.
- Wenn der Wortschatz hingegen relativ klein ist und somit alle Wörter relativ große Wahrscheinlichkeitswerte aufweisen, dann kann auch `initprob` größer gewählt werden.

Was könnten Sie mit dem in dieser Übung implementierten Classifier noch klassifizieren? Geben Sie eine für Sie interessante Anwendung an.

- Spamfilter - Das Einteilen von E-Mails in nicht schädliche und potentiell schädliche oder Werbung.
- Klassifikation von Webseiten - Ähnlich wie die im Versuch behandelte Dokumentenklassifikation, nur bezogen auf Webseiten.
- Diagnose, beispielsweise von Krankheiten - Für verschiedene 'Trainings'-Patienten, die an verschiedenen Krankheiten leiden, könnten die Symptome erfasst und der jeweiligen Krankheit zugeordnet werden. Für einen neuen Patienten kann dann anhand seiner Symptome festgestellt werden, an welcher der Krankheiten im System er mit der größten Wahrscheinlichkeit leidet. Schwierig ist hier allerdings, dass einige Symptome möglicherweise abhängig voneinander sind oder mit anderen Krankheiten zusammenhängen können.

Das einmal trainierte, sollte eigentlich persistent abgespeichert werden. Beschreiben Sie kurz wie Sie das für dieses Programm machen würden.

Die Instanzvariablen fc und cc der Classifier-Instanz müssen serialisiert werden, beispielsweise im JSON-Format. Die resultierende, tabellarische Struktur kann in ein File abgespeichert und bei Gebrauch geladen werden.

Beispiel aus unseren Unittests:

Trainiert wurden folgende Sätze:

```
"nobody owns the water" - "Good"
"the quick rabbit jumps fences" - "Good"
"buy pharmaceuticals now" - "Bad"
"make quick money at the online casino" - "Bad"
"the quick brown fox jumps" - "Good"
"next meeting is at night" - "Good"
"meeting with your superstar" - "Bad"
"money like water" - "Bad"
```

Daraus resultieren fc und cc in bereits serialisierter Form:

```
{
'fc':
  {
    'brown': {'Good': 1},
    'casino': {'Bad': 1},
    'your': {'Bad': 1},
    'meeting': {'Bad': 1, 'Good': 1},
    'night': {'Good': 1},
    'with': {'Bad': 1},
    'money': {'Bad': 2},
    'jumps': {'Good': 2},
    'nobody': {'Good': 1},
    'next': {'Good': 1},
    'water': {'Bad': 1, 'Good': 1},
    'pharmaceuticals': {'Bad': 1},
```

```
16      'online': {'Bad': 1},
17      'rabbit': {'Good': 1},
18      'owns': {'Good': 1},
19      'quick': {'Bad': 1, 'Good': 2},
20      'superstar': {'Bad': 1},
21      'fences': {'Good': 1},
22      'make': {'Bad': 1},
23      'like': {'Bad': 1}
24    },
25  'cc': {'Bad': 4, 'Good': 4}
26  }
```