

Seminar Experimentielle Evaluierung im Information Retrieval

Martin Jansson
Philip Korte
Lukas Wozniak

***Aufgabenstellung, Ablauf, Probleme,
Lösungen und Ergebnisse des
Experiments***

Gliederung

- Aufgabenstellung und Bearbeitung des gegebenen Programmes
- Durchführung der Experimente
- Auswertung der Ergebnisse

Aufgabenstellung

- Ziel des Experiments: Praktische Erfahrung bekommen im verteiltem IR durch Evaluierung von IR-Anwendungen
 - Kleine Testkollektion, Indexierung und max. 9 Retrievaldurchläufe
 - Indexierung der 24 Kollektionen
 - Resource Descriptions erstellen
 - Kosten berechnen
 - Resource Selection
 - Retrieval

Überblick Vorgehensweise

1.) INDEXIERUNG DER TESTKOLLEKTION
Klasse: IndexCollection.java

2.) *Implementierung einer Unterstützung für
TFIDF*

Klasse: TextExpDT.java

3.) *Starten der Indexierung*

Das Programm

- Über 200 Klassen
- Unpräzise Aufgabenstellung (Nicht nur setzen der Parameter, sondern auch Komplettierung des Codes notwendig)
- Probleme in der Codegestaltung
- Relevante Klassen sind IndexCollection, ComputeRD, ComputeRS, ComputeCosts, TextExpDT, ExpUtils sowie PerformRetrieval

Parameter

Gruppe B (exp5) (Martin, Philip, Lukas)

dtf:

- c=1 p0=0.5 dtf bm25
- c=1 p0=0.5 dtf tfidf
- c=1 p0=1 dtf bm25
- c=1 p0=1 dtf tfidf

cori

maxdtf

- c=1 p0=0.5 dtfmax10 bm25
- c=1 p0=0.5 dtfmax10 tfidf
- c=1 p0=1 dtfmax10 bm25
- c=1 p0=1 dtfmax10 tfidf

IndexCollection.java

- Muss jede Kollektion einlesen und parsen
- Geparster Inhalt wird an PIRE Objekt übergeben
- Ruft in TextExpDT.java computeIndex() auf, wo das Stemming vollzogen wird

TextExpDT.java

- Klasse zur Stammwortreduktion der eingegebenen Wörter
- Gegebene Version beinhaltete bm25 stemming
- Musste um tfidf erweitert werden
- Es gab Probleme bei der Kommunikation mit der Pdatalog DB

```
String tf = index.convert(Index.TF_RELATION);
```

```
String maxtf = index.convert("maxtf");
```

```
rule = Parser.parseRule(maxtf + "(D,M) :- max(M,TF,{ "+tf+"(D,~,TF)  
  }).");
```

```
/* ComputeForTopic.java*/
```

```
// Parameter setzen
```

```
storage.deleteParameters(dl,"parameter('c1','_').");  
storage.deleteParameters(dl,"parameter('P0','_').");  
storage.storeParameters(dl,"parameter('c1'," + c + ").");  
storage.storeParameters(dl,"parameter('P0'," + P0 + ").");
```

```
for(int k=51; k<=150; k++){  
String topicName = k+"";  
if(k<100){topicName="0"+topicName;}
```

```
String queryID = topicName;  
WSumQuery query = Queries.getWSumQuery(queryID, "text",  
usedStemen);
```

```
// Kosten berechnen
```

```
Map costs = costEstimator.estimateCosts(dls, query);
```

```
// Resource selection
```

```
RS rs = null;  
switch(flagDTF)  
{  
case 1: rs = new MaxNumDLDTFRS(5);break;  
case 2: rs = new MaxNumDLDTFRS(10);break;  
default: rs = new DTFRS();break;  
}
```

Automatisierte Abläufe

Skriptbasierter Aufruf der Experimente

Sequenzielle Abarbeitung der Aufgaben möglich

Vorteile:

- besseres Zeitmanagement
- besserer Überblick über die Experimente (Reihenfolge)
- Einfachere Aufrufe (Benutzerfreundlicher)

Nachteile:

- niedrige Fehlertoleranz
- hoher Aufwand bei Skripterstellung (und Codeanpassung)

Probleme (Java)

- Gescheiterte Indexierung
 - Tokenizer vs Split
 - Dokumentengewichte nicht vorhanden
- Parameter zeigen keinen Einfluss auf die Ergebnisse
- CORI und TFIDF Implementierung fehlgeschlagen

Probleme (UNIX)

- starten der Experimente
 - anmelden des Benutzers im Pool
 - verschieben der Läufe in den Hintergrund
 - keine Möglichkeit der Beobachtung (Jobs)
- Probleme beim Zugriff auf die Ergebnisse
 - Dateien
 - SQL

ap88_8c – Topic :133

c=1.0 - p0=1.0 - DTF - bm25

133 0 AP881222-0035 1 1.0 1
133 0 AP881226-0158 1 0.712738529534067 1
133 0 AP881226-0161 1 0.696711627628299 1
133 0 AP881221-0093 1 0.436279409317885 1
133 0 AP881221-0038 1 0.432167900149834 1
133 0 AP881219-0186 1 0.403647276118491 1
133 0 AP881220-0015 1 0.403505094566829 1
133 0 AP881225-0026 1 0.403220565920238 1
133 0 AP881221-0108 1 0.389640398253184 1
133 0 AP881220-0121 1 0.385520474926249 1
133 0 AP881221-0048 1 0.367097165258073 1
133 0 AP881221-0185 1 0.364466363824877 1
133 0 AP881222-0061 1 0.356795441403489 1

ap88_8c – Topic :133

c=1.0 - p0=0.5 - DTF - bm25

133 0 AP881222-0035 1 1.0 1
133 0 AP881226-0158 1 0.712738529534067 1
133 0 AP881226-0161 1 0.696711627628299 1
133 0 AP881221-0093 1 0.436279409317885 1
133 0 AP881221-0038 1 0.432167900149834 1
133 0 AP881219-0186 1 0.403647276118491 1
133 0 AP881220-0015 1 0.403505094566829 1
133 0 AP881225-0026 1 0.403220565920238 1
133 0 AP881221-0108 1 0.389640398253184 1
133 0 AP881220-0121 1 0.385520474926249 1
133 0 AP881221-0048 1 0.367097165258073 1
133 0 AP881221-0185 1 0.364466363824877 1
133 0 AP881222-0061 1 0.356795441403489 1

Vielen Dank für die Aufmerksamkeit