Machine Learning and Information Retrieval

The Combination of Expert Advice Problem

Jay Aslam

College of Computer and Information Science
Northeastern University
Combination of Expert Advice

- **Machine Learning**
  - *Problems*:
    - medical diagnostics, stock market prediction, etc.
  - *Solutions*:
    - boosting, bagging, weighted majority, etc.
Combination of Expert Advice

- Information Retrieval
  - Problems:
    - filtering (news/spam), recommendation systems, metasearch, etc.
  - Solutions:
    - Linear classifiers, collaborative filtering, on-line learning, etc.
The Metasearch Problem

Search for: chili peppers
Search and Metasearch

- Search engines:
  - Provide a ranked list of documents.
  - May provide relevance scores.
- Metasearch engines:
  - Query multiple search engines.
  - May or may not combine results.
Search engine: **Looksmart** found 117 results.
The query string sent was `-chili -peppers`

1. The Red Hot Chili Peppers
   Find photos, lyrics, updates, tour info, and news on alternative–funk–rock band the Red Hot Chili Peppers.
   Looksmart category – Red Hot Chili Pepper

2. Red Hot Chili Peppers Audio and Video
   Watch videos and listen to music by this rock/funk band.
   Looksmart category – Red Hot Chili Peppers

3. Chili and Hot Sauces
   Shop for mouth–burning chili sauces, Tabasco, hot salsas and other pepper–inspired sauces.
   Looksmart category – Chili & Hot Sauces

4. Chili and Hot Sauces
   Find chili and other hot sauce recipes, including salsas, dips, spices, and rubs, and visit the Pepper Fool.
   Looksmart category – Chili & Hot Sauces

5. Red Hot Chili Peppers – Screens and Themes
   Promotional screensaver for the funk–rock band features falling chili peppers.
   LookSmart category – Red Hot Chili Peppers Multimedia

Search engine: **GoTo.com** found 10 or more results.
Metasearch: Metacrawler
Metasearch: Profusion
Metasearch Algorithms

- Heuristics and hacks:
  - Interleave, average rank, sum scores, etc.

- Principled models:
  - Bayesian inference, election theory, etc.
  - On-line combination of expert advice.
On-line Metasearch [Aslam, Pavlu, Savell]
On-line Allocation Problem

- Investment problem:
  - Monthly paycheck…
  - Many stock brokers & mutual funds…
  - *How to invest?*
  - *How to incorporate feedback?*
On-line Allocation: The Hedge Algorithm

Start with uniform distribution \( D \)

- Choose Allocation
- Receive loss vector
- Update distribution

\[ \beta \in [0,1]; N \text{ strategies (systems)} \]

initial weights \( w^l \in [0,1]^N; \sum_{i=1}^{N} w^i = 1 \)

\[ p^t_i = \frac{w^t_i}{\sum_{i=1}^{N} w^t_i} \]

\[ l^t \in [0,1]^N \]

\[ L_H^t = p^t \cdot l^t \]

\[ w^t_{i+1} = w^t_i \cdot \beta^l_i \]

\[ L_{HEDGE} = \sum_{t=1}^{T} p^t \cdot l^t \]
Why Hedge? [Schapire, Freund]

\[ L_{HEDGE} \leq \frac{\ln \left( \frac{1}{\beta} \right) L_{BEST\ SYSTEM} + \ln N}{1 - \beta} \]
On-line Setup

Start with uniform distribution $D$

Choose Allocation

Receive loss vector

Update distribution

Hedge

FEEDBACK

SYSTEM WEIGHTS

MODIFY WEIGHTS
A Unified Model

Hedge

DOCUMENT SELECTOR

JUDGE

SYSTEM WEIGHTS

MODIFY WEIGHTS

RANK-VALUE MAP
Implementation

- map ranks to values

\[
value(r) = \frac{1}{r} + \frac{1}{r+1} + \ldots + \frac{1}{Z} \approx \ln \frac{Z}{r}
\]

\[
\text{RELEVANT} = -1 \\
\text{NONRELEVANT} = +1
\]

\[
LOSS(d, s) = \text{label}(d) \cdot value(\text{rank}_{d,s}) \approx \text{label}(d) \cdot \ln \frac{Z}{r}
\]

total loss vs. total precision vs. average precision
Experiments

- TREC 3, 5, 6, 7, 8
  - 41-129 systems
  - 50 queries per TREC
  - Metasearch combines all systems
- Use TREC judgments as user feedback
Metasearch – TREC8

Metasearch performance

Hedge

TREC8

MAP

# docs judged

0.55

0.5

0.45

0.4

0.35

0.3

0.2

0.1

0

0

50

100

150

200

250

300

350

400

450

500

Hedge

CombMNZ

best system
Metasearch – TREC 3
Metasearch – TREC 5

![Graph showing Metasearch Performance, Hedge, and TREC5 over # docs judged.](image)

- **Hedge**
- **CombMNZ**
- **best system**
Metasearch – TREC 6

![Graph showing Metasearch Performance, Hedge, and TREC6 with MAP on the y-axis and # docs judged on the x-axis. The graph compares different systems, including Hedge, CombMNZ, and the best system.]
Metasearch – TREC 7

![Graph showing metasearch performance across different datasets and metrics. The graph plots MAP (Mean Average Precision) against the number of documents judged. The performance is compared between Hedge, CombMNZ, and the best system. The graph illustrates the improved performance of metasearch as the number of documents judged increases.]
Metasearch – TREC 3,5,6,7
- **average_value** at episode t
  - “trust” in systems change with t
- **metasearch** : include already judged docs
- **pooling**
  - system evaluation
- get feedback

\[
average\_value_t(d) = \sum_{s=1}^{N} w_{s}^{t-1} \cdot value(rank_{d,s})
\]

\[
d_{t} = \arg\max_{d} \left( average\_value(d) \right)
\]

\[
LOSS\left(d_{t}, s\right) = label(d_{t}) \cdot value(rank_{d_{t}, s})
\]

\[
w_{s}^{t+1} = w_{s}^{t} \cdot \beta^{LOSS_{t}(d_{t}, s)}
\]