Query Log Analysis for User-Centric Multimedia Databases

Lyes LIMAM

llimam@liris.cnrs.fr

Passau – 16/12/08

Overview

Media user centric applications

User behavior modeling

- **Use of taxonomies for multimedia characteristics**
- **Query classification**

Conclusion

Future directions

Media user centric Applications

Some examples :
- **User-centric media services in the extended home (mobile TV, remote services…)**
- **User communities systems and platforms (P2P, Visioconference…)**
- **Multimedia information retrieval**

Need to understand the user behavior

Different points of view about the user behavior
- **Every action performed in the system**
- **User profile (mostly static)**
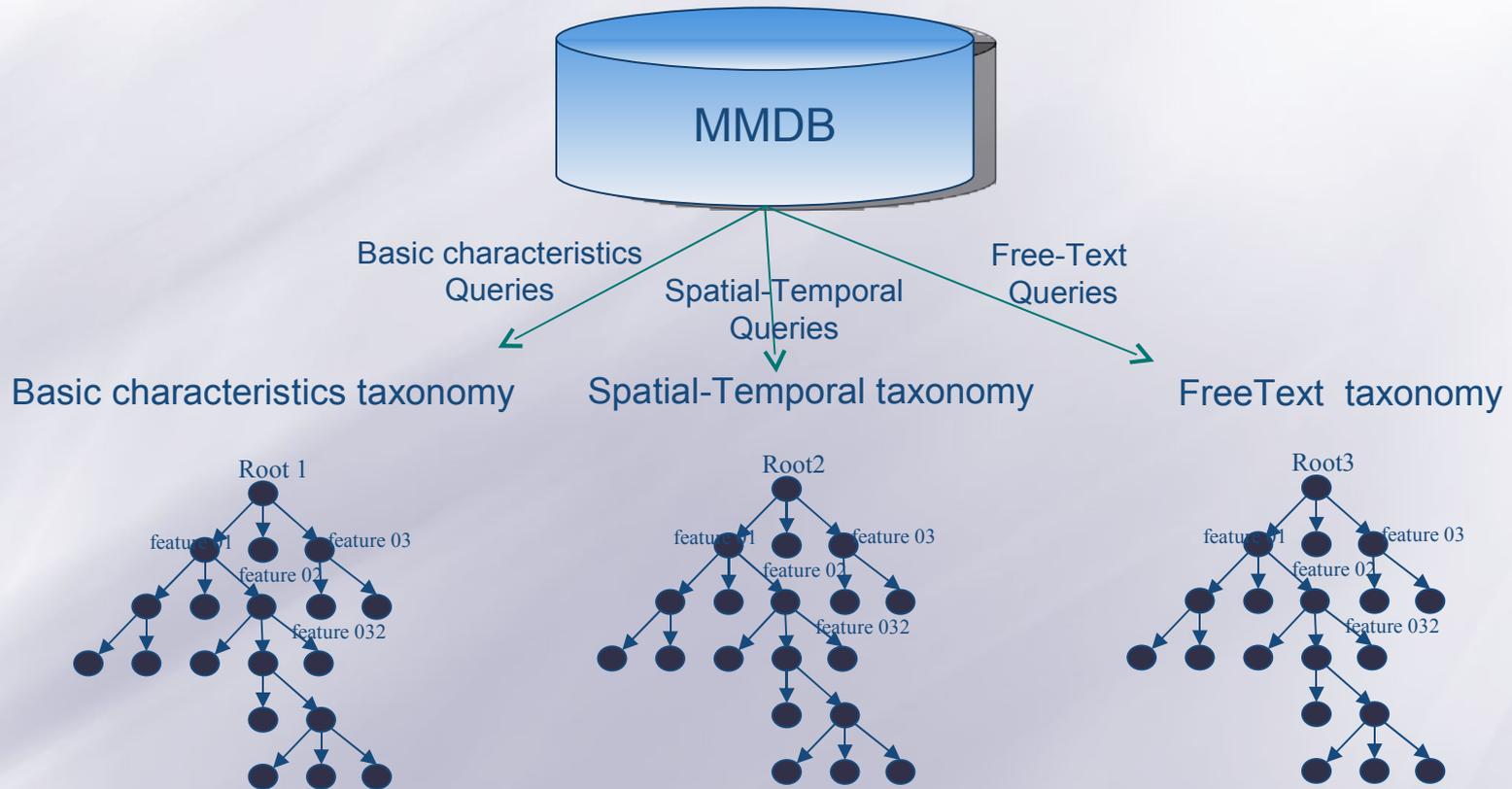- **Set of queries (input and output)**

LIRIS

User Behavior modeling

- The way that a user is interested in a multimedia content

- Query logs are used as means to extract patterns about the user behavior
  - According to the query types (categories)
- Use of taxonomies (according to categories)

Query Types

- **The requested characteristics change according to the type**

- **Different low and high level query types are considered (they follow the MPEG query format standard)**
  - ●**Basic characteristics (color, texture, shape, technical information)**
  - ●**Advanced characteristics (semantic descriptions)**
  - ●**Spatial-Temporal**
  - ●**Free Text**
  - ●**Combinations of format categories**

LIRIS

Taxonomy

The characteristics in a MMDB are organized hierarchically according to the classification schemes of MPEG-7

Root feature 0

feature 01

feature 03

feature 02

feature 032

MPEG7 Schema

Taxonomy

Multimedia DataBase → Query Type → **Taxonomy**

MMDB

Basic characteristics Queries

Spatial-Temporal Queries

Free-Text Queries

Basic characteristics taxonomy

Spatial-Temporal taxonomy

FreeText taxonomy

Root 1

feature 01   feature 03
feature 02
feature 032

Root2

feature 01   feature 03
feature 02
feature 032

Root3

feature 01   feature 03
feature 02
feature 032

Query classification basics

- **We have an observation window of a user U1 with N1 queries of type T1, N2 queries of type T2**

- **For each query type T we perform a clustering according to the concrete characteristics, that means queries having similar characteristics are grouped together**

Query Classification

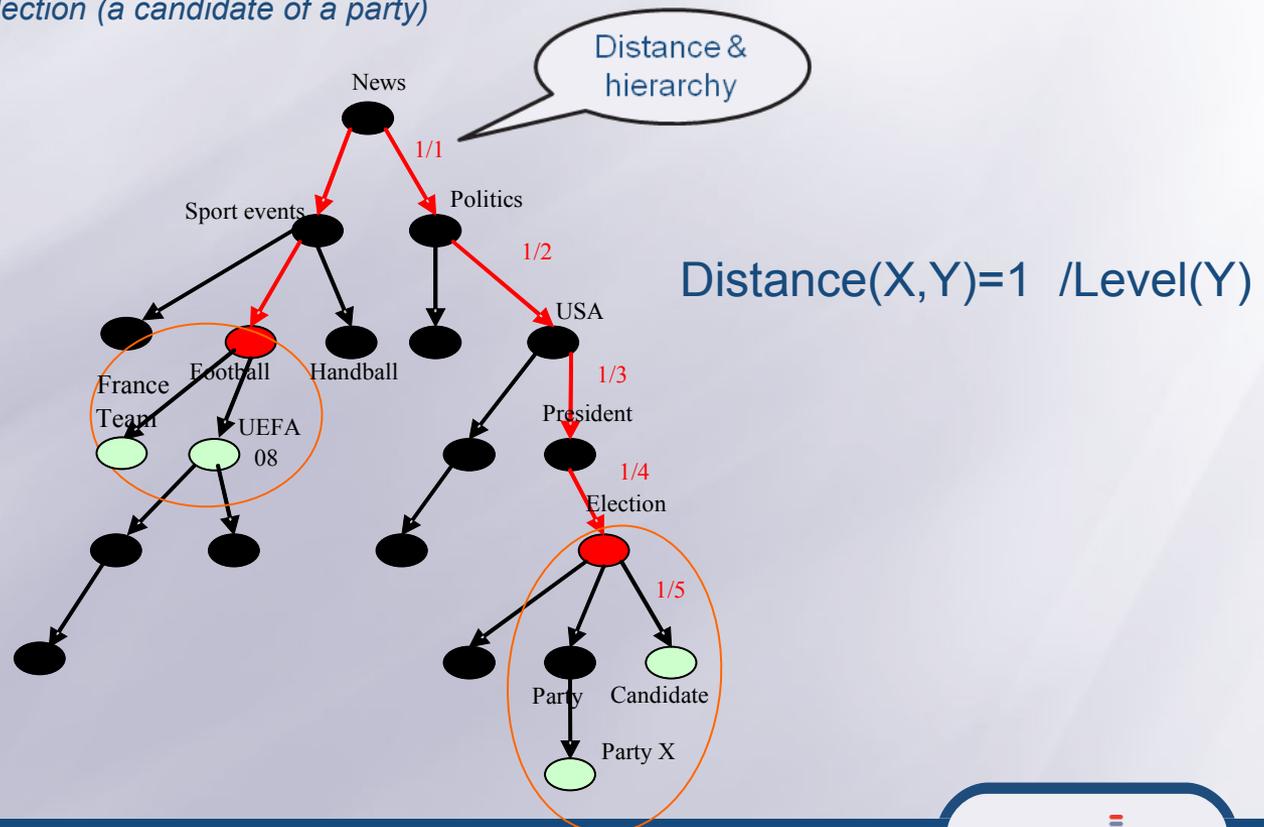**Simplistic Algorithm for query classification (e.g. Free-text queries)**

- *Q1 : France team participation in UEFA08 football competition*
- *Q2 : USA election (a candidate of a party)*

Q1 and Q2 are two queries to be compared
- Ts (threshold) = 3

Taxonomy of Multimedia features

News

Sport events

Politics

The path between sQ1 and sQ2: Dist <=? Threshold

Root of sQ1

USA

Football   Handball

France Team   UEFA 08

President

sQ1: Sub tree of the query Q1

Root of the query sQ2

A feature of the query Q1

Election

Party   Candidate

sQ2: Sub tree of the query Q2

Party X

A feature of the query Q2

Simplistic Algorithm for query classification (e.g. Free-text queries)

- *Q1 : France team participation in UEFA08 football competition*
- *Q2 : USA election (a candidate of a party)*

Distance & hierarchy
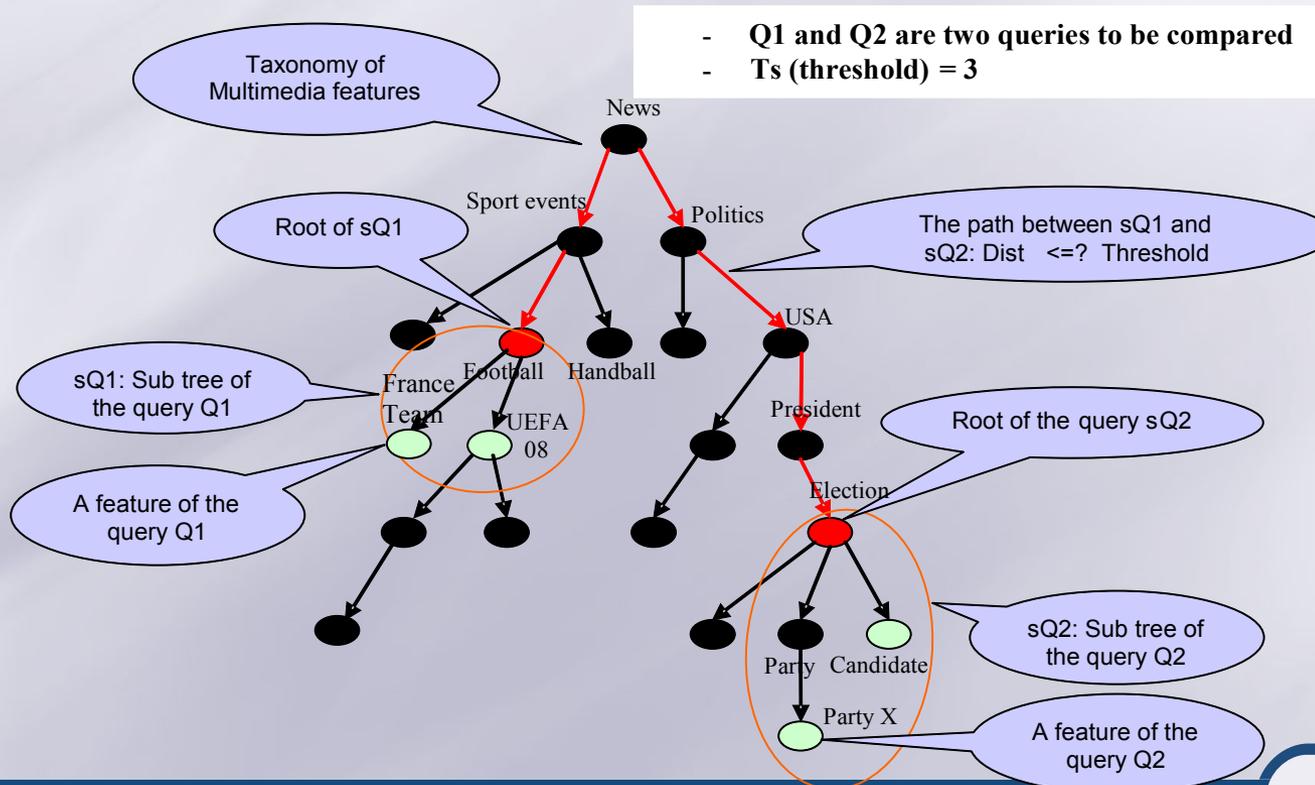
$$Distance(X,Y)=1\ /Level(Y)$$

Query Classification (simplistic algorithm)

```
Queries classification algorithm ------------------------------------------------------------
T                        // Taxonomy
S = {sQ0,  sQ1, ...}      // set of internal structures
SQj= {e0, e1...}        // set of characteristics => one internal structure=> one sub-tree of a query Qj
C = {∅}              // set of behavior classes
ci = {∅}              // ci ∈C
Pmin              // function for shortest path between two internal structure
Ts = Value          // ThresholdPmin
I= 0, K ≠ J


For each sQJ ∈S  Do
     If chekTreeNode (sQJ) = false  then          // find the root node for each sub-tree
       rootSQJ = Findroot (sQj)
End
While S  not empty   Do
    J = 0
    ci = { SQK }                          // initialize ci :  select a random sub-tree
    For each SQJ ∈S  Do
        Dist = Pmin (rootSQJ , rootSQK)
        If  Dist ≤ Ts then                  // try to find an acceptable distance
          ci = ci ∪{sQj }
          S = S / {sQj}
        EndIf
        J = J+1
        S = S / {SQK}
    End
End
 I = I+1
C = C∪{ci}                          // construct the set of classes
```

Graph theory :

- Relative neighborhood graphs
- Minimum Spanning Tree
- Graph coloring

Distance & hierarchy

Threshold graph
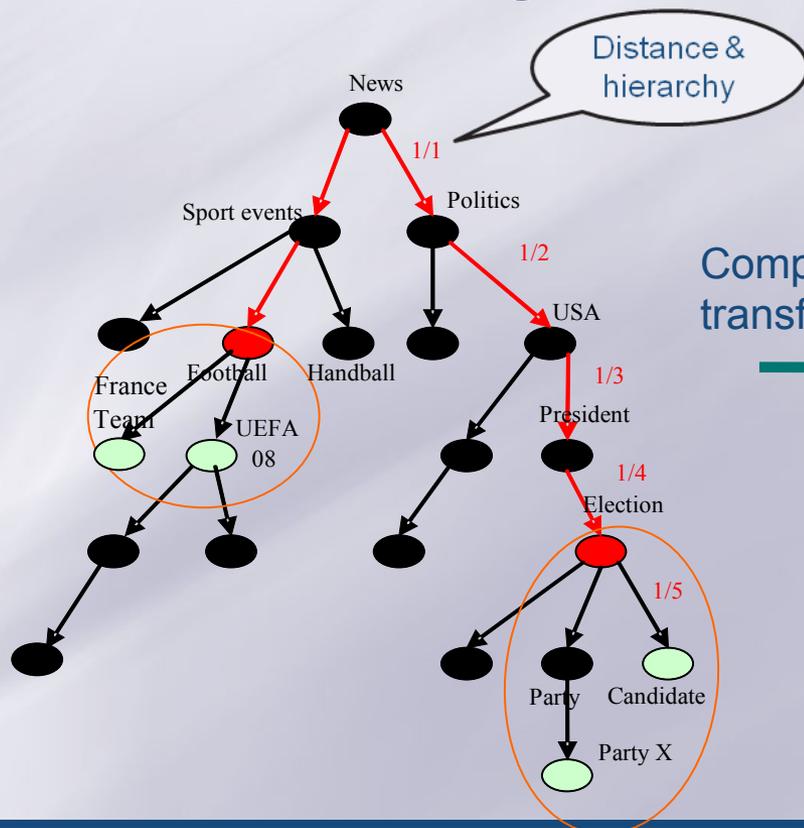
Complete graph transformation

Sum of distances
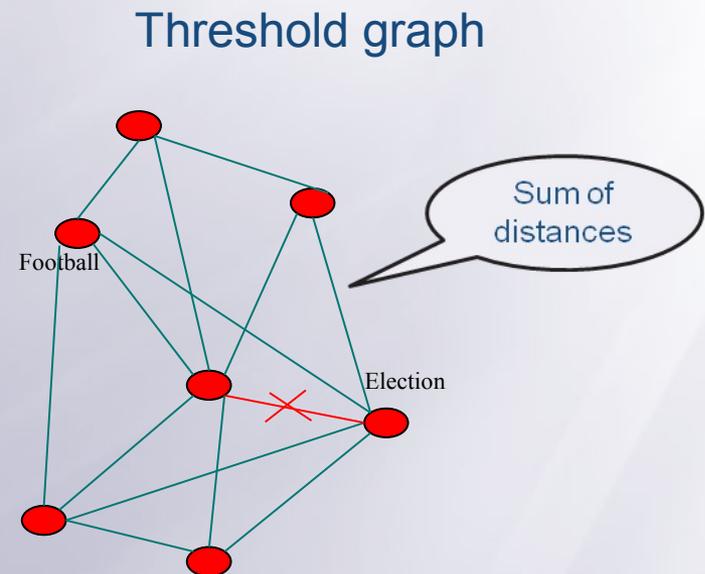
Users Grouping

- **A user is considered to have a set of behaviors each behavior is an interpretation of a cluster obtained by the former algorithm**
- **Comparing the set of behaviors forms groups of users**

Resulting clusters for one query type

Similarity with respect to behaviors

Behaviors	User 1	User 2	User 3	
{a1, a2,…} {b1,b2,..}	1			
{c1, c2,…} {d1, d2..}		1		
{e1,e2,…}			1	
{}				

Users grouping similarity

- **Every user is represented by N (query types) centers and variances. Each center is a node in the taxonomy and computed as the center of all clusters for this query type. The variance expresses the dispersion of the clusters**

- **The users are compared on the base of their centers**

- **remark : only centers of the same query type can be compared**

LIRIS

User grouping conclusions

- **Not all of the users are comparable with respect with their behaviors**
  - **E.g. users requesting query types which do not overlap**
- **For those who are comparable we have computed**
  - **Single similarity value**
  - **Overlap in query types and characteristics**
- **Propose recommendations based on the behavior of the similar users**
  - **E.g. recommend an alternative feature (color layout) from similar users (popular features)**
- **Install data filtering systems for new data**

LIRIS

Conclusion

- **Position paper presenting a new idea on media user centric MMDB**
- **User behaviour deriving based on queries**
- **MMBD + query log +Taxonomy→set of behaviours (one user) → groups of users**
- **Experimental evaluation will follow up in 2009**

Future directions

- **Deal with complex query type (combined)**
- **Case study (real logs)**
- **Compare more clustering algorithms**
- **Integrate contextual information**
- **…**

LIRIS