Integrating and Ranking Interests From User Profiles

Fabien Duchateau, Lynda Hardman
CWI, The Netherlands

CWI
Centrum Wiskunde & Informatica

ESWC / LUPAS 2010, Greece

Outline

Introduction

Many websites store a profile of their users.

- Lots of scattered profiles, even for the same user
- Profiles from different websites are seldom compatible
- Service providers use these profiles for recommendations, improved search results, etc.

Interoperability among these profiles would benefit both users and service providers

Related Work

Different approaches have been proposed:

- Representing user profiles: *FOAF*, *UserRDF*, and *GUMO*
- Aggregating or linking Web profiles such as *Mypes*[1], *Google Social Graph API*[2], OpenID[3] => redundancies in the aggregated tag cloud or implies links between public profiles
- Integration of user profiles for domains such as human resources [VDM03] or education [SCCA06] => too specific approaches

Yet, many Web applications still include their own user models

[1] http://mypes.groupme.org/mypes/
[2] http://code.google.com/apis/socialgraph/
[3] http://openid.net/

Running Example

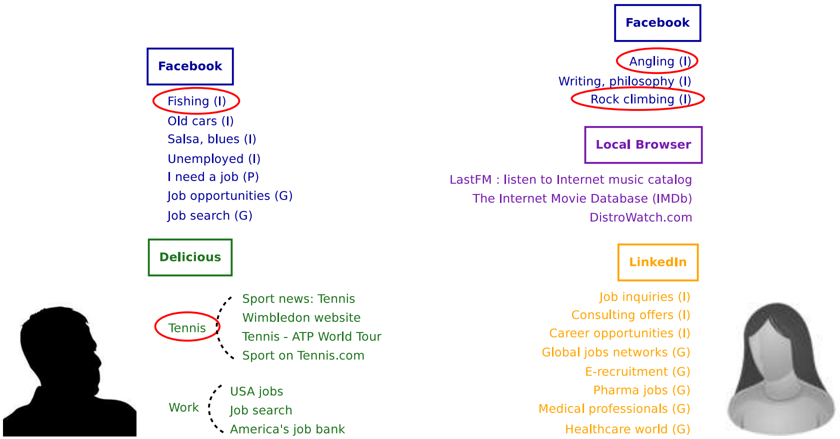

Figure: Running example with two users and their profiles

Motivations

For users:

- Integrate a profile at a higher level of abstraction for converting profiles from one model to another *(tennis and rock climbing abstracted as sport)*
- Use information already stored in their various profiles to automatically fill in empty profile based on existing ones

For service providers:

- Analysing user profiles for extracting the most relevant information to exploit (recommendations)
- Comparing different user profiles to deduce common user interests and propose related events/activities *(fishing and angling)*

Contributions

We propose an approach that:

- integrates two profiles (same or different users) by clustering their interests around the same higher-level concept
- ranks each cluster according to its importance in user profiles

Benefits:

- aggregate common user interests at different levels (low and high abstraction levels)
- extract relevant interests in large profiles or provide a summary

1. Introduction
   - Context
   - Motivations
   - Contributions

2. Our Approach
   - Overview
   - Integration
   - Ranking

3. Conclusion

Overview of our Approach (1/2)

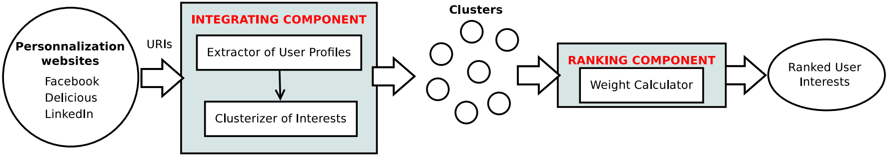

Figure: A two-step approach

Overview of our Approach (2/2)

Integrating

The idea is to create clusters of similar interests under the same (high-level) concept. To discover these concepts, we use matching techniques (terminological and linguistic).

Ranking

After the clustering, we compute the weight of each concept w.r.t. user interests.

Integration (1/6)

Before integrating and discovering high-level concepts, we need to prepare the data:

- Extracting interests from each user profile (APIs)
- Apply several techniques for cleaning the data (e.g., tokenization)

Example

medical professional => *medical, profession*

Integration (2/6)

The next step deals with matching. We match all interests from one profile to all interests from another profile.
Which matching techniques ?

- structural
- constraint-based
- linguistic
- terminological

Integration (2/6)

The next step deals with matching. We match all interests from one profile to all interests from another profile.
Which matching techniques ?

- ~~structural~~ (no structure in the profiles)
- constraint-based
- linguistic
- terminological

Integration (2/6)

The next step deals with matching. We match all interests from one profile to all interests from another profile.
Which matching techniques ?

- ~~structural~~ (no structure in the profiles)
- ~~constraint-based~~ (no constraints in the profiles)
- linguistic
- terminological

Integration (2/6)

The next step deals with matching. We match all interests from one profile to all interests from another profile.
Which matching techniques ?

- ~~structural~~ (no structure in the profiles)
- ~~constraint-based~~ (no constraints in the profiles)
- linguistic => Wordnet dictionary for its reliability in terms of quality
- terminological

Integration (2/6)

The next step deals with matching. We match all interests from one profile to all interests from another profile.

Which matching techniques ?

- ~~structural~~ (no structure in the profiles)
- ~~constraint-based~~ (no constraints in the profiles)
- linguistic => Wordnet dictionary for its reliability in terms of quality
- terminological

Integration (2/6)

The next step deals with matching. We match all interests from one profile to all interests from another profile.
Which matching techniques ?

- ~~structural~~ (no structure in the profiles)
- ~~constraint-based~~ (no constraints in the profiles)
- linguistic => Wordnet dictionary for its reliability in terms of quality
- terminological => COMA++ matching tool for its library of 17 terminological measures

Integration (3/6)

Linguistic matching:

Detecting the closest common higher-level concept between two interests based on the Wordnet dictionary[4]

- A distance is computed in terms of intermediary (Wordnet) concepts between both interests
- The search for the common concept is limited to 7 upper levels

Example

rock climbing and *tennis* => linked by the Wordnet *sport* concept
tennis [has parent] *court game* [has parent] *athletic game* [has parent] *sport* [has child] *rock climbing* (distance = 4)

[4] http://wordnet.princeton.edu/

Integration (4/6)

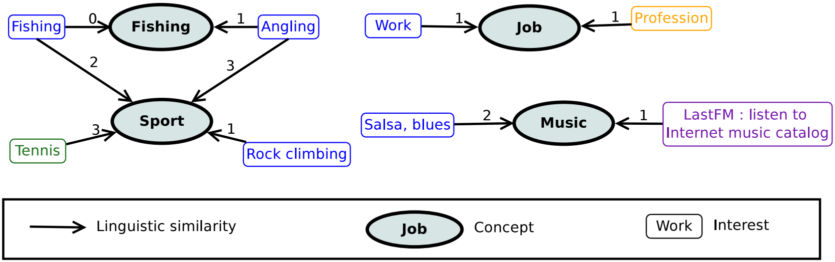

Figure: Interests [work] Linked to the Concepts [job] using Linguistic Measures [—>]

Integration (5/6)

Terminological matching:

Many interests are not matched because no Wordnet concept links them. Thus, we use COMA++ [ADMR05] to discover similarities between an interest and a concept based on their labels.
COMA++ includes a library of terminological measures and is reputed to provide acceptable quality.

Examples

job search and *job* => terminological similarity = 0.42
salsa, *blues* and *sport* => terminological similarity = 0

Integration (6/6)

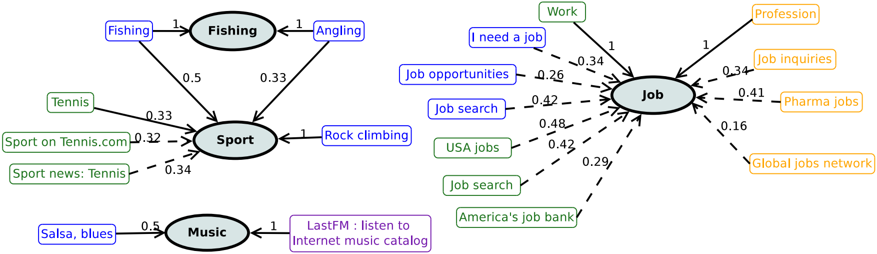

Figure: Interests [work] Linked to the Concepts [job] using Linguistic [—>] and Terminological [- ->] Measures

Ranking (1/2)

After identifying the clusters, we propose a ranking for clusters (concepts) according to their weight.

- User profiles may contain hundreds of interests (including pages and groups)
- Need for distinguishing strong interests from occasional ones

How do we rank ?

Compute a score for each cluster based on the (normalized) similarity values of the interests linked to it

Ranking (2/2)

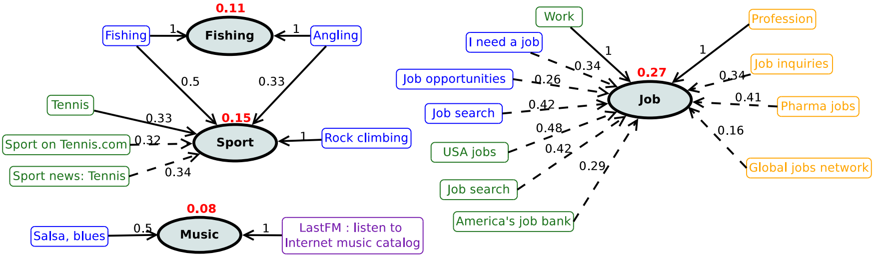

Figure: Ranked clusters (concepts) from our running example

Conclusion

We have presented a new method for:

- Integrating different profiles by clustering similar interests
- Extracting the most shared interests from profiles

Future Work

- We need more experiments on real datasets (Petamedia project)
- Relying on other resources for linguistic matching (e.g., DBpedia)
- User behaviours (frequent keyword search, frequency of visited websites)

David Aumueller, Hong Hai Do, Sabine Massmann, and Erhard Rahm.
Schema and ontology matching with COMA++.
In ACM SIGMOD, pages 906–908, 2005.

Craig Stewart, Alexandra Cristea, Ilknur Celik, and Helen Ashman.
Interoperability between AEH user models.
In APS '06: Proceedings of the joint international workshop on Adaptivity, personalization & the semantic web, pages 21–30, New York, NY, USA, 2006. ACM.

B. Vandermeulen, Joost R. Duflou, and Bart De Moor.
The role of user profiles in vector-based information retrieval.
In IKE, pages 668–669, 2003.