Exploiting Hierarchical Tags for Context-awareness

Conference Paper · January 2010

DOI: 10.1145/1871962.1871984

CITATION
1

READS
34

4 authors, including:

Jérôme Kunegis
University of Namur
72 PUBLICATIONS **1,651** CITATIONS
SEE PROFILE

Ernesto William De Luca
Fachhochschule Potsdam
127 PUBLICATIONS **807** CITATIONS
SEE PROFILE

Sahin Albayrak
Distributed Artificial Intelligence Laboratory
405 PUBLICATIONS **3,722** CITATIONS
SEE PROFILE

Exploiting Hierarchical Tags for Context-awareness

Alan Said
TU-Berlin / DAI LAb
alan.said@dai-lab.de

Jérôme Kunegis
University of Koblenz-Landau
kunegis@uni-koblenz.de

Ernesto W. De Luca
TU-Berlin / DAI LAb
ernesto.deluca@dai-lab.de

Sahin Albayrak
TU-Berlin / DAI LAb
sahin.albayrak@dai-labor.de

ABSTRACT

Context-awareness for information retrieval is challenging problem as information about the users current situation is rarely available. If such information would be available, retrieval and recommendation systems could find information relevant to their users' current context. We present a model for creating contextualized recommendations based on implicitly expressed context via keywords assigned to previously seen items. When employed, the model can improve the performance of recommender systems.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Retrieval models

General Terms

Algorithms, Experimentation, Human Factors

Keywords

Context modeling, user modeling, context-awareness, tags

1. INTRODUCTION

The vast majority of information retrieval systems either disregard or care very little about the user's and the item's context. Whether it comes to search, recommendation or classification, most of today's online portals (Google, Netflix, Amazon) operate semi context-ignorantly. If employing contextual information, they use explicit features such as location, language or device for contextualization. Information about the user's current situation, i.e. mood, company, amount of time to be spent, etc. is not used. The reason for this is most likely not an ignorant attitude, rather the lack of such contextual information about the user.

Whenever we face a personalized system which aims at helping the user find relevant information, context-awareness could be exploited to find relevant data – **if** information about the context would be available.

Explicitly asking the user for the current context implies that the service provider defines a set of contexts, this brings a risk of either leaving out relevant contexts, or actually changing the context simply by inquiring it. Furthermore, users might be unwilling to reveal their actual context and give incorrect answers. Instead, we propose a way of using a combination of hierarchical, historical and current data for implicit, non-intrusive, context identification in order to heighten the users' experience through context-awareness. The main contributions of our paper are:

- overview of how contextual tag data can be stored.
- description of how said data can be used.

ESAIR'10, October 30, 2010, Toronto, Ontario, Canada.
ACM 978-1-4503-0372-9/10/10.

2. CONTEXT-AWARENESS

Context-awareness is currently a popular topic in the recommender systems community, though due to the lack of contextual data, most approaches are either static or intrusive (or a combination of both). Static approaches are based on predefined or scheduled features, i.e. time of day, day of week, sex, genre, etc. Intrusive ones ask the user to explicitly state the current context, thus introducing a possibly context-altering step in the process as mentioned previously. Intrusive ones could be likened to ad hoc static approaches, where the static feature is requested before recommending.

2.1 Dynamic and Non-intrusive Approach

Our model attempts to implicitly detect the context by utilizing a mix of contextual and other structured features in a dataset obtained from the movie recommendation community Moviepilot[1]. At Moviepilot, movies can be assigned keyword from five hierarchical categories (in addition to the normal genre, etc. assignments found in most other movie-related datasets). These types of keywords contain the *mood*, *intended audience*, *plot*, *time* and *place* of the movie. Each of these categories is structured as a tree where the root node is the category and subsequently all other keywords in the category are its children, or the children's children, etc.

Each movie can be tagged with any number of tags from each category. Tags are assigned to movies by moderators, making the data very clean, i.e. without noise. Table 1 shows some of the most popular tags[2] and the number of assignments for each of the categories. Even though the numbers might seem low, ratings of tagged movies make up 5.9 million of the 6.2 million ratings in the dataset (>95%), thus the vast majority of popular movies have tags assigned to them. An even stronger indication of this is seen in

[1] `http://www.moviepilot.com`
[2] Translated from German

Mood	TAS	Audience	TAS	Plot	TAS	Place	TAS	Time	TAS
Exciting	1252	Families	1405	Murder	849	USA	468	21st century	458
Funny	1095	Guys	759	Love	643	New York	366	present	412
Touching	1013	Girls	307	Friendship	495	Prison	306	1990's	304
Thrilling	876	18+	263	In Love	476	Paris	254	1980's	199
Serious	631	Teens	232	Father-son relationship	432	London	254	WWII	184

Table 1: The five most popular keywords in every category and the number of tag assignments (TAS) each.

Table 2, where the number of ratings in each keyword category is shown.
By using combinations of these tag assignments in conjunction with historical rating data and external information such as current events, TV or cinema listings, it becomes possible to create personalized context models. These context models can be inferred from time-tag-external information dependencies, where time can be split into sections corresponding to a certain tag or mood or external feature. For instance, if a user consistently rates movies of a certain tag at certain points in time or events (i.e. Saturday night, June, the Oscars', etc.) we can assume the user is in a corresponding context.

Consider this very primitive example; A user regularly rates movies tagged with the *Guys* keyword from the Intended Audience category combined with the *Exciting* or *Thrilling* keywords from the Mood category. The context can be derived and applied to the recommendation process. The same user however, has reoccurring sessions where he rates movies labeled *Families* and *Funny* respectively. In this case, a probable assumption about a context change can be made and the current context can in turn be applied to the recommendation process.

Keyword type	Ratings	% of total
Mood	4,525,169	72.9
Intended Audience	2,343,939	37.8
Plot	5,887,824	94.8
Place	5,165,311	83.2
Time	4,595,547	74.0
Total_{tagged}	5,910,444	95.2
Total_{all}	6,208,990	100.0

Table 2: The number of ratings and percentage of total in each of the keyword categories.

The example is a naive version of the real process, dependencies may not be as trivial as described above. Feature dependencies have to be analyzed and context models defined. However, all essential information is contained in the data, thus models for implicit context definitions based on selected features (including the keywords, not excluding other features) can be developed.
Having done the context assumptions, including them into a recommendation process can be accomplished by many hybrid recommendation algorithm capable of higher order data, the exception here will be the dynamic feature set. A more detailed description of the processes involved can be found in [4].

3. RELATED WORK

The context-aware field of research was first discussed by Want et al. in 1992, the authors described a location-aware phone forwarding system [5]. The amount of work in the field has been steadily growing ever since. In their survey Baldauf et al. cover the general context-aware topic extensively and present core aspects of all abstraction levels in context-aware systems [2].
In recommender systems, context-awareness is becoming more and more popular. In their state of the art survey, Adomavicius and Tuzhilin present different approaches on how hybrid and contextual information has been used [1]. Cantador and Castells use a semantic runtime context through domain ontologies for recommending news articles, their approach outperforms similar context-unaware personalized approaches [3].

4. CONCLUSION AND FUTURE WORK

Adding context-awareness to recommender systems has been shown to improve their performance. We are convinced that context-aware approaches have great benefits compared to their simpler counterparts. Currently, we are working on context and behavior modeling for recommender systems and preliminary results are positive. Current state of the art research in the field of context-awareness mostly deals with static context-awareness, i.e. where context is predefined. Our approaches strive to create ad hoc context-awareness based on user interaction, behavior and historical (user and movie) data.
Given the structure of our data, we are optimistic to what can be achieved through our approach.

5. REFERENCES

[1] Gediminas Adomavicius and Alexander Tuzhilin, 'Toward the next generation of recommender systems: A survey of the State-of-the-Art and possible extensions', *IEEE Transactions on Knowledge and Data Engineering*, **17**(6), 734–749, (2005).

[2] Matthias Baldauf, Schahram Dustdar, and Florian Rosenberg, 'A survey on context-aware systems', *Int. J. Ad Hoc Ubiquitous Comput.*, **2**(4), 263–277, (2007).

[3] Iván Cantador and Pablo Castells, 'Semantic contextualisation in a news recommender system', in *CARS Workshop*, (2009).

[4] Alan Said, 'Identifying and utilizing contextual data in hybrid recommender systems', in *Proc. of RecSys 2010*, New York, NY, USA, (2010). ACM.

[5] Roy Want, Andy Hopper, Veronica Falcão, and Jonathan Gibbons, 'The active badge location system', *ACM Trans. Inf. Syst.*, **10**(1), 91–102, (1992).