

Combining Community-Based Knowledge with Association Rule Mining to Alleviate the Cold Start Problem in Context-Aware Recommender Systems

Iosif Viktoratos¹, Athanasios Tsadiras¹, Nick Bassiliades²,

¹Department of Economics, ²Department of Informatics,
Aristotle University of Thessaloniki
GR-54124 Thessaloniki, Greece
{viktorat, tsadiras, nbassili}@auth.gr

Abstract. Successful Location-Based Services should offer accurate and timely information consumption recommendations to their customers, relevant to their contextual situation. To achieve this and provide the best available recommendations to the user, researchers and developers analyse available data via exploiting data mining techniques. Unfortunately, in some cases, due to lack of available data (e.g. a relatively new member with a limited history) the above technologies and methods are not very effective. In this paper, a novel hybrid approach to alleviate the above-mentioned problem, known as cold start in context-aware recommender systems, is presented. This work aims to help researchers and developers to cope with this problem by combining a) community created knowledge, b) ontologies c) association rule mining and d) an innovative scoring function based on probability metrics.

Keywords: Context-Aware Recommender Systems, Rule-based systems, Location Based Services, Association rules, Cold start problem

1. Introduction

The past few years, Location-Based Services (LBSs) have become very popular and have played an important role in the daily activities of millions of people. Successful LBSs should offer qualitative information to every user, relevant to his/her situation (e.g. A person who is walking and searching for a coffee shop close to him/her or a person who is driving and looking for the shortest route to his/her destination). This situation, widely referred as “context”, is defined by considering information such as user profile (preferences, social state, etc.) and environment (location, time and weather) (Verbert et al., 2012). Therefore, in order to provide accurate and contextualized information to the users, researchers are focusing on collecting, studying and utilizing such kind of information. The evolution of LBSs in the past few years assists towards this direction because a huge amount of data (e.g. User profile data, Points of Interest) is available for analysis (Giurca et al., 2012).

Data mining techniques are used to analyze the available data and extract/deduce useful information so as to implement a recommender system (Burke, 2000). The general idea behind a recommender system is to propose interesting items to the customers, based on a huge amount of available data and their contextual situation (Verbert et al., 2012). Example of such systems are a) a LBS which recommends points of interest (POIs), b) an application which suggests movies or videos, or c) a website which recommends items for purchasing to the customer (García-Crespo et al., 2011). Consequently, a successful recommender system can give a huge boost to the commercial adoption of a service (e.g. The well-known success story of Amazon) and, therefore, researchers and industries are working on the evolution of such systems (Bobadilla et al., 2013).

Recommender systems, based on the techniques they use to filter the available items, can be classified into the following three main categories (Adomavicius et al., 2005; Lu et al., 2012):

a. **Collaborative filtering systems.** The best-known category of recommender systems is that of collaborative filtering systems. The basic idea of these

systems is to collect a person's available data (e.g. Ratings, preferences) and provide suggestions to him/her based on similarities with other members. Clustering techniques, rule-based techniques such as association rules, probability-based techniques, decision trees, neural networks, similarity measures and others are used for this purpose (Isinkaye et al., 2015; Portugal et al., 2018).

- b. **Content-based systems.** These systems propose items by associating the content of the items with the user's context. Techniques such as vector space model, probabilistic models such as naïve Bayes classifier, rule-based classifiers, neural networks and others are used for this purpose (Isinkaye et al., 2015).
- c. **Hybrid systems.** Hybrid systems combine some of the above techniques to get better recommendations, depending on the case.

Apart from the prior classification, recommender systems have been classified into the following categories, based on the source/type of data they use to achieve better results:

- a. **Social network-based recommender systems.** These systems take into consideration data from external sources such as social media.
- b. **Context-aware recommender systems.** The main idea behind these systems is that contextual attributes play the most important role in the recommendation process.
- c. **Demographic-based recommenders.** These systems exploit demographic attributes as the main source of data.

Unfortunately, the majority of recommender systems suffer from cold start and sparsity problems (Leung et al., 2008; Nilashi, 2017; Son, 2016). The cold start problem unfolds in two situations; the new user and the new item. The first one, arises when a person enters into a system (e.g. A system which recommends products to a consumer) and has not yet interacted with enough data objects (e.g. Rated very few products). Due to lack of data the system is not able to generate an accurate model and, thus, it cannot return satisfactory recommendations to this person (Bykau et al., 2013; Ji et al., 2015; Schein et al., 2002).

Because of these inaccurate recommendations, the user may become irritated and stop using the system (Ye, 2011). The second one, occurs when a new item is added into the system's database. Because this item has not been rated or reviewed, it cannot be included in the recommendation process (Bykau et al., 2013; Ji et al., 2015). Thus, systems should be able to be adapted as quickly as possible to the above two situations and provide useful recommendations (Lika et al., 2014; Son, 2016).

The aim of this research is to propose a novel hybrid approach (which combines association rules, probability metrics and user's context) to alleviate the cold start problem in rule-based context aware recommender systems. More specifically, this approach combines the following techniques and factors to recommend the most proper rules (rules that recommend POIs' categories) to a new user, based on his/her context:

- Community generated knowledge. Users' check-in history (when a person visits a POI) is considered.
- Association rule mining algorithms for inducing useful patterns from users' check-ins' history. This proposal uses association rule mining algorithms to extract rules from one user at a time, and not from the entire database of users' check-ins.
- Semantic web technologies such as ontologies and rules to represent users, POIs and their associations.
- A novel scoring function for ranking the rules. The proposed function combines rule relevance and popularity. Rule relevance is calculated based on probabilities, while rule popularity is based on the total of users that have this rule mined in their personal knowledge base.

In section 2, all the related works that motivated this research are presented in detail, along with our contribution. Moreover, the general idea and the design of this proposal are, also, presented in detail in section 3. Additionally, Section 4 describes all the technical and implementation details and the idea behind the development of the scoring function. Accordingly, section 5 includes a detailed evaluation of our work. Finally, the conclusions are presented in the last section of the paper, accompanied by a discussion on future directions.

2. Related work, motivation and contribution

As discussed above, the cold start problem has gained a lot of attention from the scientific community. Various methodologies have been proposed for the purpose of reducing it. In our work, these strategies have been classified in four categories (Bykau et al., 2013; Ji et al., 2015; Schein et al., 2002; Son, 2016):

- a. Solutions that apply collaborative filtering or content based techniques.
- b. Solutions that exploit external data (e.g. Social media data), or content information (e.g. Demographic data).
- c. Solutions that utilize recommendation rules.
- d. Works that require from the user to give some information to the system when he/she is logged into it for the first time.
- e. Hybrid solutions that combine the categories above.

These works are described below in sections 2.1-2.4.

2.1. Content data and collaborative filtering-based methods

Several researches utilize user and items' available attributes and apply collaborative filtering techniques. For example, Mahapatra et al. (2011) try to alleviate the cold start problem by

exploiting item attributes (in their case attributes related to wines) and collaborative filtering techniques for the purpose of recommending wines that match users' tastes. In the same spirit, Safoury et al. (2013) are taking advantage of demographic data (e.g. Age, occupation) in order to recommend movies to users that are relatively new in a system. Similarly, Lika et al. (2014) follow a three-layer method that takes into consideration collaborative filtering techniques and user demographic data. In the first two layers, they classify users into groups and select those who have similar profiles with the new user. After that, they predict the new user's ratings for the available movies and recommend the most interesting to him/her.

Raigoza et al. (2017) propose a cloud-based environment to produce a list of recommended movies. Collaborative and content-based techniques are combined with the new user's profile information to achieve this purpose.

Ke Ji et al. (2015) adopt a scalable collaborating filtering technique which exploits content information such as tags and keywords. After building a tag-keyword relation matrix, they propose a novel 3-factor matrix factorization model which integrates user interest vectors, item correlation vectors and the relation matrix.

Levi et al. (2012) implemented a system which utilizes customer reviews to overcome the cold start problem in a hotel recommender system. Semantic web technologies, text mining techniques in the customers' reviews and demographic data from their profiles are used to build user groups and hotel profiles. As soon as a new user is logged into the system, he/she is assigned to a group so as the system to be able to make useful suggestions. Along the same lines, Poirier et al. (2010) follow a two-layer approach. In the beginning, text mining techniques are applied on user reviews for the purpose of building a user-item-rating matrix. After that, collaborating filtering techniques are used to create the recommendations.

Finally, Xu et al. (2015) propose a method that predicts the new users' ratings by comparing their short rating history (less than 10 items) with the other members' ratings. When a cold start user rates an item, the system is adjusted to recommend more proper movies to him/her.

2.2. Approaches that utilize external data and social connections

Many existing works make use of external data sources such as social media in order to provide the users with more efficient solutions to the cold start problem. To start with, Castillejo et al. (2012) exploit Foursquare data to deal with the problem. They are estimating the probability of the available place categories to match user preferences by using available data such as location, social ties and other. In the same vein, Sedhain et al. (2014) take into consideration (apart from demographic data and friendships) Facebook page likes to improve the recommendations.

Tiam et al. (2017) propose an improved recommendation method, which is based on formalizing trust relationships in social networks so as to improve the performance of suggestions.

Additionally, Guy et al. (2010) propose a recommender system which considers associations among people, items, and tags. In the first stage, for building the new member's profile, the system extracts weighted lists of related people and tags. Subsequently, given this information, the system constructs a weighted list of related items and recommends the top out of the whole.

He et al. (2009) presented a social network-based recommender system which makes suggestions by taking into

account a user's preferences, items' popularity and influence from friends.

Finally, Sahebi et al. (2011) are taking advantage of social connections so as to reduce the number of users that are included in the collaborative, filtering-based recommendation process. Methods that eliminate non-relevant users can provide more accurate results.

2.3. Rule-based approaches

A common practice among researchers, to decrease the effect of the cold start, is the use of recommendation rules. Rules are IF-THEN statements that are used to represent complex relationships among the data. In such case, the IF-part (or left hand - LHS) is the antecedent of the rule, stating the context under which a recommendation can be made, and the THEN part (or right hand - RHS), is the consequence which is the actual recommendation. They can be a) mined automatically from members' history or b) constructed manually by the developers. Knowledge representation technologies are used to represent the rules (e.g. SWRL, RuleML) (Horrocks et al., 2004; Motik et al., 2012).

Regarding rule mining, association rules are frequently used to induce interesting, human understandable, patterns and help in providing useful recommendations (Pooja, 2015; Slimani et al., 2014). Many researchers have utilized them to expand persons' profiles.

For example, Shaw et al. (2010), expand persons' profiles by extracting association rules from the available data. Derived patterns and associations of objects give more information to the system so as to be able to make more accurate predictions. Leung et al. (2008) use cross-level association rules to correlate new movies with the users, regarding their attributes and their history. For example, if we take into consideration an individual who likes movies starring Al Pacino, then the system will suggest other Al Pacino movies, as well. Finally, Min et al. (2013) use granular association rules in their work. After describing users and items through granules (e.g. Comedy movies, female students), they generate association rules between them (e.g. 60% male students rated 40% drama movies that were released in 1990s).

Regarding the engineered development of rules, Ciaramella et al. (2009) use a manually constructed rule set in SWRL to determine the new user's context and proactively offer a couple of available services to him/her. Keßler (2010) implemented a RESTful service that suggests personalized surf spots at California's central coast, based on SWRL rules. Moreover, Viktoratos et al. (2015) propose a methodology which exploits POI owners' rules (user defined rules that represent their offering/marketing policy) to recommend contextualized offers to regular users, related with nearby POIs. Finally, Viktoratos et al. (2017), combine user defined rules (representing regular users' preferences/ daily patterns), POI owners' rules and social connections to recommend nearby POIs along with contextualized offers to regular users (even to a group of friends).

2.4. Approaches that require user participation

A common trend is to ask the user to fill-in some information (e.g. Extended profile data, answer questions and rate items), before using the system for the first time. Nadimi-Shahrari et al. (2014) provide a detailed survey about such kind of works, the methods that are used and how they can become efficient. One example in this domain is the work of Rashid et al. (2008). They follow a system-controlled technique and exploit a number of

measures that origin from the information theory. Their technique is system-controlled since the computations to select items are done by the system, and the system prompts its members to provide opinions on a set of items. Furthermore, Aharon et al. (2015) present an algorithm (called ExcUseMe), which selects a predefined number of users that are more likely to be interested in interacting with new items.

2.5. Hybrid approaches

To achieve better results, researchers combine some of the above techniques.

For example, Chen et al. (2013) combine the social sub-community division and the ontology decision model, so as to enhance collaborative filtering systems.

On the other hand, Sobhanam et al. (2013) combine association rules with clustering techniques. Association rules are applied to build and expand the peoples' profiles so that they can include more ratings/domains of interest (e.g. If a user likes item A, then recommend item B). Afterwards, clustering is used to group items and make predictions.

Additionally, Nalmpantis et al. (2017), combine collaborative filtering techniques with a personality test so as to provide more personalized movie recommendations.

Last but not least, Bahramian et al. (2017) propose a hybrid interactive context-aware tourism recommender system that takes into consideration the context and the person's feedbacks. By exploiting case-based reasoning techniques and neural networks, it offers personalized tours to the users.

2.6. Other approaches

Statistical-based methods have also been used to tackle the cold start problem. For example, Peng et al. (2017) designed and proposed a new convex framework for cold-start recommendations, which performs multi-level, preference regression (MPR) to predict the ratings directly.

Other researchers take advantage of the Graph Theory to provide interesting solutions. Moreover, Rong et al. (2014) use a Monte Carlo algorithm as a basis to reduce the cold start problem. At first, they define a random walk on a bipartite graph of users and items to simulate the preference propagation among users, in order to alleviate the data sparsity problem for new members. Then, they adopt a Monte Carlo algorithm to estimate the similarity between different users. Furthermore, Zhang et al. (2010) employ an algorithm which is based on graph theory so as to suggest interesting movies. More specifically, they propose a user-tag-object tripartite graph, which considers the frequencies of tags as user preferences on different topics and tag-object links as the associations between them.

2.7. Motivation and contribution

Upon doing this work and studying the related approaches, which inspired this research, some issues and challenges have been identified. These are summarized in the following points:

- a. Regarding the works in section 2.1 and 2.6, not enough content data are always available (Rafsanjaniet al., 2013). As it was discussed above, collaborative filtering methods require information so as to be accurate. Consequently, sparse or missing data (something common in LBSs) are negative factors and decrease their performance significantly.
- b. Gathering data from external sources such as social media (works in section 2.2) is not always efficient

because sometimes they are not available or cannot be used due to interoperability problems (Carrer-Neto et al., 2012). Association rule mining systems (section 2.3) also require a lot of data to be accurate.

- c. Works that require from the person to fill-in a lot of information before using the system (section 2.4) may discourage him/her from using it. Regarding this approach, it would also be an interesting problem to identify (through data analysis) which are the most important pieces of information that increase the certainty of prediction and ask only for them.
- d. Systems that possess a predefined rule base (either engineered or mined by user data) face the challenge of selecting the most interesting rules, depending on the context. These systems may contain a lot of irrelevant, to the user's context, rules that are nevertheless fired and, thus, they will provide some non-interesting recommendations (Garcia et al., 2007).
- e. Most of the above systems are usually isolated, meaning that they focus on ranking the available items to recommend them to the user and they do not care to share knowledge and/or data with other systems in a reusable, interoperable and system-independent manner.
- f. Apart from the above challenges about the cold start problem regarding the user's scope, an issue which needs to be further investigated is the new item problem. Researchers (such as Bykau et al., 2013; Ji et al., 2015) propose to add some ratings manually to new items, so as to be able to take part into the recommendation process.

These issues that were described in detail above motivated this research. Concerning the latter's scientific contribution, it can be summarized in the following main points:

1. It proposes a novel hybrid approach for the cold start problem, which combines association rules, probability metrics and the user's context. It exploits semantic technologies in a context aware recommender system by modelling and formalizing human life patterns via logical rules (in our case, rules that recommend POI categories). A brief overview of the approach has been given in the introduction section, while it is presented in detail in sections 3 and 4. As shown in Section 5, our hybrid approach improves considerably a baseline rule association mining method.
2. It provides a novel approach for solving the new item cold start problem (issue f above) without the user's intervention. In detail, by exploiting an ontology which supports hierarchy/schema (regarding the item categories/subcategories, POIs in our case) and rules that recommend POIs' categories, a completely new item/POI can be recommended to the consumer (because every POI belongs to a specific category and, therefore, it can be recommended when the corresponding rule is fired). The proposed approach even allows detecting automatically a new category to which a POI belongs to and recommends its addition to the schema.
3. It performs a detailed overview and evaluation among some of the most well-known probability metrics for scoring rules in the case of the cold start. Additionally, a novel scoring function, suitable for the case we are studying, is proposed. This function noticeably improves the performance of our method, compared

to related metrics found in the literature, as shown in Section 5.

The scientific contribution of the proposed approach leads to some technical advantages/contributions. These can be summarized in the following points:

- i. The presented approach proposes to mine rules for one user at a time so as to be able to create a richer rule set and discover rules that are not so popular but are indeed useful. By using this rule set, issues regarding the lack of data and user participation are alleviated (issues a., b. and c. above).
- ii. By mining human daily patterns (Serrano et al., 2011) and modeling them with rules and ontologies (e.g. The most popular place categories on Saturday nights), all the technical advantages of semantic web technologies are gained. Some of these are the following (Lin, 2000; Patkos et al., 2007; Viktoratos et al., 2014; Weigand et al., 2008):
 - The capability of representing the attributes of physical entities and complex connections between them. This leads to more effective context-aware services with minimal user input.
 - Knowledge sharing and semantic interoperability, by providing a formal and general knowledge representation and reasoning standard. The knowledge construction process is enhanced because it can be easily reused and extended, thus, saving a lot of time and effort for the developers. For example, the generated rule set can be used by other LBSs to alleviate the cold start problem (e.g. when a person does not give permission to a service to use his/her data from existing social media accounts, or if he/she is not willing to insert data manually).

The above technical advantages can help in dealing with the issues of data integration and sharing (points b. and e. in the related paragraph).

- iii. By making a detailed research in scoring functions, and developing a novel one, the system is able to filter out irrelevant rules and suggest the most suitable ones depending on the person's context (issue d. above).
- iv. By using association rules in large datasets, good results- regarding speed and accuracy- are achieved (Shaw et al., 2010).
- v. Because of the fact that association rules are close to the natural language, people can read them and understand why a POI is recommended or not. This approach enhances trust and may lead to better acceptance of the suggestions by the user (Tindarev et al., 2015).

3. The Proposed Approach

The aim of the proposed approach is to recommend the most proper POI recommendation rules to a person with a limited check-in history (cold start user). More specifically, regarding the available check-in history of him/her and the available domain knowledge (check-in history of other community fellows), the goal is to recommend the N top recommendation rules about POIs to the cold start user. To begin with, the proposed approach consists of the following steps (Figure 1) in order to recommend the most appropriate rules to a new member:

- **Step 1.** Data collection

- **Step 2.** Offline Rule set construction
- **Step 3.** Online Rule recommendation

These steps are described in detail in the following sections.

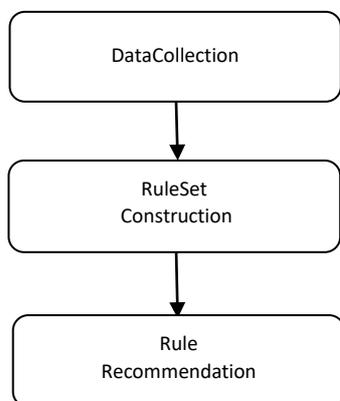


Fig. 1. The main steps of the proposed approach.

3.1. Data collection

The first step is the collection of data from a database with a satisfactory number of people and their check-in history (e.g. timestamp, coordinates and weather). Various sources (e.g. Foursquare or Facebook) can be used to collect such kind of information. When the data collection process, possibly from many sources, is completed, the next step is to integrate all the available data by transforming them into a general representation. This is achieved by using a formal ontology (e.g. Schema.org¹ in our case).

This process is summarized more formally in appendix A. After finishing the data collection process and creating the data sets, the next step is to mine the rules and create the rule sets.

3.2. Rule set construction

For the purpose of implementing the (offline) rule set construction process, an association rule mining algorithm is used (such as Apriori or FP-Growth) to mine the top rules and induce useful patterns for each user individually. The aim of this process is to create a knowledge base of user specific rules (this is named “rule set A”- S_A). Mining rules from one user at a time allows the induction of more accurate rules, since they might actually be more suitable for an individual. This is not possible if the rules are mined by the total of users at once, because such rules will fail to have the necessary (above the threshold) support and, consequently, they will not be included in the resulting rule set (Lin, 2000). Apart from this, by following this method, the specificities of each individual person can be captured. Moreover, considering an online system which follows this approach, better performance is achieved by mining rules only over one small subset of the data which is related to the target user rather than the whole dataset.

Rules can include contextual attributes in the condition (IF) part and category attribute in the recommendation (Then) part. Category attribute is used as the consequence because more frequent rules can be discovered compared to the recommendation of specific POIs. By adopting this technique, a completely new item/POI which belongs to this category can be recommended. Consequently, this leads to alleviating the new

item problem without user intervention, as it was discussed in the contribution section 2.7.

Finally, all the rules are transformed from the output format of the association rule mining algorithm into an executable machine language so as to be executable by a system.

This process is formulated in appendix A. After completing all of the above, the step to follow is “Rule recommendation”.

3.3. Rule recommendation

After the completion of the rule collection process and the creation of rule set A, the next phase is the implementation of the hybrid approach. In this step, rule set A is considered as pre-defined domain knowledge (although it has been generated offline by mining users’ data) and it is combined with rules that are mined on-line using (again) association rule mining. In detail, at run time, when a new member enters the system (with few check-ins), the system uses association rule mining to mine the top rules for him/her (this is called “rule set B”- S_B). Rule sets A and B are merged (S_{A+B, μ_c}) to create the final candidate-set for rule recommendation. If a person has no check-ins, the N most popular rules from rule set A are recommended (S_N).

The last part of the recommendation process is the design and the implementation of a scoring function to rank the rules. The top rules with the highest score are recommended to the new user in order to provide the best available suggestions and deal with the cold start problem (Figure 2). This process is represented formally in appendix A.

4. Implementation

Following the abstract presentation of the approach, in this section all of the implementation details will be described. Before going any further, it should be mentioned that the implementation is based on a rule-based context aware system that comprises a previous work of ours, called “GeoSocial SPLIS” (Viktoratos et al., 2017). This system collects users’ preferences represented in the form of rules, evaluates them and recommends places to the users based on the rules that fired.

4.1. Data collection

Regarding the first step, the data collection process, our initial source² includes data from Foursquare regarding the city of New York (yang et al., 2015). In detail, the initial data source contains 227,428 check-ins from over 1000 unique users in New York City. Each check-in is associated with a time stamp, GPS coordinates and a place category (compatible with Foursquare). After that, the existing data that are described above have been enriched with weather information³. For each timestamp, the corresponding weather history has been associated with it. After having collected the data, we aligned Foursquare categories with the Place types found at schema.org. A mapping has been created manually between Foursquare and schema.org attributes for achieving this. For example, each Foursquare category has been aligned to the most similar Place subclass of schema.org (e.g. Dentist’s Office =>schema:Dentist). Finally, for every check-in record, there are 4 context attributes (a_i): $A = \{place\ category, day, time\ period, weather\}$ (Table 1).

¹<http://schema.org/>

² <https://sites.google.com/site/yangdingqi/home/foursquare-dataset>

³<https://www.wunderground.com/>

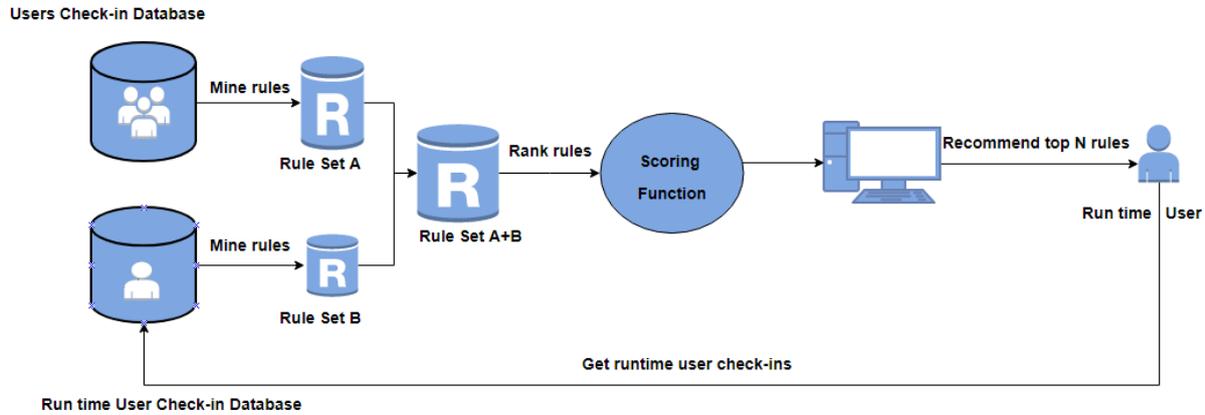


Fig. 2. The Rule recommendation process.

Table 1. Attributes and values of user data.

ATTRIBUTES (a_k)	POSSIBLE VALUES (V_{a_k})
Schema.org category	All the possible place categories that are supported by schema.org ⁴
Day	Mon-Sun
Time period	Morning (06:00-12:00), Afternoon (12:00-17:00), Evening (17:00-21:00), Night (21:00-06:00)
Weather	Clear, Cloudy, Rainy, Hazy, Snowy

4.2. Rule set construction

After the data collection process, this step is to mine the user-specific place recommendation rules and create rule set A (S_A). We have considered 500 users ($N_U=500$) with at least 100 check-ins. For each user u_i , regarding his/her check-in history C_{u_i} , the Apriori algorithm (Agrawal et al., 1994; Girotra et al., 2013) has been used to gather the top 5 rules R_{m,u_i} ($N_M=5$).

Apriori is one of the most well-known association rule mining algorithms. The basic idea behind this algorithm is to discover the most frequent item sets in a database that contains transactions. The output is a set of rules that result from these item sets and represent the relations among the data (Kumbhare et al., 2014; Qiankun, 2003). In detail, let us consider the items $O = \{O_1, O_2, \dots, O_n\}$ and a set of m unique records $R = \{R_1, R_2, \dots, R_m\}$ that each one contains a subset of data objects $R \subseteq O$. If, for example, a frequent item set $\{O_2, O_3, O_5, O_7\}$ is found, some example rules could be $\{O_2, O_3, O_7\} \Rightarrow \{O_5\}$ or $\{O_2, O_3, O_7\} \Rightarrow \{O_5, O_7\}$.

Moreover, the Weka software has been used to implement the rule mining process (Tanna et al., 2014). A script which uses

the Weka API has been created in Java Server Pages (JSP) to implement the rule mining process automatically (Duane et al., 2001). Furthermore, JSP technology has been chosen because GeoSocial SPLIS's implementation is based on it and it could easily be combined with WEKA and other technologies (Tanna et al., 2014).

Regarding the rules' structure, contextual attributes a_k , such as day, time period and weather, are included in the IF part ($cond_{m,u_i}$) and a schema.org category attribute in the Then part (rec_{m,u_i}). Table 2 illustrates a sample of the rules that result from WEKA. Furthermore, when a rule is mined by more than one user, then the pop_{R_q} attribute that represents the number of users that possess the rule R_q is increased by one, forming the pop set (see Appendix A).

After collecting all the rules, they are transformed into Jess, so as to become not only machine executable (Table 3), but also human readable (for programmers). Jess has been chosen for this purpose, because GeoSocial SPLIS uses it for rule execution. Jess is based on an improved version of the Rete algorithm and uses adaptive indexing techniques to achieve better performance (Friedman-Hill, 2003).

4.3. Rule recommendation

After completing the previous two steps (offline) and creating the rule set A , the next step is the implementation of the hybrid approach and the on-line recommendation of rules to the person. More specifically, at run time, when a new user u_c , who has less than 10 check-ins (Levi et al., 2012), enters the system, the latter uses the Apriori algorithm to mine the top five recommendation rules ($N_M=5$) forming the rule set B (S_{B,u_c}). Rule sets A and B are then merged (S_{A+B,u_c}) to create the final list of the candidate rules to be recommended to the user (S_N).

The last part of the recommendation process is the implementation of a scoring function $score_{R_q}$ to rank the rules of the merged rule sets A and B . The top five rules with the highest score ($N=5$) are recommended to the new user in order to provide the best available place recommendations and deal with the cold start problem. The number of rules has been set to five so as to be easily manageable by a person (e.g. Avoid scrolling in a mobile device). Researchers (Han et al., 2014) that provide recommendations in mobile devices keep the number of N small (e.g. 5) due to screen size limitations.

⁴<https://schema.org/docs/full.html>

Table 2. Rule examples that resulted from Weka

DAY=Mon Weather=Cloudy TIME PERIOD =Afternoon ==> SCHEMA.ORG CATEGORY=ProfessionalService
DAY=Thu Weather=Cloudy TIME PERIOD =Evening ==> SCHEMA.ORG CATEGORY=CafeOrCoffeeShop
DAY=Sat Weather=Clear TIME PERIOD =Night ==> SCHEMA.ORG CATEGORY=BarOrPub
DAY=Sun TIME PERIOD =Night ==> SCHEMA.ORG CATEGORY=Restaurant

Table 3. Rule examples in Jess

(defrule 1 (checkin (DAY Mon)(Weather Cloudy)) =>(SCHEMAORGCATEGORY ProfessionalService))
(defrule 2 (checkin (DAY Thu)(Weather Cloudy)(TIMEPERIOD Evening)) =>(SCHEMAORGCATEGORY CafeOrCoffeeShop))
(defrule 3 (checkin (DAY Sat)(Weather Clear)(TIMEPERIOD Night))=>(SCHEMAORGCATEGORY BarOrPub))
(defrule 4 (checkin (DAY Sun)(TIMEPERIOD Night))=>(SCHEMAORGCATEGORY Restaurant))

Regarding the scoring function $score_{R_q}$, the general idea behind it, is to consider and combine well known metrics regarding association rules, such as confidence and support, with the popularity concept (McGarry, 2005).

Confidence is an indication of how often the rule condition has been found to be true. Considering that the rule base contains rules in the form of "If $Cond_1$ and $Cond_2$ and ... $Cond_n$ Then Rec ", it can be defined as an estimate of the conditional probability $P(Rec|Cond)$, namely the probability of finding the recommendation Rec (Then-part or Right-Hand-Side - RHS) of the rule in records (in our case check-ins) under the assumption that these records also contain the condition $Cond$ (If-part or Left-Hand-Side - LHS). On the other hand, rule support reflects the coverage of the rule. It can be interpreted as the probability $P(Cond \cap Rec)$, i.e. the number of records containing both RHS (Then-part) and LHS (If-part) of the rule among the total number of records. It ought to be noticed that since rule condition is a conjunction of simpler condition elements, the probability of the condition is the probability of the conjunction of the elements $P(Cond) = P(Cond_1 \cap Cond_2 \cap \dots \cap Cond_n)$.

In addition to this, for the purpose of giving more weight to place categories, the support of the Then-part is considered (since it refers to place categories). Furthermore, many researchers are taking into consideration negative confidence and support (basically the misses, when the LHS is true and the RHS is false) to get better results (McGarry, 2005). Regarding our case, the experiments (see next section) signified that the above factors improved the results.

Apart from the above metrics, the popularity concept is widely used in recommender systems to improve the results. Depending on the case, it can be quantified by many metrics such as the number of likes, reviews or the times an item has been purchased (Levi et al., 2012; Sedhain et al., 2014). In this work,

the rule popularity metric for a rule R_q (pop_{R_q}) is defined as the number of users that contain the rule R_q in their rule set, divided by the total number of users (basically the probability for a rule R_q in the rule set of being possessed by a user or, in other words, the ratio of the users that possess the rule R_q). The evaluation section will demonstrate the positive effect of this factor on the results. All the above metrics are shown in Table 4.

After determining the metrics to be included in the scoring function, the next step is to combine them together. The scoring function is a weighted sum of two parts: the first part takes into consideration all the above association rule metrics; namely confidence, support, missed hits, while the second part is the rule popularity metric (pop_{R_q}).

A common practice among researchers is to multiply confidence with support in order to take into consideration both metrics (McGarry, 2005). In the same spirit, in our metric, $confidence(CondRec)$ is multiplied with $support(CondRec)$ and $support(Rec)$, as it was discussed above, to give more weight to the place category attribute. In addition to this, we subtract the misses for improved results (McGarry, 2005). The above quantity (subtraction of probabilities) ranges between -1 and 1, therefore to be normalized in the range 0-1, we add 1 and we divide by 2. From now on in this manuscript, the first part of the scoring function will be referred to as "Then Part and Missed Hits Emphasis (TME)".

After completing the first part of the scoring function (TME), the next step is to combine it with rule popularity factor (pop - the second part of the scoring function) so as to complete it. Finally, a weighted sum of the two parts constitutes the scoring function. The proposed "Then Part and Missed Hits Emphasis Popularity-based" scoring function (TME-P) is presented below (Equation 1).

Equation1

$$score_{R_q} = w_1 \times \frac{1 + P(Rec_{R_q}|Cond_{R_q}) \times P(Cond_{R_q} \cap Rec_{R_q}) \times P(Rec_{R_q}) - P(\neg Rec_{R_q}|Cond_{R_q}) \times P(Cond_{R_q} \cap \neg Rec_{R_q}) \times P(\neg Rec_{R_q})}{2} + w_2 \times pop_{R_q}$$

Table 4. Metrics and definitions

Metric	Definition	Calculation
Support ($CondRec$)	Number of times both $Cond$ and Rec are true regarding the total number of check-ins / Total number of check-ins	$P(Cond \cap Rec)$
Support ($Cond\neg Rec$)	Number of times $Cond$ is true and Rec is false regarding the total number of check-ins / Total number of check-ins	$P(Cond \cap \neg Rec)$
Support (Rec)	Number of times Rec is true regarding the total number of check-ins / Total number of check-ins	$P(Rec)$
Support ($\neg Rec$)	Number of times Rec is false regarding the total number of check-ins / Total number of check-ins	$P(\neg Rec)$
Confidence ($CondRec$)	Number of times both $Cond$ and Rec are true regarding the total number of check-ins / Number of times $Cond$ is true regarding the total number of check-ins	$P(Rec Cond)$
Confidence ($Cond\neg Rec$)	Number of times $Cond$ is true and Rec is false regarding the total number of check-ins / Number of times $Cond$ is true regarding the total number of check-ins	$P(\neg Rec Cond)$
Rule popularity (pop)	Number of users that possess a rule/ Total number of users	pop

5. Evaluation

To assess our proposal, an extensive evaluation consisting of three parts has been performed. First of all, well known probability-based metrics have been tested to evaluate their performance and find the most suitable regarding our case. Then, experiments to evaluate the user specific rule base and signify its contribution to the cold start problem have been conducted. Finally, the last part of the evaluation is about the integration of the scoring function into GeoSocial SPLIS for testing it on real people.

5.1. Scoring function evaluation

The first set of the experiments is about the evaluation of the TME scoring function and its comparison with other well-known probability metrics (Geng and Hamilton, 2006; Tamir and Singer, 2006). A detailed research has been conducted and a variety of 24 metrics have been tested. Apart from studying and presenting their performance for this purpose, the aim is to prove the contribution and the novelty of the designed evaluation function (scientific contribution point 1 and technical contribution point ii in section 2.7). For the purpose of implementing this, data from 600 users (having over 9000 check-ins in total) were taken into account. A 6-fold cross-validation was conducted to achieve more reliable results. In detail, each time the experiment was conducted, 500 were used for creating the user specific rule base (rule set S_A), as described above, and the rest 100 for testing. Each dataset contained on average 1268 rules. The minimum was 1235 and the maximum 1298.

For each one of the 100 testing users, all the existing rules of the rule base (rules from 500 users, i.e. rule set S_A) and 5 rules resulting from running Apriori on each new user u_c (i.e. rule set

S_{B,u_c}) were evaluated with the scoring function and the top 5 rules were recommended. It is worth mentioning that since the cold start problem is examined, in the experiments only the first 3, 6 and 9 check-ins, respectively, were considered for each testing user.

To evaluate the results, each one of the 5 suggested rules (R_n) was evaluated using the total of the user's check-ins in order to calculate its actual/total confidence $P_{R_n}(Rec|Cond)$. The overall score for a user u_j was the average score (confidence) of the top 5 suggested rules from rule sets S_A and S_{B,u_c} for him/her: $Sc_{u_j} = (\sum_{n=1}^5 P_{R_n}(Rec|Cond))/5$. After that, the average confidence of all 100 testing users for each run r was calculated ($T_r = (\sum_{i=1}^{100} Sc_{u_j})/100$). Finally, because the experiment was conducted for 6 different user sets / runs (6-fold cross validation), for better reliability, the overall average confidence was calculated as $T = (\sum_{r=1}^6 T_r)/6$.

As it was discussed above, well known probability metrics were also used in the first part of the scoring function (instead of TME) so as to find the most proper for this occasion. Table 5 illustrates the results in depth. In general, it can be observed that there are no extensive differences among the metrics. Nevertheless, the proposed *TME-P* metric performs slightly better in all cases. *Two-way support* also performs very well in all cases (3, 6, 9 check-ins). *Certainty factor* has good results for the case of having 9 check-ins available, as well. Finally, it should be mentioned that, concerning the case of an individual with 3 check-ins available, *relative risk* can also provide useful suggestions, although for a person with 9 check-ins it has the worst performance. This metric is highly associated to the type of data and fails to perform well if the data are noisy (check-in data are noisy) (Hu et al., 2015; Sackett, 2001). Figure 3 illustrates a comparison including the top 3 and last 3 metrics (regarding the results in 3, 6 and 9 check-ins).

Table 5. Metrics results regarding confidence for 3, 6 and 9 check-ins.

Metric	Math Type	3 ck	6 ck	9 ck
TME	$0.5 * [(1 + (P(Rec Cond) * P(Cond \cap Rec) * P(Rec) - P(\neg Rec Cond) * P(Cond \cap \neg Rec) * P(\neg Rec)))]$	0.28638	0.32253	0.34594
Two Way Support	$P(CondRec) * \log_2(\frac{P(CondRec)}{P(Cond)P(Rec)})$	0.27637	0.30213	0.32312
Certainty Factor	$\frac{P(Rec Cond) - P(Rec)}{1 - P(Rec)}$	0.26396	0.29832	0.32183
Leverage	$P(Rec Cond) - P(Cond)P(Rec)$	0.25659	0.29319	0.31456
LeastContradiction	$\frac{P(CondRec) - P(Cond\neg Rec)}{1 - P(Rec)}$	0.26706	0.28918	0.31197
Yules Y	$\frac{\sqrt{P(CondRec)P(\neg Cond\neg Rec)} - \sqrt{P(Cond\neg Rec)P(\neg CondRec)}}{\sqrt{P(CondRec)P(\neg Cond\neg Rec)} + \sqrt{P(Cond\neg Rec)P(\neg CondRec)}}$	0.27359	0.29509	0.30593
Yules Q	$\frac{P(CondRec)P(\neg Cond\neg Rec) - P(Cond\neg Rec)P(\neg CondRec)}{P(CondRec)P(\neg Cond\neg Rec) + P(Cond\neg Rec)P(\neg CondRec)}$	0.27359	0.28584	0.30593
Jaccard	$\frac{P(CondRec)}{P(Cond) + P(Rec) - P(CondRec)}$	0.26535	0.28327	0.29612
Cosine	$\frac{P(CondRec)}{\sqrt{P(Cond)P(Rec)}}$	0.26535	0.28208	0.29129
Added Value	$P(Rec Cond) - P(Rec)$	0.25766	0.28213	0.29053
Accuracy	$P(CondRec) + P(\neg Cond\neg Rec)$	0.26255	0.27464	0.28501
Confidence	$P(Rec Cond)$	0.26325	0.27621	0.28492
Lift	$\frac{P(Rec Cond)}{P(Rec)}$	0.27112	0.27456	0.28483
Linear Correlation Coefficient	$\frac{P(CondRec) - P(Cond)P(Rec)}{\sqrt{P(Cond)P(Rec)P(\neg Cond)P(\neg Rec)}}$	0.27696	0.27725	0.28473
Conviction	$\frac{P(Cond)P(\neg Rec)}{P(Cond\neg Rec)}$	0.26203	0.27423	0.28179
Odds Ratio	$\frac{P(CondRec)P(\neg Cond\neg Rec)}{P(Cond\neg Rec)P(\neg RecCond)}$	0.25805	0.27863	0.28070
Collective Strength	$\frac{P(CondRec) + P(\neg Rec \neg Cond)}{P(Cond)P(Rec) + P(\neg Cond)P(\neg Rec)}$	0.25817	0.27027	0.27825
One Way Support	$P(Rec Cond) * \log_2(\frac{P(CondRec)}{P(Cond)P(Rec)})$	0.26566	0.27135	0.27544
Gini Index	$\frac{P(Cond)[P(Rec Cond)^2 + P(\neg Rec Cond)^2] + P(\neg Cond)[P(Rec \neg Cond)^2 + P(\neg Rec \neg Cond)^2] - P(Rec)^2 - P(\neg Rec)^2}{P(Cond)P(Rec) + P(\neg Cond)P(\neg Rec)}$	0.24373	0.27242	0.27390
Klogsen	$\sqrt{P(CondRec)} (P(Rec Cond) - P(Rec))$	0.26321	0.27129	0.27323
Loevinger	$1 - \frac{P(Cond)P(\neg Rec)}{P(Cond\neg Rec)}$	0.26321	0.26912	0.27013
J-Measure	$P(CondRec) \log_2(\frac{P(Rec Cond)}{P(Rec)}) + P(Cond\neg Rec) \log_2(\frac{P(\neg Rec Cond)}{P(\neg Rec)})$	0.24373	0.25120	0.26593
Information Gain	$\log_2 \frac{P(CondRec)}{P(Cond)P(Rec)}$	0.24445	0.25023	0.26390

Metric	Math Type	3 ck	6 ck	9 ck
RelativeRisk	$\frac{P(Rec Cond)}{P(Rec \neg Cond)}$	0.27707	0.27398	0.23421

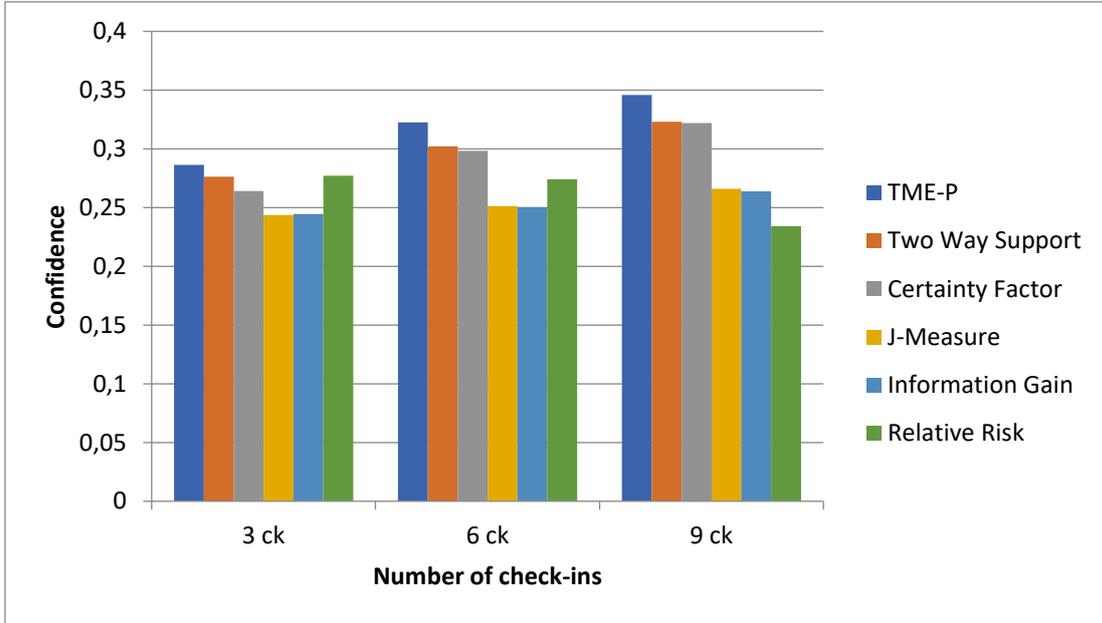


Figure 3. Comparison of metrics regarding their confidence for 3, 6 and 9 check-ins

5.2. User specific rule base evaluation

Apart from testing for the proper scoring function, we also conducted experiments to evaluate the whole approach. The aim of these experiments is to measure the effectiveness of a user specific rule base for the case of the cold start problem, demonstrate its value and prove our point about the quality of the rule base (technical contribution point iii in section 2.7).

Two different experiments were conducted for this purpose. The first one was about testing the rule sets S_A and S_{B,u_c} alone. In addition, apart from these two, another rule set (*rule set C* - S_C) was also taken into consideration. Rule set C was created by mining rules from the total of all users' check-in history at once and is considered as a baseline method. The second one was about probing for the best percentage of the rule popularity factor w_2 in the scoring function and proving that it improves the results. The experiments were conducted exactly as in section 5.1. The evaluation was conducted using the same dataset and technique as before, by calculating the overall confidence of the testing users.

In the first experiment, in addition to testing the approach which combines rule sets S_A and S_{B,u_c} as mentioned above, rule sets S_A and S_{B,u_c} and S_C were also tested alone. Table 6 and Figure 4 demonstrate the results. Merging the two rule sets S_A and S_{B,u_c} gives the best results. The most noteworthy fact is that a well-constructed user specific rule base S_A (by mining rules from a database of other community fellows) can outperform a method which simply mines rules by the run time user directly S_{B,u_c} , or a

method which mines rules by the total of all users S_C , and provide better recommendations for the case of the cold start.

The second part of this experiment evaluates the influence of the rule popularity factor (*pop*) on the scoring function and proves its contribution (scientific contribution point 1 and technical contribution point ii). Therefore, experiments were conducted to estimate the optimal weights regarding the two parts (TME - w_1 weight in equation 1 and popularity- w_2 weight) of the TME-P scoring function. The tested values were 100%-0%, 80%-20%, 50%-50%, 20%-80% and 0%-100%, respectively. Along with the proposed metric, *two-way support* and *certainty factor* metrics were also tested because they have the best performance among the rest of the examined metrics. The experiments were carried out for the case of an individual who has 9 check-ins available. Table 7 and Figure 5 present the results, showing that the performance of the scoring function, slightly but consistently, increases with the increase of the weight of the rule popularity factor, until it reaches 80% (when popularity factor has 100% weight the TME has 0% and, thus, the performance is reduced). However, the most noteworthy fact is that when we exclude the rule popularity factor from the TME-P (0% w_2 weight in equation 1), the performance is reduced significantly, which proves the worthiness of the popularity metric. Apart from this, recommending only the most popular rules ($w_2=100%$, $w_1=0%$) is an option of recommending rules to an individual with no check-in history at all, having slightly reduced but, still, considerably good performance.

Table 6. Comparison of the overall results regarding confidence for various rule sets and their combination.

Number of check-ins	3	6	9
Rule Set A+B (S_{A+B,u_c})	0.286	0.322	0.345
Rule Set A (S_A)	0.273	0.304	0.328
Rule Set B (S_B)	0.243	0.286	0.293
Rule Set C (S_C)	0.236	0.268	0.284

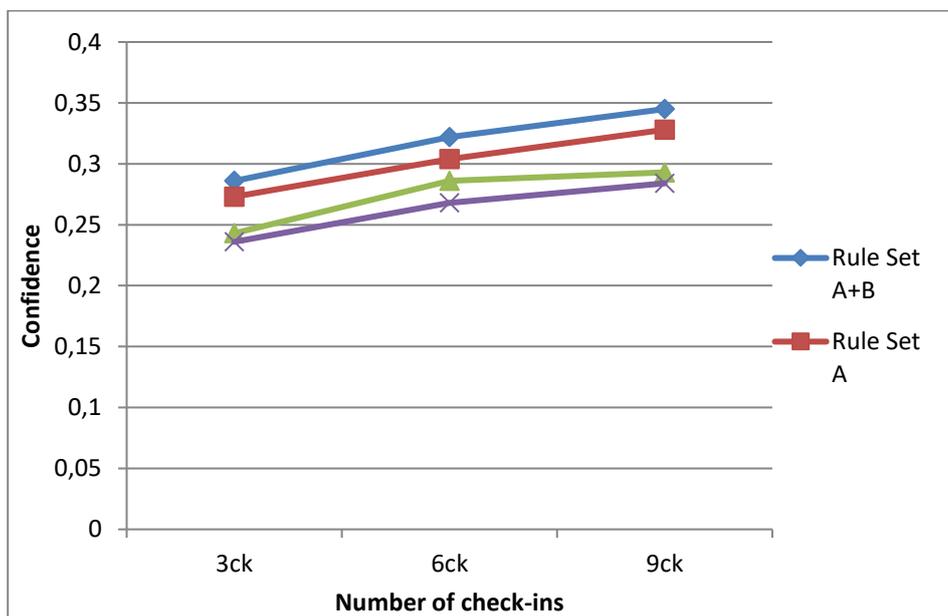


Fig. 4. Method, Rule sets A and B standalone overall results regarding confidence.

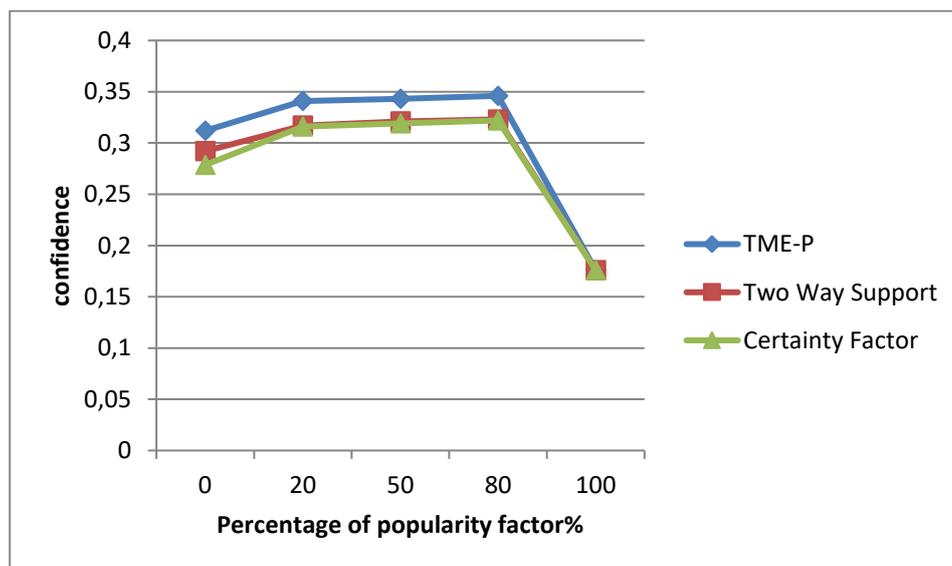


Fig. 5. Scoring function overall confidence results for 3 metrics regarding the percentage of the rule popularity factor

Table 7. Overall confidence score of metrics regarding the weights of the two parts

Percentage (%) of TME vs. rule popularity	100-0	80-20	50-50	20-80	0-100
TME-P	0.312	0.341	0.343	0.346	0.176
Two Way Support	0.2919	0.317	0.321	0.323	0.176
Certainty Factor	0.2785	0.316	0.319	0.322	0.176

5.3. Real user evaluation

The last part of the evaluation was to validate our method on real circumstances by integrating it in Geosocial SPLIS (Viktoratos et al., 2017). The system was modified in a way to be able to collect the users' check-ins, apply the approach we have proposed in this paper and, subsequently, recommend the top 5 rules.

In this evaluation, 18 users took part, ranging between the ages of 21 to 53, and with various technical and educational backgrounds. Each person was asked to make 3 check-ins in the system. The users could see both the recommended places and the recommended rules. After making the third check-in, the participants were asked to evaluate each of the 5 suggested rules in a Likert like scale by assigning the following values:

- Not interesting at all (-2)

- Not interesting (-1)
- Neutral (0)
- Interesting (1)
- Very interesting (2)

A total score over all rules was calculated every time for each person. As it can be observed (Table 8) the score could take values in the range of [-10, 10] (the sum of the scores for the 5 rules). Finally, the total average score for all the users that took part was calculated. The total average score value was 7.25/10.0, with a standard deviation at 1.83, indicating that the recommended rules were on average interesting to very interesting for the participants and provided useful suggestions. The minimum value and the maximum values among peoples' scores was 1 and 9, respectively (Table 8). However, most participants gave high scores, since the median value was 8, very close to the maximum.

Table 8. Real user evaluation results regarding minimum, maximum and average value.

	Minimum	Maximum	Average	Median	Stdev
Overall Score	1	9	7.25	8	1.83

6. Conclusions

In this manuscript, a detailed overview about the cold start problem in context-aware recommender systems has been performed. After describing the challenges and the problems of the domain in depth (lack of data, new item, interoperability, etc.), a novel hybrid approach has been proposed which combines community generated knowledge, semantic web technologies, association rules and probability metrics to alleviate the cold start problem in rule-based, context-aware recommender systems. Regarding a person with a limited check-in history available, the aim of this proposal is to recommend the most appropriate rules (rules that recommend POIs) to him/her, based on his/her context.

The presented approach:

- Constructs a user-specific rule base by mining offline existing members' check-in history.
- Uses association rules (Apriori algorithm was selected) to mine rules for a run time user (with limited check-in history).
- Merges the above two rule sets into a single rule base.
- Proposes an innovative scoring function (TME-P) to rank the available rules and to suggest the most suitable of them to the user. The scoring function combines the

relevance and the popularity of a rule. This has been implemented so as to avoid the problem of having a lot of irrelevant rules.

The presented approach exhibits the following advantages/contributions:

- Proposes a novel scoring function, suitable for the case we study.
- Uses semantic web technologies to model and formalize human life patterns via logical rules (rules that recommend POI categories).
- By using attributes compatible with a formal ontology in rules, interoperability and reusability are gained. The proposed approach generates a rule set that can be reused by other LBSs.
- New items can be recommended automatically. Due to the fact that the rules recommend POI categories, when fired, they recommend POIs that belong to these categories.
- The rule base can provide more accurate recommendations to a new/cold start user, since it combines data from all members' history (mining rules for each user individually) with the limited history of the current user.
- By mining rules for each user individually (instead of for all members at once) a more accurate rule set was constructed.

- The solution we proposed to the cold start user problem does not rely on either external data sources or information provided manually by the user.
- The user can read the rules and understand why a POI is recommended or not. This enhances trust and may lead to better acceptance of suggestions by the consumer.

The evaluation results that have been presented in section 5 confirmed all the above and demonstrated the value of this proposal. Specifically, the evaluation of the proposed TME-P scoring function shows an average of 5.7% performance improvement over the best performing of the existing scoring functions in the literature (Table 5). Furthermore, the performance of our hybrid methodology improves by an average of 20.9% the performance of a baseline rule association mining method (Table 6), that uses all users' check-ins to mine the most confident rules, whereas the combination of all proposed method components is performing on average 5.3% better than any of the components alone. Finally, the feedback from real people was really encouraging and showed that the proposed approach can work well on real systems.

Among our future plans, is to adjust the approach and evaluate it not only in terms of accuracy, but also in terms of other measures such as novelty, serendipity and diversity. Another interesting factor which needs to be examined is the effect of factor N (the number of recommended rules) on the results. Additionally, another future research direction would be the combination of the proposed method with pure knowledge-based techniques from previous works (Viktoratos et al., 2015; Viktoratos et al., 2017) in a hybrid, dynamically adaptable context-aware rule-based recommendation method that gains the benefits of both worlds. Finally, an interesting problem to explore would be to identify (through data analysis) which are the most important of pieces of information that increase the certainty of prediction and include only those both in automated data collection approaches, like ours, or in approaches that ask for users to manually fill-in their profile.

Appendix A. Formalization of our approach

Our approach consists of three phases: Data collection, Rule set construction and Rule recommendation (see Section 3), which are presented in a formalized way here.

Data collection

The formalization of the first phase of our approach defines the collections of the involved data and entities, including their attributes and values (users, context-related attributes, POIs and check-ins). In detail:

- The set of *users* is defined as $U = \{u_i \mid i \in [1 \dots N_U]\}$, where N_U is the total number of registered users.
- The set of available *attributes* (compatible with the schema.org ontology) is defined as $A = \{a_k \mid k \in [1 \dots N_A]\}$, where N_A is the number of available context attributes.
- The set of available attributes' *values* is defined as $V_{a_k} = \{val_{t,a_k} \mid t \in [1 \dots T_{a_k}]\}$, where T_{a_k} is the number of possible available values regarding the attribute a_k .
- The set of *POIs* is defined as $P = \{p_l = \langle pid, location, type \rangle \mid l \in [1 \dots N_P]\}$, where N_P is the total number of POIs, *pid* is the ID of the POI, *location* is the location of the POI and *type* is the POI category, $type \in Cat$, the set of categories of places found in the schema.org ontology. Notice that

we assume that each POI is an instance of a unique *Place Category* class.

- Each *check-in* c_j is defined as a tuple $\langle a_1=val_{a_1}, \dots, a_k=val_{a_k}, cat, u \rangle_j$, where $val_{a_k} \in V_{a_k}$ is the value of the context attribute $a_k \in A$, $u \in U$ is a user that checked-in at a place $p \in P$ and *cat* is the place category of $p = \langle pid, location, cat \rangle$. The whole set of check-ins is defined as $C = \{c_j \mid j \in [1 \dots N_C]\}$, where N_C is the total number of check-ins in the database.
- Every user u_i has a set of check-ins $C_{u_i} = \{ \langle a_1=val_{a_1}, \dots, a_k=val_{a_k}, cat, u \rangle_j \in C \mid u_j = u_i \}$, that is a subset of C and $C = \bigcup_{u_i \in U} C_{u_i}$.

Rule set construction

In the second phase, for every user $u_i \in U$ association rules are mined from his/her set of check-ins C_{u_i} . The rule set which is created is defined as $S_{u_i} = \{ R_{m,u_i} \mid m \in [1 \dots N_M]\}$. The top N_M rules are ranked using the rule confidence as calculated by the rule mining algorithm. Notice that N_M is predefined and common for all users. Each rule R_{m,u_i} is defined as a $\langle cond_{m,u_i} \rightarrow rec_{m,u_i} \rangle$ expression, where $cond_{m,u_i} = \{ \langle a_k = val_{a_k} \rangle \mid a_k \in A, val_{a_k} \in V_{a_k} \}$ and $rec_{m,u_i} \in Cat$.

After that, the *rule set* A is created, which is defined as $S_A = \bigcup_{u_i \in U} S_{u_i} = \{ R_q \mid q \in [1 \dots N_Q]\}$, where N_Q is the total number of unique rules in the rule set.

Because of the fact that a rule R_q may be included multiple times in each individual user rule set S_{u_i} , the popularity concept pop_{R_q} is defined. It is the number of times the rule R_q exists at each individual user rule set S_{u_i} and is defined as $pop_{R_q} = \sum_{u_i} exists(R_q, S_{u_i})$, where $exists(r, S)$ is a function, such that:

$$exists(r, S) = \begin{cases} 1, & \text{when } r \in S \\ 0, & \text{when } r \notin S \end{cases}$$

The last step of this phase is to define the set that includes the popularity of all rules $Pop = \{ pop_{R_q} \mid R_q \in S_A \}$.

Rule recommendation

In the third phase, the current run-time user (denoted by u_c) has a set of check-ins which is defined as $C_{u_c} = \{ \langle a_1=val_{a_1}, \dots, a_k=val_{a_k}, cat, u \rangle_j \in C \mid u_j = u_c \}$. Notice that we assume that $|C_{u_c}| < 10$. When u_c is inserted into the system, association rules techniques are used to mine rules from C_{u_c} and create the *rule set* $B: S_{B,u_c} = \{ R_{m,u_c} \mid m \in [1 \dots N_M]\}$, with the top N_M rules, as described in section 3.2. After that, the two rule sets A and B are merged together. This final set is defined as $S_{A+B,u_c} = S_A \cup S_{B,u_c}$.

Next, a scoring function $score_{R_q}(C, Pop)$ scores each rule, based on various metrics discussed in sections 4 and 5. The input to this scoring function is the set of check-ins C and the set Pop which includes the popularity of each rule. Then a set $Score$ that includes the scores of all rules in set S_{A+B,u_c} is constructed as $Score = \{ score_{R_q}(C, Pop) \mid R_q \in S_{A+B,u_c} \}$.

Finally, the set S_N that includes the top N ranked rules in set S_{A+B,u_c} is constructed as $S_N = \{ R_n \in S_{A+B,u_c} \mid score_{R_n} \in Score, score_{R_n} \in \max_N(Score), n \in [1 \dots N]\}$. This set of rules is the one presented to the current user at run-time.

Acknowledgments

We would like to thank the anonymous reviewers for their valuable comments and insights.

References

- Adomavicius G., A. Tuzhilin, Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions, *IEEE Trans. Knowl. Data Eng.* 17 (6) (2005) 734–749.
- Aharon M., Anava, O., Avigdor-Elgrabli, N., Drachler-Cohen, Shahar Golan, D., Somekh, O. ExcUseMe: Asking Users to Help in Item Cold-Start Recommendations. In *Proceedings of the 9th ACM Conference on Recommender Systems (RecSys '15)*. ACM, New York, NY, USA, 83-90 (2015).
- Agrawal R., Srikant, R. Fast algorithms for mining association rules. *Proceedings of the 20th International Conference on Very Large Data Bases, VLDB*, pages 487-499, Santiago, Chile, September 1994.
- Bahramian, Z., Abbaspour R. and Claramunt, C. A Cold Start Context-Aware Recommender System for Tour Planning Using Artificial Neural Network and Case Based Reasoning. *Mobile Information Systems*, vol. 2017, Article ID 9364903, 18 pages, 2017
- Bobadilla J., F. Ortega, A. Hernando, A. Gutiérrez, Recommender systems survey, *Knowledge-Based Systems*, Volume 46, July 2013, Pages 109-132, ISSN 0950-7051, <http://dx.doi.org/10.1016/j.knosys.2013.03.012>.
- Burke R. (2000). Knowledge-based recommender systems. In Kent, A. (Ed.), *Encyclopedia of Library and Information Systems*, Vol. 69, pp. 180–200. Marcel Dekker, New York.
- Bykau S., F. Koutrika, and Y. Velegrakis. Coping with the persistent cold-start problem. In *Personalized Access, Profile Management, and Context Awareness in Databases, PersDB*, 2013
- Cane Wing-ki Leung, Stephen Chi-fai Chan, and Fu-lai Chung. 2008. An empirical study of a cross-level association rule mining approach to cold-start recommendations. *Knowledge-Based Syst.* 21, 7 (October 2008), 515-529.
- Carrer-Neto W., M. Hernández-Alcaraz, R. Valencia-García, F. García-Sánchez. Social knowledge-based recommender system. Application to the movies domain, *Expert Systems with Applications*, Vol. 39, Issue 12, 15 September 2012, p. 10990-11000.
- Castillejo E., Almeida A., Lopez-de-Ipina D., Alleviating cold-user start problem with users' social network data in recommendation systems. *Problems and Applications in Artificial Intelligence*, 2012.
- Ciaramella A., Cimino, M.G., Lazzarini, B., Marcelloni, F. 2009. Situation-Aware Mobile Service Recommendation with Fuzzy Logic and Semantic Web. *Ninth Int. Conference on Intelligent Systems Design and Applications*, IEEE.
- Duane K. Fields, Mark A. Kolb, Shawn Bayern. *Web Development with Java Server Pages*. Manning Publications ISBN:193011012X (2001)
- Ernest Friedman-Hill: *Jess in Action. Rule-Based Systems in Java*. Manning Publications. ISBN-10: 1930110898, pp. 32-33(2003)
- Garcia J., Romero C., Ventura S., Calders T. Drawbacks and solutions of applying association rules mining in learning management systems. *Proceedings of the International Workshop on Applying Data Mining in e-learning (ADML'07)*, Crete, Greece 2007, p. 13-22.
- García-Crespo Á., López-Cuadrado, J.L., Colomo-Palacios, R., González Carrasco, I., Ruiz-Mezcua, B. 2011. Sem-Fit: A semantic based expert system to provide recommendations in the tourism domain. *Expert systems with applications*, 38.
- Geng L., H. Hamilton. Interestingness Measures for Data Mining: A Survey. *ACM Computing Surveys*, Vol 38, No.3, Article 9 (2006).
- GiurcaA., M. Tylkowski, M. Muller. Rule The Web!: Rule-based Adaptive User Experience. 2012. *Proceedings of the RuleML2012@ECAI Challenge*, at the 6th International Symposium on Rules, Montpellier, France, August 27th-29th, 2012, *CEUR Workshop Proceedings*, Vol-874 (2012)
- Guy N. Zwerdling, I. Ronen, D. Carmel, and E. Uziel. Social media recommendation based on people and tags. In *SIGIR*, pages 194–201, 2010.
- Han J.,Schmidtke H.,Xie X.,Woo W. Adaptive content recommendation for mobile users: Ordering recommendations using a hierarchical context model with granularity, In *Pervasive and Mobile Computing*, Volume 13, 2014, Pages 85-98, ISSN 1574-1192.
- He J. and W. W. Chu. A social network-based recommender system. Technical Report 090014, Computer Science Department, UCLA,2009.
- Horrocks I., Patel-Schneider Peter F., Boley Harold, Tabet Said, Groszof Benjamin, Dean Mike, "SWRL: A Semantic Web Rule Language Combining OWL and RuleML", W3C Member Submission 21 May 2004, <http://www.w3.org/Submission/SWRL/>.
- Hu T., J. Luo, H. Kautz, A. Sadilek. Home location inference from sparse and noisy data: Models and applications. *International Conference on Data Mining Workshop (ICDMW)*, pp. 1382-1387, 2015.
- Isinkaye F., Y. Folajimi, and B. Ojokoh. Recommendation systems: Principles, methods and evaluation. *Egyptian Informatics Journal*, 16(3):261--273, 2015.
- Ke Ji, Hong Shen, Addressing cold-start: Scalable recommendation with tags and keywords, *Knowledge-Based Systems*, Volume 83, July 2015, Pages 42-50.
- Ken McGarry, A survey of interestingness measures for knowledge discovery, *The Knowledge Engineering Review*, v.20 n.1, p.39-61, March 2005.
- Keßler C. 2010, A RESTful SWRL Rule Editor. RR, volume 6333 of *Lecture Notes in Computer Science*, pp. 235-238.
- Kumbhare T.A., Santosh V. Chobe. An Overview of Association Rule Mining Algorithms. *International Journal of Computer Science and Information Technologies*, vol. 5, no. 1, pp. 927-930, 2014.
- Le Hoang Son. Dealing with the new user cold-start problem in recommender systems: A comparative review. *Information Systems*, Volume 58, June 2016, pages 87-104
- Leung C.W., S.C. Chan, F.L. Chung, An empirical study of a cross-level association rule mining approach to cold-start recommendations, *Knowledge Based Systems* 21 (7) (2008) 515-529.
- Levi A., Mokryn O., Diot C. and Nina Taft. 2012. Finding a needle in a haystack of reviews: cold start context-based hotel recommender system. In *Proceedings of the sixth ACM conference on Recommender systems (RecSys '12)*. ACM, New York, NY, USA, 115-122.
- Lika B., K. Kolomvatsos, S. Hadjiefthymiades, Facing the cold start problem in recommender systems, *Expert Systems with Applications*, Vol 41, Issue 4, Part 2, March 2014, pp 2065-2073.

- LinW.: Association Rule Mining for Collaborative Recommender Systems. Master's Thesis, Worcester Polytechnic Institute, May 2000.
- Lü L., M. Medo, C.H. Yeung, Y.-C. Zhang, Z.-K. Zhang, T. Zhou, Recommender systems, *Phys. Rep.* 519 (1) (2012) 1–49.
- Mahapatra S., Tareen, A. and Yang, Y., A Cold Start Recommendation System Using Item Correlation and User Similarity. Vancouver, 2011.
- Manisha Girotra, Kanika Nagpal Saloniinocha Neha Sharma Comparative Survey on Association Rule Mining Algorithms, *International Journal of Computer Applications* (0975 – 8887) Volume 84 – No 10, December 2013.
- Meng Chen, Cheng Yang, JiechaoChen,Peng Yi, “A Method to Solve Cold-Start Problem in Recommendation System based on Social Network Sub-community and Ontology Decision Model”, 3rd International Conference on Multimedia Technology ICMT, 2013.
- Min Fan and Zhu, William 2013. Cold-start recommendation through granular association rules. arXiv: 2013.1305-1372.
- Mohammad-Hosseini, Nadimi-Shahraki and MozhdBahadorpour. Cold-start Problem in Collaborative Recommender Systems: Efficient Methods Based on Ask-to-rate Technique”, *Journal of Computing and Information Technology-CTT* 22, 2014.
- Motik Boris, Grau Bernardo Cuenca, Horrocks Ian, Wu Zhe, FokoueAchille, Lutz Carsten, (Eds.), “OWL 2 Web Ontology Language Profiles”, 2nd Edition, W3C Recommendation 11 December 2012, <http://www.w3.org/TR/owl2-profiles>.
- Nadimi-ShahrakiM.H., Mozhdode B. Cold-start Problem in Collaborative Recommender Systems: Efficient Methods Based on Ask-to-rate Technique. *CIT. Journal of Computing and Information Technology* 22.2 (2014) pp. 105-113
- Nalmpantis O., Tjortjis C. (2017) The 50/50 Recommender: A Method Incorporating Personality into Movie Recommender Systems. In: Boracchi G., Iliadis L., Jayne C., Likas A. (eds) *Engineering Applications of Neural Networks. EANN 2017. Communications in Computer and Information Science*, vol 744. Springer, Cham
- Nilashi, M., Ibrahim O., Bagherifard, K. A recommender system based on collaborative filtering using ontology and dimensionality reduction techniques, In *Expert Systems with Applications*, Volume 92, 2018, Pages 507-520, ISSN 0957-4174,
- Patkos T., Bikakis A., Antoniou G., Plexousakis D., Papadopouli M.A semantics-based framework for context-aware services: lessons learned and challenges. In: *Proceedings of 4th international conference on Ubiquitous intelligence and computing (UIC-2007)*, Vol. 4611 of LNCS, Springer, pp 839–848(2007)
- Peng F., Lu, X., Ma, C. et al. *International Journal of Machine Learning and Cybernetics*. pp. 1-14 (2017)
- Poirier D, Fessant F, Tellier I.: Reducing the cold-start problem in content recommendation through opinion classification. In: *Proceedings of the 2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology - Volume 01*, Toronto, Canada, IEEE Computer Society, WI-IAT'10, pp. 204–207 (2010a)
- Pooja R. Harne. Mining of Association Rule- A Review Paper. *International Journal of Engineering Technology, Management and Applied Sciences*, April 2015, Vol. 3 Issue 4.
- Portugal I., Alencar P., Cowan D. The use of machine learning algorithms in recommender systems: A systematic review. *Expert Systems with Applications*. Volume 97, May 2018, pp. 205-227 (2018).
- Qiankun Z., “Association Rule Mining:A Survey, Technical Report”,CAIS, Nanyang Technological University, Singapore, 2003.
- Rafsanjani A., Salim N., Aghdam A., Fard K. Recommendation Systems: a Review, *IJCER* Vol. 3, Issue 5, May 2013 pp. 47-52
- Raigoza J., Karande, V. (2017). A Study and Implementation of a Movie Recommendation System in a Cloud-based Environment. *International Journal of Grid and High Performance Computing (IJGHPC)*, 9(1), 25-36.
- Rashid A.M, Karypis G.,Riedl J., Learning preferences of new users in recommender systems: an information theoretic approach, in: *ACM SIGKDD Explorations Newsletter*, vol. 10, issue 2, 2008, pp. 90–100.
- Rong Yu, Xiao Wen, and Hong Cheng. A monte carlo algorithm for cold start recommendation. In *WWW*, pages 327–336, 2014.
- Sackett, D.L. Why randomized controlled trials fail but needn't”, *Canadian Medical Association Journal*, Vol. 165 No. 9, pp. 1226-37 (2001).
- Safoury L, Salah A (2013) Exploiting user demographic attributes for solving coldstart problem in recommender system. *Lecture Notes on Software Engineering* 1(3):303–307
- Sahebi, S., Cohen, W.: Community-based recommendations: a solution to the cold start problem. In: *RSWEB*, in Conjunction with ACM RecSys 2011 (2011)
- Schein, A. I., A. Popescul, L. H. Ungar, and D. M. Pennock. Methods and metrics for cold-start recommendations. In *Proc. of the 25th Annual Intl. ACM SIGIR Conf.*, 2002.
- Sedhain S., S. Sanner, D. Brazianus, L. Xie, J. Christensen, "Social collaborative filtering for cold-start recommendations.", *Proc. ACM Conf. Recommender Syst.*, pp. 345-348, 2014.
- Serrano D., R. Hervás, J. Bravo. 2011. Telemaco: Context-aware System for Tourism Guiding based on Web 3.0 Technology. *International Workshop on Contextual Computing and Ambient Intelligence in Tourism*.
- Shaw G., Yue Xu, and Geva S. Using association rules to solve the cold-start problem in recommender systems. In *Proceedings of the 14th Pacific-Asia conference on Advances in Knowledge Discovery and Data Mining - Volume Part I (PAKDD'10)*, 340-347, 2010.
- SlimaniT., Lazzez A. Efficient Analysis of Pattern and Association Rule Mining Approaches. *International Journal of Information Technology and Computer Science*,2014, Vol 03, pp. 70-81.
- Sobhanam S. and Mariappan A.K. A Hybrid Approach to Solve Cold Start Problem in Recommender Systems using Association Rules and Clustering Technique. *International Journal of Computer Applications* 74(4):17-23, July 2013.
- Son H. Dealing with the new user cold-start problem in recommender systems: A comparative review, *Information Systems*, Vol 58, June 2016, pp 87-104.
- Tamir R., Y. Singer., 2006. On a Confidence Gain Measure for Association Rule Discovery and Scoring., *The VLDB Journal*, 15(1), Pp. 40–52.
- Tanna P. et al. “Using Apriori with WEKA for Frequent Pattern Mining”, *International Journal of Engineering Trends and Technology (IJETT)* – Volume 12 Number 3 - Jun 2014.
- Tian H., LiangP. Personalized Service Recommendation Based on Trust Relationship. *Scientific Programming*, vol. 2017, Article ID 4106134, 8 pages, 2017.
- Tintarev N., Masthoff J. (2015) Explaining Recommendations: Design and Evaluation. In: Ricci F., Rokach L., Shapira B. (eds) *Recommender Systems Handbook*. Springer, Boston, MA.
- Verbert K., Manouselis N., Ochoa X., Wolpers M., Drachsler H., Bosnic I., Duval E. (2012) Context-aware recommender

- systems for learning: a survey and future challenges. *IEEE Trans Learn Technol* 5(4):318–335.
- Viktoratos I., Tsadiras A., Bassiliades N. Using Rules to Develop a Personalized and Social Location Information System for the Semantic Web. *RuleML 2014*: 82-96.
- Viktoratos I., A. Tsadiras, N. Bassiliades, Modeling human daily preferences through a context-aware web-mapping system using semantic technologies, *Pervasive and Mobile Computing*, Vol. 38, Part 1, July 2017, pp. 14-40.
- Viktoratos I., A. Tsadiras, N. Bassiliades: A context-aware web-mapping system for group-targeted offers using semantic technologies. *Expert Syst. Appl.* 42(9): 4443-4459 (2015).
- Weigand H., W. J. Heuvel, and M. Hiel, Rule-based service composition and service-oriented business rule management. *ReMoD*, France (2008).
- Xu J., Yao Y., Tong H., Tao X., & Lu, J. (2015). Ice-Breaking: Mitigating cold-start recommendation problem by rating comparison. In *IJCAI International Joint Conference on Artificial Intelligence*. (Vol. 2015-January, pp. 3981-3987). International Joint Conferences on Artificial Intelligence.
- Yang, D., D. Zhang, V. W. Zheng, Z. Yu. Modeling User Activity Preference by Leveraging User Spatial Temporal Characteristics in LBSNs. *IEEE Trans. on Systems, Man, and Cybernetics: Systems, (TSMC)*, 45(1), 129-142, 2015.
- Ye M. et al. On the semantic annotation of places in location-based social networks. In *Proc. 17th Int. Conf. ACM SIGKDD Knowledge Discovery and Data Mining*, 520–528 (2011)
- Zhang Z. K., Liu C., Zhang Y.C., Zhou T. Solving the cold-start problem in recommender systems with social tags. *EPL*, 2010, 92(2): 28002.