



HAL
open science

Situation Assessment for Non-Intrusive Recommendation

Imen Akermi, Rim Faiz

► **To cite this version:**

Imen Akermi, Rim Faiz. Situation Assessment for Non-Intrusive Recommendation. 12th IEEE International Conference on Research Challenges in Information Science (RCIS 2018), May 2018, Nantes, France. pp.1-12. hal-02319695

HAL Id: hal-02319695

<https://hal.science/hal-02319695>

Submitted on 18 Oct 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Open Archive Toulouse Archive Ouverte

OATAO is an open access repository that collects the work of Toulouse researchers and makes it freely available over the web where possible

This is an author's version published in:
<http://oatao.univ-toulouse.fr/22401>

Official URL

DOI : <https://doi.org/10.1109/RCIS.2018.8406670>

To cite this version: Akermi, Imen and Faiz, Rim *Situation Assessment for Non-Intrusive Recommendation*. (2018) In: 12th IEEE International Conference on Research Challenges in Information Science (RCIS 2018), 29 May 2018 - 31 May 2018 (Nantes, France).

Any correspondence concerning this service should be sent to the repository administrator: tech-oatao@listes-diff.inp-toulouse.fr

Situation Assessment for Non-Intrusive Recommendation

Imen Akermi
IRIT Laboratory
University of Toulouse 3
France
Email: imen.akermi@irit.fr

Rim Faiz
LARODEC Laboratory
University of Carthage
Tunisia
Email: rim.faiz@ihec.rnu.tn

Abstract—With the rapid growth of mobile applications, the user is increasingly confronted with a lot of information and tend to reject notifications sent by applications installed within his/her mobile device. This rejection affects the performance of many systems, especially proactive recommender systems. Therefore, it is no longer enough for a recommender system to determine what to recommend according to users' needs, but it also has to deal with the risk of disturbing the user during the recommendation process. We believe that the several embedded applications within the user's device along with other parameters could help understand and assess the user's interruptibility in some situations.

In this paper, we address intrusiveness within a proactive recommendation approach that makes use of the user's context and the applications embedded within the user's mobile device in order to assess the intrusiveness level of a given situation before recommending.

I. INTRODUCTION

With the development of web platforms and new technologies, the interest in recommender systems has significantly increased and has spread to cover multiple domains such as movies¹, tourism² and videos³.

Traditional recommender systems aim at providing relevant information to users. With the recent spread of mobile devices (smartphones and tablets), we notice that recommender systems are progressively adapting to pervasive environments in order to deliver not only relevant information to users but also when it is most needed. Indeed the amount of the contextual information provided by the mobile devices sensors such as temperature, GPS, accelerometer, etc. help to understand the users' needs and deliver recommendations without the user's request. This is called context-aware proactive (just-in-time) recommendation or zero-query search.

Proactive or Just-In-Time recommender systems involve all systems able to provide recommendations tailored to the preferences and needs of users in order to help them access useful and interesting resources within a large volume of data. The user does not need to formulate a query, this is implicit and corresponds to the resources that match the user's interests at the right time. However, despite the relevance of

the personalized information delivered to the user, he/she may choose to reject recommendations in certain situations. This abstinence may not concern the recommended information itself but it takes part in the situation the user may be in and during which the user does not want to be disturbed. Thus, it is important to include the risk of disturbing the user within the recommendation process. This is called risk-aware recommendation.

The works proposed in [1], [2], [3], [4] considered this aspect from a user modelling perspective and considered that intrusiveness is limited to figuring out implicitly the user's preferences and related information. As it comes to the works [5], [6] that integrated intrusiveness into the recommendation process, they only relied on the user's agenda activities to assess if they can send a recommendation or not.

In this paper, we propose an approach for assessing intrusiveness within a proactive recommendation approach, not only in terms of the user's agenda activities but also including the user's context with its several level of representation and other applications embedded in the user's mobile device. Indeed, we believe that this kind of contextual information could help understand the situations in which recommendations are subject to rejection.

The main contributions of this paper can be summarized as follows:

- A model for intrusiveness assessment within the recommendation process that makes use of the applications and the sensors embedded within the user's mobile device along with contextual information.
- An extensive user study evaluation for intrusive recommendation assessment.

To our knowledge, there is no existing empirical research that addresses intrusive recommendation in a mobile environment the way we tackle it in this paper.

The paper is organised as follows. We provide in section 2 a literature review about proactive recommender systems and risk-aware recommendation. Section 3 details the proposed approach. We describe in section 4 the experiments conducted and we finish in section 5 with a conclusion and thoughts for future work.

¹Netflix <https://www.netflix.com/>

²Tripadvisor <https://www.tripadvisor.com/>

³Youtube <https://www.youtube.com/>

II. RELATED WORK

We present, in this section, an overview on the proactive recommendation domain and the concepts that it entails. We also introduce the intrusiveness aspect and how it was addressed in literature.

A. Proactive recommender systems

Proactive Recommendation Systems (PRSs) as described by [2], retrieve large quantities of documents, decide what available information is most likely relevant to the user needs, and offer that information without the user's request. Ricci [7] considers that proactive recommender systems "can revolutionize the role of RSs from topic oriented information seeking and decision making tools to information discovery and entertaining companions".

The use of contextual information, particularly, in a mobile environment, is very crucial to boost the performance of such systems. The concept of context has been addressed in many works and has been defined through different aspects [8], [9], [10], [11], [12], [13], [14]. The most commonly and widely used definition for context presented it as the cognitive, the social and the professional environment which cover situations related to factors such as location, time and the current application [15], [16], [17], [18].

There are several approaches that used location as an approximation of context. The Global Positioning System (GPS) integrated or installed in the device helps to define the user's location. This location is displayed, according to latitude and longitude. Those GPS coordinates are not the only features that we can consider when defining a location.

Indeed, as discussed by [19], there are different ways to characterize the location of the mobile user :

- Absolute position
- Relative (next to, ...)
- A Place name
- A named class that represents the type of the place, eg. museum, school,

The place type can also be recovered using a Geographical Information System (GIS) such as geonames⁴ or foursquare⁵ which assigns a location category (restaurant, train station, etc.) to a given GPS coordinates. The localization accuracy helps to determine the user's context in a more precise manner [20].

Time was also used as a context dimension that helps to boost the recommendation relevance. It may be represented as a continuous variable whose values determines the specific time period at which items are rated by a given user.

Example: user A rated item I at $t = \text{June 1st, 2010 at 18:05:00}$

Another way to model time is to identify categorical values, for the time periods of interest. For example, in the tourism domain, the variable "season" can be expressed as:

$season = \{hot_season, cold_season\}$.

⁴<http://www.geonames.org/>

⁵<https://www.foursquare.com/>

Time can also be modelled in a hierarchical way which makes possible to define the degree of granularity of the time context information. Example:

$Week_{Day} = \{Monday, Tuesday, \dots, Sunday\} \Rightarrow time = \{morning, afternoon, \dots, night\}$

The user's activity may also be used as a contextual information within the recommendation process. Chen et al. [21] described the user's activity through three different schemes:

- Machine vision: using image processing and camera technology
- The user's calendar: to figure out the different activities scheduled at a certain time
- Artificial Intelligence techniques: that help to determine contextual information by leveraging low-level sensors.

The user activity may be determined from the different application and sensors installed in the mobile device such as the camera, the accelerometer, or the microphone. The data provided by the sensors can be saved in context logs in the device or sent to the server.

All of these contextual information describe user intentions and needs and constitute important factors for relevant proactive recommendations. Indeed, most of the systems that have been developed for proactive recommendation relied on contextual triggers to initiate the recommendation process. These systems can be partitioned into the following typology.

1) *Spatio-Temporal based systems*: The recommender systems that rely mainly on the spatio-temporal factors focus generally on a specific domain like tourism or restaurants recommendation.

Opperman et al. [22] developed a system called HIPPIE that proactively recommend to users upcoming events and exhibits within a tourist user guide using indoor positioning technologies and maps.

The work presented in [23], proposed a proactive recommender system for points of interests (POI) employing mainly time and the user's visiting history of POI. The latter factor was also used in [24] within a Markov chain model to predict the user's next visits. Vico et al. [25] made use of other contextual factors like the social dimension (user alone or accompanied), besides the temporal and the geographical aspects, to proactively recommend restaurants to a user.

Tong et al. [26] proposed a proactive approach for next purchase basket recommendation. They considered this approach as a binary classification problem (*buy or not*) in which features are mainly based on time and location.

2) *The user's current or past behaviour based systems*: In order to proactively recommend items to users, various approaches depend on various factors, to mention :

- The user's past or actual behavior history that includes for example previous visiting behaviors for location based systems [24], [3];
- Web browsing history/clicks [2]
- Previous reading patterns for news recommender systems [27], [28], [29]

Sae-Ueng et al. [30] analysed the user's behavior log for shopping assistance using a digital camera and RFID sensors⁶. The system recommended information about a product according to the user's behavior classified under five states: *Standing*, *Viewing*, *Touching*, *Carrying*, and *Fitting*. Elbery et al. [31] developed a carpooling recommender system that makes use of the user's past visiting history and information collected from the user's social networks accounts. The information collected is then used in a time markov chain. In [32], the authors used the users' behaviour patterns extracted from social networks to develop a personalized recommender system for e-government services.

3) *Activity-centric systems*: Other approaches considered recommendation triggers from an activity centric angle. The triggers might take the form of:

- Ongoing conversation or activity such as text messages, phone calls [33]
- Opened web pages or documents [34], [35], [36]
- The social media activity of the user such as the content of the user's tweet stream on Twitter⁷ [37], [38]

Morales et al. [37] developed a recommendation approach to suggest interesting news to users by exploiting the information in their twitter persona. They model relevance between users and news articles using a mix of signals drawn from the news stream and from twitter. This latter is used to build the profile of the social neighbourhood of the users, the topic popularity in the news and the content of their own tweet stream. They showed that the combination of microblogging platforms and real-time web signals can be interesting triggers to send notifications to users.

Phelan et al. [39] presented a news recommendation system named Buzzer, which is able to send recommendations about articles according to the conversations that are taking place on Twitter. The system uses a content-based approach by mining trending terms from both the public Twitter timeline and from the timeline of tweets generated by a user's own social graph (friends and followers). The system also looks for terms co-occurrences within the tweets and the RSS articles. Therefore, during recommendation, the articles that match the recent Twitter content will be recommended.

B. Risk-aware recommendation

The Cambridge Dictionary⁸ defines intrusiveness as an act:

"Affecting someone in a way that annoys them and makes them feel uncomfortable."

Intrusiveness was also defined in [40] as :

"A perception of psychological consequence that occurs when an audience's cognitive processes are interrupted."

The intrusiveness concept was tackled within different applications that attempted to put forward an approach for detecting it. In [41], intrusiveness or interruptibility as the authors preferred

to call it, is measured using the likelihood of the user to respond to phone calls computed using sensors embedded within the user's mobile device. These sensors were able to detect the user's proximity regarding the device :

The user holds the device

The device is close to the user's head

Siewiorek et al. [42] engineered an application that adjusts the device ring tone according to the user's surroundings inferred from the microphone, the light and the accelerometer. Alcalá et al. [43] proposed a non-intrusive application for monitoring human activity for health care using a smart meter data. The system is able to collect data implicitly, monitors the user's behaviour pattern and sends notifications when it detects behaviour anomalies.

In [44], intrusiveness is perceived as an interruption that should be avoided when a user is in a particular emotional state which is determined by a pedometer and a heart rate monitor. The authors assumed that the user's should not be interrupted or disturbed when the system detects that he/she is "*stressed*" or "*angry*".

When it comes to the recommender system domain, intrusiveness was considered in [5] as a risk of disturbing the user and was defined as:

"the possibility to disturb or to upset the user which leads to a bad answer of the user".

Several works addressed this aspect as a user modelling issue and considered that a non-intrusive recommendation approach is an approach that can implicitly figure out the users' preferences and related information [45], [46], [47].

In the following sections, we present the two types of approaches that addressed intrusiveness within RSs.

1) *Non-intrusiveness as implicit user profiling*: This concept was considered by several works from an implicit user profiling perspective assuming that non intrusiveness is keeping track of the user's preferences implicitly.

Lin [1] described the recommender system he proposed as non-intrusive as it estimates the user's preferences from the time the user spends in a shop without explicitly asking the user.

Melguizo et al. [2] used the text that was currently written by the user to recommend items that are relevant to the text that was written. They perceive this kind of recommendation approach as proactive and non-intrusive as it supports authors in the writing task without asking for their involvement.

Pu et al. [3] designed a location based recommendation system to provide the most possible interesting places to a user according to her/his implicit preference and physical moving location without the user providing her/his preference or a query explicitly. They proposed two circle concepts, physical position circle that represents spatial area around the user and virtual preference circle that is a non-spatial area related to the user's interests extracted from her/his historical visiting behaviour.

Quercia et al. [4] proposed a system that automatically recommends new friends, tracks the health of friendships and

⁶https://en.wikipedia.org/wiki/Radiofrequency_identification

⁷<https://twitter.com/>

⁸<http://dictionary.cambridge.org/>

gains awareness of the user's mood by monitoring implicitly his/her activity with mobile phones including monitoring text messages, phone calls and encounters captured by Bluetooth. They have engineered a new technology for mobile phones that silently keeps track of people's colocation, as well as frequency of voice calls and text messages. They also explored the degree to which the engine can predict users' moods (e.g., happiness, sadness) simply based on their activity. The proposed framework called FriendSensing enables new members of social-networking websites to automatically discover their friends. It also helps existing members to elicit new social relations, as they develop over time. These services uses short-range radio technologies (e.g., Bluetooth) for logging encounters and rely on colocation records to elicit relevant encounters and to arrange them into a weighted social network for recommending friends.

Unfortunately, the above mentioned works and several others dealt with the intrusiveness concept as a matter of retrieving the user's interests and not regarding the fact that the recommendation itself may disturb the user. Indeed, intrusiveness can also be an issue within the recommendation process and not only in the user's modelling process.

2) *Non-intrusiveness as non-disturbing recommendation:*

The work presented in [5] was the first to integrate intrusiveness as a phase in which they assess the risk of disturbing the user before recommending. They consider a situation as a triplet composed of *location*, *time* and *the user's agenda activity*. They define a "critical" or "risky" situation as a situation in which a user does not want to be disturbed. For each situation, they compute a risk score that depends on the risk-level of the concept describing the user's activity extracted from his/her agenda. They assume that a situation is deemed risky if its risk score exceeds a pre-defined threshold. This approach is specifically designed for employees working at a company and do not consider other type of users.

Bedi et al. [6] integrated a situation assessment phase in their approach for recommending restaurants, in which they use fuzzy logic as an inference technique that depends on *distance*, *time*, *budget* and *reachability*, to assess the non-intrusiveness level (also called the context level) of a given situation. They predefine the fuzzy sets for the context level and for the attribute *Distance* as follows :

Distance={Near, Moderate, Far}
Context-level={Low, Medium, High}

The link between the context attributes and the context-level is represented as rules; which means that the context level is inferred depending on the values of the attributes.

Example:

```
IF (Distance IS 'Near') AND
(Time IS 'In-Time') AND
(Budget IS 'Affordable') AND
(Reachability IS 'High')
THEN Context-level IS 'High'
```

The work proposed in [48] considered intrusiveness in a recommendation approach as a classification problem which aims at identifying whether a given context is "good" or "bad" to trigger the recommendation process. They collected mobile data over a three weeks user study in order to learn the classification model.

To sum up, even when some works tried to deal with the intrusiveness issue, they always tend to look at the surroundings of the user forgetting that the big amount of applications embedded in the user's phone could be the issue itself and can help figure out if recommending in a given situation is appropriate or not. Thus, in this paper, we propose to assess intrusiveness not only in terms of context as generally defined by time and location, but also considering the applications that a user is using at a given situation.

III. MEASURING INTRUSIVENESS FOR PROACTIVE RECOMMENDATION

We propose to integrate an intrusiveness assessment phase into a context-aware proactive recommendation approach that covers multiple domains [49]. It aims at recommending relevant items that match a user's situation without waiting for the user to initiate any interaction. The recommendation process is not launched until we assess the intrusiveness level of the situation.

We consider that the user's daily routine is represented as a pack of situations that reflects a specific category of interest described by the the spatio-temporal dimensions' instantiations and the user's actual activity.

A situation is characterized by four dimensions: *time of the day*, *the weekday*, *the actual location* and *the user's activity* presented respectively as: $S = (D_t, D_w, D_l, D_a)$.

To overcome the cold start problem when first using the recommender system, we predefine for typical situations, a particular category of information C to recommend (Restaurants, News, Traffic information, etc).

For example, the situation "Lunch time" is typically described by :

D_l : At work;
 D_w : Monday;
 D_t : $t \in [12 : 00, 14 : 00]$;
 D_a : the user is taking a break

For such situation, the category of information to recommend and that suits the best is "Restaurant". Therefore, we consider that a situation, with its different levels of representation, defines the changing user's need in information.

Then, the type of information needed for an actual user situation is updated according to the user's feedbacks to past situations. In this paper, regardless of the type of information to recommend in a given situation (that we addressed in a previous article [49]), we propose an approach for balancing the process of recommendation against intrusive interruptions. Indeed, there are different factors that make the user less open to recommendations and as we are working within the framework of mobile devices, we consider that the

several embedded applications in a mobile phone such as the camera, the keyboard, the accelerometer, agenda, etc. are good representatives of the user's interaction with her/his device since they somehow stand for the most undertaken activities in a mobile device such as texting messages, chatting, tweeting, browsing or taking selfies and pictures. Indeed, according to a recent study⁹, 85% of smartphone users spend more than 4 hours a day texting, surfing, talking and tweeting. Besides, 90% of the people surveyed reported using their smartphones to take pictures at least once a week. Thus, we believe that we should take into account the applications that are enabled at a given situation to figure out the user's activity. We opted for a case-based reasoning approach based on the analogous use of past cases to figure out if we could interrupt the user's current activity and send a recommendation. A user's past case is modelled as $case(premise, value)$:

- PREMISE : describes the situation $S_i = \{week_day, time_of_the_day, current_activity\}$ described by the instantiated dimensions that it entails. The premise is used to measure the similarity between the cases
- VALUE : integrates 2 parameters, $value(nb_y, nb_n)$ that refer respectively to the number of times the user agreed to receive a recommendation at situation S_i and the number of times the user rejected the recommendation at situation S_i without reading the content of the recommendation. This is to ensure that the rejection is not induced by the recommended content but rather by the user's situation.

Let S_c be the current user's situation and S the set of past situations stored in the recommendation feedback database. The system compares S_c with the situations in S in order to figure out the feedback that was given to a similar situation S_p :

$$S^* = argmax_{S_p \in S} sim(S_c, S_p) \quad (1)$$

The following sections detail the different phases of the intrusiveness assessment process:

- Retrieval: describes the matching process between the actual situation and the past ones.
- Re-use: examines the retrieved past situation's feedback in order to figure out the user's eventual response for a given recommendation.
- Revision: recovers the current user's feedback regarding a recommendation.

A. Retrieval

The similarity between two situations takes into account the similarities between the situations' features:

$$sim(S_c, S_p) = \sum_i \alpha_i sim(F_c^i, F_p^i) \quad (2)$$

⁹<https://www.comscore.com/Insights/Blog/Mobile-Matures-as-the-Cross-Platform-Era-Emerges>

Where :

F_c^i represents the i^{th} feature of the situation vector S_c (respectively S_p) and $\sum_i \alpha_i = 1$.

The following sections show how we calculate the respective similarities.

1) *Time feature similarity*: The similarity of the time feature takes into account two levels : time of the day and the week day:

- The week day

Assuming that the user lives in a Western country, the weekdays can be partitioned as following:

$$D_w \in \{work_days\{monday, \dots, friday\}, rest_days\{saturday, sunday, public_holiday\}\}$$

This partition is automatically changed according to the user's location. Indeed, while Saturdays and Sundays may be rest days in most Western countries, this is not the case for Middle-Eastern countries, where Friday is typically a rest day and Sunday is not.

We sequentially enumerate the week days (1 for monday, ..., 7 for sunday) in order to compute the similarity between two week days in terms of proximity as:

$$sim(D_w^c, D_w^p) = 1 - \frac{|D_w^c - D_w^p|}{nd} \quad (3)$$

Where nd stands for the number of the week days, which is 7.

- Time of the day

We choose to divide a daily routine into four periods (*morning, midday, afternoon* and *evening*) that are framed within 24 hours intervals.

$$D_t \in \{morning[07 : 00, 12 : 00], midday[12 : 00, 14 : 00], afternoon[14 : 00, 18 : 00], evening[18 : 00, 00 : 00]\}$$

In order to calculate the similarity between two time intervals, we rank each period from 1 (*morning*) to 4 (*evening*):

$$sim(D_t^c, D_t^p) = 1 - \frac{|D_t^c - D_t^p|}{np} \quad (4)$$

Where np stands for the number of the time periods defined, which is 4.

2) *The user's activity similarity*: At a given situation S , the system takes a snapshot of the user's current activity A_c by checking the agenda activities and the current enabled application such as driving, texting messages, tweeting or browsing, using the sensors and the applications embedded in the user's mobile device. For example, we can figure out if the user is in a meeting according to his agenda or if the user is taking a picture by checking if the camera is enabled or not.

Thus the similarity computation of the user's activity related to two situations is computed as:

$$sim(A_c^c, A_c^p) = \begin{cases} 1 & \text{if } A_c^c = A_c^p \\ \frac{2 \times d3}{d1 + d2 + 2 \times d3} & \text{else} \end{cases} \quad (5)$$

In order to overcome the drawback of syntactic similarity (perfectly matching words or phrases) [50], we compute the Wu and Palmer [51] semantic similarity of the two activities defined as the shortest path between two concepts within the Wordnet¹⁰ lexical graph, where:

$d1$ is the depth of A_c^c

$d2$ is the depth of A_c^p

$d3$ is the depth of the *least common subsumer* (LSC) which stands for the closest ancestor concept to the two activities.

B. Re-use

Once we retrieve the most similar situation to the current one, we use the past user’s feedback in order to decide whether we should send a recommendation or not. If, for the similar situation, the number of times the user disregarded the notification (nb_n) exceeds the number of times the user agreed to receive a notification (nb_y), we would take that as a “do not disturb me” feedback. We also consider the cold-start problem that arises when there is no similar situation among the past ones. When this occurs, we assume that the recommendation will not disturb the user and that we can push the recommendation into the user’s device screen.

C. Revise

The revision phase consists of recovering the user’s feedback regarding the recommendation related to the current situation.

The user’s click on the recommended information is considered as a “POSITIVE” feedback, meaning that the notification did not disturb the user. If the user chose to disregard the recommendation by swiping the notification displayed on the device’s screen, we take that as a “NEGATIVE” feedback.

The new feedback of the actual situation may serve for the construction of a new case or the update of an existing one depending on the similarity score that was previously computed. If the similarity score between a past case and the user’s actual situation exceeds a threshold $\lambda = 0,6$ (indicating that the two situations have at least two features’ values that match perfectly), we accordingly update nb_y or nb_n within the value section (i.e. feedback) of the similar situation. Otherwise, the current case along with its actual feedback will be added to the case base:

Algorithm 1: The revision process

```

if ( $Sim(S_c, S_p) \geq \lambda$ ) then
  if  $feedback_{S_c} = \text{“POSITIVE”}$  then
     $nb_y^{S_p} = nb_y^{S_p} + 1$ 
  else
     $nb_n^{S_p} = nb_n^{S_p} + 1$ 
  end if
else
  Add  $S_c$  to the case base
end if

```

IV. EXPERIMENTS

Since there is not a suitable dataset to experiment the approach we propose, we conducted a user study. Indeed, user studies [52] are good alternatives for evaluating recommender systems in which users are asked to evaluate recommendations. This kind of evaluation allows a subjective assessment of the system as surveys can be conducted along with the experiments.

A. User Study

We automatically generated situations that simulate real life situations the user might be in and that are characterized by four features : *the week day*, *time of the day*, *the current activity* the user might be doing and *the category of information* that might be recommended (News, POI, restaurant, gift idea, TV program, etc.). The possible values of the first three features were determined using a survey conducted within the IRIT lab¹¹ with colleagues from different backgrounds and age range. The values set of the fourth feature was addressed in [49]. We filtered the set of situations generated to take out those that are not likely to happen, for example, *having a meeting at late night at home*. We also made sure that the values gathered for the *current_activity* feature cover most of the activities that can be inferred from the applications and sensors installed within the device. We settled for 100 situations. Users were asked, given a situation they might be in, if they accept to get a recommendation or not. They were also asked to mention if they consider the information type (News, POI, ...) recommended at that situation as relevant or not. They also had the possibility to comment on every situation.

For example, a situation can be described to the user as:

It is **Saturday, Midday** and you are doing the following activity : **Taking a picture/selfie**

Would you accept to get a notification :

- YES
- NO

Given this situation, do you think that recommending *Restaurants* is interesting :

- YES
- NO

Comments: (Please comment your answers)

We used the crowdflower¹² platform to run the user study. In order to avoid any bias, we configured several quality control mechanisms such as speed traps which measure the time spent by a participant to answer the questions of the study. We also made sure that the participants understand perfectly English and the question they were asked. Figure 1 gives an overview about the conducted user study.

¹¹<https://www.irit.fr/>

¹²<https://www.crowdfunder.com/>

¹⁰<https://wordnet.princeton.edu/>

Getting Recommendations

Instructions ▾

Given a situation you might be in and which is characterized by the day of the week, the time of the day and the current activity you might be doing. You have to choose whether you accept or not to be interrupted and get a notification on your mobile phone about a particular information.

You might also comment on the type of information you got as a notification at this situation. For example, if you choose to accept to get a notification at a given situation, you can assess by a comment whether you liked the type of information that was recommended to you or if you would like to get something else.

It is Saturday, Midday and you are doing the following activity: Taking a picture/selfie

Would you accept to get a notification:

- YES
 NO

Given this situation, do you think that recommending this type of information is interesting: Restaurant

is this type of information interesting?

- YES
 NO

Comments

It is Saturday, Afternoon and you are doing the following activity: Walking

Would you accept to get a notification:

- YES
 NO

Given this situation, do you think that recommending this type of information is interesting: Points of interests(Park, Museum...), Concert Theaters Program, Movie Theaters Program

is this type of information interesting?

- YES
 NO

Comments

Fig. 1. The user study overview

B. Results

The purpose of this study was to gather real users' judgments about situations that might occur in real life. Thus, after parsing the collected data, we got about 1500 users who participated to this study. In this paper, we only considered the first section of this study addressing the issue of accepting to receive a recommendation or not, regardless of its content. The approach used to tackle the second section, that considers the type of information to recommend automatically according to the user's situation, is presented in [49].

In order to determine the accuracy of the approach in terms of intrusiveness assessment, we consider a cross-validation evaluation that estimates the reliability of a model based on a sampling technique. We run a K-fold cross-validation test ($K = 10$) that consists of partitioning, for each user, a sample data that is used as a training set and then use the remaining data for testing. This process is repeated for each user K times. Then, we calculated the Mean Average Precision (MAP) for every possible feature combination and for the two baselines.

$$MAP = \frac{\sum_{u=1}^U AveP(u)}{U} \quad (6)$$

$$AveP(u) = \frac{\sum_{k=1}^K rel_s}{K} \quad (7)$$

Where :

U is the number of users

rel_s is the number of correctly assessed situations for each run

K is the number of runs ($K = 10$)

S is the number of situations

Figure 3 illustrates the obtained results.

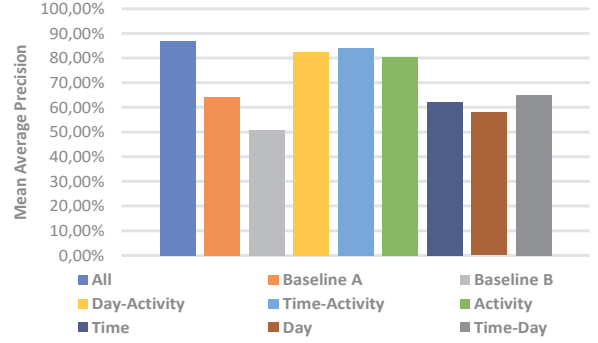


Fig. 2. The recommendation accuracy using intrusiveness assessment

As shown in Figure 2, the proposed approach, using all the features equally distributed ($\alpha = 1$, see Eq. 2), scores a MAP of 87% against 64% for Baseline A that always sends recommendations without considering the user's interruptibility and 50.81% for Baseline B that consists in not sending a recommendation when an application is ON.

We also note that the combinations that entail the activity feature, like *Activity-Day*, *Activity-Time* and *Activity*, score a high precision.

Even when we varied the weights α_i assigned to each feature, the user's current activity still takes over the other features to determine the intrusiveness level of a situation (see table I). Then, we can consider that the activity feature is a discriminative attribute for deciding whether a situation is conducive to receive a recommendation or not.

TABLE I
VARYING WEIGHTS ASSIGNED TO THE SITUATION FEATURES

| $\alpha_{activity}$ | α_{time} | α_{day} | precision |
|---------------------|-----------------|----------------|-----------|
| 0,0 | 0,0 | 1,0 | 0,65 |
| 0,0 | 0,2 | 0,8 | 0,62 |
| 0,0 | 0,5 | 0,5 | 0,62 |
| 0,0 | 0,7 | 0,3 | 0,63 |
| 0,0 | 1,0 | 0,0 | 0,67 |
| 0,2 | 0,0 | 0,8 | 0,71 |
| 0,2 | 0,2 | 0,6 | 0,74 |
| 0,2 | 0,5 | 0,3 | 0,75 |
| 0,2 | 0,7 | 0,1 | 0,73 |
| 0,5 | 0,0 | 0,5 | 0,84 |
| 0,5 | 0,2 | 0,3 | 0,85 |
| 0,5 | 0,5 | 0,0 | 0,84 |
| 0,7 | 0,0 | 0,3 | 0,84 |
| 0,7 | 0,2 | 0,1 | 0,86 |

As we explained earlier, the category of information to recommend (News, coffee shop, POI, ...) is inferred according to the user's situation [49]. Therefore, we also used this study to put forward the topical relevance of the recommended information regarding the situations that were proposed. For each situation, we measured the average precision score computed as the proportion of users who rated the recommended information, according to the given situation, as relevant:

$$AveP(u) = \frac{\sum_{s=1}^S nb_{u_rel}}{S} \quad (8)$$

Where S is the number of situations, nb_{u_rel} is the number of users who judged a situation s as relevant and U is the total number of users.

The approach scored 85% for topic relevance accuracy with an inter-agreement coefficient equal to 0,76.

C. Analysis

The performance of Baseline B, which considers that a recommendation should not be sent when an application is ON, proves that approaches that automatically consider the use of any random application at a given situation as a hinder to sending a recommendation, are not effective. It actually depends on the type of the application being used and on the user's behaviour. Indeed, given the user study data, we analysed the users' responses and behaviours regarding recommendations according to time and activity. As shown in Figure 3, we computed the proportion of users who considered recommendations, in certain activities, as annoying or not. We only put forward the 5 most used applications in a mobile device.

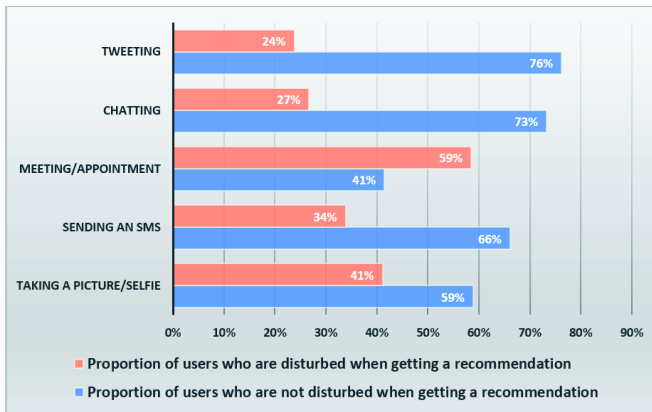


Fig. 3. The users' behaviour regarding some activities

We note that more than 70% of the participants accepted to receive recommendations when tweeting or chatting. This could be explained by the fact that people may want to share with others the recommended information. We also notice that 59% of the participants against 41% were not disturbed when getting a recommendation while taking a picture which could be somehow interpreted as senseless because we normally expect users to get annoyed if they were interrupted while

typing a message or using the device's camera. Actually, it depends on the user's preference and behaviour pattern. That is why the case-based reasoning approach we propose to address the intrusiveness aspect is revealed to be efficient since it considers every user apart and does not follow a typical trend. We also studied the user acceptance regarding receiving notifications according to the time and day of the week.

As expected and as illustrated by Figure 4, the two most important peaks to observe happen during breaks and after work. Indeed, it is during these two periods of the day that people have more spare time to spend for activities other than work and chores.

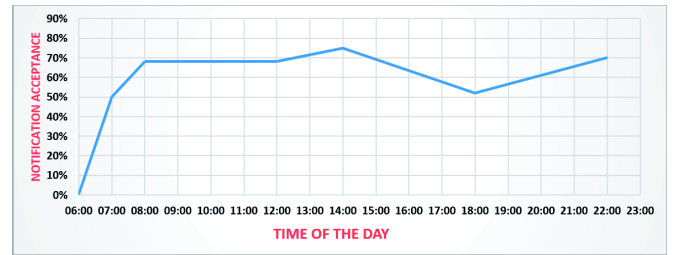


Fig. 4. The notification acceptance rate according to the time of the day

Figure 5 shows that the notification acceptance rate follows an escalating pattern starting from the beginning of the week.

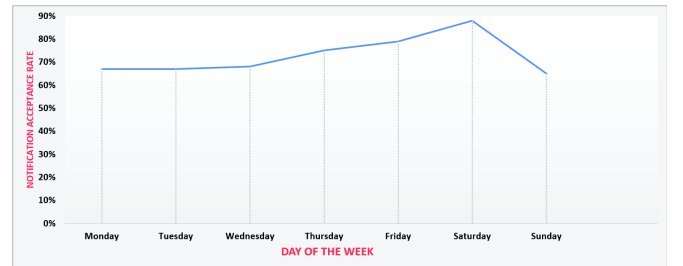


Fig. 5. The notification acceptance rate according to the day of the week

People tend to be more receptive to suggestions at the week-end.

The user study that we conducted entails a lot of information that can be used for recommender system's evaluation. Indeed, we made this user study available¹³ for the RS research community as a dataset for proactive and context-aware RS evaluation. This can help alleviate the datasets shortage and provide a framework for different approaches to be compared on a same basis.

The purpose of this study is to investigate, considering a user's situation, whether any recommendation should be sent at all, regardless of its content. However, we believe that the content is still important to determine whether a recommended item is disturbing or not.

For example, a user may not want to be disturbed usually when working but perhaps work related news is still acceptable.

¹³contact the authors

Therefore, we are currently working on integrating into the approach we proposed, a trade-off between the importance of the information to be recommended and the risk of disturbing the user. Indeed, in some situations, even though the user chose not to be disturbed, the recommended information might be worth being interrupted for, such as breaking news or an accident that happened on the user's way home. We believe that such trade-off needs to be studied.

V. APPLICATION SCENARIO

The recommendation approach that we propose has been developed by an IT company¹⁴ within the framework of a project funded by the European Union. The Implemented application is a proactive and non-intrusive recommender system that enables users to get relevant recommendations according to their current situations. The application covers multiple domain item recommendation and is tailored to the users' preferences extracted from their Facebook accounts [53] and from their behaviour pattern (browsing history and clicks on the recommended items).

The mobile application is developed within a client/server model and it is deployed on the server part. The user only gets the visible and the interactive parts of the application on his/her mobile device. According to a time trigger installed on the user's device, an implicit request including contextual information is sent to the server to be analysed in order to assess the user's situation for intrusiveness and launch the recommendation of the appropriate information [49]. The following figures gives an overview about the application (the language used in the application is french as it is going to be launched in a french speaking market).

The link provided allows the user to log into his/her facebook account in order to create his/her user profile for a personalized application usage. Once the user gives permission to access his/her likes, the application collects the required information.

As we explained in the previous sections, according to given situations, the application implicitly initiate the recommendation process after assessing the user's interruptibility and displays a notification icon entailing a brief description about the recommended information in the notification bar of the mobile device.

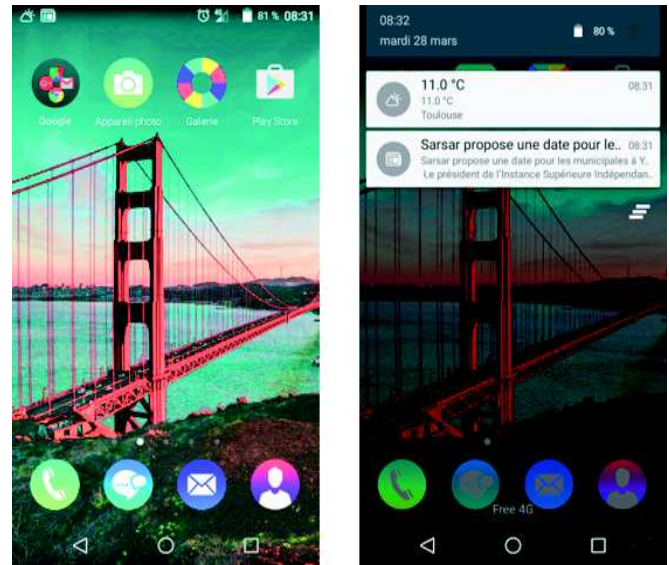


Fig. 7. Notification

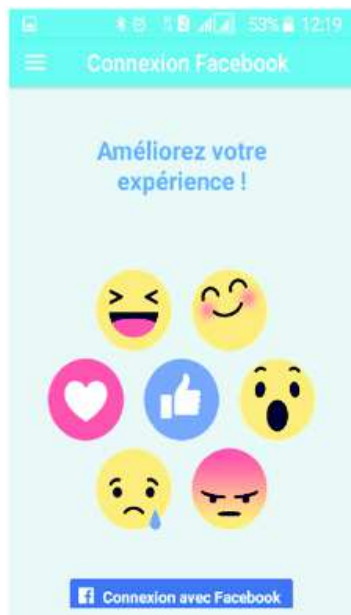


Fig. 6. Connecting on Facebook

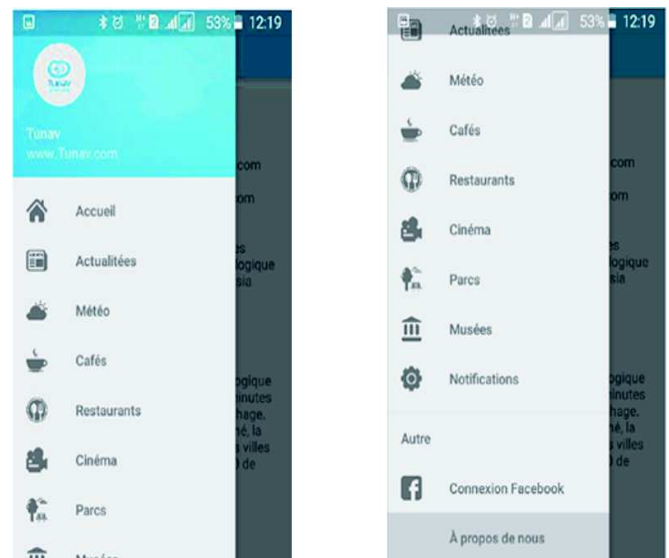


Fig. 8. Menu

¹⁴<http://www.tunav.com>

The main purpose of the application is to provide proactive information to the user without disturbing her/him, besides allowing to check for information manually in case the user does not want to wait for an implicit and proactive recommendation (see figure 8). The user has also the possibility of switching on/off notifications about a given information category.

Information gathering and the evaluation of this application on a real life basis, regarding intrusiveness assessment is still ongoing.

VI. CONCLUSION

In this paper, we introduced an approach for assessing intrusiveness within a proactive recommendation approach. The approach entails a case-based reasoning process that makes use of the user's surroundings and the applications embedded within the user's mobile device in order to assess intrusiveness before recommending. The experiments that we have conducted using a user study yielded promising results. Besides we constructed an evaluation framework based on a user study that we made available for the scientific community and that can be used to assess context-aware proactive recommender systems effectiveness.

REFERENCES

- [1] Z. Lin, "Indoor location-based recommender system," Ph.D. dissertation, University of Toronto, 2013.
- [2] M. C. P. Melguizo, T. Bogers, A. Deshpande, L. Boves, and A. van den Bosch, "What a proactive recommendation system needs - relevance, non-intrusiveness, and a new long-term memory," in *Proceedings of the Ninth International Conference on Enterprise Information Systems*, 2007, pp. 86–91.
- [3] Q. Pu, A. Lbath, and D. He, "Location based recommendation for mobile users using language model and skyline query," *International Journal of Information Technology & Computer Science (IJITCS)*, vol. 4, no. 10, pp. 19–28, 2012.
- [4] D. Quercia, J. Ellis, and L. Capra, "Nurturing social networks using mobile phones," *IEEE Pervasive Computing*, vol. 9, no. 3, pp. 12–20, 2010.
- [5] D. Bouneffouf, A. Bouzeghoub, and A. L. Gançarski, "A contextual-bandit algorithm for mobile context-aware recommender system," in *International Conference on Neural Information Processing*. Springer, 2012, pp. 324–331.
- [6] P. Bedi and S. Agarwal, "A situation-aware proactive recommender system," in *12th International Conference on Hybrid Intelligent Systems (HIS)*, 2012, pp. 85–89.
- [7] F. Ricci, "Mobile recommender systems," *Information Technology & Tourism*, vol. 12, no. 3, pp. 205–231, 2010.
- [8] B. N. Schilit and M. M. Theimer, "Disseminating active map information to mobile hosts," *IEEE network*, vol. 8, no. 5, pp. 22–32, 1994.
- [9] P. J. Brown, J. D. Bovey, and X. Chen, "Context-aware applications: from the laboratory to the marketplace," *IEEE personal communications*, vol. 4, no. 5, pp. 58–64, 1997.
- [10] N. Ryan, J. Pascoe, and D. Morse, "Enhanced reality fieldwork: the context aware archaeological assistant," *Computer Applications in Archaeology*, vol. 750, pp. 269–274, 1999.
- [11] A. K. Dey, "Context-aware computing: The cyberdesk project," in *Proceedings of the AAAI 1998 Spring Symposium on Intelligent Environments*, 1998, pp. 51–54.
- [12] B. Allen, "Information needs: A person-in-situation approach," in *Proceedings of an International Conference on Information Seeking in Context*, ser. ISIC '96, 1997, pp. 111–122.
- [13] D. H. Sonnenwald, "Exploring the contexts of information behaviour," T. D. Wilson and D. K. Allen, Eds. London, UK: Taylor Graham Publishing, 1999, ch. Perspectives of Human Information Behaviour: Contexts, Situations, Social Networks and Information Horizons, pp. 176–190.
- [14] C. Cool, "The concept of situation in information science," *Annual review of information science and technology*, vol. 35, pp. 5–42, 2001.
- [15] S. Lawrence, "Context in web search," *IEEE Data Engineering Bulletin*, vol. 23, pp. 25–32, 2000.
- [16] L. M. Quiroga and J. Mostafa, "An experiment in building profiles in information filtering: The role of context of user relevance feedback," *Inf. Process. Manage.*, vol. 38, no. 5, pp. 671–694, 2002.
- [17] P. Ingwersen and K. Järvelin, "Information retrieval in context: Irix," in *ACM SIGIR Forum*, vol. 39, no. 2. ACM, 2005, pp. 31–39.
- [18] J.-C. Bottraud, G. Bisson, and M.-F. Bruandet, "Expansion de requetes par apprentissage automatique dans un assistant pour la recherche d'information," in *CORIA*, 2004, pp. 89–108.
- [19] S. Dobson, "Leveraging the subtleties of location," in *Proceedings of the 2005 joint conference on Smart objects and ambient intelligence: innovative context-aware services: usages and technologies*. ACM, 2005, pp. 189–193.
- [20] U. Christoph, J. Von Stülpnagel, K.-H. Krempels, and C. Terwelp, "Context detection on mobile devices," in *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, vol. 15, 2010, p. 17.
- [21] G. Chen, D. Kotz *et al.*, "A survey of context-aware mobile computing research," Technical Report TR2000-381, Dept. of Computer Science, Dartmouth College, Tech. Rep., 2000.
- [22] R. Oppermann and M. Specht, "A context-sensitive nomadic exhibition guide," in *International Symposium on Handheld and Ubiquitous Computing*. Springer, 2000, pp. 127–142.
- [23] M. Braunhofer, F. Ricci, B. Lamche, and W. Wörndl, "A context-aware model for proactive recommender systems in the tourism domain," in *Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct*. ACM, 2015, pp. 1070–1075.
- [24] W. Li, C. Eickhoff, and A. P. de Vries, "Want a coffee?: Predicting users' trails," in *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 2012, pp. 1171–1172.
- [25] D. G. Vico, W. Woerndl, and R. Bader, "A study on proactive delivery of restaurant recommendations for android smartphones," in *ACM RecSys workshop on personalization in mobile applications, Chicago, USA*, 2011.
- [26] J. Tong, H. Ma, W. Liu, and B. Wang, "Time and location-based hybrid recommendation system," in *Big Data Analysis (ICBDA), 2017 IEEE 2nd International Conference on*. IEEE, 2017, pp. 677–683.
- [27] W. IJntema, F. Goossen, F. Frasinca, and F. Hogenboom, "Ontology-based news recommendation," in *Proceedings of the 2010 EDBT/ICDT Workshops*. ACM, 2010, pp. 1–6.
- [28] S. Athalye, "Recommendation system for news reader," Ph.D. dissertation, San Jose State University, 2013.
- [29] D. A. Dumitrescu and S. Santini, "Improving novelty in streaming recommendation using a context model," in *ACM RecSys Workshop on Context-Aware Recommender Systems*, 2012.
- [30] S. Sae-Ueng, S. Pinyapong, A. Ogino, and T. Kato, "Personalized shopping assistance service at ubiquitous shop space," in *Advanced Information Networking and Applications Workshop*. IEEE, 2008, pp. 838–843.
- [31] A. Elbery, M. ElNainay, and H. Rakha, "Proactive and reactive car-pooling recommendation system based on spatiotemporal and geosocial data," in *IEEE 12th International Conference on Wireless and Mobile Computing, Networking and Communications*. IEEE, 2016, pp. 1–8.
- [32] R. Ayachi, I. Boukhris, S. Mellouli, N. B. Amor, and Z. Elouedi, "Proactive and reactive e-government services recommendation," *Universal Access in the Information Society*, vol. 15, no. 4, pp. 681–697, 2016.
- [33] A. Popescu-Belis, M. Yazdani, A. Nanchen, and P. N. Garner, "A speech-based just-in-time retrieval system using semantic search," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Systems Demonstrations*. ACL, 2011, pp. 80–85.
- [34] P. Prekop and M. Burnett, "Activities, context and ubiquitous computing," *Comput. Commun.*, vol. 26, no. 11, pp. 1168–1176, 2003.
- [35] S. Dumais, E. Cutrell, R. Sarin, and E. Horvitz, "Implicit queries (iq) for contextualized search," in *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 2004, pp. 594–594.
- [36] M. Karkali, D. Pontikis, and M. Vazirgiannis, "Match the news: A firefox extension for real-time news recommendation," in *Proceedings*

of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, 2013, pp. 1117–1118.

- [37] G. De Francisci Morales, A. Gionis, and C. Lucchese, “From chatter to headlines: Harnessing the real-time web for personalized news recommendation,” in *Proceedings of the Fifth ACM International Conference on Web Search and Data Mining*. ACM, 2012, pp. 153–162.
- [38] S. O’Banion, L. Birnbaum, and K. Hammond, “Social media-driven news personalization,” in *Proceedings of the 4th ACM RecSys Workshop on Recommender Systems and the Social Web*. ACM, 2012, pp. 45–52.
- [39] O. Phelan, K. McCarthy, M. Bennett, and B. Smyth, “On using the real-time web for news recommendation & discovery,” in *Proceedings of the 20th International Conference Companion on World Wide Web*. New York, NY, USA: ACM, 2011, pp. 103–104.
- [40] H. Li, S. M. Edwards, and J.-H. Lee, “Measuring the intrusiveness of advertisements: Scale development and validation,” *Journal of advertising*, vol. 31, no. 2, pp. 37–47, 2002.
- [41] K. Hinckley and E. Horvitz, “Toward more sensitive mobile phones,” in *14th annual ACM symposium on User interface software and technology*. ACM, 2001, pp. 191–192.
- [42] D. P. Siewiorek, A. Smailagic, J. Furukawa, A. Krause, N. Moraveji, K. Reiger, J. Shaffer, and F. L. Wong, “Sensay: A context-aware mobile phone,” in *Seventh IEEE International Symposium on Wearable Computers*, vol. 3, 2003, p. 248.
- [43] J. M. Alcalá, J. Ureña, Á. Hernández, and D. Gualda, “Assessing human activity in elderly people using non-intrusive load monitoring,” *Sensors*, vol. 17, no. 2, p. 351, 2017.
- [44] K. K.-L. Liu, “A personal, mobile system for understanding stress and interruptions,” Ph.D. dissertation, Massachusetts Institute of Technology, 2004.
- [45] G. Adomavicius, N. Manouselis, and Y. Kwon, “Multi-criteria recommender systems,” in *Recommender systems handbook*. Springer, 2011, pp. 769–803.
- [46] T. Farinella, S. Bergamaschi, and L. Po, “A non-intrusive movie recommendation system,” in *OTM Confederated International Conferences “On the Move to Meaningful Internet Systems”*. Springer, 2012, pp. 736–751.
- [47] K. Palanivel and R. Sivakumar, “A study on implicit feedback in multicriteria e-commerce recommender system,” *Journal of Electronic Commerce Research*, vol. 11, no. 2, pp. 140–156, 2010.
- [48] N. Dali Betzalel, B. Shapira, and L. Rokach, “Please, not now!: A model for timing recommendations,” in *Proceedings of the 9th ACM Conference on Recommender Systems*. ACM, 2015, pp. 297–300.
- [49] I. Akermi, M. Boughanem, and R. Faiz, “Just-in-time recommendation approach within a mobile context,” in *2016 IEEE/WIC/ACM International Conference on Web Intelligence, WI 2016, Omaha, NE, USA, October 13-16, 2016*, 2016, pp. 636–639.
- [50] R. Ferreira, R. D. Lins, S. J. Simske, F. Freitas, and M. Riss, “Assessing sentence similarity through lexical, syntactic and semantic analysis,” *Computer Speech & Language*, vol. 39, pp. 1–28, 2016.
- [51] Z. Wu and M. Palmer, “Verbs semantics and lexical selection,” in *Proceedings of the 32Nd Annual Meeting on Association for Computational Linguistics*, ser. ACL ’94. Stroudsburg, PA, USA: Association for Computational Linguistics, 1994, pp. 133–138.
- [52] M. D. Ekstrand, F. M. Harper, M. C. Willemsen, and J. A. Konstan, “User perception of differences in recommender algorithms,” in *Proceedings of the 8th ACM Conference on Recommender systems*. ACM, 2014, pp. 161–168.
- [53] I. Akermi and R. Faiz, “Using Social Networks for Mobile Proactive Recommendation,” in *ACS/IEEE International Conference on Computer Systems and Applications (AICCSA), Doha, Qatar, 10/11/2014-14/11/2014*. IEEE Computer Society, 2014, pp. 755–760.

ACKNOWLEDGEMENT

The authors would like to thank the European Union program PASRI¹⁵ and Mr. Aymen ELJ, the Research and Development department manager in TUNAV¹⁶, for hosting this research and for his valuable support.

¹⁵<http://www.pasri.tn/>

¹⁶<http://www.tunav.com/fr>