PersonalWeb: An Extensible Framework to Recommend Web and Personal Information

João Guerreiro, Juliana Gomes, Daniel Gonçalves
UTL/INESC-ID
Rua Alves Redol, 9, 1000-029 Lisboa, Portugal
joao.p.guerreiro@ist.utl.pt, juliana.m.l.gomes@gmail.com, daniel.goncalves@inesc-id.pt

ABSTRACT
The large amount of information spread out among applications raised several challenges when trying to retrieve it, as information useful in the past is potentially useful in the future. Most recommender systems resort to contextual information to provide single-source document suggestions, disregarding the manifold Personal Information (PI) sources. We believe that PI can provide a richer background of the user's interests, providing personally-relevant and user-centered suggestions. In this paper, we describe an extensible framework that makes use of both contextual and personal information to provide recommendations that are relevant to the user and the task at hand, instead of only the latter. Herein we also present a preliminary evaluation of our framework still mostly based on contextual information. This pilot study presented satisfying results disambiguating contexts, suggesting web and personal documents and dealing with context changes, paving the way to its enrichment with more PI sources.

Author Keywords
Personal Information, Context-Awareness, Recommender Systems, Personally Relevant Results.

ACM Classification Keywords
H3.m. Information search and retrieval: Miscellaneous.

General Terms
Design, Experimentation, Human Factors.

INTRODUCTION
The increasing capacity of personal devices' storage along with the advent of the internet and its underlying services encourage users to maintain a growing amount of information. Our Personal Information (PI) is now scattered among several applications and *places*, such as the file system, online repositories, e-mail platforms, browser or social networks. In addition, users are reluctant in classifying their data as it is hard to predict its future value. We regularly misjudge the difficulty of re-finding it in the future and its value is often understood only then [10], neglecting that information useful in the past is potentially useful in the future [6]. Meanwhile, we have never had so many information at our disposal and available to support our tasks. However, searching on the web or personal space, people get distracted from their tasks from the moment they start searching, as unexpected search results may interrupt the users rather than help them completing their tasks [7].

There are many systems recommending task-related information. Several need user intervention [2, 8], which can be a distraction, and most recommend single-source documents (mainly web pages or personal documents). Context-awareness is vital to provide these suggestions, as it allows understanding the task at hand by resorting, for instance, to the web/personal documents opened or words written. However, considering it alone neglects the user's interests and previous interactions, providing only task-centered suggestions; though not user-centered, as their interests and needs may not be the same. Extreme cases are ambiguous words, such as *python* (snake vs programming language), but it goes beyond these cases as the suggestions relevance vary among users. For example, when talking with a specific friend, *John* always discusses *java* code-related issues. By being aware of previous interactions, it would suggest *java* related items instead of other programming languages. Resorting only to contextual information, it would depend on *java* being mentioned in this conversation to correctly direct the suggestions.

We believe that the enormous amount of PI scattered among applications can provide a richer background of the user's interests, providing user-centered results shaped to her/his needs. We also argue that PI sources should be part of a diverse set of suggestions, as users are not aware of the useful information they have at their disposal.

We built a standalone framework that considers the current context (what the user is doing, eg: words written, selected, documents/web pages opened) and the user's PI to provide user and task-centered recommendations, instead of only the latter. We performed a pilot study using our framework, yet with limited PI sources, but with the ambition to add more, making use of the framework extensibility. Results suggested good recommendations, supporting the use of PI in recommender systems, which motivated us to enrich our approach integrating more PI in the future.

CaRR 2012, February 14, 2012, Lisbon, Portugal.

RELATED WORK

There is an effort improving information retrieval by personalizing search results on demand [2, 8], however performing searches deviates the user from the current task as they need explicit input to present relevant information.

There are some systems that try to help the user by recommending documents proactively. Most of them resort to the current context to suggest web or personal documents related to the task. For example, *Watson* [1] resorts to the current document to suggest web documents that may be useful. *Tell Me More* [5] considers web and personal documents to provide additional information filtered by topics (quotes, actors, figures,..). While most approaches are limited in the information types they present, *TaskTracer* [3] identifies the current activity and present files, email messages or contacts that may be related to the task. Also, the information sources they consider to gather the context are limited as most of them only resort to the current documents. Even those resorting to other options focus only in the browser history [9].

We argue that resorting to the big amount of PI existent in our devices and web would present more diverse and user-oriented suggestions. It provides the opportunity to present more types of information, adding e-mail messages, instant messaging, contacts, posts, *tweets* to the previously used web and personal documents.

DESIGN AND RATIONALE

We built an extensible framework aiming at providing proactive user and task-related suggestions. We found crucial to provide flexibility at both ends: contextual sources to determine users' current activity; and PI sources used to direct the suggestions to users' interests and needs.

Gathering information from heterogeneous sources raises the inherent difficulty of managing it. We found a single representation to manage the information as a coherent whole instead of separate chunks, so the same information from different sources can be classified as such. With that it is possible to reinforce the information extracted from more than one source. With the most relevant *keywords* for the current context, we resort to Open Directory Project (ODP) to help disambiguating the context before sending it to the PI *Plugins*. These will be the ones providing personally relevant and user-centered suggestions by themselves and filtering the public ones, such as search engine results.

While context helps understanding what the user is doing at the moment, feeding this information to PI sources allows to naturally directing the results to what is relevant to that specific user (which may not be relevant to other users). In fact, we believe this integration between contextual and PI is the right way to provide task and user-centered results.

Framework Design

Our Framework is based in five main components: *Contextual Plugins*; *Plugin Manager*; *Data Coordinator*; *Retrieval Manager* and *Recommendation Plugins*. The *Contextual Plugins* are responsible for monitoring user activity and extracting the context from each active application and send it to the *Plugin Manager*. Its adapters convert the heterogeneous data into a common representation. The *Data Coordinator* weights the content to determine the most important attributes to store in a context database and removes the irrelevant ones. When needing suggestions, the *Retrieval Manager* requests the context (duration can be specified) where it basis the requests sent to the *Recommendation Plugins*, responsible to extract the context-related information.

Contextual Plugins

Most *Plugins* are application extensions able to extract the current context, which is the content and user's activity from a set of opened documents. This information includes the documents' entire text and some with special attributes (e.g. title, bold, selected or copied). At the moment, we have developed *Plugins* for *Mozilla Firefox*, *Mozilla Thunderbird*, *Office Excel*, *PowerPoint* and *Word*. We added two transversal *Plugins* to deal with additional information that the others cannot handle. One aims at counting the foreground windows duration and the other uses the *ContextLib* [4], a standalone library used to capture users' actions and that allowed us to be aware of the last written words, the contents in the clipboard and the items in the explorer (independent of the application).

Plugin Manager

This module contains an adapter to each *Plugin* which purpose is to convert the data to a single representation. Each *document* (independent of the *Plugin)* is converted to a representation containing the *id*, *date*, *type*, *duration (*opened*)* and *description* (the document text) and all the events that occurred on that document are associated to it using the *id*. Each event contains the *date*, the *documentId*, the *eventType* (e.g. selection, write) and the *text*. The information about the type and events allows us to weight differently the words. Likewise, we keep the records of the people related to each document (mainly the e-mail participants, document authors or comments).

Data Coordinator

This module removes irrelevant words (*stopwords)* and stores the information in the database. Then it assigns a weight to the terms collected based on term frequency (*tf*). Our corpus the set of contextual documents, so *tf* alone provides the most relevant terms for that corpus. If we considered the entire personal space, a measure like *tfidf* would probably be a better choice since it considers the term weight for each document regarding the entire corpus.

Retrieval Manager

Retrieval Manager requests the context of a specific period of time (e.g. last 2 minutes). We considered this configurable request so it was possible to deal with task

detection in the future or other similar feature that may benefit from context period specification. It extracts the concepts related to each one of the top ranked contextual keywords (from ODP). The most frequent concepts are the ones related to the actual context, which helps to narrow the context avoiding ambiguous results. Then these concepts and keywords are sent to PI *Recommendation Plugins*. The results from these *Plugins*, are then used, together with the queries, to help filtering information from public sources.

Recommendation Plugins

The *Recommendation Plugins* used resorted to *Bing*, for web pages and *Windows Search* for personal documents. These *Plugins* resort to the queries sent by the *Retrieval Manager* to extract the most relevant documents for that context (orderly). Adding *Plugins* is straightforward as resorting to the provided queries it is only necessary to add the necessary code to extract the relevant information to find context-related information.

Prototype

As a proof of concept prototype we developed a very simple standalone interface, which only concern was to present the recommended items (not aesthetics). The interface (Figure 1) consists of 2 panes to present personal and web documents. Some documents are presented with a thumbnail to ease the process of document recognition. Each document contains an image and a title, both clickable to open the document.

PILOT STUDY

We carried out a pilot study aimed at ascertaining if our framework could provide good recommendations resorting to personal and web documents, before introducing suggestions from other sources. At this phase, our main focuses were: disambiguation resorting to contextual information and ODP; the relevance of the documents suggested, to determine the quality of the queries sent to the *Recommendation Plugins*; and dealing with context changes without resorting to session boundary recognition.

Methodology

Firstly, we handed a questionnaire to the users that allowed us to create a profile (age, gender, educational attainment, profession) and perceive the usage they give to computer tools. Then the users had to perform three tasks, where they had to create a context and analyze the suggestions.

Task 1. An ambiguous specific subject (*python*, the snake) was previously defined and we provided a set of documents for the users to interact with. Then some web pages were suggested to the users by *PersonalWeb*. Users mentioned those that were related to the subject they interacted with.

Task 2. Participants had to choose a subject that they were used to work at and open a few documents related to that context. We asked them to mention documents that they were expecting to be suggested. Then we analyzed, with them, if the documents they mentioned were suggested and if there were other relevant ones besides those.

Task 3. Participants had to choose two different subjects. They navigated in those subjects sequentially and afterwards analyzed the suggestions to check if they were related with the second task only.

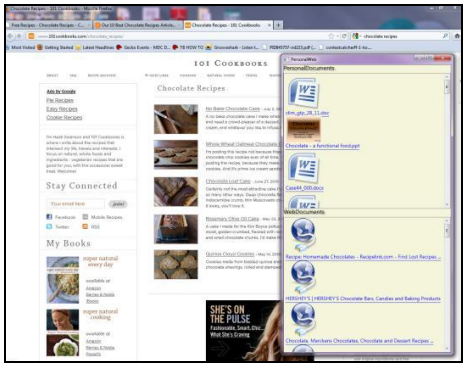

Figure 1. *Personal Web* recommending user and task related documents

Participants

We recruited 10 volunteers, mainly students from our university with ages between 18 and 25 (6); graduated between 26 and 40 (3); and one between 41 and 71.

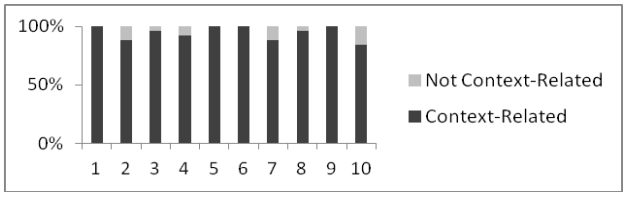

Figure 2. Percentage of Context-Related and Not Context-Related documents (total of 26) suggested, per user, in Task 1.

Results

We focused on a different aspect for each task; we wanted to assure that our framework was ready before adding more PI sources, instead of performing a full evaluation of our system. Our main goal with the first task was to acknowledge the success to disambiguate contexts resorting only to contextual information and ODP. We provided the documents used to create the context, so we focused the results only on web suggestions. In a total of 26 web pages suggested to each user, most of the suggestions (median=25) were found to be context-related (Figure 2).

In the second task, the context was from the users' workspace and they mentioned a set of documents they expected to be suggested. In this case, we focused on the personal documents' results. In half cases, all documents that the user was expecting were suggested (Figure 3) and in other two, more than a half (2 in 3 and 3 in 4).

It also presented relevant information that was not expected by the users (they tagged it as relevant in a post-test analysis). It reinforces the idea that we lose the notion of

the information we have on our personal computers. Figure 4 shows the total number of relevant and irrelevant documents suggested and overall the results were positive.

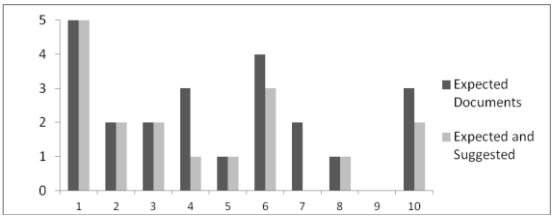

Figure 3. The number of *Expected Documents* and those that were *Expected and Suggested*, in task 2, for each user.

In task 3 our main goal was to verify if documents from the first context appeared on suggestions when working at the second one. That did not happen for 7 users but others had 1, 3 and 6 suggestions from the first context, mostly due to proximity among contexts (e.g. soccer and video games that shared some keywords, such as "*play"* and "*fun*").

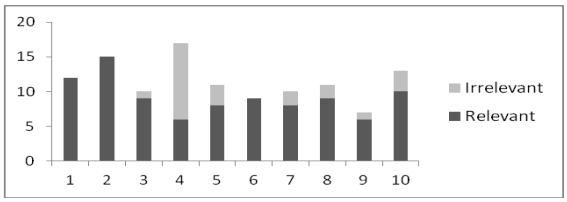

Figure 4. Total number of Relevant and Irrelevant documents in Task 2 suggestions, per user.

Discussion

These three tasks were a first impulse to our framework as it resulted in satisfying results. Task 1 showed its ability to deal with ambiguous results; Task 2 presented good personal document recommendations, either expected by the user or not; and Task 3 showed good results in dealing with context changes even without any concerns to session boundary recognition, as we believed that the context would be enough to make that distinction.

Most of the suggestions provided were considered relevant, which suggests that the queries sent to the *Recommendation Plugins* were successful. Adding more PI sources will provide more diverse and personally relevant results as web and personal documents are just a part of the information that may help the user to complete her/his task.

CONCLUSION

Information overload and fragmentation raised the big challenge of dealing with this great amount of data scattered among applications. At the same time, it also raised the opportunity to make use of all this information that was useful in the past and will probably be useful in the future. Most systems that try to recommend potentially useful information focus on contextual information and single-source recommendations, lacking the diversity and user-oriented suggestions. We built an extensible framework that is prepared to make use of the PI spread among application to provide user-centered instead of just contextual and task-related results. As a first step we evaluated a prototype with the basic functionalities, resorting to contextual information and providing recommendations of web and personal documents. Results suggested that the joint use contextual information with PI can help the user by recommending items both in line with the task at hand and with the user's background. These results motivated us to add a manifold set of PI sources (social networks, mobile data, mail messages, among others), which we believe will boost the framework results in a further and more exhaustive evaluation. As we were not concerned with the aesthetics in the first prototype, we will also focus our efforts on understanding how to make the suggestions without unnecessarily disturbing the user.

ACKNOWLEDGMENTS

João Guerreiro was supported by the Portuguese Foundation for Science and Technology, grant SFRH/BD/66550/2009.

REFERENCES

1. Budzik, J. and Hammond, K.J. User interactions with everyday applications as context for just-in-time information access. *In Proc of IUI, ACM* (2000).
2. Daoud, M. et al. A session based personalized search using an ontological user profile. *In Proc. of Symposium on Applied Computing, ACM Press* (2009).
3. Dragunov, A.N. et al. TaskTracer: a Desktop Environment to support multi-tasking knowledge workers. *In Proc. of IUI, ACM* (2005).
4. Barata, G. Blaze: Automating User Interaction in Graphical User Interfaces. *Master Thesis at Instituto Superior Técnico* (2009).
5. Iacobelli, F. et al. Tell Me More, not just "More of the Same". *In Proc. of IUI, ACM Press* (2010).
6. Jones, W. Finders, keepers? The present and future perfect in support of personal information management. *First Monday, 9* (2004).
7. Marshall, C.C. and Jones, W. Keeping encountered information, pages 66-67. ACM (2006).
8. Sieg, A., Mobasher, B., Lytinen, S. and Burke, R. Using concept hierarchies to enhance user queries in web-based information retrieval. *In Proc. of ICAIA* (2004).
9. Teevan, S. T. Dumais, and E. Horvitz. Personalizing Search via Automated Analysis of Interests and Activities. In Proc. of Research and Development in Information Retrieval, ACM Press, (2005).
10. Teevan, J. et al. How people find personal information. Personal Information Management, chapter 2, *University of Washington Press* (2007).