Mobile Clustering Engine

Abstract. Although mobile information retrieval is seen as the next frontier of the search market, the rendering of results on mobile devices is still unsatisfactory. We present CREDINO, a clustering engine for PDAs based on the theory of concept lattices that can help overcome some specific challenges posed by small-screen, narrow-band devices. CREDINO is probably the first clustering engine for mobile devices freely available for testing on the Web. An experimental evaluation, besides confirming that finding information is more difficult on a PDA than on a desktop computer, suggests that mobile clustering engine is more effective than mobile search engine.

1 Introduction

The diffusion of high performance mobile phones and PDAs, together with the increasing willingness of mobile users to turn to their portable devices to find web content, products and services, are creating a new market (e.g., [10, 11]). However, mobile search must still face a number of technical limitations present in such devices, such as small screen, limited user input functionalities, and high cost connection. The result is that Information Retrieval (IR) by means of commercial search engines such as Google (http://mobile.google.com/) may become a tedious, long, and expensive process for mobile users.

In this paper we tackle the problem of mobile IR using a clustering engine approach, which consists of grouping the results obtained in response to a query into a hierarchy of labeled clusters. This approach is well known, especially due to the popularity of Vivisimo, which won the "best meta-search engine award" assigned by SearchEngineWatch.com from 2001 to 2003. The advantages of the cluster hierarchy can be summarized as follows: it makes for shortcuts to the documents of interest, it displays potentially good terms for query refinement, and it provides a higher level view of the topic, which is particularly useful for unknown domains. An additional benefit is that it helps disambiguating polysemous queries.

It is arguable that the features of a clustering engine approach appear even more suitable for mobile IR, where a minimization of user actions (such as scrolling and typing), device resources, and the amount of data to be downloaded are primary concerns. Furthermore, such features seem to nicely comply with the recent changes in search behaviour, as observed in some recent user studies. For instance, according to [11], mobile users are more likely to enter shorter queries, less likely to scroll past the first few search results, both less able and less willing to access graphics-heavy web content. Despite such good potentials, however, the application of clustering engines to small mobile devices has not received much attention so far.

The first main objective of this paper is to help fill this gap. We build on CREDO, a clustering engine based on the theory of concept lattices described in [4, 5]. CREDO was developed for a desktop computer and does not scale to a small mobile device. We study which requirements must be met to extend desktop clustering engine to mobile search engine and then present CREDINO (small CREDO, in Italian), a version of the CREDO system for PDAs. CREDINO takes the cluster hierarchy produced in response to a query by CREDO and displays the cluster hierarchy on a PDA, handling the subsequent interaction with the user. CREDINO is available for testing at http://credino.dimi.uniud.it/. To the best of our knowledge, it is the first system of this kind on the Internet.

As a clustering engine offers a complementary view to the list of results returned by current search engines, it is interesting to compare the retrieval performance of the two approaches. Very few studies are available which do this for a desktop computer (one notable exception being [8]), let alone for mobile search. On the other hand, it is also useful to evaluate whether mobile IR is indeed less effective than desktop IR, in particular when using a search engine. This is one of the main hypotheses that motivate our research, although there is a lack of empirical observations.

The second main objective of this paper is to offer some insights into these somewhat overlooked issues. We compare the retrieval performance of CREDO and CREDINO to that of a conventional search engine on the respective device, through an experimental study involving external subjects searching a set of topics using the two retrieval methods on both devices. Our results suggest that mobile clustering engine can be faster and more accurate than mobile search engine, while confirming that mobile IR is less effective than desktop IR.

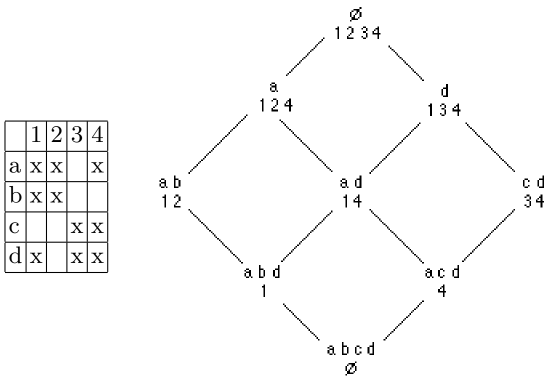
The rest of the paper is organized as follows. We begin by giving some background on concept lattices and CREDO, followed by a description of CREDINO. After discussing some related work, we turn to the experimental part, describing goals, design, and findings. Finally, the paper offers some conclusions and directions for future work.

2 Background: The concept lattice of Web results

Our approach is based on concept data analysis, which combines a strong mathematical background with a set of efficient manipulation algorithms [4]. Here we recapitulate its main characteristics for IR applications.

In essence, any collection of documents described by a set of terms can be turned into a set of concepts, where each concept is formed by a subset of terms (the concept intent) and a subset of documents (the concept extent). The intent and extent of any concept are such that the intent contains all terms shared by the documents in the extent, and the extent contains all documents that share the terms in the intent.

More formally, consider a binary relation I between a set of documents D and a set of terms T. We write dIt to mean that the document d has the term



	1	2	3	4
a	x	x		x
b	x	x		
c			x	x
d	x		x	x

Fig. 1. The concept lattice of a simple collection

t. For a set $X \subseteq T$ of terms and a set $Y \subseteq D$ of documents, we define:

$$X' = \{d \in D \mid dIt \ \forall t \in X\} \text{ and } Y' = \{t \in T \mid dIt \ \forall d \in Y\}.$$

A *concept* of (D, T, I) is a pair (X, Y) where

$$X \subseteq T, \ Y \subseteq D, \ X' = Y, \text{ and } Y' = X.$$

The set of concepts can be ordered by the standard set inclusion relation applied to the intent and extent that form each concept, i.e,

$$(X_1, Y_1) \leq (X_2, Y_2), \text{if } X_1 \supseteq X_2 \text{ (which is equivalent to } Y_1 \subseteq Y_2\text{)},$$

with the resulting lattice yielding a subconcept/superconcept relation. The bottom concept is defined by the set of all terms and contains no documents, the top concept contains all documents and is defined by their common terms (possibly none). As an illustration, Figure 1 shows a very simple bibliographic collection consisting of four documents (1, 2, 3, 4) described by four terms (a, b, c, d), with the corresponding concept lattice.

The document lattice (i.e., the concept lattice built from the given document-term relation) can thus be seen as a particular form of hierarchical conceptual clustering. Thanks to its mathematical properties, it supports various tasks of text analysis based on inter-document similarity, including query refinement, browsing retrieval, document ranking, and text mining [4]. Most relevant to this paper, this approach has been implemented in the CREDO clustering engine to organize and explore web retrieval results. Here we give a brief overview of the system, which is best described in [5].

CREDO forwards a user query to an external Web engine and collects the first 100 results. Then it extracts a set of terms for each result and builds the corresponding concept lattice, the levels of which are displayed on demand using a simple hierarchical representation. In order to keep the number of top concepts small, the first level of the lattice is built using a narrower set of term than those used to build the lower levels. CREDO can be tested at http://credo.fub.it.

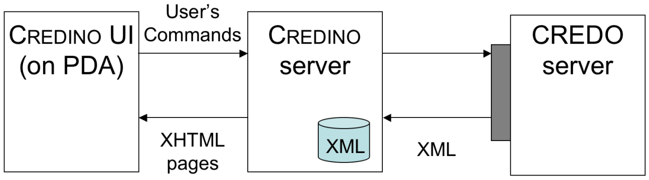


Fig. 2. Overall Credino architecture

3 Credino

Using a clustering engine approach for mobile search poses additional requirements compared to those which must be met in a desktop search. There are two main reasons why CREDO, like other clustering engines developed for a desktop computer, cannot be used on a PDA.

The first general requirement concerns usability and accessibility issues. Simply reproducing the frame-based CREDO interface would lead to a virtually unusable interface on a small screen, with an unacceptable amount of scrolling. In particular, users would be required to scroll horizontally, which is a very tedious way to see an entire CREDO screen.

A second constraint comes from bandwidth considerations. As the current implementation of CREDO is based on computing the whole hierarchy and sending out the results of all possible cluster selections at once, it is suitable for medium or broadband internet connections. By contrast, mobile devices connection to the Web usually has a low bandwidth (like GPRS) and is not free, the billing depending on the amount of data transmitted. Therefore it is important to choose, for Credino, an architecture that minimizes the amount of data transmitted to the mobile device.

The bandwidth constraint suggests to rely on an intermediate server (henceforth Credino server) to minimize both the amount of data sent to the mobile device and the load on the CREDO server (see Figure 2). The Credino server connects on one side to user's PDA and connects on the other side to (a slightly modified version of) the CREDO search engine.

Credino receives user's commands (query execution, cluster expansion, cluster contraction, visualization of the content of a cluster, visualization of a Web page) as HTTP requests from a PDA. In some cases (e.g., for a query execution command), Credino forwards the command (appropriately formatted into another HTTP request) to the CREDO server, which processes it and returns the result as an XML data structure. We purposefully developed an XML version of CREDO output (rather than relying on the HTML plus Javascript output produced by the standard version of CREDO) so as to facilitate subsequent processing by Credino. The result of a query, consisting of clusters and documents, is then locally stored by Credino. Thus, in some cases, Credino can directly execute user's commands without connecting again to CREDO; i.e., after a query, Credino can deal with all subsequent user actions until the next query is issued or the visualization of a web page is requested. Credino answers are returned to the Web browser on the PDA as XHTML page in the HTTP response. Credino and CREDO servers could run on the same computer, but in general they can be different (as in our current implementation).

Credino server is implemented as PHP scripts (plus domxml PHP module for XML parsing) on a Linux server running the Apache Web server. The Web pages sent to the PDA Web browser adhere to the XHTML 1.0 and CSS standards, thus Credino interface can be visualized on any PDA browser conforming to the W3C standards. In our experiment we have used the Pocket Internet Explorer Web browser running on a Windows Mobile 2003 iPAQ PocketPC 5450.

To complete the description of Credino, we present some snapshots of its user interface. Figure 3(a) shows Credino's home page with the query "tiger". Figure 3(b) and (c) show the first-level clusters displayed by Credino in response to the query "tiger". Like most of the words on the Web, "tiger" has multiple meanings; the clusters, in addition to refer to the animal, highlight several other meanings, including the golf champion ("woods"), the computer operating system ("mac os"), the race car ("racing"), the Boy Scouts of America ("cub"), etc. Note that the query box is at the bottom of the page, to save space for the first results. The user can expand a cluster into its sub-clusters by clicking on the "+" icon. In Figure 3(b) the cluster "software" is expanded into "mac os 10.4" and "other".

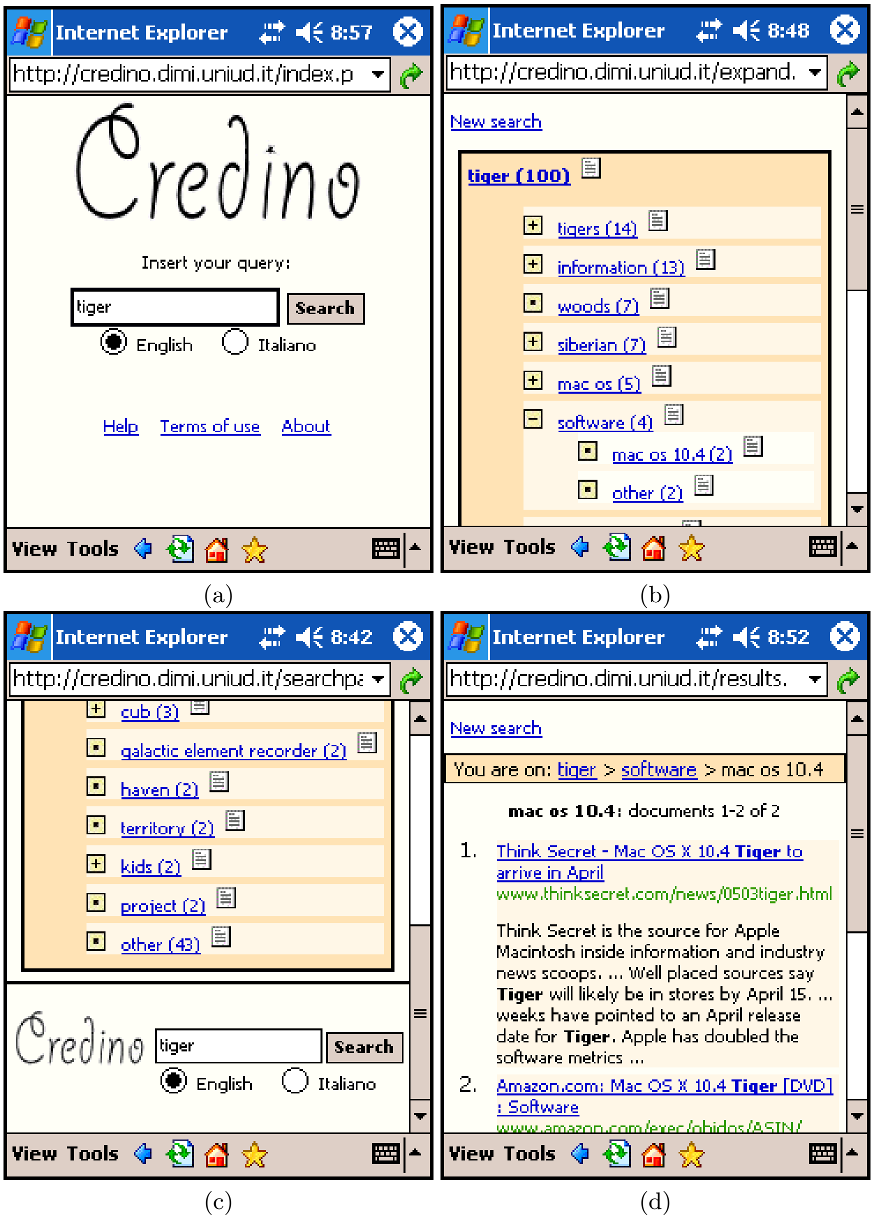
The user may also see the snippets of the documents contained in a cluster by clicking on the cluster name or on the icon on the right. Figure 3(d) shows the snippets of the documents associated with the selected sub-cluster. To provide information to users as to where they are located within the hierarchy, we use the breadcrumb trail metaphor; i.e., a sequence of clusters from the root to the current page. Path breadcrumb trails are dynamically displayed during the interaction between the user and the system, as shown in Figure 3(d) for the path: tiger > software > mac os 10.4.

4 Related Work

4.1 Clustering engines

Over the last few years, clustering engines have proved a viable alternative to conventional search engines. Following the popularity of Vivisimo, a bunch of industrial systems implement Web-snippet clustering: Mooter, Copernic, iBoogie, Kartoo, and Clusty, among others. This issue has also gained attention in the academic research field, although there are comparatively few implemented prototypes available on line (including CIIRarchies [13], SnakeT [8], and CREDO). Even major search engines such as Google and Yahoo! have recently shown a strong interest in this technology.

Most clustering engines employ a two-step procedure. Cluster labels are first generated by extracting short sequences of words (not necessarily contiguous)



(a)

(b)

(c)

You are on: tiger > software > mac os 10.4

mac os 10.4: documents 1-2 of 2

1. Think Secret - Mac OS X 10.4 **Tiger** to arrive in April
   www.thinksecret.com/news/0503tiger.html

   Think Secret is the source for Apple Macintosh inside information and industry news scoops. ... Well placed sources say **Tiger** will likely be in stores by April 15. ... weeks have pointed to an April release date for **Tiger**. Apple has doubled the software metrics ...
2. Amazon.com: Mac OS X 10.4 **Tiger** [DVD] : Software
   www.amazon.com/exec/obidos/ASIN/

(d)

Fig. 3. Credino's home page with the query "tiger" (a); Credino's clusters for "tiger" (b) and (c), with the cluster "software" expanded; snippets of the documents associated with the path: tiger > software > mac os 10.4 (d)

from the snippets, and then a cluster hierarchy is built with or from such labels. The systems differ in the choice of the lexical method used to extract the labels and in the algorithm for hierarchy construction. CREDO uses strict single-word indexing; however, it can easily produce multiple-word labels, reflecting the causal (or deterministic) associations between words in the given query context. For instance, for the query "tiger" (see Figure 3), CREDO returns some multiple-word concepts such as "mac os" and "galactic element recorder", reflecting the fact that, in the limited context represented by the results of "tiger", "mac" always co-occurs with "os", and "galactic" with "element" and "recorder".

A clustering engine based on concept lattices (this applies to both CREDO and Credino) presents some advantages over other clustering engines, which are more heuristic in nature, due to its reliance on a mathematical theory:

1) The clusters can be better justified, whereas the use of similarity metrics and heuristic decisions may result in the omission of clusters that are as plausible as those generated, or in a failure to include valuable clusters that are relatively rare. Figure 3, for instance, shows that even clusters formed by very few documents may easily appear at the top levels of the lattice. This can be useful to find unknown or less popular meanings of the given query; e.g., for the query "tiger", the concepts "tiger-cub" (related to the Boy Scouts of America), and "tiger-galactic element recorder" (related to a NASA program). Such meanings would probably go undetected by other clustering engines.

2) Cluster labeling is integrated with cluster formation, because a concept intent is univocally determined by a concept extent and viceversa. By contrast, cluster formation and cluster labeling are usually treated separately, thus implying that it may be difficult to find a good description for a given set of documents, or, symmetrically, a set of documents that fit a certain description.

3) The structure is a lattice instead of a tree, which facilitates recovery from bad decisions while exploring the hierarchy and can better adapt to the user. For instance, the document about the Mac OS Tiger software can be reached through two paths: *tiger* - *mac os* - *software* or *tiger* - *software* - *mac os*. Neither path is better than the other, but one may better fit a particular user's paradigm or need.

4.2 Search and Mobile Devices

Searching from small mobile devices has received quite a lot of attention, with dedicated publications and workshops in the main IR and human-computer interaction forums (see,e.g., [6, 7]). One of the central issues is the notion of search context, with its various forms: personalization and location are currently being exploited in a number of research and industrial efforts to filter and narrow retrieval results (see, e.g., [9]).

Other approaches have focussed on finding faster and more accurate decisions about the utility of the retrieved documents, using for instance keyphrases extracted from documents [15], summarization of Web pages and HTML forms [3], and related queries [12]. In [2], it is advocated that clustering helps to present information in a more dense and effective way on small devices, with a pilot study demonstrating small-screen access to a digital library system.

Speaking of mobile search, data organization and data visualization are obviously strictly interconnected. The use of specific data visualization schemes for small screens is discussed in [14], along with a set of guidelines to facilitate orientation and navigation, such as preferring vertical scrolling over page-to-page navigation and horizontal scrolling.

5 Experimental Evaluation

We have designed a comprehensive experiment aimed to evaluate how the choice of retrieval method (clustering engine or search engine) and device (PDA or desktop computer) affects retrieval performance. In particular, the main goals of our experiment are: (i) comparing the retrieval performance of clustering engine and search engine on both a PDA and a desktop computer, (ii) comparing the retrieval performance of PDA and desktop computer using both a clustering engine and a search engine approach.

We used 4 systems in the evaluation: 1) Credino (**PDA-Clustering**), 2) CREDO (**Desktop-Clustering**), 3) a mobile search engine obtained from Credino by switching off its clustering module (**PDA-Search**), 4) a desktop search engine obtained from CREDO by switching off its clustering module (**Desktop-Search**). We note by Δ followed by the initials of the two methods (devices) and the initial of one device (method) the differences in performance. For instance, ΔCS_D is the "Difference between Clustering and Search engine on Desktop computer". Figure 4 shows the scenario.

We tested 48 subjects in the experiment. They were computer science students or young faculty members at the University of Udine. As none of them was aware of CREDO and Credino systems before the experiment, and more than 90% of them were not users of any clustering engine, they were trained.

We used the four following topics, which represent various types of web searches (e.g., navigational, transactional, and informational) and are characterized by different levels of term ambiguity and difficulty:

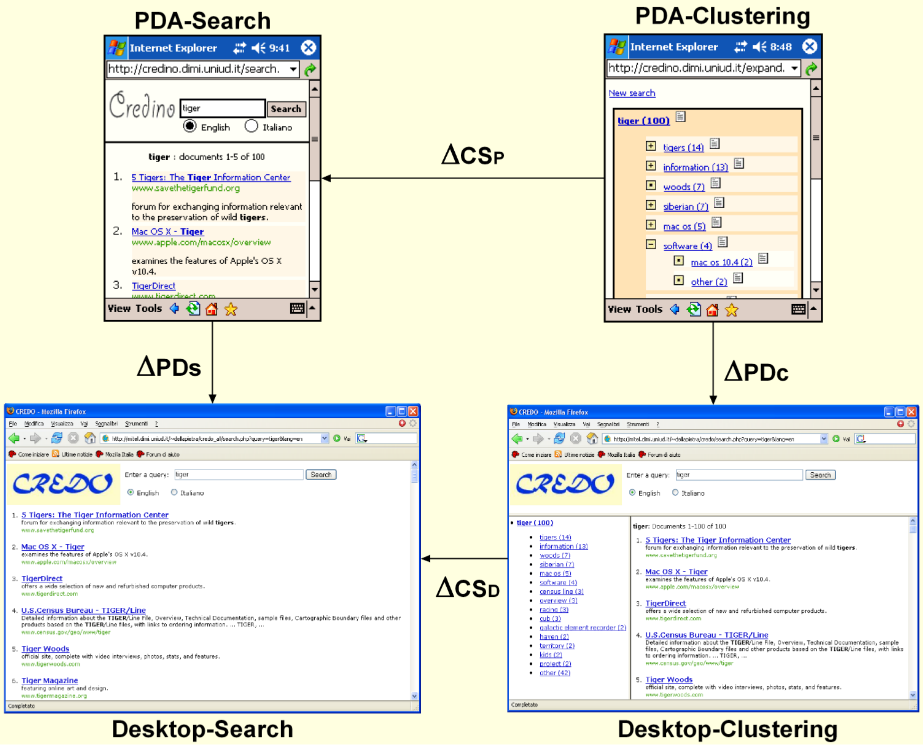
T1 "Your task is to find the Web site of the worldwide institution regulating the chess game".

T2 "Imagine you are a tourist going to visit Potenza.[1] You are interested in finding information about available accommodations, in particular you want to book a hotel room online".

T3 "You have to find a friend of yours which is on holidays in South Italy. You cannot reach him by phone, and you only know that he is in a place called 'Canasta' (and you do not know if it is a hotel, camping, village, etc.)."

T4 "Imagine that you have to find information concerning some athletes in the Boxing world. You are looking for an English-language web site that would allow you to search by name, weight, nationality, etc.".

[1] Potenza is an Italian city.



http://credino.dimi.uniud.it/search.

Credino tiger Search

● English ○ Italiano

tiger : documents 1-5 of 100

1. 5 Tigers: The Tiger Information Center
www.savethetigerfund.org
forum for exchanging information relevant to the preservation of wild **tigers**.

2. Mac OS X - Tiger
www.apple.com/macosx/overview
examines the features of Apple's OS X v10.4.

3. TigerDirect
www.tigerdirect.com

http://credino.dimi.uniud.it/expand.

New search

tiger (100)
- tigers (14)
- information (13)
- woods (7)
- siberian (7)
- mac os (5)
- software (4)
  - mac os 10.4 (2)
  - other (2)

http://mitel.dimi.uniud.it/~dellapietra/credo_all/search.php?query=tiger&lang=en

CREDO Enter a query: tiger Search

● English ○ Italiano

1. 5 Tigers: The Tiger Information Center
forum for exchanging information relevant to the preservation of wild **tigers**.
www.savethetigerfund.org

2. Mac OS X - Tiger
examines the features of Apple's OS X v10.4.
www.apple.com/macosx/overview

3. TigerDirect
offers a wide selection of new and refurbished computer products.
www.tigerdirect.com

4. U.S.Census Bureau - TIGER/Line
Detailed information about the **TIGER**/Line File, Overview, Technical Documentation, sample files, Cartographic Boundary files and other products based on the **TIGER**/Line files, with links to ordering information. ... TIGER, ...
www.census.gov/geo/www/tiger

5. Tiger Woods
official site, complete with video interviews, photos, stats, and features.
www.tigerwoods.com

6. Tiger Magazine
featuring online art and design.
www.tigermagazine.org

http://mitel.dimi.uniud.it/~dellapietra/credo/search.php?query=tiger&lang=en

CREDO Enter a query: tiger Search

● English ○ Italiano

- tiger (100)
  - tigers (14)
  - information (13)
  - woods (7)
  - siberian (7)
  - mac os (5)
  - software (4)
  - census line (3)
  - overview (3)
  - racing (3)
  - cub (3)
  - galactic element recorder (2)
  - haven (2)
  - territory (2)
  - kids (2)
  - project (2)
  - other (42)

tiger: Documents 1-100 of 100

1. 5 Tigers: The Tiger Information Center
forum for exchanging information relevant to the preservation of wild **tigers**.
www.savethetigerfund.org

2. Mac OS X - Tiger
examines the features of Apple's OS X v10.4.
www.apple.com/macosx/overview

3. TigerDirect
offers a wide selection of new and refurbished computer products.
www.tigerdirect.com

4. U.S.Census Bureau - TIGER/Line
Detailed information about the **TIGER**/Line File, Overview, Technical Documentation, sample files, Cartographic Boundary files and other products based on the **TIGER**/Line files, with links to ordering information. ... TIGER, ...
www.census.gov/geo/www/tiger

5. Tiger Woods
official site, complete with video interviews, photos, stats, and features.
www.tigerwoods.com

Fig. 4. The 4 systems tested in the experiment, and the 4 "Δs".

It is well known that evaluating the effectiveness of interactive IR systems is a difficult task, for which there are no standard metrics available [1]. We took into account the two main aspects of the overall retrieval performance: *Success*, representing the degree to which the goal has been accomplished, and *Speed*, computed on the basis of the amount of time to complete the task. A single numeric effectiveness value was then computed by computing their product, i.e.:

$$\text{Performance} = \text{Success} * \text{Speed} \qquad (1)$$

In our experimental setting there are two independent variables (device and method) and one dependent variable (the retrieval performance on the set of topics). The 48 subjects were randomly split into two groups with 24 subjects, each group being assigned to one device only (i.e., desktop computer or PDA), and each subject in either group performed all four finding tasks using both methods (clustering and search).

We now turn to the results. In Table 1 we show the values of *Performance* (Equation 1) obtained for each topic by each retrieval method and device, normalized from 0 to 100 and averaged over the subgroup of subjects who performed the relative tasks.

	T1	T2	T3	T4
PDA-Clustering	67,60	62,28	51,68	32,47
PDA-Search	54,77	49,94	41,91	27,57
Desktop-Clustering	63,49	65,77	61,16	37,66
Desktop-Search	65,73	53,25	42,15	46,14

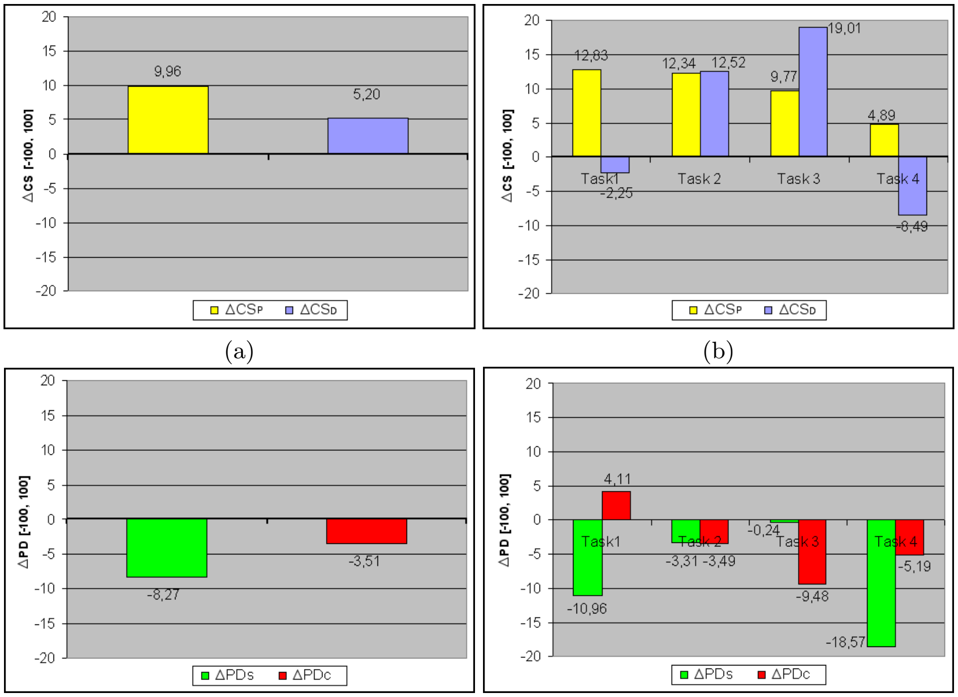
Table 1. Performance by method and device on individual topics

In Figure 5(a) we show the values of ΔCS_P and ΔCS_D averaged over the topic set. The figure shows that the clustering engine approach, on average, performed better than the search engine approach on both devices, with the difference being statistically significant for ΔCS_P (the non-parametric Mann-Whitney U test 1-tail gives $p = 0.016$).

A topic by topic analysis (Figure 5(b)) shows that while ΔCS_P was always positive, ΔCS_D presented considerable variations. Thus, at least in the case of desktop computer, the result depends on the specific topic being considered. This behavior is now analyzed more in depth. Topics 2 and 3 were characterized by some ambiguity and very useful clusters, so clustering was better than search both on PDA and desktop computer. For topics 1 and 4, the clusters produced by the system were pretty good, which explains the good performance of clustering on PDA. On the other hand, we found that, for topics 1 and 4, the subjects searching with search engine were able to detect good snippets and come up with effective query refinement strategies, whereas on the PDA device the screen size limitations and the typing constraints might have prevented them to do so in a fast and effective manner.

Figures 5(c) and 5(d) are the duals of Figures 5(a) and 5(b), obtained by swapping the roles of methods and devices. Figure 5(c) shows that the performance of PDA was on average lower than that of desktop computer across both retrieval methods, with the difference being statistically significant for ΔPD_S ($p < .05$). Figure 5(d) shows that ΔPD_S was negative for all topics and that ΔPD_C was always negative except for topic 1. The fact that on topic 1 clustering engine on PDA was better than clustering engine on desktop computer may seem somewhat surprising. As the cluster hierarchy produced for topic 1 was fine (see comment above), one possible explanation for the disappointing performance of desktop computer is that the presence of the snippets distracted the user away from the clustered results.

Overall, these findings represent an indication that (a) mobile clustering engine outperforms mobile search engine, whereas desktop clustering engine is not necessarily better than desktop search engine, and (b) mobile IR is worse than desktop IR, although it can be occasionally better for the clustering method.



(a) (b)

(c) (d)

Fig. 5. Mean retrieval performance of clustering versus search for each device (a); Values of ΔCS_P and ΔCS_D on individual topics (b); Mean retrieval performance of PDA versus desktop computer for each retrieval methods (c); Values of ΔPD_S and ΔPD_C on individual topics (d).

6 Conclusions and Future Work

We showed that mobile clustering engine is not only feasible, as demonstrated by the system Credino, but also effective. We found that the retrieval performance of mobile clustering engine, while remaining in general inferior to that of desktop clustering engine, was better than mobile search engine.

Of course, more experiments are needed to support these findings. It is interesting to see what happens as we choose a larger set of more typical queries, referring to broader or better known domains. One possibility is to experiment with the test collection made available by [8], although it is not easy to evaluate the retrieval performance of a hierarchical clustering engine in a precision/recall style. Also, it would be useful to experimentally compare CREDO and Credino to some of the other few clustering engines that have been proposed, although a mobile version of the latter systems is not available at the moment.

Credino can be technically improved. We plan to re-implement the mobile device client as a complete application, to further reduce the amount of data transmitted to the mobile device and to improve the usability of the system by reducing the response time of the interface. Another direction for future work is to develop a version of Credino for cellular smart phones.

References

1. E. Berenci, C. Carpineto, V. Giannini, and S. Mizzaro. Effectiveness of keyword-based display and selection of retrieval results for interactive searches. *International Journal on Digital Libraries*, 3(3):249–260, 2000.
2. G. Buchanan, M. Jones, and G. Marsden. Exploring small screen digital library access with the Greenstone digital library. In *Proceedings of the 6th European Conference on Research and Advanced Technology for Digital Libraries*, LNCS 2458, pages 583–596, Rome, Italy, 2003. Springer.
3. O. Buyukkokten, O. Kaljuvee, H. Garcia-Molina, A. Paepcke, and T. Winograd. Efficient web browsing on handheld devices using page and form summarization. *ACM Trans. Inf. Syst.*, 20(1):82–115, 2002.
4. C. Carpineto and G. Romano. *Concept Data Analysis — Theory and Applications*. Wiley, 2004.
5. C. Carpineto and G. Romano. Exploiting the potential of concept lattices for information retrieval with CREDO. *Journal of Universal Computer Science*, 10(8):985–1013, 2004.
6. F. Crestani, M. Dunlop, M. Jones, S. Jones, and S. Mizzaro, editors. *International Journal of Personal & Ubiquitous Computing, Special Issue on Interactive Mobile Information Access*. Springer-Verlag, 2006. In press.
7. F Crestani, M. D. Dunlop, and S. Mizzaro, editors. *Mobile and Ubiquitous Information Access, Mobile HCI 2003 International Workshop, Udine, Italy, September 8, 2003, Revised and Invited Papers*, volume 2954 of *Lecture Notes in Computer Science*. Springer, 2004.
8. P. Ferragina and A. Gulli. A personalized search engine based on web-snippet hierarchical clustering. In *WWW2005: The 14th World Wide Web Conference*, 2005. http://www2005.org/.
9. M. Halvey, M. Keane, and B. Smyth. Predicting navigation patterns on the mobile internet using time of week. In *Proceedings of the 14th International World-Wide Web Conference*, Chiba, Japan, 2005.
10. http://www.webpronews.com/insiderreports/searchinsider/wpn-49-20050708YahooAndTheQuestForMobileSearchSupremacy.html.
11. http://www.marketingvox.com/archives/2005/07/28/.
12. D. Kelly, F. Diaz, N. J. Belkin, and J. Allan. A user-centered approach to evaluating topic models. In *ECIR 2004*, pages 27–41, 2004.
13. D. J. Lawrie and W. B. Croft. Generating hiearchical summaries for web searches. In *Proceedings of SIGIR03*, 2003.
14. M. Noirhomme-Fraiture, F. Randolet, L. Chittaro, and G. Custinne. Data visualizations on small and very small screens. In *ASMDA 2005: Proceedings of Applied Stochastic Models and Data Analysis 2005*, 2005. http://asmda2005.enst-bretagne.fr/.
15. Jones S., Jones M., and Deo S. Using keyphrases as search result surrogates on small screen devices. *International Journal of Personal and Ubiquitous Computing*, 8(1):55–68, 2004.