Contents lists available at ScienceDirect

Journal of Network and Computer Applications

journal homepage: www.elsevier.com/locate/jnca

Adaptation of RSS feeds based on the user profile and on the end device

Christos Bouras *, Vassilis Poulopoulos, Vassilis Tsogkas

Research Academic Computer Technology Institute, N. Kazantzaki, Panepistimioupoli and Computer Engineering and Informatics Department, University of Patras, 26504 Rion, Patras, Greece

ARTICLE INFO

Article history:
Received 27 February 2008
Received in revised form
9 February 2010
Accepted 9 February 2010

Keywords:
Summarization algorithms
Categorization procedure
Data reprocessing
Efficient text summarization
Personalizing the web
Text categorization
Sentence weighting

ABSTRACT

In the last decade, the advances in technology along with the ease of access to information have dramatically changed the World Wide Web status during the last few years. The Internet acts as a means of finding useful information and more specifically news articles. Additionally, more and more people want to utilize their mobile devices towards the scope of reading news articles. The aforementioned situation generates a significant, yet almost untouched problem: easily locating interesting news articles on a daily basis within the space that is available on the small screen device. In our work, we propose a framework that, by utilizing RSS feeds, is able to personalize on the needs of the users and on the capabilities of their device, in order to present to them only a fraction of the news articles and merely the useful information that derives from them. Deploying a generalized, multi-functional mechanism that produces efficient results for the situation described, seems to be a panacea for most of the text-based, information retrieval needs. Within this framework we created PeRSSonal, a mechanism that is able to create personalized, pre-categorized, dynamically generated RSS feeds focalized on the end user's small screen device. The system is based on algorithms that incorporate the user into the categorization and summarization procedures, while the articles are presented back to him/her according to her interests and the client device capacity.

1. Introduction

The advances in technology along with the ease of access to information have changed dramatically, during the last few years, the World Wide Web status. This change has also affected the manner and the frequency that news articles are written and published on the Internet. Everyday, thousands of articles are generated by the vast amount of news portals and blogs that exist on the WWW. This sense of freedom is attracting more and more users, not only to read in a daily basis their "Internet newspaper", but also to create their own articles or sources of articles. Besides, the latest "blogging" trend is not only targeted on publishing a personal dairy, but also acts as a medium of information exchange. Towards this direction research is already underway (e.g. Baron, 2006) in order to observe the phenomenon of journalism on the WWW.

Despite the fact that the aforementioned conditions can be considered as an innovation for our world and as a founder of democracy, they generate a number of repeated problems for the users of the Internet who try to access information via their handheld devices. These kind of systems (small screen devices), that are becoming more and more popular, already do have the power to run complex interactive applications (Fitzmaurice et al., 1993). However, their main problem lies on the available space that their monitor has in order to help users track and read articles from news portals. Despite the increasing resolution of PDA screens, limitations on the physical size of these will prevent the devices from ever reaching parity with desktop computers (Gutwin and Fedak, 2004). Smartphones, PDAs and Ultra-Mobiles all have limited screen area compared to a conventional PC. For example, the Apple iPhone[1] specifications include a 3.5-in (diagonal) widescreen display, with 480 × 320 pixel WQVGA resolution. Other smart phones may have a screen of only 2.5 in, a PDA of 4 in and Ultra-Mobile PC 7 in, with resolutions from QVGA (240 × 320 pixels) to 800 × 480 or larger for a Ultra-Mobile PC. All those reference values are far from PC screen resolutions and thus the content needs to be adapted to different sizes and shape screens.

The matter of presenting the information to the end-user is the "one side" of the problem that we are tackling in this manuscript. The "other side", derives from the vast amount of information which forces users to specific tags or makes the tracing of information within a news portal a tedious task. As news articles that are published daily from the major news portals are rapidly

* Corresponding author. Tel.: +30 2610 960375.
E-mail addresses: bouras@cti.gr (C. Bouras), poulop@cti.gr (V. Poulopoulos), tsogkas@cti.gr (V. Tsogkas).

[1] iPhone © is a trademark of Apple company—http://www.apple.com/iphone/.

1084-8045/$ - see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.jnca.2010.02.004

increasing in numbers, the act of locating “the useful ones” is becoming more and more demanding. With the term “useful articles” we define the articles that the Internet user is really interested in reading, and they vary from user to user. A simple search using the most notable search engines for specific terms, is frequently of little use since the returned results are often thousands or millions, or have not yet been updated with up-to-date news articles. Consider that news and articles can be produced with a frequency of minutes a frequency that is impossible to be reached from the search engines. As far as the searching tools, existing in many news portals, are concerned, their service is often frustrating since they do not provide adequate filtering capabilities resulting to unstructured or random outcomes.

In this manuscript we present a novel Internet service whose main scope is to support Internet users who are interested in reading, on a daily basis, specific news categories and we focalize mainly on users with small screen devices. PeRSSonal (Bouras et al., 2008), the personalized summarization system based on the RSS protocol that we are constructing, is being developed modularly: based on already constructed web fetching, text pre-processing, text summarization and categorization techniques, we are adding novel personalization algorithms to provide a unique information filtering framework to end users. The challenge is twofold: we do not only have to locate the news articles that the user is interested in reading, but we also have to present information in such a way that the user will be able to read the most representative parts of it. Within these limitations, we are presenting the personalization module of the mechanism along with its user interface and profile generation algorithm.

The well-known RSS protocol, which is based on the XML language, helps users confront consolidated information from websites and especially news portals. It is adopted by almost all the news portals and generally by websites whose content is updated very often. Its goal is to provide the users with a title and a summary of an article, or of an important part of information that was published within a website, and let the user decide whether she wants to view the complete article or not. Lately, it is widely used from search engines and meta-portals in order to locate rapidly changes in the content of a website. Despite the fact that creating dynamic RSS content is not a difficult procedure, most news portals are mis-utilizing them in a sense that, the received feeds provide little information about the event since they consist only of a couple of sentences usually not conveying much of the article’s information.

Based on the fact that Internet users are becoming familiar with this protocol, we are developing a system that is utilizing RSS feeds in order to present them with filtered information in a more structured manner than the RSS feeds provided by the major portals. More specifically, our system collects news articles from major and minor news portals which are predetermined by the system’s administrator. In essence, we have chosen a wide variety of new portals using their RSS feeds for pulling articles updated in a frequent manner. Some of the news portals that we are indexing are: CNN, BBC, Washington Post, Reuters, MSNBC and many more. Notice also that more than one RSS feeds can been chosen from each news portal provided that duplicate insertion of identical articles is avoided.

Pre-processing techniques are applied to the collected articles and then categorization and summarization algorithms are used in order to refine them. Additionally, we empower the mechanism with personalization algorithms in order to include the end-user to the whole procedure and thus the system is empowered with the ability to produce isolated RSS feeds (title and summary for the latest articles), for each user, according to their personal device and preferences (personalization). The results generated by the described system are transmitted to the end users via RSS feeds following the 2.0 standard. However any XML-based protocol for news syndication, such as atom feeds, could be chosen, provided that it covers the need for textual and possibly image data transmission.

Aiming towards the determination of the effectiveness of the proposed framework, we evaluated the PeRSSonal system from three perspectives. Firstly, we evaluated its ability to present adequate information to the end users according to the device that the user is utilizing. We measured the fractions of text that are presented to the user according to the available space on her screen. Next we evaluated the effectiveness of PeRSSonal according to the accuracy of the results. We finally measured the coverage of the user’s choices and needs that the results convey, determining also the overall improvement of the system with the adopted changes.

The rest of the manuscript is structured as follows. The next section summarizes the related work on the field of text categorization, automatic text summarization, as well as, personalization and adaptation of content according to the users needs. Section 3 presents the architecture of the PeRSSonal mechanism, while Section 4 introduces algorithmic aspects of the complete mechanism. In Section 5 we present the results of the evaluation procedure that took place and in Section 6 we give some conclusions from the current work. Finally, in Section 7 we present some possible future work that can further improve the PeRSSonal system.

2. Related work

The mechanism that we have created includes, among others, algorithms for categorization and summarization. Since the dynamically created personalized RSS feeds is an innovative feature and relies on the aforementioned algorithms we will present the state of the art of classification and summarization procedures.

Text classification (categorization) is the process of deciding on the appropriate category for a given document. Classification tasks include determining the topic area of an essay; deciding to what folder an email message should be directed; and deciding on which newsgroup a news article belongs (e.g. Google news). The purpose of text categorization as depicted by Hayes et al. (1988) is to accompany readers to their search of news articles, by creating and maintaining key categories which hold articles related with a specific topic of interest (Hsu and Lang, 1999; Antonellis et al., 2006). New articles are categorized to the pre-defined categories using some criteria which vary from one technique to another. The use of pre-defined categories can be relatively coarse-grained, i.e. only some basic, unrelated to each other, categories are defined, such as: business, education, science, etc., or fine-grained where many categories, which are frequently overlapping with each other, are introduced.

The goal of summarization as expressed by Radev et al. (2005) is the generation of a summary out of one or more, usually related to each other, articles and hence easing the user from the tedious task of reading large texts. A summary (Wasson, 1998) usually helps readers identify interesting articles or even understand the overall story about an event. In most of the times, the summarization approaches are based upon a “sentence level” (Goldstein et al., 1999), where each sentence is rated according to some criteria (e.g. important keywords, lexical chains, etc.). Some techniques (Ferragina and Gulli, 2005) try to find special words and phrases in the text, while others like Hayes et al. (1988) compare patterns of relationships between sentences. Taking into consideration the length of the sentences or the word case has also been tried (Herman, 2000).

While some summarization techniques try to extract the most important sentences, as far as a certain measure is concerned, others attempt to generate the summary using a knowledge-based representation of the content or a statistical model of the text (Kummamuru et al., 2004). Recently (Baron, 2006), there is an effort to find the dynamic portions of a document and use this to produce good summaries based on the hypothesis that the higher the number of dynamic parts containing a term, the more important this term is for the summary.

Despite the extensive work on the field of summarization, little effort has been made towards the direction of combining summarization techniques with the RSS news transmitting channels. Almost all news feeds provided by news portals (e.g. Google news) consist of a title and some of the first sentences of the article (if not just the first words).

The physical size of small screen devices, limits the maximum displayable content, which can be no larger than the dimensions of the machine in which it is embedded. On the other hand, the need for displayed text to be legible, defines another, more subtle boundary; if the size of text cannot be reduced below a threshold of legibility, then, as the screen shrinks in size, and less information may be shown on it, the user will be required to increase the level of interaction with the device in order to get to desired information. Our research work aims to deal with problems of this kind providing solutions that are device independent.

3. Architecture

PeRSSonal, as presented in Bouras et al. (2008), follows a classic n-tier architectural approach. The system consists of multiple layers which work autonomously and collaborate through a centralized database. The web interface handles the information flow into the mechanism which is then directed to the interior subsystems. Text pre-processing techniques follow and the results are forwarded to the next level of analysis where core information retrieval (IR) techniques take place. Finally, the outcomes are presented to the end users though the information presentation subsystem. The collaboration between the distributed systems is based on open standards utilizing XML for input and output which are supported by each part of the system and by the communication with a centralized database. Fig. 1 depicts the architecture of the complete mechanism with its enhanced parts in the dashed box.

The procedure of the mechanism as depicted in Fig. 1 is: (a) capture pages from the internet and extract the useful text (text containing the article's body), (b) parse the extracted text and preprocess it, (c) summarize and categorize the text and (d) personalize the results and present them to the end user.

In order to capture the pages, a simple focused web crawler is used. The crawler receives as input the addresses that are extracted from existing RSS feeds, deriving from several major news portals. These RSS feeds point directly to web pages where news articles exist. The crawling procedure is distributed across multiple systems which communicate with the centralized database. Crawled HTML pages are stored without any other element of the web page (images, css, javascript, etc. are omitted). During this analysis level, our system isolates the "useful text", which includes the title and the main body of the article, from the HTML page. More information about this procedure can be found in Bouras et al. (2005). By storing only the useful text, as well as some other page meta-data, such as URL and insertion date, the database is populated with news articles that are ready for the text pre-processing step.

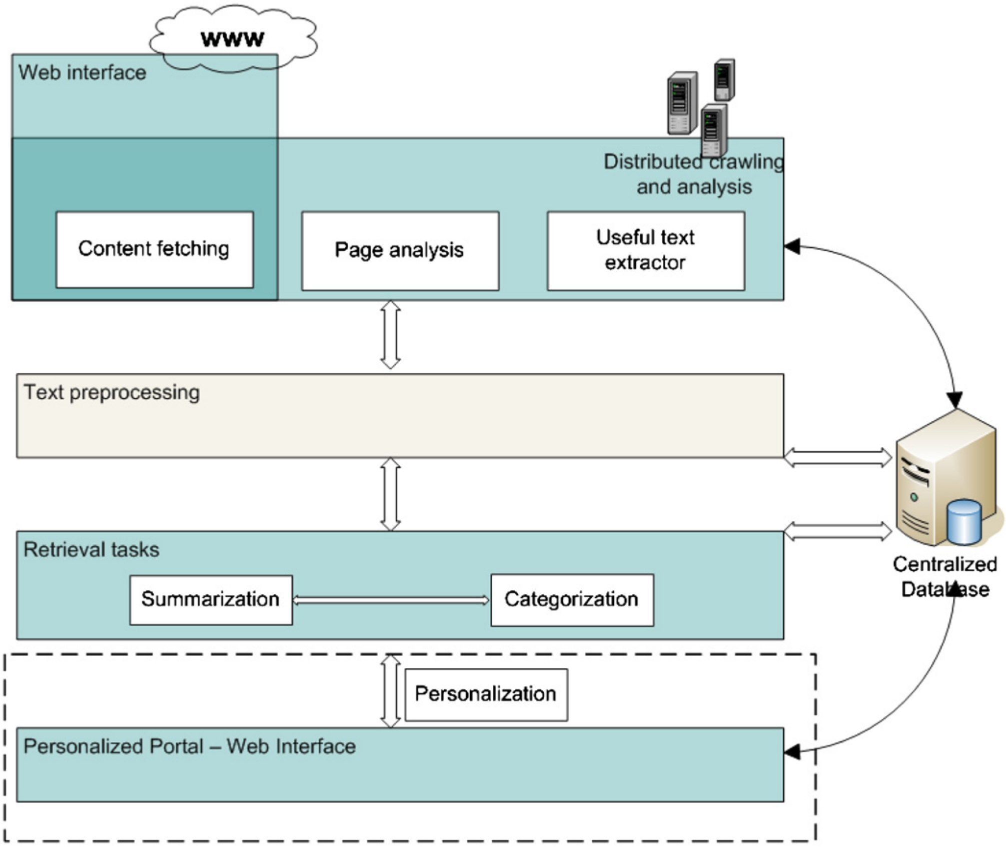


Fig. 1. PeRSSonal's architecture.

The second analysis level receives as input XML structured information, deriving either from the database or from raw XML files, which include the article's title and body. Its main scope is to apply text pre-processing algorithms on the article, providing as output keywords, their location into the text and their frequency of appearance in it. These results are necessary in order to proceed to the third analysis level. Information about our pre-processing mechanism can be found in Bouras et al. (2006).

The core procedures of PeRSSonal are located in the third analysis level, where the summarization and categorization subsystems reside. Their main scope is to characterize the article with a label (category) and produce a summary out of it. The results are led to the personalization module (fourth level) of our mechanism, where the profile generation algorithm is applied. Personalized summaries are finally presented back to the end users though the web portal or though RSS feeds. The role of the personalization layer is to "feed" each user only with articles that she "wants" to face according to her dynamically created profile.

Our centralized database holds data generated and needed by each level of analysis, such as raw article bodies, keywords, generated summaries, categorization results, personalization data, as well as logging of user accesses.

4. Algorithmic aspects

In order to analyze how each algorithm is applied on the texts we will present the procedure that is followed in each step. The complete flow of information of the PeRSSonal system is pictured in Fig. 2.

The procedures of fetching the articles from the WWW, pre-processing them, categorize and automatically summarize them are analyzed superficially as the intention is to focalize on the web interface and the personalization factors of both the Web Portal and the personalized RSS that is offered to the visitors of the Portal.

4.1. Article fetching and useful information extraction

The procedure of the News fetching subsystem, as depicted in Fig. 2, is: (a) capture pages from the WWW and analyze them extracting the useful text, (b) store the useful text in the centralized database. The algorithm that fetches the article is very simple and is based on the fact that every web portal includes a series of RSS that are offered to the end user. Instead of having to visit every page of numerous news portals that exist on the WWW, we fetch their RSS and more specifically the ones that includes the daily "top stories". From the XML structure of the feeds we can obtain the most important articles that are published to each news portal together with information about the title of the articles, the exact URLs and the dates of publication.

Following the articles' URLs extraction from each feed, the crawler, currently working as a wrapper, changes its functionality and becomes a simple crawler which visits every single URL extracted from the RSS feeds in order to obtain the HTML code of the latest articles.

Extracting the useful text from the HTML pages collected by the crawling mechanism, is a procedure where the fetched web page is analyzed and the contiguous parts of it, which include a large amount of text, are considered to include useful information. The useful text extraction is based on the fact that the HTML pages can be depicted as a tree with every tag holding a node on the tree, while every leaf includes pure text. In order to extract the useful text we utilize a clipping extraction technique described by Bouras et al. (2005) that is able to recognize the nodes that have high informational value and almost zero hub value. Informational value is high for the nodes that include large amounts of text while hub value is high for nodes that include medium amount of text and many links.

The extracts of the useful text sub-system are mainly the body of the text and maybe a representative title. These parts are processed by the text pre-processing sub-system. This mechanism is assigned with the task of "cleaning" the text and extracting the keywords. Information about this system can be found in Bouras et al. (2006). The outcomes of the pre-processing mechanism are: stemmed keywords, their frequency in the text and their position in the text. This information is enough for the following sub-systems of our mechanism in order to apply categorization and summarization on the text.

4.2. Categorization procedure

The categorization subsystem is based on the cosine similarity measure, dot products and term weighing calculations. The system is initialized with a training set of humanly pre-categorized articles, collected from major news portals. The categorization module receives as input the extract of the pre-processing mechanism. This is (a) an XML structured source containing stemmed keywords, their absolute frequency and their relative frequency in the article and (b) the XML structure that contains the article's title and body. After the initialization of the training set, the categorization module creates lists of keywords

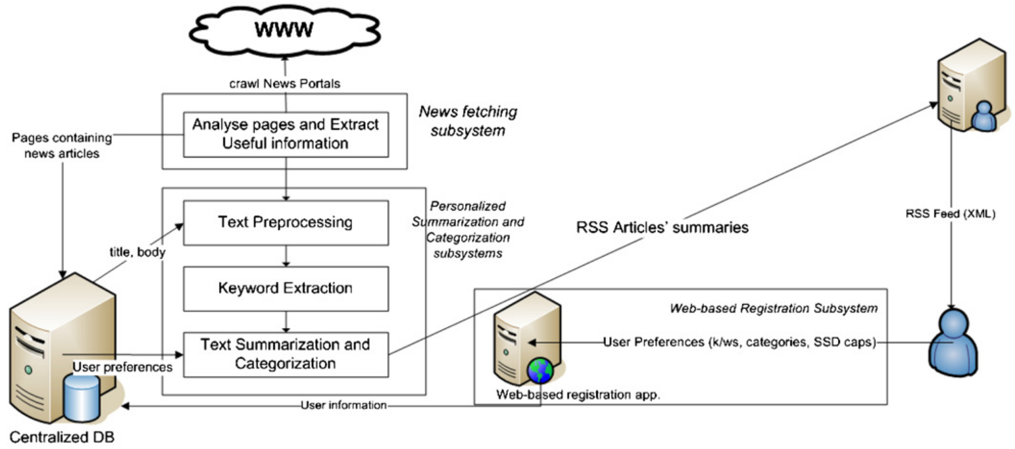

WWW
crawl News Portals
Analyse pages and Extract Useful information
News fetching subsystem
Pages containing news articles
Text Preprocessing
Keyword Extraction
Text Summarization and Categorization
Personalized Summarization and Categorization subsystems
title, body
User preferences
Centralized DB
User information
RSS Articles' summaries
RSS Feed (XML)
Web-based Registration Subsystem
User Preferences (k/ws, categories, SSD caps)
Web-based registration app.

Fig. 2. Flow of information.

that are representative of a unique category, consisting of keywords with high frequency in a specific category and small or zero for the others. The creation of the lists is helpful for labelling newly arriving articles but has proven by Bouras et al. (2007) to be also helpful for the summarization procedure.

Our system consists of seven categories constructed from a knowledge-base of 7000 pre-categorized articles derived from multiple corpuses. The constructed categories are: business, politics, sports, education, science, technology and entertainment. Each article belongs only to one of the seven categories and the corpuses are balanced to the categories, meaning that each category consists of exactly 1000 articles. This was achieved via recording articles for a period of time from representative portals/RSS feeds, which were posting articles belonging to each particular category. By sharing the amount of articles used for each category no bias is introduced to the training set of the categorization procedure. Despite the fact that seven categories seem too few, considering the plethora of news categories at several news portal, we have chosen only categories with semantic meaning that does not overlap. Moreover, simple classification tests with our classifier have revealed average precision of 0.75 and recall of 0.5 which can be considered as satisfactory. Note however that an exhaustive experimentation of the classification mechanism is beyond the scope of this manuscript (Bouras et al., 2007).

When a new article is fetched, the text similarity algorithm is applied to it and the system decides which is the category that better represents the content of the text. Categorization results are accepted only if certain thresholds, which are explained in Bouras et al. (2007), are met.

If the categorization procedure fails to categorize an article, then a simple assumption is made: a well-summarized text that includes only the important information of a text and thus, only the important keywords of it, has higher possibility to be categorized than the original text. After a failure of the categorization procedure, the system summarizes the text and attempts a second categorization. If the categorization fails again the text is noted as generic and is also labeled as per the category with the highest score of the first categorization attempt. The note of the generic category is used as a pointer that the categorization results might not be always correct for those articles.

4.3. Summarization procedure

If a text is not categorized then an attempt for a generic summarization is made. During this attempt we utilize two metrics: (a) the existence of each keyword in the title and (b) the frequency of each keyword. We call these factors k_1 and k_2. A keyword with very high frequency in the text is considered to be representative of it and thus any sentence that includes this keyword can be thought as representative of the text. Additionally, any keyword of the text that also exists in the title is marked, so the sentences that include this keyword are more representative of the text. However, a keyword's frequency alone is not sufficient for judging the keyword's importance. For this reason, some extra heuristics, which will be described later, are used.

At this point, k_1 derives from the following equation:

$$k_1 = 1+0.1x \tag{1}$$

where x is the times that the keyword appears in the title. The 0.1 factor is used on purpose for weighting the frequency of a keyword in the article's title (Bouras et al., 2006).

k_2 derives from the following equation:

$$k_2 = 1+1.2y \tag{2}$$

where y is the possibility of a keyword appearing n times in a sentence. Assuming a sentence with length m of a text with length t, the possibility of the keyword to appear n times is

$$y = p(n|m)\cdot(m|t) \tag{3}$$

The 1.2 factor in Eq. (2) is used for weighting the aforementioned probability value and has proven to be giving good results (Bouras et al., 2006).

Based on these heuristics, we create a summary which consists of the most representative sentences of the text. In order to determine these, we deploy a score for each sentence according to the factors k_1 and k_2. Assuming that the text T has s sentences where i = [1…s] and f keywords where k = [1…f], each sentence is assigned a score according to the following equation:

$$W_i = \sum(1+rel(fr(kw_{k,i})))(k_1+k_2) \tag{4}$$

where $rel(fr(kw_{k,i}))$ is the relative frequency of the keyword k in sentence i.

After creating a generic summary, we retry to achieve a categorization, as the summarized text is more refined and consists only of important sentences and not of the whole text which may include sentences with keywords that are distracting the categorization procedure.

On the other hand, if the initial categorization attempt is successful, the procedure that is followed in order to summarize a text after differs from the aforementioned steps due to the fact that another factor is included in the scoring. This factor is the keywords ability to represent the category to which the document belongs. As long as the text is categorized, we can utilize this factor in order to create a more efficient summary. The theory that we are relying on is that, if the text is categorized, then there exist some keywords in the text that are representative of the texts category. This information can lead us to the use of another factor, k_3 that covers the ability of the keyword to represent a category. Assuming that the relative frequency of a keyword within a category is cf, k_3 can be computed as

$$k_3 = A\cdot(1+cf_i) \tag{5}$$

where A is the "special weight" of k_3 and is added in order to represent how much the computation of the sentence weighting will be relied on factor k_3. After experimental procedure, we concluded that a best fitted value for A is 1.2. Though, it can be set to 1 if we do not want to rely on the k_3 factor, or it can be increased to 1.8 in order to rely mainly on the k_3 factor and actually omit the k_1 and k_2 factors. Values less than 1 and more than 1,8 produce unexpected results as in the first occasion k_3 leads to lessening of the sentence weight while in the second case the result does not rely at all on k_1 and k_2 (it is like not using them). If a text's keyword does not belong to the category of the text, then k_3 is set to 1. A procedure that is experimented at the time being is allowing k_3 to get negative values by examining whether the keyword text belongs to a category other than the one of the text. In this occasion, we assume that the keyword is representative of another category and not the text's category and hence, the overall weight of the keyword's sentence has to lessen. With the use of k_3, the overall weighting equation is depicted below.

$$W_i = \sum(1+rel(fr(kw_{k,i})))(k_1+k_2)k_3 \tag{6}$$

The use of factors k_3 and k_4 (derived from the personalization process as will be described in the next subsection) makes the extracted summary more tolerable to the "frequency-only" k_1 and k_2 metrics described earlier, resulting thus to better summaries.

4.4. Web interface

The Web-based registration and user's interface subsystem represent the initial interface between the whole mechanism and the end user. A user registers in the system providing information about (i) her small screen device (device capabilities) and (ii) her categories' preferences. This information is stored in the centralized database and is used later at the personalized summarization procedure.

While registering, each user is prompted with the seven categories that exist in the mechanism and is asked to assign a rate to each category according to her preference. The score varies from -5 to 5, where "-5" means "I dont like at all" and "5" means "I like very much". By selecting zero (0), the user indicates her neutral statement against the respective category (Fig. 4).

Relying on these selections, we can create a simple user profile. At first, we create a list of the categories that the user likes and the ones that she does not like. This can help us with an initial "cleaning up" when selecting which news articles the user is interested in. The user is not just prompted to select the "likes" and "dislikes", but she selects a weight for each category. By utilizing these data we are able to create a more detailed user profile, which consists of a list of keywords that indicate the keywords that the user likes and the ones that she dislikes, followed by a relative frequency. The creation of the profile is constructed with the help of the following algorithm. Note that the described process is imprortant for the system in order to startup the profile for each user; moreover this process takes at most 5 minutes.

Algorithm 1. Create_profile()

```
for all selection s do
  if s!=0 then
    Keyword_name_usr = select 20*s keywords from category
    keywords {the keywords used for categorization,
    summarization etc}
    Keyword_weight_usr = select (2*s*relative frequency)
    from category keywords {the same list as above}
  else
    Keyword_name_usr = select 10 keywords from category
    keywords
    Keyword_weight_usr = select relative_frequency from
    category.keywords
  end if
  Insert into user profile keyword_name_usr,
  keyword_weight_usr
  if exists then
    Update user profile set keyword_weight +=
    keyword_weight_usr where keyword_name =
    keyword_name_usr
  end if
end for
```

From the user selections, we choose 20 s keywords, where s is the user's selection, (if user chooses 4 we select 80 keywords) from the training set's list, ordering the list by keyword's relative frequency in descending order. Additionally, we select the relative frequency of these keywords multiplied by 2 s (if the user chooses -3 and the keyword has relative frequency equal to 0.02 then we extract -0.12). In this way, we end up selecting what is needed for the personalization procedure:

- Many keywords from the categories that the user has selected with high score (either positive or negative) and few keywords from the categories that the user has selected with low score.
- High positive value for the relative frequencies of the keywords belonging to categories that the user has selected with high preference, and low negative value for the frequencies of the keywords belonging to categories that the user has selected with negative preference.

These measures can help us refine the results presented to the user. By utilizing this information we can achieve the following:

- Select texts from the categories that the user likes.
- Select texts from the categories that the user likes and do not belong to a category that the user dislikes.
- Refine the outcomes of the summaries by adding the personalization factor.

The aforementioned procedure, gives us the ability to add another factor used for creating personalized summaries. The factor utilized is called k_4 and can be used as a product to Eq. (4) or (6).

Assuming that for a user we have constructed a list of keywords followed by their relative frequency (preference of the user), k_4 derives from the following equation:

$$k_4 = B \cdot (1+uf_i) \tag{7}$$

where uf_i is the user's preference for the keyword i and B is the "special weight" of k_4 and defines how much will k_4 affect the result of the sentence weighting. After experimental procedure we have concluded to the value 1.8 for B.

When we have knowledge of an articles category, we apply the k_4 factor on Eq. (6), while when we cannot categorize, we apply the factor on Eq. (4) as

$$W_i = \sum (1+rel(fr(kw_{k,i})))(k_1+k_2)k_3k_4 \tag{8}$$

$$W_i = \sum (1+rel(fr(kw_{k,i})))(k_1+k_2)k_4 \tag{9}$$

Eq. (8) can provide us with information for creating a generic personalized summary for the articles that cannot be categorized. In this occasion the uncategorized article is not fully generic. It is an article that seems to belong to more than one category or the system cannot clearly define a unique labeling and thus it suggests more than one category for the specific article. In that occasion we can still refine which article to select to provide to the end-user as we have knowledge of the user's general preferences (likes and dislikes about categories). For example, a user that likes entertainment and politics and dislikes business and health will not be provided with an uncategorized article which stands between business and health or seems to be about business, health and politics.

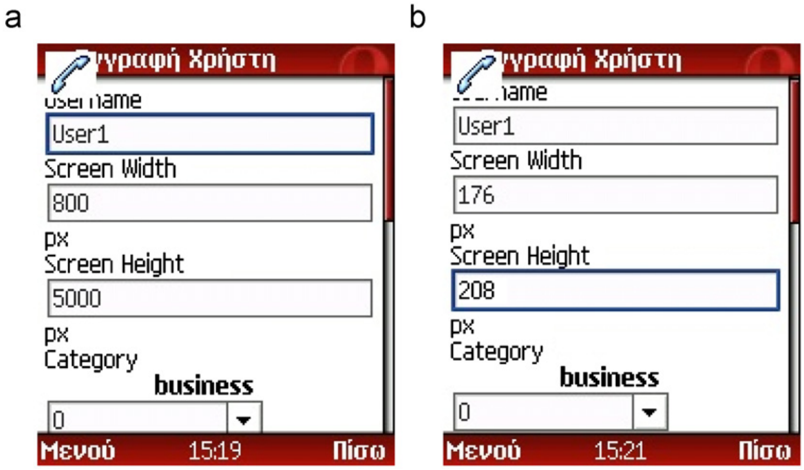
The two steps refinement of the articles described earlier, is very helpful, firstly to decide which articles to present to the end-user, and secondly, how to present the articles to the specific device of the user. This gives our mechanism with two unique features: the ability to select which articles to present to the user relying on the her preferences, personalizing in this way a dynamically created RSS feed, and the ability to personalize the dynamically created RSS feed (summary) on the end-users device, transferring only the amount of data that can be viewable within a limited amount of pages at the specific small screen device. Note that the amount of pages, based on the transmitted content, varies from device to device, but as a general rule we try to limit this paging as much as possible enhancing performance and ease of use.

Recording of the user choices on reading particular articles is achieved in the following manner: the RSS feeds that the system server, include summaries of articles indexed by the system, as well as links to the originating articles. When a user wants to read more about a specific article, by clicking the provided url she is redirected to the originating article via a webpage recording her interest before the redirection. This approach is similar to the one that the Google search engine uses for recording user clicks on the presented results.

5. Evaluation and experimental results

In this section we provide some screenshots of the web interface we implemented for the PeRSSonal system and the evaluation of the mechanism. In order to evaluate our mechanism we conducted three sets of experiments. Firstly we present the environment through which the user is able to register to the system and we evaluate the created summaries for some use-cases.

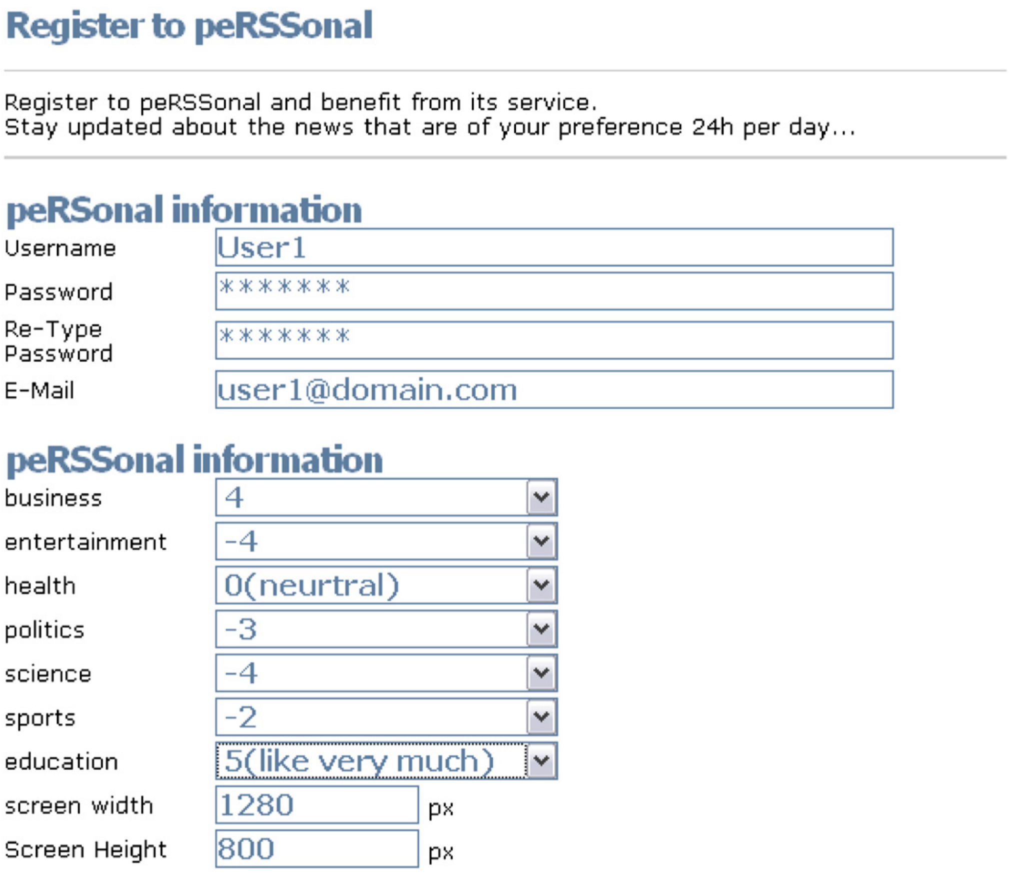
When a new user arrives, she provides her username and her screen capabilities (Fig. 3a). The later is auto-detected by the system but can also be user-modified and is necessary in order to define (i) the length of the news summaries sent back to the user and (ii) the number of news articles that are best suited for the device capabilities. In case a user uses her registered account with a device that is detected to have significantly different resolution



(a)
γγραφή Χρήστη
Username
User1
Screen Width
800
px
Screen Height
5000
px
Category
business
0

(b)
γγραφή Χρήστη
Username
User1
Screen Width
176
px
Screen Height
208
px
Category
business
0

Fig. 3. User registration. (a) User data input. (b) User preferences.



Register to peRSSonal

Register to peRSSonal and benefit from its service.
Stay updated about the news that are of your preference 24h per day...

peRSonal information

Username	User1
Password	*******
Re-Type Password	*******
E-Mail	user1@domain.com

peRSSonal information

business	4
entertainment	-4
health	0(neurtral)
politics	-3
science	-4
sports	-2
education	5(like very much)
screen width	1280 px
Screen Height	800 px

Fig. 4. Registration though the portal.

capabilities than the one registered, the resolution currently detected is used for determining the above parameters. A user also provides her categories preferences (Fig. 3 b), in the form of rating from -5 to $+5$ (Fig. 4), as well as any keywords that are of her special interest and should be highly rated through the article and sentence rating procedure.

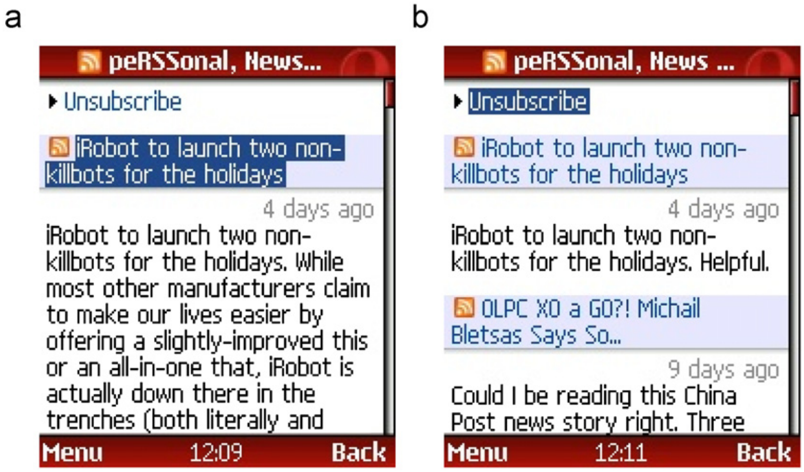
When an unregistered user requests an RSS feed, an RSS response, which contains the default summaries, is sent back (Fig. 5a). On the other hand, if the user is registered, she is fed with a personalized summary (Fig. 5 b) corresponding to her profile. The important factor to keep in mind is that different users receive different RSS responses, which vary in terms of news': length, ordering, amount, and categories. It is possible that two users receive the same articles but different summaries; this is the case of Figs. 6a and b.

In order to evaluate the summarizer of the proposed mechanism, we followed an extensive experimentation and comparison of it with some well-known text summarization systems. In this scope, we created a user's profile with high preference in the "business" category and zero to the others. Next, we randomly collected 40 articles, which seemed to be relevant to the "business" category (as far as their in-portal categorization is concerned), from various news portals and categorized them using the mechanism's categorization module. Afterwards, we examined the precision and recall outcomes when these articles are fed both to our system and to the MS Word and MEAD summarization mechanisms. The results are depicted in the graphs of Figs. 7 and 8.

From the previous graphs it is deducted that the PeRSSonal's summarizer outperforms the MS WORD on an average of 20% as far as precision is concerned and by 25% better recall. Compared with the MEAD summarizer, the proposed mechanism exhibits 10% and 14% better average rates respectively.

Experimentation with the same set of articles took place in order to determine the overall improvement of PeRSSonal with the appliance of the new personalization and profile creation algorithm. The results, depicted in the graphs of Figs. 9 and 10, convey an average improvement of 8% for the precision of the summarizer and 26% for its recall. The results, even though deriving from a relatively small set of articles, constitute a significant upgrade for the system.

Apart from the obviously different responses of the mechanism under different circumstances and the efficiency of the summarization mechanism, we needed to evaluate the positive effect that this had on the system's users. During this test phase, we created 10 user profiles with specific preferences concerning the categories. We ensured that these people were receiving daily to their RSS reader the feeds from 10 portals and the feed from our portal (which collects articles from all these 10 portals). We examined how many of these articles were of interest to the users in either of the cases.



iRobot to launch two non-killbots for the holidays

4 days ago

iRobot to launch two non-killbots for the holidays. While most other manufacturers claim to make our lives easier by offering a slightly-improved this or an all-in-one that, iRobot is actually down there in the trenches (both literally and

iRobot to launch two non-killbots for the holidays

4 days ago

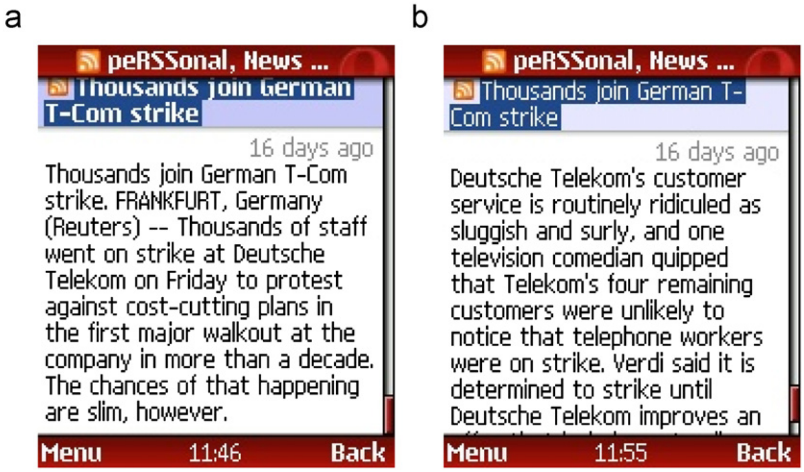
iRobot to launch two non-killbots for the holidays. Helpful.

OLPC XO a GO?! Michail Bletsas Says So...

9 days ago

Could I be reading this China Post news story right. Three

Fig. 5. (a) Default RSS feed. (b) Personalized RSS feed.



Thousands join German T-Com strike

16 days ago

Thousands join German T-Com strike. FRANKFURT, Germany (Reuters) -- Thousands of staff went on strike at Deutsche Telekom on Friday to protest against cost-cutting plans in the first major walkout at the company in more than a decade. The chances of that happening are slim, however.

Thousands join German T-Com strike

16 days ago

Deutsche Telekom's customer service is routinely ridiculed as sluggish and surly, and one television comedian quipped that Telekom's four remaining customers were unlikely to notice that telephone workers were on strike. Verdi said it is determined to strike until Deutsche Telekom improves an

Fig. 6. (a) RSS response for user A about a specific article. (b) RSS response for user B about the same article.

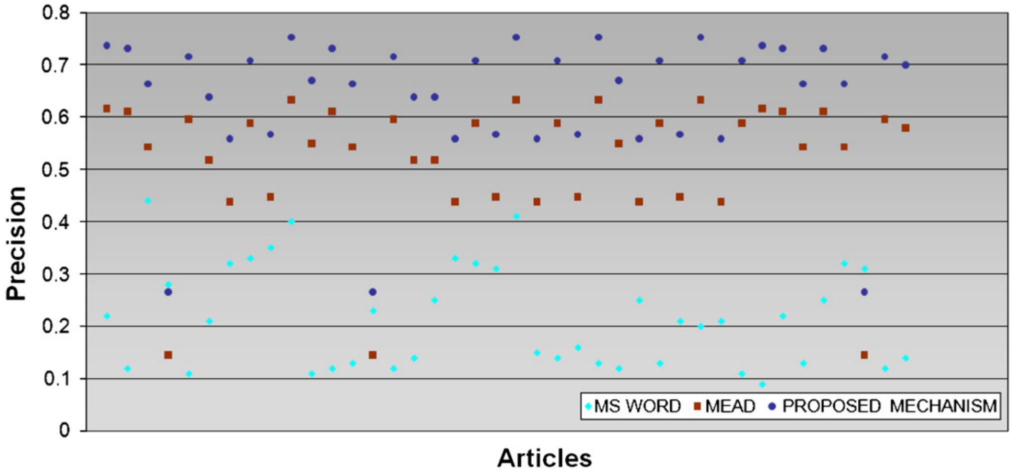


Fig. 7. Precision comparison between the proposed summarization mechanism and the MEAD and MS WORD summarizers.

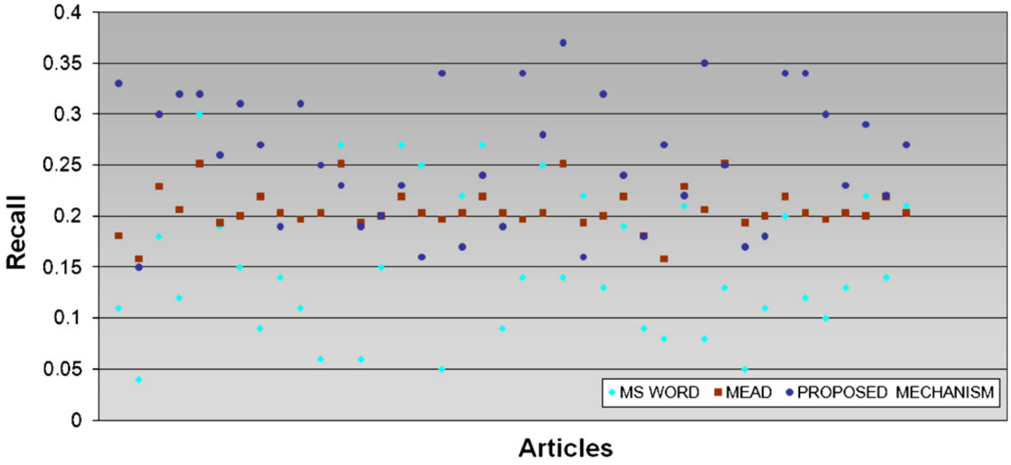


Fig. 8. Recall comparison between the proposed summarization mechanism and the MEAD and MS WORD summarizers.

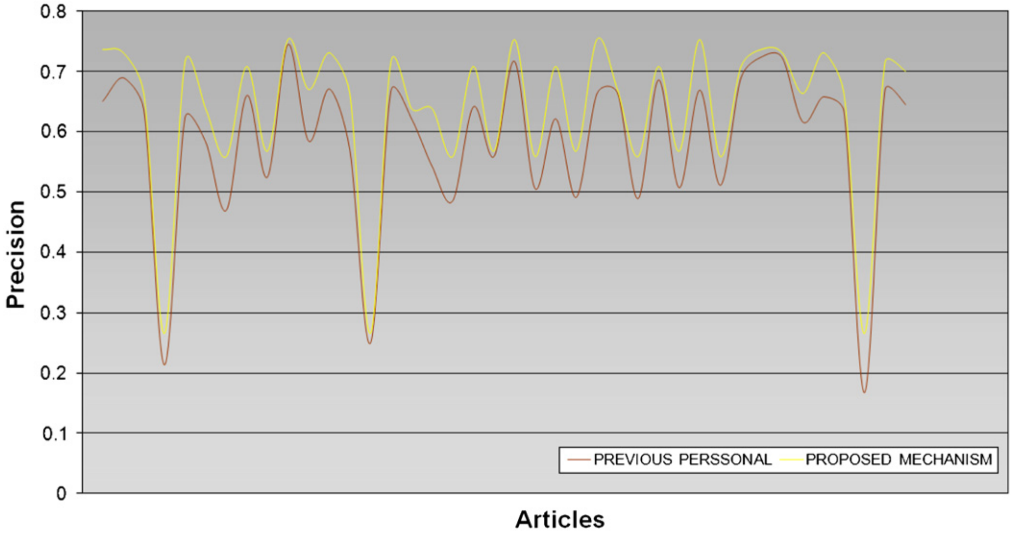


Fig. 9. Precision comparison of the summarization procedure between the proposed (new) PeRSSonal and the original.

From the following graphs, it is clearly depicted that PeRSSonal presents an average of 85% less articles daily but the percentage of articles that the users seem to be interested in is more than 40% of the presented ones, while in the second occasion the users are interested in reading only the 7% of the articles presented. This means that the mechanism can achieve better

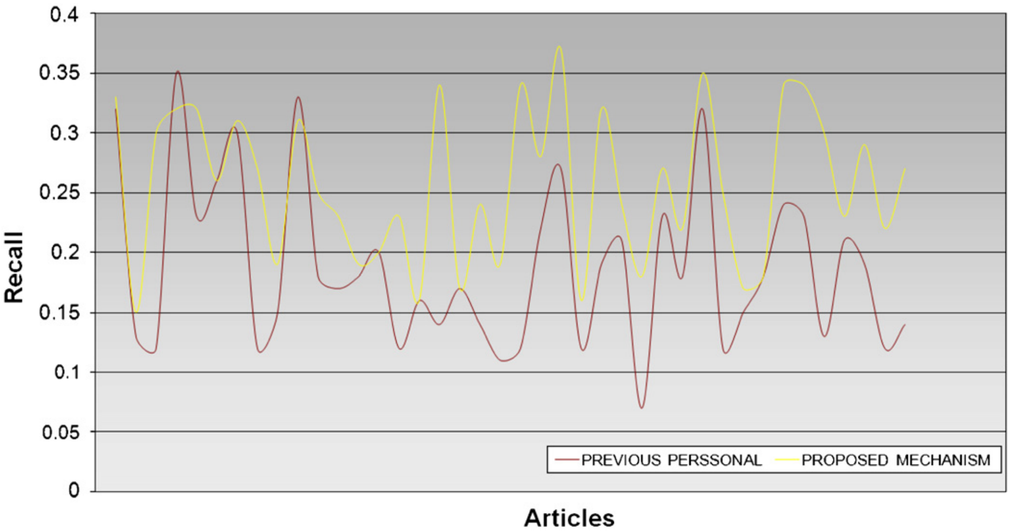


Fig. 10. Recall comparison of the summarization procedure between the proposed (new) PeRSSonal and the original.

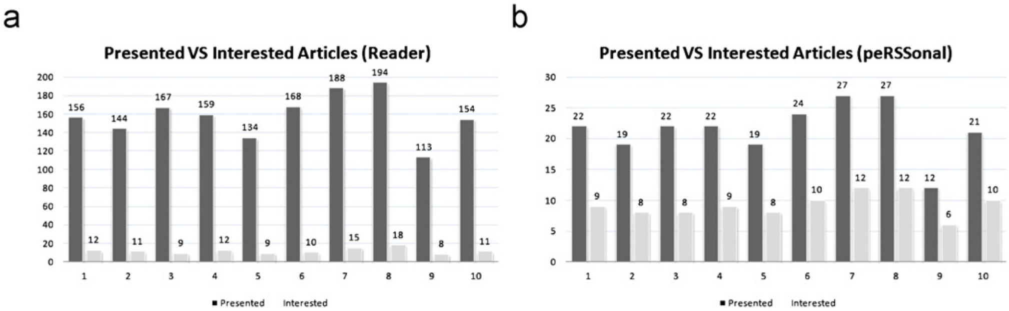


Fig. 11. (a) Presented and interesting articles directly from all news portals. (b) Presented and interesting articles from personal.

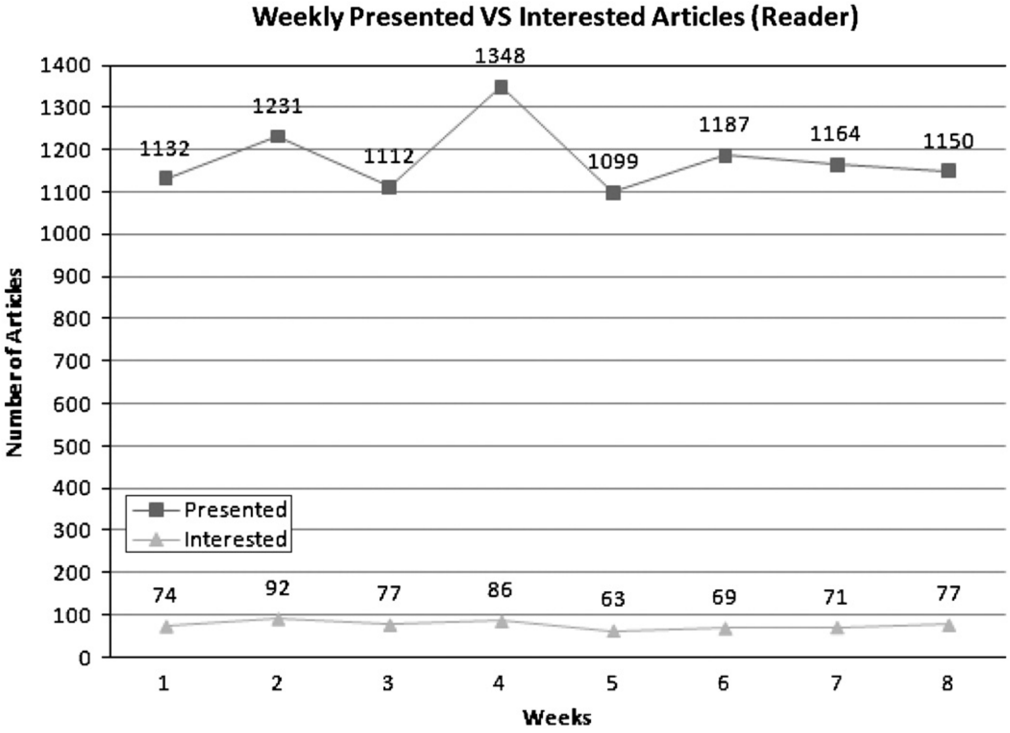


Fig. 12. Weekly presentation of articles from the RSS reader. (b) Weekly adaptation of personal to the profile of the user.

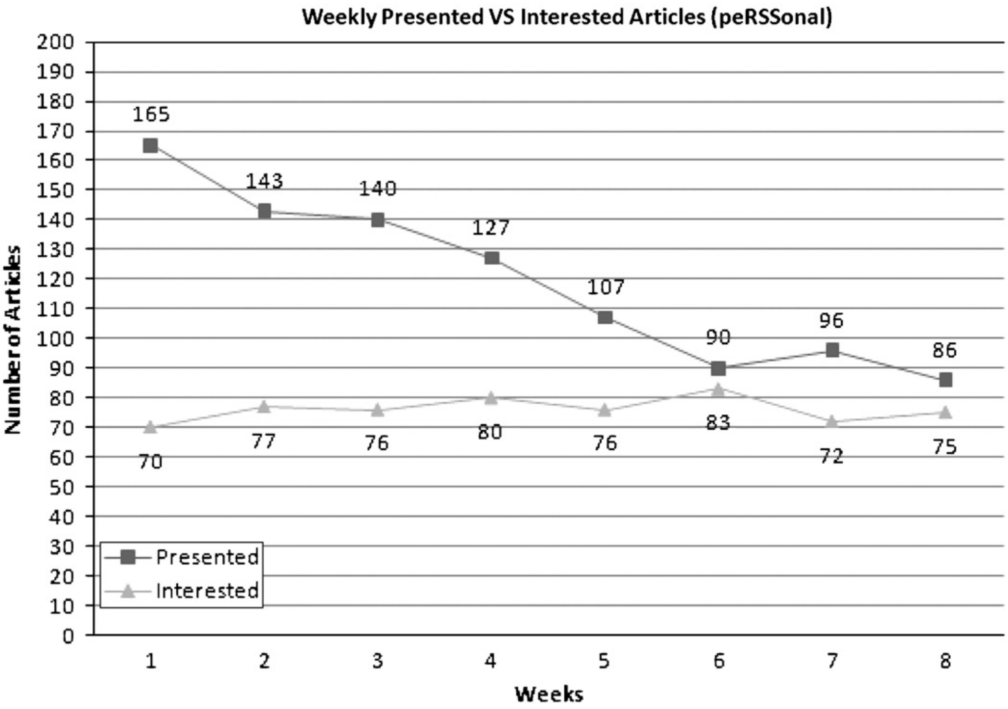


Fig. 13. Weekly adaptation of personal to the profile of the user.

clearing up of the feeds that the user is really interested to read (Fig. 11).

A second experimentation that is focused on the adaptability of the mechanism proves that the mechanism is able to produce even more effective results by establishing the exact profile of the user. The RSS feeds include the original URL of the HTML page that includes the article. Our RSS' URL includes a URL to our portal that redirects the user to the original HTML page. This is done in order to record the articles that the user selected to read. By monitoring the activity of the user we are able to create a list of keywords that represent her profile and thus in less than seven weeks time the system is adapted to the users needs.

As observed from the graphs of Figs. 12 and 13, which represent the system's adaptation to the user, the mechanism is able to adapt to the user's dynamic profile. What we want to achieve is the converging of the "presented" line to the interesting line as the number of articles that seem to be interesting to the user does not change through the time. The aforementioned graphs prove two basic functions of the mechanism. First, the mechanism is able to act as a filter so that articles that are of low interest to the user are not presented to him/her and second, the mechanism is able to dynamically create the profile of the end user and in less than seven weeks time present only the information that is important to the user.

An important factor to take in mind is that different users receive different RSS responses which vary in terms of news': length, ordering, amount, and categories. It is possible (and desirable) that two users receive different summaries from the same articles, as depicted in Fig. 6.

The mechanism is such that, extensive experiments can be carried out in order to observe every aspect of it and every possible state. Briefly, the PeRSSonal mechanism is able to create personalized, dynamically created RSS feeds with variant summary according to the user's profile and according to the end device of the user. The system is able to adapt on the specific user's needs and act as a complete personalized micro-site or personalized RSS in order to cover the needs of the most stern users.

6. Conclusions

In this manuscript, we presented a mechanism that is able to complete a procedure of collecting news from major news portals, categorize and summarize them, and present them personalized back to the end-users. PeRSSonal is enhanced with a personalization algorithm that is able to generate a fast-adaptable user profile that serves ideally the user with information of interest to her. The mechanism is improved significantly over its initial presentation, while its performance is much better than the two reference summarization models that it was compared with. This system is extremely helpful for the internet users who are spending a lot of time trying to find news of their interest through major or minor news portals or even through RSS feeds. Despite the fact that the personalized micro-sites that exist even within some portals resolve part of the problem, still the refinement of the results and the personalization on the specific device and the specific needs of the user is a huge problem. The procedure of accessing all the news portals in order to collect useful information is part of our everyday life, though, the information that is shown to the screen of the end user includes almost 80% of not needed, or even trash information.

7. Future work

Even though the evaluation of the mechanism revealed better results than other summarizers, it is important to note that PeRSSonal is designed to serve as a unified, multi-layered internet service. Each layer can be improved or replaced separately. For the future, we are considering some enhancement for the system. A news tracker system that will be able to track the changes on news articles from their origins; as more and more articles about a specific theme are published on several news portals or even on the same news portal, we should be able to collect all similar articles and present a summary of them back to the end user, providing also with the several links that the articles derive from and let the user make her choice on which link to follow.

Furthermore, we will be researching towards mining and incorporating in our presentation system, multimedia data extracted from news web pages.

Additionally, the automated procedure of collecting news articles will be empowered by a more effective focused crawler in order to avoid the collection of unneeded data, while at the same time we will enrich our system with multimedia content capabilities. The keyword extraction mechanism will also be extended to include multilingual support with the use of language-specific dictionaries, stemmers and stopword lists. Finally, since the system is capable of working at a very high speed, creating dynamically the RSS for users in real time, we are considering the creation of an add-on for news portals that will incorporate some of the PeRSSonal's procedures, e.g. enabling real-time creation of personalized RSS feeds for the end-user directly through existing portals.

References

Antonellis I, Bouras C, Poulopoulos V. Personalized news categorization through scalable text classification. In: Proceedings of the 8th Asia-Pacific web conference; 2006. p. 391–401.

Baron D. Persistent media bias. Journal of Public Economics 2006;90(1–2):1–36.

Bouras C, Dimitriou C, Poulopoulos V, Tsogkas V. The importance of the difference in text types to keyword extraction: evaluating a mechanism. In: Arabnia HR, editor. International conference on internet computing. Las Vegas, Nevada, USA: CSREA Press; 2006. p. 43–9.

Bouras C, Kounenis G, Misedakis I, Poulopoulos V. A web clipping services information extraction mechanism. In: Third international conference on universal access in human–computer interaction, Las Vegas, Nevada, USA, 22–27 July 2005.

Bouras C, Poulopoulos V, Tsogkas V. Efficient summarization based on categorized keywords. In: Stahlbock R, Crone SF, Lessmann S, editors. Dmin. Las Vegas, Nevada, USA: CSREA Press; 2007. p. 285–91.

Bouras C, Poulopoulos V, Tsogkas V. PeRSSonals core functionality evaluation: enhancing text labeling through personalized summaries. Data & Knowledge Engineering 2008;64(1):330–45.

Ferragina P, Gulli A. A personalized search engine based on web-snippet hierarchical clustering. In: Special interest tracks and poster proceedings of WWW-05, international conference on the world wide web, 2005. p. 801–10.

Fitzmaurice G, Zhai S, Chignell M. Virtual reality for palmtop computers. ACM Transactions on Information Systems (TOIS) 1993;11(3):197–218.

Goldstein J, Kantrowitz M, Mittal V, Carbonell J. Summarizing text documents: sentence selection and evaluation metrics. In: Proceedings of the 22nd annual international ACM SIGIR conference on research and development in information retrieval; 1999. p. 121–8.

Google news. 〈http://www.news.google.com/〉.

Gutwin C, Fedak C. Interacting with big interfaces on small screens: a comparison of fisheye, zoom, and panning techniques. In: Proceedings of the 2004 conference on graphics interface; 2004. p. 145–152.

Hayes P, Knecht L, Cellio, M. A news story categorization system. In: Proceedings of the second conference on applied natural language processing; 1988. p. 9–17.

Herman E. The propaganda model: a retrospective. Journalism Studies 2000;1(1): 101–112.

Hsu W, Lang S. 1999. Classification algorithms for NETNEWS articles. In: Proceedings of the 8th international conference on information and knowledge management; 1999. p. 114–21.

Kummamuru K, Lotlikar R, Roy S, Singal K, Krishnapuram R. A hierarchical monothetic document clustering algorithm for summarization and browsing search results. In: Proceedings of the 13th conference on world wide web; 2004. p. 658–65.

Radev D, Otterbacher J, Winkel A, Blair-Goldensohn S. NewsInEssence: summarizing online news topics. Communications of the ACM 2005;48(10):95–8.

RSS. Rdf—resource description framework 〈http://www.w3.org/RDF/〉.

Wasson M. Using leading text for news summaries: evaluation results and implications for commercial summarization applications. In: Proceedings of the 17th international conference on computational linguistics, vol. 2; 1998. p. 1364–8.

XML. Extensible markup language 〈http://www.w3.org/XML/〉.