Evaluating the Unification of Multiple Information Retrieval Techniques into a News Indexing Service

Christos Bouras[1,2] and Vassilis Tsogkas[1]
[1]*Computer Engineering and Informatics Department, University of Patras, Patras, Greece*
[2]*Computer Technology Institute and Press "Diophantus", 26500 Rion, Patras, Greece*

Keywords: Clustering, Text Preprocessing, User Personalization, n-Grams, W-kmeans.

Abstract: While online information sources are rapidly increasing in amount, so does the daily available online news content. Several approaches have being proposed for organizing this immense amount of data. In this work we explore the integration of multiple information retrieval techniques, like text preprocessing, n-grams expansion, summarization, categorization and item/user clustering into a single mechanism designed to consolidate and index news articles from major news portals from around the web. Our goal is to allow users to seamlessly and quickly get the news of the day that are of appeal to them via our system. We show how, the application of each one of the proposed techniques gradually improves the precision results in terms of the suggested news articles for a number of registered system users and how, aggregately, these techniques provide a unified solution to the recommendation problem.

1 INTODUCTION

During the last decade, the advances in information technology and the mere increase of devices capable of access to information have changed dramatically the access to the World Wide Web. The Internet has long surpassed the printer as a medium for news delivery and will soon be doing the same with the television. Additionally, more and more people, apart from reading news articles on their PCs, want to utilize their mobile devices for this scope. The aforementioned situation generates a huge problem commonly internet users have come across: locating useful news articles that match their needs, on a daily basis, can be a time-consuming and frustrating experience.

Over time, several approaches have been proposed in order to deal with the afore-mentioned situation within the scope of recommendation systems. These approaches generally fall into two categories: content based and collaborative. Content-based recommenders (Pazzani, 2007) typically analyze a set of objects, usually textual descriptions of the items previously rated or viewed by a user, and build a model of user interests based on the features of the objects rated by that user. Thus, the user profile is a structured representation of the user interests which is exploited to recommend new potentially relevant items. Collaborative filtering techniques on the other hand, (Lops, 2007), predict the preferences of a user for an item by weighting the contributions of similar users, called neighbors, for that item. Similarity between users is computed by comparing their rating styles, i.e. the set of ratings given on the same items or by means of their browsing habits.

Our recommendation approach can be classified as 'hybrid' since it is mainly content-based with some collaborative filtering features that enhance the algorithm with the ability to automatically adapt over time to the continuously changing user choices. As in the works of (Kim et al., 2006; Yu et al., 2003) and in contrast to other single-minded CF techniques, we derive the item groups by arranging the information that is extracted from several information retrieval (IR) techniques, like clustering and also inferred by previous user behavior. We also incorporate various techniques such as text pre-processing, personalization in order to assist our recommender.

Text pre-processing, commonly being the first step after content fetching, is an important function for any news indexing system. A series of methodologies, including noun processing and part of speech tagging, as presented in (Bouras, 2008), are employed in this step in order for the text to be

Bouras C. and Tsogkas V..
Evaluating the Unification of Multiple Information Retrieval Techniques into a News Indexing Service.
DOI: 10.5220/0004998000330040
In *Proceedings of 3rd International Conference on Data Management Technologies and Applications* (DATA-2014), pages 33-40
ISBN: 978-989-758-035-2

'cleaned-up' adequately. As an outcome of text pre-processing, the text's representation is able to feed the follow up techniques in a way that their results are significantly benefited. Among these methodologies are:

- Useful text extraction, where only the useful text of a web page is kept by removing videos, pictures, etc.
- Stop words removal, where only the words that 'make sense' are kept. This means that words like articles, numbers, pronunciations, etc. are stripped off.
- Stemming, where only the lexicographical root of the word is kept.

Clustering methodologies are used in order to partition a given data set into similar subsets, by defining some notion of similarity/dissimilarity (Hand, 2001). In general, several metrics over which a distance measure can be defined are associated to items in the data set; informally, the partitioning process tries to put in the same subset neighboring samples and in different subsets distant samples. Several heuristics have been applied into this area, combining agglomerative and partitional algorithms (Bianco, 2005), or WordNet synsets (Lops, 2007). Moreover, several approaches have been applied to the domain of user clustering, e.g. (Ntoutsi, 2012; Tang, 2005).

Into this area, we proposed (Bouras and Tsogkas, 2010) a new clustering method, called W-kmeans, which improves the traditional k-means algorithm by enriching its input with WordNet hypernyms. The WordNet lexical reference system, organizes a variety of linguistic relations into hierarchies/hypernyms (Is-a relation) and W-kmeans uses them as a pre-processing stage before the regular k-means algorithm. We extended this algorithm to the domain of user clustering (Bouras and Tsogkas, 2011), where we investigated how user clustering alone can affect the recommender's performance.

Personalized search (White, 2013) is another important research area that aims to resolve the ambiguity of query terms. To increase the relevance of search results, personalized search engines create user profiles that capture the user's personal preferences and, as such, identify the actual goal of the input query. Their conclusion was that positive preferences are not enough to capture the fine-grained user interests.

User profiling strategies can be broadly classified into two main approaches: the document-based one and the concept-based one. Document-based user profiling methods aim at capturing the user's clicking and browsing behaviors. User's document preferences are first extracted from the click through data and then used to learn the user behavior model which is usually represented as a set of weighted features. Concept-based user profiling methods aim at capturing user's conceptual needs.

The contribution of the current work is twofold. Firstly, we present how the suggested methodologies that we have implemented and published over the last 4 years are aggregated towards the direction of the useful news delivery problem. Secondly, to evaluate, step by step, how the appliance of each methodology affects positively the effectiveness of the system. In essence, we are evaluating a real-world application which applies several state of the art techniques on data gathered from users browsing and utilizing our platform as a tool for their day to day news consumption.

The rest of the manuscript is structured as follows. Section 2 gives an overview of the information flow while section 3 goes more in-depth into the analysis of the proposed system. Section 4 presents our experimental results regarding the system's evaluation. Finally, in section 5 we present the conclusions of the current paper and propose some key areas for further research.

2 FLOW OF INFORMATION

Figure 1 gives an overview of the information flow that we are making use of for our indexing and news recommendation service. As its input, our system fetches news articles generated by news sources from around the Web. This is an offline procedure running constantly into the background and once articles as well as metadata associated with the news articles are fetched, they are stored in the centralized database from where they are picked up by the procedures that follow. The article's metadata include information such as: its title, its date of publication, associated images and much more. This information is used in one way or another by our presentation module when articles are selected for viewing by the system users.

Following the article fetching, keyword extraction takes place with several dimensionality reduction as well as feature expansion heuristics. Next come the core IR processes of our system, namely: summarization, categorization, item (news article) and user clustering. The results from all those processes are utilized by the personalization module of our system in order to decide upon the items that are appropriate for each particular system

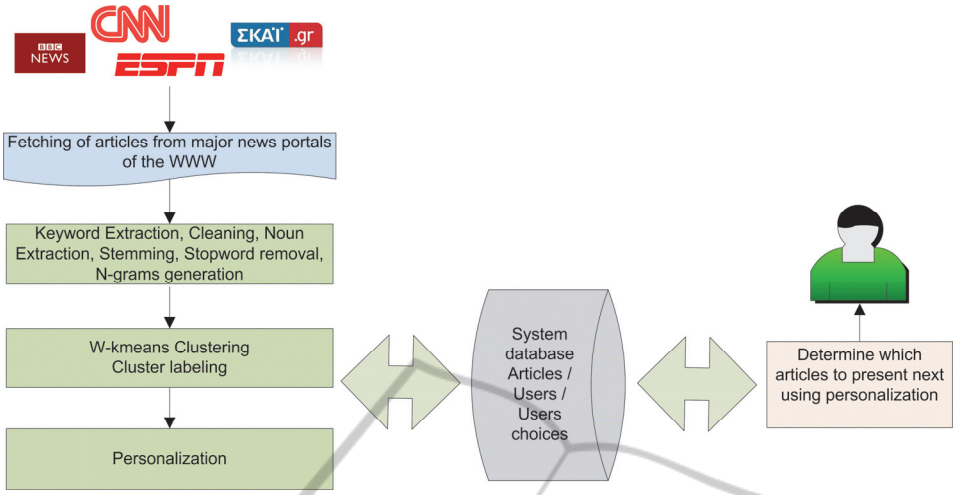

Figure 1: Flow of Information.

user. A high-level analysis of the afore-mentioned procedures is given on the next section. Note that since each one of these methodologies have been previously covered in the literature by the respective authors and the purpose of this paper is to cover the unification of these methodologies (as well as evaluate them) and thus the reader is urged to study the cited papers for a more in-depth coverage of each of these processes.

3 ANALYSIS

Keyword extraction (and text preprocessing in general) is a key process of the system as a whole. It is applied on the fetched article's content and results to the set of keywords that each article consists of. At this analysis level, we apply some typical dimensionality reduction techniques, which include: *stemming*, *stop word removal* and *filtering of low frequency words*. In addition to the above, we also utilize techniques regarding:

- *Feature Selection / Reduction,* where we attempt to select a subset of the textual features that are most useful for the IR tasks that follow. This is achieved: *a)* via *POS tagging and noun extraction* (Bouras and Tsogkas, 2008), and *b)* via *pruning of words*, appearing with low frequency throughout the corpus, which are unlikely to appear in more than a small number of articles. These textual elements are usually the result of 'noise' i.e. typographical errors of the source or errors during the fetching / parsing phase
- *Feature Generation / Extraction,* where new features, are sought for representation. In our case, this is achieved via *WordNet generated hypernyms*. We are employing a novel approach that extends the standard k-means algorithm by using the external knowledge from WordNet hypernyms for enriching the *"bag of words"* (BOW) representation generated prior to the core IR processes. Initially, for each given keyword of the article, we generate its graphs of hypernyms leading to the root hypernym (commonly being 'entity' for nouns). Following, we combine each individual hypernym graph to an aggregated one, essentially an article-hypernym tree. There are practically two parameters that need to be taken into consideration for each hypernym of the aggregate tree-like structure in order to determine the hypernym's importance: the depth and the frequency of appearance. Given these two parameters, we can estimate each hypernym's weight using formula (1)

$$W(d,f) = 2 \cdot \frac{1}{1+e^{-0.125(d^3 \frac{f}{TW})}} - 0.5 \quad (1)$$

where d stands for the node's depth in the aggregate graph, f is the frequency of appearance of the node to the multiple graph paths and TW is the number of total words that were used for generating the graph (i.e. total keywords in the article).

Keyword extraction, thus making use of the *vector space model*, generates the term-frequency vector which describes each article as a BOW representation (words – frequencies vector) to the key information retrieval techniques that follow: article categorization, text summarization and clustering.

In our previous research (Bouras and Tsogkas, 2013), we enhanced the feature generation process of the preprocessing stage by also extracting n-

grams from the text and indexing them into the database. The process is similar to that of keyword extraction, which could actually be thought of as the trivial case of n-gram extraction (with n=1). For each article and for values of n from 2 to 6, we identify the *word n-grams* of the input text and properly index them into our database. In this scenario, the overall similarity, either between two articles or between an article and a class, i.e. category or cluster, is not portrayed only in terms of keyword the frequency / inverse document frequency (*kf-idf*) numerical statistic, but rather as a combination of *kf-idf* and its n-gram counterpart statistic, let's call it: gram frequency / inverse document frequency, *gf-idf*. Note also that stemming is not being applied on the extracted n-grams since stemming techniques are negating any attempt to gather useful word n-grams of any size (stemming is applied though on words not part of the extracted n-grams). This *gf-idf* weight for each n-gram j could be expressed as shown in equation (2)

$$W_{ngj} = gf - idf_j = freq_j * \log\frac{N}{M} \quad (2)$$

where N is the total number of articles in the database and M is the total number of articles containing the n-gram j. Given the above, we could express the total weight of each sentence, given its keywords and n-grams as in equation (3).

$$Si = A*(\sum w_{k,i} * \gamma_{k,i}) + B*\sum w_{ngj} \quad (3)$$

where i is the sentence under question, $w_{k,i}$ is the *kf-idf* of keyword k in sentence i, $\gamma_{k,i}$ the weighting factor of each keyword k depending on:

- its relative frequency of appearance on the article's body
- its relative frequency of appearance in the article's title and
- its impact on the categorization and summarization sub-processes.

Usually, a keyword belonging to the article's title should be more important in conveying the sentence's / article's meaning. Furthermore, as the summarization procedure of our system is based on the selection of the most representative sentences via means of weighting them, the categorization outcomes can be helpful in adjusting more effectively the weighting of the sentences. Common sense implies that a keyword that has very high frequency for a specific category, should give more weight to the sentence in which it appears, while a keyword that has small or zero frequency for a category could add less to the weight of a sentence. Intuitively, a keyword that is included into the extracted ones of an article that is representative of a category other than the one that the article is in, would give negative weight to the sentence. Summarizing the above, the weighting factor of each keyword k belonging to sentence i, $\gamma_{k,i}$ could be expressed like so:

$$\gamma_{k,i} = (\alpha * F_k + \beta * F_{k,title}) * Y * (Z(i,g)) \quad (4)$$

with:

- α and β being the weighting factors of the importance for the keyword i to belong to the article / article's title respectively. In our research: $\alpha=0.1$ and $\beta=0.9$).
- F_k being the relative frequency of keyword k into the given news article,
- $F_{k,title}$ being the relative frequency of the keyword k into the article's title,
- Z(i,g) being the relative frequency of appearance of keyword i into the category g that the article was found to belong to,
- Y being the weighting factor of the categorization / summarization effect. In our research: $Y=1$

Finally, from formula (3), w_{ngi} is the weight of n-gram j (*gf-idf* statistic). We can normalize the effect that the keywords and n-grams have on the keyword extraction process in a linear way by using the two parameters A, B, mentioned in (3), so that:

$$W'_{kwi} = W_{kwi} * A \quad (5)$$

$$W'_{ngi} = W_{ngi} * B \quad (6)$$

and:

$$A + B = 1 \quad (7)$$

In our previous research we have determined that the best results for the domain of news articles are: $B=0.3$ and $A=0.7$.

The generated enhanced feature list feeds the k-means clustering algorithm that follows. It is important to note, however, that the clustering process is independent from the rest of the steps, meaning that it can easily be replaced by any other clustering approach. As a result of the above, our clustering algorithm operates using a) *keywords*, b) *enriched hypernyms* from keywords and c) *previously extracted word n-grams*. Given the number of desired clusters, let k, partitional algorithms like k-kmeans, find all k clusters of the data at once, such that the sum of distances over the

items to their cluster centers is minimal. Moreover, for a clustering result to be accurate, besides the low intra-cluster distance, high inter-cluster distances, i.e. well separated clusters, are desired.

The personalization module that follows, can easily adapt to subtle user preference changes. Those changes, as expressed by the user's browsing behavior, are detected and continuously adjust the user's profile. The algorithm uses a variety of user-related information in order to filter the results presented to the user. Furthermore, it takes into account in a weighted manner the information originating from the previous levels regarding the summarization / categorization and news / user clustering steps.

User profiles from multiple users and timeframes are then clustered using our clustering algorithm forming profile clusters. Given a user *u* and a set of news articles *R* on which *u* provided, either implicitly or explicitly, a positive or negative feedback according to his/her interests (positive or negative respectively), a user profile Up is maintained, analyzed by two parts. The positive part, U_p^+ consists of keywords from news articles judged positively by *u*, while the negative part, U_p^- consists of keywords from news articles judged negative judged by *u*. Our algorithm enhances the user profiles with hypernyms deducted from the WordNet database, using a heuristic manner similar to that of article clustering described earlier. Those profile clusters are used at the recommendation stage in order to enhance the system's usage experience by providing more adapted results to users revisiting the site. When a user comes back, her clustered profile is recalled. Articles matching her profile are then extracted and considered for user recommendations. The above algorithm steps are thoroughly presented by (Bouras and Tsogkas, 2011).

4 EXPERIMENTS AND EVALUATION

In order to evaluate the performance and accuracy of our platform, we conducted a series of experiments. The experimental flow was as follows:

- Evaluate current performance
- Apply new methodology
- Re-evaluate and compare results

What came out of this approach was a set of data showing the overall trend with regards to the specific criteria / evaluation metric that we used to evaluate our system.

For our experimentation we analyzed the logs regarding the browsing patterns, as well as the article recommendations, offered to *30 of the registered system users*. The users had been using the system through its various stages, i.e. the techniques described in section 3 were applied or not applied without the users having any a-priori knowledge about the system changes. The user selections as well as the system's recommendations were recorded throughout this process. Due to the nature of news article, which have to be *'new'*, our system would disregard those beyond a pre-defined time period (3 months) since those would not have any real merit in terms of recommendation usage, plus, they wouldn't have much chance of being selected by the users. As a result, even though the total indexed articles in our system is over *50,000* articles, the actual amount of articles, i.e. the used corpus, was *3,000 recent articles* belonging to various fields of interest: politics, technology, sports, entertainment, economy, science and education.

One of the most widely used evaluation metrics for predicting performance of recommender systems is Mean Absolute Error (MAE). MAE, expresses the average absolute deviation between predicted and true ratings and can be computed using formula (8).

$$MAE = \frac{\sum r'(u,i) \in R' \mid r(u,i) - r'(u,i) \mid}{\mid R' \mid} \quad (8)$$

r(u, i) being the preference of user u for article *i* and *r'(u,i)* the predicted / recommended preference for user *u* of articles belonging to *R'*.

For our first experiment, we measured the MAE between the actual selections of the users, i.e. the articles that were selected for viewing and the article recommendations made by our system in the various stages:

1. Without any heuristics and by suggesting news articles only recently added to the database.
2. When keyword extraction and article categorization was utilized for the recommendations.
3. When in addition to the methodologies exercised in 2, article clustering was also utilized for the recommendations.
4. When in addition to the methodologies exercised in 3, user clustering was also utilized for the recommendations.
5. When in addition to the methodologies exercised in 4, n-grams extraction was also utilized for the recommendations.

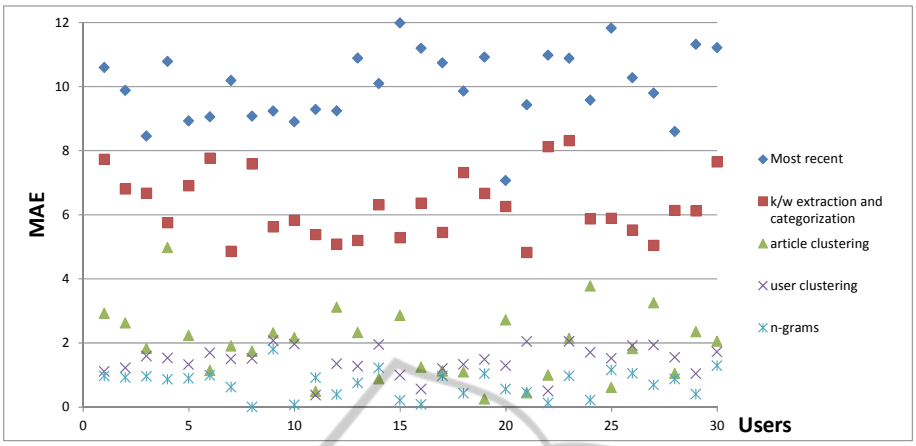

Figure 2: MAE of recommendations when various techniques are applied.

Fig. 2 depicts the MAE results that were obtained during this experimental procedure. From it, we can observe a general tendency of significant decline in the MAE values both gradually after the application of each technique, and aggregately when all of them were applied.

Table 1: MAE normal distribution values over all 30 users.

Approach	Mean Value	Standard Deviation
Most recent	10	1.1
K/W Extraction and Categorization	6	1.1
Article Clustering	2.5	0.9
User Clustering	1.5	0.5
n-grams	0.5	0.5

More precisely, the average MAE scores were reduced from 10.01 when no heuristic / technique were applied (recommending only latest articles) to 6.28 when k/w extraction and categorization was exercised. What this means from a physical point of view is that when k/w extraction and categorization was applied, there were 6.28 mis-recommendations made by the system to the users reading articles. Article clustering reduced dramatically the average MAE scores to 1.95. The above was a significant improvement which was also observed in our previous work. Following, the appliance of user clustering reduced the average MAE scores to 1.44 and finally, when making use of n-grams, the average MAE scores dropped even further to 0.73, that is on average there was less than 1 mis-recommendation by the system on any single user. Another observation regarding the previous data was that by generating a normal probability plot of the discovered MAE values shows that the data line up along the diagonal, and are very close to a normal distribution using the mean and standard deviation values provided in Table 1. Note however that a proof of the normal distribution of the data of this experiment is beyond the scope of this paper.

For our second experiment, we tried to evaluate the overall performance and efficiency improvement of our system when each of the methodologies is applied. As an evaluation metric we used the *F-measure*, as defined in formula (9). The F-measure is a weighted combination of the precision and recall metrics (the harmonic mean) giving a good estimate of an IR system's performance.

We define a set of target articles, denote C, that the system suggests and another set of articles, denote C', that are visited by the user after the recommendation process. Moreover, $|c'_i,c_j|$ is used to denote the number of documents both in the suggested and in the visited lists.

$$F(c'_i,c_j) = 2 \cdot \frac{r(c'_i,c_j)p(c'_i,c_j)}{r(c'_i,c_j)+p(c'_i,c_j)} \quad (9)$$

where:

$$r(c'_i,c_j) = \frac{|c'_i,c_j)|}{|c'_i|} \quad (10)$$

$$p(c'_i,c_j) = \frac{|c'_i,c_j|}{|c_i|} \quad (11)$$

Using the same user browsing data as in the previous experiment, we extracted the average F-measure results for all users over a period of 50 days of system usage. For this experiment, every 10 days each of the methodologies described in section 3 was aggregately applied to the recommendation engine, i.e. the recommendations were as explained in Table 2.

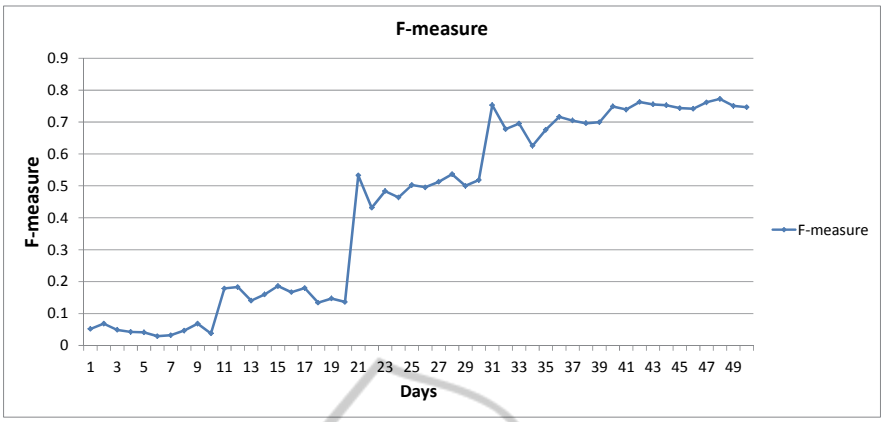

Figure 3: Average F-measure results for user recommendations when various techniques are applied.

Table 2: Switching Recommendation approaches based on time.

Days	Recommendation approach
1-10	only latest articles
11-20	keyword extraction and article categorization assisted methodology
21-30	article clustering also applied
31-40	user clustering also applied
41-50	n-grams extraction also applied

The produced F-measure results for the system's recommendations, averaged over all users, are depicted in Fig. 3.

From the above results we can observe that the recommendations which utilize all of the suggested methodologies significantly outperform each one of them individually. More specifically, while the average F measure results starts from around *0.05* when only recent articles are recommended, it approaches the value of *0.8* when n-grams extraction is also applied. Again, article clustering gave a significant boost with regards to the system's performance: from *0.15* when only k/w extraction and categorization was exercised, to *0.47*. Another observation is that, the improvement gets generally greater after some days of system usage. The above has two explanations: a) the system has more data regarding the user's choices/preferences, and b) the system has more time to generate more coherent and generally better user clusters. Initially the F-measure scores are lower due to the fact that the recommender hasn't yet determined the user profiles to an acceptable extend.

The above results also have straightforward implications from a physical point of view in terms of the quality of the suggested content: news articles are generally of great interest to the users, matching their profiles, and almost all of them are selected for viewing, even at a later stage. Another important observation is that the recommendations generally stabilize (to the average value of the same methodology) within the 10 days' timeframe, without much deviation. This is explained by the fact that that our recommendation engine converges fast to the divertive user interests, a result noted also in Bouras and Tsogkas (2011).

5 CONCLUSIONS AND FUTURE WORK

In the current work we have presented an overview of the various components of our system, a news indexing and personalization service. The methodologies used, autonomously presented in the literature before, were evaluated in terms of the system's recommendation ability both in an add-on manner, applying each one on top of the previous ones, and aggregately. A generic enough flow of information was presented, not bound to the actual algorithms / subsystems.

During our experimentation with a number of system users, we have found that the performance gains were getting bigger as each technique was applied. In terms of MAE scores, the average value dropped from *10.1*, when results were presented without any particular criterion (using latest news articles) to *0.73* when all of our techniques and heuristics were applied. A similar improvement was observed over a time period of 50 days aggregately for the same user base in terms of the F measure metric which went from *0.05*, when using latest news articles to *0.8*, when all our techniques and heuristics were applied

In our opinion the above results statistically show that recommendation systems can leverage much more from the appliance of various methodologies which are working in tandem in a heuristic fashion, as opposed to single-minded approaches, however smart or well implemented, when applied autonomously. In essence, collaborative approaches work far better when combined with content-based ones (i.e. k/w extraction techniques).

For the future we are planning to improve each of the presented methodologies in order to fine tune them and achieve even better results. Furthermore, we intent to extend our experimentation to a larger user database and corpus statistically analyzing the results that we shall obtain to a greater extend.

ACKNOWLEDGEMENTS

This research has been co-financed by the European Union (European Social Fund – ESF) and Greek national funds through the Operational Program "Education and Lifelong Learning" of the National Strategic Reference Framework (NSRF) - Research Funding Program: Heracleitus II. Investing in knowledge society through the European Social Fund.

REFERENCES

Bouras, C., & Tsogkas, V., 2008. Improving text summarization using noun retrieval techniques. *In Knowledge-Based Intelligent Information and Engineering Systems* (pp. 593-600).

Bouras, C., & Tsogkas, V., 2010. W-kmeans: clustering news articles using WordNet. *In Knowledge-Based and Intelligent Information and Engineering Systems* (pp. 379-388).

Bouras, C., & Tsogkas, V., 2011. Clustering user preferences using W-kmeans. *In Signal-Image Technology and Internet-Based Systems (SITIS)*, 2011 Seventh International Conference on (pp. 75-82). IEEE.

Bouras, C., & Tsogkas, V. Enhancing news articles clustering using word n-grams. 2013. *2nd Intenational Conference on Data Management Technologies and Applications*, Reykjavvk, Iceland

Bianco, A., Mardente, G., Mellia, M., Munafo, M., & Muscariello, L., 2005. Web user session characterization via clustering techniques. *In Global Telecommunications Conference*, 2005. *GLOBECOM'05. IEEE* (Vol. 2, pp. 6-pp). IEEE.

Hand, D. J., Mannila, H., & Smyth, P., 2001. *Principles of data mining*. MIT press.

Kim, B. M., Li, Q., Park, C. S., Kim, S. G., & Kim, J. Y. (2006). A new approach for combining content-based and collaborative filters. Journal of Intelligent Information Systems, 27(1), 79-91.

Lops, P., Degemmis, M., & Semeraro, G., 2007. Improving social filtering techniques through WordNet-Based user profiles. *In User Modeling 2007* (pp. 268-277).

Ntoutsi, E., Stefanidis, K., Nørvåg, K., & Kriegel, H. P., 2012, Fast group recommendations by applying user clustering. In Conceptual Modeling (pp. 126-140).

Moore, R., Lopes, J., 1999. Paper templates. In *TEMPLATE'06, 1st International Conference on Template Production*. SCITEPRESS.

Pazzani, M. J., & Billsus, D., 2007. Content-based recommendation systems. *In The adaptive web* (pp. 325-341).

Smith, J., 1998. *The book*, The publishing company. London, 2nd edition.

Tang, N., & Vemuri, V. R., 2005. User-interest-based document filtering via semi-supervised clustering. In Foundations of Intelligent Systems (pp. 573-582).

White, R. W., Chu, W., Hassan, A., He, X., Song, Y., & Wang, H., 2013. Enhancing personalized search by mining and modeling task behavior. *In Proceedings of the 22nd international conference on World Wide Web* (pp. 1411-1420). International World Wide Web Conferences Steering Committee.

Yu, K., Schwaighofer, A., & Tresp, V. (2002, August). Collaborative ensemble learning: Combining collaborative and content-based information filtering via hierarchical Bayes. In Proceedings of the Nineteenth conference on Uncertainty in Artificial Intelligence (pp. 616-623).