INTERSOCIAL-I1-1.2, Subsidy Contract No. <I1-12-03>, MIS Nr 902010

European Territorial Cooperation Programme

Greece-Italy 2007-2013

INTERSOCIAL: Unleashing the Power of Social Networks for Regional SMEs

Deliverable D3.4.2: Tools for Enhancing SMEs Social Presence

Action 3.4: Development of Innovation Devices

WP3: Social-Oriented Product Promotion Mechanisms

Priority Axis 1: Strengthening competitiveness and innovation
Specific Objective 1.2: Promoting cross-border advanced new technologies

Financed by the European Territorial Cooperation Operational Programme "Greece-Italy" 2007-2013, Co-funded by the European Union (European Regional Development Fund) and by National Funds of Greece and Italy

Tools for Enhancing SMEs Social Presence

Deliverable D3.4.2 Action 3.4

Workpackage WP3: Development of Innovation Devices

Responsible Partner:	UOI (LP)		
Participating Partner(s):			
SAT:	APCE (P4)		
WP/Task No.:	WP	Number of pages:	15
Issue date:	2013/8/31	Dissemination level:	Public

Purpose: Developing tools for enhancing the presence of SMEs in social sites
Results: The development of a new tool, called POIKILO, that allows the visualization of the collected datasets to allow a deeper understanding of their meaning
Conclusion: POIKILO is available for use to analyze data

Approved by the project coordinator: Yes
Date of delivery to the JTS/MA: 20/9/2013

Document history

When	Who	Comments
15/9/2013	E. Pitoura, E. Koletsou	Initial version
16/9/2013	I. Fudos	Editing

Table of Contents

Section 1 Introduction

In recent years, we have witnessed an unprecedented growth of social networking. Social networks offer new means and forums for world-wide product promotion as well as huge repositories of data for advanced market analysis and trend identification

The InterSocial project, funded in the frame of the Territorial Cooperation Program aims at exploring social networking to enhance the competitiveness of small and medium enterprises (SMEs) in neighboring regions of western Greece and south Italy. In particular, the project aims at promoting advanced new technologies as related to the use of social networking to both (a) improve the web presence of SMEs and (b) use information provided by such sites for targeting advertisement and adaptive service provision.

In this report, we present a tool, termed *POIKILO,* that allow users to visualize datasets collected by social analysis tools for further processing.

POIKILO offers users a graphical representation of their datasets, so that they can extract valuable information. A special feature of the tool is that in the case of large datasets (which is often the case with social network data), the tool selects the most representative ones to present to the user based on their relevance and diversity.

Data diversification has attracted considerable attention recently as a means of enhancing the quality of results presented to users. Consider, for example, data collected about a specific keyword mentioned in a number of tweets. Presenting representative tweets such as tweets posted by users at different locations, or by users with different demographic characteristics (such as age, sex, education) are more valuable than presenting homogeneous results.

There have been various definitions of diversity [DP10], based on (i) *content* (or similarity), i.e., selecting items that are dissimilar to each other (e.g., [ZMK+05]), (ii) *novelty*, i.e., selecting items that contain new information when compared to what was previously presented (e.g., [CKC+08]) and (iii) *semantic coverage*, i.e., selecting items that belong to different categories or topics (e.g., [AGH+09]). Most approaches rely on assigning a diversity score to each item and then selecting either the k items with the highest score for a given k (e.g., [AK11,FMT12, QYC12]) or the items with a score larger than some threshold (e.g., [YLA09]). Alternatively, in [DP12A], a tuning parameter called *radius* explicitly expresses the desired degree of diversification which determines the size of the diverse set presented to the users.

Different diversification methods aim at optimizing different diversification criteria. Often, it is not clear what method is more suitable for a specific application.

POIKILO assist users in locating, visualizing and comparing diverse results based on a suite of different diversification models and algorithms. We provide implementations of a wide variety of diversification approaches for retrieving diverse results. For the case in which the degree of diversification is specified by a radius, we also provide an interactive zoom-in and zoom-out form of functionality.

Often, results are associated with a *relevance score*. POIKILO includes various methods for combining relevance and diversity in selecting representative data items.

Furthermore, we consider the case of *streaming* data, where data change over time, such as in the case of social network postings. In this case, the representative items to be presented change as well. We employ a sliding window streaming model and provide options to navigate between consequent windows of streaming datasets.

This report provides a description of the tool. In the rest of the report, in Section 2, we provide the theory behind POIKILO, in particular the models and algorithms supported by the tool, while Section 3 includes a short manual of how to use the tool. A demo is also available.

Section 2 Technical Details: Models and Algorithms

In this section we present the theory behind POIKILO. In particular, we briefly introduce the related diversification models and algorithms. Most of these models involve the use of a distance function. We have implemented the most common distance functions (e.g., Euclidean, cosine). In addition, users can select which of the attributes of each item will be used for diversification. We also introduce relevance and streaming.

Diversification models

The most widespread diversity models are related to the *k-dispersion problem*, defined as selecting k out of a set P of data items in some space, such that some objective function is maximized. Common objective functions include MaxMin and MaxSum. Given a distance metric d and an integer k, $k > 1$, MaxMin aims at locating a subset S of P with k representative of P data items, such that, the minimum pairwise distance among any items in S is maximized, whereas, MaxSum aims at maximizing the sum of the respective pairwise distances.

In particular, MaxMin aims at maximizing the function f_{min} and MaxSum the function f_{sum} defined as follows.

$$f_{min}(S, d) = \min_{p, q \in S} d(p, q)$$
$$f_{sum}(S, d) = \sum_{p, q \in S} d(p, q)$$

Intuitively, MaxMin aims at discouraging the selection of nearby items, while MaxSum at increasing the average pairwise distance among all items.

DisC is a recently proposed model that combines coverage and diversity [DP12A]. Let $N_r(p)$ be the neighborhood of a data item p , i.e., the items lying at distance at most r from p, where $r \geq 0$ is a tuning parameter called *radius*. Let also $N_r^+(p)$ be the set $N_r(p) \cup p$. Intuitively, we would like to select exactly one item from the neighborhood of each item p in the dataset.

Formally,

Let P be a set of data items and r, $r \geq 0$ a real number. A subset $S \subseteq P$ is an r-Dissimilar and Covering subset, or an *r-DisC* diverse subset, of P if the following two conditions hold:

(i) (coverage condition) $\forall\, p \in P$, there exists $q \in N_r^+(p)$ such that $q \in S$, and

(ii) (dissimilarity condition) $\forall\, p, q \in s$, with $p \neq q$, it holds that $d(p, q) > r$.

For any dataset P, we would like to select the smallest number of dissimilar and covering items to represent it. Thus, we define the minimum DisC subset problem as follows.

Given a set *P* of data items and a radius *r*, the minimum DisC subset problem is to find an *r*-DisC diverse subset *S** of *P*, such that, for every *r*-DisC diverse subset *S* of *P*, it holds $|S^*| \leq |S|$.

The DisC model allows an interactive mode of operation where, after being presented with an initial set of data items for some radius *r*, a user can see either more or less data items by selecting a new radius r'. Decreasing r (i.e., selecting r' < r) results in selecting more items or *zooming in*, while increasing r (i.e., selecting r' > r) results in selecting more items or *zooming out*. Zooming can be global, in the sense that the radius *r* is modified similarly for all items in *P*, or local, i.e., modifying the radius only for a specific area of the data set.

To support an incremental mode of operation, the new set *S'* of data items (corresponding to the new radius *r'*) presented to the user should be as close as possible to the already seen result *S* for radius *r*. $. Ideally, $S' \supseteq S$, when $r' < r$ and $S' \subseteq S$ when $r' > r$. Although in general there is no monotonic property among the optimal *r*-DisC diverse and *r'*-DisC diverse subsets of a set of items *P*, we provide heuristics to achieve these requirements.

Often, *clustering* has been proposed as an alternative to selecting representative diverse items. In this case, the diverse set consists of representatives from each cluster. For example, *k-medoid* seeks to minimize $1/|P| \sum_{p \in P} d(p, c(p))$, where $c(p)$ is the closest item of *p* in the selected subset.

We also consider other diversification models, such as the *Greedy Marginal Contribution* and *Greedy Randomized with Neighborhood Expansion* models introduced in [VRB+11] POIKILO can be easily extended with additional methods as well.

Figure 1 shows the represented diverse subset computed by POIKILO for MaxMin and MaxSum and Figure 2 for DisC and *k*-medoids. Generally, MaxSum and *k*-medoids fail to cover all areas of the dataset; MaxSum tends to focus on the outskirts of the dataset, whereas k-medoids clustering reports only central points, ignoring sparser areas. MaxMin performs better in this respect. However, since MaxMin seeks to retrieve objects that are as far apart as possible, it fails to retrieve objects from dense areas; see, for example, the central areas of the clusters in Figure 1. DisC gives priority to such areas and, thus, such areas are better represented in the solution. Note also that MaxSum and k-medoids may select near duplicates, as opposed to DisC and MaxMin.

Algorithms

Due to the NP-hardness of most of the models of the diversification problem, a number of different heuristics have been proposed (e.g., see [EUY94]). POIKILO provides various implementations of different variations of such heuristics.

For MaxMin and MaxSum, a simple iterative greedy heuristic has been shown to provide ½ approximations of the optimal solution. In this heuristic, first, the two furthest apart items of P are added to S. Then, at each iteration, one more item is added to S. The item that is added is the one that has the maximum distance from the items already in S. Interchange heuristics are often used as well. Such heuristics are initialized with a random solution S and then iteratively attempt to improve that solution by interchanging an item in the solution with another item that is not in the solution. Usually, the item that is eliminated from the solution at each iteration is one of the two closest items in it. We provide various interchange heuristics, e.g., performing at each iteration the first interchange that improves the solution (First-Interchange) or considering all possible interchanges and perform the one that improves the solution the most (Best-Interchange).

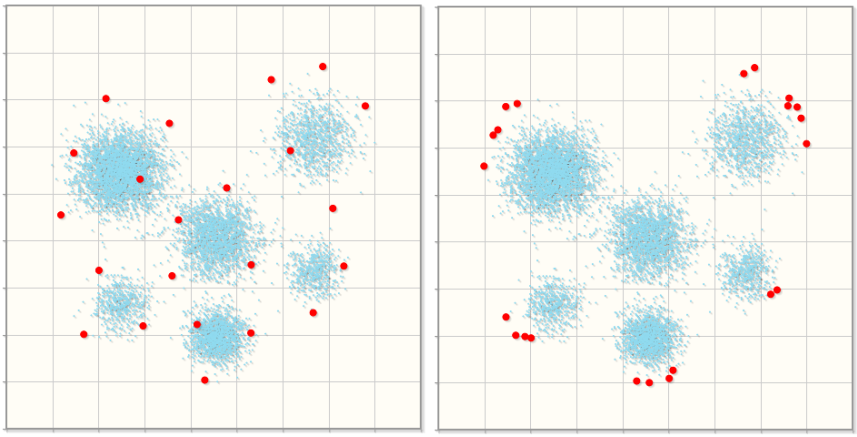

Figure 1: POIKILO results using MaxMin (left) and MaxSum (right), selected representative results (set S) are shown in red, while the original dataset (set P) is shown in blue.

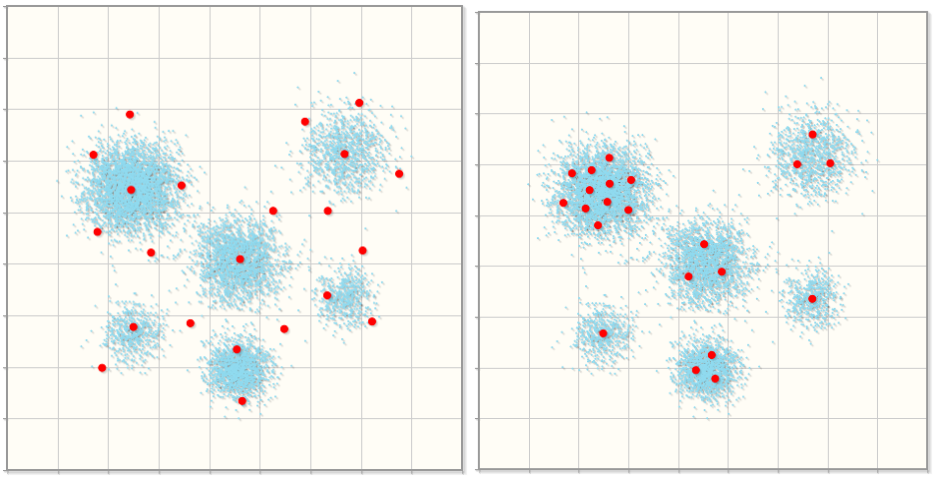

Figure 2: POIKILO results using DisC (left) and *k*-medoid (right), selected representative results (set S) are shown in red, while the original dataset (set P) is shown in blue.

POIKILO also provides an implementation of all the algorithms presented in [DP12A] for computing DisC diverse subsets. These are graph-based algorithms that use a spatial index structure, namely the M-tree, to efficiently execute neighborhood queries. We briefly describe some of them next. Let us call *black* the items of P that are in S, *grey* the items covered by *S* and *white* the items that are neither black nor grey. The *Basic-DisC* heuristic initially considers that *S* is empty and all items are white. The algorithm proceeds in rounds; until there are no more white items, it selects an arbitrary white item *p*, colors *p* black and colors all items in $N_r(p)$ grey. The *Greedy-DisC* heuristic, instead of selecting white items arbitrarily at each round, selects the white item with the largest number of white neighbors, that is, the white item that covers the largest number of uncovered items.

For zooming-in, i.e., for $r' < r$, we can construct r'-DisC diverse sets that are supersets of *S* by adding items to it. The items to be added are either selected randomly or in a greedy manner, where at each turn the item that covers the largest number of uncovered items is selected. For zooming-out, i.e., for $r' > r$, in general, there may be no subset of *S* that is an r'-DisC diverse subset. We provide a suite of algorithms that focus on minimizing $S \setminus S'$ i.e., the set of items that belong to the previous diverse subset S but are removed from the new one S', and $S' \setminus S$ i.e., the set of the new items in S' not in *S*.

Relevance

In many cases, the items of a dataset are associated with a relevance score. For example, a post may have a high relevance score based on the like it has received, or a tweet based on the number of times it has been re-tweeted. In such cases, it is important to select as representative the items that have high scores.

To model relevance, we assume *a relevance function* $w : P \rightarrow R+$ that assigns a relevance score to each data item.

Dispersion-based models combine relevance and diversity using parameters for tuning the degree of diversification. Most common approaches use weights, for example a parameter σ, $0 \leq \sigma \leq 1$, to weight the relevance of each item against its distance from other items during the selection process (a method called MMR [CG98]) or using a parameter λ, $\lambda > 0$, to favor the selection of diverse results among relevant ones. In the latter case, the corresponding relevance-aware diversity functions for MaxMin and MaxSum are:

$$f_{minrel}(S, d) = \min_{p \in S} w(p) + \lambda \min_{p,q \in S} d(p, q)$$
$$f_{maxrel}(S, d) = (k-1) \sum_{p \in S} w(p) + 2\lambda \sum_{p,q \in S} d(p, q)$$

In POIKILO, users are given to option to select how to combine relevance with diversity and specify the value of related tuning parameters.

For the DisC model, we define the Weighted *r*-DisC Diverse Subset Problem as follows:

Given a set *P* of data items, a relevance function w : P → R and a radius *r*, the weighted *r*-DisC problem is to find an *r*-DisC diverse subset *S** of P, such that, for every *r*-DisC diverse subset S of P, it holds that $\Sigma_{p \in S^*} (1 / w(p)) \leq \Sigma_{p \in S} (1 / w(p))$.

Figure 3 shows the represented diverse subset computed by POIKILO without and with taking relevance into account when selecting represented items.

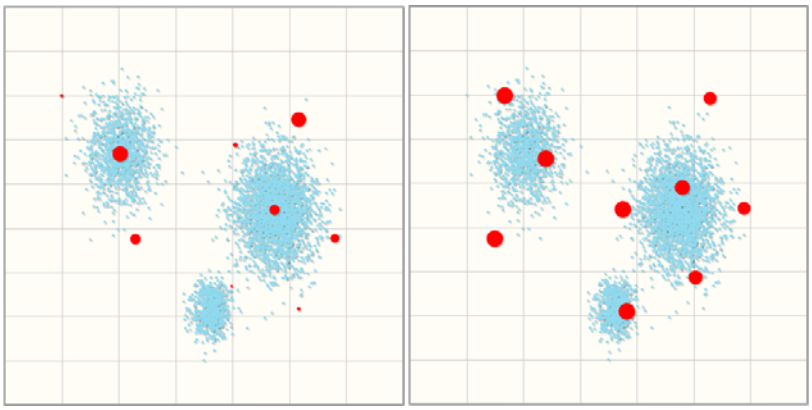

Figure 3: POIKILO results using DisC without relevance (left) and with relevance (right) selected representative results (set S) are shown in red where larger size represent higher relevance, while the original dataset (set P) is shown in blue.

Again, we provide implementations of many different algorithms for handling relevance.

Streaming data

We also consider the dynamic case in which the dataset change over time, as for example, in the case of social network postings. We adopt a sliding-window model where diverse items are computed over sliding windows of length w in the input dataset. The length of the window w can be defined either in time units (e.g., the most diverse items among the items posted in the last hour) or in number of items (e.g., the most diverse items among the 100 most recent postings).

We have implemented the index-based algorithms proposed in [DP12A] using Cover Trees to dynamically update the diverse subset of each window. We also provide the option to enforce the continuity properties proposed in {DPx, DP12b] among consequent windows. For example, the order in which the diverse items are delivered to the users should follow the order of their generation. Also, an item should not appear, disappear and then re-appear in the presented diverse set.

Section 3 How to Use the Tool

The POIKILO tool is a Web Application implemented in Java EE using JavaServer Faces 2.0. The system architecture of the tool can be seen in Figure 4. POIKILO can be accessed via a simple web browser using an intuitive GUI.

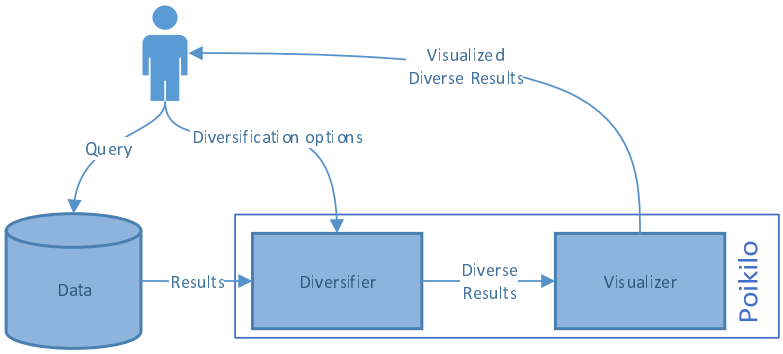

Figure 4: The system architecture of the POIKILO tool

Users can upload datasets, submit queries, see diverse results and tune a variety of diversification parameters. Figure 5 shows an example of the user interface of POIKILO. Users can upload datasets and set various parameters to control the visualization.

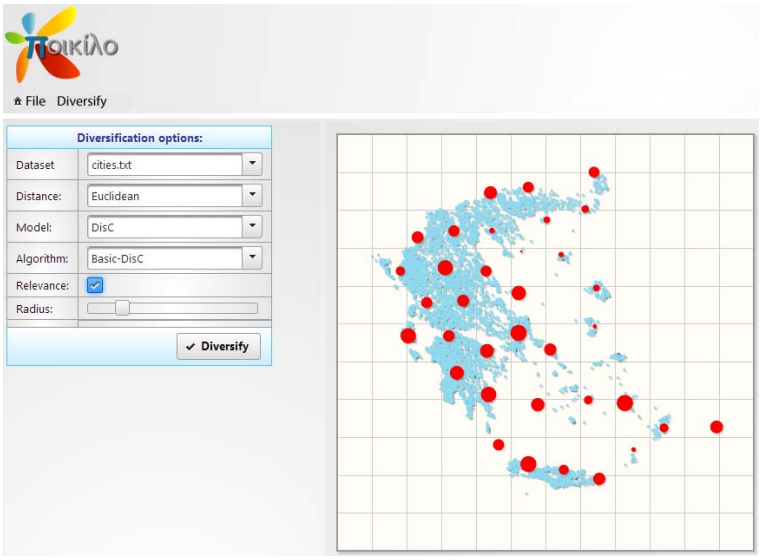

Figure 5: The interface of the POIKILO tool.

Users can upload their own dataset using the GUI. The datasets are given as multi-dimensional vectors. They can also provide relevance scores. As a part of the available demonstration of the tool, we also provide a number of datasets, both real and synthetic. Our synthetic datasets consist of points in the 2D plane. Points are either uniformly

distributed in space or form clusters of different sizes. Relevance scores are also assigned to items in a uniform or clustered way. We also use a number of real datasets, such as datasets collected from the social analysis tools.

Upon entering the system, users are presented with a panel providing a wide variety of different options (see Figures 6 and 7). The users can select:

- a dataset among the ones available (either, provided by the system, or previously uploaded by the user),
- a distance metric (e.g., Euclidean, cosine, Harversine),
- a diversification model (e.g., DisC, MaxMin, MaxSum),
- an algorithms and values of model-specific parameters based on the selected diversification model. For example, for the MaxMin model they can select a *Greedy* (or *Best-Interchange* etc) algorithm and set a value $k = 10$ (Figure 6), while for DisC model they can select a BasicDisC (or Greedy-DisC, etc) algorithm and set $r = 0.001$ (Figure 7).
- either, to account for relevance or not during the selection of representative results
- to treat the input data as streaming by specifying a window length.

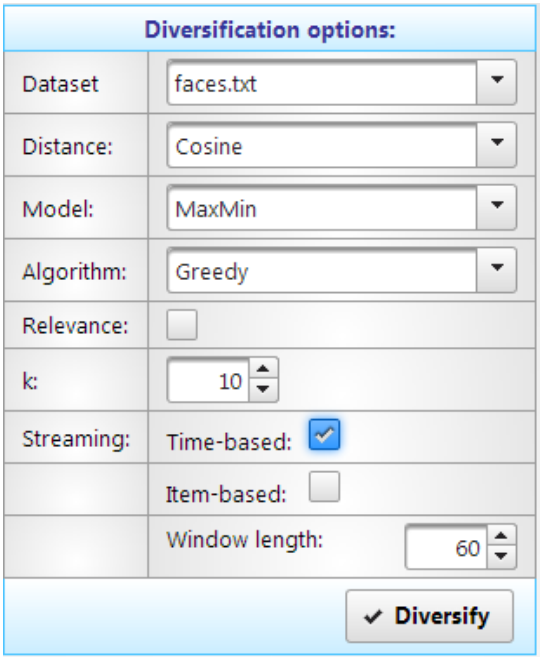

Figure 6: Setting the values of the various visualization parameters (options for the MaxMin model)

The computed representative subset is presented to the users along with additional information, such as its size and the average pairwise distance among the selected items.

For point data, a visualization of the whole dataset is presented, in which diverse items are represented in a different size and color (see for example Figure 5). If relevance is considered, the size of each diverse item corresponds to its relevance score, i.e., the larger this score is, the larger the item appears. Users have the option to hide the non-diverse items if they wish.

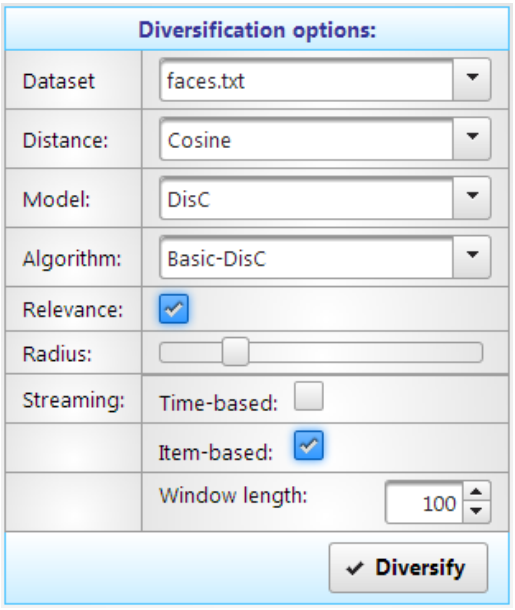

Figure 7: Setting the values of the various visualization parameters (options for the DisC model)

When the DisC model is used, after being presented with the diverse subset for the specific radius, users have the option to tune the degree of diversification by zooming-in or zooming-out of the presented subset (Figure 8). A sliding bar is provided, which users can slide to dynamically increase or decrease the value of *r* without having to specify it explicitly (see Figure 7).

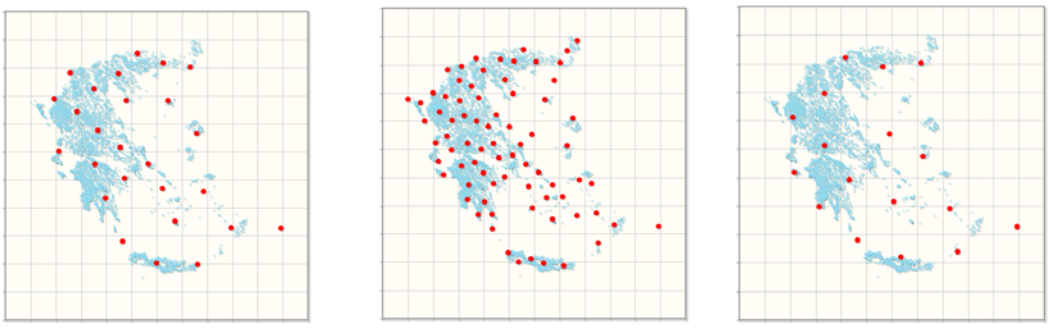

Figure 8: An example of zooming. After being presented with the original set (left), the users can zoom in by selecting a smaller radius (middle) or zoom out by selecting a larger radius (right)

Finally, when users use the streaming option, they have the opportunity to see how representative data items change as new items enter and leave the current window by

navigating between windows via next and previous buttons. Users can also request the enforcement of continuity properties among consequent windows.

References

[AGH+09] R. Agrawal, S. Gollapudi, A. Halverson and S. Ieong. Diversifying search results. In *WSDM*, 2009.

[AK11] A. Angel and N. Koudas. Efficient diversity-aware search. In *SIGMOD*, 2011.

[CG98] J. G. Carbonell and J. Goldstein. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *SIGIR*, 1998.

[CKC+98] C. L. A. Clarke, M. Kolla, G. V. Cormack, O. Vechtomova, A. Ashkan, S. Buttcher and I. MacKinnon. Novelty and diversity in information retrieval evaluation. In *SIGIR*, 2008.

[DPx] M. Drosou and E. Pitoura. Diverse set selection over dynamic data. In *IEEE Trans. Knowl. Data Eng.* (to appear).

[DP10] M. Drosou and E. Pitoura. Search result diversification. *SIGMOD Record*, 39(1), 2010.

[DP12a] M. Drosou and E. Pitoura. Disc diversity: result diversification based on dissimilarity and coverage. In *PVLDB*, 6(1):13--24, 2012.

[DP12b] M. Drosou and E. Pitoura. Dynamic diversification of continuous data. In *EDBT*, 2012.

[EUY94] E. Erkut, Y. Ulkusal and O. Yenicceriogluu. A comparison of p-dispersion heuristics. In *Computers & OR*, 21(10), 1994.

[FMT12] P. Fraternali, D. Martinenghi and M. Tagliasacchi. Top-k bounded diversification. In *SIGMOD*, 2012.

[QYC12] L. Qin, J. X. Yu and L. Chang. Diversifying top-k results. In *PVLDB*, 5(11):1124--1135, 2012.

[VRB+11] M. R. Vieira, H. L. Razente, M. C. N. Barioni, M. Hadjieleftheriou, D. Srivastava, C. Traina and V. J. Tsotras. On query result diversification. In *ICDE*, 2011.

[YLA09] C. Yu, L. V. S. Lakshmanan and S. Amer-Yahia. It takes variety to make a world: diversification in recommender systems. In *EDBT*, 2009.

[ZMK+05] C.-N. Ziegler, S. M. McNee, J. A. Konstan and G. Lausen. Improving recommendation lists through topic diversification. In *WWW*, 2005.