Preference-enriched Faceted Search for Voting Aid Applications

Yannis Tzitzikas and Eleftherios Dimitrakis

***Abstract*—Most Voting Advice Applications (VAAs) are questionnaire-based systems. In this paper we introduce, analyze and evaluate an alternative approach; we show how *Preference-enriched Faceted Search (PFS)* can be used as a VAA. The introduced approach is more expressive, since it allows users to prioritize their preferences, it is more transparent since users can see how each preference affects the ranking of the parties, and it is more informative since during the interaction they can see the options associated to each party. Moreover we introduce an enrichment of PFS with scores that quantify the degree up to which a party satisfies the preferences of a user. Finally we compare the PFS with the questionnaire-based method according to various criteria, and we describe two task-based evaluations with users that we have carried out whose results were very positive.**

***Index Terms*—Voting Advice Applications, Faceted Search, Preferences.**

I. Introduction

Voting Advice Applications (or Voting Aid Applications), for short VAAs, are in use in many countries worldwide, from European countries (for more than one decade [3]), to Asian countries [19] and Latin America [20]. The value of such systems is that (a) they enable political parties to make their views known to voters in a more "structured" and easily comparable method, and (b) they allow citizens to become more knowledgeable about the positions of the political parties which in turn could lead in better vote choices. Most VAAs are questionnaire-based: the user is given a series of multiple choice questions and in each one of them he has to select the choice he agrees more with. At the end of the process the system computes the similarity between the user and each political party. The underlying model of these systems is the model of *issue voting* [5] (see [22] for a review).

In this paper we introduce and elaborate on a quite different approach. We show how the recently proposed *Preference-enriched Faceted Search* [28], for short *PFS*, can be used as a VAA. The main idea is the following: Political parties are modeled as objects. Each object (political party) is described according to a set of facets (each corresponding to one political aspect, topic or issue) by assigning to it one or more terms from each facet. The user *explores* the information space (the set of political parties) using the familiar interaction of *faceted search* which is *enriched* with actions that allow expressing his/her *preferences* over the terms of each facet. After each such action, the set of objects is ranked accordingly. The ranked set of objects is actually the set of political parties ranked according to the preferences of the user. It is also worth making an analogy: In the context of e-commerce users do not buy products or services using questionnaires, but they use faceted search (as for example in `eBay`, `booking.com`) for being able to explore the choices according to various aspects and finally to decide. Analogously, we believe that the flexibility and the expressive power of the PFS process could turn out to be very useful to citizens, and could be the basis for the next generation VAAs.

The proposed approach has several interesting properties. In brief, it is more expressive since it allows users to *prioritize* their preferences in a *qualitative* manner something that is not possible with the questionnaire-based VAAs. It is also more transparent since users can see how each preference action affects the ranking of the parties. It is also more informative in the sense that during the interaction the users can see which of the options are associated to each party. In addition, in questionnaire-based VAAs the users are obliged to answer all the questions (usually around 30) for seeing any result. Instead, with the proposed approach the user can express as many preferences as he wishes to. In a nutshell, the key contributions of this paper are:

- We propose and elaborate on how PFS (preference-enriched faceted search) can be used as a Voting Advice Application (VAA).
- We demonstrate the proposed interaction using a pilot application for PFS as a VAA for the parliament elections of January 2015 in Greece.
- We analyze the benefits of this approach, in comparison to the questionnaire-based approach (time flexibility, expressiveness, responsiveness, transparency, process flexibility).
- We provide methodological hints for applying this approach.
- We enrich PFS with scores for quantifying the degree up to which a party satisfies the preferences of a user.
- We report the results of two task-based evaluations with users; the results are very promising.

Issues like how such tools affect the eventual electoral behavior of voters (as in [18], [6]) are beyond the scope of our work. The rest of this paper is organized as follows.

Yannis Tzitzikas: Computer Science Department, University of Crete, Greece, and Institute of Computer Science (ICS), Foundation for Research and Technology - Hellas (FORTH)

Eleftherios Dimitrakis: Department of Mathematics and Applied Mathematics, University of Crete, dimitrakis@csd.uoc.gr

Manuscript received Sept. 2016; revised Nov. 2016.

Section II describes background and related work. Section III describes the approach by showing how we have used the system *Hippalus* for the needs of the elections of January 25, 2015 in Greece. Section IV analyzes various aspects of the approach. Section V describes the pilot usage of the system and the results of the first evaluation with users. Section VI introduces the quantification of preferences and reports the results of the second evaluation with users. Finally, Section VII concludes the paper and identifies issues for further work and research. The deployment that we used for the elections of January 25, 2015 is accessible through the following address: `http://www.ics.forth.gr/isl/Hippalus` (requires Firefox version 8+).

II. Background and Related Work

A. Voting Advice Applications

In each Voting Advice Application (VAA) the user has to select the most preferred choice from a series of multiple choice questions and then the application computes the similarity between the user and each political party. Voting Advice Applications are used in many countries for presidential, parliamentary, regional or municipal elections. Such systems help citizens to understand how they match with a political party, based on the program of the party. This is useful because quite often it is rather difficult and time consuming to find and read the political programmes of the political parties. Furthermore, it has been argued that these applications must be commended as they focus people's attention on the programmes and on policy issues, compelling parties in this way to discuss substance instead of personalities, images and campaign events. The interested reader can refer to [1] for a recent state-of-the art of VAAs, and to [13]. One critical issue that requires attention is the selection of the questions. For instance, [29] ends with a plea for a careful selection of VAA statements and for a proper process of benchmarking based on survey data. An example of a VAA for the Greek elections of January 2015 is the system *HelpMeVote*[1]. That system was used in Greece for the general elections of May 2012, June 2012 and January 2015 [3]. Apart from *HelpMeVote*, another VAA that has been used in Greece is the *Choose4Greece* [16][2]. The theoretical background underlying Voting Advice Applications, like *HelpMeVote*, is the model of *issue voting* [5]. According to that model voters choose the party that is closest to their own preferences on a set of political issues. The problem can be modeled as follows [15]. Let $Q = \{q_1, \ldots, q_M\}$ be the set of M questions (else called issues or statements), and $P = \{p_1, \ldots, p_C\}$ be the set of C political parties (or candidates). Each political party p_j has answered each question $q_k \in Q$. Each user u_i does the same. The answers of political parties are coded by experts or answered by representatives of the political parties. Every political party and user can be represented by a vector, $\vec{u_i} = \{u_{(i,1)}, \ldots, u_{(i,M)}\}$ and $\vec{p_j} = \{p_{(j,1)}, \ldots, p_{(j,M)}\}$ respectively, where $u_{(i,k)}$ $p_{(j,k)}$ are the answers of the i-th user and j-th party, respectively, to the k-th question. Commonly the set of answers is a 6-point Likert scale: $L = \{$ 1 (Strongly disagree), 2 (Disagree), 3 (Neither agree or disagree), 4 (Agree), 5 (Strongly agree), 6 (No opinion) $\}$. The 6th point is usually ignored because it does not correspond to a particular stance. The objective of the system is to approximate the hidden function $h : \mathbb{R}^M \times \mathbb{R}^M \rightarrow \mathbb{R}$, where $h(\vec{u}, \vec{p})$ is the estimation of the relevance of user u with political party p. Typically $h(\vec{u}, \vec{p}) \in [0, 1]$, or $h(\vec{u}, \vec{p}) \in [-1, 1]$. The top suggestion p_s for user u is the party that has the highest estimated relevance i.e the $p_s \in P$ such that $h(\vec{u}, \vec{p_s}) = \max\{h(\vec{u}, \vec{p}) \mid p \in P\}$, in other words $p_s = argmax_p\{h(\vec{u}, \vec{p}) \mid p \in P\}$. We can also consider a function $r(\vec{u}, \vec{p}) \in [1..T]$ that returns the *rank* of p to the user u, Obviously, from $h(\vec{u}, \vec{p})$ one can derive $r(\vec{u}, \vec{p})$ by sorting in descending order. The computation of $h(\vec{u}, \vec{p})$ is usually based on the distance of the vectors $d(\vec{u}, \vec{p})$, or using a case-based method as in [3]. As distance measure the Euclidean distance (L_2) can be used, or many others (a number of distance measures are discussed in [21]).

There are variations or extensions of VAAs for various objectives, for instance [15] presents an extension of VAAs for community-based recommendations and comparisons.

B. Preference-enriched Faceted Search (PFS) and the system Hippalus

At first we discuss Faceted Search and Dynamic Taxonomies, then preferences, and afterwards the extension of faceted search with preferences, i.e. PFS. Finally we describe *Hippalus*, a system that supports PFS.

Dynamic taxonomies (faceted or not) is an interaction framework based on a multi-dimensional classification of data objects allowing users to browse and explore the information space in a guided, yet unconstrained way through a simple visual interface [25]. Features of this framework include: (a) display of current results in multiple categorization schemes (called facets, or dimensions, or just attributes), (b) display of facets and values leading to non-empty results only, (c) display of the count information for each value (i.e. the number of results the user will get by selecting that value), and (d) ability to refine the focus gradually, i.e. it is a session-based interaction paradigm in contrast to the stateless query-and-response dialogue of most search systems. Faceted search is currently the de facto standard in e-commerce (e.g. eBay, booking.com), and its popularity and adoption is increasing, e.g. in professional search [7], in semantic search [9], and in several other domains, e.g. see [27] for a recent survey of methods for faceted search of RDF/S datasets.

Preferences. Commonly, preferences are not hard constraints, but wishes, simple or complicated ones (covering one or more aspects), which might or might not be satisfied. Such wishes might be independent, or might

[1] http://helpmevote.gr
[2] http://www.choose4greece.com/

affect each other, even in conflicting ways. In general, preferences can be defined either using a *qualitative* approach [17] or a *quantitative* approach [2]. According to the former, preferences are described directly, using a preference relation $\succ_{Pref}$ (i.e. $x \succ_{Pref} y$), while according to the latter, preferences are described indirectly by defining scoring functions (i.e. $Score(x) > Score(y)$). The qualitative approach is more powerful and expressive than the quantitative approach, since not every preference can be modeled using scoring functions [10].

Preference-enriched Faceted Search (PFS). Most faceted search systems order facets and zoom-points in lexicographical order, or order facets and zoom-points based on the number of indexed objects. Other systems, like eBay, only present a manually chosen subset of facets to the users, and the zoom-points are again ranked based on the number of indexed objects. There are several works that propose methods for identifying and ranking facets and zoom-points according to various criteria. However, and to the best of our knowledge, the only model that allows users to define explicitly the desired *preference structure* in a gradual and flexible manner (i.e. aligned with the principles of faceted search), also anticipating attributes with hierarchically organized values and possibly set-valued, is the one proposed in [28]. In brief, in that work the interaction model of faceted search is enriched with actions that allow the user to express *preferences* for ranking the facets, their values, and the objects. The preference actions allow expressing *best values*, *worst values*, *relative preferences* (e.g. I prefer A to B), and *around to* preference. Moreover it offers actions for *composing* preferences using *Priority*, *Pareto* [17], *Pareto Optimal (i.e. skyline [4])*, and *Combination* (i.e. order according to priority; the rest actions are the least prioritized and use Pareto composition). We shall hereafter call this interaction *PFS*: *Preference-enriched Faceted Search*.

Hippalus [24] is a publicly accessible web system that realizes the PFS as defined in [28]. Figure 1 shows an annotated screenshot of *Hippalus*. *Hippalus* uses a left bar for the facets and the corresponding zoom points, since this is the most widely adopted approach or policy for FDT visualization (evidenced by the UI design of global systems like booking.com, eBay). *Hippalus* offers the preference related actions through *right-click* activated pop-up menus (through HTML5 context menus).

The key point is that the user has two main kinds of actions: (a) the classical *left-clicks* of faceted search that *change the focus* (i.e. change the viewed set of objects), and (b) *right-clicks* that express preference which *rank the focus*. At any time the user can issue any kind of action he wishes to. *Hippalus* displays the set of objects of the focus, ranked according to the expressed preferences, in the central part of the screen. Finally, the right part of the screen offers frames that show the history of object restrictions (i.e. the left clicks), the history of preference actions (i.e. the rights clicks), and offers a method for prioritizing preference by defining a bucket order of the facets in a drag and drop manner. The information base

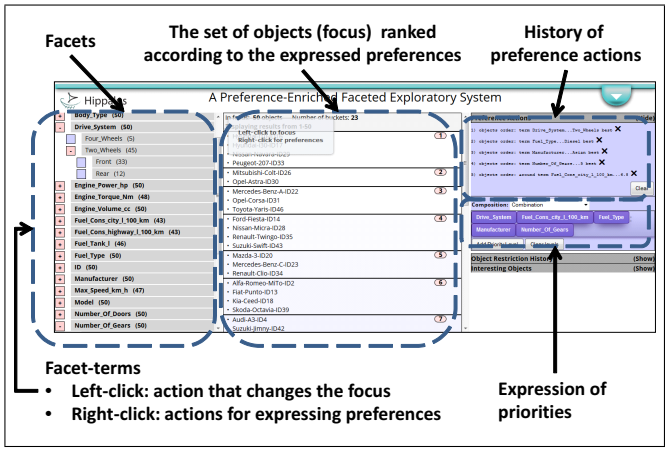

Figure 1. Annotated screenshot from an application of *Hippalus* for car selection

that feeds *Hippalus* is represented in RDF/S[3] according to an RDF Schema adequate for representing objects described according to dimensions with hierarchically organized values.

III. The PFS-based Approach for VAAs

At first (in §III-A) we describe how PFS can be used for building VAAs. Then (in §III-B) we describe the dataset that describes the political parties in the Greek elections of January 2015, and finally (in §III-C) we describe the PFS-based interaction through an example.

A. The Model (for turning PFS a VAA)

The main idea is that political parties are represented as objects, each described according to a number of facets by assigning to it one or more terms from each facet. The user explores the information space (the set of political parties) using the familiar interaction of faceted search which is enriched with actions that allow expressing his/her preferences over each facet. After each such action, the set of objects is ordered accordingly. The ranked set of objects is actually the set of political parties ranked according to the preferences of the user. In particular, let $P = \{p_1, \ldots, p_C\}$ be the set of C political parties (or candidates). We have K facets $F = \{F_1, \ldots, F_K\}$ each associated with a taxonomy $(T_i, \leq_i)$ where T_i is a set of terms, or values, while $\leq_i$ is a possible empty partial order over T_i enabling to organize the values of T_i hierarchically. In comparison to *issue voting* (as described in §II), here we do not have M questions or issues. Instead we have K facets each corresponding to one criterion or issue. In each one of these facets we do not have a single option but we can have more than one options. The stances of the political parties and the users are not described using any Likert scale. Instead each political party p is described by associating it with one or more values from each facet, i.e. with those values that correspond to its political stances. Let $\vec{p_i}$ denote the description of p_i in that

[3]https://www.w3.org/standards/techs/rdf#w3c_all

space, and $\vec{P}$ the set of descriptions of all parties of P. To give a small example, suppose that $P = \{p_1, p_2, p_3\}$ and $F = \{Ideology, Privatization, Dismissals\}$, and assume that $\vec{P}$ is as follows:

Party	Ideology	Privatization	Dismissals
p_1	Left	Against privatization	Against
p_2	Right	Privatization	No more than 5%
p_3	Right	Privatization	Whatever is required

A user u explores the information space and expresses gradually a set of qualitative (i.e. relative) preferences over the terms of each facet. Let $actions_u$ denote the set of *preference actions* expressed by u. These actions define a preference relation (a binary relation) over each T_i, denoted by $\succ_{i,u}$, and then these binary relations are *composed* to define a preference relation over the elements of the information space i.e. over $V = T_1 \times \ldots \times T_K$. Note that since a party can be associated with more than one values from a facet (not in this example), it is more precise to define V as the cartesian product $\mathcal{P}(T_1) \times \ldots \times \mathcal{P}(T_K)$ where $\mathcal{P}(T_i)$ denotes the powerset of T_i. For example, a party could have governed with self reliance and in a coalition government, meaning that in the corresponding facet that party would have been associated with two values. Returning to our example consider a user u whose $actions_u$ contain two relative preferences: $Right \succ_{1,u} Left$, and $Against\ privatization \succ_{2,u} Privatization$. These two can be composed without using any priority (i.e. using Pareto composition [17]). The linear extension of the composed preference relation will be the following *bucket order* of $T_1 \times T_2$ (i.e. a linear order of subsets of $T_1 \times T_2$):
$\langle (Right, AgainstPrivatization),$
$\{(Right, Privatization), (Left, AgainstPrivatization)\},$
$(Left, Privatization)\rangle$.
It contains three blocks: the first contains the most preferred pair, the second block contains two equally preferred pairs, and the last one contains the least preferred pair. The user can also express priorities amongst the preferences of different facets. In our example, the user may express that his/her preferences over $Ideology$ have more priority than those over $Privatization$. In that case the resulting bucket order would be:
$\langle (Right, AgainstPrivatization),$
$(Right, Privatization),$
$(Left, AgainstPrivatization),$
$(Left, Privatization)\rangle$.

The above example indicates how the actions in $\succ_{actions_u}$ define a preference relation over V. Since the description of the parties $\vec{P}$ is a subset of V ($\vec{P} \subseteq V$), the actions in $actions_u$ define a preference relation over P denoted as $(P, \succ_{actions_u})$. From $(P, \succ_{actions_u})$ a *bucket order* of P, i.e. a linear order of subsets of P, is produced through topological sorting. Let denote this bucket order by $BO_{actions_u}$, i.e. $BO_{actions_u} = \langle b_1, \ldots, b_Z \rangle$ where b_1 contains the most preferred parties, while b_Z the least preferred. All b_i ($1 \leq i \leq Z$) form a *partition* of P (i.e. they are pairwise disjoint and their union is P). The number of blocks Z ranges between 1 and C. Obviously, if $Z = C$ then the ranking forms a linear order of P, while if $Z = 1$ then all parties are equally ranked (this is true at the beginning of the interaction). Equivalently, we can say that the aforementioned approach ranks the parties i.e. it computes a function $r(actions_u, \vec{p}) \in [1..Z]$ where $Z \leq T$.

B. Forming the Dataset

Here we describe the created dataset for the political parties in the Greek elections of January 2015. At first we selected the parties to be represented, basically we selected all parties that are widely known in Greece. Then we designed the facets, i.e. the criteria. We selected 8 criteria. Since their names are in Greek, Figure 2 shows them in Greek and provides also their translation in English. For each facet we specified its values (only the terms of the facet *Ideology* are hierarchically organized). Then we described each party by associating it with one or more values from each facet based on the political programmes of the parties, as found in the Web. We also included two facets for enabling users to express their preferences on various qualities of the particular parties, i.e. *trust* and *skills*. This is the reason for including facets F_3 and F_4. Each such facet has a name and $|P|$ terms, one for each party (having as name the party name). This enable the user to state his/her preferences regarding trust and skills of the desired parties.

C. The Interaction through an Indicative Scenario

Here we describe the PFS-based interaction through an example.

Initial State: At the beginning we see in the middle vertical area the list of all political parties, i.e. the set P, sorted lexicographically as shown in Figure 3. The circle-enclosed one, i.e. "(1)", means that all (12) parties are in the first block, i.e. that they are all equally preferred. According to the notations in Section III-A, since the user has not specified any preference action, we have $actions_u = \emptyset$, and consequently $\succ_{actions_u} = \emptyset$ and the bucket order of parties is $BO_{actions_u} = \langle b_1 \rangle$ where $b_1 = P$, i.e. all political parties in one block. By clicking one party p_i (or by mouse over) the user can get a card that shows the values of that party according to the defined criteria (i.e. the vector $\vec{p_i}$), as we see for the party "ΣΥΡΙΖΑ". The left bar shows the list of facets F and by clicking on the symbol [+], of a facet F_i the values of that facet (i.e. T_i) become visible, as it has been done in the dimension "Παράνομη Μετανάστευση" (*Illegal immigration*). We can see that this facet has three values. At the right side of each value we see a count indicating how many of the parties have that value.

1) Step 1: Preferences about Currency: Suppose that the user expands the facet "Νόμισμα" (*Currency*), right-clicks on the term "Ευρώ" (*Euro*) and from the menu that pops up, selects the option *Best*. The new ranking of the parties is shown in Figure 4. We can see that the set of parties is now divided into two blocks. The first contains the most preferred parties (i.e. those parties that support Euro) which are 8 in number, while the second the

F_1	Έχει Κυβερνήσει	**Has Governed**
	– Όχι δεν έχει κυβερνήσει	No, it has not governed
	– Ναι, έχει κυβερνήσει αυτοδύναμα	Yes, it has governed with self-reliance
	– Ναι, σε κυβέρνηση συνεργασίας	Yes, it has governed in collation
F_2	Απολύσεις στο Δημόσιο	**Dismissals in the Public Sector**
	– Άγνωστο	Unknown
	– Κατά	Against
	– Υπέρ	In favor
F_3	Εμπιστεύομαι τις Ικανότητες	**(I Trust the Skills of)**
	– ΑΝΕΛ	ANEL
	– ΑΝΤΑΡΣΙΑ	ANTARSIA
	– ...	...
	– ΧΡΥΣΗ ΑΥΓΗ	GOLDEN DAWN
F_4	Εμπιστεύομαι τις Προθέσεις	**I Trust the Intentions of**
	– ΑΝΕΛ	ANEL
	– ΑΝΤΑΡΣΙΑ	ANTARSIA
	– ...	...
	– ΧΡΥΣΗ ΑΥΓΗ	GOLDEN DAWN
F_5	Ιδεολογία	**Political Ideology**
	– Άγνωστη	Unknown
	– Αριστερή	Left
	– Κέντρο	Centre
	∗ Κεντρο-αριστερή	Centre-Left
	– Ολοκληρωτική	Totalitarian
	∗ Ακροδεξιά	Extreme-right
	∗ Κομμουνιστική	Communist
	– Σοσιαλιστική	Socialistic
	– Τεχνολογική	Technological
	– Φιλελεύθερη	Liberal
F_6	Νόμισμα	**Currency**
	– Άγνωστο	Unknown
	– Δραχμή	Drachmas
	– Ευρώ	Euro
F_7	Παράνομη Μετανάστευση	**Illegal immigration**,
	– Όλοι έξω από την Ελλάδα	All should leave Greece
	– Όλοι χωράνε στην Ελλάδα	Greece has room for all
	– Χρειάζεται Έλεγχος και Διαχείριση	Control and management is required
F_8	Παρουσία στη Βουλή	**Presence in parliament**
	– Έχει Παρουσία στη Βουλή	Has presence in parliament
	– Δεν έχει Παρουσία στη Βουλή	Has not presence in parliament

Figure 2. The facets and their terms (in Greek and English)

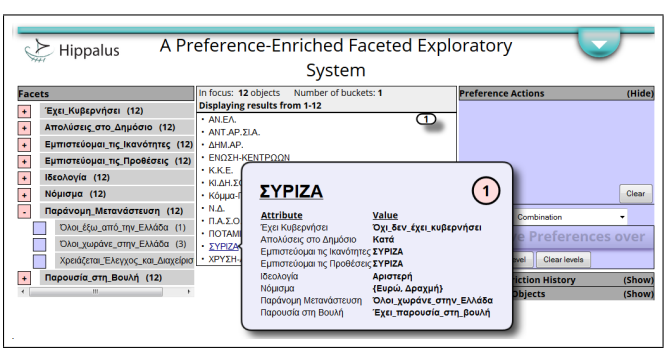

Figure 3. The first screen showing the list of all parties.

less preferred ones (those parties whose program does not contain Euro). The parties belonging to the same block are equally preferred. Formally, here we have $actions_u = \{Best(Euro)\}$. This defines a preference relation over the terms of facet F_6, which in turn defines $\succ_{actions_u}$. The induced linear order is $BO_{actions_u} = \langle b_1, b_2 \rangle$ as shown in Figure 4.

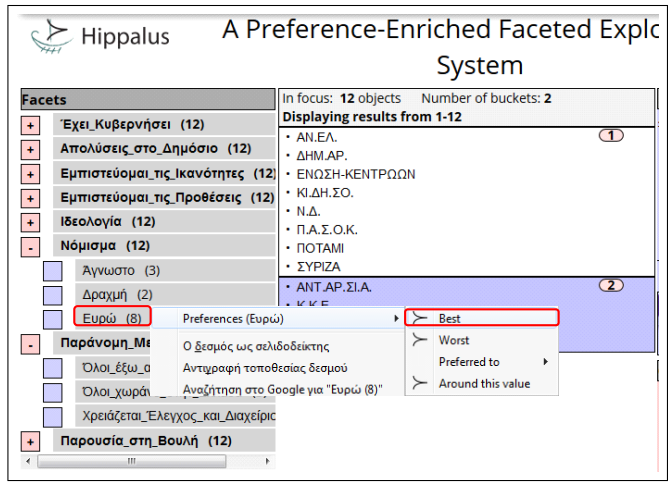

Figure 4. Specifying Euro as the Best value in the facet "Νόμισμα" (Currency)

2) Step 2: Preferences about Illegal Immigration: Now suppose that the user selects the facet F_7 (Παράνομη Μετανάστευση - *Illegal Immigration*). Assume that the user is against the first option "Όλοι Έξω Από Την Ελλάδα" (*All should leave Greece*). With right click he can express that this is the *worst* option. Notice at this point the expressive power (in comparison to questionnaires): the user does not have to select only one option; he may just want to state only the option he does not prefer. Such an action implies that all the rest options are more preferred than "Όλοι Έξω Από Την Ελλάδα"(*All should leave Greece*). We can see the new ranked list of parties in Figure 5. We observe that now we have three blocks. Notice that the last preference action did not affect the first block.

Formally, here we have $actions_u = \{Best(Euro), Worst(\text{Όλοι Έξω Από Την Ελλάδα})\}$. The last preference action defines a preference relation over the terms of facet F_7. Overall, we have one preference action related to F_6 and one related to F_7. Since no priority has been expressed, the preferences expressed over different facets are considered *equally important*, therefore the *Pareto composition* ([17]) is used. The Pareto composition of two preference relations $P1$ and $P2$, denoted by $P1 \otimes P2$, is defined as:

$$x \succ_{P1 \otimes P2} y \text{ iff } (x_1 \succ_{P1} y_1 \wedge (x_2 = y_2 \vee x_2 \succ_{P2} y_2)) \vee (x_2 \succ_{P2} y_2 \wedge (x_1 = y_1 \vee x_1 \succ_{P1} y_1))$$

Consequently in our context the relation $\succ_{actions_u}$ is derived by Pareto composition. The induced linear order $BO_{actions_u} = \langle b_1, b_2, b_3 \rangle$.

3) Step 3: Preferences about Presence in Parliament: Now suppose that the user opens the dimension F_8 "Παρουσία στη Βουλή" (*Presence in Parliament*) for expressing that he prefers parties that have presence in the

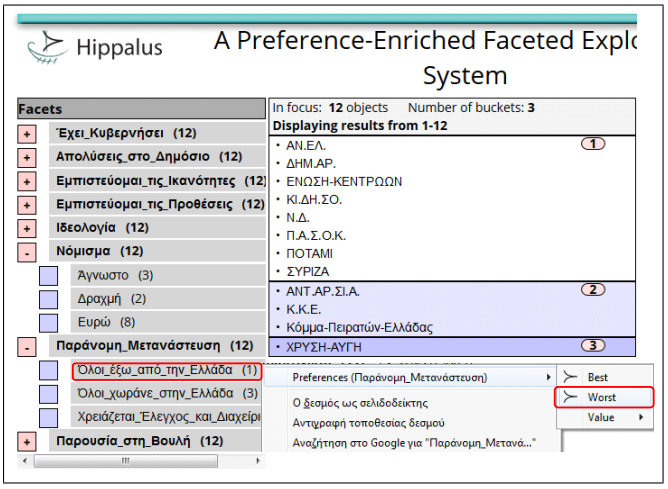

Figure 5. Specifying "Όλοι Έξω" as *Worst* from the facet "Παράνομη Μετανάστευση" (Illegal Immigration)

parliament (instead of small parties that are not already represented in the parliament). Suppose that he selects "Έχει παρουσία στη Βουλή" (*Has Presence in Parliament*) as the *best* value. The new ranked list of the parties is shown in Figure 6. We observe that the first block is now different; "ΕΝΩΣΗ-ΚΕΝΤΡΩΩΝ" is now moved from the first block to the second block because it has never managed to enter the parliament. Again, this action has defined a new preference over one facet and the produced ranked list of parties is derived using Pareto composition.

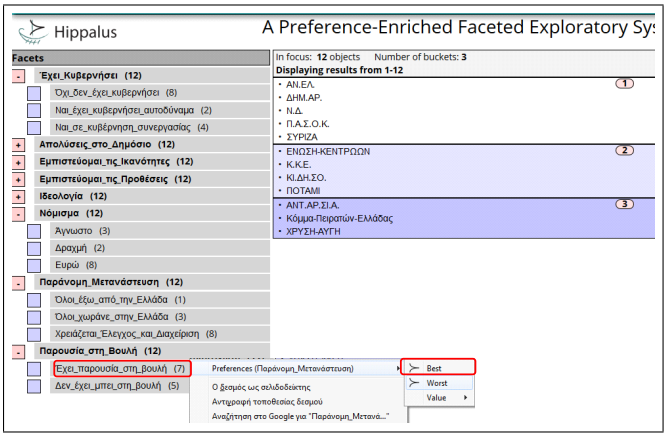

Figure 6. Specifying "Έχει παρουσία στη Βουλή" as *Best* in the facet "Παρουσία στη Βουλή" (Presence in Parliament)

4) Step 4: Preferences about Political Ideology: The user then opens the facet "Ιδεολογία" (*Ideology*) for expressing that he prefers "Φιλελεύθερη" (*Liberal*) to "Αριστερή" (*Left*). This is a *relative preference*, i.e. a preference between two values. The results are shown in Figure 7. Now we can see that the first block contains two parties, while the second block contains only one party.

The only difference is that here we have the expression of a relative preference over a facet, i.e. "Φιλελεύθερη" $\succ_{actions_u}$ to "Αριστερή", i.e. *Liberal* $\succ_{actions_u}$ *Left*.

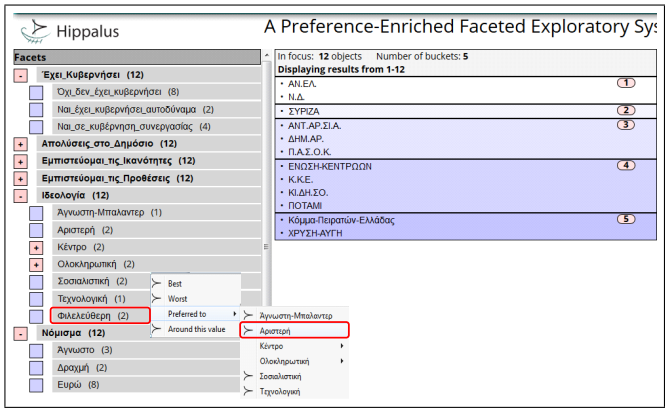

Figure 7. Expressing that "Φιλελεύθερη" (Liberal) is preferred to "Αριστερή" (Left) in the facet "Ιδεολογία" (Ideology)

5) Step 5: Preferences about Participation to Governments : Now the user opens the facet "Έχει Κυβερνήσει" (*Has Governed*) for expressing that he does not prefer parties that have governed with self-reliance. He therefore marks the value "Ναι έχει κυβερνήσει αυτοδύναμα" (*Yes, it has governed with self-reliance*) as *Worst* option. The results are shown in Figure 8. Now can see that the first block contains only one party.

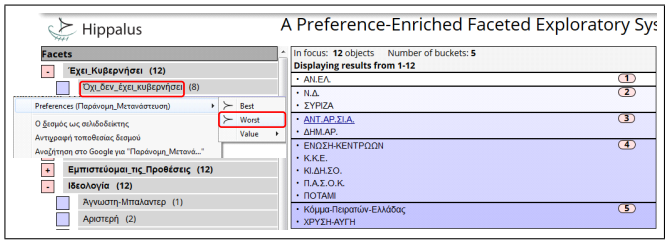

Figure 8. Expressing that "Έχει κυβερνήσει αυτοδύναμα" (Has governed with self-reliance) is *Worst* in the facet "Έχει Κυβερνήσει" (Has Governed)

6) Prioritization: Note that at any point of the interaction, the user can see in the right frame all the preferences that have been expressed so far, as shown in the upper part of Figure 9. The user can delete any if he wishes to. In our scenario so far the user has not specified any *priority* between the preference statements he has made. The default option is that all preferences are combined using *Pareto*, i.e. they all have the same priority. However the user can define if he wishes to, priorities. At the right part of Figure 9 at the middle, we can see the five blue (gray) buttons each corresponding to a facet over which he has expressed preferences. Suppose that for the user is important not to vote a party that has governed with self-reliance in the past. Therefore he can define two levels of priorities using the frame at the bottom right (as shown in Figure 9). The first level will contain all preferences related to the facet "Έχει Κυβερνήσει" (*Has Governed*), while the second level will contain the rest. We can see the two levels, and the final ranking of the political parties. This is an example of prioritized composition, where the *prioritized composition* [17] of two preference relations $P1$

and $P2$, denoted by $P1 \rhd P2$, meaning that $P1$ has more priority than $P2$, is defined as:

$$x \succ_{P1 \rhd P2} y \text{ iff } x_1 \succ_{P1} y_1 \vee (x_1 = y_1 \wedge x_2 \succ_{P2} y_2)$$

In our context, let a_i and a_j be two sets of preference actions, where a_i (resp. a_j) consists of actions on terms from F_i (resp. F_j). An action requesting a prioritization $a_i \rhd a_j$, yields a ranking that is derived by ordering each block defined by the preference a_i, using the preferences in a_j. Therefore the produced bucket order interprets prioritized composition ($\rhd$) as follows:

$$x \succ_{P1 \rhd P2} y \text{ iff } x_1 \succ_{P1} y_1 \vee (x_1 \equiv_{P1} y_1 \wedge x_2 \succ_{P2} y_2)$$

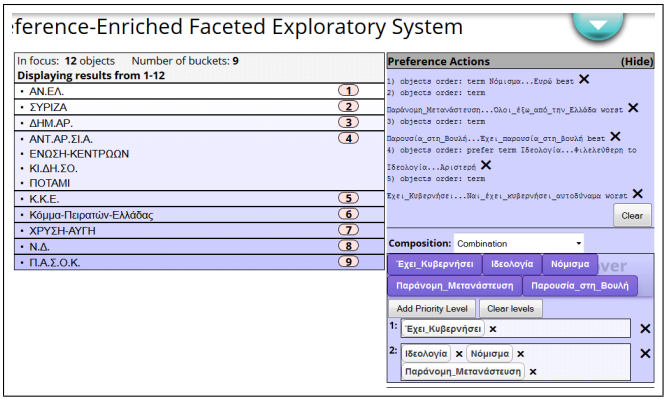

Figure 9. Upper right: history of expressed preference. Bottom right: specification of priority levels

7) Step 7: Investigation of Different Prioritization: The user is also able to continue and add new preferences or delete expressed preferences over the facets. The priorities will remain the same, unless the user changes them again.

The underlying theoretical framework allows expressing preferences at different levels of granularity and it can resolve automatically the conflicts that may arise, using a scope-based resolution rule (detailed in [28]). For example suppose that the facet Ideology contained four terms $\{Right, Left, Communism, ExtremeRight\}$ organized hierarchically as follows: $Communism \leq Left$, $RadicalLeft \leq Left$, $ExtremeRight \leq Right$, and $Liberals \leq Right$. Now consider a user that expresses the following two relative preferences: $Right \succ Left$ and $Communism \succ ExtremeRight$. They are conflicting in the sense that the first favors the right ideology while the second favors the left ideology. However the used terms have different granularity and therefore the preference actions have different scopes, which can be resolved. The induced ranking of the leaf terms (those used for describing parties), will be $\langle Liberals, RadicalLeft, Communism, ExtremeRight \rangle$.

8) Focus Restriction Actions: In the current scenario we have seen only expression of preferences. However the user can also *restrict* at any point the choices (according to the classical faceted search interaction). For instance suppose that the user decides to see only the parties which do not support solely *Euro* or solely *Drachmas*, but leave the issue of currency quite open. To this end he can go to the facet "Νόμισμα" (*Currency*) and left-click the option "Άγνωστο" (*Unknown*). As shown in Figure 10, the user now sees the 3 parties which are ordered according to the preferences expressed so far.

The frame with the history of object restrictions (bottom-right in Figure 10) can be used for canceling this selection and seeing again the set of 12 parties. And of course the user can add or delete preferences and so on.

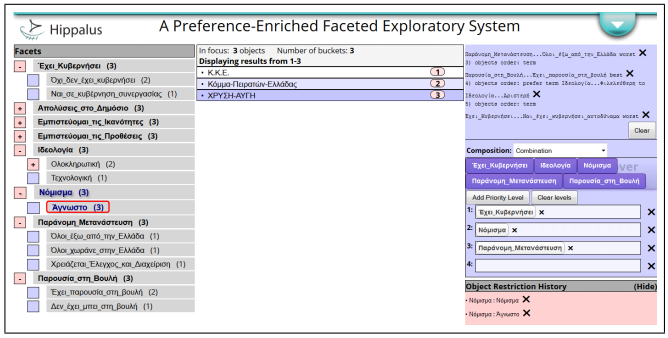

Figure 10. The result of a focus restriction action

IV. Analysis

A. Questionnaire-based versus Preference Enriched Faceted Search

Below we compare the Questionnaire-based approach with the PFS-based approach according to seven aspects, namely *time flexibility*, *expressiveness*, *responsiveness*, *transparency*, *process flexibility*, *simplicity*, and *matching output*.

Time Flexibility

Questionnaire-based: All questions have to be answered. The time required is linear with respect to the number of questions.

PFS-based: The user decides which facets to use and how many preferences to provide. Consequently, we can say that PFS is more scalable with the number of criteria since increasing their number does not necessarily increase the minimum time that a user should spend. Another related merit, is that the values of the dimensions can be hierarchically organized. The user can exploit this structuring since he can express preferences using intermediate (in the hierarchy) values, avoiding in this way to express preferences of all of the narrower terms.

Expressiveness

Questionnaire-based: In most VAAs the user cannot specify which topics/questions have more priority (most tacitly assume that all citizens have the same priorities over their preferences, something which obviously does not hold).

PFS-based: The user can specify the desired priorities of the criteria, enabling in this way a more precise expression of the user's aggregated preferences. Furthermore, the theoretical framework of PFS is based on qualitative preferences, which are more expressive than the quantitative

ones [11], and thus from questionnaire-based VAAs with weighted questions.

Responsiveness

Questionnaire-based: The user gets the results at the end of the process, after having answered all of the questions.

PFS-based: The user can observe how every action affects the ranking of the political parties at any step of the process.

Transparency

Questionnaire-based: The user cannot understand easily or validate how the results have been derived. Therefore it is questionable whether users trust such systems, especially if they get back an unexpected result. Questionnaire-based systems could be maliciously biased (e.g. for a party p_i one could use in the distance functions a $\vec{p_i}$ that is not the same with the one shown at the end of the process, i.e. in the screen that shows in parallel the user's answers and the party's answers and there is no easy (straightforward) method to detect it.

PFS-based: Due to the responsive nature of the PFS-based method, the user can easier understand and validate the results at every step of the process.

Flexibility (order-independence) of the Process

Questionnaire-based: The process is linear: the user answers a question and proceeds to the next. In most cases the user cannot go back and change one response.

PFS-based: The process is not linear. The user can express preferences in any order he wishes to. The user can at any point delete a preference that he has expressed. This freedom allows the system to be used also for hypothesis testing (e.g. the user changes the priorities and investigates whether this affects or not the produced ranking).

Simplicity

Questionnaire-based: Very simple for all users, and consequently user-friendly. Just one click per question. Linear process.

PFS-based: Not as simple as the questionnaire-based method. Although the users are acquainted with the Faceted Search method (since it is the de facto standard in e-commerce), its extension with preference is novel, and therefore most users have not yet used this kind of interaction. However the results of previous usability tests (reported in detail in [23] and summarized in [24]) indicate that the method is simple and effective.

Matching Output

Questionnaire-based: At the end of the process the user gets back a linear list of the parties each accompanied by a degree expressing the degree of match between the opinions of the user and the party. The list is in decreasing order with respect to these degrees. The degrees can be positive or negative, indicating a positive or negative correlation.

PFS-based: A bucket order, i.e. a linear order of blocks, each being a set of parties. No degrees of match are given. However the extension that will be described in §VI extends PFS with such degrees.

B. Methodology for Applying the Approach

Here we describe the main steps of the process for applying this model.

(1) Select the political parties P to include.
(2) Define the set of facets $F = \{F_1, \ldots, F_K\}$ and define their terms $\{T_1, \ldots, T_K\}$. Select terms which are clear. Investigate whether the values of one or more T_i should be hierarchically organized since the intermediate values are beneficial both for the presentation of the values, as well as for the expression of the preferences.
(3) Describe the parties P with respect to the defined facets and their terms (by asking representatives of the political parties), i.e. define the set $\vec{P}$.
(4) Check if it is useful to include facets that allow users to express their preferences on qualities of the parties, e.g. trust, skills, etc. Each such facet has a name and $|P|$ terms, one for each party.
(5) If the number of facets is high, investigate ordering the facets by putting first those that correspond to issues on which citizens seem to give higher priority.
(6) Express the F and $\vec{P}$ in CSV format and load it to the system *Hippalus*.
(7) Communicate the system to the users. From our experience with users, the instructions that should be given are: (a) the interaction has two main kinds of actions: actions for *restricting* the set of parties, and actions for *ranking* the set of parties (the second are activated through right clicks), (b) you can start from any facet that you want (the order does not make any difference), and (c) if you would like to express *priorities*, then use the frame at the right side to specify them.

V. Usage and Evaluation

A. Pilot Usage

The system was deployed in January 18, 2015, i.e. one week before the elections of the January 25, 2015. It was announced in the Information Systems Laboratory of FORTH-ICS and in social media. A web form for reporting problems and giving feedback was provided. Over 100 distinct users visited the system the first day. For reasons of privacy the server-side logging was not activated nor analyzed.

B. User Evaluation

One month after the elections (end of February of 2015) we conducted a user evaluation with the following objectives: (a) to get general and specific feedback from users, (b) to investigate whether the users managed to express their actual preferences, and get as most preferred party the party that they actually voted in the January 25 elections (of course this depends also on the particular facets, and for this reason the questionnaire included a question for testing the suitability of the eight particular facets), and (c) to see to what extend the users were

satisfied by this application and to collect comments and feedback for improving the approach and the system in the future.

The process took place in the Computer Science Department of the University of Crete. The author demonstrated the system in the audience of two courses: one for undergraduate and one for graduate students. In particular the author demonstrated a scenario like the one presented in Section III-C in less than 10 minutes. The students were asked to use the system, when at home (the same day or the same week), and then to fill a questionnaire. It was stated to them clearly, that they should not rush up. The participation to this evaluation was optional. Twenty persons (20) eventually participated. The number was sufficient for our purposes, since according to [8] 20 evaluators are enough for getting more than 95% of the usability problems of a user interface.

C. Questionnaire

We used an online questionnaire. Below we copy it. At first it contains the task description and then the questions. After each question, we show the results of the survey in the form of percentages written in bold.

Task. *With Firefox (≥v8) connect to Hippalus*[4] *and try to express your political preferences. Specifically try to express the preferences that determined your vote in the Greek parliamentary elections of January 25, 2015. After that please fill the following questionnaire.*

Questions:

1) Do you think that:
   ☐ Hippalus-VAA could help citizens to get informed about the stances of the political parties **(69% ☒)**
   ☐ Hippalus-VAA helped you to understand how you select what to vote **(69% ☒)**
2) Were the particular 8 facets useful for expressing your preferences over political parties?
   ◯ Yes, they were complete **(23% ⊙)**
   ◯ Yes, but I would like some additional facets that are not included **(77% ⊙)**
   ◯ No, they are not useful **(0% ⊙)**
3) By using the system did you manage to express your preferences and get a ranked list of parties that satisfied you?
   ◯ Yes I managed to express my preferences, and I got the party that I voted in the first block **(55.5% ⊙)**
   ◯ I managed to express my preferences but the party that I voted was not in the first block **(33.3% ⊙)**
   ◯ I could not express my preferences **(11.1% ⊙)**
4) Did you use the right frame for specifying priorities?
   ◯ Yes, I used the right frame for defining priorities **(95% ⊙)**
   ◯ No, I did not use the right frame for defining priorities **(5% ⊙)**
5) How much time you spent using the system?
   ◯ Less than 5 minutes **(16% ⊙)**
   ◯ 5-15 minutes **(73% ⊙)**
   ◯ More than 15 minutes **(11% ⊙)**
6) Have you used any other VAA systems like HelpMeVote?
   ◯ Yes, I have used systems like HelpMeVote **(61.1% ⊙)**
   ◯ No, I have not used any system like HelpMeVote **(38.8% ⊙)**
7) How would you rate the system Hippalus as a Voting Advice Application?
   ◯ Very Useful **(11.1% ⊙)** ◯ Useful **(77.7% ⊙)**
   ◯ Neutral **(11.1% ⊙)** ◯ Not Useful **(0% ⊙)**
   ◯ Useless **(0% ⊙)**
8) Your feedback is important. Please use the textbox below for reporting problems that you encountered, other comments, or suggestions for future improvements (e.g. on missing facets, usability problems, etc). ⟨*textbox*⟩
9) Question for participant's sex and age. Male **(72.2%)**, Female **(27.7%)**. Ages: **19-31**

[4]http://www.ics.forth.gr/isl/Hippalus

D. Analysis of the Questionnaire

The most important results is that 55.5% of the participants stated that they managed to express their preferences and got the party that they actually voted in the first block, and 33.3% stated that they managed to express their preferences, meaning that in total 88.8% stated that they managed to express their preferences. This is worth noting and we should have in mind that in questionnaire-based VAAs it has been reported that users sometimes get quite different results from different VAAs [29]. For those users that did not manage to obtain through their preference actions in the first block the party they voted, one possible explanation is that the set of facets were not complete (since 77% of them stated they would like more facets). Another explanation is that some users probably did not understand the system or used it wrongly. The second more important result is that 95% of the users defined priorities, justifying the importance of priorities in VAAs. Finally, in the overall rating, 11.1% of the participants rated the system *Very Useful*, and 77.7% *Useful*, hence in total 88.8% of the participants were positive. It is also interesting to note that 69% stated that the system helped them to understand how they selected what to vote.

VI. Quantifying the Satisfaction of Preferences

A. Motivation and Problem Statement

Here we show how we can enrich the produced bucket order of parties with scores that express the degree up to which each party satisfies the preferences of the users. This allows the user to distinguish cases where the parties in the first block satisfy entirely his preferences (i.e. 100%), or at a high degree, than cases where this does not hold, i.e. cases where even if the parties are in the first block satisfy very few preferences. Specifically, we would like to define a *scoring function* that expresses the degree up to which p fulfils the preferences in $actions_u$. For a party $p \in P$, let denote this score by $score(actions_u, \vec{p})$ and suppose its range is the interval $[1, 100]$ (as we move towards the last block the scores should be decreasing). We do not want to replace the bucket order with a score-based ranking; instead we want to *enrich* it with scores. However, a scoring function should respect the preferences, and we can state this formally as:

Definition 1 (Consistency): A scoring function *score* is *consistent* with the qualitative-based bucket order, if for any two parties p_x, p_y and any set of user actions $actions_u$, it holds: if $pos(p_x) < pos(p_y)$ then $score(p_x, actions_u) > score(p_y, actions_u)$ where $pos(p_i)$ is the position of p_i in the bucket order defined by $actions_u$. $\diamond$

B. A Scoring Function for the Problem at Hand

Several scoring functions could be developed. Below we shall describe one such function. We will define it gradually: first we define a score per facet, then for Pareto composition and finally for prioritized composition. In the following we assume the notations introduced in §III-A. Let $p_{j,i}$ denote the stance of p_j in facet F_i, Z_i be the number of blocks of $(T_i, \succ_{actions_u})$, and r_j be the position of $p_{j,i}$ in the linear extension of $(T_i, \succ_{actions_u})$.

Case: One Facet. For each facet F_i, party p_j and $actions_u$ we can define a score taking the value of Z_i if party p_j has the most preferred stance(s) of facet F_i according to the preferences in $actions_u$. Specifically we define: $score_0(F_i, p_j, actions_u) = Z_i - r_j + 1$. Note that those values that reside in the first block receive the score Z_i, while the values of the last block receive the score 1.

Case: More than one Facet, No Priorities. Consider $N \leq K$ facets (recall K from §III-A) and suppose the user has not expressed any priority, therefore the preferences are composed using Pareto. We can define the overall score of a party by summing up the normalized scores (in range $[1, {}^{100}/_{K}]$) of p_j for all facets, i.e. we can define:

$$score_{Pareto}(p_j, actions_u) = \sum_{i=1}^{N} \frac{(score_0(F_i, p_j, actions_u) - 1) * ({}^{100}/_{K} - 1)}{Z_i - 1} + 1 \quad (1)$$

The value ${}^{100}/_{K}$ is the maximum score that we would like a party to take with respect to each facet F_i. In our running example we have $K = 8$, consequently ${}^{100}/_{K} = 12.5$.

Case: More than one facet and Prioritized Composition. For defining a function that is consistent with the qualitative preferences (recall Def. 1) we first have to normalize the scores with respect to each F_i in range $[1, M]$ where M is the maximum of all Z_i (i.e. $M = \max\{Z_i\}$). To this end instead of $score_0$ we can use $score_1$ defined as:

$$score_1(F_i, p_j, actions_u) = \frac{(score_0(F_i, p_j, actions_u) - 1) * (M - 1)}{Z_i - 1} + 1 \quad (2)$$

Note that the minimum value is 1, while the maximum value is M. Now consider a prioritization $e = F_1 \rhd \ldots \rhd F_L$, meaning that F_1 is the most important facet while F_L is the least important facet. Let pos_i be the position of facet F_i in the expression e, and K_j be the number of facets that exist in the j-th priority level. We shall define a weight for each facet in a prioritization as follows:

$$weight_i = (M + 1)^{\sum_{j=pos_i+1}^{L} K_j} \quad (3)$$

Notice that the first in priority facet will get the weight $(M + 1)^{L-1}$, while the last in priority will get the weight $(M+1)^0 = 1$. Now we can multiply the score of each party in that facet with the weight coefficient:

$$score_2(F_i, p_j, actions_u) = score_1(F_i, p_j, actions_u) * weight_i \quad (4)$$

Now for getting the total score of a party, we sum up the scores for all facets and then we normalize the result in range $[1, N * {}^{100}/_{K}]$ i.e.:

$$score_{Priority}(p_j, actions_u) = \frac{(\sum_{i=1}^{N} score_2(F_i, p_j, actions_u) - 1) * (N * {}^{100}/_{K} - 1)}{M * \sum_{i=1}^{N} weight_i - 1} + 1 \quad (5)$$

This formula ensures that if $v \succ_{F_1 \rhd F_2} v'$ then certainly the score of v is greater than the score of v' as stated by the following proposition:

Proposition 1: The scoring function defined in equation 5 is consistent. $\diamond$ *(the proof is given in the extended version of this paper)*

C. Implementation and Screenshots.

The new release of the system, offers a new mode where the scores are computed and displayed. A screenshot of the user interface in this mode that corresponds to Figure 9 of the scenario described in §III-C, is shown in Figure 11. In this figure we can see the matching degree between the stances of each party and the user's actions based on the scoring function introduced in §VI-B. Since in this scenario the user has expressed priorities, the Prioritized Composition is used and therefore the total scores are computed with the Equation 5.

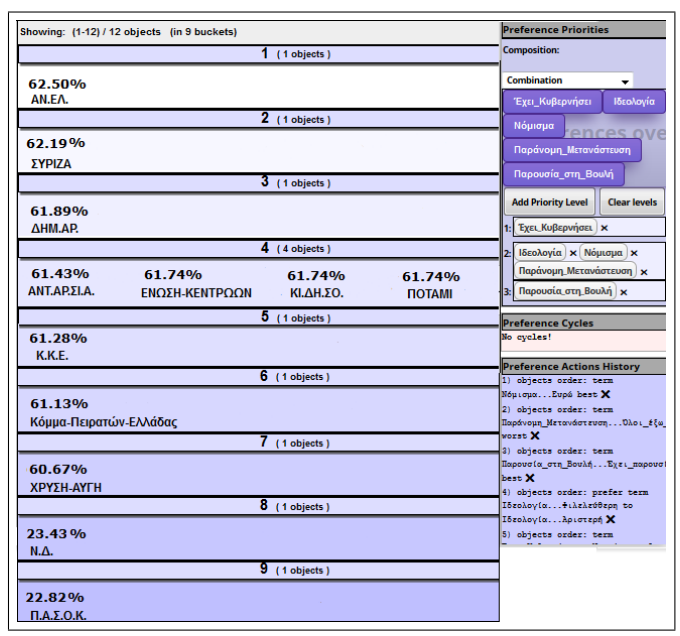

Figure 11. The interaction of Figure 9 in quantitative mode.

D. Evaluation

To evaluate this feature we conducted a second user study. The users should use both interfaces and decide which one of them they prefer. The evaluation started on May and ended on June 2016. The process this time was different: instead of face-to-face demonstration we prepared a video tutorial[5] (of 2.4' minutes) of the system. We invited by email various volunteers (different from those of the first user study) to participate to the evaluation. Thirty-eight persons (38) eventually participated with ages from 19 to 34. At first, we asked them to watch the video tutorial, after to do the same task as in §V-C and then to fill the following questionnaire which also shows the answers that we obtained:

1) Do you think that:
   ☐ Hippalus-VAA could help citizens get informed about the stances of the political parties **(84.2%** ☒ **)**
   ☐ Hippalus-VAA helped you understand how to select what to vote **(60.5%** ☒ **)**
2) Were the particular 8 facets useful for expressing your preferences over political parties?
   ◯ Yes, they were complete **(28.9%** ⊙**)**
   ◯ Yes, but I would like some additional facets that are not included **(65.8%** ⊙**)**
   ◯ No, they are not useful **(5.3%** ⊙**)**
3) By using the system did you manage to express your preferences and get a ranked list of parties that satisfied you?
   ◯ Yes I managed to express my preferences, and I got the party that I voted in the first block **(81.6%** ⊙**)**
   ◯ I managed to express my preferences but the party that I voted was not in the first block **(10.5%** ⊙**)**
   ◯ I could not express my preferences **(7.9%** ⊙**)**
4) Did you use the right frame for specifying priorities?
   ◯ Yes, I used the right frame for defining priorities **(63.2%** ⊙**)**
   ◯ No, I did not use the right frame for defining priorities **(36.8%** ⊙**)**
5) Have you used any other VAA systems like HelpMeVote?
   ◯ Yes, I have used systems like HelpMeVote **(44.7%** ⊙**)**
   ◯ No, I have not used any system like HelpMeVote **(55.3%** ⊙**)**
6) How would you rate the system Hippalus as a Voting Advice Application?
   ◯ Very Useful **(28.9%** ⊙**)** ◯ Useful **(57.9%** ⊙**)**
   ◯ Neutral **(10.5%** ⊙**)** ◯ Not Useful **(0%** ⊙**)**
   ◯ Useless **(2.6%** ⊙**)**
7) Were the scores useful to you?
   ◯ Very Useful **(36.8%** ⊙**)** ◯ Useful **(39.5%** ⊙**)**
   ◯ Neutral **(18.4%** ⊙**)** ◯ Not Useful **(2.6%** ⊙**)**
   ◯ Useless **(2.6%** ⊙**)**
8) Were the scores representative of your preferences?
   ◯ Very Much **(26.3%** ⊙**)** ◯ Much **(57.9%** ⊙**)**
   ◯ Somewhat **(7.9%** ⊙**)** ◯ A little **(5.3%** ⊙**)**
   ◯ Not at all **(2.6%** ⊙**)**
9) Which mode of operation do you prefer?
   ◯ The mode with the scores **(94.7%** ⊙**)**
   ◯ The mode without the scores **(5.3%** ⊙**)**
10) ⟨ *Textbox for feedback (as in the previous evaluation)*⟩
11) Question for participant's sex and age. Male **(63.2%)**, Female **(36.8%)**. Ages: **19-34**.

The results according to the operation mode for the quantification of the user's preferences were very positive, since 94.7% of the participants stated that they prefer the mode with the scores. As regards the scoring function, 36.8% of the participants rated the scores Very Useful, while 39.5% of the participants rated the scores Useful, hence 76.3% of the users were positive. As regards the quality of the scores, 26.3% of the participants rated Very Much, 57.9% rated Much, hence 84.2% of the users were positive.

As regards the entire system, 81.6% of the participants stated that they managed to express their preferences and got the party that they actually voted in the first block, while 10.5% stated that they managed to express their preferences but the party that they voted was not in the first block, meaning that in total 92.1% of the participants managed to express their preferences. Note, that the majority of the participants stated that they have beginner programming experience or no experience at all. It is interesting to mention here, that the percentage of users that did not manage to obtain through their preferences the party they voted in the first block is reduced compared to the previous evaluation (described in §V-B). It is also worth noting that 63.2% of the participants defined priorities between their preferences which makes clear the importance of priorities in VAAs.

Finally, in the overall rating of the system as a VAA, 28.9% of the participants rated the system Very Useful, while 57.9% Useful, hence in total 86.8% of the participants were positive. In addition 84.2% of the participants stated that *Hippalus* could help them get informed about the stances of the political parties and 60.5% stated that Hippalus helped them understand how they selected what to vote.

VII. Concluding Remarks

We have presented a novel approach for building Voting Advice Applications that is based on the Preference-enriched Faceted Search (PFS) approach. We have detailed the approach and the interaction by describing how we have used it for setting up a pilot system for the parliament elections of January 25, 2015 in Greece. Subsequently we analyzed the merits of this approach, in comparison to the widely used questionnaire-based approaches, which showed that the PFS-based method is beneficial with respect to time flexibility, expressiveness, responsiveness, transparency, and process flexibility. The results of the first user study were very positive: most of the participants (88.8%) were positive, and 95% of them defined priorities, justifying the importance of priorities in VAAs. Subsequently we extended PFS with scoring for quantifying the degree of match to between parties and users, and we conducted a second user study with different participants, one year after the elections. The results were very positive (94.7% of the participants stated that they prefer the mode with the scores). Since this is the first work on that subject, there are plenty of issues that are worth further work and research. One topic is the analysis of the log files in order to

[5] available at https://youtu.be/JK4ZXqfvfDE

understand the behavior of the users and accordingly improve the interaction if required (while respecting privacy [14]). Another topic that is worth investigating is using this approach not only as a VAA but also as a voting system where the user will be able to submit his/her vote (bucket order) at the end. Moreover, the individual preferences could also be submitted, enabling in this way to identify not only the winning party, but also the issues/stances that are more popular. Finally, and as regards the system *Hippalus*, extensions with spatial maps [12], or profiles (in the spirit of [26]) could be investigated.

References

[1] *Voting Advice Applications in Europe: The state of the art*. Scriptaweb, 2010.

[2] R. Agrawal and E. L. Wimmers. A Framework for Expressing and Combining Preferences". In *SIGMOD '00: Procs of the 2000 ACM SIGMOD international conference on Management of data*, pages 297–306, New York, NY, USA, 2000. ACM.

[3] I. Andreadis. Voting advice applications: a successful nexus between informatics and political science. In *Proceedings of the 6th Balkan Conference in Informatics*, pages 251–258. ACM, 2013.

[4] S. Borzsony, D. Kossmann, and K. Stocker. The skyline operator. In *Procs of the 17th International Conference on Data Engineering (ICDE'2001)*, pages 421–430. IEEE, 2001.

[5] E. G. Carmines and J. A. Stimson. The two faces of issue voting. *The American Political Science Review*, pages 78–91, 1980.

[6] Z. Enyedi. The influence of voting advice applications on preferences, loyalties and turnout: An experimental study. *Political Studies*, pages 1467–9248, 2015.

[7] P. Fafalios and Y. Tzitzikas. Exploratory Professional Search through Semantic Post-Analysis of Search Results. In *Professional Search in the Modern World*, volume 8830 of *Lecture Notes in Computer Science*, pages 166–192. Springer, 2014.

[8] L. Faulkner. Beyond the five-user assumption: Benefits of increased sample sizes in usability testing. *Behavior Research Methods, Instruments, & Computers*, 35(3):379–383, 2003.

[9] S. Ferré. Expressive and scalable query-based faceted search over sparql endpoints. In *The Semantic Web–ISWC 2014*, pages 438–453. Springer, 2014.

[10] P. Fishburn. *"Utility Theory for Decision Making"*. Wiley, New York, 1970.

[11] P. C. Fishburn. Utility theory for decision making. Technical report, DTIC Document, 1970.

[12] M. Germann, F. Mendez, J. Wheatley, and U. Serdült. Spatial maps in voting advice applications: The case for dynamic scale validation. *Acta Politica*, 50(2):214–238, 2015.

[13] F. Hirzalla and L. van Zoonen. *Online voting advice applications: foci, findings and future of an emerging research field*, pages 87–103. Edward Elgar Publishing, Inc., Cheltenham, UK, 2015.

[14] A. Kaskina and A. Meier. Integrating privacy and trust in voting advice applications. In *2016 Third International Conference on eDemocracy & eGovernment (ICEDEG)*, pages 20–25. IEEE, 2016.

[15] I. Katakis, N. Tsapatsoulis, F. Mendez, V. Triga, and C. Djouvas. Social voting advice applications - definitions, challenges, datasets and evaluation. *IEEE Transactions on Cybernetics*, 44(7):1039–1052, 2014.

[16] I. Katakis, N. Tsapatsoulis, V. Triga, C. Tziouvas, and F. Mendez. Clustering online poll data: towards a voting assistance system. In *Seventh International Workshop on Semantic and Social Media Adaptation and Personalization (SMAP), 2012*, pages 54–59. IEEE, 2012.

[17] W. Kießling. "Foundations of Preferences in Database Systems". In *VLDB'02: Procs of the 28th International Conference on Very Large Data Bases*, pages 311–322. VLDB Endowment, 2002.

[18] A. Ladner and J. Pianzola. Do voting advice applications have an effect on electoral participation and voter turnout? evidence from the 2007 swiss federal elections. In *International Conference on Electronic Participation*, pages 211–224. Springer, 2010.

[19] D.-c. Liao, B. Chen, M. J. Jensen, and C. W. Pritchard. *Political Behavior and Technology: Voting Advice Applications in East Asia*. Springer, 2016.

[20] A. Marzuca, U. Serdült, and Y. Welp. Questão Pública: first voting advice application in Latin America. In *International Conference on Electronic Participation*, pages 216–227. Springer, 2011.

[21] F. Mendez. Matching voters with political parties and candidates: An empirical test of four algorithms. *International Journal of Electronic Governance*, 5(3):264–278, 2012.

[22] F. Mendez. What's behind a matching algorithm: A critical assessment of how voting advice applications produce voting recommendations. In S. Marschall and D. Garzia, editors, *Matching Voters with Parties and Candidates*, pages 49–66. ECPR Press, Colchester, 2014.

[23] P. Papadakos. *Interactive Exploration of Multi-Dimensional Information Spaces with Preference Support*. PhD thesis, Computer Science Department, University of Crete, November 2013.

[24] P. Papadakos and Y. Tzitzikas. Hippalus: Preference-enriched faceted exploration. In *EDBT/ICDT Workshops*, pages 167–172, 2014.

[25] G. M. Sacco and Y. Tzitzikas, editors. *"Dynamic Taxonomies and Faceted Search: Theory, Practise and Experience"*. Springer, 2009.

[26] L. Terán and A. Kaskina. Enhancing voting advice applications with dynamic profiles. In *Proceedings of the 9th International Conference on Theory and Practice of Electronic Governance*, pages 254–257. ACM, 2016.

[27] Y. Tzitzikas, N. Manolis, and P. Papadakos. Faceted exploration of RDF/S datasets: a survey. *Journal of Intelligent Information Systems*, pages 1–36, 2016. (online first).

[28] Y. Tzitzikas and P. Papadakos. Interactive exploration of multidimensional and hierarchical information spaces with real-time preference elicitation. *Fundamenta Informaticae*, 122(4):357–399, 2013.

[29] S. Walgrave, M. Nuytemans, and K. Pepernams. Voting aid applications between charlatanism and political science: the effect of statement selection. In *conference Voting Advice Applications (VAAs): between charlatanism and political science*, 2008.

Appendix

Yannis Tzitzikas is Associate Professor of Information Systems in the Computer Science Dep. at University of Crete and Affiliated Researcher of the Information Systems Laboratory at FORTH-ICS (Greece). He has published more than 100 papers in refereed international conferences and journals, including prestigious journals and venues (e.g. ACM Transactions on the Web, VLDB Journal, IEEE Transactions on Knowledge and Data Engineering, JIIS, JDAPD, ISWC), and he has received two best paper awards.

Eleftherios Dimitrakis has graduated first in the Department of Mathematics and Applied Mathematics of the University of Crete with Grade 8.29/10 and he also won a best performance award for the academic year 2011-2012 (with Grade 8.44/10). He is currently MSc student in the Computer Science Department of the University of Crete.