PlantRT : a Distributed and Diversified Recommendation Tool for Citizen Sciences

Maximilien Servajean, Esther Pacitti, Miguel Liroz-Gistau, Alexis Joly, Julien Champ

PlantRT : a Distributed and Diversified Recommendation Tool for Citizen Sciences*

Plateforme de Recommandation Distribuée et Diversifiée pour les Sciences Citoyennes

Maximilien Servajean* , Esther Pacitti*, Miguel Liroz-Gistau†, Alexis Joly† and Julien Champ‡
*Inria & Lirmm, University of Montpellier, Email : {servajean, pacitti}@lirmm.fr
†Inria & Lirmm, Email : {miguel.liroz_gistau, alexis.joly}@inria.fr
‡Inria & Lirmm, Email : julien.champ@lirmm.fr

Résumé— **De nombreux domaines scientifiques produisent et consomment des volumes de données considérables (*e.g.* biologie, astronomie, physique). Dans ce travail, nous nous intéressons au traitement de données de types images, produites à grande échelle par des botanistes dans le cadre de *Pl@ntNet*[1]. L'objectif de notre prototype, PlantRT, est de permettre l'étude de l'évolution et des corrélations de familles de plantes diverses. Chaque image, ou observation, est ainsi produite de façon personnelle – par chaque citoyen ou botaniste participant au projet –, et peut être stockée sur différents types de sites (*e.g.* ordinateurs personnels, smartphones, serveurs, clouds). Par ailleurs, l'émergence des systèmes de recommandation distribués favorise le partage, la découverte et la mise en relation de ces données produites par chaque citoyen impliqué. Cet article présente PlantRT un prototype de système de recommandation multi-sites, prenant en compte la diversité des profils des citoyens, ou utilisateurs, et des données, favorisant, par exemple, la découverte de nouvelles espèces de plantes d'une même famille ou de la même zone géographique ainsi que le passage à l'échelle. Nous montrons ici, en particulier, plusieurs cas d'utilisations ainsi que le déploiement de PlantRT en détails.**

I. Introduction

Many fields of science are currently massive producers of diverse data items, or items (*e.g.* images, experimental datasets, plant observations). For instance, in botany, the emergence of citizen sciences has fostered the creation of large and structured communities of nature observers (*e.g.* *e-bird*[2], *xeno-canto*[3], *Tela Botanica*[4], *Pl@ntNet* [3]) who started to produce outstanding collections of image records (*i.e.* items). These images are captured by volunteer nature observers, and stored in different and heterogeneous sites (*e.g.* PCs, Smartphones, Clouds, Serveurs). Building an accurate knowledge of the identity, the geographic distribution and the evolution of living species is essential for a sustainable development of humanity as well as for biodiversity conservation. However, scaling up such collaborative approaches to real-world ecological surveillance systems involving millions of contributors is still challenging.

The emergence of distributed search and recommendation systems favors personalized retrieval of items (*e.g.* images). In such systems, each user shares a set of items in its local node. Then, given a user u and a query q – generally keyword based –, the goal is to recommend items in a personalized way. Generally, the recommended items are chosen based only on the query initiator's profile and on the query itself. Diversified search and recommendation is a personalized method [6] that, given a query q and the query initiator u, recommends items taking into account the query, u's profile, and the diversification of both the relevant items and the profiles of the users sharing them. In the context of *Pl@ntNet*, user profiles are defined based on the observations made by each user. Diversification enables the discovery of items referring to plants from different species within the same family, or concerning the same geographical area. This is very useful for botanists who wish to understand the plants biodiversity.

In [2] we proposed a distributed peer-to-peer solution for search and recommendation with no diversification. Here we propose a generic multi-site approach in the sense that sites may be heterogeneous (*e.g.* PCs, Smatphones, Clouds). As presented, each user has her own profile based on the items she shares. However, for multi-users sites (*i.e.* Clouds, Servers), we propose to exploit the concept of virtual nodes. That is, in multi-users sites, in addition to the users' profiles, we adopt the approach of clustering them through *k-means* [1]. Therefore each multi-users site will also be composed of k virtual nodes' profiles (*i.e.* means). The set of users that are similar with a specific means, defines a virtual node. Virtual nodes are useful to define the logical overlay topology for distributed search and recommendation and may be implemented as a simple thread. In addition, virtual nodes enables to index items in a personalized and efficient manner [1].

In our demo, we show in details how PlantRT was deployed over 10 multi-sites (*i.e.* Clouds' virtual machines and a PC) using virtual nodes. Our demo uses a dataset

*Work conducted within the Institut de Biologie Computationnelle and partially funded by the labex NUMEV and the CNRS project Mastodons.

[1] http://www.plantnet-project.org

[2] http://ebird.org

[3] http://www.xeno-canto.org

[4] http://www.tela-botanica.org

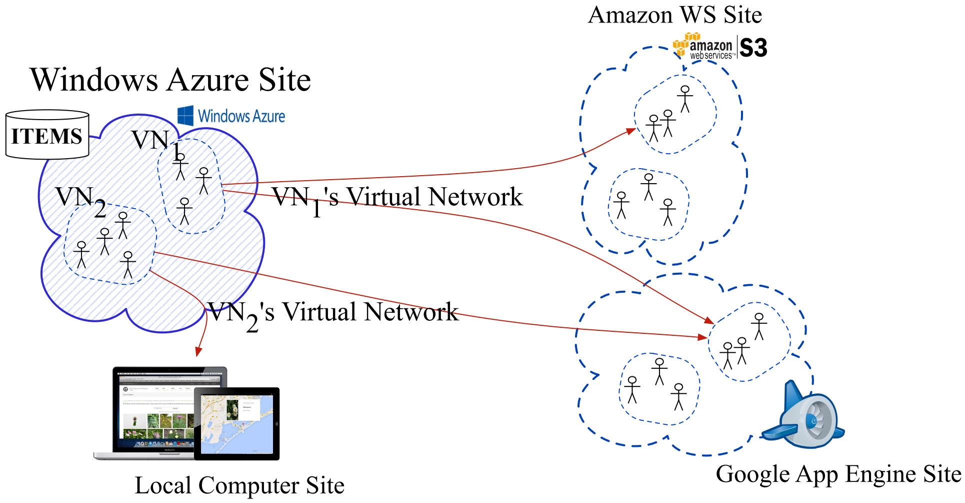


Figure 1: Example of a PlantRT multi-site network with 4 sites.

provided by *Tela Botanica* containing more than $10,000$ observations produced by $1,500$ real citizens. We placed items on different sites and we demonstrate the behaviour of PlantRT with different types of queries, submitted by users having various profiles, to show that our distributed recommendation approach provides good results in terms of diversification, recall and performance. For this purpose, we create several scenarios varying the number of users (and virtual nodes), sites and the size of the corpus. In addition, we also demonstrate the efficiency of the distributed $TF \times IDF$ algorithm, necessary to index the items of the distributed corpus using the images meta-data. This indexation is useful to create the inverted lists, thus enabling *top-k* processing for search and recommendation. To the best of our knowledge, this is the first work that provide a real demonstration of a complete distributed and diversified recommendation tool using real scientific data.

This paper is structured in the following way. First, Section II presents an overview of PlantRT's architecture. Followed by Section III, which presents the details of our distributed $TF \times IDF$ solution. Section IV describes the demonstration itself, including the use case and details about the prototype's deployment. Finally, Section V concludes the paper.

II. PlantRT Architecture's Overview

In this section, PlantRT's architecture is discussed. An important element of the prototype is the concept of virtual nodes. That is, each site hosts 1 to n virtual nodes, each referring to a set of 1 to m users having similar profiles, and produced using *k-means*. Virtual nodes have two benefits. First, they enable to index items in a personalized way [1], thus improving *top-k* performances. Second, they permit to define the logical overlay topology, noted *VN-Network*, using gossip protocols. Notice that the *VN-Network* is used during query processing to guide recommendation among sites.

As presented, PlantRT distributed search and recommendation platform is composed of multi-sites (*e.g.* server, PC, cloud). Each site hosts 1 to n virtual nodes, each referring to a set of 1 to m users having similar profiles. Each user shares a set of items (*i.e.* observations images) that are associated to meta-data and used to produce her profile [6]. The meta-data refer to the plant's family, species and genus, the observation's location and a description. Meta-data and users' profiles are presented through a $TF \times IDF$ vector, the latter being used by *k-means* to produce the virtual nodes. Notice that $TF \times IDF$ is computed in a completely distributed manner, thus avoiding centralization, as explained in Section III.

In our approach, virtual nodes are used both to personalize indexation and to facilitate the distributed overlay construction – users with different profiles will forward their queries through the overlay to different sites.

Thus, each virtual node holds an index (inverted lists) that refers to the items of the related site. Inverted lists associate for each keyword the set of items that contain it. Finally, the site items are ranked in a personalized way with respect to the users' profile of the corresponding virtual node [1]. This enables to improve *top-k* processing performance over the inverted lists. Notice that virtual nodes of a same site, have access to all the items of that site. Figure 1 presents an example composed of four sites, three being virtual machines in the cloud and one being a local computer. In this example, we focus on the case of the *Windows Azure*'s site where two virtual nodes have been produced; also, the site is associated to a database containing all items shared in it.

Additionally, virtual nodes are connected together through a distributed overlay called *VN-network*. It enables to connect diverse virtual nodes taking into account profile diversity. That is, a virtual node has its own profile, corresponding to its center (*i.e.* average of all profiles in the virtual node), computed during the clustering process with *k-means*. By diversifying the *VN-network* which is used to guide recommendation, both results diversification and recall levels are increased while latency is minimized [7]. In Figure 1, each virtual node of the *Windows Azure*'s site is connected to two other virtual nodes owned by two different sites. The first one, VN_1 is connected to both *Amazon* and *Google*'s sites while the second one, VN_2, is connected to both the local computer and *Google*'s site.

The *VN-network* overlay is built in two steps:

1) based on random gossiping, each site is aware of remote virtual nodes available on the network,
2) at each gossip exchange, each local virtual node applies a distributed profile diversification clustering algorithm [7], to build its *VN-network*.

Finally, to retrieve items, users submit queries. A query is generally keyword based $q = k_1, ..., k_p$.Whenever a user u submits a query q, it is forwarded to all nodes in its local virtual node's *VN-network*. Each virtual node receiving the query repeats the same procedure until the query *TTL* (*i.e.* time to live) is reached. Finally, each involved virtual node, including u's one, returns its relevant items, selected within its site, with respect to q and u's profile, by using a specific similarity measure (*e.g.* Jaccard).

III. PlantRT Distributed $TF \times IDF$ Indexing

As presented in Section II, in PlantRT, items refer to images that are associated to textual meta-data. Thus, in order to be indexed in inverted lists, items are represented by a $TF \times IDF$ [4] vector.

In this section, we present in details our distributed solution for $TF \times IDF$ computation because it is an original solution for distributed indexation that does not depend on any centralized authority. Recall that $TF \times IDF$ is a score that defines the importance of a term given an item and the global corpus (*i.e.* set of all items stored in the whole system), which, in our case, is completely distributed. The intuition behind our solution is that we only need average statistics of the global corpus to compute the full score.

More precisely, the $TF \times IDF$ score of a given term t is obtained by multiplying the term frequency (TF) by its inverse document (*i.e.* item) frequency (IDF) within the global corpus I. While the first part can be computed locally, the latter needs a distributed protocol since the global corpus is not available. Thus, to compute *IDF* at each site s, we use a gossip-based protocol. Generally, *IDF* is computed as follows:

$$IDF(t, I) = log\frac{|I|}{F_{t/I}} \tag{1}$$

Where $F_{t/D}$ is the number of items in the whole corpus I that contain the term t.

The goal of our distributed approach is to compute the average number of items per site, noted $|I|/n$, and for each term t, the average number of items that contain it per site, noted $\frac{F_{t/D}}{n}$, where n is the total number of sites. Then, we simply exploit the property of convergence of gossip-based protocols applied to *P2P* average computing [5]. Based on these averages, each site will be able to compute the IDF vector locally.

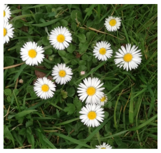
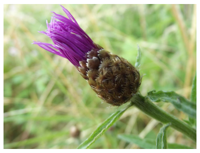
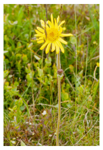
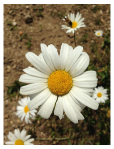
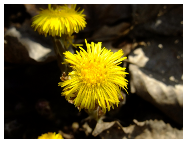
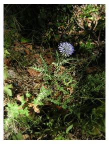
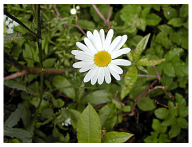
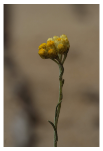
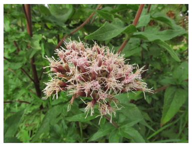
$q = asteraceae$		
Un-diversified	Profile Diversity	
Élodie Dujardin	Marie Dupont	Pierre Durand
		
		
		

Table I: Search and recommendation example with profile diversity.

IV. PlantRT Demonstration

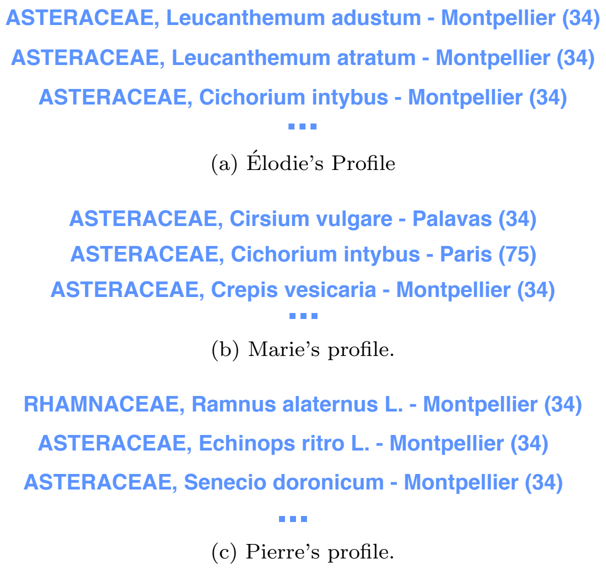
In this section, we show characteristics of PlantRT. First in Section IV-A, we present the use case of our demonstration. Then in section IV-B, we show details of the deployment and of the behavior of the prototype.

A. PlantRT *Use Case*

Recall that we rely on diversified search and recommendation that takes into account the query, the query initiator's profile, and the diversification of both the relevant items and the profiles of the users sharing them. Here we show the behavior of PlantRT over our distributed platform with different types of queries, submitted by users having various profiles, to show that our distributed recommendation approach provides good results in terms of recall and diversification.

Table I presents the results obtained by executing the query "asteraceae" – which corresponds to a very large family of plants, including the well known daisy – over our whole dataset of $10,000$ observations and $1,500$ users. We chose three users to analyze the returned items.

In Table I, each column refers to the results obtained by a single user. In the first column, corresponding to Élodie Dujardin, results are un-diversified. In the two other columns, corresponding to Marie Dupont and Pierre Durand, results have been obtained using profile diversification. Figure 2a, 2b and 2c presents the profiles of respectively Élodie Dujardin, Marie Dupont and Pierre Durand. Recall that the users' profiles are derived from



ASTERACEAE, Leucanthemum adustum - Montpellier (34)

ASTERACEAE, Leucanthemum atratum - Montpellier (34)

ASTERACEAE, Cichorium intybus - Montpellier (34)

...

(a) Élodie's Profile

ASTERACEAE, Cirsium vulgare - Palavas (34)

ASTERACEAE, Cichorium intybus - Paris (75)

ASTERACEAE, Crepis vesicaria - Montpellier (34)

...

(b) Marie's profile.

RHAMNACEAE, Ramnus alaternus L. - Montpellier (34)

ASTERACEAE, Echinops ritro L. - Montpellier (34)

ASTERACEAE, Senecio doronicum - Montpellier (34)

...

(c) Pierre's profile.

Figure 2: Users' profiles.

their observations; therefore each line refers to the observed plant's family and species associated to the location of the observation.

Not diversifying the results leads to the first column which only contains very redundant plants observations. In the other hand, diversifying the results enables to retrieve a broader spectrum of plants species. For instance, the first result of Marie Dupont is a plant which species is *Cirsium vulgare (Savi) Ten*, while the second one corresponds to the species *Crepis vesicaria*. Similarly, Pierre's first results is *Senecio doronicum*, while the second one is *Echinops ritro*. These species can be observed in their respective profile in Figure 2.

We now show how our diversified recommendation method can be applied in the case where a user wishes to retrieve plants observations made in the geographical area where she is. In this case, the query is the geo position and relevant items are chosen taking into account the geographical area around q, its initiator's profile and the diversification of both the relevant items and the profiles of the users sharing them.

For instance, in Figure 3, Pierre Durand went hiking around Montpellier. He is represented by the blue dot in the center of the map, thus referring to the query $q = \langle latitude :43.74\rangle, \langle longitude :3.9\rangle$. The results of the query are represented by the red pins. They refer to diversified observations made in the area. For instance, in the figure, we zoomed on an observation of a plant from the *Rahmnaceae*'s family.

B. PlantRT *Deployment and Robustness*

In our demo, we show the diverse steps necessary to deploy PlantRT over a multi-site architecture using virtual nodes. In addition, we show that PlantRT is robust and performs well.

Two main steps are needed to deploy PlantRT. The first step consists in preparing the environment for the prototype; the second one consists in preparing PlantRT to be executed in the previous environment.

First, an environment must be created to host PlantRT. This environment refers to the machine (*e.g.* local computer, server or a virtual machine in the cloud) and the set of applications needed to host the prototype.

More precisely, two applications must be installed, one being the application server and the other one being a database. . The database is used to store all metadata such as users' profiles or items shared. In our case, we use *MongoDB*[5]. Once the first step is complete, PlantRT must be configured to be run in this specific environment. The configuration consists in just a few informations: the *IP* of an existing site to connect to, the maximum number of virtual nodes, the frequency of the gossip protocol and the frequency of the distributed $TF \times IDF$ protocol. Once configured, PlantRT can be uploaded in the application server (*i.e. tomcat*[6]) and will starts automatically.

During the initialization, several operations are performed by PlantRT automatically. First, the virtual nodes are built and associated to a set of inverted lists. In our case, each set of inverted lists is an instance of *Lucene*[7]. Then, the user – or administrator in the case of multi-users site – specified an *IP* address to which PlantRT will connect. The *IP* refers to an existing remote PlantRT's site s_r. Once the connection is established between the two sites, virtual nodes from s_r are retrieved to the local site and the gossip protocol is started and executed periodically at a system defined frequency. The gossip view (*i.e.* random view) are implemented as simple arrays. The last protocol to be initialized and started is $TF \times IDF$ distributed protocol since it relies on the gossip view to connect to other sites. The initialization consists in building a $TF \times IDF$ vector with the site's local corpus. Once initialized, periodical exchanged are established to compute the $TF \times IDF$ vector. Notice

[5] http://www.mongodb.org

[6] http://tomcat.apache.org

[7] http://lucene.apache.org

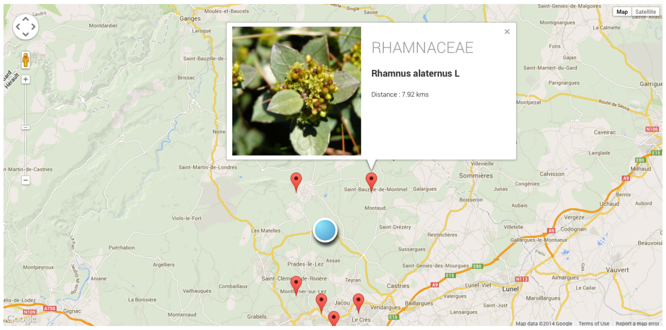


Figure 3: PlantRT's example of geo recommendation next to Montpellier, France.

that this protocol runs indefinitely to take into account new items added to any sites. After this final step, the PlantRT becomes accessible to the users.

In our demo, we show that virtual nodes and a diversified *VN-Network* enable to achieve very good results in term of diversification, recall (*i.e.* the proportion of results answering a query retrieved) and performance. More precisely, with different numbers of sites and virtual nodes, we show that PlantRT reaches centralized-like recall results. Additionally, we have simulated up to $6,000$ nodes still reaching recall results of $99,9$ and very good performance [7].

The distributed *TF* × *IDF* protocol shows very good convergence properties. In our demo, we show that only a few exchanges at each site are needed to compute a *TF* × *IDF* vector with respect to the global corpus.

V. Conclusion

This paper presents PlantRT a multi-sites diversified search and recommendation tool for citizen sciences, and more precisely, for botanists.

We proposed an original multi-site platform that is built over virtual nodes. This enables to reduce network costs and to improve performances of top-k processing. Our demo is done using real scientific data over a concrete multi-site platform. We show several use cases for diversified search and recommendation. In addition, we show, in details, how we deployed ou platform.

The still evolving source code is available at the following address: http://www2.lirmm.fr/˜servajean/prototypes/plant-sharing/plant-rt.html.

References

[1] S. Amer-Yahia and M. Benedikt. Efficient network aware search in collaborative tagging sites. *VLDB Endowment '08*, 1(1):710–721, 2008.

[2] Fady Draidi, Esther Pacitti, Didier Parigot, and Guillaume Verger. P2Prec: a Social-Based P2P Recommendation System. In *CIKM*, pages 2593–2596, 2011.

[3] A. Joly, H. Goëau, P. Bonnet, B. Vera, J. Barbe, S. Souheil, Y. Itheri, J. Carré, E. Mouysset, J.F. Molino, N. Boujemaa, and D. Barthélémy. Interactive plant identification based on social image data. In *Ecological Informatics*, 2013.

[4] KS Jones. A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation*, 1972.

[5] W. Kowalczyk, M. Jelasity, and AE. Eiben. Towards data mining in large and fully distributed peer-to-peer overlay networks. In *BNAIC*, pages 203–210, 2003.

[6] Maximilien Servajean, Esther Pacitti, Sihem Amer-Yahia, and Pascal Neveu. Profile Diversity in Search and Recommendation. In *WWW Companion*, pages 973–980, 2013.

[7] Maximilien Servajean, Esther Pacitti, Miguel Liroz-Gistau, Sihem Amer-Yahia, and Amr El Abbadi. Exploiting Diversity in Gossip-Based Recommendation. In *Globe and BDA (submitted)*, 2014.