CLUSTERING WEB-BASED COMMUNITIES USING SELF-ORGANIZING MAPS

Juan J. Merelo-Guervós, Beatriz Prieto, Alberto Prieto, G Romero, Pedro Castillo Valdivieso
Depto. Arquitectura y Tecnología de Computadores, Universidad de Granada
C/ Daniel Saucedo Aranda, s/n
18071 Granada (Spain)

Fernando Tricas
Depto. Informática e Ingeniería de Sistemas
C/ María de Luna, 1
50018 Zaragoza (Spain)

ABSTRACT

Web-based communities, such as those created around weblogs, form increasingly complex networks, and new tools are needed to map and understand them. Creating a community map allows the visualization of community standing and relationship, and it can be used to discover which members of the community have similar interests. Since hyperlinks are an indication of a member interests, the set of all hyperlinks by a member can be used to represent it; this set can then be used to perform clustering procedures on the group of all community members. In this paper, such procedure is carried out using Kohonen's self-organizing map (SOM), a neural-net like method used to create maps. SOM can be used to divide the set of webs under study in communities/clusters, and, at the same time, visualize them, so that a map for *community navigation* can be created out of the initial map. This procedure has been applied to the Blogalia weblog community (hosted at http://www.blogalia.com/), a thriving community of around 200 members, created in January 2002. In this paper we show how SOM discovers interesting community features, as well as its possible shortcomings when mapping communities.

KEYWORDS

Weblogs, neural networks, self-organizing maps, clustering, web-based communities

1. INTRODUCTION

Web-based diaries, or weblogs, sometimes called also simply *blogs*, have become increasingly popular in the last few years. World-wide, there could be several millions; in any given language, other than English, figures can be up to one hundred thousands. Even as weblogs are some times perceived as little more than post-adolescent rants, they form a community of readers/writers, establishing long-running relationships. A weblog by itself need not be important, but as part of a community, its importance cannot be disregarded. All weblogs in the world can be seen as components of a set of communities, each one with its own idols, axioms, enemies, and hierarchies; communities are not clear-cut, and a particular weblog might belong to several communities at the same time.

However, from an external (for somebody who is completely alien to weblogs) or even an internal (for a weblogger who is interested in knowing a bit deeper the community he/she belongs to) point of view, it is difficult to discover these communities. If we equate communities to clusters, from the statistical point of view, there are many clustering algorithms available, which can give you an idea of how a group of websites is divided into communities; but, then, it is difficult how close each community is to the rest. On the other hand, there are visualization algorithms that can map a high-dimensional community to a low-dimensional (2 or 3 dimensions) map, but, then, another clustering algorithm must be performed on this new representation. The data we are going to apply the algorithm is the set of blogs included in the Blogalia community (http://www.blogalia.com/), a Spanish weblog hosting site (mostly written in Spanish), which started in 2002

with a few blogs and has now expanded to around 200 blogs with more than 100 authors. Each blog is represented by a vector whose components are the number of times it links to others in Blogalia; if a blog such as http://fernand0.blogalia.com/ links to http://atalaya.blogalia.com/ [1] 7 times, the corresponding element will hold the value 7.

Links have been chosen over, for instance, words, since they are easily parseable from the original; this election allows for a low-dimensional representation of each blog (which will represented by a vector with as many components as blogs in the community). It is also univocal: a link clearly identifies origin (the weblog it has been found it) and destination (from the URL). Other kind of data could have been used for representing each blog, such as a vector representing the text used in it, but this presents several problems. First, the representation of each blog would need a high-dimensional vector; moreover some amount of preprocessing would have to be done on each word to obtain its root, put it in the correct normal form, and so on. Links represent a real relationship among the blogs they join: they imply, at least, that one has read the other, which shows, in a sense, a sort of *community* relation; communities are created by reading, writing about other blogs, commenting on them, and so on. It is true that there might be other members of the community not uncovered by this method (for instance, loyal readers or people who use comments to participate); it is also true that one member of the community could be linked to another via some other blog that does not belong to Blogalia; however, we do not attempt to say the last word about community structure in the blogosphere (as is usually called the set of all weblogs), just to show a method that, if you consider hyperlinks a good enough indicator of community relationship, makes communities and subcommunities stand out.

In this paper, we have chosen Kohonen's Self-Organizing Map [Kohonen, 1990], which is an unsupervised neural-network like algorithm that performs clustering of input data, and, at the same time, maps it to a two-dimensional surface, doing both things at the same time. In this paper we will prove that the self-organizing map discovers underlying community structure efficiently.

The rest of the paper is organized as follows: first, we make a brief introduction to Kohonen's self-organizing map in section 2; then, we expose the state of the art in community mapping and discovery. The next section is devoted to present the results of applying Kohonen's self-organizing map to community discovery in Blogalia, and, finally, our conclusions and an outline of future work is presented.

2. KOHONEN'S SELF-ORGANIZING MAP

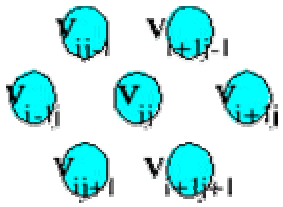

Figure 1. Self-organizing map with hexa neighborhood. Each circle labeled with V represents a vector with the same dimensions as the input vector in the training set

Kohonen [Kohonen, 1990] originally proposed his map based in a previous model by von der Marlsburg as a model for self-organizing visual domains in the brain. Kohonen's SOM is composed of a set of n-dimensional vectors, arranged in a 2-dimensional array. Each vector is surrounded by other 6 (hexagonal) or 8 (rectangular arrangement) vectors, in what is called its *neighborhood*.

Kohonen's SOM, as many other heuristic methods, must be *trained* on the data it is going to model; training proceeds as follows:

1Second and first author's weblogs, respectively

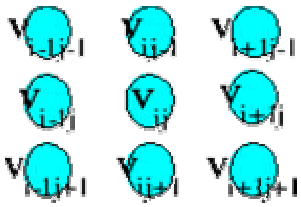

Figure 2. Self-organizing map with square neighborhood.

1. A new vector from the training set (the set of data we want to be modeled) is chosen randomly.
2. The closest vector in the SOM is computed, and called the *winner*.
3. All vectors in the neighborhood of the winning vector are updated so that they become closer to the input vector by a factor α.
4. Neighborhood and α are updated.
5. After a predetermined number of iterations, stop.

The self-organization in the self-organizing maps comes from the fact that different neighborhoods are updated every time a new vector is presented; besides, it is self-organized in the sense that learning proceeds in an unsupervised way. Other than that, SOM is similar to any other clustering algorithm such a k-means, but, in this case, clusters are also arranged geographically; that is why it is said it performs topographical mapping. Main applications of the self-organizing map are:

- *Visualization*: projection from a high-dimensional space to a two-dimensional maps highlights hidden relationships between data set members.
- *Clustering*: unlike other algorithms such as k means, each cluster will be represented by several vectors.
- *Interpolation* or *function modeling*: it is not specially suited for this purpose, but if each vector *v* has an assigned value *f(v)*, these values can be projected on the map, and unknown values deduced from it.
- *Classification* if the original data set is sorted in several classes, each map vector can be calibrated with a class, and then used for classification. Even if it is not as efficient for classification as other neural net algorithms, the fact that it can handle missing values make it quite useful in those cases.
- *Vector quantization*: since the map is a model of a data set, its members can be used to represent that data set, each vector can be quantized by assigning it to its closest representative in the map.

There are many software packages that implement SOM, such as the *SOM Toolbox* for Matlab, or the `som` package for R, but the most popular is probably SOM_PAK[2], created originally by Kohonen's team themselves. This package includes command-line programs for training and labeling SOMs, and several tools for visualizing it: `sammon`, for performing a Sammon projection of data, and `umat`, for applying the cluster-discovery UMatrix [Ultsch, 1993] algorithm. We will use these programs in this work.

So far, Kohonen map has been used for such diverse applications as protein secondary structure prediction [Andrade et al., 1993], information retrieval [Kaski et al., 1997], rum age visualization [Quesada et al., 2000], and algorithm visualization [Romero et al., 2003].

3. COMMUNITY DISCOVERY AND VISUALIZATION

For a community that can be described as a graph, such as the one we are dealing with here, there are several ways of defining subcommunities. The loose definition is that a subcommunity is a set of nodes that link more to each other than to other nodes *outside* the community; this concept is usually called a *clique.* However, other, less strong definitions of subcommunity, can be used: we would talk about *n-cliques, n-clans*, *k-plexes* and others. All of them are valid definitions, and can be used in some cases; however, some of them are restrictive, in the sense that they take into account only binary relations, and do not consider the direction of the link; in the case at hand, direction is important: usually, some blog that has been pointed to by other might not even be aware of it. Most of them do not create an easy visual picture of the community they are describing.

2The program is free and can be downloaded from http://www.cis.hut.fi/~hynde/lvq/.

But there are additional problems: for starters, there is no universally accepted definition of *community*. The informal definition would put a community as a set of websites related to each other in some way, usually by topic. However, there are at least two different ways of defining *relation*: one is content-based, and the other link-based. From the point of view of content, two websites are related if they deal approximately with the same topics. From the point of view of links, two websites are related if they link to each other in either direction. But these two definitions are actually correlated: Menczer has proved [Menczer, 2001] that pages that link to each other are semantically related. Besides, there are several additional problems with content-related communities: if you define a community by keywords, synonymous and hypernims might make some websites to be missed; and in rapidly-changing websites, such as weblogs, there is no single topic, even no single set of topics, that can define it. In any case, a term frequency/inverse document frequency representation is usually highly-dimensional, much more so than using links to other members of the set of webs that is going to be studied. For a small set of sites, link-based representation is much more compact.

In any case, content or links are used to create a complex network with the set of sites under study, and then a community must be defined as some measure that distinguishes, or makes apart, some sites from others. There are several possible structures that could be considered communities: *cliques,* or set of sites that link to each other, *bipartite cliques,* set of sites which all link to another, different, set of sites [Caldarelli, 2002], *k-cores or factions,* set of sites connected to, at most, *k* other sites in the group, or *bipartite cores*, which includes both the connector and the connected sites. Most of these structures can be computed with programs such as Pajek[3] or UCINET[4], but need, in advance, some parameters such as the number *k* of links or the number of cores we want to divide the original set into.

There are other algorithms that detect partitions of the original set according to properties of links, as opposed of properties of nodes. One of them is the Newman-Girvan algorithm [Girvan, Newman, 2002], which detects which links have the property that, when removed, isolate some part of the original set. Clusters are then computed according to the position of these links. This is an excellent algorithm for discovering communities, used, for instance, in [Guimerá et al., 2002], but, once again, it does not discover the internal structure of each community, or the characteristic that defines it.

In this paper, we will use Kohonen's self-organizing map as a tool for mapping communities, which allows easy community visualization, discovers the underlying topic that defines each one, and allows assigning new websites to a community just by looking at its links.

4. MAPPING WEBLOG COMMUNITIES

The set of websites we are working with corresponds to the weblogs hosted by the Spanish weblog site Blogalia (http://www.blogalia.com/); it hosts around 200 weblogs, of which only 162 actually link or are linked by other weblogs; these are the ones used in our study. All stories published in Blogalia up to September 2003 were used for the study; there were around eleven thousand, which included around seventeen thousand links. Of those, roughly a quarter were links to other members of the community. Each weblog is represented by the set of output links to other members of Blogalia. Of course, and due to this decision, other websites, or weblogs, are not considered, even having to ignore someones that are closer to some blogs hosted in Blogalia than most of the inhabitants of this site.

First, we used Pajek to plot input cores in this set. Input cores, as has been defined before, group blogs according to the number of input links and where they come from.

3Pajek can be downloaded from http://vlado.fmf.uni-lj.si/pub/networks/pajek/
4UCINET can be downloaded from http://www.analytictech.com/

Figure 3. Graph representation of all blogs in blogalia, made with Pajek. Nodes are colored according to the core they belong to. The most densely connected core is indicated by the purple color. Yellow nodes are the most sparsely connected, and the other colors indicated several qualities in between. There are around 10 different clusters.

On the other hand, if we use *factions* as a definition of sub-community, and compute it using UCINET, we obtain the division shown in Table .

Table 1. Division of Blogalia into factions, as computed by UCINet. The number of factions was preset to 3. All the blog URLs are in the form: http://NAME.blogalia.com/, where the name is the string shown here.

Faction	Components
#1	caboclo esbardalladas silly tubo oracle ender pacotilla hazte-escuchar dragon palabrejas jaio-la-espia dibujante walkyria tse1 saliva mp bilbao polinesia elforastero superiores terisa simbiosis ljtarrio yildelen quotidianum gargantua1 oier smith chewie odisea osito yamato canopus evasivas clio prestige copensar rimero gargantua peaton aeiou akin eledhwen gnudista paleofreak jomaweb pawley ciencia15 daurmith jkaranka verbascum blogzine fbenedetti javarm atalaya www rvr fernand0
#2	tannhauser cuentacuento qotidianum jarvarm spamzoo russellbeattie demetro humedadrelativa vendell unhombretranquilo angelina barbara protoastronomo ocio hunter circulos reval 6cuerdas trunks bontos fondoazul guetto gripe acuarioland cacharreando electroduende aire neutrina mayoral miralado ie teo yogurtu amsel xdreus crisei bep cothinkhealth omar pepino entrelineas sanador exploraciones munchi borja copensalud planetaneverland confrontacion blojj metro prueba blogometro
#3	arclnx gofio miatalaya aldor yamisa melicerte latino estilo-005 gaecosita estilo-007 estilo-006 feo riviera kerberos estilo-004 mikel estilo-05 estilo-001 estilo-003 batiburrillo estilo-002 beta erizoazul magufos elcubo profes forward isilien maiz elda hispamed cominaii sieyin kakasico luiso morwen ventanas putten cca pipodols jcohen cthulhunam rubenlnx robertfernandez mirada escepticismo neuronal enpelotas hadez desarrollo rivendel hronia

The same data was analyzed using Kohonen's self-organizing map. The software used was SOM_PAK version 3.2, with the parameters shown in table .

Table 2. Parameters used to train Kohonen's self-organizing map in this paper.

Parameter name	Value
Neighborhood type	**Hexa**
Neighborhood function	**Bubble**
Map x size	**8**
Map y size	**8**
First training period: length	**2000**
Neighborhood radius	**10**
Training constant	**0.1**
Second training period: length	**10000**
Neighborhood radius	**1**
Training constant	**0.01**

From the links array, two different analysis were performed: by rows and columns. Rows represent the set of blogs every blog links to, and columns represent the set of blogs that links to a particular one. That means that SOM was applied to blogs represented by *incoming* and *outgoing* links. On each map, Umatrix analysis [Ultsch, 1993] was applied: this analysis shows how the set is clustered, so that *natural* clusters tend to stand out.

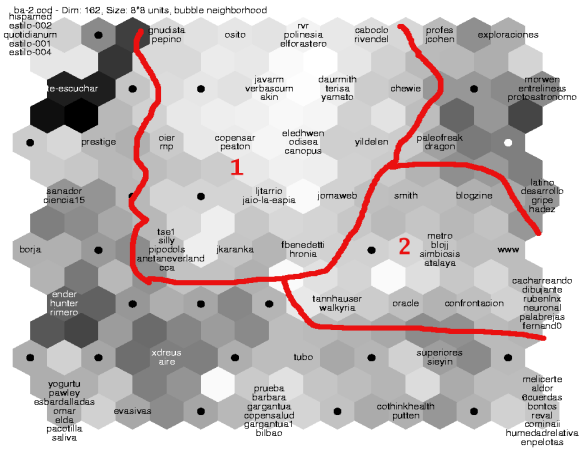

Figure 4. UMatrix map obtained from the SOM trained using rows as input. Two different areas can be observed, and have been labeled as 1 and 2; clusters correspond to "clear" zones separated by dark hexagons.

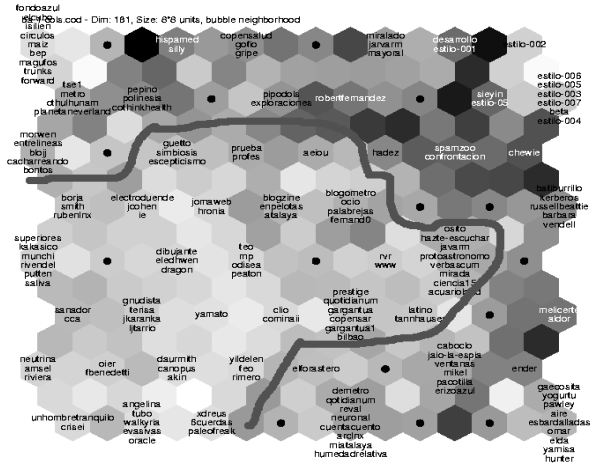

Figure 5. UMatrix map obtained form the SOM trained using columns as input. A single block or cluster is observed; it has been marked with a dark line

Different results have been obtained by training representing blogs by incoming or outgoing links. In the first case, shown in figure 5, a single block, containing the most usually linked-to blogs, stands out. This block roughly corresponds to the purple core shown in figure 3, and the first faction shown in table 1. The scenario that uses outgoing links is shown in figure 4 is a bit more discriminating, but, once again, makes apart factions and cores as computed by other methods.

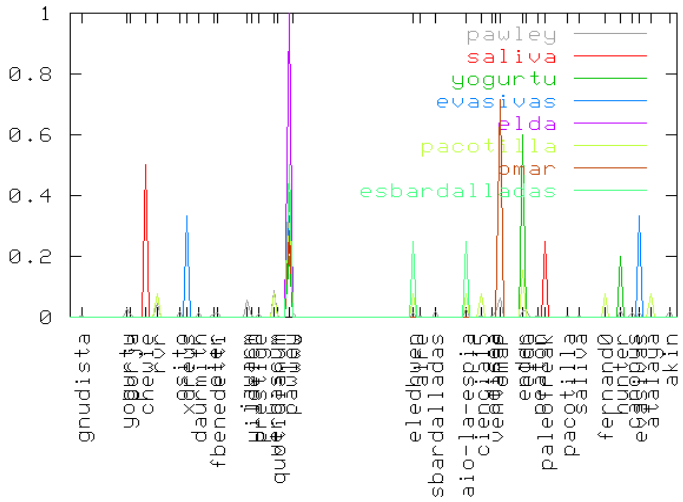

Figure 6 Plot of relative outgoing link strength for those blogs placed by the SOM in the SW corner

But it would also be interesting to look at what makes blogs cluster together in a single node, or what they have in common. It would be cumbersome to look at each and every node, but, if we look at a couple of them, for instance, the Southwest corner of figure 4, we obtain the plot shown in figure 6: most of them have a peak of links to pawley; for instance, elda has a single link, and it corresponds to that blog; pawley has also many links to itself, and so on. There are also some other coincidences: a few links to omar, for instance.

The same happens with the remaining clusters: they have many links to a blog or set of blogs, which makes the euclidean distance among them relatively small. That means that the blogs mapped to a single node roughly correspond to *bipartite cores* [Caldarelli, 2002], that is, set of nodes whose link pattern is similar.

There is an additional advantage in using Kohonen's self-organizing map: besides being able to distinguish among different groups, we can navigate using them. Since we now that blogs mapped to a single node are close to each other, and are also close to the blogs mapped to the nodes surrounding them, we could create a path from one blog to another, or use it as a recommendation for users or maintainers of a single blog. Since it works as a mathematical map, another blog, not belonging to this community, can also be mapped to it just by taking into account links to the set of blogs already mapped (or links from them).

5. CONCLUSION

Web content creation has undergone lately, under the influx of easy content-management programs such as weblogs, an extraordinary expansion, which, so far, shows no sign of abating. Interest groups are created spontaneously among web users, and it is interesting to study and identify these groups from the sociological, economical and technological point of view. Since web-community formation is generally spontaneous, without an explicit register or inscription by those that integrate them, and, besides, a particular website might belong to several communities, one of the first problems posed by its study is their identification and representation.

In this paper, we propose using a well-known technique: Kohonen's self-organizing maps. This technique ha been tested on many different problems, yielding good results, and has got a good community of experience. As has been shown in this paper, communities identified by analysing self-organizing maps using UMatrix are on a par with those identified using other techniques, such as *faction* analysis or *core* extraction, with the additional advantage that *community navigation* can be done by using the map: blogs on the same node, or adjacent nodes, *belong* (in a fuzzy sense) to the same community. The self-organizing map, besides highlighting the different communities and groups present on the sample, make an useful visual representation.

The authors of this work intend to follow up on it along one of these lines:

- Using self-organizing maps to visualize evolution of a set of blogs, and the community formation that goes along with it, by mapping different stages in its life.
- Using other algorithms, such as a fuzzy version of Kohonen's self-organizing map [Pascual, 2000].
- Applying different representations for each blog, using blog content, instead of blog links: for instance, TFIDF (term frequency/IDF) or latent semantic analysis.

ACKNOWLEDGEMENT

This paper has been funded in part by project TIC2003-09481-C04, of the Spanish ministry of science and technology, and a project awarded the Quality and Innovation department of the University of Granada. Fernando Tricas is with the Group of Discrete Event Systems Engineering (GISED), and his work is partially supported by project TIC2001-1819, financed by the Spanish Ministerio de Ciencia y Tecnologia. We are also grateful to Víctor Ruiz for his creation and continuing support of blogalia, and his members, for their support during the realization of this work.

REFERENCES

Caldarelli, G. 2002. *Introduction to complex networks*. Proceedings of the 7th Conference on Statistical and Computational Physics Granada. Online at http://www.cosin.org/Publications.html.

Girvan, M., & Newman, M. 2002. Community structure in social and biological networks. *Proc. natl. acad. sci.*, **99**(12), 7821-7826.

G.Romero, M.G.Arenas, P.A.Castillo, & J.J.Merelo. 2003.Visualization of neural network evolution. Lecture Notes in Computer Science,LNCS, nos. 2686-2687, pp 534-541. Springer-Verlag.

Guimera, R., Danon, L., Diaz-Guilera, A., Giralt, F., & Arenas, A. 2002 (Nov). *Self-similar community structure in organisations*. Condensed Matter, abstract cond-mat/0211498. Available online at: *http://arxiv.org/abs/cond-mat/0211498*.

Kaski, S. 1997. Computationally efficient approximation of a probabilistic model for document representation in the websom full-text analysis method. *Neural processing letters*, **5**(2), 69-81.

Kohonen, T. 1990. The self-organizing map. *Procs. IEEE*, **78**, 1464 ff.

A. Pascual; M. Bárcena; Juan-Julián Merelo-Guervós; J. M. Carazo, 2000. Mapping and fuzzy classification of macromolecular images using self-organizing neural networks. *Ultramicroscopy*, **84**, 85-99.

Quesada López-Martínez, J.; Juan-Julián Merelo-Guervós; M. J. Oliveras;J. González; M. Olalla; R. Blanca; M. C. 2002. Application of artificial aging techniques to samples of rum and comparison with traditionally aged rums by analysis with artificial neural nets. *Journal of agricultural and food chemistry*, **50**(6), 1470-1477.

Menczer, Filippo. 2001 (June). *Links tell us about lexical and semantic web content*. Online at *http://arxiv.org/abs/cs.IR/0108004*. Technical Report Computer Science Abstract CS.IR/0108004.

Morán, M. A. Andrade; P. Chacón; Juan-Julián Merelo-Guervós; F. 1993. Evaluation of secondary structure of proteins from UV circular dichroism spectra. *Protein engineering*, **6**(4), 383-390.

Ultsch, Alfred. 1993. Self-organizing neural networks for visualization and classification. *Pages 307-313 of:* Opitz, O., Lausen, B., & Klar, R. (eds), *Information and classification*. London, UK: Springer.