An Extensible Framework to Sort Out Nodes in Graph-based Structures Powered by the Spreading Activation Technique: The ONTOSPREAD approach.

Jose María Álvarez Rodríguez
(Department of Computer Science, University of Oviedo, Spain josem.alvarez@weso.es)

José Emilio Labra Gayo
(Department of Computer Science, University of Oviedo, Spain jelabra@weso.es)

Patricia Ordoñez De Pablos
(Department of Business Management, University of Oviedo, Spain patriop@uniovi.es)

Abstract The aim of this paper is to present an extensible framework for the Spreading Activation technique. This technique is supported by the ONTOSPREAD framework enabling the development, configuration, customization and execution of the Spreading Activation method on graph-based structures. It has been used for a long time to the efficient exploration of knowledge bases built on semantic networks in Information and Document Retrieval domains. Now the emerging Web of Data and the sheer mass of information now available make it possible the deployment of new services and applications based on the reuse of existing vocabularies and datasets. A huge amount of this information is published using semantic web languages and formats such as RDF, implicit graph structures developed using W3C standard languages: RDF-Schema or OWL, but new flexible and scalable methods to create added-value services and exploit the data are required. That is why ONTOSPREAD is considered to be relevant in order to provide a new way to implement the double process of activation and spreading of concepts. in graph-based structures, more specifically to browse and rank resources in the Web of Data realm. The original constraints like weight degradation according to the distance are provided in combination with others coming from the extension of this technique like the converging paths reward. Finally an evaluation methodology and two examples using the well-known ontologies GALEN and SNOMED CT are presented to validate the goodness, the improvement and the capabilities of this technique applied to an specific domain like clinical decission support systems.

Key Words: information and document retrieval, decision-support systems, recommending and tagging systems, algorithms, api
Category: K.4, K.4.1, D.2, D.2.2

1 Introduction

The Spreading Activation technique (hereafter SA) introduced by [7], in the field of psycho linguistics and semantic priming, proposes a model in which all relevant information is mapped on a graph as nodes with a certain "activation

value". Relations between two concepts are represented by a weighted edge. If a node is activated their activation value is spread to their neighbor nodes. This technique was adopted by the computer science community and applied to the resolution of different problems, see Sect. 2, and it is relevant to the medical fields sector in the scope of: 1) construction of hybrid semantic search engines; 2) ranking of information resources according to an input set of weighted resources 3) recommendation of medical terms using well-known ontologies in a particular sector and 4) decision-support easing the access to the information in large databases . Thus this technique provides a connectionist method to retrieve data like brain can do. Although SA is widely used, more specifically in recent years has been successfully applied to ontologies, a common and standard framework is missing and each third party interested in its application must to implement its own version [31] of SA.

Taking into account the new information realm and the leading features of putting together the SA technique and the Semantic Web and Linked Data initiatives, new enriched services of searching, matchmaking, recommendation or contextualization can be implemented to fulfill the requirements of access information in different trending scopes like e-health, e-procurement, e-tourism or legal document databases [2]. More specifically in the e-health sector there is a growing need to automate the processes related to the tagging of electronic clinical records and to create tools for the clinical decision support. That is why SA is relevant to the field of clinical knowledge management technologies through its capacity to process large databases and exploit the know-how of previous records easing the recommendations of information resources.

The proposed work aims to provide a framework for SA to ease the configuration, customization and execution over graph-based structures and more specifically over RDF graphs and ontologies. It is relevant to medical systems access and interoperability due to the fact that this technique is based on a set of proven algorithms for retrieving and recommending information resources in large knowledge bases. Following the specific contributions of this work are listed: 1) study and revision of the classical constrained SA; 2) study and definition of new restrictions for SA applied to RDF graphs and ontologies; 3) implementation of a whole and extensible framework (called ONTOSPREAD) to customize and perform the SA based; 4) outlining of a methodology to configure and refine the execution of SA and 5) an example of configuration and refinement applying SA over two well-known ontologies: GALEN and SNOMED-CT.

1.1 Organization

This paper is structured as follows: in Section 2, we review the relevant work in medical systems and the common applications of SA. In Section 3, we provide a description of the design and implementation of an open framework for SA

technique, explaining the algorithm, restrictions, etc. Afterwards, in Section 4 we apply the ONTOSPREAD framework over the GALEN and SNOMED-CT ontologies to evaluate the SA technique for recommending concepts in medical systems. Finally, we evaluate the results of the previous executions and present some conclusions.

2 Related Work

Since SA was introduced by [7] in the field of psycho linguistics and semantic priming it has been applied to the resolution of problems trying to simulate the behavior of the brain using a connectionist method to provide an "intelligent" way to retrieve information and data.

The use of SA was motivated due to the research on graph exploration [25, 1]. Nevertheless the success of this technique is specially relevant to the fields of Document [17] and Information Retrieval [6]. It has been also demonstrated its application to extract correlations between query terms and documents analyzing user logs [8] and to retrieve resources amongst multiple systems [29] in which ontologies are used to link and annotate resources.

In recent years and regarding the emerging use of ontologies in the Semantic Web area new applications of SA have appeared to explore concepts [26, 5] addressing the two important issues: 1) the selection and 2) the weighting of additional search terms and to measure conceptual similarity [15]. On the other hand, there are works [18] exploring the application of the SA on ontologies in order to create context inference models.The semi-automatically extension and refinement of ontologies [20] is other trending topic to apply SA in combination with other techniques based on natural language processing. Data mining, more specifically mining socio-semantic networks[32], and applications to collaborative filtering (community detection based on tag recommendations, expertise location, etc.) are other potential scenarios to apply the SA theory due to the high performance and high scalability of the technique. In particular, annotation and tagging [19] services to gather meta-data [12] from the Web or to predict social annotation [4] and recommending systems based on the combination of ontologies and SA [10] are taken advantage of using SA technique. Also the semantic search [30] is a highlight area to apply SA following hybrid approaches [2, 28] or user query expansion [24] combining metadata and user information.

Although SA is widely accepted and applied to different fields open implementations[1] are missing. Moreover the Apache Mahout [2] project, a recent scalable machine learning library that supports large data sets, does not include an implementation of SA instead of providing algorithms for the classification, clus-

[1] Texai company (`http://texai.org/`) offers a proprietary implementation of SA.
[2] `http://mahout.apache.org/`

tering, pattern mining, recommendation and collaborative filtering of resources in which SA should be representative.

From a medical point of view the majority of errors in health delivery systems are not necessarily due to human errors, in some cases the organization and the processes and services are in charge of some decissions [3]. Clinical Decision Support Systems (CDSS) [16, 22] show potential benefits in their integration with the existing human practices including the ability to:

- influence clinicians behaviour and reduce variability of outcomes across various health professionals and increase the standardisation of processes towards evidence-based guidelines.
- combine and synthesise complex related pieces of information.
- facilitate access to clinical information and reporting of results through greater accessibility of data and improved display of information.
- identify patterns within the patient data which must be acted upon (e.g. abnormal or inconsistent findings, alerts,...)
- ...
- doing all of the aforementioned while preserving the independence of health professionals.

CDSSs can become important process standardisation and error preventing tools but the inherent difficulty and complexity in designing explicit conceptual models of the medical diagnosis and algorithms to exploit this information would limit the usefulness of such systems. Besides they have other inherent limitations: recommendations issued by the systems can only be as good as the techniques can manage the models, data and information. In that sense new approaches have appeared to develop clinical diagnosis systems by using semantic web technologies to infer diseases from symptoms, signs and laboratory tests formalized as logical descriptions like [11, 13, 14]

In the case of ontologies in medical systems, two main well-known ontologies are being used:

1) GALEN [27] is a large project developing terminology servers and data (23,141 concepts and 950 relations) entry systems based on a Common Reference, or CORE [3], model for medical terminology. The authors use the term "ontology" here to indicate the model of the categories within the universe of discourse, plus sufficient information about those categories to allow them to be classified automatically. They take"ontologies" to be language independent, using the broader term 'terminology' for an ontology linked to linguistic information.

[3] `http://www.opengalen.org/`

The GALEN ontology is attempting to meet five challenges[4], highlighting: 1) to facilitate clinical applications and 2) to bridge the gap between the detail required for patient care and the abstractions required for statistical, management, and research purposes.

2) SNOMED has been used in medical information systems like Pathology information systems (PathIS) for many years. Cancer Protocols published by the College of American Pathologists (CAP) are being encoded with SNOMED CT concepts with the main objective of consistently capture the explicit meaning of each checklist item. This can simplify reporting for cancer registries and improve retrieval and analysis of cancer data [33]. In [23] a system to classify lung TNM (Classification of Malignant Tumors) stages from free-text pathology reports, using SNOMED CT for the extraction of key lung cancer characteristics from free-text reports is presented. Other application of SNOMED CT is a system [21] to automatically assign SNOMED CT codes for anatomic sites, tissues, pathologic findings and diagnoses in full-text pathology reports (excluding cytology) to SNOMED CT concept descriptors, that shows a positive predictive value for anatomic concepts of 92.3% and positive predictive value for diagnostic concepts of 84.4% .Nowadays, the most common uses [9] of SNOMED CT are concept search (72%) and coding of clinical data (60%). On the other hand, the January 2011 SNOMED CT International Release content hierarchy includes more than 293,000 active concepts with formal logic-based definitions, organized into top-level hierarchies. SNOMED CT contains more than 765,000 active English-language descriptions for flexibility in expressing clinical concepts. It also provides more than 830,000 logically-defining relationships enable consistency of data retrieval and analysis.

This review of GALEN and SNOMED implies that 1) they are widely accepted in the construction of medical systems with different purposes and 2) they represent a large controlled vocabularies and models with logical foundations that requires efficient algorithms for its exploitation and navigation.

3 ONTOSPREAD Framework

3.1 Background

In this section, the theoretical model of *SA* [7, 25] is reviewed to illustrate the basic components and the operations performed by SA during their execution, specially the spreading of the activation from a node to their adjacent nodes. This model is made up of a conceptual network of nodes connected through relations (conceptual graph). Taking into account that nodes represent domain objects or classes and edges relations among them, it is possible to establish a

[4] `http://www.opengalen.org/background/background0.html`

semantic network in which SA can be applied. The process performed by the algorithm is based on a thorough method to go down the graph using an iterative model. Each iteration is comprised of a set of beats, a stepwise method, and the checking of a stop condition. SA is comprised of three stages: *Preadjustement* and *Postadjustement* that are usually in charge of performing some control strategy over the target semantic network and the set set of activated concepts and the *Spreading* stage in which concepts are activated in activation waves. The calculation of the activation rank I_i of a node n_i is defined as follows:

$$I_i = \sum_j O_j \omega_{ji} \tag{1}$$

I_i is the total inputs of the node n_i, O_j is the output of the node n_j connected to n_i and ω_{ji} is the weight of the relation between n_j and n_i. If there is not relation between n_j and n_i then $\omega_{ji} = 0$.

The activation function f is used to evaluate the "weight" of a node and decide if the concept is active.

$$N_i = f(I_i) = \begin{cases} 0 & \text{if } I_i < \jmath_i \\ 1 & \text{if } I_i > \jmath_i \end{cases} \tag{2}$$

N_i is 1 if the node has been activated or 0 otherwise. $\jmath_i$, the threshold activation value for node i, depends on the application and it can change from a node to others. The activation rank I_i of a node n_i will change while algorithm iterates.

3.2 Constrained Spreading Activation

One of the leading features of SA technique is its flexibility to fit to the resolution of different kind of problems. From the configuration point of view some constraints presented in [6] have been customized to improve the expected outcomes of the execution according to the domain problem.

Distance: nodes far from an activated node should be penalized due to the number of needed steps to reach and activate them.

Path: the activation path is built by the activation process from a node to other and this process can be guided according to the weights of relations (edges).

Multiple outputs (Fan-Out): "highly connected" nodes can guide to a misleading situation in which activated and spread nodes are not representative, these nodes should be skipped or penalized by the algorithm.

Threshold activation: a node n_i will be spread iif its activation value, I_i, is greater than a threshold activation constant $\jmath$.

The aforementioned theoretical model is an excellent start point to design a framework for *SA* but from the domain expert point of view some configuration requirements to apply this technique to ontologies are missing. That is why a set of extensions are proposed to deal with the specific features of RDF graphs and ontologies.

Context of activation $\mathbb{D}_{com}$: the framework is able to manage some ontologies at the same time and concepts can be defined in different ontologies identified by a context URI (or namespace). The double process of activation and spreading will only be performed in the set of active contexts $\mathbb{D}_{com}$.

Definition 1. Let $\mathbb{D}_{com}$ an active domain, if a concept c_i is activated o spread then $c_i \in \mathbb{D}_{com}$.

Minimum activation value $N_{\min}$: only concepts with an activation value N_k greater than $N_{\min}$ will be spread. This constraint comes from the theoretical model of *SA*.

Maximum number of spread concepts $\mathbb{M}$: the process of activation and spreading will be performed, at the most, until $\mathbb{M}$ concepts had been spread.

Minimum number of spread concepts $\mathbb{M}_{\min}$: the process of activation and spreading will be performed, at least, $\mathbb{M}_{\min}$ concepts had been spread.

Time of activation t: the process of activation and spreading will be performed, at the most, during t units of time.

Output Degradation O_j: one of the keypoints to improve and customize the algorithm is to define a function h that penalizes the output value O_j of a concept c_j.

1. Generic customization: h calculates the output of a concept c_j according to its degradation level.

$$O_j = h(I_j) \tag{3}$$

Basic case: if $h_0 = id$, the output value O_j takes the level of the activated concept c_j as its value.

$$O_j = h_0(I_j) = I_j \tag{4}$$

2. Customization using **distance**: h_1 calculates the level activation of the concept c_j according to the distance from the initial concept $c_l \in \Phi$[5] to

[5] Set of initial concepts.

the node that has activated it. The activation value should decrease if the distance from Φ grows thus the algorithm follows a path from c_l to c_j: $I_l > I_j$.

The function h_1 penalizes the output of concepts (decreasing their rank) far from the "activation core" and rewards closed concepts. Thus, let d_j, where $d_j = min\{d_{lj} : \forall n_l \in \Phi\}$:

$$O_j = h_1(I_j, d_j) = \frac{I_j}{d_j} \tag{5}$$

3. Customization using **beats**: the function h_2 calculates the degradation of the concept using the number of iterations k:

$$O_j = h_2(I_j, k) = (1 + \frac{I_j}{k}) \exp(-\frac{I_j}{k}). \tag{6}$$

3.3 Specification of the SA technique

The entry point to SA technique is the set of initial concepts that will generate a new set of the most relevant concepts. Ontologies based on the RDF graph model are a graph where each node n_i represents a concept c_i and the edge ω_{ji} is the semantic relation between c_j y c_i. The final result of the algorithm is a set of sorted pairs (n_i, I_i) that builds the set of output concepts, where $n_i \approx c_i$ and $I_i \approx w_i$ (the relevance of the concept). The implementation of SA, see Algorithm 3.3, comprises of two sets of concepts that store information about the state of the algorithm: 1) $\mathbb{D}_{com}$ are all the concepts in the semantic network and 2) Φ is the set of initial activated concepts, c_j^k is the spreading concept at the k-th iteration (from which other concepts are activated).

Set $\mathcal{A}$: queue of **activated** concepts (candidates to be spread).

$$\mathcal{A}^0 = \Phi \tag{7}$$

$$\mathcal{A}^k = (\mathcal{A}^{k-1} \cup \{c_i : \forall c_i / \omega_{ji}^k > 0\}) - \{\mathcal{G}^k\} \tag{8}$$

Set $\mathcal{G}$: set of **spread** concepts.

$$\mathcal{G}^0 = \emptyset \tag{9}$$

$$\mathcal{G}^k = \mathcal{G}^{k-1} \cup \{c_j^k\} \tag{10}$$

Finally, the calculus of the activation value of a concept c_i at iteration k, indicated by I_i^k, is defined. At 0 iteration the activation value c_i is calculated as follows:

$$I_i^0 = \begin{cases} 1 & \text{if } c_i \in \Phi \\ 0 & \text{if } c_i \notin \Phi \end{cases} \tag{11}$$

at k iteration, the activation value of c_i from element c_j^k to c_i is calculated as follows:

$$I_i^k = \begin{cases} I_i^{k-1} & \text{if } \omega_{ji}^k = 0 \\ I_i^{k-1} + \omega_{ji}^k I_j^{k-1} & \text{if } \omega_{ji}^k > 0 \end{cases} \tag{12}$$

Algorithm 1 *Pseudocode of Spreading Activation*

Require: $\Phi \neq \emptyset$
Ensure: $\mathcal{G} \neq \emptyset$
  $\mathcal{A} \leftarrow \Phi$
  $\mathcal{G} \leftarrow \emptyset$
  while $\mathcal{A} \neq \emptyset$ AND $card(\mathcal{G}) < \mathcal{G}_{\min}$ AND $N_k \geq N_{\min}$ **do**
    $n_k \leftarrow extract(\mathcal{A})$
    $\mathcal{G} \leftarrow \{n_k\} \cup \mathcal{G}$
    for all $n_i / w_{ki} > 0$ **do**
      $N_i \leftarrow N_i + w_{ki} N_k$
      $\mathcal{A} \leftarrow (\{n_i\} \cup \mathcal{A}) - \mathcal{G}$
    end for
  end while
  return $\mathcal{G}$

3.4 Improving Spreading Activation

Some improvements in the calculus of the activation value of a concept have been introduced in order to get a more complete and accurate technique. If some paths of activation converge to the same node and the source nodes are different then this node should be relevant and a reward is applied to the nodes presented in these paths.

Definition 2. Let p_i the number of paths that start and finish in different nodes of Φ[6] and they go through the node c_i and they only contain nodes belonging to $\mathcal{G}$. This improvement assigns a new value to the activation value of each node c_i indicated by I_i^* and it is calculated by means of the function g:

$$I_i^* = g(I_i, p_i) \tag{13}$$

[6] The reward is not applied to nodes in Φ.

In this case, a relaxed reward function has been chosen, Eq. 14, and, it is applied in the *Postadjustment* stage, thus the original semantics and behavior of *SA* algorithm remains.

$$g(x, y) = x(\log(y + 1) + 1) \tag{14}$$

x is the reward constant, it can be defined according to the context and y is the number of times that a concept c_i must be rewarded.

3.5 Refining Spreading Activation

The whole configuration of the algorithm can be made by default but a customization to a particular domain should be carried out by a domain expert taking into account the specific issues of that domain and considering it as a new stage of the ontology o graph modeling process. Since *SA* uses weights in relations to calculate the activation value of the concepts, different "patterns" have been identified to manage the direction of the spreading process: 1) Ascending seeks for the activation of concepts more generic than the current ("superclass"); 2) Descending seeks for the activation of concepts more specific than the current ("subclass"); 3) Nominal seeks for the activation of instances instead of concepts ("instance of") and 4) Crossing seeks for the activation of concepts and instances connected through a certain relation $\mathcal{R}$. These control patterns can be put together in order to fit as much as possible the focus and direction of the double process of activation and spreading.

3.6 Design and implementation of ONTOSPREAD

ONTOSPREAD framework[7], see Fig. 1 is addressed by an open and extensible design applying best practices on software design and development like design patterns and refactoring. The *Player* class handles the execution of the algorithm in a stepwise way. It is an application of the *Iterator* design pattern to perform the activation and spreading processes. The state of the algorithm is captured in a separate class (*OntoSpreadState*) that makes possible to serialize the current state and back to a previous one. On the other hand, the *SA* process comprises of three sub-processes (see Sect. 3.1): *OntoSpreadPreAdjustment*; *OntoSpreadRun*; and *OntoSpreadPostAdjustment*. Moreover, the process carries on the information about the knowledge base using the DAO pattern thus the framework is independent from the modeling language of the semantic network (RDF-based vocabularies and OWL are now supported).

The keypoint to design the algorithm lies in how and where the information will be available at different iterations. Secondly, an unique entry point to

[7] `http://code.google.com/p/ontospread/`

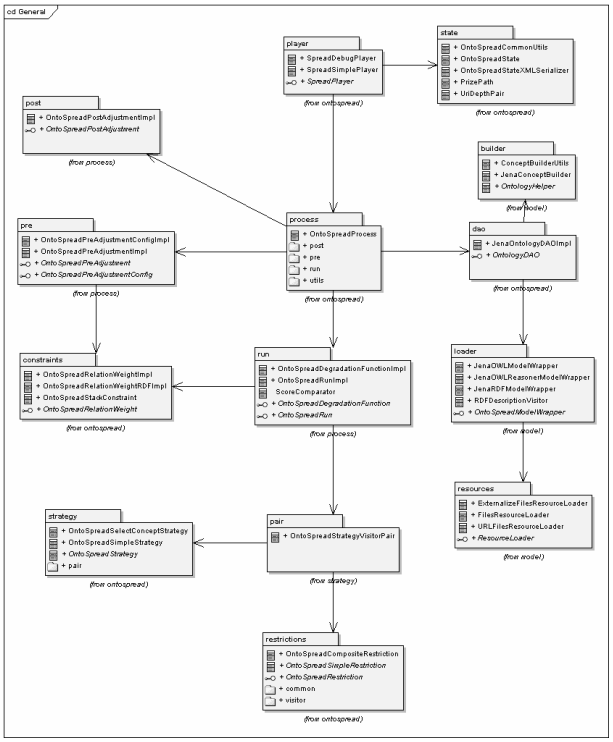

Figure 1: ONTOSPREAD Overview Diagram.

the state of the algorithm should be available trying to avoid illegal accesses. This object (*OntoSpreadState*) stores the next information: 1. Spread concepts. 2. Active concepts. 3. Paths of activation. 4. Concept to be spread. 5. Generic swap area (to share information among iterations). Moreover, the extensibility and flexibility of the algorithm is subjected to a good design of the restrictions and their evaluation process. The next features and design patterns are used to design and implement the model of restrictions of SA:

- Any restriction can be considered as a simple restriction and can be evaluated to a boolean value.
- Conditions or actions in the algorithm can be comprised of several restrictions.
- The extension points of the algorithm, included through a *Template Method* design pattern, are strategies to carry out an specific action. Each strategy can be subjected to one or more restrictions.
- Each restriction can be simple or comprised of others. *Composite* pattern.

- Each action is an strategy. *Strategy* pattern.
- A strategy implies one restriction (or a set of them) thus the strategy is a client of the *Composite* of restrictions.
- The evaluation of the restrictions to get their value (boolean) is carried out through a *Visitor* pattern that fits perfectly to evaluate and walk in composite objects. It consists on: apply the strategy that modifies the state of the algorithm and assert this change of state by means of the restrictions applied to this strategy.

3.7 Development of ONTOSPREAD API and Supporting Tools

The development of the API has been performed using Semantic Web and Java technologies like: Jena[8] API, JAXB[9], Maven[10] o Spring[11]. Moreover two tools are provided to test and debug different configurations of the algorithm:

ONTOSPREAD-TEST It is a tool for the automatic execution and reporting of batch tests. It provides an user-oriented framework to configure, combine and load several configurations (e.g. restrictions, weights, initial concepts, etc.) for *SA* and get results. A XML vocabulary using XML-Schema and the *Extensible Content Model* XML design pattern has been defined to build the configuration of the *SA* process. The designing of this vocabulary is oriented to be used with JAXB, this technology enables us automatically the processes of marshalling and unmarshalling Java classes providing an easy way to configure, load and serialize the different configurations and results.

ONTOSPREAD Inspector. It is a graphical debugger of *SA* algorithm using the graph library JpowerGraph[12] and the SWT[13] toolkit. It enables to configure, load, run (one step or stepwise) and view the evolution of the semantic network with the concepts (activated, spread, weights, relations, etc.).

Finally, a set of source code metrics using Eclipse[14] and the metrics plugin[15] have been extracted to demonstrate and measure its quality. Table1 presents these metrics that measures cohesion and coupling of the software.

[8] `http://jena.sf.net`
[9] `http://java.sun.com/developer/technicalArticles/WebServices/jaxb/`
[10] `http://maven.apache.org`
[11] `http://www.springframework.org/`
[12] `http://jpowergraph.sf.net`
[13] `http://www.eclipse.org/swt/`
[14] `http://www.eclipse.org`
[15] `http://metrics.sf.net/`

Table 1: Source Code Metrics

Source Code Metrics						
ID	*Def.*	*Total*	*Avg.*	*Std. Dev.*	*Max*	*Scope*
TLOC	Total Lines of Code.	5272				
CA	Afferent Coupling.		6.524	10.545	48	Package.
RMD	Normalized Distance, $\|RMA + RMI - 1\|$.		0.32	0.347	1	Package.
NOM		0.065	0.296	3		Type.
RMI	Instability: $CE/(CA + CE)$		0.567	0.387	1	Package.
NBD	Nested Block Depth.		1.204	0.516	4	Method.
LCOM	Lack of Cohesion of Methods (*Henderson-Sellers*).		0.18	0.289	0.957	Type.
VG	McCabe Cyclomatic Complexity.		1.297	0.735	6	Method.
RMA	Abstractness.		0.113	0.186	0.667	Package.
CE	Efferent Coupling.		1.976	1.282	5	Package.
DIT	Depth of Inheritance Tree.		1.607	0.915	4	Type.

All of these values are in the default range defined in the plugin thus the quality of the source code with the desired feautures of *high cohesion* and *low coupling* are assured.

4 Evaluation of ONTOSPREAD

The validation of the algorithm depends on the configuration of the activation and spreading processes to fit it to the different domain issues. SA is determined by the target semantic network and therefore the defined domain knowledge (concepts and relations) is the key part to adjust its behavior. On the other hand, taking into account that the activation and spreading is guided by the weights of relations their specification is fundamental to get the desired outputs. The methodology to test the implementation of the algorithm is subjected to these conditions but a step-wise refinement method can be outlined:

1. Use a well-known semantic network (ontology, etc.): concepts and relations.
2. Define a potential set of initial concepts (Φ) and their initial activation value (usually 1.0).
3. Specify the weights of the relations to that domain knowledge.

4. Combine the different restrictions provided by the framework.
5. Select the degradation function.
6. Add the reward techniques to increase the activation value of certain nodes.
7. Try to evaluate new activation functions for their further implementation.
8. Repeat these steps until getting the most appropriated set of output concepts to that domain knowledge.

To apply this methodology, the GALEN and SNOMED CT [16] ontologies have been selected. They are well-known and referenced ontologies in the biomedicine domain and they are widely used in reasoning and decision support processes. The design of the experiment depends on: the ontology, the weights of relations, the set of initial concepts, the set of restrictions, the degradation function and the extensions to reward nodes. In the case of GALEN, the set of initial concepts (Φ) with an initial value 1.0 is: "#AdvancedBreastCancer" and "#NAMEDSymptom". The weights of the relations are fixed to a default value of 1.0. On the other hand, the set of initial concepts (Φ) with an initial value 1.0 in SNOMED-CT is: "#Articular_cartilage_of_lunate" and "#Articular_tissue_sample"

The refinement of the algorithm will enable us to get a set of output concepts similar to the process that a brain will do. The degradation functions and the reward technique will be alternatively combined checking the output of the algorithm.

After the execution of the different configurations, see Tab. 2 and Tab. 3, some statistics have been extracted out of the results. The main differences between the tests lies in the number of activated nodes and their activation values due to the restrictions that guide the evolution of the algorithm through the graph and the structure of the ontologies. It is also remarkable that the reward of paths, in this case, does not imply changes in the output set. This situation demonstrates that a depth knowledge of the semantic network is needed to take advantage of the SA extensions. Nevertheless the output of the algorithm helps us to establish a set of weighted resources that can be used to retrieve documents, make recommendations or search in large databases with enriched queries.

5 Conclusions and Future Work

This work provides a configurable and extensible framework to support the SA technique. It allows the configuration of restrictions and their combination to

[16] The OWL version of SNOMED CT has been generated using "Simple SNOMED Module Extractor" provided by the OWL Research Group at Manchester University: `http://owl.cs.manchester.ac.uk/snomed/`. Other tests have been carried out using the OWL version of SNOMED CT provided by the IHTSDO.

Config & Stats/Test	T_1	T_2	T_3	T_4	T_5	T_6
Minimum activation value $N_{\min}$	1.0	1.0	1.0	1.0	1.0	1.0
Maximum number of spread concepts $\mathbb{M}$	50	50	10	10	50	50
Minimum number of spread concepts $\mathbb{M}_{\min}$	20	20	5	5	20	20
Output Degradation O_j	h_1	h_2	h_1	h_2	h_1	h_2
Reward (No,Yes)	N	N	N	N	Y	Y
Context of activation $\mathbb{D}_{com}$	DEFAULT					
Activated Nodes	62	79	15	15	62	79
Spread Nodes	20	20	5	5	20	20
Highest activation value	7.5	3.9896	1.5	1.90	7.5	3.9896
Deepest spread path	10	16	2	2	10	16
Concepts (name:value)	NAMED Symptom: 7.5, Primate: 2.28	Multi Cellular Eukaryota: 3.9896, Opis thokonts: 3.9885	NAMED Symptom: 1.5, Advanced Breast Cancer: 2.28	NAMED Symptom: 1.90, Advanced Breast Cancer: 1.90	NAMED Symptom: 7.5, Primate: 2.28	Multi Cellular Eukaryota: 3.9896, Opis thokonts: 3.9885
Time (msec.)	9	8	2	3	11	12

Table 2: Configuration and statistics of results after the execution and refinement of SA over the GALEN ontology.

Config & Stats/Test	T_1	T_2	T_3	T_4	T_5	T_6
Minimum activation value $N_{\min}$	1.0	1.0	1.0	1.0	1.0	1.0
Maximum number of spread concepts $\mathbb{M}$	50	50	10	10	50	50
Minimum number of spread concepts $\mathbb{M}_{\min}$	20	20	5	5	20	20
Output Degradation O_j	h_1	h_2	h_1	h_2	h_1	h_2
Reward (No,Yes)	N	N	N	N	Y	Y
Context of activation $\mathbb{D}_{com}$	DEFAULT					
Activated Nodes	136	76	16	16	136	76
Spread Nodes	20	20	5	5	20	20
Highest activation value	2.16	9.49	1.5	1.85	4.14	13.68
Deepest spread path	10	7	3	3	10	16
Concepts (name:value)	Upper extremity part: 2.16, Articular cartilage of wrist joint: 1.66	Structure of radioulnar joint: 9.49, Inferior radioulnar joint structure: 8.5	Articular cartilage of lunate: 1.5, Wrist region structure: 1.0	Articular cartilage of lunate: 1.85, Joint structure of wrist and/or hand: 1.0	Upper extremity part: 4.14, Joint structure of wrist and/or hand: 1.2	Wrist joint structure: 13.68, Inferior radioulnar joint structure: 8.5
Time (msec.)	36	17	3	4	42	17

Table 3: Configuration and statistics of results after the execution and refinement of SA over the SNOMED CT ontology.

get the most accurate set of output concepts. One of the features that turns SA to a widely accepted algorithm lies in its flexibility but some disadvantages are also presented: the adjusting and refinement of restrictions and weights of the relations, the selection of the degradation function and the use of reward functions. This framework minimizes these advantages with an extensible library that can be applied to different scenarios like medical systems, in particular biomedicine, CDSS, etc. providing enriched services of annotation, searching or recommendation.

The main improvement in the algorithm consists on the flexibility of the refinement methodology. An automatic learning algorithm to create SA configurations according to ontologies should be developed. Thus, the training stage of SA could generate the best configuration for a specific domain. The algorithm could optimize the selection of input parameters like the weights of the relations, the degradation functions or the combination of restrictions. Beside new measures related to instances such as 'Cluster Measure", "Specifity Measure" or both could be used in the process of activation/spreading. Also the selection of the next node to spread is based on a "first better" strategy (if two nodes have the same activation value) because of this fact other selection strategies should be implemented. Finally a new version of the SA is being specified and developed following the Map/Reduce[17] programming model with the objective of getting a distributed version of this technique for processing large data sets.

Acknowledgements. ONTOSPREAD was initially developed in BOPA [2] project that is one of *Semantic Web Use Cases and Case Studies*[18] collected by W3C. Currently it is being applied to the process of searching public procurement notices in the "10ders Information Services"[19] project, partially funded by the Spanish Ministry of Industry, Commerce and Tourism (TSI-020100-2010-919) and the European Regional Development Fund (EFDR), leaded by "Gateway Strategic Consultancy Services"[20] and developed in cooperation with "Exis TI"[21] and WESO Research Group. Also we would like to thank the Spanish Ministry of Health for the license of SNOMED CT [22].

References

1. J. Anderson. A Spreading Activation Theory of Memory. *Verbal Learning and Verbal Behavior*, (1):261–295, 1983.
2. D. Berrueta, J. Labra, and L. Polo. Searching over Public Administration Legal Documents Using Ontologies. In *Proc. of JCKBSE 2006*, pages 167 – 175, 2006.

[17] `http://labs.google.com/papers/mapreduce.html`
[18] `http://www.w3.org/2001/sw/sweo/public/UseCases/CTIC/`
[19] `http://rd.10ders.net`
[20] `http://gateway-scs.es/`
[21] `http://www.exis-ti.com/`
[22] `http://www.msps.es/profesionales/hcdsns/areaRecursosSem/snomed-ct/`

3. M.-M. Bouamrane, A. Rector, and M. Hurrell. Experience of using owl ontologies for automated inference of routine pre-operative screening tests. In *Proceedings of the 9th international semantic web conference on The semantic web - Volume Part II*, ISWC'10, pages 50–65, Berlin, Heidelberg, 2010. Springer-Verlag.
4. A. Chen, H.-H. Chen, and P. Huang. Predicting social annotation by spreading activation. In *Proc. of ICADL'07*, pages 277–286, Berlin, Heidelberg, 2007.
5. H. Chen and T. Ng. An Algorithmic Approach to Concept Exploration in a Large Knowledge Network (automatic thesaurus consultation): Symbolic Branch-and-Bound search vs. connectionist Hopfield net activation. *J. Am. Soc. Inf. Sci.*, 46(5):348–369, 1995.
6. P. Cohen and R. Kjeldsen. Information Retrieval by Constrained Spreading Activation in Semantic Networks. *Inf. Process. Manage.*, 23(4):255–268, 1987.
7. A. Collins and E. Loftus. A spreading activation theory of semantic processing. *Psychological Review*, 82(6):407–428, 1975.
8. H. Cui, J. Wen, J. Nie, and W. Ma. Query Expansion by Mining User Logs. *IEEE Transaction on Knowledge and Data Engineering*, 15(4):829–839, July 2003.
9. G. Elhanan, Y. Perl, and J. Geller. A survey of direct users and uses of snomed ct: 2010 status. *AMIA Annu Symp Proc*, 2010:207–11, 2010.
10. Q. Gao, J. Yan, and M. Liu. A Semantic Approach to Recommendation System Based on User Ontology and Spreading Activation Model. In *NPC '08: Proc. of the 2008 IFIP*, pages 488–492, Washington, DC, USA, 2008. IEEE Computer Society.
11. Á. García-Crespo, A. R. González, M. Mencke, J. M. G. Berbís, and R. C. Palacios. Oddin: Ontology-driven differential diagnosis based on logical inference and probabilistic refinements. *Expert Syst. Appl.*, 37(3):2621–2628, 2010.
12. F. Gelgi, S. Vadrevu, and H. Davulcu. Improving Web Data Annotations with Spreading Activation. In *WISE*, pages 95–106, 2005.
13. A. R. González, J. E. L. Gayo, G. Alor-Hernández, J. M. Gómez, and R. Posada-Gómez. Adonis: Automated diagnosis system based on sound and precise logical descriptions. In *CBMS*, pages 1–8, 2009.
14. A. R. González, M. Mencke, G. Alor-Hernández, R. Posada-Gómez, J. M. Gómez, and A. A. Aguilar-Lasserre. Medboli: Medical diagnosis based on ontologies and logical inference. In *eTELEMED*, pages 233–238, 2009.
15. S. Gouws, G.-J. V. Rooyen, and H. Engelbrecht. Measuring Conceptual Similarity by Spreading Activation over Wikipedia's Hyperlink Structure. In *Proceedings of the 2nd Workshop on The People's Web Meets NLP: Collaboratively Constructed Semantic Resources*, pages 46–54, Beijing, China, August 2010.
16. R. Haynes, N. Wilczynski, and C. C. D. S. S. S. R. ccdss. Effects of computerized clinical decision support systems on practitioner performance and patient outcomes: Methods of a decision-maker-researcher partnership systematic review. *Implementation Science*, 5(1):12+, Feb. 2010.
17. H.R.Turtle. *Inference Networks for Document Retrieval.* PhD thesis, University Illinois, Urbana, IL, USA., 1991.
18. A. Katifori, C. Vassilakis, and A. Dix. Ontologies and the brain: Using spreading activation through ontologies to support personal interaction. *Cognitive Systems Research*, 11(1):25–41, 2010.
19. J. Labra, P. Ordoñez, and J. Cueva. Combining Collaborative Tagging and Ontologies in Image Retrieval Systems. 2007.
20. W. Liu, A. Weichselbraun, A. Scharl, and E. Chang. Semi-Automatic Ontology Extension Using Spreading Activation. *Universal Knowledge Management*, 0(1):50–58, 2005.
21. H. J. Lowe, Y. Huang, and D. P. Regula. Using a statistical natural language parser augmented with the umls specialist lexicon to assign snomed ct codes to anatomic sites and pathologic diagnoses in full text pathology reports. *AMIA Annu Symp Proc*, 2009:386–90, 2009.

22. M. Musen, Y. Shahar, and E. Shortliffe. Clinical Decision-Support systems. In K. Hannah, M. Ball, E. Shortliffe, and J. Cimino, editors, *Biomedical Informatics*, Health Informatics, chapter 20, pages 698–736. Springer New York, New York, NY, 2006.
23. A. N. Nguyen, M. J. Lawley, D. P. Hansen, R. V. Bowman, B. E. Clarke, E. E. Duhig, and S. Colquist. Symbolic rule-based classification of lung cancer stages from free-text pathology reports. *Journal of the American Medical Informatics Association : JAMIA*, 17(4):440–445, 2010.
24. J.-Y. Nie. Query Expansion and Query Translation as Logical Inference. *J. Am. Soc. Inf. Sci. Technol.*, 54(4):335–346, 2003.
25. S. Preece. *A Spreading Activation Network Model for Information Retrieval*. PhD thesis, University Illinois, Urbana, IL, USA., 1981.
26. Y. Qiu and H. Frei. Concept-based query expansion. In *Proceedings of SIGIR-93*, pages 160–169, Pittsburgh, US, 1993.
27. A. L. Rector, J. E. Rogers, and P. A. Pole. The GALEN high level ontology. pages 174–178. IOS Press, Jan. 1996.
28. C. Rocha, D. Schwabe, and M. de Aragão. A Hybrid Approach for Searching in the Semantic Web. In *WWW*, pages 374–383, 2004.
29. K. Schumacher, M. Sintek, and L. Sauermann. Combining Metadata and Document Search with Spreading Activation for Semantic Desktop Search. In S. Bechhofer, M. Hauswirth, J. Hoffmann, and M. Koubarakis, editors, *Proc. of ESWC*, pages 569–583. Springer, June 2008.
30. J. Suchal. On finding power method in spreading activation search. In V. Geffert, J. Karhumäki, A. Bertoni, B. Preneel, P. Návrat, and M. Bieliková, editors, *SOFSEM (2)*, pages 124–130. Safarik University, Slovakia, 2008.
31. P. Todorova, A. Kiryakov, D. Ognyanoff, I. Peikov, R. Velkov, and Z. Tashev. D2.4.1 Spreading Activation Components (v1). Technical report, LarKC FP7 project–215535, 2009.
32. A. Troussov, M. Sogrin, J. Judge, and D. Botvich. Mining Socio-Semantic Networks Using Spreading Activation Technique. 2008.
33. M. M. Van Berkum. Snomed ct encoded cancer protocols. *AMIA Annu Symp Proc*, page 1039, 2003.