METRIZED SMALL WORLD PROPERTIES BASED DATA STRUCTURE

V. Krylov
MERA Labs
Nizhny Novgorod, 603163, Russia
vkrylov@meralabs.com

A. Logvinov
MERA Labs
Nizhny Novgorod, 603163, Russia
alogvinov@meralabs.com

A. Ponomarenko
MERA Labs
Nizhny Novgorod, 603163, Russia
aponom@meralabs.com

D. Ponomarev
MERA Group
Nizhny Novgorod, 603163, Russia
dmp@meranetworks.com

Abstract

We introduce the information retrieval oriented data structure to build very large, scalable, loosely structured and unstructured distributed data storage.

The main idea is to represent data as a set of structured storage units on which a semi-metric can be defined which characterizes the relative relevance of each unit. Then a complex graph can be constructed whose vertices are the storage units and the edges are selected in such a way that the graph has the small world properties and is in accordance with the introduced metric (Metrized Small World Feature). Addition and removal of the data items causes the graph to evolve, while the retrieval of information is based on generating a new vertex, connecting it to the graph and setting up a search process of the data vertices metrically close to the request vertex. Due to the special properties of the constructed graph, the search is accomplished on average in the number of steps logarithmic of the storage size. We built a prototype of such a storage where the data items are represented by XML documents and the graph is expressed by means of XLink. The analysis of the graph properties we performed confirmed the possibility of building efficient XML data storages which contain hundreds of petabytes of data.

1 INTRODUCTION

The problem of information retrieval in huge distributed data storages is often complicated by the high data addition speed which lead to lagging of the indexing processes and consequently to low relevance of the search results. The direct search in the storage turns out to be more effective but extremely slow because the little structured iteration is unavoidable. Thus it is of interest to organize the data storage in such a way that the relevant documents form the linked clusters which are reachable in a small number of iteration steps.

In this paper we propose an approach where each document receives a list of links to other documents in such a way that every pair of documents is connected by a sequence of links with the average length determined by a slowly growing function of the document number. For a pair of documents close to each other according to the selected relevance criterion the sequence of links between them is even shorter. The link generation algorithm is based on viewing the set of documents as a complex graph with the small world properties [1,5,6], which is in accordance with the semi-metric defined on that set of documents which characterizes the proximity of documents according to the selected relevance criterion. We called the graphs with such properties the Metrized Small World Graphs (MSW). This approach allows to establish a logarithmic dependency of the average relevant document search length on the number of documents and to use diverse relevance criteria.

Furthermore storing the documents in such a structure allows to serve the fully decentralized dynamical storages (cloud storages) where both the sources of continuously appended documents and the sources of queries can be totally independent and geographically distributed. The implementation mechanism of the proposed structure is based on building a special platform over the file system which supports addition of the new documents, removal of the documents, adding the search query templates and search of the relevant documents. The prototype of the XML document storage platform was developed and subjected to analysis. The analysis of the graph properties confirmed the possibility of building efficient XML data storages whose sizes are hundreds of petabytes.

2 METRIZED SMALL WORLD STRUCTURE

In the proposed data structure the data is partitioned into integral storage units which we will call Information Objects (IO). Let's assume that the query processing can be decomposed to search and retrieval of one or more IOs followed by the logical or mathematical operations on their content. In the present paper we consider only the search and retrieval problem of the information objects in the storage which conform to a given relevance criterion. Let's define the level k $(k = 2, \dots l)$ information object as a structure of $k-1$ level objects. Let a semi-metric $\rho[1](a_i, a_j)$ be defined on a set A of level $k = 1$ information objects $\{a_i, a_j\}$ (later called atomic) which indicates the proximity of one object to another. For identical objects this semi-metric has zero value. The relation $\rho[1](a, x) \leq \alpha$ for any fixed α determines the set of atomic objects x which are in α neighborhood of IO a. The value inversely proportional to a can be interpreted as a relevance indicator of information objects to a given object a. On the atomic object level all queries are reduced to the presentation of a certain fixed object for the purpose of searching all IOs in the storage which have the given relevancy value to the presented object.

For information objects of the next level $k = 2$ (later denoted as $IO[2]$) a semi-metric $\rho[2]$ can be also introduced which is induced by the atomic level semi-metric taking into account the properties of the structure which forms the next level IOs. Thus the set of structured information objects turns out to be a semi-metric space. Taking similar steps one can construct semi-metric spaces of any level. Here we will limit our discussion with $k = 2$. We'll say that if the second level IOs have the form $[S; a_i, a_j, a_k, .. a_l]$ where S is a structure and the set $a_i, a_j, a_k, .. a_l$ consists of atomic objects then the second level patterns will have the form $[\bar{S};\ a_i, x, a_k, .. x]$. Here $\bar{S} \subseteq S$ is an arbitrary substructure of the IO structure and x are indefinite atomic objects. It is evident that the set of patterns can be imbedded into the semi-metric space of information objects if every pattern is defined as a subset of IOs in which all x act as indices and can have the value of any permitted atomic object. For any fixed pattern the minimal distance to any information object can be defined by iterating through all permitted x values and calculating the value of $\rho[2]$ semi-metric between the obtained information objects and the given IO. Let this minimal distance be the distance d between the pattern and the selected IO. Calculating distances to all IOs in the storage it is possible to find a subset having the minimal distance to the given pattern. Let this subset of IOs be the maximal relevant subset relative to the given pattern. If we extend the maximal relevant subset by appending it with the elements $IO[2]$ for which $\rho[2] - d \leq \propto$ we will obtain the subset of IOs with the relevance level value inversely proportional to $d + \propto$.

The practical application of the object space metrization approach described above consists in the possibility of mapping the information search and retrieval queries to the sets of patterns of the corresponding level.

Indeed if a query can be formulated as "find all objects with a subset of attributes defined with a certain precision while other attributes are irrelevant" then the query has an unambiguous mapping to the pattern and the search of the corresponding information objects is reduced to finding the maximal relevant subset of IOs for that pattern.

The choice of the structure for the second level information objects is usually dictated by the semantic considerations. In the majority of cases tree graphs are used as such structures. In our interpretation the vertices of such a graph are associated with atomic objects and the pseudometric in $IO[2]$ space can be defined in general by a tensor or its matrix. The elements of this matrix are determined by the choice of the root vertex of the object tree and are called perspective metrics.

In certain cases a decrease of significance of distances between the values of atomic objects as they recede from the root vertex should be taken into account. In such cases there should be used special weight functions determining the impact of the differences between atomic objects on different structure levels. The formula for the main metric can thus be defined as

$$\rho[2]_m = \min_{T_1 T_2} \sum_i \rho[1]\,(a_{i1}, a_{i2}) w(a_{i1}) w(a_{i2})$$

The second level information objects can also be combined into structures, thus defining the set of third level objects. In the present paper we suggest combining of all second level objects into a single structure represented by a complex graph possessing the special properties:

1. Decentralization: every structure element must be connected with other elements using pointers (links) in some uniform way.
2. Strong connectivity: to ensure search in a small number of operations compared to the number of elements a relatively short path must exist between every two arbitrary elements of the structure.
3. Locality: every element must store a small number of links (O(log(n)) or O(1)).
4. Metric clustering: pairs of elements having low metric value belong to the common clusters.

Tree-like structures, e.g. binary or B-trees, conform to the second and the third criteria but fail to conform to the first criterion of decentralization. In tree-like structures the search must begin from the root element, which prevents the creation of a decentralized structure.

A structure with the star topology can conform to the first and the second criteria but it contains a central element which stores the links to all other elements thereby failing the third criterion.

In the literature [1,2,3] the structures are described which conform to the first three criteria. These structures are small world graphs. These graphs usually satisfy the following criteria:

1. Power law distribution of vertex degrees.
2. The average shortest path length between two arbitrary vertices is proportional to the logarithm of the number of vertices.
3. The clustering coefficient remains the same while the number of vertices grows.

The article [4] describes the evolution algorithms of such graphs based on the probabilistic method of generation of new vertices and edges. To solve our task the fourth criterion must be satisfied which is not found in literature. We developed and proposed the dynamic method for construction of the $IO[3]$ structure compliant with the properties 1-4 which we called the metrized small world graph. In this graph the semi-metric structure of the set $IO[2]$ is coordinated with the topological graph structure, i.e. with the set of its edges.

3 CORE ALGORITHM

Let an element $v_i \in IO[2]$ be generated in every random moment t_i so that the elements form the set of the information objects $\{v_i\}$ in the order of their appearance. Assume that the graph constructed on the vertices $\{v_0, v_1, \dots v_{i-1}\}$ denoted as $G_{i-1} \in IO[3]$ is an MSW graph. The problem is to add v_i to the existing graph in a way that the resulting graph $G_i \in IO[3]$ is also an MSW graph and $G_i \,/\, G_{i-1} \cap G_i \not\subset G_{i-1}$, i.e. the addition of a new vertex to the graph does not require adding or removing edges incident to vertices other than the newly added one.

First we consider the base algorithm which does not regard the metric properties. The idea is to select m elements from the structure proportional to their vertex degrees when a new element is being added. If we follow the links between elements randomly choosing the next one we will visit the higher degree vertices with higher probability.

We propose the following algorithm with the parameters n and m, where n is the number of algorithm steps per single addition (determines MSW characteristics preservation), m is the number of links established by the element being added , $n \geq m$.

Let's assume that the structure already contains $i-1$ elements and we want to add the i-th element. Then the algorithm is as follows:

1. Arbitrarily select an element v_k where $1 \leq k \leq i-1$.
2. Let VisitedList be the set of visited elements.
3. Let CandidateList be the set of candidate elements for link establishment.
4. Assume that CandidateLists initially contains only v_k.
5. For $j < -1$ to n do
    a. Select an arbitrary element p from CandidateList not contained in VisitedList. If no such element exists then break.
    b. Remove p from CandidateList.
    c. Add p to VisitedList.
    d. Add the set of p neighbor elements not contained in VisitedList to CandidateList.
6. Mutually connect the v_i element with m arbitrary elements from VisitedList.

Second we consider the basic algorithm which performs the addition of a new information object regarding the semi-metric $\rho[2]$ defined on the set of elements. In order to ensure the preservation of MSW properties of the structure on any stage it is necessary to establish links in a way that the element which are close by the semi-metric are separated by a small number of links, i.e. that they belong to a common cluster.

We modified the previous algorithm so that the newly added element establishes links with m closest elements. The modified algorithm:

Add-Metric(V,Vi ,Vk, n, m)

1. Arbitrarily select an element v_k where $1 \leq k \leq i-1$.
2. Let VisitedList be the set of visited elements.
3. Let CandidateList be the set of candidate elements for link establishment sorted by value of semi-metric to v_i in ascending order.
4. Assume that CandidateLists initially contains only v_k.
5. For $j < -1$ to n do
    a. Sort CandidateList by value of semi-metric to v_i in ascending order.
    b. Select the first element p from CandidateList not contained in VisitedList. If no such element exists then break.
    c. Add p to VisitedList.
    d. Add the set of p neighbor elements to CandidateList.
6. Mutually connect the v_i element with m arbitrary elements from VisitedList.

Thus we have considered the algorithm of construction of the third level structure with MSW properties. Next we will demonstrate the method of search and retrieval of information from such structure. Assuming that the execution of the original query can be reduced to the operations on the set of second level object extracted from the third level structure using one or more patterns, let's discuss the process of finding a second level IO using the specified pattern.

Step 1. The query pattern is interpreted as an $IO[2]$ element and added to the $IO[3]$ structure using the algorithm described above regarding the metric characteristics of the elements specified in the pattern and not regarding the unspecified atomic objects.

Step 2. All elements having the minimal distance to the connected pattern are searched for in the $IO[3]$ structure. The found elements $(S; a_1, a_2, \dots, a_l) \in IO[2]$ form the set of information objects with maximal relevance to the query pattern.

The approach described above is based on the direct data analysis to determine element relevance. But the approach can also be used for building a distributed index or for implementing a distributed hash table.

4 SIMULATION

In the implementation prototype the elements of the structure are represented by XML documents. Each document is accessible via HTTP by its unique URL and has a set of XLink links to other document which is also accessible by a unique URL trivially calculated using the URL of the document. This set of links contains all links emanating from the given document. For every link the reverse link exists in the list of links for the document which is the target of the link. When a new document is added to the structure a link to this document is dynamically added to the list of links of every existing document to which the new document must be linked. We provide the results obtained from execution of the add-metric algorithm using XML media item descriptions as test data. Structures containing 1000, 5000, 10000 and 20000 elements were assembled using this algorithm. The graphs of vertex degree and shortest path length distributions are shown in Figures 1-3. In Figure 3 you can see how the average shortest path length between vertices changes with increasing number of vertices in the structure.

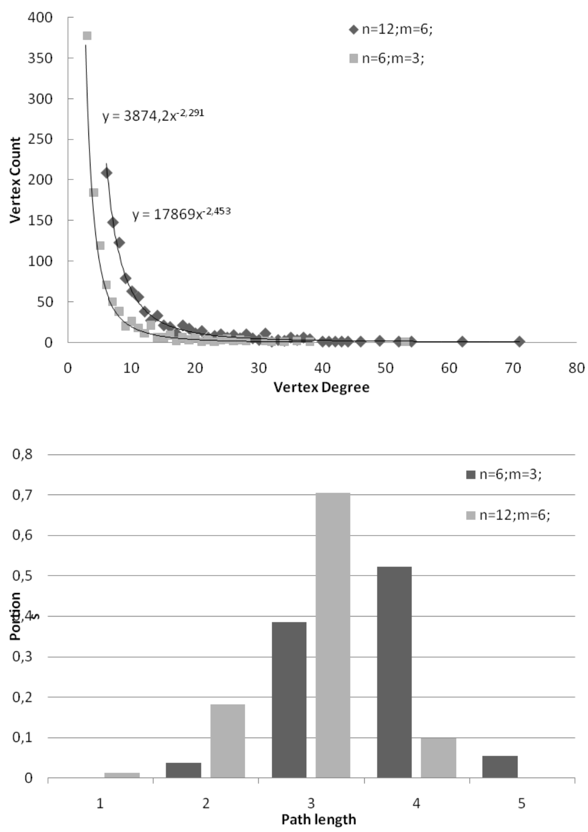

Figure 1: Vertex degree distribution and shortest path length distribution. Number of vertices in the structure is equal to 1000.

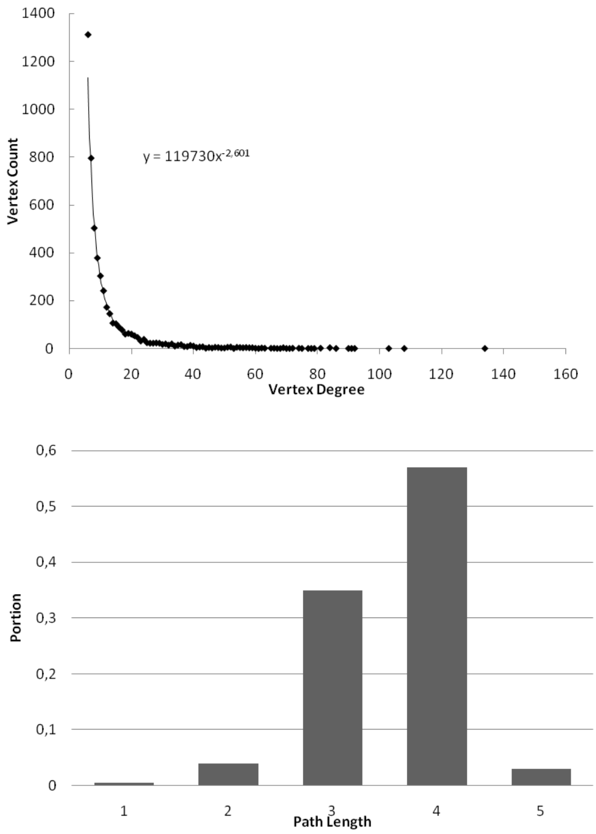

Figure 2: Vertexes degree distribution and shortest path length distribution. Number of vertices in the structure is equal to 5000.

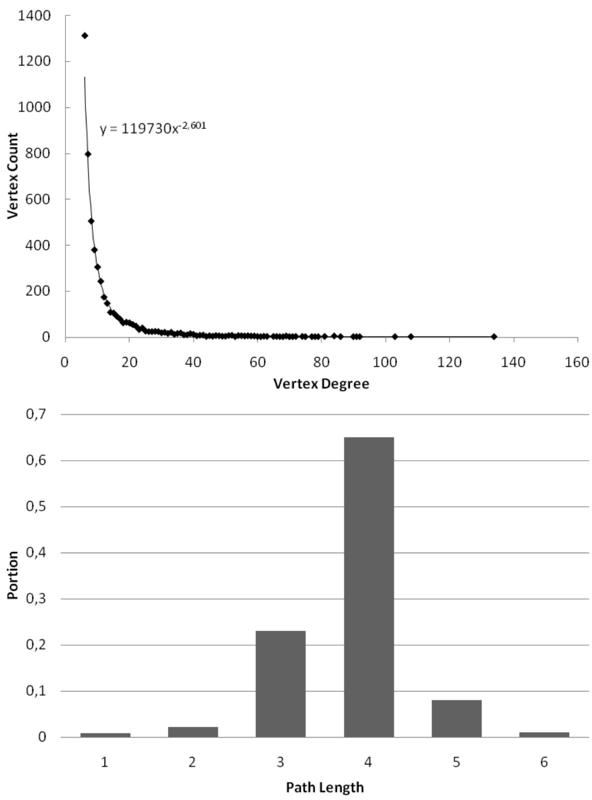

Figure 3: Vertex degree distribution and shortest path length distribution. Number of vertices in the structure is equal to 10000.

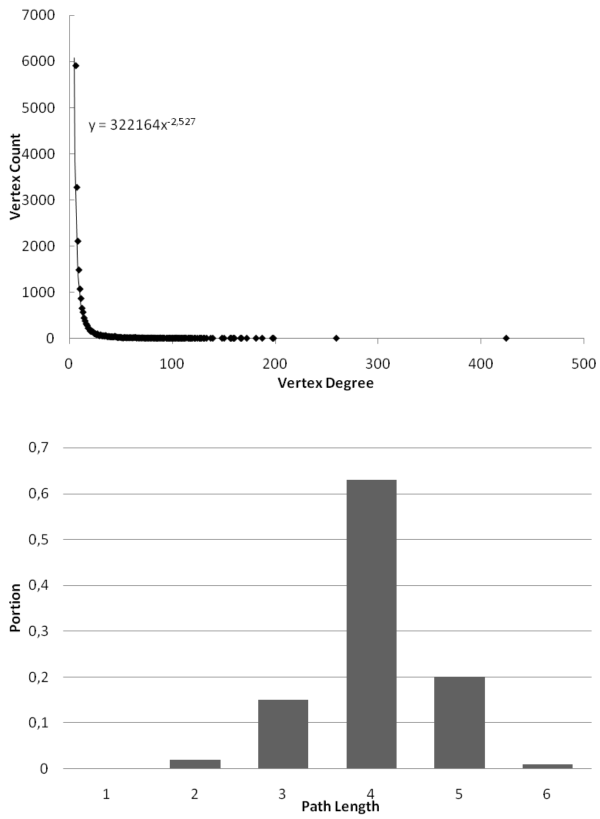

Figure 4: Vertex degree distribution and shortest path length distribution. Number of vertices in the structure is equal to 20000.

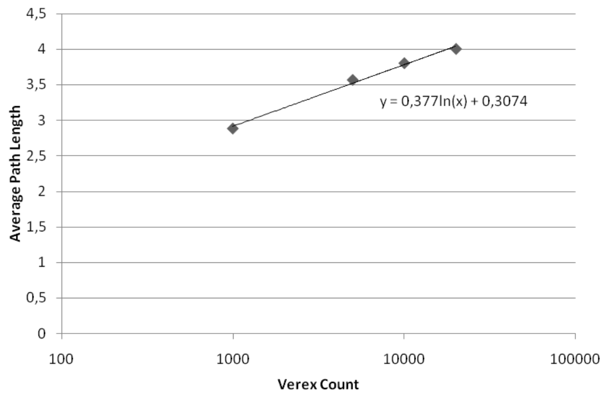

Figure 5: Shortest path length distribution averaged over all structures

5 REFERENCES

[1] Eli Ben-Naim, Hans Frauenfelder, Zoltan Toroczkai. “Complex Network”. Springer, 2004

[2] S. Milgram. “The Small World” *Psychol. Today* 2, 60 (1967); M. Kochen (ed.) (Ablex, Norwood, NJ, 1989)

[3] R. Albert and A.-L. Barabasi. Statistical mechanics of complex networks. *Rev. Mod. Phys.*, 74(1):47{97, January 2002.

[4] F. C. W. Aiello and L. Lu. Random evolution in massive graphs. In *Proceedings of the 42nd IEEE symposium on Foundations of Computer Science*, page 510. FOCS, IEEE Computer Society, 2001

[5] Watts, D. J., 1999, *Small Worlds* (Princeton, New Jersey: Princeton University Press).

[6] Amaral, L. A. N., Scala, A., Barthelemy, M. and Stanley, H. E., 2000, Classes of small-world networks, *Proc. Nat. Acad. Sci. U.S.A.*, 97, 11149;

[7] Newman, M. E. J., 2000, Models of the small world, *J. Stat. Phys.*, 101, 819;