# Mathematical Model and Algorithm for Evaluating Web Document Object Similarity

***B. Mo'minov**, Doctor of Technical Sciences, Professor Tashkent State University of Economics.*

***Sherzod Abdimannonovich Husanov**, Senior Lecturer, Department of Multimedia Technologies, Tashkent University of Information Technologies named after Muhammad al-Khwarizmi.*

**Abstract -** This article develops a mathematical model and algorithm for determining and evaluating the similarity of web document objects. The proposed approach allows for assessing the functionality of web page objects, their interrelationship, and their effectiveness based on user interactions. In addition, algorithmic solutions have been designed to analyze object similarity and interactions using a graph model.

**Keywords:** Web document, object similarity, graph model, clustering, algorithm, user interaction.

**Introduction** In modern web systems, analyzing user activity and determining the relationships among objects on a page are of great importance. Identifying object similarity increases the efficiency of web pages and ensures adaptability to user needs. Therefore, this article develops a mathematical model and algorithm for evaluating web document objects based on their similarity. The calculation of web document object similarity, as mentioned in the paragraph above, is divided into two parts, and models and algorithms for evaluating object similarity are proposed. Let the similarity between web document objects be represented in matrix form as $\mathbf{Q = (q_{(i,j)})}$, where each element $\mathbf{q_{(i,j)}}$ corresponds to the similarity between the *i*-th and *j*-th objects of a web page. If each attribute of an object is considered as a node of a directed graph, a graph model can be constructed to represent the similarity of web document objects.

Let the directed graph be denoted as **D = (V, E)**, where **V** is the set of vertices (objects), and **E** is an ordered set, $\mathbf{e_i} \in \mathbf{V}$, $i = 1, \ldots, n$, with $n$ being the number of objects. For each vertex, let $\mathbf{e_i^I} \in \mathbf{V}$ represent the incoming edges and $\mathbf{e_i^O} \in \mathbf{V}$ represent the outgoing edges, where $i = 1, \ldots, n$ and $n$ is the total number of objects. The weights of these edges, denoted as $\mathbf{w_{ij}}$, represent the degree of similarity from object *i* to object *j*. A directed graph can be constructed as follows.

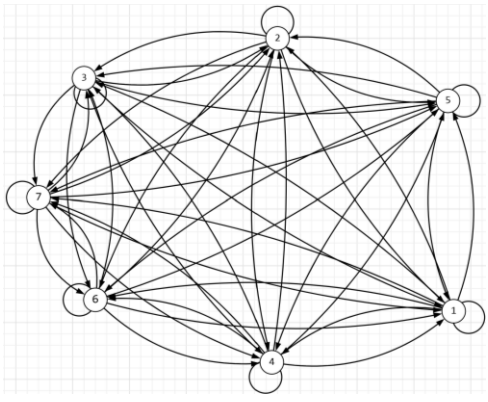

**Figure 1.** Graph model based on the similarity of web document objects. In the graph model based on the similarity of web document objects, the similarity weights are not shown. These weights can be expressed in matrix form as follows.

$$Q = \begin{pmatrix} q_{1,1} & q_{1,2} & q_{1,3} & \cdots & q_{1,n} \\ q_{2,1} & q_{2,2} & q_{2,2} & \cdots & q_{2,n} \\ q_{3,1} & q_{3,2} & q_{3,3} & \cdots & q_{3,n} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ q_{n,1} & q_{n,2} & q_{n,3} & \cdots & q_{n,n} \end{pmatrix} \quad (1)$$

Based on the graph model of web document objects, the following definitions are introduced for evaluating similarity:

**Definition 1.** When *i* = *j*, the similarity between the *i*-th and *j*-th objects of a web page is considered identical, i.e., $\mathbf{q_{(i,j)} = 0}$.

From this, it follows that for $\mathbf{q_{(i,j)}}$, using the metrics of the incoming edge $\mathbf{e_i^I}$ and the outgoing edge $\mathbf{e_j^O}$, we have $\mathbf{q_{(i,j)} = e_i^I - e_j^O = 0}$.

The similarity matrix of web document objects is not a fully symmetric matrix, meaning that its values are not identical. Based on the matrix values, for each $\mathbf{e_i^I}$, the corresponding row of **Q** (i.e., $(q_{(i,1)}, q_{(i,2)}, \ldots, q_{(i,n)})$) represents incoming edges, and for each $\mathbf{e_j^O}$, the corresponding column of **Q** (i.e., $(q_{(1,i)}, q_{(2,i)}, \ldots, q_{(n,i)})$) represents outgoing edges.

To evaluate the activity of an object on a web page, it is proposed to draw conclusions based on its similarity matrix using the following expression.

$$W_j = \sum_{i=1}^{n} q_{i,j} - \sum_{i=1}^{n} q_{j,i} \quad , j = 1..m \quad (2)$$

For the value obtained from expression (2), the following properties should be introduced:

If $W_j > 0$, it means that from object $q_i$ a transition to another object has occurred, and this object is considered highly important for the web page.

If $W_j = 0$, it means that the number of transitions from $q_i$ to other objects and the number of transitions from other objects to $q_i$ are equal, indicating that this object is good (satisfactory) for the web page.

If $W_j < 0$, it means that a transition has occurred from another object to $q_i$, and this object is considered highly unsatisfactory for the web page. Such an object should be improved or removed as soon as possible.

From the values of the matrix defined by expression (1), it is also proposed to evaluate the most active and most passive objects.

The most active object is evaluated as follows:

dmax(qi,qj)=max𝑓0j(min𝑓0i(qi,j))d_{max}(q_i, q_j) = \max_j (\min_i (q_{i,j}))dmax(qi,qj)=jmax(imin(qi,j))

The most passive object is evaluated as follows:

dmin(qi,qj)=min𝑓0j(min𝑓0i(qi,j))d_{min}(q_i, q_j) = \min_j (\min_i (q_{i,j}))dmin(qi,qj)=jmin(imin(qi,j))

Objects that fall between the most active and most passive ones are considered good objects, and are evaluated as follows:

dxi(qi,qj)=max⁡(dxi(qi,qj),Wi+dxi(qi,qj)) d_{x_i}(q_i, q_j) = \max(d_{x_i}(q_i, q_j), W_i + d_{x_i}(q_i, q_j))dxi(qi,qj)=max(dxi(qi,qj),Wi+dxi(qi,qj))

Evaluating the similarity of objects in a web document makes it possible to classify and rank them.

To determine the rating of web document objects, it is necessary to calculate the number of user interactions with that object — that is, the number of entries and exits. This can be achieved by tracking cursor movements or mouse actions on the web page and counting the increments for each object.

For this purpose, five main mouse events can be used:

MouseDown – movement into the object's active area along the *x*-coordinate;

MouseUp – movement into the object's active area along the *y*-coordinate;

MouseMove – movement leaving the object's active area;

Click (mouseClick) – single mouse click within the object's active area, indicating a selection action;

DbClick (mouseDbClick) – double mouse click within the object's active area, also indicating a selection action; if this event occurs, it signifies navigation to another object.

All of the mouse actions listed above exist and are implemented in all programming and scripting languages. These events can be handled for each object defined on any web page.

The following approach is proposed for managing these events:

For MouseDown and MouseUp events, assign each object a unique identifier *i (i = 0)*. The operation *i++* is performed once in real time during the event for that object.

For the MouseMove event, assign a unique identifier *j (j = 0)*. The operation *j++* is performed once in real time during the event for that object.

For the Click event, assign a unique identifier *c (c = 0)*. The operation *c++* is performed once in real time during the event for that object.

For the DbClick event, assign a unique identifier *dc (dc = 0)*. The operation *dc++* is performed once in real time during the event for that object.

These variables operate from the moment the web page is launched until statistical data collection is complete. For each object, these four events function continuously and are calculated as follows:

EI=i+c,EO=j+dcE_I = i + c, \quad E_O = j + dcEI=i+c,EO=j+dc

Here, E_I represents the incoming interactions for the object, and E_O represents the outgoing interactions. The characteristics of the events are divided into two categories accordingly.

By expressing incoming and outgoing interactions in a graph model, it is possible to construct an information model. In this case, the current object under consideration is represented as a vertex of the graph, while

the incoming and outgoing interactions are represented as directed edges.

Let the directed graph be expressed as G = (O, E), where O is the set of vertices, representing the objects, and E is the ordered set of directed edges corresponding to incoming and outgoing interactions ($e_i \in V$, i = 1, …, n, where *n* is the number of objects).

Let $e_i^I \in V$ represent incoming interaction edges, and $e_i^O \in V$ represent outgoing interaction edges, for $i = 1, \ldots, n$. The weights of these edges, denoted as $w_{ij}$, indicate the number of incoming or outgoing interactions from object *i* to object *j*.

A directed graph can be constructed as follows.

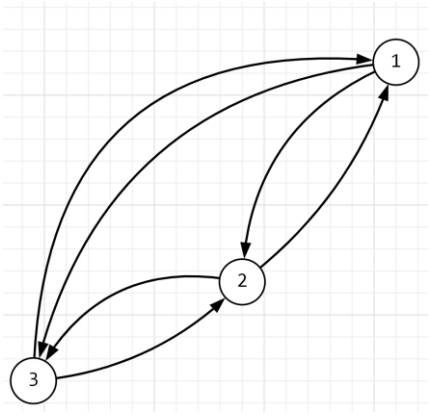


**Figure 2.** Example of a graph model based on web document object interactions.

In general, let a web document consist of *n* web pages, and let each page contain *m* objects.
The total number of objects is therefore **N = n × m.**

The incoming and outgoing interactions of the web document can be represented in the form of a **matrix M.**
For this purpose, the following conditions are defined:

- Each corresponding **row** of matrix **M** represents the **outgoing interactions.**
- Each corresponding **column** of matrix **M** represents the **incoming interactions.**
- In this case, the **diagonal elements** of matrix **M** are calculated using the expression:

$$E_D = \frac{E_I + E_O}{2}$$

The general form of matrix **M** is expressed as follows:

$$M = \begin{pmatrix} E_{D_{1,1}} & E_{O_{1,2}} & E_{O_{1,3}} & \cdots & E_{O_{1,n}} \\ E_{I_{2,1}} & E_{D_{2,2}} & E_{O_{2,3}} & \cdots & E_{O_{2,n}} \\ E_{I_{3,1}} & E_{I_{3,1}} & E_{D_{3,3}} & \cdots & E_{O_{3,n}} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ E_{I_{n,1}} & E_{I_{n,2}} & E_{I_{n,3}} & \cdots & E_{D_{n,n}} \end{pmatrix}$$

From matrix **M**, for each object with $\mathbf{E_i^{Ik}}$ (incoming interactions) and $\mathbf{E_i^{Ok}}$ (outgoing interactions), the following expression holds true:

$$E_I^k = \sum_{i=1, i\neq k}^{N} E_{I_{k,i}}, E_O^k = \sum_{i=1, i\neq k}^{N} E_{O_{k,i}} \; k = 1 \ldots N \qquad (2)$$

From expression (2), the following conclusions can be drawn:

If for each *k*, $\mathbf{E_i^{Ik} - E_i^{Ok} < 0}$, this object needs to be **updated or modified**.

If for each *k*, $\mathbf{E_i^{Ik} - E_i^{Ok} = 0}$, this object is considered **good or satisfactory**.

If for each *k*, $\mathbf{E_i^{Ik} - E_i^{Ok} > 0}$, this object is considered **very good**, and such objects should be **increased or replicated**.

Based on the above proposal, a modified version of the matrix **M**, denoted as **M***, is obtained for the web document.

$$M^* = \begin{cases} 1, & E_I^k - E_O^k > 0 \\ 0, & E_I^k - E_O^k = 0 \\ -1, & E_I^k - E_O^k < 0 \end{cases}$$

Based on the values of the **M*** vector, it becomes possible to visually identify the objects within the web document that require modification.

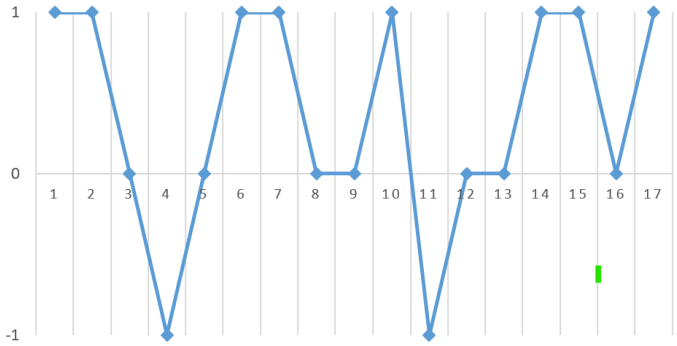


Figure 3. Graph of interaction-based values for a web document consisting of 17 objects.

Based on the incoming and outgoing interactions of web documents, modifications can be made to their objects. However, there may also be objects that complement the best-performing objects, meaning they belong to the same class.

To identify such objects—especially when the classes have not been predefined during web document analysis—the clustering method can be applied. Using clustering, it is possible to determine which objects belong to the same class within a web document.

For this purpose, the following metric can be used:

$$d_{x_i}(x_i, x_j) = \max\left(d_{x_i}(x_i), |x_i - x_j| d_{x_i}(x_j)\right)$$

Using the $\mathbf{d_{(xi)}}$ metric, the design of **cluster-based clustering** was implemented (see **Figure 4**).

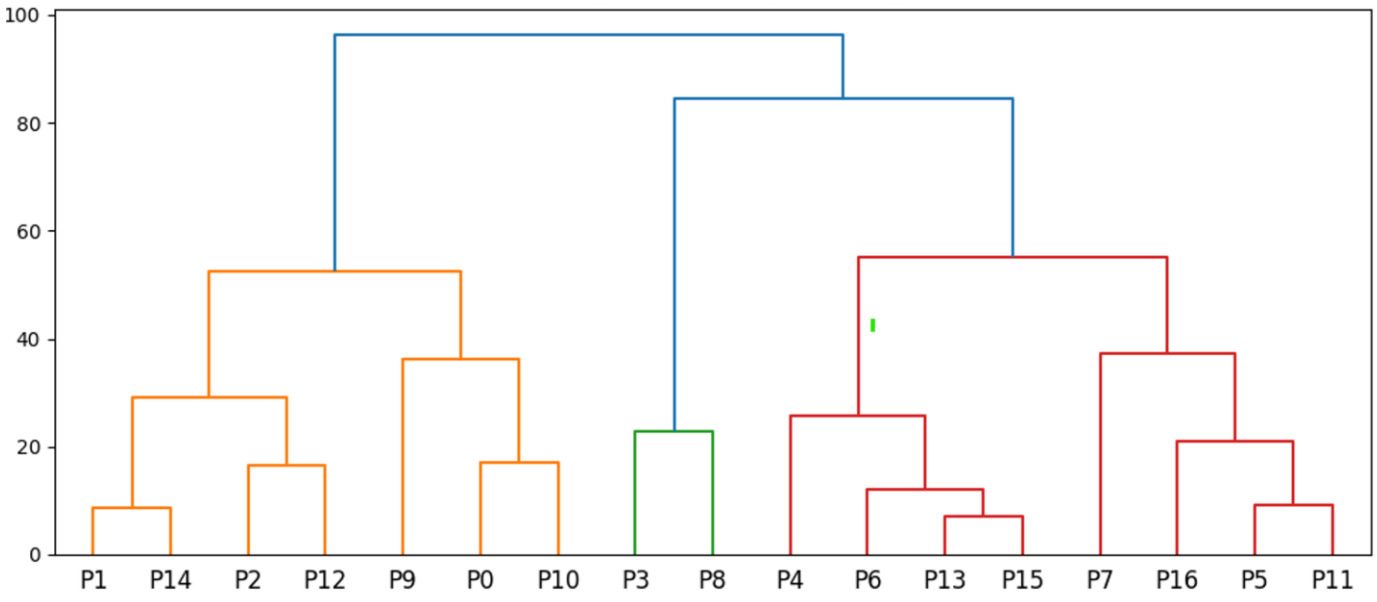


**Figure 4.** Example of clustering for a web document consisting of 17 objects.

Based on the mathematical models for evaluating web document object similarity described above, a **general evaluation algorithm** has been developed. It includes the following components:

**Similarity of objects**

**Number of interactions with objects**

**Similarity based on the number of interactions with objects**

Separate subprograms are created for each of these three mathematical models, and they return values in **matrix form**.

To bring these values into a unified format, the following **criteria** are introduced:

For the **similarity of objects**, an interval **[a, b]** is defined, and the following expression is used within this range.

$$O_1(x) == \begin{cases} 1, & x > b \\ 0, & x \in [a,b] \\ -1, & x < a \end{cases}$$

For the **number of interactions with objects**, the variable is defined as $x = E_I^k - E_O^k$ and the following expression is used for evaluation.

$$O_2(x) == \begin{cases} 1, & x > 0 \\ 0, & x = 0 \\ -1, & x < 0 \end{cases}$$

For the **similarity based on the number of interactions with objects**, a family of classes $\mathbf{C}^+$ and $\mathbf{C}^-$ is introduced, and the following expression is used for evaluation.

$$O_3(x) == \begin{cases} 1, & x \in C^+ \\ 0, & x \notin C^+ \; va \; x \notin C^- \\ -1, & x \in C^- \end{cases}$$

The general expression for determining the **similarity of objects** is defined as follows:

$$O = \frac{O_1(x) + O_2(y) + O_3(z)}{3}$$

Based on the proposed **object similarity evaluation model**, the algorithm is defined as follows:

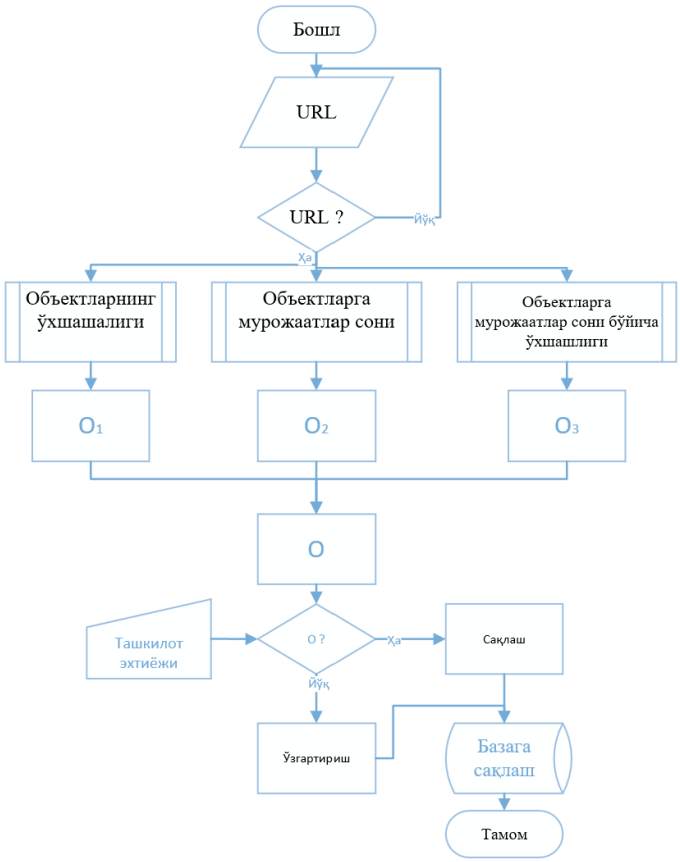


Figure 5. Algorithm for evaluating web document object similarity.

Based on this algorithm, it becomes possible to update web documents by modifying their objects and to address issues related to attracting users.

Currently, most organizations have developed their own computational

mechanisms based on well-known traditional methods.
The proposed evaluation model based on web document object similarity is primarily aimed at educational and legal websites.

In this evaluation process, there exist events that perform similar functions, and these must be processed within the document in the same manner.
This issue will be discussed in the following paragraph.

## References

[1]. Viyyameena M. K., Kavita K. A Survey on Similarity Measures in Text Mining. Machine Learning and Applications: An International Journal, Vol. 3, No. 1, March 2016.

[2]. Vu M., et al. Modelling Text Similarity: A Survey. 2023.

[3]. Cha S. H. Comprehensive Survey on Distance/Similarity Measures.

[4]. Battler D. A Short Survey of Document Structure Similarity Algorithms. 2004.

[5]. Mohammed S. M., Jaksi K., Zeebari S. R. M. A State-of-the-Art Survey on Semantic Similarity for Document Clustering Using GloVe and Density-Based Algorithms.

[6]. Ahmed T. Exploring Mathematical Models and Algorithms for Plagiarism Detection in Text Documents: A Proof of Concept.