A recommendation system has been built for a web resource's users that applies statistics about user activities to provide recommendations. The purpose of the system operation is to provide recommendations in the form of an orderly sequence of HTML pages of the resource suggested for the user. The ranking procedure uses statistical information about user transitions between web resource pages. The web resource model is represented in the form of a web graph; the user behavior model is shown as a graph of transitions between resource pages. The web graph is represented by an adjacency matrix; for the transition graph, a weighted matrix of probabilities of transitions between the vertices of the graph has been constructed. It was taken into consideration that user transitions between pages of a web resource may involve entering a URL in the address bar of a browser or by clicking on a link in the current page. The user's transition between vertices in a finite graph according to probabilities determined by the weight of the graph's edges is represented by a homogeneous Markov chain and is considered a process of random walk on the graph with the possibility of moving to a random vertex. Random Walk with Restarts was used to rank web resource pages for a particular user. Numerical analysis has been performed for an actual online store website. The initial data on user sessions are divided into training and test samples. According to the training sample, a weighted matrix of the probability of user transitions between web resource pages was constructed. To assess the quality of the built recommendation system, the accuracy, completeness, and Half-life Utility metrics were used. On the elements of the test sample, the accuracy value of 65–68 % was obtained, the optimal number of elements in the recommendation list was determined. The influence of model parameters on the quality of recommendation systems was investigated

Keywords: recommendation system, web graph, transition graph, Markov chain, random walk

UDC 004.912
DOI: 10.15587/1729-4061.2021.233501

DESIGN OF A RECOMMENDATION SYSTEM BASED ON THE TRANSITION GRAPH

Natalia Guk
Doctor of Physical and Mathematical Sciences, Professor, Head of Department*
Olga Verba
Senior Lecturer*
Vladyslav Yevlakov
Corresponding author
Postgraduate Student*
E-mail: yevlakov@gmail.com
*Department of Computer Technologies
Oles Honchar Dnipro National University
Haharina ave., 72,
Dnipro, Ukraine, 49010

Received date 23.04.2021
Accepted date 31.05.2021
Published date 29.06.2021

How to Cite: *Guk, N., Verba, O., Yevlakov, V. (2021). Design of a recommendation system based on the transition graph. Eastern-European Journal of Enterprise Technologies, 3 (4 (111)), 24–31. doi: https://doi.org/10.15587/1729-4061.2021.233501*

1. Introduction

One of the most popular search queries among Internet users is directly related to commercial activities, that is, the search and purchase of certain products and services [1]. E-commerce, today, is a strong area that is constantly expanding, improving, and covering different areas of the life of a modern person. To increase the efficiency of online stores, recommendation systems are most often introduced into their functionality [2]. Based on the accumulated data and algorithms for their processing, the recommendation system can provide the user with recommendations that take into consideration his preferences and interests at the current time regarding the choice of goods and services.

During the operation of a web resource (online store or information portal), a large amount of commercial information accumulates, which requires automatic processing and can be used to build a recommendation system.

From the point of view of practical implementation, the simplest is non-personalized recommendations that are built, for example, based on the rating of goods according to buyers' estimates, based on the frequency of sales of a product or group of goods. When building such recommendations, researchers face a number of issues, the main among which is the problem of cold start and the relevance of the formulated recommendation. For new products and new users, the necessary information for ranking is missing, so the recommendations may be unreliable. Different ways of smoothing data and calculating credibility intervals do not eliminate these problems.

Therefore, the most promising is to build systems of personalized recommendations that are based on accumulated transaction data and a built-in model of user behavior or a group of similar users.

When building a model of the recommendation system, it is important to distinguish data related to a particular user from noise. It is also necessary to take into consideration the peculiarities of behavior that determine the specific preferences of users to a particular product, and highlight similar users to provide group recommendations. To eliminate the problem of the previous lack of some user and product data, it is relevant to develop heuristic approaches by which the necessary initial data can be generated.

2. Literature review and problem statement

In the last few years, many technological advances have been made in the field of building and implementing recommendation systems for web resources. Several approaches are used. The main types of recommendation systems on the Internet are based on methods for evaluating the content of an Internet resource or collaborative filtration. These approaches were studied in work [3], which considered the methods of

constructing classifiers for content filtration and ways to calculate the coefficient of similarity of users or objects in collaborative filtration.

Content filtering involves images of units of content in a feature space that makes it possible to compare them to the interests of users who have previously saved them to a user's profile. The rating of the goods in the list of recommendations is determined by the degree of conformity of the goods to the interests of the user. The difficulties arising from the application of this approach are associated with the impossibility of a structured image of some features of the object, for example, its text description provided to the object of review, etc. In this case, a vector description of the sign in the space of words is used. In the same space, it is necessary to obtain a description of the user's interests, then it becomes possible to compare the vectors of objects and users in order to determine their proximity. To improve the vector description of objects, Tf-Idf metrics are widely used, which take into consideration the frequency of use of words in the description [4]. Thus, matching words that are used less frequently has greater significance to make recommendations than coincidences on signs that are used more frequently.

The second approach in the construction of recommendation systems relies on collaborative filtration. Paper [5] devised a methodology for selecting the current video content, taking into consideration the needs of the user. The recommendation system is built in the form of a web application, the algorithm for providing recommendations uses the method of collaborative filtration. Input data for the algorithm operation are formed by users by providing assessments of the video materials being viewed. The quality assessment of the developed system was checked and evaluated on YouTube video materials.

Work [6] compares the quality of recommendation systems that use content and collaborative filtering using well-known movie rating data sets MovieLens and Netflix. It has been shown that taking into consideration the content makes it possible to build a recommendation system that is aimed at the needs of a certain user, and collaborative filtration systems have a better predictive ability.

Algorithms based on collaborative filtering collect information about users' preferences and find the most similar ones among them. Therefore, they are able to supplement the missing user data with available data about similar neighbors, so that they can eliminate the problem of cold start. The similarity of users means the convergence of their interests and preferences, and to determine the similarity, various metrics are used, such as the Pearson correlation, the cosine distance between vectors, the distance of Jacquard, the distance of Hemming. A significant drawback of such algorithms is their quadratic complexity, for the elimination of which simplifications are used associated with modifications of the algorithm for calculating paired distances between objects. Assumptions that simplify the user behavior model are also used. For example, a hypothesis is formulated about the immutability of the user's interests over time and the hypothesis about the complete coincidence of preferences for users whom the algorithm recognized as close to each other. In addition, such algorithms require normalization of the results of the evaluation of goods provided by users.

More difficult to implement in practice is the approach based on the formalization of expert knowledge about the subject area – ontology. The use of this approach makes it possible to take into consideration the complex dependences between objects and make the recommendations more accurate. An overview of recommendation systems and their ability to take into consideration the context in the formation of individual recommendations is given in [7]. The authors indicate that the creation of semantically developed images with formal axiomatization is a complex and lengthy process. In addition, during the existence of such systems, knowledge must be updated and maintained up to date.

In addition, when building recommendation systems, a hybrid approach is developing that combines the ideas of previous methods of providing recommendations [2]. With the use of hybrid methods, a balanced combination of results is built, for example, recommendations for collaborative and content filtration, or a mixture of recommendations from several sources is provided. Sequential processing of recommendations is also possible when the results selected by content filtering are sent to the collaborative filtration algorithm [8]. However, the use of a hybrid approach requires justification of the technique of combining recommendations and determining the weights of each component of the integrated recommendation.

The above review reveals that existing approaches to the construction of personalized recommendation systems do not solve the problem of cold start, significantly simplify the model of user behavior during a search, have high computational complexity, and are poorly scalable. The solution to this problem may be the development of new algorithms specific to this class of data processing tasks.

3. The aim and objectives of the study

The purpose of this study is to develop a recommendation system for a web resource, which is based on semantic links of web resource pages and the results of processing statistical data on user actions. This could eliminate the cold start problem for new users and build recommendations online.

To accomplish the aim, the following tasks have been set:

– to build a web resource model in the form of a web graph and a model of user behavior in the form of a navigation graph;

– to apply a Random Walk with Restarts method to rank HTML pages of the web resource recommended for viewing, build an algorithm and program implementation of the proposed technique;

– to select appropriate metrics to assess the quality of the built recommendation system;

– to apply the proposed approach to an actual web resource and analyze the results of the computational experiment.

4. The study materials and methods

The task of building a recommendation system is formulated as follows: let U be a set of users, V – a set of HTML pages of a web resource. It is required to construct the mapping of F [2]:

$$F: \ U \times V \to r,$$

where r is the rating of page $v \in V$ for user $u \in U$, $r \in R$.

The mapping of F measures the feasibility of a $v \in V$ page recommendation for $u \in U$ user. The task of the recommendation system is to select for each user $u \in U$ such a page $v' \in V$, which

$$\forall u \in U, \ v'_u = \arg\max_{v \in V} F(u, v).$$

We propose an approach to building a recommendation system based on the ranking of web resource pages. To perform the ranking procedure, it is proposed to analyze the structure of the web resource and take into consideration information about the behavior and actions of users when visiting the resource, which is accumulated by the system of collecting statistics of visits.

To build a recommendation system, the theory of Markov processes was used. It is believed that the process of user transitions between HTML pages of a web resource corresponds to a state change in the Markov homogeneous chain. The web resource model is depicted as a graph. Statistical information about user transitions accumulates and is displayed as a weighted navigation graph. The process of user transition between web resource pages is considered as a process of random walk on the graph. A Random Walk with Restarts method is used to rank graph vertices.

To assess the quality of the formulated recommendations, metrics of accuracy, completeness, and Half-life Utility (HLU) were used. The graph of an actual web resource is constructed using the crawler program. JavaScript scripts are embedded to aggregate user transition data into the HTML page structure. The analysis of constructed graphs is carried out using metric characteristics: diameter, average vertex degree, average path length between two vertices, graph density. Existing user conversion observation data between web resource pages is divided into a training and test sample. The quality analysis of the built recommendation system is carried out using the average values of accuracy, completeness, and HLU metrics for the list of users from the test sample.

5. Results of the development of the system of providing recommendations to users of the web resource

5. 1. Construction of mathematical models of the web resource and user behavior when viewing web resource pages

Discrete mathematical models are widely used to describe the structure of the site. With the help of such models, semantic clustering of web resource pages is carried out, communities of web space users are separated according to certain characteristics, metric characteristics of graphs are studied to determine the types of web resource pages.

To represent the structure of a web resource, it is proposed to use the hypertext model in the form of a graph given in [9, 10].

A web resource is represented as a $G(V,E)$ graph, in which the vertices are HTML pages and the edges are the links between them; V is the set of vertices; E is the set of edges; $E=\{e(v_i,v_j)|link(v_i,v_j)\}$; $link(v_i,v_j)$ is the function of switching from the HTML page v_i to the HTML-page v_j.

The hypertext model is supplemented by information about user actions on the site. The user's transition between hypertext pages can be aimed at finding a landing page that matches the query or to visit pages of a specific topic. To simulate user behavior when searching for information on the site, we shall use the user transition graph between the pages of the web resource, based on the $G(V,E)$ web graph. The transition graph is depicted by a weighted graph $G(V,E,W)$, where W is the set of weights of the edges.

The weight of the edge $e(v_i,v_j)$ is determined by the number of user transitions between pages $v_i, v_j \in V$. In the case when there is no transition between the pages of the site $v_i, v_j \in V$, the weight of the corresponding edge is 0.

Then each user's route between the hypertext pages can be represented as an ordered sequence:

$$P=\left(v_1,v_2,...,v_l\right),\ v_i \in V,$$

where l is the number of HTML pages visited by the user during one session.

Since user transitions between pages in different sessions do not depend on each other, we can consider the movement of users between pages of the site as a random process. When a web resource is visited, the user views the vertices as independent equally distributed random variables. Assume that at any given time the conditional distribution of future states of a process with specified current and past states depends only on the current state and does not depend on past states. That is, the user's transition to a specific page of the site depends only on the current page on which the user is currently located [11].

Given these features, the process of user transition between pages is a Markov process with discrete time and discrete states.

In this case, each route P_k of the user within one session can be depicted as a subgraph:

$$G_{P_k}=\left\{V_{P_k},E_{P_k},W_{P_k}\right\},$$

where $V_{Pk} \subset V$ is a set of vertices visited by the user; $E_{Pk} \subset E$ is a set of edges through which the transitions between vertices were made; $W_{Pk} \subset W$ is a set of edge weights in the subgraph.

The weight of each edge $e(v_i,v_j)$ in the subgraph corresponds to the number of transitions between the HTML pages v_i, v_j within the P_k route. The resulting user subgraphs are combined to form a directed weighted transition graph:

$$G_P=\left\{V_P,E_P,W_P\right\},$$

in which the weight of the edge is calculated as the sum of the weights of the edges between pairs of vertices v_i, v_j from each subgraph.

The adjacency matrix of the transition graph takes the following form:

$$C_P=\begin{bmatrix} c_{11} & \cdots & c_{1n} \\ & \cdots & \\ c_{n1} & \cdots & c_{nn} \end{bmatrix},\ c_{jk} \in N,$$

where element $c_{ij}=\sum_{k=1}^{K} w\left(e\left(v_i,v_j\right)\right),\ w \in W_{P_k};\ w(e(v_i,v_j))$ is the weight of the edge between the vertices v_i, v_j in each P_k subgraph.

Using the adjacency matrix of the navigation graph, a matrix of probabilities of transitions between the pages of the M_{ij} web resource is built. Considering that the probability of navigation between pages is directly proportional to the number of user transitions between these pages, we obtain:

$$m_{ij}=\begin{cases} \dfrac{c_{ij}}{\sum_{j=1}^{n} c_{ij}}, & \forall i,j \mid\ e\left(v_i,v_j\right) \in E; \\ 0, & \forall i,j \mid\ e\left(v_i,v_j\right) \notin E, \end{cases}$$

where n is the number of vertices of the graph. The matrix elements are $m_{ij} \geq 0$ and normalized for columns.

When moving through a web resource, the user can navigate in two ways – by entering a URL in the browser address bar (INPUT) or by navigating (CLICK) to the page by a link from the current page. Therefore, our proposed model adequately describes the process of moving the user through the web resource. In terms of graph theory, the user can jump from the starting vertex v^* or to an adjacent vertex, or to an arbitrary vertex of the graph.

5. 2. Using the Random Walk with Restarts method to rank web resource pages. Algorithm and software implementation

The user's transition between vertices in a finite graph, taking into consideration the probabilities of transitions, is equivalent to changing states in the Markov homogeneous chain. Since the web resource model is represented as a graph, the user's movement between HTML pages should be considered as a random walk in the graph with the ability to jump to an arbitrary vertex of the graph [11].

Using data on semantic links between pages and taking into consideration the number of user transitions, it is necessary to rank the vertices of the graph associated with the arbitrary predefined vertex v^*, where the user is located. To rank the vertices of the graph, taking into consideration the probability of getting to them from any vertex v^*, we shall use the Random Walk with Restarts (RWR) method [12].

Unlike classical algorithms of random walk, the graph in the RWR algorithm takes into consideration the probability that the process can go not only to the adjacent one but also to any vertex of the graph, including the starting one. This corresponds to the user behavior model on web resource pages.

In addition, given the data of the probability matrix of transitions, we assume that transitions to vertices are carried out by chance but not exactly likely.

Random walk on the $G(V,E,W)$ graph begins at some vertex v^*, after t steps, it goes to the vertex v_i, from this vertex, at $(t+1)$ wandering step, can move to any vertex v_j. In the random walk step, the user from the starting vertex v^*, at probability $(1-\gamma)$, goes to the arbitrary vertex of the graph v_i, or, at a probability of γ, to one of the adjacent v^* vertices.

The set of vertices and the probability of transition between them form the Markov chain, and the probability of getting from the initial (starting) vertex v^* to any vertex v_i can be determined by solving the following equation:

$$P^{(t+1)} = \gamma \cdot M \cdot P^{(t)} + (1-\gamma)q, \quad (1)$$

where $P^{(t)}$ is a vector column in which the component $P_i^{(t)}$ equals the probability of getting to the vertex v_i from the starting vertex v^*; M is the probability matrix of transition, m_{ij} is the probability of transition from vertex v_j to vertex v_i; q is the initial (start) vector-column, in which $q_{v^*}=1$, and the other components are equal to 0.

If random walk at the $(t+1)$ iterative process step reaches the state when $P^{(t+1)}=P^{(t)}$, then the resulting distribution $P^{(t)}$ is stationary. Vector $P^{(t)}$) consists of probability values to get to each of the vertices of the graph, starting from the starting vertex v^*.

Thus, the $P_i^{(t)}$ value determines the degree of binding of the vertices v^* and v_i, and the resulting vector $P^{(t)}$ can be interpreted as a measure of the proximity of each of the vertices of the graph in relation to the starting vertex v^*. By sorting the components of the resulting vector $P^{(t)}$ in descending order of values and selecting the first L vertices at the top of the list, one can create a list of graph vertices that are most relevant and recommended to the user to visit.

The iterative process (1) is carried out until the convergence of the sequence of $\|P^{(t+1)}-P^{(t)}\|^2 \le \varepsilon$ estimates in the norm of space L_2 or upon reaching the maximum step number of the iteration process.

Since the directed graph of the web resource contains a large number of vertices and may be poorly connected, a significant number of iterations may be required to achieve the convergence of the iteration process. The time it takes to obtain a solution depends linearly on the iteration number, and, due to the need to perform the operation of multiplying the matrix by a vector, the quadratic depends on the power of the set V. Therefore, the modification of the RWR method [13] is used in our work; equation (1) is depicted as follows:

$$(I-\gamma M)\cdot P^{(t)} = (1-\gamma)q, \quad (2)$$

a solution to it takes the following form:

$$P^{(t)} = (1-\gamma)(I-\gamma M)^{-1} q, \quad (3)$$

where I is the unit matrix.

The inverse of the matrix is found using the Gauss-Jordan method, which has a complexity of $O(n^3)$. This determines the overall complexity of the applied algorithm since the operation of multiplying a matrix by a vector has a lower complexity of $O(n^2)$.

The weighted matrix of transition probability is based on statistical information about users visiting web resource pages at the model construction stage, so it is the same for all users.

The iterative process (3) can be organized from any arbitrary vertex of the graph at which the user is located. That is, the search for graph vertices recommended for the user to view can be done for an arbitrary user by assigning vectors $P^{(0)}$ and q for him.

The estimation of the probability of hitting an arbitrary vertex of the graph from the starting one, obtained by the RWR method, has several advantages over methods that use pairwise metrics to assess the proximity of the graph vertices. To assess the relevance of crawl vertices for a particular user, our proposed approach takes into consideration the structure of the graph, the functional connections between the vertices of the graph, as well as data on the behavior of other users of the web resource.

The proposed approach to building a recommendation system involves the following sequence of actions:

– build a web resource graph and its adjacency matrix;

– construct a graph of user transitions and its weighted adjacency matrix;

– construct a matrix of the probability of user transitions;

– set algorithm parameters and initial data about a particular user;

– organize calculations according to iterative formula (3) until the convergence is achieved;

– sort the components of the resulting vector $P^{(t)}$ in descending order and create a list of pages recommended for visiting.

As a result, we obtain a vector, the components of which correspond to the probabilities of user transitions to certain pages of the web resource. According to the received vector for a certain user, a recommendation is built in the form of a sequence of pages of the site for viewing.

The proposed approach is implemented by the following algorithm:

Initializing: Construct a web resource graph, build a navigation graph, and appropriate adjacency matrices. According to the adjacency matrix of the transition graph, construct a normalized weighted probability matrix M. Set $t=0$; set the value for γ, vector $P^{(0)}$, starting vertex v^*. Form vector q; set L – the number of pages recommended for the user for viewing:

1. Form a matrix $(I-\gamma M)$, build an inverse matrix for it.

2. Calculate $P^{(t)}=(1-\gamma)(I-\gamma M)^{-1}q$.

3. Check meeting the condition $\|P^{(t+1)}-P^{(t)}\|^2\le\varepsilon$; if the condition is satisfied, then proceed to step 4; otherwise, $t=t+1$, go to step 2.

4. The resulting vector $P^{(t)}$ contains the probability of getting into each of the existing vertices of the graph, starting from the starting vertex v^*.

5. Sort the components of the vector $P^{(t)}$ in descending order, select the first L components to form a list of recommendations to the user.

The g parameter value was set to 0.15. The selection was made in accordance with the PageRank model [11] taking into consideration observations of user movements over HTML pages of the web resource. It is believed that the user, starting from the starting vertex v^*, first makes an average of 6 CLICK-type conversions by links, and then makes an INPUT type transition to get to the new start page.

For the practical implementation of the proposed approach, a web graph of the site was built using the crawler program [15]. In the resulting $G(V,E)$ graph, all vertices are unique, that is, one HTML web resource page corresponds to one vertex in the graph. The web graph of the site is represented by an adjacency matrix.

To accumulate information about user actions over a certain time, the web resource was connected to the visit statistics collection system. The web analytics service recorded user transitions between HTML pages during the search session, as well as conversion types. JavaScript scripts were embedded to aggregate user transition data into the HTML structure of web pages. User actions are written to web server log files in chronological order in the form of data sets:

<URL, DATA, TIME, TYPE>, (4)

where URL – address of the viewed web page, DATA – date of visit, TIME – viewing time, TYPE (INPUT/CLICK) – a way to go to the page.

The session denotes the active actions of the user on the site – page transitions. If within 30 minutes, the user does not perform any action, then a new session begins for him. In addition, a new session begins if the user enters a new search query in the address bar. Entries that are accumulated during one session and side by side form URL_{out}-URL_{in} pairs.

The presence of such pairs increases the weight of the corresponding web graph of the site by unity. As a result of combining session records of a significant number of users of the site, a corresponding transition graph was built.

For the RWR algorithm to work, a weighted matrix of probabilities of transitions between pages is constructed. For an individual user of the site, the initial approximation $P^{(0)}$ and the initial vector q with a defined starting vertex were formed.

The initial data obtained from the site user sessions were divided into sampling to train and test the recommendation system. According to the training sample, a transition graph was built. The sample for testing was used to assess the quality of the built recommendation system.

5. 3. Selection of metrics for assessing the quality of the recommendation system

In recommendation systems, to assess the quality of the recommendation provided, it is usually used a measure of the deviation of the values obtained using the recommendation system, and the actual values of the product rating for a particular user. The deviation measure is calculated in the norm of space L_1 or L_2 on all elements of the test sample [12]. However, when providing recommendations in the form of a relevant sequence of pages for viewing, the use of these metrics for evaluating the quality of recommendations is impossible. Therefore, accuracy and completeness metrics were used, which are most often used to assess the quality of solutions in information analysis tasks and are calculated as follows [14]:

$$P=\frac{T}{T+F};\ R=\frac{T}{T+FN},$$

where T is the number of elements from the test sample correctly determined by the RWR algorithm; F is the number of elements that are missing in the test set but determined by the algorithm; FN is the number of elements from the test sample that are not in the list obtained using the algorithm.

Since the recommendations are provided as a sequence of pages to view, the Half-Life Utility (HLU) metric [12] was also used to assess the quality. Using this metric enables finding the location of items in the recommendation list. An item farther away from the head of the recommendation list will gain less weight. The metric value is defined as follows:

$$HLU=100\cdot\frac{\sum_u R_u}{\sum_u R_u^{\max}},$$

where $R_u=\sum_{v\in V}\frac{\max(r(u,v)-d,0)}{2^{(p(u,v)-1)/(h-1)}}$ determines the relevance of the list of recommendations for the user $u\in U$; $r(u,v)$ is the page v rating for user u; d is the average rating; $p(u,v)$ is the page v number in the list of recommendations for the user u; h is the number in the list of recommendations, which corresponds to the page, which, with a probability of 50 %, would be viewed by the user.

5. 4. Results of computational experiment

We tested the proposed approach using an actual website of the *Country Seeds* online store (Ukraine) [16]. The web graph $G(V,E)$ of the resource was built, the power of sets was determined: $|V|=486$, $|E|=12{,}096$. The web graph of the site is represented by an adjacency matrix. The navigation graph was based on user conversion analytics between web resource pages that accumulated over 180 days.

To evaluate the built web graph and transition graph, some of their metric characteristics were calculated. The diameter of the graph, the average degree of the vertex, the average length of the path, the density factor of the graph were determined: the values are given in Table 1. The diameter of the graph defines the maximum distance between the two vertices through the other vertices. The mean degree of the vertex is equal to the average number of vertices with which the vertex is associated. The average path length between two vertices is equal to the average number of edges that connect the vertices. The density of the graph is defined as the ratio of the number of edges of graph $G(V,E)$ to the maximum possible number of graph edges with the same number of vertices.

Table 1

Metric characteristics of built graphs

Metric characteristic	Web graph of the Internet site semena-dnepr.org.ua $\|V\|=486, \|E\|=12{,}096$	Transition graph $\|V\|=486, \|E\|=9{,}134$
Graph diameter	13	11
The average power of vertices	18.2	14.1
The average path length	2.306	2.1
Graph density	0.039	0.0091

To assess the quality of the proposed approach, the average values of accuracy and completeness metrics for the list of users who found themselves in the test sample were evaluated. As elements of the test set, sequences consisting of the web resource pages visited by the user were considered, starting with some starting vertex. At the same time, the first 7 transitions ($L=7$) were considered as recommended pages.

Fig. 1 shows the distributions of relevant recommendations among the first L elements of the recommendation list. A solid line indicates the distribution obtained using the proposed approach. A dotted line shows the distribution calculated using the known PageRank technique [11].

The influence of the sample division proportions into the training and test ones on the quality of the results obtained from recommendations was studied. Fig. 2 shows the dependences of the HLU metric values on the number of items in the recommendation list, which are built for different ratios of the size of the training and test sampling.

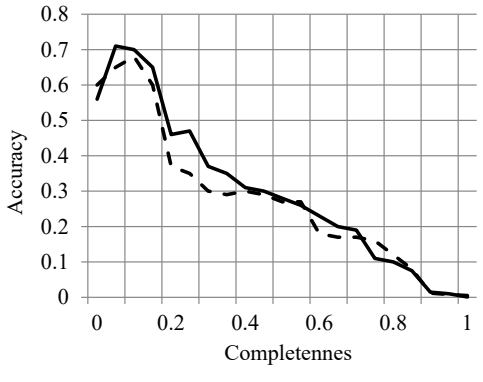

Fig. 1. Distribution of relevant recommendations obtained by RWR algorithm and PageRank algorithm [11]

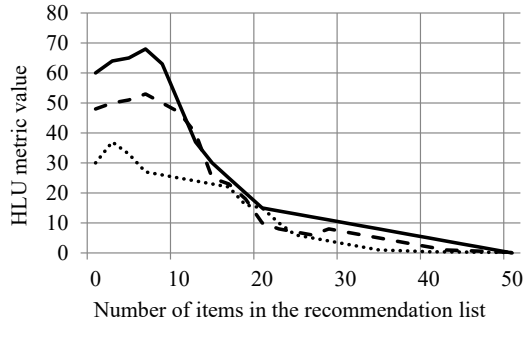

Fig. 2. Half-life Utility metric values dependence on the number of items in the recommendation list for different ratios of the size of the training and test sampling

A solid line corresponds to the case where the training sample includes 80 % of user session records, and the test sample – 20 %; a dotted line corresponds to the ratio of 70/30 %, a small dotted line corresponds to 50/50 %.

6. Discussion of results of the recommendation system performance

The analysis of the values of metric characteristics of graphs given in Table 1 is necessary to predict the time costs for the execution of the algorithm. The time for calculating the product of a matrix by a vector, which is necessary for the organization of iterative process (3), proportionally depends on the number of nonzero elements in the probability matrix of transitions. In the case when the transition graph has a large dimensionality and low density, RWR-based calculations may be performed too slowly. The low speed of obtaining an ordered list of pages recommended for viewing would make it impossible to form recommendations to the user under an online mode.

The comparison of the distribution of relevant recommendations obtained using our proposed approach with the distribution obtained using the well-known PageRank methodology [11] testifies to the effectiveness of the proposed approach (Fig. 1). Unlike the PageRank methodology, our proposed approach takes into consideration not only the semantic structure of the web resource but also the accumulated statistics on user behavior, which makes the model more accurate and more powerful.

It follows from the analysis of Fig. 2 that the size of the training sample significantly affects the quality of the recommendations. The best HLU metric values are 60–68 %; they are achieved when the sample division ratio is 80 % and 20 % for the training and test sampling, respectively. With a decrease in the size of the training sample, the quality of recommendations is significantly reduced. In the case when the training and test sample is equal, the worst accuracy indicators according to the HLU metric at the level of 30–35 % were obtained.

One can see from the analysis of Fig. 2 that the maximum values of the HLU metric were achieved for the first elements from the recommendation list v regardless of the proportions of the distribution into the training and test sample. As the item number in the recommendation list increases, the accuracy of HLU is significantly reduced. Since the highest accuracy values of 65–68 % were obtained for the first 7 items in the recommendation list, the optimal recommendation list size does not exceed 10 items.

In contrast to existing approaches to making recommendations, our proposed approach does not use information about the content and metadata of web resource pages, so it does not require syntax analysis of their content, which greatly simplifies the procedure for making recommendations and reduces the time to build them.

The disadvantages of the proposed approach include the fact that the results of the algorithm operation significantly depend on the quality of the initial statistics, which take into consideration only the movement of users between pages. For pages whose visit frequency is low, the quality of recommendations would be worse. To improve the quality of the recommendation system, the source data may consider other characteristics of the user's behavior, for example, the time spent on the page, the request that landed the user at the vertex, the type of device used, and others. According to the obtained data, it is possible to pre-cluster both web resource pages and users, which could make it possible to form group recommendations.

The result of the recommendation system performance depends significantly on the computational accuracy of matrix operations, which is reduced due to sparsity, large size, and other features of the built matrix of probabilities of transitions between the pages of the web resource. To eliminate this problem, in the future, it is proposed to apply additional procedures for regulating the solutions obtained.

7. Conclusions

1. The web resource model has been built as a web graph, and the user behavior model has been represented by a transition graph. The user's transition between vertices in a finite graph is carried out according to probabilities, which are calculated by the values of weights of the graph of transitions. We have determined that the user transition process between web resource pages is a Markov process with discrete time and discrete states.

2. A Random Walk with Restarts method is used to rank graph vertexes associated with an arbitrary predefined vertex. The use of this method makes it possible to perform random walks on the graph with the ability to jump to an arbitrary vertex of the graph, which corresponds to the model of user behavior when searching for information.

Taking into consideration data on the semantic links between web resource pages and at the same time statistics on user transitions between pages has allowed us to eliminate the cold start problem for the new user and improve the accuracy of recommendations.

To eliminate the computational complexities associated with the processing of a sparse matrix of probabilities of transitions, the proposed algorithm used a modification of the RWR method. The choice of algorithm parameters has been justified taking into consideration the peculiarities of the behavior of users of the web resource.

Software implementation involves the implementation of procedures for building a semantic structure of a web resource, collecting, and processing statistical data of user sessions.

3. To assess the quality of the recommendation system, the metrics of accuracy, completeness have been used, which could be applied in cases where there are no valid product rating values for a particular user. A Half-life Utility metric took into consideration the location of items in the list of recommendations, which is important when solving the problem of ranking the list of recommendations.

4. The proposed approach was applied to an actual website on the Internet. Using quality metrics, the comparison was performed with the well-known page ranking methodology PageRank, the influence of algorithm parameters and dimensions of the training sample on the quality of recommendations was investigated. The analysis of the performance of the accuracy metrics of the recommendations provided has allowed us to determine that the optimal size of the recommendation list does not exceed 7–10 HTML pages. It was established that the proposed approach could be applied to compile a list of recommendations to the user online.

References

1. Jansen, B. J., Booth, D. L., Spink, A. (2008). Determining the informational, navigational, and transactional intent of Web queries. Information Processing & Management, 44 (3), 1251–1266. doi: https://doi.org/10.1016/j.ipm.2007.07.015
2. Adomavicius, G., Tuzhilin, A. (2005). Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. IEEE Transactions on Knowledge and Data Engineering, 17 (6), 734–749. doi: https://doi.org/10.1109/tkde.2005.99
3. Meleshko, E. V., Semenov, S. G., Khokh, V. D. (2018). Research of methods of building advisory systems on the internet. Control, Navigation and Communication Systems, 1 (47), 131–136. doi: https://doi.org/10.26906/sunz.2018.1.131
4. Aizawa, A. (2003). An information-theoretic perspective of tf–idf measures. Information Processing & Management, 39 (1), 45–65. doi: https://doi.org/10.1016/s0306-4573(02)00021-3
5. Parfenenko, Y., Kovtun, A., Verbytska, A. (2019). Recommended information system for video search. Transactions of Kremenchuk Mykhailo Ostrohradskyi National University, 5 (118), 97–102. doi: https://doi.org/10.30929/1995-0519.2019.5.97-102
6. Candillier, L., Jack, K., Fessant, F., Meyer, F. (2009). State-of-the-Art Recommender Systems. Collaborative and Social Information Retrieval and Access, 1–22. doi: https://doi.org/10.4018/978-1-60566-306-7.ch001
7. Uschold, M., Gruninger, M. (2004). Ontologies and semantics for seamless connectivity. ACM SIGMOD Record, 33(4), 58–64. doi: https://doi.org/10.1145/1041410.1041420
8. Covington, P., Adams, J., Sargin, E. (2016). Deep Neural Networks for YouTube Recommendations. Proceedings of the 10th ACM Conference on Recommender Systems. doi: https://doi.org/10.1145/2959100.2959190
9. Stotts, P. D., Furuta, R. (1988). Adding browsing semantics to the hypertext model. Proceedings of the ACM Conference on Document Processing Systems – DOCPROCS'88. doi: https://doi.org/10.1145/62506.62516
10. Ol'shevskiy, A. I., Kondrat'eva, A. A. (2008). Opisanie sposobov predstavleniya web-saytov v vide freymovoy modeli dlya realizatsii funktsional'nyh operatsiy v Internet-klientskih sistemah. Iskusstvennyy Intellekt, 1, 110–116. Available at: http://dspace.nbuv.gov.ua/handle/123456789/6551
11. Brin, S., Page, L. (1998). The anatomy of a large-scale hypertextual Web search engine. Computer Networks and ISDN Systems, 30 (1-7), 107–117. doi: https://doi.org/10.1016/s0169-7552(98)00110-x

12. Herlocker, J. L., Konstan, J. A., Terveen, L. G., Riedl, J. T. (2004). Evaluating collaborative filtering recommender systems. ACM Transactions on Information Systems, 22 (1), 5–53. doi: https://doi.org/10.1145/963770.963772
13. Tong, H., Faloutsos, C., Pan, J.-Y. (2007). Random walk with restart: fast solutions and applications. Knowledge and Information Systems, 14 (3), 327–346. doi: https://doi.org/10.1007/s10115-007-0094-2
14. Huk, N., Dykhanov, S., Matiushchenko, O. (2020). Algorithm for building a website model. Bulletin of V.N. Karazin Kharkiv National University, series «Mathematical modeling. Information technology. Automated control systems», 47, 25–34. Available at: https://periodicals.karazin.ua/mia/article/view/16486
15. Olson, D. L., Delen, D. (2008) Advanced Data Mining Techniques. Springer, 180. doi: https://doi.org/10.1007/978-3-540-76917-0
16. Nasinnia krainy. Available at: http://semena-dnepr.org.ua/

In modern conditions, due to the vastness of the territory of Kazakhstan, with a certain probability, natural disasters such as earthquakes, floods, avalanches, as well as accidents, destruction of buildings, epidemics, release of chemical toxic substances at industrial enterprises, fires in educational and medical institutions are possible, which justifies the relevance of modern methods and technologies for solving the problem of evacuation.

The peculiarity of this work lies in the formation of an integrated approach for organizing the evacuation process both in peacetime as training for the event of an emergency situation (emergency), and in the event of the emergency itself. A conceptual diagram of an evacuation system is proposed that uses heterogeneous sources for receiving and transmitting information about the onset of an emergency. The input and output sources for receiving and transmitting information about the number of people in the building are determined. The main purpose of the system is to form an operational real-time evacuation plan.

This work is the result of a phased implementation of an integrated evacuation system, which consists in building a mathematical model and a method for solving the problem of maximum flow in the network. A mathematical model has been developed for the optimal flow distribution along the Grindshiels network with the analysis of the flow formation and the characteristics of people's motion in enclosed spaces. A game-theoretic approach and mathematical methods of the theory of hydraulic networks for finding an equilibrium state in flow-distribution networks have been developed. An algorithm for solving the evacuation problem using the graph approach is proposed.

The results of this paper make it possible to systematically organize training evacuations, prepare resources, train the personnel responsible for evacuation in order to quickly respond in an emergency and carry out the evacuation process in order to avoid major consequences

Keywords: maximum flow, optimal plan, Grindshiels network, Nash equilibrium, evacuation planning

UDC 004.9
DOI: 10.15587/1729-4061.2021.234959

DEVELOPMENT OF A SYSTEMATIC APPROACH AND MATHEMATICAL SUPPORT FOR THE EVACUATION PROCESS

Yedilkhan Amirgaliyev
Doctor of Technical Sciences, Professor,
Chief Researcher, Head of the Laboratory*
Aliya Kalizhanova
PhD, Associate Professor
Almaty University of Power Engineering and Telecommunications named after Gumarbek Daukeyev
Baytursynuli str., 126/1, Almaty,
Republic of Kazakhstan, 050010
Ainur Kozbakova
Corresponding author
PhD, Associate Professor, Leading Researcher*
E-mail: ainur79@mail.ru
Zhalau Aitkulov
Research Associate*
Aygerim Astanayeva
Doctoral Student, Researcher*
*Laboratory of Artificial Intelligence and Robotics
Institute of Information and Computational Technologies
of the Committee of Science of the Ministry of Education
and Science of the Republic of Kazakhstan
Shevchenko str., 28, Almaty,
Republic of Kazakhstan, 050010

Received date 26.04.2021
Accepted date 07.06.2021
Published date 29.06.2021

How to Cite: *Amirgaliyev, Y., Kalizhanova, A., Kozbakova, A., Aitkulov, Z., Astanayeva, A. (2021). Development of a systematic approach and mathematical support for the evacuation process. Eastern-European Journal of Enterprise Technologies, 3 (4 (111)), 31–42. doi: https://doi.org/10.15587/1729-4061.2021.234959*

1. Introduction

The relevance of evacuation from a building as a way to protect the population in peacetime has increased in recent years. The practice of modern life suggests that the population is increasingly exposed to dangers as a result of natural disasters, accidents and catastrophes in industry, fire in buildings, terrorist attacks, and this need may also be caused by poor-quality construction of administrative and residential premises. Especially prompt and successful evacuation is