Ukrainian Catholic University

Master Thesis

Topological approach to Wikipedia article recommendation

Author:
Maksym Opirskyi

Supervisor:
Petro Sarkanych, PhD

A thesis submitted in fulfillment of the requirements
for the degree of Master of Science

in the

Department of Computer Sciences
Faculty of Applied Sciences

Lviv 2020

Declaration of Authorship

I, Maksym OPIRSKYI, declare that this thesis titled, “Topological approach to Wikipedia article recommendation” and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed:

Date:

UKRAINIAN CATHOLIC UNIVERSITY

Faculty of Applied Sciences

Master of Science

Topological approach to Wikipedia article recommendation

by Maksym OPIRSKYI

Abstract

Human navigation in information spaces has increasing importance in ever-growing data sources we possess. Therefore, an efficient navigation strategy would give a huge benefit to the satisfaction of human information needs. Often, the search space can be understood as a network and navigation can be seen as a walk on this network. Previous studies have shown that despite not knowing the global network structure people tend to be efficient at finding what they need. This is usually explained by the fact that people possess some background knowledge. In this work, we explore an adapted version of the network consisting of Wikipedia pages and links between them as well as human trails on it. The goal of our research is to find a procedure to label articles that are similar to a given one. Among others, this would lay a foundation for a recommender system for Wikipedia editors, which will suggest links from the given page to the related articles. Our work is, therefore, providing a basement for enhancing the Wikipedia navigation process making it more user-friendly.

Acknowledgements

I would like to express my gratitude to Petro Sarkanych, my thesis supervisor for his tremendous amount of help, ideas and advice during this project. I am also grateful to Grammarly for partially covering my tuition fees. Special thank to Ukrainian Catholic University for providing a really good place to study and for organizing exciting courses, especially the course on Network Science, which resulted in this work. Last but not least, my gratitude comes to my family that constantly supported me throughout this period.

Contents

Chapter 1

Introduction

Over the last five years, the total number of web-pages has nearly doubled from 900 million pages in 2014 to more than 1.7 billion pages in 2019 [1]. With this increase in size, it is obvious that the amount of information has grown as well, making its retrieval much harder.

The Web itself is a good example of a large information network, where pages play role of nodes and links between pages are the edges. Furthermore, information networks can often be divided into smaller pieces, each with narrower specialization. Even then, the task of finding a particular bit of knowledge is not trivial, since smaller networks can have complex structure as well. However, having the goal, humans can search for it quite efficiently, as shown by West and Leskovec [48]. This fact is explained in a way, that often the structure of the network preserves real-world properties, in a sense that two concepts that are related have a higher chance to be connected (or to be within the small range of each other) in the network. Even though humans do not know the global structure of connections between concepts, they usually have an idea or vague understanding of their relatedness and can efficiently exploit this background knowledge when searching for information [21, 22].

Nevertheless, the notion of the relatedness can be ambiguous and often is context-dependent. For example, consider concepts Water, Molecule and Swimming. Clearly, Water and Molecule are related closely, as well as Water and Swimming. However, if one is, say, looking for information about the water states, she barely needs anything related to swimming. Moreover, the structure of the network does not only rely on the semantic relatedness between the concepts it describes. The structure can be dictated by the purpose of the web resource and its design. These arguments demonstrate the need for the tools that would help humans to navigate through the networks efficiently.

In this work, we pursue the goal of developing a procedure that could mark similar pages, based on the topological properties of the network. To this end, we explore the subset of the Wikipedia network. In this network, articles are represented by nodes and hyperlinks between articles are edges in the corresponding graph. Wikipedia is a good approximation to other information networks as it is a collection of concepts and its structure partially represents human knowledge. On the one hand, the aforementioned procedure would allow unveiling the connection between the concepts. On the other hand, one can use it for purely applied tasks. One such task is the recommendation of similar pages for the "See also" section of the article. At the moment, not every article has this section. Moreover, the present "See also" sections were manually created by the Wikipedia editors. There are many articles in Wikipedia, and it could be time-consuming to scan through them in order to obtain good recommendations. Therefore, we propose to reduce the articles one

[1] https://www.internetlivestats.com/total-number-of-websites/

needs to walk through by recommending only related ones. There are many methods that can be used for recommending[23, 34, 37, 42, 17, 16, 39]. Among them, we chose to use random walks which proved to be useful in the context of recommender systems [51, 6, 10, 18].

Technically, the problem we pose is a ranking problem: given an article, find and output a list of top k related articles. Implementation of such an algorithm would allow enhancing the navigation by proposing articles that are relevant for the user. Similarly, editors would be able to use it to create the "See also" section. This would make Wikipedia browsing experience more pleasant for both the information seekers and the contributors.

We highlight the research side of this work by posing the following questions:

- Can we use topological information about the network in order to obtain recommendations for articles?
- Is this information sufficient? If not, do the human navigation paths enrich this information and contribute to the goodness of the recommendation?
- Which of these options - solely structural information or navigational information - is more important for good recommendation?

The remainder of this work is structured as follows. In Chapter 2 we overview the work that has been done in the researched topic. In Chapter 3 we describe the datasets we use in work. Chapter 4 describes our method. In Chapter 5 we conduct the research of the method properties. Chapter 6 presents the evaluation strategy and results. In Chapter 7 we discuss the extension of our method and analyze the significance of our approaches. Finally, Chapter 8 presents our conclusions.

Chapter 2

Related work

In this chapter we provide a brief overview of the topic. The problem at hand is complex and is mainly related to four areas of research. First and foremost, the "See also" section of the Wikipedia page is added to help users to navigate to similar pages. Thus it is essential to understand the process of navigation in networks. This topic is discussed in Section 2.1. Next, in Section 2.2, we discuss patterns in search strategies. We plan to use random walks to develop a recommender system. The former are discussed in Section 2.3, while the latter - in Section 2.4.

2.1 Navigation in networks

Taking into account that the topology of a network possesses neither short nor long ordering, the structure of the network far away from an element does not allow to predict its position. Thus developing an efficient navigation algorithms for network searches is not a trivial task. In 2000 Kleinberg showed that networks that possess small-world property could be efficiently navigable [22]. In particular, he proved that time needed by any decentralized algorithm is bounded by a polynomial in $logN$, where N is the number of nodes in the network, provided that network is constructed using a specific model.

This finding was used in [1] to perform searches on social networks. The resulting search paths were similar to human-like ones, and the efficiency depended on the background knowledge of the network structure.

Further, Trattner et al. [44] applied decentralized search algorithms with so-called hierarchical knowledge of the network to model human searches in information networks. They use an algorithm for decentralized search, which is based on previous work and utilizes given hierarchical knowledge. Evaluation of their models of hierarchies is done by comparing them to human navigation paths. They conclude that the model using a hierarchy based on the network topology produces results that are the closest to human search trails.

In [49] West et al. use data gathered from Wikispeedia[1], an on-line game, where players are given a source node and a target node, and the goal is to reach the target using the least number of steps, clicking on the hyperlinks exclusively. The dataset consists of players' game paths. Authors use this information to develop a new information-theoretic metric which measures the semantic distance between concepts that Wikipedia articles represent. They also develop an approach to filter out concepts that have a small distance to the goal concept but are, in fact, irrelevant. They use concepts that are labelled as relevant or irrelevant by humans to teach a neural network to do this kind of filtering. Proposed method outperforms Latent Semantic Analysis which is validated by the psychological community. However,

[1] https://www.cs.mcgill.ca/~rwest/wikispeedia/

one flaw of their approach is the inability to calculate the similarity if one of the articles never occurred in any trail. In order for the method to work well, a huge amount of paths to calculate probabilities is required.

In light of the previously mentioned studies, Niebler et al. [33] investigate the ability to extract useful semantic information from various web resources. They argue the usability of game-like navigational data like Wikispeedia in computing semantic relatedness between concepts that Wikipedia contains. In turn, they study unconstrained data, where users are allowed to jump to any other article (i.e. use search field) and try to see whether it is suitable for extracting semantic information. Using an adaptation of their previous work, they show that unconstrained navigation data indeed can be used for semantic extraction.

West and Leskovec [47] further explore human navigation, this time comparing it to the automatic navigation performed by synthetic agents. The main requirement to the agents is to be local, in a sense that every step in a search can only be based on the local network characteristics and a given target. They used two types of agents. First group of agents only calculates properties of every possible node before making a step. Second group contains algorithms that learn either from the user paths or from previous steps. The authors show that agents can perform the task of transition from goal to target more effectively than humans. They conclude that the background knowledge humans possess in not necessary for successful Wikipedia navigation. However, humans, in contrast to algorithms, rarely get entirely lost. It is explained by the fact that humans can make longstanding plans.

In [25] Lamprecht et al. study how the structure of each article influences user navigation. They firstly confirm that lead sections and infoboxes[2] of articles contain links to more general concepts. Using data about user click behavior in Wikipedia as well as Wikispeedia game paths, they explore what influences users' trajectories. Then, they compare distributions of ground truth game clicks and clicks suggested by first-order Markov model with different choices of transition probabilities to examine the influence of factors like the structure of the page or its generality on human choice. Next, they analyze only goal-oriented navigation using the same approach but construct the models for each step of the path of each length and every optimal path length. Overall, they confirm that human navigation is biased towards the article structure.

2.2 Patterns in search strategies

In [48] West and Leskovec ask why human navigation paths, while different from optimal ones, are still efficient on average. They confirm that both high degree and high similarity are valuable for the choices, but at the early stages of the game people choose high degree nodes, and then the textual similarity of the articles becomes significant while approaching the end of the game. Furthermore, the similarity of articles is more significant for successful games. Finally, using the Markov model with two types of click probability - binomial logistic and learning-to-rank (multinomial) - authors develop a method to predict the target of the game having only the beginning of the path. This method beats the baseline that predicts the target by choosing an article that has the highest textual similarity with the last article in a given path.

While successful games contain a considerable amount of useful information, unfinished game paths can provide valuable findings as well. Authors in [41] aim to

[2] Infobox is a short summary of the article, usually placed in the top right corner.

identify search abandonment reasons by analyzing unfinished paths of Wikispeedia players and comparing them to finished ones. They report that properties like PageRank or indegree of the target are influential on game abandonment since the difference between these properties in finished and unfinished games is statistically significant. On the other hand, source properties are not a factor for abandonment, although the difference in the latter case is statistically significant as well. Another finding is that unsuccessful missions can be characterized not as "getting lost" in the network, which can be described as increasing shortest path length (SPL) to target or source, but as orbiting close around the target and inability to find a required link. Authors also analyze back clicking patterns, reporting that users have a higher probability of backtracking when SPL or $tf\text{-}idf$ distance [29](calculated as one minus $tf\text{-}idf$ similarity) to the target is getting bigger. What is more, backtracking is more probable in these situations for better players, meaning that they not only can better plan their moves but also better understand when they are getting lost. Authors also develop a model that predicts whether a user will abandon the mission, whether the next click will be back-click and whether a user will give up the search after current node, having the first couple of clicks as input.

Contrasting to [48] research conducted in [38] focuses on strategies in a scenario where no explicit target is specified. Rodi et al. explore Wikipedia user click data and consider the last article visited as the target. To find patterns of search strategies, authors develop a vectorial representation of the page where coordinates represent 13 abstract topics. Authors simulate human paths with random walks with transition probabilities based on click fractions available from the dataset and compare them to Wikispeedia real paths. They report that Wikipedia readers tend to start from abstract pages and narrow down to the specific ones, while Wikispeedia players first head to hubs and then narrow down. While semantic distance between consecutive pages is changing the most at the beginning and the end of the path, distance to the target keeps decreasing monotonically.

2.3 Random walks

The term "random walk" was first introduced in "The Problem of the Random Walk" by Pearson in 1905 [35]. This term refers to the mathematical object that is used to describe stochastic processes. This term is widely used in many fields of modern science. For example, in physics, random walks are used to describe such phenomena as diffusion or Brownian motion [15]. In economics, there are models which utilize random walk to model the stock market prices [14]. In chemistry, a polymer in a solvent is represented as a random walk [15].

Simply speaking, a random walk is a sequence of transitions between elements of state space. The state spaces can be different, and the definition is dependent on a particular application. The possibility of transition is encoded as a probability of moving from one state to another. One simple example of a random walk is the walk on 1-dimensional integer line. If the walker is placed at the origin of the axis and is allowed to take a step of length one either to the left or to the right with equal probabilities, then the movement of this walker is a random walk.

The main advantage of a random walk is its Markovian nature. It means that probabilities at each step of a walk are only defined by the previous steps. This property allows for an analytical description of a walk. The aspect which is crucial in the description of a walker is the topology. Random walks are often considered embedded in space. Two most commonly used cases are walks on lattices and walks on graphs. Below, we expand more on each of these cases.

2.3.1 Random walks on lattices

Given a lattice, random walk on it is defined to be a sequence of random steps starting from some point and moving only to adjacent points according to some probability distribution. In the easiest case, the probabilities are homogeneous. If we consider the one-dimensional chain, then the probability of making a step to any of the two directions will be equal to $p = 1/2$. Let's denote $P_N(m)$ to be the probability of a walker to be at position m at a step N. Than one can write the following

$$P_{N+1}(m) = \frac{1}{2}P_N(m-1) + \frac{1}{2}P_N(m+1). \tag{2.1}$$

Subtracting $P_N(m)$ from both sides of the Eq. 2.1 and performing continuous limit leads to the partial differential equation

$$\frac{\partial P}{\partial t} = D\frac{\partial^2 P}{\partial x^2}. \tag{2.2}$$

This simple example shows how random walks are related to such processes as diffusion and Brownian motion. Solving this equation with conditions that at the beginning the walker started from the origin, and the probabilities of reaching infinities are equal to 0 leads to the probability function

$$P(x,t) = \frac{1}{\sqrt{4\pi Dt}} \exp\left(-\frac{x^2}{4Dt}\right). \tag{2.3}$$

This probability function supports the obvious conclusion that the average coordinate of a walker $\langle x \rangle = 0$. However, the mean square displacement grows with time according to the law

$$\langle x^2 \rangle = \int_{-\infty}^{\infty} x^2 P(x,t)dx = 2Dt, \tag{2.4}$$

or in a more convenient form $\sqrt{\langle x^2 \rangle} \propto t^{1/2}$. This result means that the average distance from the origin to the walker grows as a square root of time.

It is worth mentioning that this conclusion holds not only in one dimension but for any dimensionality. The key property, which bolster Eq. 2.4 is the homogeneity of space. Graphs, in general, are not homogeneous; thus, random walks on them are less predictable. In the next Subsection we are going to expand on this case more.

2.3.2 Random walks on graphs

Walks on graphs can be seen as a generalization of random walks on lattices. Given a directed weighted graph $G = (V, E)$, a random walk on this graph is defined as a sequence of steps starting from some node and randomly moving to one of its neighbors and repeating this process. The probabilities of moving to a particular neighbor are expressed via the weight of the edge between the current node and its neighbor. Naturally, weights of all outgoing edges of a node should sum up to one, i.e.

$$\forall u \sum_{v \in N(u)} w_{uv} = 1 \tag{2.5}$$

where w_{uv} denotes the weight of an edge between u and v, and $N(u)$ is set of neighbours of u.

Random walk on a graph is also a special case of Markov chains. It is a stochastic process that satisfies Markov property (often called memorylessness), that is probability of being in state x_j at time n is dependent only on the state x_i which was visited at time $n-1$, technically:

$$Pr(X_{n+1} = x | X_n = x_n, X_{n-1} = x_{n-1}, ..., X_1 = x_1) = Pr(X_{n+1} = x | X_n = x_n) \quad (2.6)$$

Markov chain is described by its transition matrix P_n, a $|V| \times |V|$ matrix with nonnegative entries, whose rows sum up to 1. This matrix is known as Markov matrix. Element p_{ij} represents a probability of moving from state i to state j at timestep n. One important property of random walks is reversibility. Reversibility means that for every sequence of random steps on a graph, the reverse sequence is also a random walk [2]. There is a class of random walks that possess time-homogeneity property. Throughout this work, we will be dealing with time-homogeneous random walks, though others exist. Time-homogeneity means that for each timestep n the transition probability between states i and j is the same, i. e.

$$P_0 = P_1 = ... = P_n = P \quad \forall n \quad (2.7)$$

It is easy to see that probability of finding oneself in state j starting from state i in two steps is given by the i,jth entry of matrix P^2. Similarly, the probability of being in state j starting from state i in k steps is given by the i,jth entry of matrix P^k. Given transition matrix P and a random vector $v \in \mathbb{R}^{|V|}$ that represents the initial distribution of being at each node, random walk can be seen as a series of multiplications of P and v. Although it is unknown where exactly the walker would end up being after k steps, it is possible to obtain some estimates of its position. It is given by a stationary distribution vector π, that is defined as a vector that does not change under application of P, i. e.

$$\pi P = \pi \quad (2.8)$$

It is easy to see that π corresponds to (left) eigenvector of P associated with eigenvalue 1. P always has eigenvalue 1, since its rows add up to 1. Moreover, it can be shown that this is the largest eigenvalue of P. This gives a possibility to calculate the stationary distribution vector using e. g. Power iteration method [30].

Intuitively, π represents probabilities of finding oneself in each of the states after a long random walk. This idea is used, for instance, in the PageRank algorithm [8], that ranks pages on the Web according to their importance.

2.4 Recommender Systems

Algorithms beneath the recommender systems are traditionally classified into content-based and collaborative [3, 27]. Drawbacks of each type of algorithms alone can be overcome by combining them, which results in hybrid recommendation techniques [3]. However, there are other approaches. For example, in [20] collaborative filtering algorithm is used. Its central problems - cold start and data sparsity - are attacked by using Wikipedia for extracting similarity information between items and using it to compute artificial ratings for items that are not rated by the user. Authors use textual similarity of the articles, as well as category similarity and degree information.

This differs from our approach, as we aim to extend information used by taking into account the topological properties of the network and users' trails.

In [11] a bot that routes tasks for Wikipedia editors is developed. It uses the textual information and link structure available from Wikipedia to make recommendations about work that the editor should do based on the history of their previous contributions.

Recommender system described in [7] utilizes a hybrid approach to combine different sources that can be used to produce music recommendations. It uses Wikipedia to get information that is most relevant to the user's profile. In this setting, data from Wikipedia is just a supportive mechanism to the whole system.

Random walks are also used for the recommender systems. In [51] a random walk is set on the graph derived from Wikipedia to calculate semantic relatedness. The authors use personalized PageRank, which is a random walk with weighted probabilities for each step. The outcome is that they were able to achieve small improvements on a state-of-the-art measure. This result also yields that topology of an information network, Wikipedia in this case, reflects semantic relatedness between concepts.

Authors in [6] introduce a ContextWalk algorithm that is based on a random walk on a context graph. Transition probabilities in this algorithm are weighted. The graph has multiple layers, where each layer represents different types of contextual information. This algorithm was successfully used for the movie recommendation.

The research conducted in [10] is also devoted to developing a recommender system for the movie database based on the random walk. Authors consider a bipartite graph of users and movies they marked. Random walks show a similar time- and memory-efficiency to the other methods considered in the paper.

Hickcox and Min in [18] analyze Wikispeedia network to develop a recommender system for similar Wikipedia articles. To this end, they set up a random walker with a step probability dependent solely on the next node properties. The nodes that are occurring the most number of times in a series of random walks are considered as similar. As a recommendation quality measure, they used the tf-idf similarity. Five (including uniform random walk) out of seven probability schemes scored nearly the same value. Taking into account that for the uniform random walk distance from the source grows as a power law $R \propto t^{\nu}$, where ν is so-called Flory exponent (see, e.g. [43])[3], this leads to the conclusion that the methods suggested in [18] perform poorly, and, most probably, recommend nearest neighbours of a given article. To the best of our knowledge, [18] is the only work, where tasks similar to ours were put forth. We describe how our approach differs from one considered by Hickcox and Min in Chapter 4.

[3] This exponent is dependent on the properties of space.

Chapter 3

Datasets

In this section we describe the data we are going to use for the thesis. Mainly, we use two sets of data. First one is the Wikispeedia game dataset, which contains the image of Wikipedia from a 2007 CD version for schools and human trails on it. Second is the dataset we collected with information on the links present in the "See also" sections of Wikipedia pages. The second one is used for the evaluation of the methods we consider. Below we go more into details to describe both of these datasets.

3.1 Wikispeedia network

In the research, we use the data available from human-computation game Wikispeedia. The game can be understood as a walk on the network, where articles are represented by nodes and directed link between nodes exists if there is a hyperlink from one article to another. This dataset consists of two parts: Wikispeedia graph and collection of human game paths.

We choose this dataset for multiple reasons. Firstly, its size allows for easier computations and faster testing of hypotheses. Secondly, it still represents an information network, since it is a version of Wikipedia for schools from 2007. Thirdly, apart from the structure of the network, we also have human trails on them as additional information.

For a better understanding of the Wikispeedia network, we compute the basic topological properties of it. Since links in Wikipedia represent a reference to another article, we compare Wikispeedia network with another example of information networks - citation network [24, 26, 12]. To this end, we denote A_{ij} as an adjacency matrix of the Wikispeedia graph. If $A_{ij} = 1$, then there is a directed link from vertex i to vertex j. Since the edges are pointing from one article to another, we are dealing with a directed graph and $A_{ij} \neq A_{ji}$.

Two measures define the size of a network - the number of nodes N and the number of edges L. Wikispeedia graph contains $N = 4592$ nodes and $L = 119882$ links. Citation network, on the other hand, has $N = 12590$ nodes and $L = 49759$ edges. One can see that the Wikispeedia network is much more dense compared to the citation network. This might be due to the fact that in order for one paper to cite another, the authors have to work out the paper they want to cite, while for Wikispeedia graph links are often added to general concepts, which are known to everyone. Main properties of the networks are listed in Table 3.1.

For the directed network, each vertex is described by in-degree $deg^{-}v$ and out-degree $deg^{+}v$. These values are defined through the adjacency matrix in the following way

$$deg^{-}v = \sum_{i} A_{iv}, \quad deg^{+}v = \sum_{i} A_{vi}. \tag{3.1}$$

Property	Wikispeedia	Citation network
N	4592	12590
L	119882	49759
$min_{(v \in V)} deg^- v$	0	0
$max_{(v \in V)} deg^- v$	1551	227
$min_{(v \in V)} deg^+ v$	0	0
$max_{(v \in V)} deg^+ v$	294	617
$\langle deg^- v \rangle$	26.11	7.9
$\langle deg^+ v \rangle$	26.11	7.9
$\langle l \rangle$	3.2	4.37
r	-0.056	0.012
$max_{(u,v)} l(u,v)$	9	10
GCC size	88.0%	99.2%
$\langle c \rangle$	0.11	5×10^{-5}
C	0.1	3×10^{-4}
α^-	1.39	1.44
α^+	1.81	1.86

TABLE 3.1: Topological properties of the Wikispeedia network comparing to the citation network. N and L denote the numbers of nodes and edges in the network, deg^- and deg^+ are in- and outdegree, $\langle l \rangle$ denotes average shortest path length, r is assortativity by degree, $max_{(u,v)} l(u,v)$ is network diameter, $\langle c \rangle$ is average local clustering coefficient, C is global clustering coefficient, α^- and α^+ are in- and outdegree distribution exponents.

In Table 3.1 we list maximum, minimum and average values of these properties. Wikispeedia network has much larger values of the mean and maximum node degree. This reflects the difference in the nature of the networks. While scientific citations are used only when the author considers it is needed for the paper, links in Wikipedia are suggested by the system itself, naturally increasing the number of links.

Another property that is essential for the networks is degree distribution $p(k)$. This function shows the probability of a randomly chosen node to have a degree k. For the directed networks, it makes sense to consider distributions of in- and outdegree separately. In Fig. 3.1 degree distributions are shown in double logarithmic scale. These distributions qualitatively are similar to the corresponding plots for the citation network [24, 26, 12].

The tails of both distributions look like a power-law function $p(k) \propto k^{-\alpha}$. This type of behaviour is inherent for scale-free networks [19]. For scale-free networks degree distribution exponent α plays a similar role to that of dimensionality of regular lattices. Thus exponent α is important for understanding the features of the network.

Usually, for small networks $p(k)$ contains noise. To filter it out, cumulative degree distribution $P(k)$ is considered. It is defined as

$$P(k) = \sum_{x=k}^{\infty} p(x). \tag{3.2}$$

It is easy to see that for the scale-free network with a degree distribution exponent α

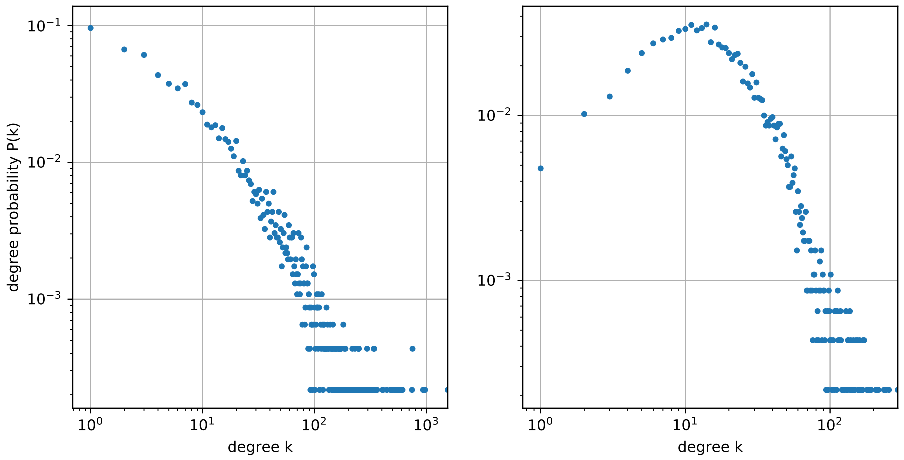


FIGURE 3.1: Degree distributions of Wikispeedia graph; (left) indegree distribution, (right) outdegree distribution

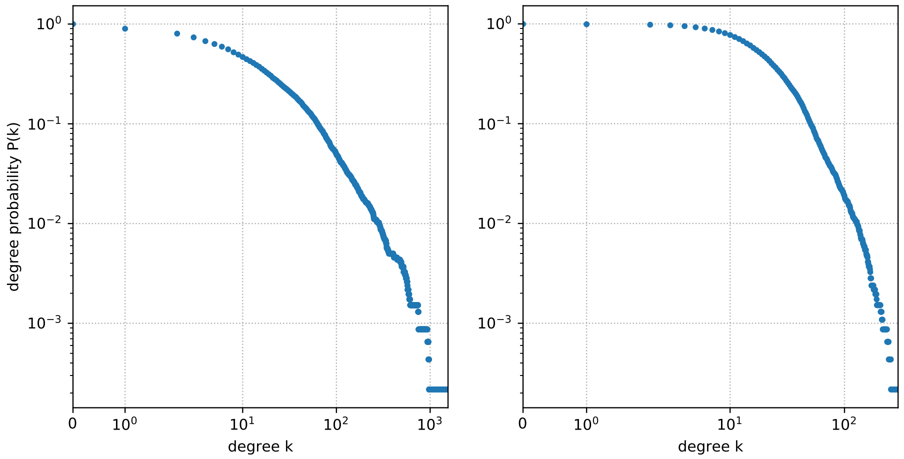


FIGURE 3.2: Cumulative degree distributions of Wikispeedia graph; (left) indegree distribution, (right) outdegree distribution.

cumulative degree distribution behaves like $P(k) \propto k^{1-\alpha}$. This function is smoother, thus it is easier to fit a power-law and determine the value of α. We distinguish indegree distribution exponent α^- and outdegree distribution exponent α^+. Fitting degree distribution to a power-law yields $\alpha^- = 1.39$ and $\alpha^+ = 1.81$. These values are similar to the corresponding values of the citation network we took for the comparison, mainly $\alpha^- = 1.45$ and $\alpha^+ = 1.86$.

A path in the graph is a sequence of nodes, where there is a link going from a node to the next node in the sequence. Connected component is a subgraph where there is a path between any pair of nodes. The largest connected component often called the giant connected component (GCC) is the component with the highest amount of nodes. For complex networks, it is inherent to have GCC nearly covering the whole network. For the Wikispeedia graph, the GCC is about 88% of the whole network, while for the citation network this value is much higher, reaching 99.2%.

Distances in a graph are defined as a number of edges a walker needs to traverse to get from one node to another. Distance between two nodes in the network can be found as a minimal power l, for which the following condition holds

$$(A^{l-1})_{ij} = 0, \quad (A^l)_{ij} \neq 0. \tag{3.3}$$

We calculate the average shortest path length $\langle l \rangle$ in order to describe the network as a whole. For the Wikispeedia network $\langle l \rangle = 3.2$, while for the reference network it is $\langle l \rangle = 4.37$. This is fascinating because in the graph made of more than 4000 nodes, any two nodes are separated on average by less than four steps.

Another property that gives an insight into the size of the network is its diameter d. It is defined as

$$d = max_{(u,v)} l(u,v). \tag{3.4}$$

For the Wikispeedia graph d is only 9, suggesting that the most distant articles still can be reached in a few steps.

Clustering coefficients of the network are noteworthy. We consider two definitions for the clustering coefficients. Local clustering coefficient is defined on a node and shows a fraction of the node's neighbors and the maximal possible number of edges in the node's neighborhood[46], i. e.

$$C_i = \frac{|e_{uv} : u, v \in N(i), e_{uv} \in E|}{k_i(k_i - 1)} \tag{3.5}$$

where $N(i)$ is neighborhood of node i, that is set of nodes that are connected to i with an edge, and k_i is the degree of node i. High value of local clustering coefficient means that the node's neighborhood is close to being a clique, giving an insight into how tightly connected the nodes are in a given area of the network. The average local clustering coefficient $\langle C \rangle$ describes the network as a whole.

Global clustering coefficient [28] of a graph is defined as a fraction of closed triplets in graph divided by the number of all triplets, where triplet is set of three vertices connected either by three (closed) or two (open) edges. High values of the global clustering coefficient tell that there is a huge tendency for nodes to cluster.

In addition to the citation network, we compare clustering coefficients of our network with clustering coefficients of the random graph with the same number of nodes and edges. For comparison, we chose the Erdös-Rényi model [13]. There are two variants of this model. We construct the random graph using $G(n, M)$ model, i.e. model, where the number of nodes and edges is fixed. It is equivalent to randomly (with uniform probability) choosing the graph with n nodes and M edges from the set of all such graphs. Average local clustering coefficient for the Erdös-Rényi graph is 0.0056, and global clustering coefficient is 0.011. These values are order(s) of magnitude smaller than that of Wikispeedia graph. Together with the small value of the average shortest path length, this demonstrates the small-world property of the network. On the contrary, this feature is not observed for the citation network.

Assortativity [32] shows to which extent nodes that share common properties tend to link to each other. We consider assortativity by degree that would show whether nodes with similar degree values have a high chance to be connected. Degree assortativity for directed networks is defined as follows:

$$r = \frac{\sum_{jk} jk(e_{jk} - q_j^{in} q_k^{out})}{\sigma_q^{in} \sigma_q^{out}}, \tag{3.6}$$

where j and k are remaining degrees of the nodes(i.e. degree of a node minus one), e_{jk} is fraction of edges that connect nodes of remaining degrees j and k; q_j^{in} and q_j^{out} are remaining degree distributions (indegree, outdegree respectively), i.e. fraction of nodes that have remaining degree j; σ_q is standard deviation of distribution q. Assortativity is thus a Pearson correlation coefficient of node attributes. This quantity lies in $-1 \leq r \leq 1$ with $r = 1$ meaning fully assortative and $r = -1$ meaning fully disassortative graph. We observe value that is close to zero for both Wikispeedia and Citation networks. The Wikispeedia network is slightly disassortative, meaning that nodes with small degree have a small tendency to link to nodes with high degree.

3.2 Human trails

Besides the topological structure of the Wikispeedia network, the database contains human game paths. According to the game rules, the user is given the start and the end pages. The goal of the game is to find the path involving the least number of steps. Game paths data are organized as a file where each row contains a sequence of article titles, representing a single path. In the game setting, the user is allowed to return to pages that she visited earlier. Back clicking is denoted by "<" symbol. There are 51318 finished and 24875 unfinished paths in the dataset. We used only finished paths, since they represent successful search instances. Having the collection of game paths, we construct a directed weighted graph, where the weight of the edge between two nodes represents the number of players' transitions from one node to the other.

3.3 Recommendation retrieval

Wikispeedia data is preprocessed subset of Wikipedia that contains articles on general topics. However, textual information available is limited, since this data does not contain any lists, references and other typical attributes of Wikipedia page. In order to have ground truth recommendations for similar pages, one needs to obtain "See also" sections for every article in Wikispeedia subset where it was present. We were unable to find the version of the Wikipedia dump that was used to build the Wikispeedia game. To overcome this problem we took the closest Wikipedia dump that we were able to get. It was from 2010. We parsed this version in order to obtain articles that are present in Wikispeedia dataset. For each article that had "See also" section, we extracted recommendations. For these recommendations, we cross-referenced them again with the Wikispeedia network. All together we did three steps of reducing the data from the Wikipedia dump of 2010. Doing so, we obtained 530 articles with available recommendations. These pages will be used to test all our assumptions and evaluate the performance of our algorithms.

Chapter 4

Proposed method

Random walk agents have been used for the recommender systems in the past, as stated in Chapter 2. Here we are going to explain in detail how we are going to set up the agents in our case. We will consider a number of probability distributions to test which one of them works the best. In particular, the properties that will be of interest for us are the following: textual similarity, distances, Pearson similarity and user game paths.

4.1 Random walk agents

To answer the questions posed in Chapter 1, we construct a number of random walk agents with different transition probability options. Correct choice of these options should produce paths that contain nodes which are similar to the starting one. The options use structural information about the network as well as human navigation paths. Essentially, all of them can serve as similarity metrics between articles. Hence, transition probabilities are proportional to similarity scores between articles. We now briefly describe each agent.

4.1.1 Uniform

The most straightforward setup for the random walk agent is the uniform agent. In this scenario transition probability from node i to node j is divided evenly among the articles that article i points to

$$p_{ij} = \frac{A_{ij}}{deg^{+}(i)}. \tag{4.1}$$

This random walk agent tends to visit nodes with high values of indegree. Furthermore, it assumes no similarity between the nodes. With all these drawbacks, this agent reproduces the results of a PageRank algorithm [9], which is used by Google to rank pages in search results.

4.1.2 Textual similarity-based

In this setting transition probability between articles i and j is proportional to the textual similarity of these articles divided by graph distance between them, i.e.

$$p_{ij} \propto \frac{sim(i,j)}{l(i,j)^{\alpha}}, \tag{4.2}$$

where $sim(i,j)$ denotes the similarity between the articles i and j and α is a positive parameter. In order to calculate the similarity between the pages, we represent them as vectors using tf-idf representation [40]. tf-idf produces a document representation by counting word occurrences. Given the collection of documents D and set of words in all the documents S, tf-idf representation of a document is a $|S|$-dimensional vector v. To this end, tth entry in dth document vector is

$$v_{dt} = tf(t,d) \cdot idf(t,D), \tag{4.3}$$

where $tf(t,d)$ is the frequency of term t in document d;

$$idf(t,D) = \frac{1}{log(|\{d \in D : t \in d\}|)}, \tag{4.4}$$

i.e. the reciprocal of the logarithm of the number of documents that contain term t. While numerator is big if term t occurs often, denominator penalizes for terms that are present in a huge number of documents. The idea is that words that occur in many documents are in some sense useless, as they do not have much semantic weight. It is common to use cosine similarity to measure how similar are the document representations. It is calculated as:

$$sim(i,j) = \frac{i \cdot j}{||i|| \cdot ||j||} = \frac{\sum\limits_{k=1}^{|S|} i_k j_k}{\sqrt{\sum\limits_{k=1}^{|S|} i_k^2} \cdot \sqrt{\sum\limits_{k=1}^{|S|} j_k^2}} \tag{4.5}$$

It is worth noting here, that tf-idf is considered as one of the predominant approaches in text-based recommender systems [4]. In our case, we will be using not a cosine similarity but just a dot product. This is due to the fact that dot product in the nominator of Eq. (4.5) is rather small, making all articles very similar.

Denominator of the step probability is represented by graph distance between article i and j. The idea is that topological distance should have a negative impact on the similarity between articles. We vary α parameter to measure the impact of the graph distance penalization. Notably, this approach has no limitations on the connectivity of the articles in the network. The reason for this is that recommendations for the "See Also" section are suggested to add a link in the network, thus limiting to walks only over the edges might not be a valid approach.

4.1.3 Pearson similarity-based

Besides textual similarity, we also consider the topological similarity between nodes. One of the characteristics that show it is Pearson similarity. It shows how many common neighbours two nodes share compared to the random graph. [31]. Pearson similarity of node i and j is calculated as:

$$r_{ij} = \frac{\sum_v (A_{iv} - |V|^{-1}k_i)(A_{jv} - |V|^{-1}k_j)}{\sqrt{\sum_v (A_{iv} - |V|^{-1}k_i)^2}\sqrt{\sum_v (A_{jv} - |V|^{-1}k_j)^2}} \tag{4.6}$$

where $|V|$ is the number of vertices in the graph, A_{iv} represents (i,v)th element of the adjacency matrix A of a graph, k_i is the (out)degree of node i. The value of r_{ij} that is greater than zero tells that two nodes have more neighbors than they would have, had they chosen their neighbors uniformly at random. This is a standard measure of similarity between graph nodes. Within this scheme, we allow a walker to take steps not only to the nearest neighbours.

4.1.4 Transition-based

In this setting transition probability is proportional to k_{ij} - the number of transitions from node i to node j in all the Wikispeedia game paths that we possess. This is equivalent to constructing a weighted graph of game paths, where weight is the number of people's transitions between nodes and taking the weight of the edge between nodes i and j as k_{ij}. The idea behind this choice of probability is that the more people move from one node to another, the more is the chance that there is a relation between concepts that articles describe. Therefore, this choice of probability encodes human knowledge about the relation between concepts.

4.2 Recommendation prediction

Having the random walkers described above the next natural step is to combine them in order to enhance the quality of recommendations. We manually constructed the rules for the transitions. We propose to automatically infer which of the rules are more important for the recommendation. To do this, we train a binary classifier on the task of predicting whether article j will be recommended to article i. The features of our data are metrics for a pair of articles described above. Positive examples for classification consist of established recommendations, i. e. a pair of articles such that the one article is in the "See also" section of another article. To obtain negative examples, for each article, we randomly sample articles that are not recommendations for the current one. Doing this way, we construct a balanced dataset on which we can train the classifier. Having the fitted parameters of the classifier, we could use them as importance coefficients in the linear combinations of metrics to design a combined agent that uses all the rules simultaneously.

Chapter 5

Random walks on Wikispeedia

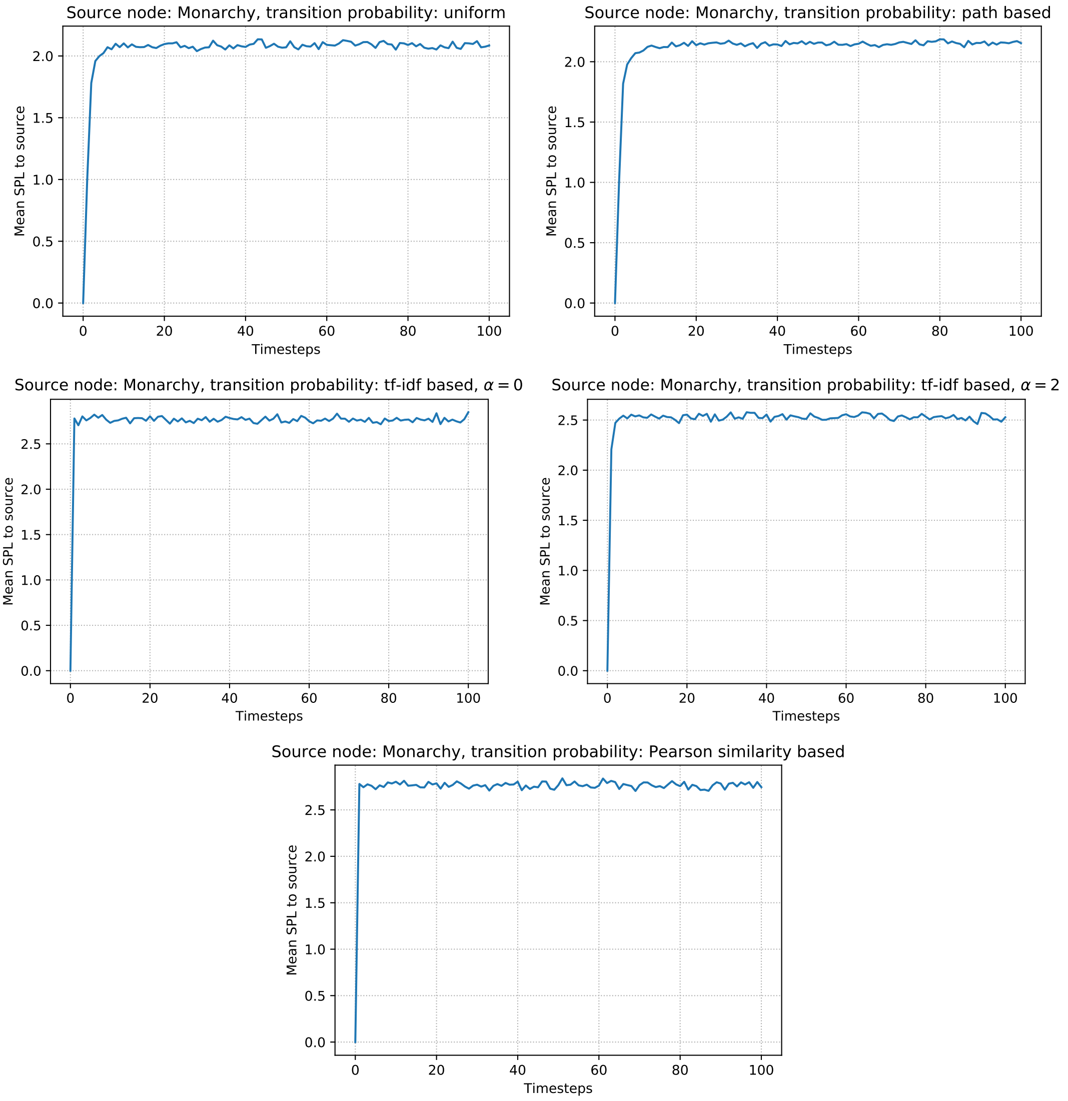
In this chapter we discuss the application of the random walk strategies introduced in Chapter 4 to the Wikispeedia graph.

5.1 Mean distance to the source

Before we proceed to the recommendations, we study the behavior of the agents. The first characteristics we will be looking at is how does the distance from the agent to the starting point change with time. We refer to distance between two nodes in the graph as to the number of edges in the shortest path between these vertices. Information about distances is summarized in Figure 5.1. These plots were obtained for the starting node *Monarchy* by averaging over 1000 walks. Corresponding plots for other origins show similar behavior. In each simulation, we fixed the starting point, calculated distance from it to the current position of the agent and averaged this distance across the simulations. All experiments showed similar results, so here we are reporting the behavior on the first 100 steps because it is stable in terms of the mean distance to the source. Agents that use textual similarity have greater mean distance to the starting point as the number of steps increases when compared to uniform and transition-based agents. Naturally, tf-idf agent with no penalization on distance ($\alpha = 0$) has greater mean distance than its counterpart ($\alpha = 2$). All agents are orbiting around the source at a distance that is less than the average shortest path length of the graph. This fact can contribute to the hypothesis that topology of the graph incorporates knowledge about semantic relatedness between articles. Since agents are moving in correspondence with different similarity metrics, the fact that the mean distance is less than average suggests that similar articles are located in the close neighborhood of the starting point.

Another interesting observation is that uniform agent and the one based on the game paths show smaller values of the mean distance to the origin. This is due to the fact that these two agents were only allowed to take steps to nearest neighbors, while the other three could jump without constraints.

All these plots can be compared to the behavior of an agent on the regular lattice. In our case, the plateau is reached in a few steps, while on the lattice SPL grows with time as a square root of the number of steps. This observation leads to the assumption that the space for an agent can be limited to the third coordination sphere. In our setup, this limit gives no significant advantage, but for the large networks, as the current version of Wikipedia, it might significantly speed up the process.


Source node: Monarchy, transition probability: uniform

Source node: Monarchy, transition probability: path based

Source node: Monarchy, transition probability: tf-idf based, $\alpha = 0$

Source node: Monarchy, transition probability: tf-idf based, $\alpha = 2$

Source node: Monarchy, transition probability: Pearson similarity based

FIGURE 5.1: Mean distance from source node for different random walk agents

5.2 Relation between degree and number of visits

We also explore the relation between the number of visits to a node and its degree. Results can be seen in Figure 5.2. Firstly, in the case of a uniform agent, Pearson correlation between the number of visits and the degree of the node is huge.

In the case of transition-based agent, correlation is significant as well. It shows that people tend to click on articles that have many links and highlights the importance of hubs in information search.

Textual similarity-based agents are significantly different from this point of view. The agent with no distance penalization has a significantly lower correlation between the number of visits and the degree of a node compared to the other agent. If graph distance is not penalized, the agent only considers tf-idf similarity of the articles, meaning that the topology of the graph is not taken into account. When the distance is penalized, correlation increases. This suggests that in case of zero penalization agent chooses textually similar articles that can be close or distant to the source (recall, that mean distance to the source is bigger in this case) and with penalization, only closer ones are chosen. The dependency between the degree of a node and its number of visits is greater for articles that are closer to the source.

Pearson similarity-based agent also exhibits the relationship between degree and number of visits. Correlation is not as big as in previous cases. Moreover, the slope of the fitted line would be small. This suggests that the number of visits to the node does not heavily depend on its degree in this case.

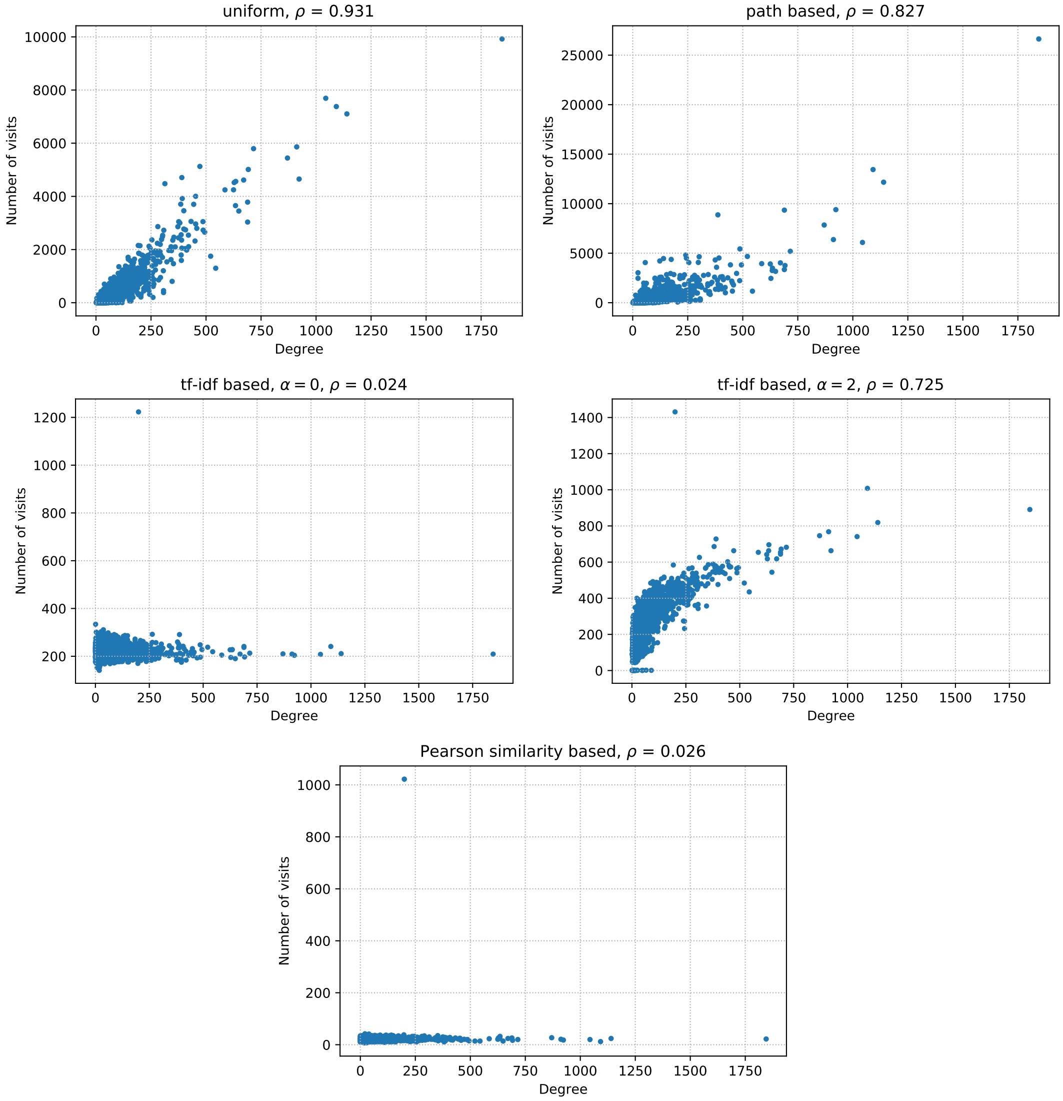


FIGURE 5.2: Relation between degree and number of visits. Source node is *Monarchy*, ρ denotes Pearson correlation coefficient.

5.3 Similarity changes

We continue the exploration of agents behavior by studying how do other properties of the agent change with time. This experiment has a similar setup to the one where we calculated the mean distance to the starting point. Here, we perform 1000 simulations, each with 100 steps and averaging across simulations as well.

Since most of our metrics express similarity between articles, we can analyze how these similarities changes as agent walks. Results are presented in Figure 5.3.

We need to note that these similarities are not comparable to each other due to their different range. Still, they all share quite similar behavior, namely that they are stable. Firstly, a textual similarity-based agent with no distance penalization has a mean similarity around 5.7 with fluctuations being about ±0.02. When distance penalization is acquired, the first step shows higher similarity. This can be explained by the fact that in the first step, graph distance remains small, then it increases, causing the metric to decrease. A spike is also present in the graph of transition-based metric. Though this type of metric is representing not necessarily the similarity between articles, since it is just the number of human transitions between these articles, it should encode a relation between concepts that are described by this article. This is why we are referring to the number of transition as to the proxy of similarity. In the first few steps, its values are bigger, but then they decrease and keep fluctuating around a value of 1.15.

We also measure the mean similarity of ground truth articles according to each metric and report these results in Table 5.1. Almost all ground truth recommendations have a similarity that is greater than agents produce, except Pearson similarity. It is almost identical to the results produced by the agent. Lower values of similarity produced by agents are natural since random walk does not necessarily always jump on similar articles.

Metric	Mean similarity
Textual similarity, $\alpha = 0$	5.92
Textual similarity, $\alpha = 2$	4.05
Pearson similarity	0.52
Number of transitions	5.31

TABLE 5.1: Mean similarity score for ground truth articles.

The stability of all agents in terms of their similarity to the starting point suggests that there is no point in having a long random walk since there is no significant difference between articles visited say on 20th step and articles visited on 100th step. On the other hand, it can be helpful to make the walk longer in order to obtain a better approximation of stable distribution corresponding to the transition probability matrix of the agent. What is better in terms of recommendations remains, for now, open question and is a good basis for future work.

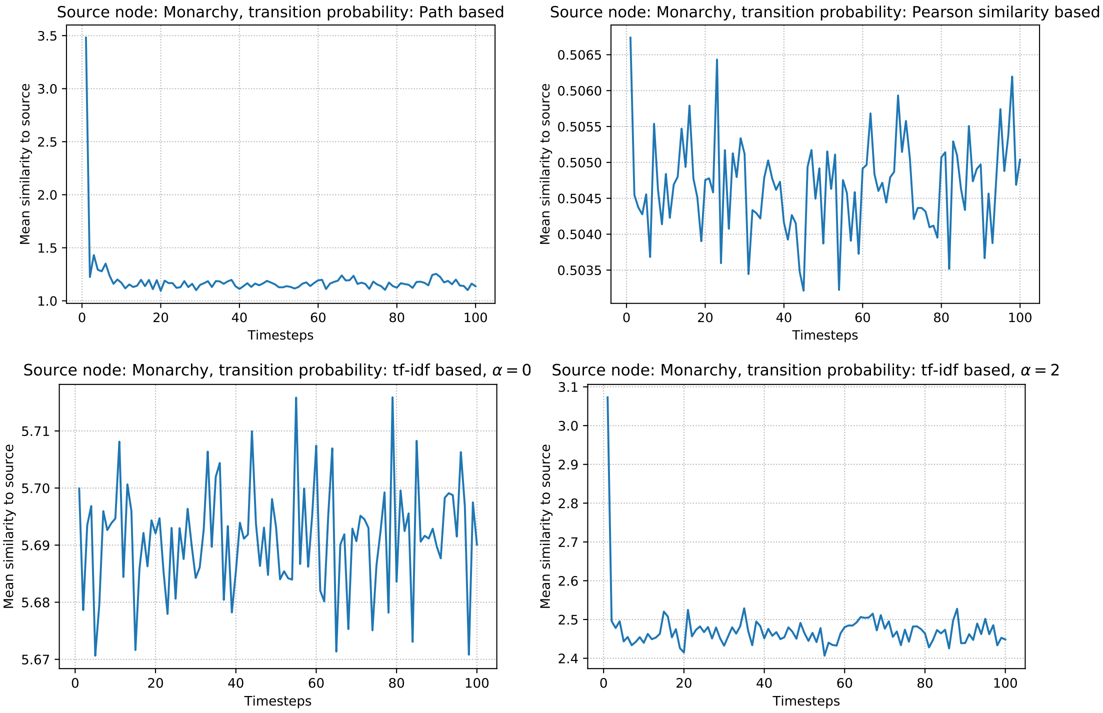


FIGURE 5.3: Similarity changes with time according to designed similarity metrics

Chapter 6

Evaluation

In this chapter we explain how we evaluate recommendations produced by random walk agents and present results of the evaluation. To this end, we use two evaluation metrics, variation of total reciprocal rank [36] and mean average precision(MAP) [29]. We discuss the results of every metric in Sections 6.1 and 6.2, respectively. In Appendix A we list the recommendations produced by every agent.

6.1 Rank-based evaluation

We evaluate agent results on the ground truth recommendations that we extracted from the 2010 year version of the Wikipedia. For each article that has true recommendations, we run the algorithms and compare lists produced by agents with true recommendations. Since the recommendation is a ranking problem, it is desirable to have relevant items (true recommendations) at the beginning of the list. One of the metrics that consider the ranking of relevant items is the mean reciprocal rank (MRR)[45, 36]. MRR is defined as the average(across queries) of reciprocal positions of the first relevant item in the list produced by a recommendation algorithm. This quantity gives a general understanding of the quality of the recommender system. The main drawback of this approach is obvious - it takes into account only the first relevant recommendation. In our scenario, this is inappropriate since there can be more than one relevant recommendation for the article and each of the recommendations is equally important. Let us say one agent successfully predicted recommendation on position 1 but did not manage to find any other relevant pages. At the same time, the second agent suggested three relevant recommendation on positions 2,3 and 4. Based on the MRR metric, the first agent will be scored higher. However, we believe that having three relevant recommendations down the list is better than having one on the top. Authors in [36] also use total reciprocal rank that simply sums all the ranks of relevant items in a list produced by an algorithm. We adapt this metric with a difference that we normalize it by *ideal* ranking sum to see how distant are we from ideal recommendation. We define this modified metric as follows:

$$V@k = \frac{\sum\limits_{i=1}^{k} r_i^{-1}}{\sum\limits_{i=1}^{min(k,rel)} i^{-1}}, \tag{6.1}$$

where k is a cutoff, rel is the number of relevant items, r_i is rank of the ith relevant item in the list that was produced by an algorithm. Cutoff means that only first k items of the list are considered. Irrelevant items are not taken into account. Denominator denotes ideal ranking, thus serving as a normalization to the quantity above.

This is in the range $[0, 1]$, with 1 meaning that recommendation is perfect. In table 6.1 we report the results of agents.

Transition based agent showed the best results. This tells us that information encoded in users' game paths is useful for making recommendations for similar articles. Interestingly, an agent with uniform transition probabilities has the second highest score. Simple topological information like the degree of the vertex and ability to randomly(uniformly) jump between neighbors can be better than more sophisticated similarity metrics like textual or Pearson similarity. Surprisingly, the score for the *tf-idf* based agent is the lowest among those, considered in our research, even though this metric is prevalent for the recommender systems [4]. In addition, the results of textual similarity-based agents show that when graph distance is taken into account recommendations are closer to the ground truth.

Agent	$V@50$
Uniform	0.0021
Transition based	0.0028
Textual similarity based, $\alpha = 0$	0.0004
Textual similarity based, $\alpha = 2$	0.0008
Pearson similarity based	0.0008

TABLE 6.1: Random walk agents results, rank based.

6.2 MAP-based evaluation

Another metric we consider is the mean average precision. MAP is a more common metric in information retrieval and ranking problems comparing to the metric described above. It is defined as the mean of average precision scores for a set of queries. Average precision of a query is defined to be the area under the precision-recall curve:

$$AP = \int_0^1 p(r)dr, \tag{6.2}$$

where $p(r)$ is outputs precision for given recall level r. Technically, it is computed as

$$AP = \frac{\sum_{k=1}^{n} P(k) \cdot I(k)}{rel}, \tag{6.3}$$

where $P(k)$ denotes precision of sublist consisting of first k items of the whole list, $I(k)$ is an indicator function that equals 1 if kth item is relevant and 0 otherwise and *rel* is the number of relevant items. MAP is also used with k as a cutoff value, meaning that for each AP we consider only the first k items in the produced recommendation lists. MAP penalizes relevant items appearing later in the list, so it is appropriate for ranking tasks. However, when there is a small number of relevant items for a query, MAP can score low values. For instance, let there be queries for which there are two and ten relevant recommendations, respectively. Then, MAP for a first query can easily be less than that for the second query, because in the first case, there are only two relevant recommendations. In the second case, there are more chances to obtain some score, because there are more relevant items. These arguments also apply for the metric described in Section 6.1 - when there a few true recommendations, this score tends to be lower. In our scenario out of 530 articles that have ground truth recommendations, 384 have only one recommendation. This can introduce unreasonably low values of MAP. In table 6.2 we report the results of MAP-based evaluation.

Agent	$MAP@50$
Uniform	0.0019
Transition based	0.0024
Textual similarity based, $\alpha = 0$	0.0003
Textual similarity based, $\alpha = 2$	0.0006
Pearson similarity based	0.0006

TABLE 6.2: Random walk agents results, MAP-based.

The results are coherent with rank-based evaluation. Noteworthy is the fact that MAP is more strict as it gives similar but lower scores than the first approach. Transition-based approach is also best here. Textual similarity-based agents show similar behavior, namely that distance penalization positively impacts recommendation quality.

Chapter 7

Predicting recommendations

In this chapter we present and discuss the results of the recommendation prediction. We also analyze coefficients of the trained classifier and discuss how metrics that were used in random walk agents affect its prediction.

We split the dataset generated as described in Section 4.2 into train and test sets (70% vs 30%), train a classifier on the first subset and evaluate on the second. Since we formulated this task as a binary classification problem, there are many options for classifiers that can be used. We train a logistic regression model [5]. We chose this classifier because apart from outputting the prediction for articles, it allows examining the parameters that were fitted during classifier training. This is of course, not the only classifier that tries to minimize some loss function with respect to its parameters. However, parameters of logistic regression can be interpreted naturally, and the significance of every input feature can be computed. Moreover, these parameters can be used as weighting coefficients for similarity metrics that we designed in order to combine them.

A single training example for the classifier consists either of the set of metrics discussed in Section 4.1 computed for article i and its ground truth recommendation j or the set of these metrics computed for article i and a random article that is not a recommendation for i. The former is considered as a positive example, while the latter is a negative one. Apart from the metrics discussed, we add more features that can be helpful for prediction. These are $deg^{-}(i), deg^{+}(i), deg^{-}(j), deg^{+}(j)$, where i is an article having ground truth recommendation and j is either true recommendation or random article. All the metrics except k_{ij}, which is the number of game transitions, are symmetric. Hence, in this case, we do not consider order in pair *(article, recommendation)*. While this is important, we leave it as a future work.

For evaluation we use standard metrics for binary classification: accuracy, precision, recall and F_1 score [29]. Accuracy is defined as a fraction of correctly predicted items. Precision and recall are defined as follows:

$$Prec = \frac{tp}{tp + fp}, \quad Rec = \frac{tp}{tp + fn}, \tag{7.1}$$

where tp is the number of correctly predicted items that belong to the positive class, fp is the number of items erroneously classified as positive, fn is the number of items erroneously classified as negative. F_1 score is defined as a harmonic mean of precision and recall, i.e.:

$$F_1 = \frac{2(Prec \cdot Rec)}{Prec + Rec} \tag{7.2}$$

We report classifier results in table 7.1. It is possible to predict the true recommendations with high accuracy. In our case, we achieved 92% precision. Moreover, recall is decent; namely, we correctly classified 76% of true recommendations.

Accuracy	85.26%
Precision	92.23%
Recall	76.39%
F_1 score	83.57%

TABLE 7.1: Results of the classifier.

Next, we analyse the coefficients obtained by the model training. Since all the features under considerations have different scale, the absolute values of coefficients are incomparable. To this end, we have features that represent similarities between pairs of articles. All the features with corresponding parameters learned by the model and their p-values are listed in Table 7.2.

Feature	Coefficient	p-value
$sim(i,j)$	0.3835	.057
$\frac{sim(i,j)}{d(i,j)^2}$	1.4650	<.001
k_{ij}	0.0096	.936
$deg^-(source)$	0.0004	.769
$deg^+(source)$	-0.0092	.018
$deg^-(candidate)$	-0.0020	.183
$deg^+(candidate)$	0.0340	<.001
r_{ij}	-4.4596	.001

TABLE 7.2: Fitted coefficient for features and their significance. p-value < .05 identifies that null hypothesis that parameter equals zero is rejected at .05 significance level. p-value estimates were obtained using Wald test.

Firstly, all the metrics considered for random walk agents except the Pearson similarity have a positive effect on the prediction, i.e. increase in these types of similarity leads to increase in the probability of the article pair to be classified as a match (i.e. one is a recommendation for other). However, a textual similarity that is not penalized by distance, as well as the number of transitions, are insignificant. It does not imply that these metrics are irrelevant for making a recommendation. This means that for this dataset, the parameters corresponding to these features are not statistically different from zero. Interesting is the fact that the increase in Pearson similarity actually decreases the probability to be a match. This can suggest that this property cannot be used for the recommendations, but more thorough analysis needs to be performed in order to make such conclusions.

Added features suggest that indegree of both articles in a pair are insignificant and that only outdegrees matter. Increase of source outdegree decreases the probability of a match, while an increase of candidate outdegree increases it. It is unclear, why indegree of the candidate is insignificant since it is natural to assume that it should increase the probability of a match because high indegree suggests that the article is popular. The size of the dataset available for training and testing can explain this.

Chapter 8

Conclusions

Navigation in information networks has been a subject of research for many years now. One of the fringes of this topic is the development of recommender systems which will assist users and enhance the process of navigation. The study conducted in this work is devoted to the recommender system for Wikipedia articles. We explored the topological properties of this network and used them for finding pages similar to the given one. To this end, we performed research on an old version of Wikipedia from 2007, which was used for the creation of the Wikispeedia game [49]. Our calculations show that Wikispeedia is a small-world network with scale-free properties. This enhances the navigation process, as shown by [21]. Among other sources, we use the data available from user navigation history in order to infer similar articles. All the results we obtained are reproducible and publicly available[1].

As was shown earlier by [51] and [10], random walks can be used in recommendation systems. We constructed a number of random walk agents, each with transition probabilities that reflect the similarity relations between articles in different ways. Among similarity measures, we considered textual similarity between the articles, topological similarity and their combinations. In contrast to random walks on lattices, our agents prove to behave differently. The main distinction is that our agents remain in the limited area of space. This leads to the suggestion that one does not need long walks in order to obtain similar articles. However, as noted in Chapter 5, long and short walks need to be compared in order to be confident which one is better. We leave it as a future work.

Similar conclusion can be made on similarities of a given node to the origin. After a few first steps, all of them reach a plateau, which is much lower than the average values for ground truth recommendations.

Given the ground truth recommendations that we obtained through parsing the 2010 version of Wikipedia, we evaluate our algorithms using two metrics. They show consistent results. Random walk agent that walks according to the number of human transitions between articles showed the best result on the evaluation set. This suggests that information encoded in human navigation trails is useful for recommendation purposes. Another interesting finding is that the agent which uniformly jumps on the current node's neighbors achieves comparable result. So the local structure of the network reflects similarities of the nodes. Analogous result was obtained in [51].

We took a step forward and identified how the agents' transition rules should be combined in order to achieve better recommendation quality. To this end, we trained a binary classifier on the task of predicting whether two articles will produce a match, given their similarities. The parameters fitted during the classifier training can be used to linearly combine the aforementioned transition rules. We evaluate the performance of the classifier using precision, recall, accuracy and F_1-score. The

[1] https://github.com/pgmvp/wiki-rec

classifier reached 85% accuracy using our small database and the limited amount of actual recommendation. Authors in [10] showed that for a similar task 10 times increase in the size of the database allowed to improve prediction accuracy from around 80% to nearly 93%. For our purpose, this can be achieved by using Wikipedia clickstream database [50]. It contains statistics on using links in Wikipedia graph. This is left for further improvement of the recommendations.

Apart from the recommendation prediction, we analyze fitted parameters and study their significance. For the dataset we obtained, not all the metrics were significant. Interesting is the fact that Pearson similarity negatively impacts the probability of recommendation.

To sum up, firstly, we believe that applying random walk methods in the context of recommendation systems can be fruitful. Secondly, Wikipedia and its navigation history contain much information that can be extracted and used for various purposes.

Appendix A

Recommendations

These are examples of recommendations of every agent. The article recommendations were made for is "British English".

A.1 Transition based

United States, Europe, Christianity, France, Earth, Mammal, Africa, Telephone, Atlantic Ocean, Latin, Music, Animal, Thailand, Mediterranean Sea, Science, California, Japan, Gold, European Union, Plant, Television, Mathematics, Cretaceous, Finland, Religion, Norway, Water, Roman Catholic Church, Chemistry, Great Britain, Telecommunication, China, Volcano, North America, Egypt, Gravitation, Romania, Whale shark, Nuclear weapon, Russia, Light, United Nations, Lion, Cold War, People's Republic of China, Astronomy, Zebra, Physics, Chemical element, Alexander Graham Bell.

A.2 Uniform

Russia, Germany, China, Latin, People's Republic of China, Christianity, France, United States, Jew, Currency, Spain, Italy, Television, Atlantic Ocean, Paris, World War II, Islam, Soviet Union, Israel, Rome, Europe, Poland, Animal, Portugal, Sun, Ice age, 20th century, Ottoman Empire, Proton, Ernest Hemingway, Communism, Roman Empire, Ming Dynasty, Fish, History, Reptile, Indonesia, Yemen, Sea, Scientific classification, Eukaryote, Agriculture, Vertebrate, Hungary, Bacteria, Afghanistan, Hydrogen, Egypt, Planet, Mathematics.

A.3 Textual similarity based, $\alpha = 0$

Whitney Joins The JAMs, Australian constitutional law, Lunar eclipse, Krypton, Volleyball, Walt Disney, Igor Stravinsky, Aesthetics, Bread, Mitochondrial Eve, Epazote, Georg von Boeselager, Tacitean studies, Hampstead Heath, Slavery, Amsterdam, Superconductivity, Linux, Open cluster, Flag of Hong Kong, Tropical Storm Bonnie (2004), Retinol, Lancia Flaminia, HD 217107, Sand, Kurt Cobain, William Shakespeare, Psittacosaurus, Cheers, Pygmy Hippopotamus, Zebra, Niger, Ship, Medicine, Morphine, Kaffir lime, Hurricane Dennis, Mercia, Romania, J. K. Rowling, Biotechnology, Forensic facial reconstruction, Star, North America, Minnesota, Rabbit, Revised Standard Version, Great Lakes, Latvia, Forth.

A.4 Textual similarity based, $\alpha = 2$

Arabic language, Charles Dickens, Gadolinium, Animal, Japan, Crime, Onion, Cocoa, Drinking water, Beirut, Currency, Ottoman Empire, Cormorant, Mauritania, Hydrochloric acid, Extinction, Thomas Aquinas, Richard Nixon, Naval Battle of Guadalcanal, Portuguese language, U2, Music, Monty Python, Detroit, Michigan, Mind, Guild, El Aaiún, Linear algebra, Drought, Ammonia, Welding, Raney nickel, J. K. Rowling, Vietnam, Finance, Photosynthesis, Buckingham Palace, Tantalum, Henry VII of England, George III of the United Kingdom, Thermodynamics, Liverpool, Isle of Man, Saint Petersburg, Costa Rica, Hernán Cortés, West Bank, Kyoto Protocol, Herbivore, Canaletto.

A.5 Pearson similarity based

Mixed-breed dog, Moldova, Music of Thailand, Education, Renminbi, Montevideo, Westport Country Playhouse, Vole, Johannes Gutenberg, Investment banking, Tea, Santamaría (volcano), Dundee United F.C., David Attenborough, Copenhagen Fire of 1728, Quatermass and the Pit, The Canadian, Henry David Thoreau, List of European countries, San Jose, California, William Hogarth, Pompeii, Lake Baikal, Tropical Storm Henri (2003), Nicobar Long-tailed Macaque, Nadia Comăneci, The Lord of the Rings, History of nuclear weapons, Faroe Islands, Avalanche, 1973 oil crisis, 19th century, Lord Voldemort, Video, Speed of light, Himalayas, Sauroposeidon, Neon, Bison, Wake Island, Thiamine, Ununpentium, Sand, Outer Hebrides, Alchemy, Anthropology, Bob Marley, Congo River, Ernest Rutherford, Snow.

Bibliography

[1] Lada Adamic and Eytan Adar. "How to search a social network". In: *Social networks* 27.3 (2005), pp. 187–203.

[2] David Aldous and James Fill. *Reversible Markov chains and random walks on graphs*.

[3] Daniar Asanov et al. "Algorithms and methods in recommender systems". In: *Berlin Institute of Technology, Berlin, Germany* (2011).

[4] Joeran Beel et al. "paper recommender systems: a literature survey". In: *International Journal on Digital Libraries* 17.4 (2016), pp. 305–338.

[5] Christopher M Bishop. *Pattern recognition and machine learning*. springer, 2006.

[6] Toine Bogers. "Movie recommendation using random walks over the contextual graph". In: *Proc. of the 2nd Intl. Workshop on Context-Aware Recommender Systems*. 2010.

[7] Svetlin Bostandjiev, John O'Donovan, and Tobias Höllerer. "TasteWeights: a visual interactive hybrid recommender system". In: *Proceedings of the sixth ACM conference on Recommender systems*. ACM. 2012, pp. 35–42.

[8] Sergey Brin and Lawrence Page. "The anatomy of a large-scale hypertextual web search engine". In: *Computer networks and ISDN systems* 30.1-7 (1998), pp. 107–117.

[9] Fan Chung and Wenbo Zhao. "PageRank and random walks on graphs". In: *Fete of combinatorics and computer science*. Springer, 2010, pp. 43–62.

[10] Colin Cooper et al. "Random walks in recommender systems: exact computation and simulations". In: *Proceedings of the 23rd International Conference on World Wide Web*. ACM. 2014, pp. 811–816.

[11] Dan Cosley et al. "SuggestBot: using intelligent task routing to help people find work in wikipedia". In: *Proceedings of the 12th international conference on Intelligent user interfaces*. ACM. 2007, pp. 32–41.

[12] *DBLP network dataset – KONECT*. Oct. 2017. URL: http://konect.cc/networks/dblp-cite.

[13] Paul Erdős and Alfréd Rényi. "On the evolution of random graphs". In: *Publ. Math. Inst. Hung. Acad. Sci* 5.1 (1960), pp. 17–60.

[14] Eugene F Fama. "The behavior of stock-market prices". In: *The journal of Business* 38.1 (1965), pp. 34–105.

[15] Christian von Ferber et al. *From Brownian motion to self-avoiding walks and Lévy flights*. 2013.

[16] Mustansar Ali Ghazanfar, Adam Prügel-Bennett, and Sandor Szedmak. "Kernel-mapping recommender system algorithms". In: *Information Sciences* 208 (2012), pp. 81–104.

[17] Xiangnan He et al. "Neural collaborative filtering". In: *Proceedings of the 26th international conference on world wide web*. International World Wide Web Conferences Steering Committee. 2017, pp. 173–182.

[18] Jocelyn Hickcox and Chris Min. "Customized Random Walk for Generating Wikipedia Article Recommendations". In: (2015).

[19] Yu Holovatch et al. "Complex networks". In: *Journal of Physical Studies* 10 (2006), pp. 247–289.

[20] Gilad Katz et al. "Using Wikipedia to boost collaborative filtering techniques". In: *Proceedings of the fifth ACM conference on Recommender systems*. ACM. 2011, pp. 285–288.

[21] Jon M Kleinberg. "Navigation in a small world". In: *Nature* 406.6798 (2000), p. 845.

[22] Jon M Kleinberg. "Small-world phenomena and the dynamics of information". In: *Advances in neural information processing systems*. 2002, pp. 431–438.

[23] Yehuda Koren, Robert Bell, and Chris Volinsky. "Matrix factorization techniques for recommender systems". In: *Computer* 8 (2009), pp. 30–37.

[24] Jérôme Kunegis. "KONECT – The Koblenz Network Collection". In: *Proc. Int. Conf. on World Wide Web Companion*. 2013, pp. 1343–1350. URL: http://dl.acm.org/citation.cfm?id=2488173.

[25] Daniel Lamprecht et al. "How the structure of wikipedia articles influences user navigation". In: *New Review of Hypermedia and Multimedia* 23.1 (2017), pp. 29–50.

[26] Michael Ley. "The DBLP Computer Science Bibliography: Evolution, Research Issues, Perspectives". In: *Proc. Int. Symposium on String Process. and Inf. Retr.* 2002, pp. 1–10.

[27] Pasquale Lops, Marco De Gemmis, and Giovanni Semeraro. "Content-based recommender systems: State of the art and trends". In: *Recommender systems handbook*. Springer, 2011, pp. 73–105.

[28] R Duncan Luce and Albert D Perry. "A method of matrix analysis of group structure". In: *Psychometrika* 14.2 (1949), pp. 95–116.

[29] Christopher Manning, Prabhakar Raghavan, and Hinrich Schütze. "Introduction to information retrieval". In: *Natural Language Engineering* 16.1 (2010), pp. 117–123.

[30] RV Mises and Hilda Pollaczek-Geiringer. "Praktische Verfahren der Gleichungsauflösung." In: *ZAMM-Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik* 9.2 (1929), pp. 152–164.

[31] M. E. J. Newman. *Networks: an introduction*. Oxford; New York: Oxford University Press, 2010. ISBN: 9780199206650 0199206651. URL: http://www.amazon.com/Networks-An-Introduction-Mark-Newman/dp/0199206651/ref=sr_1_5?ie=UTF8&qid=1352896678&sr=8-5&keywords=complex+networks.

[32] Mark EJ Newman. "Mixing patterns in networks". In: *Physical Review E* 67.2 (2003), p. 026126.

[33] Thomas Niebler et al. "Extracting semantics from unconstrained navigation on wikipedia". In: *KI-Künstliche Intelligenz* 30.2 (2016), pp. 163–168.

[34] Arkadiusz Paterek. “Improving regularized singular value decomposition for collaborative filtering”. In: *Proceedings of KDD cup and workshop*. Vol. 2007. 2007, pp. 5–8.

[35] Karl Pearson. “The problem of the random walk”. In: *Nature* 72.1867 (1905), p. 342.

[36] Dragomir R Radev et al. “Evaluating Web-based Question Answering Systems.” In: *LREC*. 2002.

[37] Francesco Ricci, Lior Rokach, and Bracha Shapira. “Introduction to recommender systems handbook”. In: *Recommender systems handbook*. Springer, 2011, pp. 1–35.

[38] Giovanna Chiara Rodi, Vittorio Loreto, and Francesca Tria. “Search strategies of Wikipedia readers”. In: *PloS one* 12.2 (2017), e0170746.

[39] Tuukka Ruotsalo et al. “SMARTMUSEUM: A mobile recommender system for the Web of Data”. In: *Web semantics: Science, services and agents on the world wide web* 20 (2013), pp. 50–67.

[40] Gerard Salton and Michael J McGill. “Introduction to modern information retrieval”. In: (1986).

[41] Aju Thalappillil Scaria et al. “The last click: Why users give up information network navigation”. In: *Proceedings of the 7th ACM international conference on Web search and data mining*. ACM. 2014, pp. 213–222.

[42] Xiaoyuan Su and Taghi M Khoshgoftaar. “A survey of collaborative filtering techniques”. In: *Advances in artificial intelligence* 2009 (2009).

[43] Mikhail V Tamm and Kirill Polovnikov. “Dynamics of polymers: classic results and recent developments”. In: *arXiv preprint arXiv:1707.09885* (2017).

[44] Christoph Trattner et al. “Exploring the differences and similarities between hierarchical decentralized search and human navigation in information networks”. In: *Proceedings of the 12th international conference on knowledge management and knowledge technologies*. ACM. 2012, p. 14.

[45] Ellen M Voorhees et al. “The TREC-8 question answering track report”. In: *Trec*. Vol. 99. Citeseer. 1999, pp. 77–82.

[46] Duncan J Watts and Steven H Strogatz. “Collective dynamics of ‘small-world’networks”. In: *nature* 393.6684 (1998), p. 440.

[47] Robert West and Jure Leskovec. “Automatic versus human navigation in information networks”. In: *Sixth International AAAI Conference on Weblogs and Social Media*. 2012.

[48] Robert West and Jure Leskovec. “Human wayfinding in information networks”. In: *Proceedings of the 21st international conference on World Wide Web*. ACM. 2012, pp. 619–628.

[49] Robert West, Joelle Pineau, and Doina Precup. “Wikispeedia: An online game for inferring semantic distances between concepts”. In: *Twenty-First International Joint Conference on Artificial Intelligence*. 2009.

[50] Ellery Wulczyn and Dario Taraborelli. “Wikipedia Clickstream”. In: (Feb. 2017). DOI: 10.6084/m9.figshare.1305770.v22. URL: https://figshare.com/articles/Wikipedia_Clickstream/1305770.

[51] Eric Yeh et al. "WikiWalk: random walks on Wikipedia for semantic relatedness". In: *Proceedings of the 2009 workshop on graph-based methods for natural language processing*. Association for Computational Linguistics. 2009, pp. 41–49.