2

Theoretical Backgrounds

As discussed, Web data involves in a complex structure and heterogeneous nature. The analysis on the Web data needs a broad range of concepts, theories and approaches and a variety of application backgrounds. In order to help readers to better understand the algorithms and techniques introduced in the book, it is necessary to prepare some basic and fundamental background knowledge, which also forms a solid theoretical base for this book. In this chapter, we first present some theoretical backgrounds and review them briefly.

We first give an introduction of Web data models, particularly the data expressions of textual, linkage and usage. Then the basic theories of linear algebra especially the operations of matrix and tensor are discussed. The two essential concepts and approaches in Information Retrieval - similarity measures and evaluation metrics, are summarized as well. In addition, some basic concepts of social networks are addressed in this chapter.

2.1 Web Data Model

It is well known that the Internet has become a very popular and powerful platform to store, disseminate and retrieve information as well as a data respiratory for knowledge discovery. However, Web users always suffer the problems of information overload and drowning due to the significant and rapid growth in amount of information and the number of users. The problems of low precision and low recall rate caused by above reasons are two major concerns that users have to deal with while searching for the needed information over the Internet. On the other hand, the huge amount of data/information resided over the Internet contains very valuable informative knowledge that could be discovered via advanced data mining approaches. It is believed that mining this kind of knowledge will greatly benefit Web site design and Web application development, and prompt other related applications, such as business intelligence, e-Commerce, and entertainment broadcast etc. Thus, the emerging of Web has put forward a great deal of challenges to Web researchers for Web-based

G. Xu et al., *Web Mining and Social Networking,*
DOI 10.1007/978-1-4419-7735-9_2, © Springer Science+Business Media, LLC 2011

information management and retrieval. Web researcher and engineer are requested to develop more efficient and effective techniques to satisfy the demands of Web users.

Web data mining is one kind of these techniques that efficiently handle the tasks of searching needed information from the Internet, improving Web site structure to improve the Internet service quality and discovering informative knowledge from the Internet for advanced Web applications. In principle, Web mining techniques are the means of utilizing data mining methods to induce and extract useful information from Web information and service. Web mining research has attracted a variety of academics and researchers from database management, information retrieval, artificial intelligence research areas especially from knowledge discovery and machine learning, and many research communities have addressed this topic in recent years due to the tremendous growth of data contents available on the Internet and urgent needs of e-commerce applications especially. Dependent on various mining targets, Web data mining could be categorized into three types of Web content, Web structure and Web usage mining. In the following chapters, we will systematically present the research studies and applications carried out in the context of Web content, Web linkage and Web usage mining

To implement Web mining efficiently, it is essential to first introduce a solid mathematical framework, on which the data mining/analysis is performed. There are many types of data expressions could be used to model the co-occurrence of interactions between Web users and pages, such as matrix, directed graph and click sequence and so on. Different data expression models have different mathematical and theoretical backgrounds, and therefore resulting in various algorithms and approaches. In particular, we mainly adopt the commonly used matrix expression as the analytic scheme, which is widely used in various Web mining context. Under this scheme, the interactive observations between Web users and pages, and the mutual relationships between Web pages are modeled as a co-occurrence matrix, such as in the form of page hyperlink adjacent (inlink or outlink) matrix or session-pageview matrix. Based on the proposed mathematical framework, a variety of data mining and analysis operations can be employed to conduct Web mining.

2.2 Textual, Linkage and Usage Expressions

As described, the starting point of Web mining is to choose appropriate data models. To achieve the desired mining tasks discussed above, there are different Web data models in the forms of feature vectors, engaged in pattern mining and knowledge discovery. According to the three identified categories of Web mining, three types of Web data/sources, namely content data, structure data and usage data, are mostly considered in the context of Web mining. Before we start to propose different Web data models, we firstly give a brief discussion on these three data types in the following paragraphs.

Web content data is a collection of objects used to convey content information of Web pages to users. In most cases, it is comprised of textural material and other types of multimedia content, which include static HTML/XML pages, images, sound

and video files, and dynamic pages generated from scripts and databases. The content data also includes semantic or structured meta-data embedded within the site or individual pages. In addition, the domain ontology might be considered as a complementary type of content data hidden in the site implicitly or explicitly. The underlying domain knowledge could be incorporated into Web site design in an implicit manner, or be represented in some explicit forms. The explicit form of domain ontology can be conceptual hierarchy e.g. product category, and structural hierarchy such as yahoo directory etc [206].

Web structure data is a representation of linking relationship between Web pages, which reflects the organizational concept of a site from the viewing point of the designer [119]. It is normally captured by the inter-page linkage structure within the site, which is called linkage data. Particularly, the structure data of a site is usually represented by a specific Web component, called "site map", which is generated automatically when the site is completed. For dynamically generated pages, the site mapping is becoming more complicated to perform since more techniques are required to deal with the dynamic environment.

Web usage data is mainly sourced from Web log files, which include Web server access logs and application server logs [234, 194]. The log data collected at Web access or application servers reflects the navigational behavior knowledge of users in terms of access pattern. In the context of Web usage mining, usage data that we need to deal with is transformed and abstracted at different levels of aggregations, namely Web page set and user session collection. Web page is a basic unit of Web site organization, which contains a number of meaningful units serving for the main functionality of the page. Physically, a page is a collection of Web items, generated statically or dynamically, contributing to the display of the results in response to a user request. A page set is a collection of whole pages within a site. User session is a sequence of Web pages clicked by a single user during a specific period. A user session is usually dominated by one specific navigational task, which is exhibited through a set of visited relevant pages that contribute greatly to the task conceptually. The navigational interest/preference on one particular page is represented by its significance weight value, which is dependent on user visiting duration or click number. The user sessions (or called usage data), which are mainly collected in the server logs, can be transformed into a processed data format for the purpose of analysis via data preparing and cleaning process. In one word, usage data is a collection of user sessions, which is in the form of a weighted vector over the page space.

Matrix expression has been widely used to model the co-occurrence activity like Web data. The illustration of a matrix expression for Web data is shown in Fig.2.1. In this scheme, the rows and columns correspond to various Web objects which are dependent on various Web data mining tasks. In the context of Web content mining, the relationships between a set of documents and a set of keyword could be represented by a document-keyword co-occurrence matrix, where the lows of the matrix represent the documents, while the columns of the matrix correspond to the keywords. The intersection value of the matrix indicates the occurrence of a specific keyword appeared in a particular document, i.e. if a keyword appears in a document, the corresponding matrix element value is 1, otherwise 0. Of course, the element value could

also be a precise weight rather than 1 or 0 only, which exactly reflects the occurrence degree of two concerned objects of document and keyword. For example, the element value could represent the frequent rate of a specific keyword in a specific document. Likewise, to model the linkage information of a Web site, an adjacent matrix is used to represent the relationships between pages via their hyperlinks. And usually the element of the adjacent matrix is defined by the hyperlink linking two pages, that is, if there is a hyperlink from page i to page j ($i \neq j$), then the value of the element a_{ij} is 1, otherwise 0. Since the linking relationship is directional, i.e. given a hyperlink directed from page i to page j, then the link is an out-link for i, while an in-link for j, and vice versa. In this case, the ith row of the adjacent matrix, which is a page vector, represents the out-link relationships from page i to other pages; the jth column of the matrix represents the in-link relationships linked to page i from other pages.

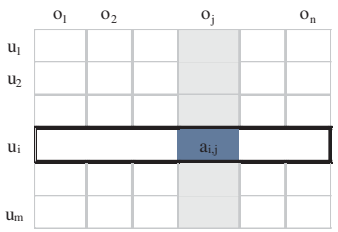


Fig. 2.1. The schematic illustration of Web data matrix model

In Web usage mining, we can model one user session as a page vector in a similar way. As the user access interest exhibited may be reflected by the varying degree of visits on different Web pages during one session, we can represent a user session as a collection of pages visited in the period along with their significant weights. The total collection of user sessions can, then, be expressed a usage matrix, where the ith row is the sequence of pages visited by user i during this period; and the jth column of the matrix represents the fact which users have clicked this page j in the server log file. The element value of the matrix, a_{ij}, reflects the access interest exhibited by user i on page j, which could be used to derive the underlying access pattern of users.

2.3 Similarity Functions

A variety of similarity functions can be used as measuring metrics in vector space. Among these measures, Pearson correlation coefficient and cosine similarity are two well-known and widely used similarity functions in information retrieval and recommender systems [218, 17].

2.3.1 Correlation-based Similarity

Pearson correlation coefficient, which is to calculate the deviations of users' ratings on various items from their mean ratings on the rated items, is a commonly used similarity function in traditional collaborative filtering approaches, where the attribute weight is expressed by a feature vector of numeric ratings on various items, e.g. the rating can be from 1 to 5 where 1 stands for the lest like voting and 5 for the most preferable one. The Pearson correlation coefficient can well deal with collaborative filtering since all ratings are on a discrete scale rather than on an analogous scale. The measure is described below. Given two users i and j, and their rating vectors R_i and R_j, the Pearson correlation coefficient is then defined by:

$$sim(i,j) = corr(R_i, R_j) = \frac{\sum_{k=1}^{n} \left(R_{i,k} - \overline{R_i}\right)\left(R_{j,k} - \overline{R_j}\right)}{\sqrt{\sum_{k=1}^{n} \left(R_{i,k} - \overline{R_i}\right)^2 \sum_{k=1}^{n} \left(R_{j,k} - \overline{R_j}\right)^2}} \tag{2.1}$$

where $R_{i,k}$ denotes the rating of user i on item k, $\overline{R_i}$ is the average rating of user i.

However, this measure is not appropriate in the Web mining scenario where the data type encountered (i.e. user session) is actually a sequence of analogous page weights. To address this intrinsic property of usage data, the cosine coefficient is a better choice instead, which is to measure the cosine function of angle between two feature vectors. Cosine function is widely used in information retrieval research.

2.3.2 Cosine-Based Similarity

Since in a vector expression form, any vector could be considered as a line in a multiple-dimensional space, it is intuitive to define the similarity (or distance) between two vectors as the cosine function of angle between two "lines". In this manner, the cosine coefficient can be calculated by the ratio of the dot product of two vectors with respect to their vector norms. Given two vectors A and B, the cosine similarity is then defined as:

$$sim(A,B) = \cos\left(\overrightarrow{A}, \overrightarrow{B}\right) = \frac{\overrightarrow{A} \cdot \overrightarrow{B}}{\left|\overrightarrow{A}\right| \times \left|\overrightarrow{B}\right|} \tag{2.2}$$

where "$\cdot$" denotes the dot operation and "$\times$" the norm form.

2.4 Eigenvector, Principal Eigenvector

In linear algebra, there are two kinds of objects: scalars, which are just numbers, and vectors, which can be considered as arrows in a space, and which have both magnitude and direction (though more precisely a vector is a member of a vector space). In the context of traditional functions of algebra, the most important functions

in linear algebra are called "linear transformations", and particularly in the context of vector, a linear transformation is usually given by a "matrix", a multi-array of numbers. In order to avoid the confusion in mathematical expression, here the linear transformation of matrix is denoted by $\mathbf{M}(\mathbf{v})$ instead of $f(x)$ where $\mathbf{M}$ is a matrix and $\mathbf{v}$ is a vector.

If A is an n-by-n matrix, then there is a scalar number λ for $\mathbf{A}$ and a nonzero vector $\mathbf{v}$ (called an eigenvector for A associated to λ) so that $\mathbf{Av} = \lambda\mathbf{v}$. The eigenspace corresponding to one eigenvalue of a given matrix is the set of all eigenvectors of the matrix with that eigenvalue.

$$E(\lambda) = \{\mathbf{v} : \mathbf{Av} = \lambda\mathbf{v}\} = \{\mathbf{v} : (\mathbf{A} - \lambda\mathbf{I})\mathbf{v} = 0\} \tag{2.3}$$

The basic equation $\mathbf{Av} = \lambda\mathbf{v}$ can be rewritten as $(\mathbf{A} - \lambda\mathbf{I})\mathbf{v} = 0$. For a given λ, its eigenvectors are nontrivial solutions of $(\mathbf{A} - \lambda\mathbf{I})\mathbf{v} = 0$. When this equation is regarded as an equation in the variable λ, it becomes a polynomial of degree n in λ. Since a polynomial of degree n has at most n distinct answers, this could be transformed to a solving process of at most n eigenvalues for a given matrix. The eigenvalues are arranged in a ordered sequence, and the largest eigenvalue of a matrix is called the principal eigenvalue of the given matrix. Particularly, in some specific applications of matrix decomposition with eigenvalue like in *Principal Component Analysis* (PCA) or *Singular Value Decomposition* (SVD), some eigenvalues after a certain position in the ordered eigenvalue sequence are decreased to very small values such that they are truncated by that certain position and discarded. Then the remaining eigenvalues are left together to form an estimated fraction of matrix decomposition. This estimate is then used to reflect the correlation criterion of approximation of the row and column attributes. In case that eigenvalues are known, they could be used to compute the eigenvector of the matrix, which is also called latent vectors of matrix $\mathbf{A}$.

Eigenvalues and eigenvectors are widely used in a variety of applications that involve in the computation of matrix. In spectral graph theory, for example, an eigenvalue of a graph is defined as an eigenvalue of the graph's adjacency matrix $\mathbf{A}$, or of the graph's Laplacian matrix, which is either $\mathbf{T} - \mathbf{A}$ or $\mathbf{I} - \mathbf{T}^{-1/2}\mathbf{A}\mathbf{T}^{1/2}$, where T is a diagonal matrix holding the degree of each vertex. The kth principal eigenvector of a graph is defined as either the eigenvector corresponding to the kth largest eigenvalue of $\mathbf{A}$, or the eigenvector corresponding to the kth smallest eigenvalue of the Laplacian matrix. The first principal eigenvector of the graph is also referred to as the principal eigenvector. In spectral graph applications, principal eigenvectors are usually used to measure the significance of vertices in the graph. For example, in Google's PageRank algorithm, the principal vector is used to calculate the centrality (i.e. hub or authority score) of nodes if the websites over the Internet are modeled as a complete directed graph. Another application is that the second smallest eigenvector can be used to partition the graph into clusters via spectral clustering.

In summary, given the operation of a matrix performed on a (nonzero) vector changing its magnitude but not its direction, then the vector is called an eigenvector of that matrix. The scalar which is used to complete the operation by multiplying the eigenvector is called the eigenvalue corresponding to that eigenvector. For a given

matrix, there exist many eigenvalues, each of them could be used to calculate the eigenvectors.

2.5 Singular Value Decomposition (SVD) of Matrix

The standard LSI algorithm is based on SVD operation. The SVD definition of a matrix is illustrated as follows [69]: For a real matrix $A = [a_{ij}]_{m\times n}$, without loss of generality, suppose $m \geq n$, there exists a SVD of A (shown in Fig.2.2)

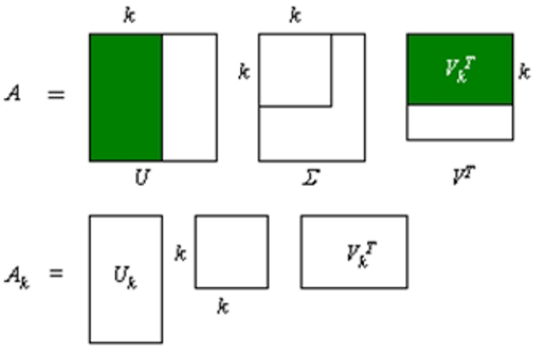


Fig. 2.2. Illustration of SVD approximation

$$A = U\begin{pmatrix}\Sigma_1\\0\end{pmatrix}V^T = U_{m\times m}\sum_{m\times n}V^T_{n\times n} \tag{2.4}$$

where U and V are orthogonal matrices $U^TU = I_m$, $V^TV = I_n$. Matrices U and V can be respectively denoted as $U_{m\times m} = [u_1, u_2, \cdots u_m]_{m\times m}$ and $V_{n\times n} = [v_1, v_2, \cdots v_n]_{n\times n}$, where u_i, $(i = 1, \cdots, m)$ is a m-dimensional vector $u_i = (u_{1i}, u_{2i}, \cdots u_{mi})^T$ and v_j, $(j = 1, \cdots, n)$ is a n-dimensional vector $v_j = (v_{1j}, v_{2j}, \cdots v_{nj})^T$. Suppose $rank(A) = r$ and the single values of A are diagonal elements of σ as follows:

$$\sum = \begin{bmatrix}\sigma_1 & 0 & \cdots & 0\\ 0 & \sigma_2 & \ddots & \vdots\\ \vdots & \ddots & \ddots & 0\\ 0 & \cdots & 0 & \sigma_n\end{bmatrix} = diag(\sigma_1, \sigma_2, \cdots \sigma_m) \tag{2.5}$$

where $\sigma_i \geq \sigma_{i+1} > 0$, for $1 \leq i \leq r-1; \sigma_j = 0$, for $j \geq r+1$, that is $\sigma_1 \geq \sigma_2 \geq \cdots \sigma_r \geq \sigma_{r+1} = \cdots = \sigma_n = 0$

For a given threshold $\varepsilon (0 < \varepsilon < 1$, choose a parameter k such that $(\sigma_k - \sigma_{k+1})/\sigma_k \geq \varepsilon$. Then, denote $U_k = [u_1, \cdots, u_k]_{m\times k}$, $V_k = [v_1, \cdots, v_k]_{n\times k}$, $\sum_k = diag(\sigma_1, \cdots, \sigma_k)$, and $A_k = U_k \sum_k V_k^T$.

As known from the theorem in algebra [69], A_k is the best approximation matrix to A and conveys the maximum latent information among the processed data. This property makes it possible to find out the underlying semantic association from original feature space with a dimensionality-reduced computational cost, in turn, is able to be used for latent semantic analysis.

2.6 Tensor Expression and Decomposition

In this section, we will discuss the basic concepts of tensor, which is a mathematical expression in a multi-dimensional space. As seen in previous sections, matrix is an efficient means that could be used to reflect the relationship between two types of subjects. For example, the author-article in the context of scientific publications or document-keyword in applications of digital library. No matter in which scenario the common characteristics is the fact which each row is a linear combination of values along different column or each column is represented by a vector of entries in row space. Matrix-based computing possesses the powerful capability to handle the encountered problem in most real life problems since sometimes it is possible to model these problems as two-dimensional problems. But in a more complicated sense, while matrices have only two "dimensions" (e.g., "authors" and "publications"), we may often need more, like "authors", "keywords", "timestamps","conferences". This is exactly a high-order problem, which, in fact, is generally a tensor represents. In short, from the perspective of data model, tensor is a generalized and expressive model of high-dimensional space, and of course, a tensor is a generalization of a matrix (and of a vector, and of a scalar). Thus, it is intuitive and necessary to envision all such problems as tensor problems, to use the vast existing work for tensors to our benefit, and to adopt tensor analysis tools into our interested research arenas. Below we discuss the mathematical notations of tensor related concepts and definitions.

First of all, we introduce some fundamental terms in tensor which have different meanings in the context of two-dimensional cases. In particular we use order, mode and dimension to denote the equivalent concepts of dimensionality, dimension and attribute value we often encounter and use in linear algebra. For example a 3rd-order tensor means a three-dimensional data expression. To use the distinctive mathematical symbols to denote the different terms in tensor, we introduce the following notations:

- Scalars are denoted by lowercase letter, a.
- Vectors are denoted by boldface lowercase letters, $\mathbf{a}$. The ith entry of $\mathbf{a}$ is denoted by $\mathbf{a}_i$.
- Matrices are denoted by boldface capital letters, e.g., $\mathbf{A}$. The jth column of $\mathbf{A}$ is donated by $\mathbf{a}_j$ and element by a_{ij}.
- Tensors, in multi-way arrays, are denoted by italic boldface letters, e.g., X. Element (i,j,k) of a 3rd-order tensor X is denoted by X_{ijk}.
- As known, a tensor of order M closely resembles a Data Cube with M dimensions. Formally, we write an Mth order tensor $\mathsf{X} \in R^{N_1 \times N_2 \times \cdots N_m}$, where N_i,

$(1 \leq i \leq M)$ is the dimensionality of the ith mode . For brevity, we often omit the subscript $[N_1, \cdots, N_M]$.

Furthermore, from the tensor literature we need the following definitions [236]:

Definition 2.1. *(Matricizing or Matrix Unfolding) [236]. The mode-d matricizing or matrix unfolding of an Mth order tensor* $\mathsf{X} \in R^{N_1 \times N_2 \times \cdots N_m}$ *are vectors in* R^{N_d} *obtained by keeping index d fixed and varying the other indices. Therefore, the mode-d matricizing* $\mathsf{X}_{(d)}$ *is in* $R^{\prod_{i \neq d}(N_i) \times N_d}$.

Definition 2.2. *(Mode Product)[236]. The mode product* $\mathsf{X} \times_d U$ *of a tensor* $\mathsf{X} \in R^{N_1 \times N_2 \times \cdots N_m}$ *and a matrix* $\mathbf{U} \in R^{N_d \times N^{'}}$ *is the tensor in* $R^{N_1 \times \cdots \times N_{d-1} \times N^{'} \times N_{d+1} \times \cdots \times N_M}$ *defined by:*

$$\mathsf{X} \times_d \mathbf{U}(i_1, \ldots, i_{d-1}, j, i_{d+1}, \ldots, i_M) = \sum_{i_d=1}^{N_i} \mathsf{X}(i_1, \ldots, i_{d-1}, i_d, i_{d+1}, \ldots, i_M)\mathbf{U}(i_d, j) \tag{2.6}$$

for all index values.

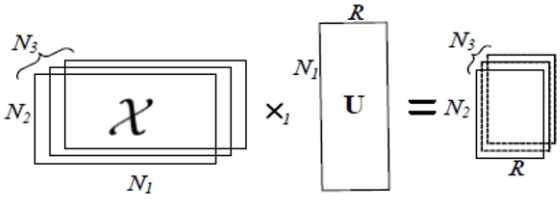


Fig. 2.3. An example of multiplication of a 3rd-order tensor with a matrix

Figure 2.3 shows an example of 3rd order tensor X mode-1 multiplies a matrix $\mathbf{U}$. The process consists of three operations: first matricizing X along mode-1, then doing matrix multiplication of $\times_1$ and $\mathbf{U}$, finally folding the result back as a tensor.

Upon definition 2.1, we can perform a series of multiplications of a tensor $\mathsf{X} \in R^{N_1 \times N_2 \times \cdots N_m}$ and $U_i\big|_{i=1}^{M} \in R^{N_i \times D_i}$ as: $\mathsf{X} \times_1 U_1 \ldots \times_m U_M \in R^{D_1 \times \cdots \times D_M}$, which can be written as $\mathsf{X} \prod_{i=1}^{M} \times_i U_i$ for clarity. Furthermore, we express the following multiplications of all $\mathbf{U}_j$ except the i-th i.e. $\mathsf{X} \times_1 U_1 \cdots \times_{i-1} U_{i-1} \times_{i+1} U_{i+1} \cdots \times_M U_M$ as $\mathsf{X} \prod_{j \neq i} \times_j U_j$.

Definition 2.3. *(Rank-*$(R_1, \cdots, R_M)$ *approximation). Given a tensor* $\mathsf{X} \in R^{N_1 \times \cdots N_M}$*, its best Rank-*$D_1, \cdots, D_M$ *approximation is the tensor* $\tilde{\mathsf{X}} \in R^{D_1 \times \cdots D_M}$ *with* $\operatorname{rank}\left(\tilde{X}_{(d)}\right) = D_d$ *for* $1 \leq d \leq M$*, which satisfies the optimal criterion of least-square-error, i.e.* $\arg\min \left\|X - \tilde{X}\right\|_F^2$.

The best Rank-$(R_1, \cdots, R_M)$ approximation is $\tilde{X} = Y \prod_{j=1}^{M} \times_j U_j$, where the tensor Y is the core tensor of approximation, $Y \in R^{N_1 \times \cdots \times N_M}$ and $U_j \Big|_{j=1}^{M} \in R^{N_j \times D_j}$ is the projection matrices.

2.7 Information Retrieval Performance Evaluation Metrics

An information retrieval process begins when a user enters a query into the system. A query is a collection of keywords that represent the information needs of the user, for example search terms in Web search engines. In information retrieval a query does not uniquely identify a single object in the information repository. Instead, several objects may match the query, perhaps with different degrees of relevancy. Each information piece is crawled from the Web and stored in the repository, i.e. database with an index or metadata in the IR systems.

Most IR systems compute a numeric score on how well each object in the database matches the query, and rank the objects according to this value. The ranked results are then returned to the user for browsing. Therefore, using various matching and ranking mechanisms results in totally different search results, in turn, arising a great challenging in evaluating the performance of IR systems [17].

2.7.1 Performance measures

Many different measures for evaluating the performance of information retrieval systems have been proposed. Apparently the measures require a collection of documents and a query. All common measures described here assume a ground truth notion of relevancy: every document is known to be either relevant or non-relevant to a particular query [2].

Precision

Precision is the fraction of the documents retrieved that are relevant to the user's information need.

$$\text{precision} = \frac{|\{\text{retrieved documents}\}| \cap |\{\text{relevant documents}\}|}{|\{\text{retrieved documents}\}|} \tag{2.7}$$

In binary classification, precision is analogous to positive predictive value. Precision takes all retrieved documents into account. It means how many percentages of retrieved documents are relevant to the query.

Recall

Recall is the fraction of the documents that are relevant to the query that are successfully retrieved.

$$\text{recall} = \frac{|\{\text{retrieved documents}\}| \cap |\{\text{relevant documents}\}|}{|\{\text{relevant documents}\}|} \tag{2.8}$$

In binary classification, recall is called sensitivity. So it can be considered as the how many percentages of relevant documents are correctly retrieved by the query.

F-measure

The traditional F-measure or balanced F-score is defined by taking both of precision and recall into account:

$$F = \frac{2 \cdot precision \cdot recall}{precision + recall} \tag{2.9}$$

This is also sometimes known as the F_1 measure, because recall and precision are evenly weighted. An ideal search mechanism or IR system requires finding as many relevant documents from all relevant existing in the repository as possible, but sometimes it is not easy to achieve the optimal results simultaneously. Thus F-measure (or sometimes called F-score) gives an overall performance indicator of the system.

Mean Average Precision (MAP)

Precision and recall are single-value metrics based on the whole list of documents returned by the system. For systems that return a ranked sequence of documents, it is desirable to also consider the order in which the returned documents are presented. Average precision reflects ranking relevant documents higher. It is the average of precisions computed at the point of each of the relevant documents in the ranked sequence:

$$MAP = \frac{\sum_{n=1}^{N} P@(n) \cdot rel(n)}{|\text{relevant documents}|} \tag{2.10}$$

where n is the rank, N the number retrieved, $rel(n)$ a binary function indicating the relevance of page at the given rank n, and $P@(n)$ precision at a given cut-off rank that defined: $P(n) = \frac{|\text{relevant retrieved documents at or below rank n}|}{n}$

Discounted Cumulative Gain (DCG)

In the cases of using a graded relevance scale of documents rather than a binary value of relevance in a search engine result set, the above metrics could not effectively measure the performance required. To deal with this, in particular, another evaluation quantity i.e. *Discounted Cumulative Gain* (DCG) is proposed to measure the usefulness, or gain, of a document based on its position in the result list. The gain is accumulated cumulatively from the top of the result list to the bottom with the gain of each result discounted at lower ranks.

The premise of DCG is that highly relevant documents should ideally appear higher in a search result list to achieve the bigger accumulated gain. Otherwise it

would be penalized as the graded relevance value is reduced logarithmically proportional to the position of the result. The discounted CG accumulated at a particular rank position p is defined as [123]:

$$DCG_p = rel_1 + \sum_{l=2}^{p} \frac{rel_i}{\log_2 i} \tag{2.11}$$

2.7.2 Web Recommendation Evaluation Metrics

There are a number of evaluation measures for Web recommendation, here we list three metrics mostly mentioned in the context of Web usage mining and Web recommendation - Mean Absolute Error, Weighted Average Visit Percentage and Hit Ratio.

Mean Absolute Error (MAE) is a widely used metric in the context of recommender systems, which compares the numerical recommendation scores against the actual user rating for the user-item pair in the test dataset [222]. *MAE* considers the deviation of recommendations against the true user-rating scores and calculates the average of the deviation sum. Given a prediction score p_i for an item with a actual rating score q_i by the user, *MAE* is expressed by

$$MAE = \frac{\sum_{i=1}^{N} |p_i - q_i|}{N} \tag{2.12}$$

The lower the MAE, the more accurately the recommendation systems perform.

The second metric, named the *Weighted Average Visit Percentage* (WAVP) [184], is used to evaluate the quality of user profiling in recommendation systems [258]. This evaluation method is based on assessing each user profile individually according to the likelihood that a user session which contains any pages in the session cluster will include the rest pages in the cluster during the same session. The calculating procedure of $WAVP$ metric is discussed as follows: suppose T is one of transaction set within the evaluation set, and for s specific cluster C, let T_c denote a subset of T whose elements contain at least one pageview from C. Moreover, the weighted average visit percentage of T_c may conceptually be determined by the similarity between T_c and the cluster C if we consider the T_c and C as in the form of pageview vector. Therefore, the $WAVP$ is computed as:

$$WAVP = \Big(\sum_{t \in T_c} \frac{t \cdot C}{|T_c|}\Big) \Big/ \Big(\sum_{p \in SCL} wt(p, SCL)\Big) \tag{2.13}$$

where SCL represents the use session cluster, $wt(p, SCL)$ is the weight of page p in session cluster SCL. More details refers to [258].

From the definition of $WAVP$, it is known that the higher the $WAVP$ value, the better quality of the obtained user profiles is.

The third metric is called *hit ratio* [128] that measures the effectiveness in the context of top-N Web recommendation. Given a user session in the test set, we extract the first j pages as an active session to generate a top-N recommendation set. Since

the recommendation set is in a descending order, we then obtain the rank of $j+1$ page in the sorted recommendation list. Furthermore, for each rank $r > 0$, we sum the number of test data that exactly rank the rth as $Nb(r)$. Let $S(r) = \sum_{i=1}^{r} Nb(i)$, and the hit ratio hr is defined as

$$hr = S(N)/|T| \tag{2.14}$$

where $|T|$ represents the number of testing data in the whole test set.

Thus, hr stands for the hit precision of Web recommendation process. Apparently, the hr is monotonically increasing with the number of recommendation - the bigger value of N (number of recommendations) the more accurate recommendation is.

2.8 Basic Concepts in Social Networks

2.8.1 Basic Metrics of Social Network

In the context of social network analysis, some basic concepts or metrics are well used to analyze the connections or interactions between vertexes in the network. Many measures are defined from the perspectives of psychology, sociology and behavior science. Even with more powerful computing paradigms are introduced into social network analysis, they still form a solid foundation for advanced social network analysis. Let's first look at some of them.

Size

Size means the number of vertexes presented in a network, and is essential to calculate other measures. This parameter can give us a general idea of how large the network is.

Centrality

Centrality gives a rough indication of the social power of a vertex in the network based on how well it impacts the network. "Betweenness", "Closeness" and "Degree" are all measures of centrality.

Density and Degree

In real social networks, the fully connected network is rarely happened instead, the less connected network is more often. In order to measure how dense the vertexes in the network are associated, *Density* measure is introduced. Given a network consisting of n vertexes, the total edges or ties between all possible vertexes are in a size of $n \times (n-1)$. Hence *Density* denotes the ratio of the number of all existing edges to the total possible number of edges in the network. *Degree* is the actual number of edges contained in the network.

Betweenness and Closeness

Betweenness and *Closeness* are both the magnitude measures that reflect the relationships of one vertex to the others in the network.

Clique

Clique represents, in the context of social network, a sub-set of a network in which vertexes are more closely connected to one another than to other members of the network. In some extents, clique is a similar concept to community, which means the members within the same group have a high similarity in some aspects, such as cultural or religious belief, interests or preferences and so on. The clique membership gives us a measure of how likely one vertex in the network belongs to a specific clique or community.

2.8.2 Social Network over the Web

Interactions and relationships between entities can be represented with an interconnected network or graph, where each node represents an entity and each link represents an interaction or a relationship between a pair of entities. Social network analysis is interested in studying social entities (such as people in an organization, called actors) and their interactions and relationships by using their network or graph representations.

The World Wide Web can be thought of as a virtual society or a virtual social network, where each page can be regarded as a social actor and each hyperlink as a relationship. Therefore, results from social network analysis can be adapted and extended for use in the Web context. In fact, the two most influential link analysis methods, PageRank and HITS, are based on the ideas borrowed from social network analysis.

Below, two types of social network analysis, i.e. centrality and prestige, which are closely related to hyperlink analysis and Web search, are introduced. Both of them are measures of degree of prominence of an actor in a social network.

Centrality

Intuitively, important or prominent actors are those that are linked or involved with other actors extensively. Several types of centrality are defined on undirected and directed graphs. The three popular types include degree centrality, closeness centrality, and betweenness centrality. For example, using a closeness centrality, the center of the network is defined based on the distance: an actor i is central if it can easily interact with all other actors. That is, its distance to all other actors is short. Let the shortest distance from actor i to actor j be $d(i,j)$, measured as the number of links in a shortest path. The closeness centrality $C_c(i)$ of an actor i is defined as

$$C_c(i) = (n-1) \Big/ \sum\nolimits_{j=1,\ldots,n} d(i,j) \tag{2.15}$$

Prestige

Prestige is a more refined measure of prominence of an actor than centrality in that it distinguishes between the links an actor receives (inlinks) and the links an actor sends out (outlinks). A prestige actor is defined as one who receives a lot of inlinks. The main difference between the concepts of centrality and prestige is that centrality focuses on outlinks while prestige focuses on inlinks. There are several types of prestige defined in the literature. The most important one in the context of Web search is perhaps the rank prestige.

The rank prestige forms the basis of most Web page link analysis algorithms, including PageRank and HITS. The main idea of the rank prestige is that an actor's prestige is affected by the ranks or statuses of the involved actors. Based on this intuition, we can define $P_R(i)$, the rank prestige of an actor i as follows.

$$P_R(i) = A_{1i}P_R(1) + A_{2i}P_R(2) + \cdots + A_{ni}P_R(n) \tag{2.16}$$

where, $A_{ji} = 1$ if j points to i, and 0 otherwise. This equation says that an actor's rank prestige is a function of the ranks of the actors who vote or choose the actor. It turns out that this equation is very useful in Web search. Indeed, the most well known Web search ranking algorithms, PageRank and HITS, are directly related to this equation.

Summary

In this chapter, we briefly include some basic but important concepts and theories used in this book. The mathematical descriptions and formulations are reviewed and summarized in eight sections. This chapter and the successive Chapter 3 form the foundation part of this book.



http://www.springer.com/978-1-4419-7734-2

Web Mining and Social Networking
Techniques and Applications
Xu, G.; Zhang, Y.; Li, L.
2011, XVII, 210 p., Hardcover
ISBN: 978-1-4419-7734-2