UNIT 12 WEB AND SOCIAL NETWORK ANALYSIS

Structure **Page No.**

12.0 INTRODUCTION

In the previous units of this block, you have gone through various methods of analysing big data. This unit introduces three different types of problems that may be addressed in data science. The first problem relates to the issue of advertising on the web. advertisement on the web is most popular with search queries. A typical advertisement-related problem is: what advertisements are to be shown, as a result of a search word? This unit defines this problem in detail and shows an example of an algorithm that can address this problem. Next, this unit discusses the concept of a recommender system. A recommender system is needed, as there is an abundance of choices of products and services. Therefore, a customer may need certain suggestions. A recommender system attempts to give recommendations to customers, based on their past choices. This unit only introduces content-based recommendations, though many other types of recommender systems techniques are available. Finally, this unit introduces you to concepts of the process of finding social media communities. You may please note that each of these problems is complex and is a hard problem. Therefore, you are advised to refer to further readings and the latest research articles to learn more about these systems.

12.1 OBJECTIVES

After going through this Unit, you will be able to:

- Define the issues relating to advertisements on the web
- Explain the process of solving the AdWords problem.
- Define the term long tail in the context of recommender systems.
- Explain the model of the recommendation system and use the utility matrix
- Implement a mode for the content-based recommendations
- Define the use of graphs for social network
- Define clustering in the context of social network graphs.

12.2 WEB ANALYTICS

Web analytics is the process of collecting, analysing and reporting data collected from a website with the objective of measuring the effectiveness of various web pages in achieving the website's purpose. A webpage consists of a large number of sections or divisions or fragments. Each of these fragments has a certain purpose, for example, a website may consist of an advertisement section, multiple information sections, a navigation section etc. Web analytics may also be used to measure the traffic on the website, especially after an event, such as for an eCommerce website after a sale offer is announced or for a university after the announcement of admission.
What kind of data is collected for web analytics? In general, for web analytics the following information may be collected by different websites:

- The user location and identification of the user, such as IP address, if permitted.
- The time spent by users on the website and the pages or sections accessed by the user during her/his stay at the website.
- The number of users, who are accessing the websites at different times. The number of simultaneous users.
- Information about the kind of browser and possibly the machine used by a user to access the website.
- If the user has clicked on any external link given on the website.
- Is any advertisement on the website clicked?

In addition, based on the objectives of the web application related information may be collected.

Web analytics is performed to address the basic question: Is the website able to fulfil the objectives with which it was created? Some of these objectives may be:
Is the website informative for users? This would require analysis of the number of users visiting, the time spent by them on information pages, whether are users coming again to the website etc.
How can cloud services used for hosting the website be optimised? This question may be performed by doing the traffic analysis and simultaneous user data etc.
Thus, web analytics may be very useful for enhancing the efficiency and effectiveness of a website. In the subsequent sections, we discuss some of the specialized applications of website-related information.

12.3 ADVERTISING ON THE WEB

Advertising is one of the major sources of revenue for many professions like Television, newspapers etc. The popularity of WWW has also led to the placement of advertisements on web pages. Interestingly one of the most popular places to put advertisements is the output of a search engine. This section describes some of the basic ways of dealing with advertising on the WWW and some of the algorithms that can be used to find the outcome of using the advertisements on the WWW.

12.3.1 The issues

In order to define the issues related to web advertisements, first let us discuss different ways of placing advertisements on the Web.

Historically, the advertisements were placed on the websites as banners and were charged for the number of impressions or webpage visits consisting of that banner. This model was similar to advertisements of magazines that used to charge based on their circulation. However, on web pages, defining the rate of advertisements in this way is very inefficient. Most advertisers define the performance of a webpage advertisement, as the ratio of the number of times an advertisement is clicked from a webpage to the number of impressions of the advertisement on that webpage. As the performance of banner-based advertisements was very poor, the targeted impression of advertisements based on demography started resulting in a slightly better click ratio. However, even this kind of advertisement also had poor performance. Thus, a newer form of advertisement is being practised on the website, which was first designed by an old company named Overture in 2000 and later redesigned by Google in 2002. It is a performance-based advertisement, which is called AdWords and is defined as follows:

1. The advertisements were placed as the result of a search term
2. The advertisers are asked to bid for placing an advertisement on the result of search terms or keywords.
3. The advertisements are displayed (*which advertisement?)* when that search engine displays the result of the specific search term.
4. The advertisers are charged if a displayed advertisement is clicked on.

You may need to design an algorithm to find out, *which advertisement* will be displayed as a result of a search query. This is called the AdWords problem. The AdWords problem can be defined as follows:

Given:

- A sequence or stream of queries, which are arriving at a search engine, regularly. The typical nature of queries is that only the present query is known, and which query will come next cannot be predicted. Let us say, the sequence of the keyword queries is *q1, q2, q3*, …
- On each type of query, several advertisers have put their bids. Let us say that *m* bids are placed on each type of query, say *b1, b2, ..., bm.*
- The probability of clicking on an advertisement shown for a query, say *p1, p2, ...*
- The Budget stipulated by an advertiser, which may be allocated for every day, for *n* advertisers, say *B1, B2, B3, ...Bn*.
- The maximum number of advertisements that are to be displayed for a given search query, say *t*, where $t < m$.

The objective of the Problem:

Select the sub-set of advertisements to be displayed on the arrival of a query, which maximises the profit of the search engine.

The Output:

- The size of the selected sub-set of advertisement should be equal to *t*

- The advertiser of the selected advertisement has made a bid for that type of query.
- In case the ith advertisement is selected then the Budget left should be more than the bid amount, *Bi >= bi.*

12.3.2 The Algorithms

The category of algorithms to address the AdWords problem is known as online algorithms. An online algorithm works on a partially available data set for making committed decisions at a given instance of time. You may compare such algorithms to offline algorithms, which are the algorithms that we normally write. Offline algorithms process the complete data sets. AdWords problem is the typical case for an online algorithm to be used, as on arrival of a query a decision is to be made to show few advertisements. This decision then cannot change, as advertisements are shown along with the display of results.

Greedy Algorithm:
One of the simplest algorithms to solve this problem would be to use the greedy algorithm, where the important considerations here are to show the advertisements which have (1) high bid value and (2) high chances of getting clicked, as the payment will be made only if the advertisement is clicked. This will be subject to the constraint that the advertiser's budget has not been over. For example, consider a query *q1* that has the following bids:

Advertiser	Bid (*bi*) in INR	Budget of Bidder (*Bi*) per day	Probability of clicking that advertisement (*pi*)	Probable Revenue	Rank of advertisement selection, if Bi >=bi
X	80	160	0.01	0.8	3
Y	60	60	0.02	1.2	1
Z	30	30	0.03	0.9	2

Figure 1: A sample advertiser's Bid

Thus, assuming that only one advertisement is to be displayed in the results, the greedy algorithm will display the advertisement of advertiser Y, as it is ranked 1. However, please notice that these impressions of the advertisement of Y will be displayed only till the time the advertisement is clicked, as once it is clicked the Budget for the advertiser will become zero for that day. Please also note that we are assuming there would be many queries which are of the same type as query *q1*.

One of the questions here is how to compute the probability of clicking the advertisement, this information can be computed only after certain experimentation and for a new advertisement this value would not be known. A discussion on this is beyond the scope of this unit, you may refer to further reading for more details on this issue.

The greedy algorithm, as discussed here may be a useful way of increasing revenue. However, it may not be able to provide the optimal possible revenue. In fact, it has been shown that in the worst case, a greedy algorithm would be

able to produce ½ of the optimal revenue. This can be explained with the help of the following worst case.

Consider there are two types of queries *Q1* and *Q2* having the following bidders:

Query Type	Advertiser	Bid (*bi*) in INR	Budget of Bidder (*Bi*) per day	Probability of clicking that advertisement (*pi*)	Probable Revenue
Q1	X	80	160	0.01	0.8
Q2	X	80	160	0.01	0.8
	Y	80	160	0.01	0.8

Figure 2: A sample bid for the worst case

Please note that query *Q1* has only one bidder X and query *Q2* has two bidders X and Y. You may also note that all other parameters are the same for the queries. Now assume the following sequence of queries occurs:

Q2, Q2, Q1, Q1

Now, assume that the advertisement of X is selected randomly for the impression on query *Q2* and it also gets clicked, then the allocation of the advertisements would be:

X, X, -, -

This occurs because the advertisement budget of X is over after 2nd click and only X has bid for Query *Q1*. This is a typical problem of an online algorithm, where that information about future queries is not known. You may observe that if this sequence is known before then an optimal selection would have been:

Y, Y, X, X

Which would optimise the revenue. In the worst case, you may also observe that the Greedy algorithm has earned ½ of the revenue earned by the Optimal algorithm.

Is there a better algorithm than this Greedy algorithm? Fortunately, a better algorithm was developed for this problem. This is called a Balanced algorithm. In this algorithm, in case of a tie, the advertisement of the advertiser with a higher budget is displayed. For example, advertisement selection for the query sequence given above would be:

X, Y, X, -

Please note that once X's advertisement is displayed and clicked, it has a lower budget than that of Y, which will result in the display of Y's advertisement in the second instance of *Q2*. You may please observe that the case of the balanced algorithm earns about ¾ of the optimal revenue. You may go through the further readings for more details on these algorithms.

Check Your Progress 1:

Question 1: What is the AdWords problem?

Question 2: What is an online algorithm?

Question 3: Differentiate between greedy and balanced algorithms of AdWords problem.

12.4 RECOMMENDATION SYSTEMS

In the last section, we discussed the AdWords problem. In this section, we focus on another important domain of problems called recommendation systems.

Let us first try to answer the question: What is the need for recommender systems?

Consider you have to buy some products from a very large list of products, as shown in Figure 3. For example, you have to buy a TV and you are not sure about the size, type, brand etc and a very large choice of sizes, types and brands exists. In such situations, you would like to consider impartial advice. This may be the reason the recommender systems were created.

Figure 3: The need for a Recommendation system

In case you are looking for an item from a very large set of items, say thousands, and your requirements are not very well specified then rather than searching for that kind of item, you may like suggestions about the item from some known person. The recommender system does this job, as it recommends the product, which may be needed by you, based on the information known to the system about you.

12.4.1 The Long Tail

Why did recommendation systems become popular? This can be attributed to the web-based marketing of the products. In direct marketing, the shelf space of the store is very important. However, as marketing moves to the web, a very large number of similar products can be sold leading to marketing of choice. Figure 4 shows an interesting term called the long tail of products. In this era of the digital market, a large number of products are on sale. Some of these products sell at a very high rate, but most of the products have a low demand. These low-demand products are also very large in number and can be of very good quality and are termed long-tail products (see Figure 4).

In the era of the direct market, only those products that were selling the most were finding a place on the shelf and were being stocked. However, in this digital market even the products, which have low demand can be sold, as they need not be stocked at all locations, but can be made available on demand. Thus, long-tail products are now available to general users. For example, suppose you want to buy only albums of classical music, then earlier the stores which used to sell them were very difficult to find. However, now you may be able to order them easily. In addition, suppose you order some such system using the digital market, then based on your choices it is possible to suggest to you some other classic albums that you may find interesting. This is the advantage of the recommender system.

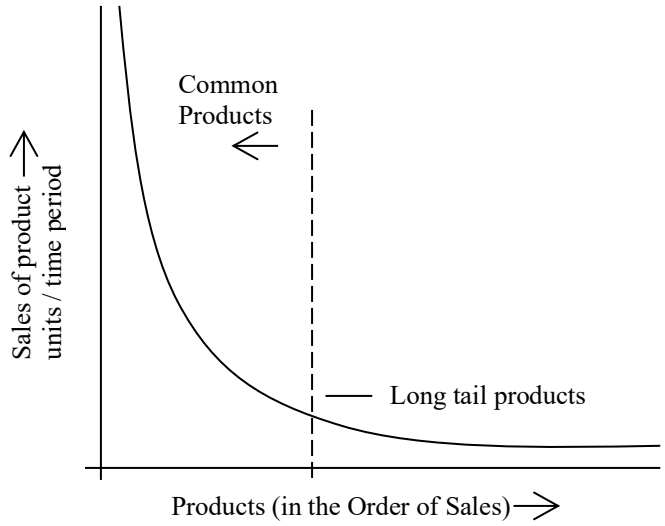

Figure 4: Product sale and long tail [1]

Some of the most common applications of the recommender systems are for recommending books, music albums, movies, published articles etc. The usefulness of the recommender system can be gauged from the fact that there are a large number of cases that highlights the success of the recommender system application. For example, a recommender system of books found that several purchasers of a book, say a book named A, are also purchasing not a very popular but highly rated book B. The system started showing this book, as a recommendation to the purchasers of Book A. It turns out that after a while even book B became one of the best sellers. Thus, recommendation system applications have great potential, as they can inform purchasers about the availability of some good items, which were not known to them due to the very large number of items.

12.4.2 The Model

There are three basic types of recommended systems:

1. Recommendations by the Editors/Critics/Staff: This is the simplest kind of recommendation system, which are in existence for a long time. In such systems, recommendations of editors or critics or staff are recorded and reported. One of the major weaknesses of such systems is that it does not take input from the actual customers.

2. Aggregates recommendations: Such information is aggregated from the various user or customer activities, for example, the most popular video or the most purchased product or the highly rated services etc. However, these recommendations may not suitable to your requirements, as these recommendations are generic.
3. User-specific Recommendations: These recommendations are specifically being made for a specific user based on his/her past web activities such as online purchases, searches, social media interactions, viewing of video, listening to audio etc.

In this section, we will discuss the model for user-specific recommendations. The specific problem of user-specific recommendations can be stated as:

Given:

- A set of users or customers, say S who has given some rating to some of the products/services.
- A set of products or services, say P, being sold or provided by the web application.
- A set of ratings for every pair of customers and products, these ratings may use a star rating scale of 0 to 5 or 0 to 10 or just 0 (dislike), 1(like). It may be noted that the values of 0 to 5 or 0 to 10 are ordered in terms of liking a product with 5 or 10, as the case may be, being the highest level of liking.

  These ratings may be represented by using a model or a function as:

  $$f: S \times P \rightarrow R$$

  This mapping can be presented using a utility matrix, which is a sparse matrix (see Figure 5).

The objective of the Problem:

To find the expected rating that may be given by a customer s to a product p, which s/he has not rated, based on her/his other ratings.

The Output:

- Customer-specific recommendations based on the computation of expected ratings.

	P1	P2	P3	P4	P5
C1	3	1			2
C2	4			1	
C3		1	5		3

Figure 5: A sample Utility Matrix (3 customers and 5 products on a 0 to 5-star rating)

The Key Problems of Making a Recommender system: In order to make a recommender system you need to address the following three problems:

1. Gathering the ratings into a utility matrix: One of the simplest ways for gathering the rating would be to directly ask the customer to rate a product, however, very few people spend time to rate a product, therefore, in addition to this direct rating most e-commerce website implicitly rate their products. But how may this implicit rating work? Well, consider a case, where a customer returns a purchased item, which means he rates the item lowly; on the other hand, a purchased

item may get a good implicit rating. An item which is purchased again by a customer should be given a high implicit rating. A utility matrix may contain both direct and implicit ratings given by a customer.

2. Computing the unknown ratings for every customer. The focus here is to find the high unknown ratings only, as you would like to recommend a product to a customer only if he is expected to rate it highly. The key problem in finding the unknown ratings is that the utility matrix is sparse. In addition, a new product has no ratings when added to the list. There are three different types of approaches to computing rating systems – the content-based approach, the collaborative filtering and the latent factor-based approach. We will only discuss the content-based approach in the next section. You may refer to the further readings for the other methods.
3. Evaluating the performance of the recommender system: One way of computing the performance of a recommender system is based on the fact that a recommendation has resulted in a successful purchase or not. You may refer to the further readings for more details on this.

12.4.3 Content-Based Recommendations

The content-based recommender system is designed to recommend products/services to a customer based on his/her earlier highly rated recommendations. Some of the commonest examples of a content-based recommendation system are the recommendation of movies, research articles, friends etc. For example, if you highly recommend a movie then a content-based recommendation system is likely to recommend movies of similar actors or genres etc. Similarly, if you highly rate a research article the content-based recommendation system will recommend research articles of similar content or nature. How is such a recommendation made? Figure 6 defines this process:

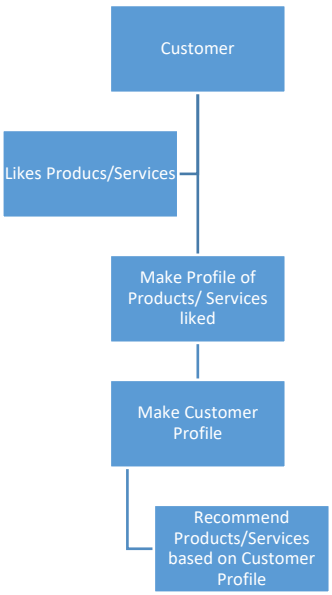

Figure 6: Content-based Recommender System

The profile of products or services is defined with the help of a set of features. For example, the features of a movie can be the actors, genre, director, etc.,

whereas a research article's features may be authors, title, metadata of research articles, etc. The accumulation of these features makes the customer profile. But how do we represent the product profile? One way to represent the product profile would be to use a set of vectors. For example, to represent a movie, we may use a feature vector consisting of the names of the actors, names of the director and types of genres. You may notice that this vector is going to be sparse. For research articles, you may use the term frequency and inverse of document frequency, which were defined in an earlier unit. The following examples show how the utility matrix and feature matrix can be used to build a customer profile.

Consider you are making a recommendation system of movies and the only feature you are considering is the genre of the movies. Also, assume that movies have just two genres – X and Y. A person rates 5 such movies and the only ratings here are likes or no ratings, then the following may be a portion of the utility matrix for the customer, let us say Customer Z. Please note that in Figure 8, 0 means no ratings and 1 means like.

	M1	M2	M3	M4	M5	M6	M7
Cz	1	1	1	1	1	0	0

Figure 8: A Sample Utility matrix

Further, assuming that among the movies liked by customer Z, movies M1, M2 and M3 are of genre X and M4 and M5 are of genre Y, then the article profile for the movies is given in Figure 9. Please note that in Figure 9, 1 means the movie is of that genre, whereas 0 means that the movie is not of that genre.

	M1	M2	M3	M4	M5
X	1	1	1	0	0
Y	0	0	0	1	1

Figure 9: A sample product feature matrix

This product matrix now can be used to produce the customer Z profile as:

Feature Genre X profile = sum of the row of X/Movies rated
= 3/5

Feature Genre Y profile = 2/5

You may please note that the method followed here is just finding the average of all the genres. Though for an actual study, you may use a different aggregation method.

However, in general, the customers rate the movies on a 5-point scale of say, 1 to 5. With 1 and 2 being negative ratings, 3 being neutral rating and 4 and 5 being positive ratings. In such as case the utility matrix may be as shown in Figure 10.

	M1	M2	M3	M4	M5	M6	M7
Cz	1	1	2	4	2	NR	NR

Figure 10: Utility matrix on a 5-point rating scale (1-5)

NR in Figure 10 means not rated. Now considering the same product feature matrix as Figure 9. You may like to compute the profile of customer Z. However, let us use a slightly different method here.

In general, on a 5-point scale, each customer has their way of assessing these ratings. Therefore, it may be a good idea would be to normalize the ratings of each customer. For that first find the average rating of a customer. This customer has rated 5 movies with an average rating of 10/5 =2. Next, use this average rating to mean neutral rating and subtracting it from the other rating would make the ratings as:

	M1	M2	M3	M4	M5
Normalize ratings of Cz	-1	-1	0	2	0

Figure 11: Normalized Utility matrix

In this case, you may use the following method to create the customer Z profile:

Feature Genre X profile
= sum of normalised rating of genre X/Movies rated of genre X
= Normalised rating of (M1+M2+M3)/3
= (-1-1+0)/3 = -2/3

Feature Genre Y profile
= sum of normalised rating of genre Y/Movies rated of genre Y
= Normalised rating of (M4+M5)/2
= (0+2)/2=1

Thus, customer Z has a positive profile for genre Y.

There can be other methods of normalization and aggregation. You may refer to the latest research on this topic for better algorithms.

You have the product/service profiles, as well as customer profiles available to you. Now, the next question is: how will you recommend a product or a service to a customer?
Please notice that both the product/service profile and the customer profiles are high-dimensional vectors, therefore, you may use the cosine distance to compute the distance between the two vectors using the cosine distance vector formula, which allows you to find the angle, say *x*, between two vectors using the vector dot product. The cosine similarity is defined as 180-*x*. You must recommend those products or services to the customer, which are highly similar to its customer profile. More details on this can be studied from further readings.

Check Your Progress 2:
Question 1: What is a recommender system?

Question 2: Define the term long tail.

Question 3: Define the process of content-based recommendations.

12.5 MINING SOCIAL NETWORKS

In the previous section, we discussed the recommender system. Social media has an immense amount of data about people's interactions among themselves. This data can be processed to produce useful information about certain products, services and communities. Social media data is obtained from the social media networks such as Facebook, Instagram, Twitter, LinkedIn, and many more. In general, on social media people form connections with one

another, for example, friends on Facebook. One of the key questions that may be asked relating to social media is: To find the communities, a sub-set of related groups, from amongst a very large number of people, for example, one social media you may be part of a group of your classmates of the school as one community, a job-related group as another etc. One of the characteristics of a community is that people of one community may know each other and possibly share the same interests, whereas they may not know the people of other communities. In this section, we discuss the mining of social media networks in more detail.

12.5.1 Social Networks as Graphs

A social media network consists of a large number of people their connections and interactions with other detail. How can you represent the people and their connections?
Social media networks can best be represented with the help of a graph where a node represents a person and links represent the relationship of the person with others. In general, these links can be undirected, however, in certain cases, the links can be one-directional or directed. Figure 12 represents a typical social media network graph.

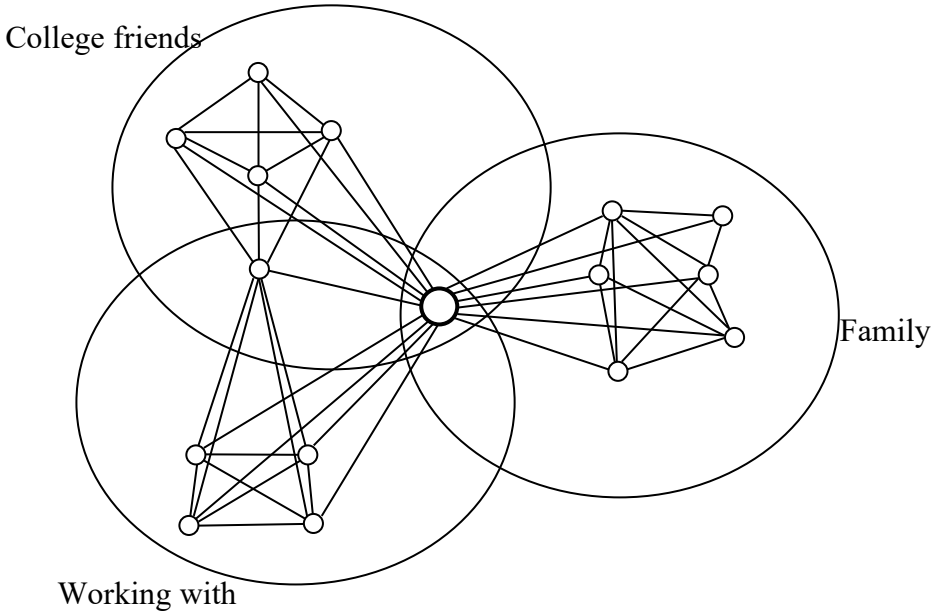

Figure 12: A Social Media Network Graph

Please observe that in Figure 12, you may be connected to several communities and there may be overlap in certain communities. This is the typical nature of social media networks graph.

12.5.2 Varieties of Social Networks

There are many different kinds of social networks. Most of these networks are used for sharing personal or official information. Some of the basic categories of these networks are:

1. Traditional social networks: Mainly used for sharing information among communities of friends or people who come together for a specific purpose. These networks are primarily analysed for getting information about groups. An interesting type of social network in this category is the collaborative research network. Such a network includes links among authors who have co-authored a research paper. In addition, one of the communities of this network is the editors of the research work. These networks can be used to find researchers who share common research areas.
2. Social review or discussion or blogging networks: Such networks create a node for the users and make links if two user review/discuss/blog the same topic/article. These networks can be used to identify communities, which have similar thinking and inclinations.
3. Image or video sharing networks: These networks may allow people to follow a person hosting a video. A follower may be linked to a host in such networks. The meta tags of videos and images and comments of the people watching the videos may be used to create communities that share common interests.

In addition, there can be a large number of information networks that can be represented as graphs. Next, we discuss some of the issues relating to finding communities on a social network graph.

12.5.3 Distance Measure of Social Network Graphs

As discussed in the previous section, social media networks can be represented using graphs. For finding the communities in this social media graph, one may use clustering algorithms. One of the clustering requirements is finding the distance between two nodes. Thus, we must define a method to find the distance between two nodes in a social media network graph. You may recollect that in a social network graph, a node represents a person or entity and a link represents a connection between two entities. However, the problem here is that as the link represents the friend or connection, it does not have any weight assigned to it. In such a situation, the following distance measure may be considered for social network graphs:

The distance between two nodes A and B, $dis_{A,B}$ is:

$dis_{A,B} = 1$; if there is a link from A to B.

$dis_{A,B} = \infty$; if there is no direct link from A to B

The distance measure, as suggested above has one basic issue, which is as follows:

Consider three nodes A, B and C with only two of these node pairs (A, B) and (B, C) are connected; then:

$dis_{A,B} = 1$; $dis_{B,C} = 1$; $dis_{A,C} = \infty$

However, this may violate the true distance measure rule:

$$dis_{A,B} + dis_{B,C} >= dis_{A,C}$$

Thus, the definition of distance, when there is no direct link may need to be redefined, if any traditional clustering methods are to be considered.

12.5.4 Clustering of Social Network Graphs

The social media network consists of a very large number of nodes and links. As stated earlier, one of the interesting problems here is to identify a set of communities in this graph. What are the characteristics of a community or cluster in a graph?

A community or cluster in a graph is a subset of a graph having a large number of links within the subset, but having less number of links to other clusters. You may observe that in Figure 12 such clusters exist, though the cluster of "college friends" and "working with" have few common nodes, but, in general, the previous statement holds. One additional, feature of the social media network graph is that a cluster can be further broken into sub-clusters. For example, in Figure 12, the "college friend" cluster may consist of two sub-clusters: Undergraduate College Friends and Postgraduate College friends.

You may please note that such clustering is very similar to hierarchical clustering or k-means clustering used in machine learning, with the difference that here you are wanting to find clusters in graphs and not in large datasets of points.

Given:

A social network undirected graph, say *G (V, E),* where *V* is the set of nodes, which represent an entity such as a person, and *E* is the set of edges, which represent the connections, such as friends. The links are not assigned any weight (see Figure 13)

The objective of the Problem:

To find good clusters, which maximise the links within a cluster and minimise the links between clusters.

The Process:

Find a set of minimal edges, which when removed creates a cluster. For example, in figure 13 the edges {C, E} and {D, F}. This will divide the graph given in Figure 13 into two clusters {A, B, C, D} and {E, F, G, H}.

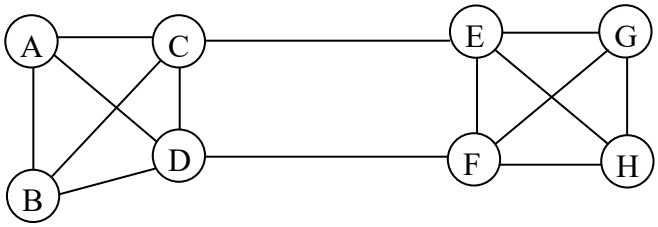

Figure 13: Social Network Graph *G (V, E)*

Several algorithms have been developed for efficient clustering and the creation of communities in social network graphs. You can refer to the further readings for a detailed discussion on this topic.

Check Your Progress 3:

1. Why are graphs used to represent social media networks?
2. What are the different types of social media networks?

3. Define the objective of clustering in social network graphs.

12.6 SUMMARY

This unit introduces you to some of the basic problems that can be addressed by data science. The Unit first introduces you to the concept of web analytics, which is based on a collection of information about a website to determine if the objectives of the website are met. One of the interesting aspects of doing business through the web is web advertising. This unit explains one of the most important problems of advertisement through web AdWords problem. It also proposes a greedy solution to the problem. The unit also proposes a better algorithm, named a balanced algorithm, than the greedy solution to the AdWords problem. Next, the Unit discussed the recommender system, which results from long-tail products. The recommender system aims at providing suggestions to a person based on his/her ranking of past purchases. These recommendations are, in general, such that recommended products may be liked by the person to whom the recommendations are made. In the context of recommender systems, this unit discusses the concept of the long-tail, utility matrix. In addition, the unit discusses two algorithms that may be used for making content-based recommendations. Finally, the unit discusses the social media network, which is represented as graphs. The unit also introduces the process of clustering for the social media network. You may please go through further readings and research papers for more detail on these topics.

12.7 SOLUTIONS/ANSWERS

Check Your Progress 1:

1. AdWords problem is basically to select a set of advertisements to be displayed along with the result of a search query. The decision to choose advertisements is taken based on the data such as bidding and budget of the advertiser, the probability of clicking on a given advertisement by a user; to maximise the profit of the advertiser.
2. An online algorithm is one, which makes a committed decision based on currently available data. It may be noted that such an algorithm does not have access to the complete set of data, as is the case of an offline algorithm. Online algorithms are useful to address the AdWords problem, as the complete sequence of the query stream is not known at the time of making decisions about the selection of an advertisement to display.
3. The decision to select an advertisement to display in the Greedy algorithm is based on the probable revenue of the advertisement. In case, two advertisements have the same probable revenue, then any one of them is selected at random. Whereas in the case of the balanced algorithm, the selection is based on probable revenue and remaining budget.

Check Your Progress 2:

1. In the present scenario of online marketing, a large number of choices of products are available. A recommender system is a system, which based on previous ratings or purchases made by a customer, suggests newer products or items that are expected to be liked by this customer.
2. The term long tail is coined for online marketing, where a large number of products, which may be of good quality, have not been purchased as frequently as popular products. A company may have a very large list of such products, therefore, despite fewer sales, overall long-tail products can generate substantial revenue.

3. The content-based recommendations collect the customer ratings of various products and services. It also makes the profile of products based on various attributes. The customer ratings and product profiles are used to make a profile of the customer as per the attributes, which are then used to make specific recommendations of the products, which are expected to be rated highly by the customer.

Check Your Progress 3:

1. Most social media networks represent a set of entities and their relationships. An entity can be represented with the help of a node, which can also store the activities performed by that entity. The relationships can best be represented with the help of an edge between two nodes. Social media networks have a large number of complex relationships. The graphs are the most natural way of representing the nodes and complex relationships. In addition, this allows the use of graph-based algorithms for analytics.
2. Different types of social media networks include traditional social networks like Facebook, Twitter etc; social review of discussion or blogging-based networks and many image and video sharing networks.
3. One of the analyses of social networks is to find social communities, which have a large number of connections within but have less number of connections with other communities. This can best be performed with the help of finding clusters in graphs.

12.8 REFERENCES/FURTHER READINGS

1. Leskovec J., Rajaraman R, Ullman J, **Mining of Massive Datasets,** 3rd Edition, available on the website http://www.mmds.org/
2. Gandomi A., Haider M., **Beyond the hype: Big data concepts, methods, and analytics**, International Journal of Information Management, Volume 35, Issue 2, 2015, Pages 137-144, ISSN 0268-84012.