ADAPTIVE MODEL FOR WEB SERVICE RECOMMENDATION

Prof.Dr.TorkeyI.Sultan
Information Systems Department,Faculty of Computers&Information
Helwan University, Cairo, Egypt

Dr. Ayman E. Khedr
Information Systems Department,Faculty of Computers& Information
Helwan University, Cairo , Egypt

Fahad Kamal Alsheref

Information Systems Department,Faculty of Computers& Information
Helwan University, Cairo, Egypt

ABSTRACT

The Competition between different Web Service Providers to enhance their services and to increase the users' usage of their provided services raises the idea of our research. Our research is focusing on increasing the number of services that User or Developer will use. We proposed a web service's recommendation model by applying the data mining techniques like Apriori algorithm to suggest another web service beside the one he got from the discovery process based on the user's History.

For implementing our model, we used a curated source for web services and users, which also contains a complete information about users and their web services usage. We found a BioCatalogue: our proposed model was tested on a Curated Web Service Registry (BioCatalogue).and 70 % of users chose services from services that recommended by our model besides the discovered ones by BioCatalogue.

KEYWORDS

Web services discovery, BioCatalogue, Data Mining, and Recommendation System

1. INTRODUCTION

Web service is one of the important inventions in our technology world, because it offered several properties such as Interoperability, Usability, Reusability and Deployability. In addition,it can be integrated into applications over networks through a structured programming interface. Software applications written in various programming languages and running on various platforms can use web services to exchange data over the Internet.

Web services have become as a commodity in market while the users had become as a customer who search for a suitable service for his work. Thinking of web service suggestion as commodity and customers raise the idea of using idea of Recommender systems that used by E-commerce sites. Companies to suggest items to their customers use recommendation systems. The Products recommendation process can be done through analysing several shared properties between customers like nationality, site, demographics, customer's behaviour and buying history. Through analysing these properties, the customer future buying behaviour can be predicted.By using the same concept, user can search for a web service by writing his query then the service discovery

DOI : 10.5121/ijwsc.2013.4403

agent finds the suitable service. Our model works after the discovery model finishing his process by finding the required web service and the user accepts this service then our model works as a service recommender based on the analysis of users history. [1]

For example when user searches for a service of text mining then after finding it our proposed model analyses all records of users that used this service and gets list of other services that they used. Then the recommender model applies data mining methods like Association rule that using Apriori algorithm to find recommended services, then the user will decide to choose any of the recommended services.

Applying this approach leads to expand the usage of web services that provided by the web services provider, and it gives the users more information about the most used services that used by other users who used the same service.

The rest of this paper is organized as follows. Section 2 provides the background. Section 3 contains the IR web service's discovery model that represents the basic functions in our proposed model. Section 4 represent our model with its modifications. Section 5 studies the evaluation of our proposed model on case study. Finally, Section 6 contains the conclusion and future work.

2. BACKGROUND

In this section, we provide a small survey of the techniques that used in our research. These techniques are: 1) Association rule, 2) BioCatalogue A Curated Web Service Registry.

A. Association rule

Association Rule is one of Data mining techniques that used to find the associations between several classes or clusters of items. In addition,it named by Market Basket analysis or Affinity analysis. Applying this technique leads to find the association between different groups or items such as the relations between students and courses For example:

Students taking CC482 often also take CC481 and CC483.

Association rulesprocess is composed intotwo-stage process:

- Find all frequent itemsets.
- Generate strong rules from frequent item sets.[2]

Formal Definitions:

Support is the number of transactions that includeItem A.

$$\text{Support} = \text{Probability}(A)$$

Confidence is the number of transactions that contain item B and must containitem A.

$$\text{Confidence} = \text{Probability}(B \text{ if } A) = P(B/A)$$

Strong Rules

Let:
MinSupminimum value of support threshold
MinConfminimumvalue of confidence threshold

Therefore, the strong rule is defined by the value that its support and confidence are more or equal the MinSup and MinConf.

Frequent Itemsets

- An itemset is a set contains items. (i.e. a subset of L).
  - Each itemset that containing L items called aL-itemset.
- The support of an itemset is the percentage of the total number of transactions.
- An itemset that itssupport are more or equal the MinSup and MinConf.is called a frequent itemset.

Finding the Frequent Itemsets

The process of finding the frequent item is divided into two phases:

Phase 1: The Apriori algorithm: is an algorithm for finding patterns in data. It is based on an observationfor example in the supermarket if we analysed sales of two items like milk and rice and we found that most of customers that buying milk also buying rice that means there is a strong association between the sales of these two items, this fact can be used as a hint to the manager of the super market to put the milk and rice in same corner to increase the sales of the two items.

The Basic Idea:

All non-empty subsets of a frequent itemsets must themselves be frequent.
Why?

If itemset X is not frequent that means Probability of X is less than MinSup.
If another itemY is insertedbesideXthat means the new generatedset is XUnionY.
Clearly ProbabilityXUnionY<= Probability(X)
Hence ProbabilityXUnionY <MinSup.
Thus the frequent (L+1)-itemsets can be derived if we already know the frequent L-itemsets.
How?
Form (L+1)-itemsets by joining pairs of itemsets from the frequent L-itemsets.
Ignore all sets that includes subsets which is not frequents.

Steps to Perform Apriori Algorithm: [4][7]

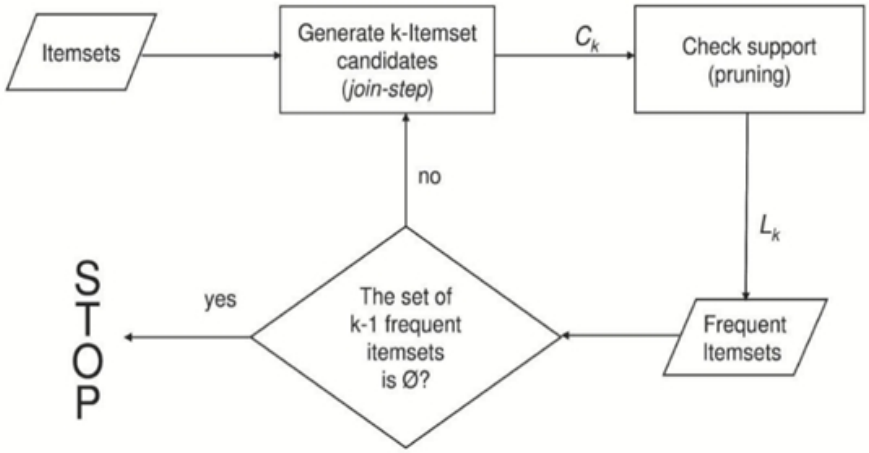

Figure 1 Apriori Algorithm

Phase 2: Generate Association Rules from Frequent Itemsets: For all generated pairs of frequent itemsets the union of each pair is also frequent, the confidence of the rule between pairs A, B:

c = support (A U B) / support (A).

If confidence value of the association rule is not less the MinConf that is meaningthe rule is strong and should be added to frequent itemset, in the following section we show the application of Apriori in our model. [4]

B. BioCatalogue A Curated Web Service Registry

BioCatalogue is a web service registry dedicated to the life science community. BioCatalogue provides a means, by which a bioinformatician, for instance, can subscribe his /herfavourite web service within the catalogue, e.g., by uploading the web service description file (WSDL). The Bioinformatician has the ability to add and edit the description of the web service that is included in the WSDL document with mapping service's elements to domain ontology. Missing terms can be also substituted with other tags that can be mined for new ontology terms. [5]

BioCatalogue Functionalities:

These sections overview the main functionalities that will be supported by BioCatalogue.

- Integrated access to life science web services. BioCatalogue will provide an integrated access to life science web services: it will allow users to locate web services that are implemented by different providers and that are hosted by remote server. [5]
- Rich Description of Web Services. Available web services are rarely described, and when they are, they are poorly described [1]. it gives a full description of the functional capabilities supplied by the service and the format of the inputs, the ranges for any parameters who else uses it and why etc.[5]
- Curation of Service Descriptions. From bioinformatics view,we are looking at curated data as a prerequisite for data integration. In this respect, and as pointed out earlier, to locate the appropriate web service, the scientists will need some information about the service. BioCatalogue will provide scientists with this information.[5]

- Web Service Discovery. BioCatalogue allows users to discover the web service using mainly the following mechanisms. Keyword-based service retrieval allows users to locate web service of interest by providing as input one or multiple key words that provide some information about the service, e.g., the name of service, its tag. Advanced search capabilities will also be supported by BioCatalogue.[5]
- Interoperability. BioCatalogue will provide programming tools that can be used by other applications, such as myExperiment and Taverna, to programmatically access the web service registry.[5]

The BioCatalogue also contains a set of public RESTful endpoints that can be integrated and used programmatically with other programming languages. We used these API's to test our proposed model and to verify the results.

3. RELATED WORK

A.The IR-style Web Services Discovery

Chen Wu proposed IR-style Web services discovery approach that was illustrated in Fig. 2, in which term tokenization constitutes an important step for WSDL term processing. Initially, service providers deploy their Web services accessible to the public via the Web. In doing so, they also publish a service description, i.e., the WSDL documents, which captures the functional capabilities and technical details (e.g., transport bindings) of a Web service. [6][8]

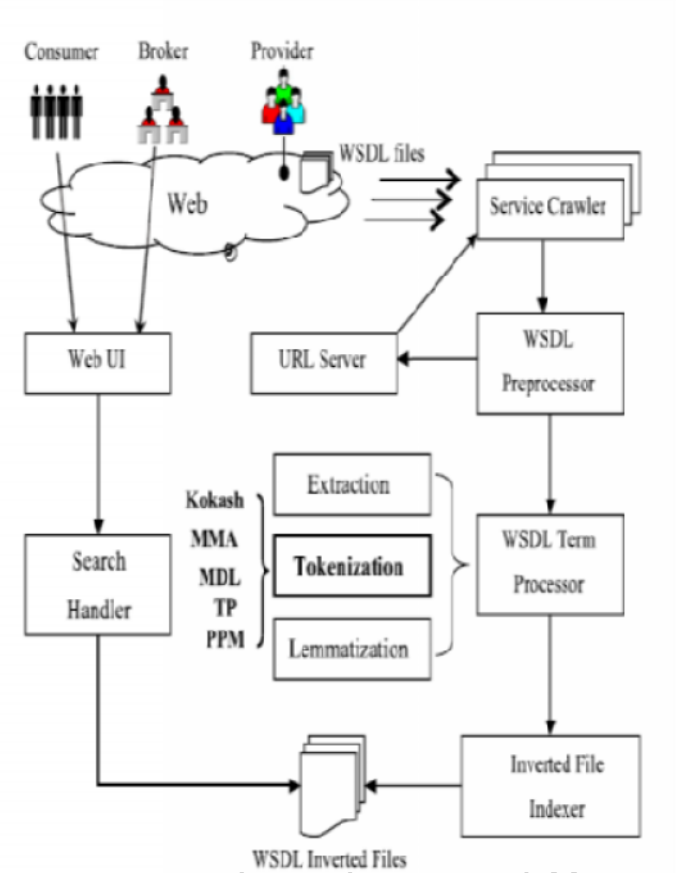

Fig 2: IR-style service discovery approach. [6]

A number of Service Crawlers, which fetch WSDL files from the Internet, can collect these service descriptions. Alternatively, they can also be collected from some well-known service datasets such as one of the WS curated catalogue .Crawlers hand over retrieved WSDL files and associated HTML files to the WSDL Pre-processor for link analysis. This yields a list of new URLs that may point to some new WSDL files. The URL server such as Pica-Pica Web Service Description Crawler assigns these URLs to an idle crawler. [6]

All retrieved WSDL files are then passed to the WSDL Term Processor, which (1) parses WSDL files and extracts important data (e.g., operations, messages, data types, etc.), (2) tokenizes extracted content into separate terms (the focus of this paper), and (3) carries out other linguistic tasks such as lemmatization and stop-word elimination, etc. Five tokenization methods (two baselines – Kokash and MMA – and three statistical methods MDL, TP, and PPM) are used in (2).

The WSDL Processor generates the 'term document', which contains separated words in a flat structure. The term document is transferred to the Inverted File Indexer. The indexer takes as inputs tokenized and lemmatized terms with their associated occurrences information in each document and generates as outputs the compiled data arrangement with pre-aggregated information optimized for fast searching. The data structure of inverted index is consistent with the notion of term-document matrix, which consists of term vectors as matrix rows and document vectors as matrix columns. The term vector is sorted that allows fast lookup operation. After finishing the document tokenization and indexing processes, the search handler component will extract the key terms from the query then it will search via extracted terms in the inverted file index , after that the most matched WSDL documents is returned based on terms frequency in the WSDL documents [6][8].

Chen's model is based on the document retrieval and it proved to increase the lookup operation but it does not contains any web service recommendation process ,so our modified model is based on adding the web service recommendation component based on user's history analysis . The modification is presented on the next section.

4. WEB SERVICE RECOMMENDATION MODEL

The Web Service Recommendation Model Approach Is Illustrated In Fig 3.IR-web service discovery in the related work part is responsible for the Web service discovery process, which remains as it is without any changes but our contribution is presented in adding the Web Services recommendation component to IR-web service discovery approach. These components will expand the WS discovery to suggest other services based on same users behaviour who chose the same web service.

Web Service Recommenation Components

A web service's recommendation component is divided into several sub processes that are shown in the following figure and each part is explained in Fig 4.

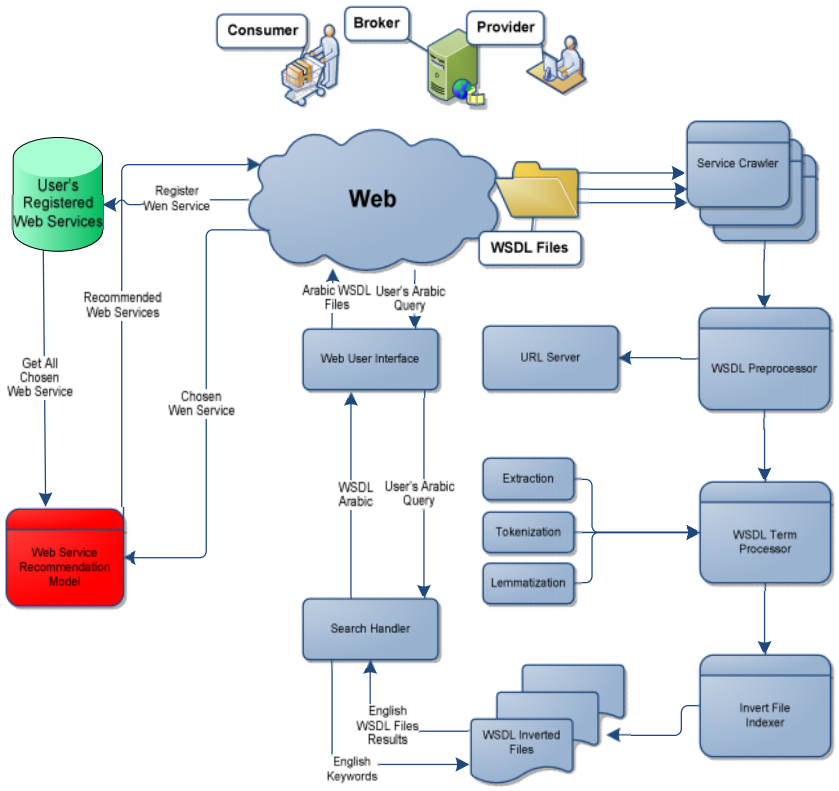

Fig 3: Web service Recommendation Model

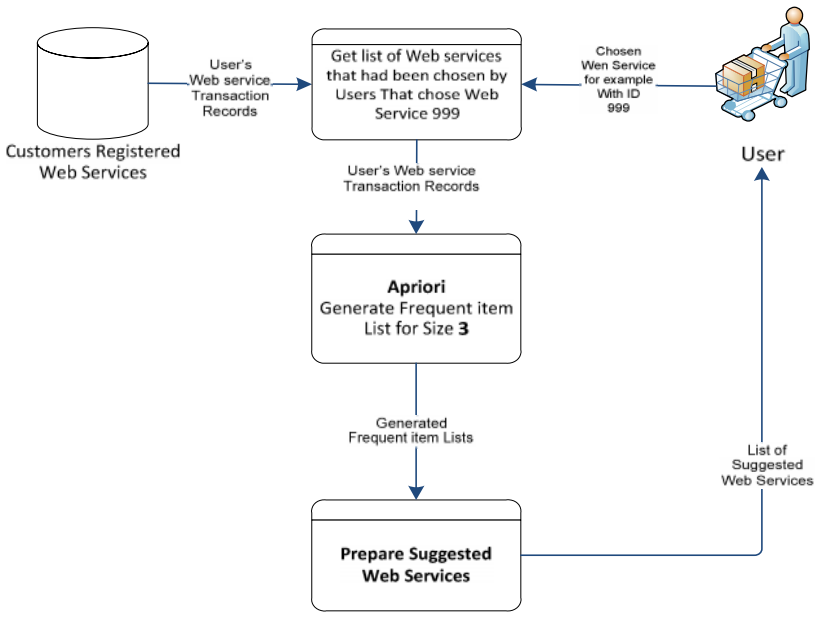

Fig 4: Web service Recommendation Model

Our proposed model starts from the point that IR-Web service discovery model ends, the IR model responsible for return the WSDL documents that matches the user's query then the user makes his decision to choose one of the returned services. The user registers his service by using BioCatalogue's APIs.

The proposed model consists ofthree main processes:

A. First Process:

The first process takes the chosen web service as an input, and then it queries the data store that contains users' data with their registered web services by the following Pseudo code:

- Get the chosen web service (A) by user U1
- Get All users that registered the web service (A)
- For each userGet his registered services.

For each user prepare the list as following:<User I,W1,….Wi>,After preparing these pairs then it is send to the next process.

B. Second Process:

A process that contains the Apriori algorithm, this process takes a list of transaction records that were extracted from the first process.

We choose the list size 3 to reduce the memory, processor and time usage and our objective to suggest one service that is the most similar to users' behavior.

As we mentioned in the section of background the Apriori algorithm is working until the item frequent list is empty, the size of pairs in frequent list is incremented by 1until the list is empty, we put our stop condition is that:*The items count in each Pair is three*

In addition, the final frequent list will be as following example:

L= {{A, B, C}, {A, D, F}…….}
Note: the chosen service is A

Then the process sends this list to the next process.

C. Third Process:

This process computes the confidence form each pairs with the condition that all we know is A like:

For {A, B, C}:

Confidence=A ➡ BC and so on.

Then the pair with the greatest confidence value it will send to the user as follow:

Suppose {A, B, C} is the winner pair: then the process suggest the services B and C to the user, then he will decide if he will use it or not.

5. EVALUATION

For evaluate our proposed model we build an experiment by using the BioCatalogue curated Web Services that has the following abilities:

- Search keyword.
- Input and output data search.
- Display all details about web service.

The previous functions are available on the BioCatalogue.com website through a defined web interface, also there are an available API's to access all BioCatalogue.com searching, filtering, browsing data and WSDL documents.We defined 10 users' profiles and registered them in the BioCatalogue, and for each user we searched for a service and registered it, the following table shows each user with his registered service.

In this point, we will display the steps for recommending web services for the first user and table 6 shows the rest of users with their decision to register on it.

Web service recommendation for of user1:

A. The IR-web service discovery returned a web service titled by "Convert PDF to Text"
B. The BioCatalogue API returned a list of users' transaction that chose Convert PDF to Text" shown in table 3, and table 2 contains symbol of each service to make the calculations easier for readers.
C. Applying the Apriori algorithm with confidence 70%, min_support2, with 3 item size, with existence of "Convert PDF to Text" and eliminate other lists that don't satisfy this condition.
D. Table 4 shows frequent lists with size 2 items
E. Applying the Apriori algorithm with confidence 70% lists in Table 5A and Table 5B.
F. Display the recommended web services to users.

Table 1: Users and their registered services

No	User ID	Registered Web Service
1	User1	Convert PDF to Text
2	User2	Cleaning Text from unwanted classes
3	User3	Text search and retrieval from large databanks
4	User4	Protein Sequence Analysis on pfam
5	User5	Named Entity Recognition
6	User6	Biomedical Named Entity Recognizer
7	User7	Document Discovery
8	User8	Document Similarity
9	User9	Document Clustering
10	User10	Classify text
11	User11	JDispatcherService

12	User12	PeptideLocator
13	User13	phenomine
14	User14	MutalyzerService
15	User15	etandem

Table 2: Services and their symbols

No	Service Name	Symbol	SupportCount
1	Convert PDF to Text	A	9
2	Image Retrieve	B	7
3	Text search and retrieval from large databanks	C	6
4	Similarity sequence databases	D	2
5	Chemical text mining	E	2

Then as shown in Table 5B the item list that satisfy our conditions and the confidence threshold is A=>B, so the output of our model is:

Recommend Service B for User1Service B titled by
"Image Retrieve"

Table 3: Users Transactions

No	Transactions
1	A, B, E
2	A,B, D
3	A,B, C
4	A, B, D
5	A, C
6	A,B, C
7	A, C
8	A, B ,C, E
9	A, B, C

Table 4: Frequent Item lists with size 2

No	Itemsets	Support	Decision
1	AB	7	OK
2	AC	6	OK
3	AD	2	OK
4	AE	2	OK
5	BC	4	Eliminated for Not Existence of A
6	BD	2	Eliminated for Not Existence of A
7	BE	2	Eliminated for Not Existence of A
8	CD	0	NO
9	CE	1	NO
10	DE	0	NO

Table 5A: Frequent Item lists with size 3

No	Itemsets	Support	Decision
1	ABC	4	OK
2	ABD	2	OK
3	ABE	2	OK
4	ACD	0	NO
5	ADE	0	NO

Table 5B: Frequent Item lists with size 3 with Confidence

No	Itemsets	Confidence	Decision
1	A=>BC	4/9	NO
2	A=>BD	2/9	NO
3	A=>BE	2/9	NO
4	A=>B	7/9	OK
5	A=>D	6/9	NO
6	A=>E	2/9	NO
Note: All other Itemsets are eliminated because we put the existence of Service A a basic condition			

Table 6: Users and Their Recommended Web Services

No	Users	Recommended Web Services	User's Decision
1	User1	Image Retrieve	Yes
2	User2	Cocoa	Yes
3	User3	Bio Labeler	No
4	User4	Concept Recognizer Service	Yes
5	User5	Document Similarity	Yes
6	User6	Get Amino Acid Sequence	Yes
7	User7	Cocoa	Yes
8	User8	Classify text	No
9	User9	Concept Recognizer Service	No
10	User10	NeMine	Yes

The percentage of users' satisfaction = (number of agreed)/ (Total number of users)*100

= (7/10)*100

= 70%

6. CONCLUSION

Recommendation systems is a powerful tool to increase the sales of products and increase the customer loyalty , from the same view point we proposed our web service recommendation model to suggest more web services to users. This model significantly enhances user's satisfaction thus increases average basket value and user's lifetime values.

Applying the Apriori algorithm has some limitations like:

- Large number of iterations.
- Should put support and confidence threshold[2]
-

For getting over this limitations we focused on the selected service by user the reduced the iterations number of data and it stops on the item list size 3 that made the execution life time is lower than the traditional one.

Our future work to merge the proposed model with our previously proposed model "Cross language information retrieval model".

7. REFERENCES

[1] JianfengHu,Bo Zhang, Product Recommendation System, in:CS224W Project Report, 2012.
[2] Yang GongXin, The research of improved Apriori mining algorithm in bank customer segmentation, in 2nd International Conference on Computer Science and Electronics Engineering, 2013.
[3] Chun-Kit Chui, Ben Kao, and Edward Hung,Mining Frequent Itemsets from Uncertain Data, 2006.
[4] Othman Yahya, Osman Hegazy, EhabEzat, An Efficient Implementation Of Apriori Algorithm Based On Hadoop-Mapreduce Model, IJRIC, 2012.
[5] Carole A. Goble1, Khalid Belhajjame1,BioCatalogue: A Curated Web Service Registry for the Life Science Community, 2009.

[6] Chen Wu,WSDL term tokenization methods for IR-style Web services discovery,Science of Computer Programming,2011.
[7] Han J. &Kamber M. Data Mining Concepts and Techniques. San Francisco, CA, Elsevier Inc,,2006.
[8] TorkyI.Sultan,Ayman E. Khedr,Fahad Kamal ALsheref.Cross Language Information Retrieval Model For Discovering Wsdl Documents Using Arabic Language Query. IJACSA,2013.

AUTHORS

1. Prof.Dr.TorkeyI.Sultan
Professor in Information Systems Department, Faculty of Computers &Information Helwan University, Cairo, Egypt
tsultan@consultant.com

2. Dr. Ayman E. Khedr
Assistant professorin Information Systems Department, Faculty of Computers &Information Helwan University, Cairo, Egypt
ayman_khedr@Helwan.edu.eg

3. Fahad Kamal ALsheref
Lecturer Assistant in Management Information Systems department , Modern Academy, Cairo , Egypt
dr.fahad@hq.helwan.edu.eg