Comprehensive Understanding of Intelligent User Interfaces

Sarang Shaikh
and M. Ajmal Sawand
Sukkur Institute of Business Administration,
Sukkur, Sindh, Pakistan

Najeed Ahmed Khan
NED University of Engineering and Technology,
Karachi, Pakistan

Farhan Badar Solangi
Federal Board of Revenue,
Islamabad, Pakistan

***Abstract*—This paper represents basic discussion for one of the latest advances in the technology, known as *Intelligent User Interface (IIUI)* which is a combination of two major fields of computer science, namely, HCI & Artificial Intelligence. The paper first discusses basic definitions, motivation to this research and UIMS *(User Interface Management System)* along with example of user interface models to understand user interfaces in detail. The four major classes *(with their examples)* of these interfaces have been taken as a method for this study. The overall discussion summarizes some basic principles used to create these interfaces, components that are important in the generation of IUIs and decision making process in IUI for the reader to understand working of IIUIs.**

***Keywords*—Intelligent user interfaces; HCI; artificial intelligence; IIUI**

I. Introduction

Intelligent User Interfaces refers to the study and use of the two major fields of the Computer Science that are *Human Computer Interaction (HCI)* and *Artificial Intelligence (AI)*. HCI provides efficient user interfaces designing techniques and AI is used to automate or to build intelligence in those interfaces. Basically the term *IIUIs* suggests that an interface or interface like system, which is interacting with the user, must generate some output that the user considers it an intelligence, e.g., if the user dont know how to copy files in the windows operating system then in the windows help there must be assistance available for this when user searches for this, if user clicks the wrong button then an automatic message must be appeared for the help of the user, if the user has previously selected some options on the interface based on those options the system must understand the interests of the user and generates output according to it.

Moreover, the intelligent systems with interfaces cannot be called as IIUIs because they are intelligent in machine point of view but not in users point of view. IIUIs directly deals with users goals, requirements, efficiency, effectiveness, etc. these type of interfaces are quite frequent now a days due to advancement in the technology. These are basically dynamically generated, run time decision makers for the users according to users input, etc. Below in the paper, different information for these type of interfaces in provided, that might be helpful in understanding this new technology. Now the road map for rest of the paper. Section II will explain state of the art, Section III will discuss section II in more detailed way, Section IV will explain decision making process in IIUIs, Section V will conclude the whole study and Section VI will suggest some future work for this study.

II. State of the Art

Since inception of the computers, these machines were only used for business and mathematical calculations. These were performing efficiently for such selected functions because the inputs were limited and constrained based. As computer is information processing tool so it is important to make computer efficient and effective in processing information and interpreting user inputs correctly. For this, the need of the user interface had been possibly a main issue in the computers because computer machines only understand binary (machine) code but that code cannot be interpreted by humans easily. If human tries to input that code, it is time consuming task. High level languages solved this problem to some extent for humans but human still need an interface to communicate (to provide) correct input for the correct functions in the computers.

Basically user interfaces is a complete study in the field of *Human Computer Interaction*. The very first type of interaction was input through punch cards and output was printed on some paper or anything else, after that keyboards were used for input and monitors were used for output. At that time the first interface was created known as *Command Line Interface (CLI)*. That was a command based interface in which user inputs the command in textual format and machine understands it by converting it into machine code. That interface worked quite best at that time for simple processing or calculations because the inputs were limited and interaction was simple. As time passes, the computer became more general and application programs like word processors and spread sheets were developed. It is a clear matter of understanding that for such type of programs interaction through CLI is quite difficult so a new and advance way of interaction was developed known as *Graphical User Interface (GUI)*. In which mouse and icons were used to interact.

Today, the technology is being so advanced that task like image processing, face detection, emotion and speech recognition, body behavior detection, automation of machines, etc is demanding for a new type of interactive user interfaces that can provide facility to the human and the machines to interact in these technologies. No doubt, GUIs are the most effective way of interaction but they are static and fixed interfaces. Advance technologies are dynamic, they are working in real time scenarios so they also require real time and dynamic interfaces to interact. So with the help of Artificial Intelligence and Human Computer Interaction such type of interfaces are being created known as *IIUIs*. These interfaces change their behavior according the real time scenario on which it is implemented. In [1], authors have discussed the historical background and presented introduction of IIUIs, moreover they have tried their level best to make these interfaces understand.

A. Understanding User Interfaces

First of all, we need to understand what is an interface? *An interface provides a means of communication between two or more objects* i.e., in computer systems the command line or graphical user interfaces helps the human to communicate with machine by giving the inputs, at the same time it helps the machine to give output to the human using some hardware. Similarly, we need to cope up what is intelligence means in this context? In the perspective of interfaces, *the intelligence might be in predicting what the user wants to do, and presenting information with this prediction in mind*. It means to use information in a proper and intelligent way.

Since, the intelligence in the interfaces varies from user to user because for every user there are different requirements and understanding levels. Intelligence is just like a future step which is one step forward from us, when we reach to it intelligence finishes. Point here is, what components of the interface should be intelligent. The main points are discussed below:

1) **System Functionality:** The interface must have the information about the user tasks and how to interact with the system. With this information, the interface can be intelligent.
2) **Intelligent about the user:** The interface must be intelligent about the user by using user models. This is done when system uses different modes for input and output to the human like voice, visual, etc.
3) **Interface must be sensitive about needs of user:** By using different user models, the system must understand on run time that what is the need of the user? One specific case can be system detects that user wants help while attempting some particular task. For e.g. if the user is continuously pressing the wrong button then system must generates some hints about the correct button or highlight the correct button which should be pressed or generates some messages [2].

B. User Interface Management Systems (UIMS)

Every user interface management system uses some user interface models because these models create notations for UIMS which describes the user interfaces and their implementations. There are various types of techniques are being developed for explanation of user interfaces, these techniques are divided into two major parts. One is about the design of the interface and another one is about the implementation of the interface. Design techniques are not of such importance because they just show the thoughts of a designer but implementation techniques are most important because they produce the executables for the interfaces. The techniques used in UIMS represent the no. of interfaces that can be created by UIMS, for creating general UIMS it is required that the design techniques must support max no. of user interfaces. Basically the term UIMS represents also an user interface.

The selection of the user model for UIMS is very important because user model decides what service or features should be provided in the user interfaces. For example, if the user model is divided into different components then UIMS should have to apply design and implementation tools for each component. This defines that before designing the UIMS for user interface, the user model should be designed first. The best user model ever known is **Seeheim Model**. The best thing about this model is that it is a generalized model; it can be integrated with any UIMS without having any issues of design or implementation for that model. This is helpful in studying just the model not different UIMSs [3].

This model divides the design and implementation of user interfaces into three components:

1) **Presentation Component:** Deals with the physical view of the user interfaces, including input and output devices, interaction and display techniques, screen arrangement. It deals directly with the hardware devices and it serves as a lexical part of the interfaces.
2) **Dialogue Control Component:** Deals with the communication between user and the computer system. This component is responsible for structure of the inputs, commands, dialogues provided by the user. This is known as syntactic level of the user interfaces.
3) **Application Interface Model Component:** This is a logical part of the user interfaces, in which the interaction between the user and all remaining application procedures is handled. The information processed by the user is understandable to the machine in this component and according to it the computer works and generates the output. Fig. 1 explains working of above mentioned basic components of UIMS.

All of these components communicate with each other by passing tokens same as in compilers [4]. The token contains the name and values; the name of the token defines the type of the token. The token going from user to the application program is called an input token whereas the token going back from application program to the user is called an output token. For e.g. while interaction performed in the interface like text box, drop down menu or button, the token names can be text box, drop down and button. The values can be those values which are entered by the user as an input into the interface.

C. Principles for creating IIUIs

When combining automation, i.e., artificial intelligence with user interfaces generates some key problems. E.g. poor assumptions for users goals and requirements, insufficient consideration for automated tasks, poor timing for tasks and wrong analysis of the automated decisions and actions. For this some principles have been defined which helps in understanding user requirements and creating effective interfaces.

1) **Developing significant value added automation:** To provide intelligent interfaces, which provide original value over solutions achievable with direct manipulation.

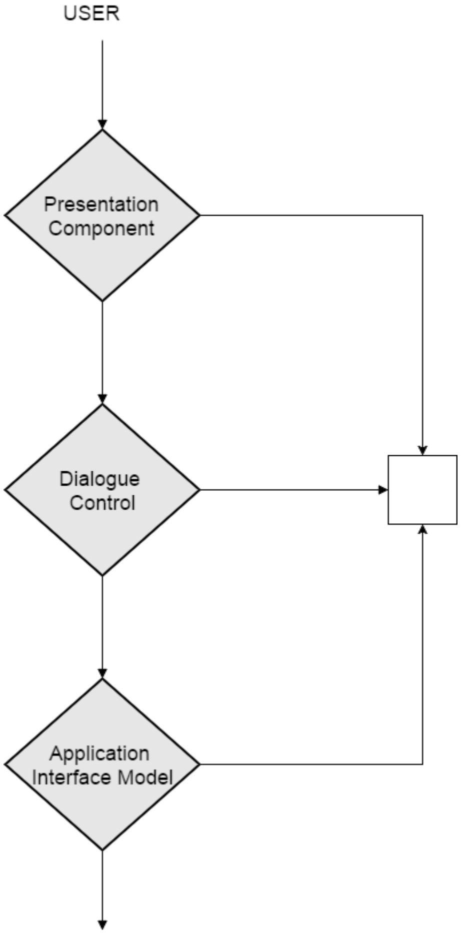

Fig. 1. User interface model [1].

2) **Considering uncertainty about users goals:** Computers are often uncertain about the goals of the users. So different ways for inferring the intentions of the users can employed by the system.
3) **Considering the status of users attention in the timing of services:** The timing of the automated actions can be major factor as cost for the actions. Agents should apply users attention models to check for timing costs and benefits.
4) **Inferring ideal actions in lights of costs, benefits and uncertainty:** Automated actions taken against users goals are directly related with costs and benefits. The cost for those actions can be controlled by considering some expected values / assumptions for the costs.
5) **Employing dialogues to resolve key uncertainties:** If a system hasnt any knowledge about users intentions tasks, it should be able to involve user in a detail dialogue to understand users actions while keeping costs minimum.
6) **Allowing efficient direct invocation and termination:** In some cases of uncertainties, system can take poor decisions about an intelligent interface. There should be several ways to provide options to the user during communication to start or terminate or control that automated service.
7) **Minimizing the costs of poor guesses about actions and timings:** The design and implementation of intelligent interfaces must be in a way such that there must be minimum costs for poor decisions taking by the system.
8) **Continuing to learn by observing:** Intelligent interfaces must be designed to work with users needs and they have ability to adopt new users goals during interaction.

D. Components of Intelligent Interfaces

Below are the components defined for intelligent interfaces, which helps in constructing efficient interfaces and also to understand above described principles [5].

1) The User Model:
Definition: This is an important part of the intelligent interfaces which creates the bases for the design and implementation of the interface. Without this component the system would not have any information regarding what the user will do? It also describes how to present the information, how and what type of help interface has to give and how the user will interact with the interface?

Characteristics:

- The system tries to adjust its behavior according to the users.
- The system takes responsibility to confirm the user-system communication.
- System ensures that various users with various requirements can use the interface.

Importance of the User Models: It helps the interface to decide what to ask, how to ask from the user, how to interpret the responses generated by the user and how to resolve ambiguity created by users actions. It helps the interface to provide help and advice to the user, to evaluate relevance of the tasks performed by the user, to recognize and correct misconceptions in the users mind. It helps the interface to understand users information seeking style by recognizing users goals and plans. It helps the interface to provide output to the user, deciding what to present and how to present.

2) The Multimodal Communication:
Definition: The use of different ways of communication in the interfaces is known as multimodal communication. These ways can be mouse, natural language entered by typing or voice, gesture, point on screen using pen, etc. This component always requires help of user model because depending on input and output of the user the way of communication is selected whether it should be textual, visual or vocal.

Characteristics:

- It enables the users to access the system naturally by using advance interaction ways like gestures or voice input.
- It gives the user more freedom such that the user is not restricted to sit in front of the system.

Importance of the Multimodal Communication: It is most important that the situations like when user is not able or user don't want to use all of his senses for interaction. For example: If user has to closely look to the system for long duration, so the visual output would be not a good idea. Instead of this the voice or sound output would be better so that user can pay attention for long time.

3) Plan Recognition:
Definition: Plan recognition is used in intelligent interfaces to decide what user plans to do. It uses the system information, the user model and actions of the user to recognize plans of the user. For example: If there is a taxi automation system, the user wants to select the vehicle for 04 peoples and send it. When user will select no. of passengers, the interface will automatically suggest that which vehicle will be suitable? Or it will notify the user that you will send car for 04 peoples. So you want to send this time or not?
Characteristics:

- To recognize plans of the users before the user enters any goal or intention.
- To predict the user actions and provide guidance along the way.

Importance of the Plan Recognition: To automate user actions so that user doesnt want to select each and every option every time. To use past history of users actions so that it can inform the user for taking correct and efficient decisions.

4) Dynamic Presentation:
Definition: This component helps to display data for different users in different ways in which they want. The display of the data should be in an understandable way. Display depends upon the user model of the user such that what is requirement of the user. Different users have different requirements.

Characteristics:

- This component allows the display of the data either not to be in detail or not to be so short. But it must be in an effective style according to the user demands.
- This helps for display of different types of data in different formats.

Importance of the Dynamic Presentation: This helps in displaying data in an intelligent manner. For users who want to see just one table, this will show it. Instead, if user want to see 10 tables then that will also be displayed dynamically on run time.

5) Natural Language:
Definition: This is one of the best techniques to create effective and automated interfaces. The user selects the options or enters the inputs in simple natural language that he/she knows, instead of remembering commands for each task and input. The user enters the task what he wants to happen.
Characteristics:

- It will completely remove the GUI based style because there will be no more menus or lists for selecting different actions or performing tasks.
- There will be no list of commands for the system.

Importance of the Natural Language: It will help the users to communicate directly in their own language. Users dont need to memorize commands or huge menus creation will be removed. Communication will be simple and effective.

6) Intelligent Help:
Definition: Intelligent help presents the users helps which they want at the particular time or situation. This component is very useful in complex systems where asking for help returns in more information than necessary or information that is not according to the users needs.
Characteristics:

- Some plan recognition techniques can be used in intelligent help to acquire about users tasks and then to provide specific help according to the query.
- This feature requires the knowledge for application functionality.

Importance of the Intelligent Help: It is important feature for intelligent interfaces because it will help the users in an efficient manner by providing automatic help during any communication problem. It will generate error messages or help messages or hints to reduce the extra time taken by users [6], [7], [8].

III. Discussion

This section involves the discussion about some IIUIs studied and evaluated as an example from various sources. Basically there are four type of general interfaces or classes defined as an example below:

1) Direct Manipulation Interface Adaptation: In these type of interfaces, the user models are used to predict the goals or tasks of the user or to identify the patterns of the users inputs. Due to these type of predictions the intelligence can be easily adapted in four ways defined below: [3]

- **Speculative Execution:** The predicted users actions are already started to be carried out such that when user inputs a command or action the earlier phases for completing that command is already done or on the way towards completion, due to this phase.
- **Pattern Completion:** If the several input behavioral patterns are clear and same then different commands can be combined to create a big or general command.
- **Rapid Issue:** The commands that are used to execute several predicted actions or action that are performed by the user can be allowed to be accessed by the user in a rapid manner.
- **Assistance:** The system can offer help by using the knowledge it have about users goals or tasks.

Below is the practical example of these type of interfaces created and implemented by Microsoft and the UNIX systems. The very first example is *Davison and Harisch* system for predicting UNIX commands [9]. This system is based on Probabilistic User Model, which contains likeliness for the commands which are previously issued commands. This system outputs the top five predictions according to the user input for commands. This system proved to be efficient 75 percent. The second one example is MS Windows 95 and MS Office 97. In MS Windows 95, there were separate user accounts facility, in which according to the input account of the user whether guest or any other account, the privileges of providing menus, permissions, etc was controlled. This was a type of personalized adaption. The recent files or human input history was easily and rapid accessible by the user. The menus were designed for the ease of the user with each option and rapid access. The help was available for each task related to the users input.

2) Informative Interfaces: Now a days, the use of data or information is increasing day by day, this is possible because of www or the internet. This type of interfaces deals with intelligence with the data. Intelligence of the data basically defines filtering in the data that which data should be display by the interface to the user according to its query [10], [11]. There are basic two main methods for filtering information:

- **Content Based Filtering:** This type of filtering uses objects descriptor relationship over the information, such as documents words relationship. It observes that which type of information is demanded by the user and after providing that information, it again observes that which type of data user is selecting from that provided information. An example for this type of interface is *SySkill and Webert*, this system recommends the web pages on a given topic provided by the user according to its likeness.
- **Collaborative Filtering:** This type of information filtering is used when objects are available but the descriptors are not or it can be clearly defined that when the interests of the user are not clearly defined. Firstly this filtering analysis the interests of the user, then it analysis the interests of other user profiles also. Finally, based on both of these analysis it suggests the information or data on that given topic asked by the user. This basically works on current user as well as other profiles interests. This type of filtering is quite helpful in film or music or art data filtering. An example of this type of interface is *FilmFinder*, that provides users the films that they might like.

There is a common informative interface based system known as *Langs News Weeder*. This system provides information about stories and web pages. Content based filtering is used to predict the interests of the current user accessing that system and collaborative filtering is used to provide general information on that topic from others profiles also of the same interests.

3) Generative Interfaces: This type of interface is used to generate new data according to previously recorded data values. The main purpose of this system is to improve the quality of the data by reducing data entry time for a human; this will surely help the beginners doing any task in case of performance. An example is provided to understand further this interface, the system is known as *Clavier* [12]. This system basically works on arrangement of air craft parts. When it is provided with lists of parts, from the past successive arrangements it lists out the most efficient arrangement which was perfect and containing all parts. Another example of this is *Hermens and Schlimmers Generative Interfaces*. This system is used to fill out repetitive forms. Each time the user fills a form the new form is generated on the basis of previous forms data with default values from previous data. Previous data is just like a training for generative interfaces [13].

4) Programming By Demonstration (PBD) Interfaces: This type of interface is used to communicate simple repetitive programs or tasks generated by user without implementing that programs. The user just has to tell the tasks to the system and all the work is done by the system. The basic example for this type of interfaces or systems is a simple *Macro Reader*, when user clicks the play button; it continuously records until the user clicks the stop button. To create PBD systems, there are few things that must be understood well [14].

- **Representation of User Tasks:** This is concerned with the input data by the user to the PBD system. The data should be accurate to be processed such that system can create meaningful information from that input. For example: by clicking a mouse or menu item must also send some textual information to process to the system, rather than just to send co-ordinates of the clicks.
- **Representation of Predictions:** This is important in how to represent the predictions done by the system to the user. The best way to solve this is to provide information to the user which is easily under stable by the user. For example: when user is going to click a button, the message is generated by the system prediction that you are going to click Button ID 01 to process. This is quite awkward in case of user, instead of this the button must be highlighted or some graphical change when user is going to click that button. This is more supportable by the user.
- **The Domain Knowledge:** The knowledge of the user tasks that which type of actions users can take. For example: in case of macro recorder, the user can start recording, stop recording, play recording, re-recording, save the recording, etc.
- **Termination Conditions:** This is about performance or efficiency of the PBD system, that how much it can support user interaction / usage in case of providing information. For example: on each click to provide the user each and every message, is not right. The user will be get irritated on seeing message on each iteration. For example: while recording a voice, if user speaks a letter, then system will generates message that you spoke a letter, proceed to further. This is not an efficient way.

IV. Decision Making in Intelligent Interface Systems

The IIUIs have to make several decisions during communication, the level up to that interface is considered to be intelligent depends upon its decision making during run time communication with the user, the tasks performed, nature of the application, etc. All of these factors are combined together and called adaptive of the IIUIs. The more intelligent interface is the more adaptive [15]. The adaptation strategy can also be called as decision making process. There are some attribute about the decision making process which can be helpful for a intelligent interface system [16], [17], [18]. Fig. 2 elaborates process of decision making in IIUIs. Furthermore, this process includes some of basic decision making factors discussed below:

1) **What to Adapt:** This refers to the adaption or acceptance of the content of the information, sequences of the interaction, objects and techniques of the interactions. That is why it is called user computer interaction or known as *Adaptation Constituents*.
2) **When to Adapt:** During the run time adaptation in user computer interaction, it is important to analyze or monitor the time because the adaptation decisions are based on time. These are known as *Adaptation Determinants* i.e., requirements of the tasks, users sensory load, etc.
3) **Why to Adapt:** This depends upon the goals of the user; designer has to adapt such interfaces for communication according to requirements of the user. This is commonly known as *Adaption Goals*.
4) **How to Adapt:** Adaptation is done by some proper way known as *Adaptation Rules*. These rules combine the adaptation constituents, determinants & goals to create interfaces according to it.

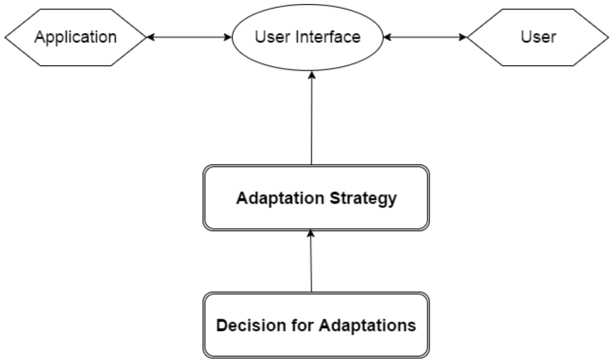

Fig. 2. Adaption at the highest level [2].

Although these attributes are quite important in decision making process, but not efficient in this way. Because these attributes are just used to create adaptation rules nothing else. In the case of existing systems, this approach fails because the adaptation rules become hard coded due to these attributes. They cant be used for multiple interfaces at a time. We have to change complete rules for another interface to be created. This will surely make the decision making process very inefficient [19].

The adaptation determinants that are used in decision making usually involves characteristics of user and information and task that is being to be performed. Different models like task user model, dialogue model and application model are used to manage these characteristics. Also based on the requirements of the application different determinants can be effectiveness, efficiency, graphic designs, etc. [20], [21], [22]. This is not fix that I any one set determinants for a application then that cant be changed. For every researcher or developer there are different determinants and their characteristics for a same application. User models are being used to represent determinants and their characteristics.

The adaptation rules are used to operate a system, these are basically hard coded and cant be changed dynamically. So it is quite a big issue that rules cant be modified, until and unless they are not to be re created again. These rules guide a system at different levels of human computer interaction. For example: If the task / sub goal requires spatial information, then it is better to use visual media resources, If the thing that has to be displayed is a analysis of complex structure of the system, then use network charts [23], If the number of secondary values of a object is 8 then interaction object selected must be a list box [24].

The above discussion suggests that adaptation rules are being generated on the basis of fixed determinants, constituents & goals. This makes it specific for a system, there rules are not flexible to be applied on any system. If any one of them needs modification then all of the four things will be modified. Because they all are inter linked with each other. As explained diagramatically in Fig. 3.

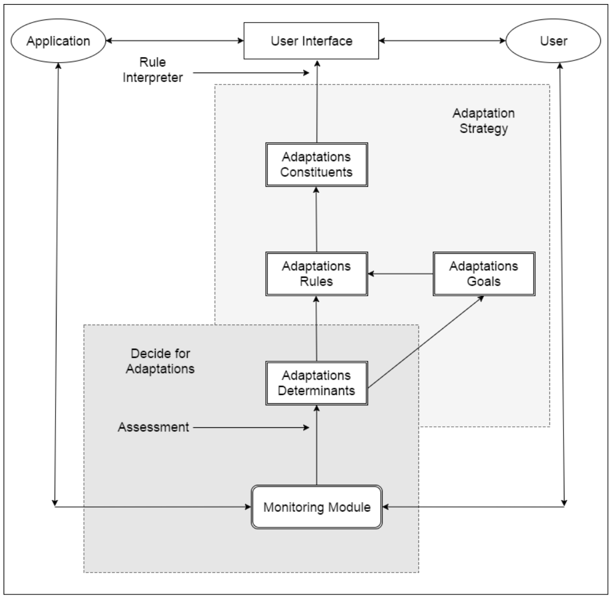

Fig. 3. Decision making in IIUIs [2].

V. Conclusion

Today in the era of technology there is a need for such type of interfaces because today computers are not limited just to the office or scientific computing or calculation, but it has become an important part of our lives. We, humans, are entirely dependent on each task on this machine whether it is to type a document, to drive the car, to study something, to calculate something, to make presentations, the internet; a huge invention of computers, online shopping, online music, etc. Also in the field of medical sciences, surgery, analysis of the internal body, to cure serious diseases, etc. In every field of human life, this machine is involved. If up to such level, this machine is involved then the interaction must be as efficient as possible. CLI or GUIs are also very good interaction styles, but they are static and limited. They cant perform multi-tasking like IIUIs or systems. When the user has to work with speech recognition, face detection, analysis of information, intelligent searching like Google, natural language processing like Google translator. When the user doesn't want to repeat the same options every time when the user trying to see it's previous tasks & many tasks like these. So, nowadays there is a huge need for different intelligent interfaces that can meet user criteria because the expectation of the user towards the technology is increasing day by day. Finally, concluded that the need of intelligence in machines is increasing day by day so advances in the Artificial Intelligence & Human Computer Interaction is quite needed.

VI. Future Work

As this paper is a comprehensive and introductory survey of IIUIs, so one of the major future work is to make this paper as a base and find a research problem. The research problem can be either related to applications of these interfaces or further enhancement of these interfaces in terms of studies. Furthermore, this paper will create basis for other researchers of same domain for work in coming future.

References

[1] Ross, E., *Intelligent user interfaces: Survey and research directions*, University of Bristol, Bristol, UK, 2000.

[2] Horvitz, E., *Principles of mixed-initiative user interfaces*, Proceedings of the SIGCHI conference on Human Factors in Computing Systems. 1999. ACM.

[3] Sullivan, J., J.W. Sullivan, and S.W. Tyler, *Synergistic Use of Direct Manipulation and Natural Language*, IIUIs. 1994.

[4] Green, M. (1986), *A survey of three dialogue models.*, ACM Transactions on Graphics (TOG), 5(3), 244-275.

[5] Keolle, D., *IIUIs. 2000 [cited 2017-05-05];*, Available from: http://web.cs.wpi.edu/Research/airg/IntInt/intint-paper-intro.html.

[6] Tyler, S.W., et al., *An intelligent interface architecture for adaptive interaction*, Readings in IIUIs. 1991. ACM

[7] Chin, D.N., *Intelligent interfaces as agents*, Readings in IIUIs, 1998: p. 358.

[8] Kass, R. and T. Finin., *General user modeling: A facility to support intelligent interaction*, Readings in IIUIs. 1991. ACM

[9] Davison, B.D. and H. Hirsh., *Predicting sequences of user actions*, Notes of the AAAI/ICML 1998 Workshop on Predicting the Future: AI Approaches to Time-Series Analysis. 1998.

[10] Langley, P., *Machine learning for adaptive user interfaces*, KI-97: Advances in artificial intelligence. 1997. Springer.

[11] Langley, P., *User modeling in adaptive interface*, UM99 User Modeling (pp. 357-370). Springer Vienna.

[12] Hinkle, D., & Toomey, C., *Clavier: Applying Case-Based Reasoning to Composite Part Fabrication*, IAAI.

[13] Hermens, L.A. and J.C. Schlimmer., *A machine-learning apprentice for the completion of repetitive forms*, Artificial Intelligence for Applications, 1993. Proceedings., Ninth Conference on. 1993. IEEE.

[14] Cypher, A., & Halbert, D. C., *Watch what I do: programming by demonstration*, MIT press.

[15] Szekely, P. (1989), *Structuring programs to support intelligent interfaces*, (No. ISI/RS-89-231). UNIVERSITY OF SOUTHERN CALIFORNIA MARINA DEL REY INFORMATION SCIENCES INST.

[16] Zhuang, S., *Adaptive User Interfaces [cited 2017-05-10];*, Available from: https://susiezhuang.wordpress.com/2012/06/08/adaptive-user-interfaces/.

[17] Dieterich, H., et al., *State of the art in adaptive user interfaces*, Human factors in information technology, 1993. 10: p. 13-13.

[18] Stephanidis, C., C. Karagiannidis, and A. Koumpis, *Decision making in IIUIs*, Proceedings of the 2nd international conference on IIUIs. 1997. ACM.

[19] Arens, Y., Hovy, E. H., & Vossers, M. (1992), *On the knowledge underlying multimedia presentations*, (No. ISI/RR-93-370). UNIVERSITY OF SOUTHERN CALIFORNIA MARINA DEL REY INFORMATION SCIENCES INST.

[20] Sutcliffe, A. and P. Faraday, *Designing presentation in multimedia interfaces*, Proceedings of the SIGCHI Conference on Human factors in computing systems. 1994. ACM.

[21] Mackinlay, J. (1988), *Search architectures for the automatic design of graphical presentations*, ACM SIGCHI Bulletin, 20(1), 76.

[22] Chappel, H. and M. Wilson, *Knowledge-based design of graphical responses*, Proceedings of the 1st international conference on IIUIs. 1993. ACM.

[23] Bernsen, N. O. (1992), *Matching information and interface modalities: An example study*, Working Papers in Cognitive Science.

[24] Bodart, F., & Vanderdonckt, J. (1994, August), *On the problem of selecting interaction objects*, BCS HCI (pp. 163-178).