IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Volume-4, Special Issue-1 E-ISSN: 2347-2693

# Space Reduction using String and Synonym Matching Algorithm (SSMA)

**Papri Ghosh***
Computer Science & Engineering
Surendra Institute of Engineering & Management
Siliguri, India
paprighosh06@gmail.com

**Tejbanta Singh Chingtham**
Computer Science & Engineering
Sikkim Manipal Institute of Technology
Majitar, India
chingtham@gmail.com

**Mrinal Kanti Ghose**
Computer Science & Engineering
Sikkim Manipal Institute of Technology
Majitar, India
mrinal.ghose@smu.edu.in

***Abstract*—The Command Based System (CBS) is one of the fields of Human Computer Interaction (HCI). In the modern era the Command Based System is enormously explored by many scientists and it is used in many fields. This study proposes to develop a new approach to build up a Command Based System which will accept a Text as an input and take the corresponding action. In this paper the String and Synonym Matching Algorithm (SSMA) is made to develop a String Command Based System (SCBS). The String and Synonym Matching Algorithm (SSMA) Algorithm focuses on the comparison between the users given string with the preloaded strings in the system database. The comparison is made in two ways i.e. a straight string comparison and the comparison between all the synonyms of the input string with the preloaded strings from the system database. The trick of matching the synonyms of the user given string reduces the space in the system database and increases the flexibility of the user to give more input with same meaning.**

***Keywords-****Human Computer Interaction (HCI), Command Based System (CBS), String Command Based System (SCBS), Unimodal System, String and Synonym Matching Algorithm (SSMA), Space Complexity.*

## I. INTRODUCTION

Human-Computer Interaction (HCI) is concerned with interactive computing systems for human and with the study of major phenomena surrounding them [1]. It can be also said that the Human-computer Interaction, which was popularized by Card, Moran, and Newell in their book, "The Psychology of Human-Computer Interaction", 1983. The authors used the term HCI in 1980 [2] for the first time It is a discipline concerned with the study, design, construction and implementation of human-centric interactive computer systems [3]. The journey still continues with new designs of technologies and as systems becomes more and more sophisticated each day, the research in this area has been growing at a very fast pace in the last few decades [4]. The concept of Human Computer Interaction (HCI), sometimes called as Man-Machine Interaction or Interfacing [5], [6], was represented with the emergence of computer, or more generic machine. Usability of a system with a certain functionality is the range and degree by which the system can be used efficiently and adequately to accomplish certain goals for certain users [7]. The available technology could also affect how different types of HCI are designed for the same purpose. Few examples are using commands, menus, graphical user interfaces (GUI) [8], or virtual reality to access functionalities of any given computer [9].

The Human Computer Interaction design seeks to discover the most efficient way to design understandable electronic messages. It involves Study, Planning and Design of the interaction between users and computers [10]. There are two types Human Computer Interaction (HCI) Architecture:

A. *Unimodal:* *Only one modality is used i.e. Visual Based, Audio-Based, Sensor-Based etc. [4].*

B. *Multimodal:* *More than one modality are used i.e. PC, Smart Phones etc. [4].*

The Command Based System (CBS) is one kind of Human Computer Interactive system where the user is giving some command to the system and the system is responding dependent upon that. In this paper first the overview of a Command Based System (CBS) and String Command Based System (SCBS) are given. Then String and Synonym Matching Algorithm (SSMA) is proposed and elaborated. After that the implementation of SSMA algorithm is given.

## II. STRING COMMAND BASED SYSTEM

The Command Based System (CBS) is one kind of Human Computer Interactive System where the user is giving commands to the system applying some methodologies and the system is responding dependent upon the command [11-14]. In every Command Based System (CBS) there is a database with preloaded commands and the corresponding actions. When the user is giving the command, the system first check whether the user given command is matched with any other commands stored in the system database or not. If the user given command is matched with the system database, then the system is taking the corresponding action of the matched command.

The working principal of the Command Based System (CBS) is similar with the Rule Based Expert System where the Rules and the Immediate Actions are loaded in the system database. The current situation is getting compared with the rules, and if matched with any rules from the system database, then the corresponding command will be taken.

There are many examples of the Command Based System (CBS) i.e. String Based Command [11][12], Speech Based Command [13][14], Gesture Based Command [15], Image Commands etc. In this paper the String Based Command System (SCBS) is elaborated. The String and Synonym Matching Algorithm (SSMA) is proposed for implementation in the next section and the SSMA is being implemented by using VB 6.0, Microsoft Access 2007 Database and the Microsoft Word Dictionary 2007 in the fourth section.

## III. STRING AND SYNONYM MATCHING ALGORITHM

The String and Synonym Matching Algorithm (SSMA) is proposed to execute two different procedures together.

- One is to compare the Input String with the Preloaded Stored String Commands in the system database. It will check directly that the Input String is fully compared with any of the Preloaded Strings or not using string comparing algorithm.
- Applying the second technique of the algorithm the proposed system will accept the synonym of the given inputted string with the Preloaded Stored String so that the system can reduce the storage memory of the database up to some extent.

ALGO: SSMA

1. START
2. Text ← Inputted String
   //Taken from the Speech to Text Technique
3. Str[] ←Return_Synonym(Text)
   // The Return_Synonym() returns the list of synonyms of the given parameter using *MicrosoftWord Dictionary* and store it in the Str[] String array
4. IF Text is Matched with any Command in the System Database
   a. THEN Take the corresponding Action
5. ELSE IF any string in Str[] is Matched with any Command in the System Database
   a. THEN Take the corresponding Action
6. ELSE
   a. Give Message *"No Action Found in System Database"*
7. END

## IV. FLOWCHART OF PROPOSED ALGORITHM

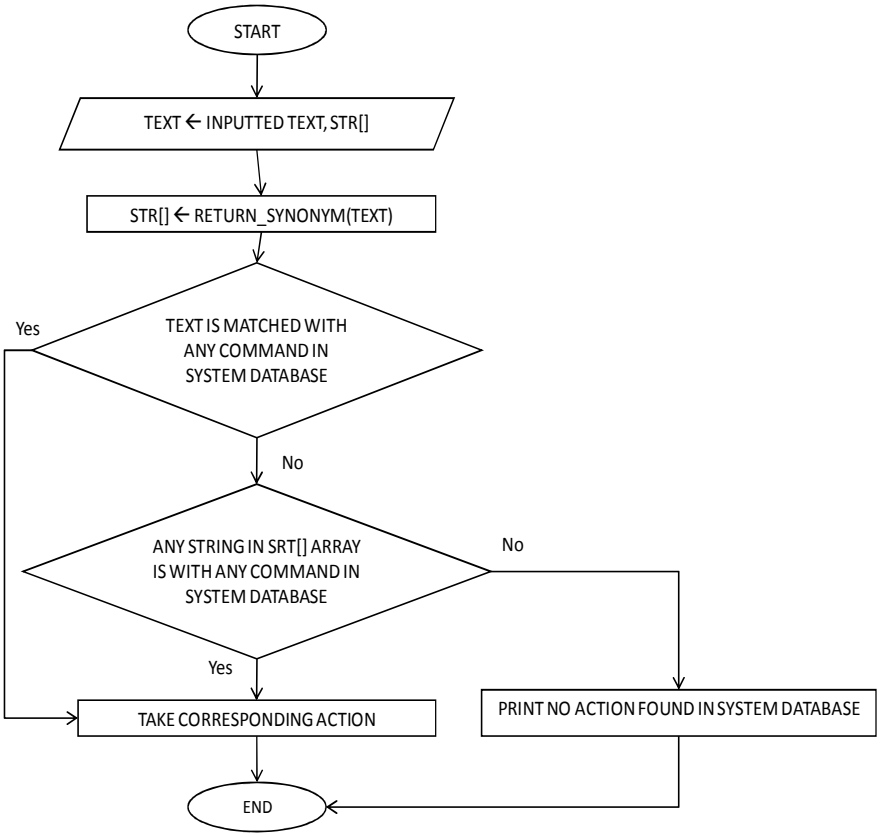


Fig.1: Flowchart of SSMA

## V. SSMA IMPLEMENTATION

Applying the String & Synonym Matching Algorithm (SSMA), one system is being implemented using VB 6.0, Microsoft Access 2003, and Microsoft Word Dictionary 2003. The system can accept the string as a command and take the corresponding action if the given command or any synonym of that given command is there in the System Database. The Fig.2 shows the system database. In the system database three attributes is being used; NO denotes number, COMD denotes Command and ACTION denotes the corresponding actions.

STRDB

| NO | COMD | ACTION | Add New Field |
|---|---|---|---|
| 1 | RUN | "explorer.exe" | |
| 2 | RESTART | shurdown /r | |
| 3 | SHUTDOWN | shutdown /s | |
| 4 | LOGOFF | shutdown /l | |
| 5 | HIBERNET | shutdown /a | |
| 6 | NOTEPAD | notepad.exe | |
| 7 | COMMAND | cmd | |
| 8 | CALCULATOR | "C:\Windows\system32\calc.exe" | |
| 9 | WRITE | write.exe | |

Fig.2: Database Representation

## VI. SYSTEM SNAPSHOTS FOR SSMA

The snapshots of the String and Synonym Matching Algorithm (SSMA) are given below in the Fig. 3. The Fig.3.1 is the first look of the system where four blocks are there. The "Enter a String" Block is to give the String as a command by a user. In Fig.2 it can be seen that 'run' command is there in the system database. So in Fig.3.2 'run' command is given by the user. After giving the 'run' command user has to click on the 'Search" button. The corresponding Action of the 'run' command is "explorer.exe". When user is clicking on the 'Search' button, the corresponding Command and the Action those are matched with the user given command is shown in the Command and Action Block. In the right hand side all the synonyms of the given command are shown along with the command.

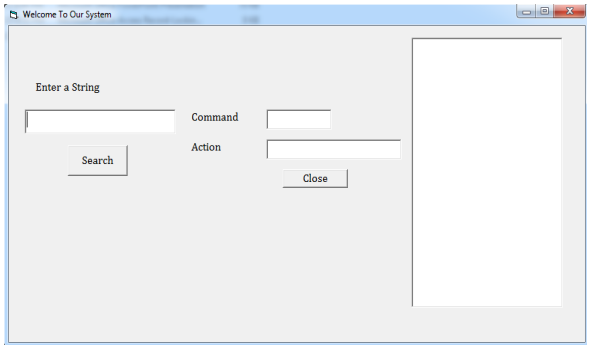


Fig.3.1: 1st Snapshots of SSMA

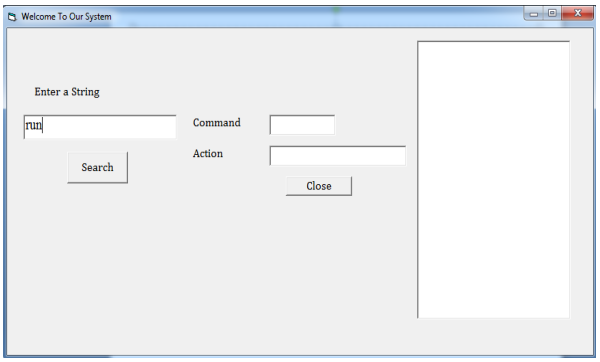


Fig.3.2: 2nd Snapshots of SSMA

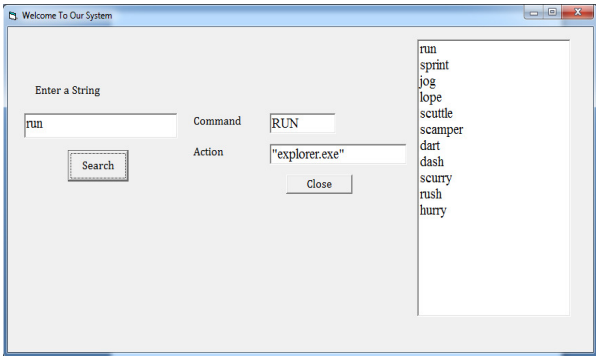


Fig.3.3: 3rd Snapshots of SSMA

If the user is giving 'sprint' as a command in the text box and click on the search button, then all the synonyms are shown in the right hand side box. In the right hand side box 'run' string is there which is getting matched with the system database. So that the in the Command and the Action Box 'RUN' and 'explorer.exe' are shown in the Fig.3.4.

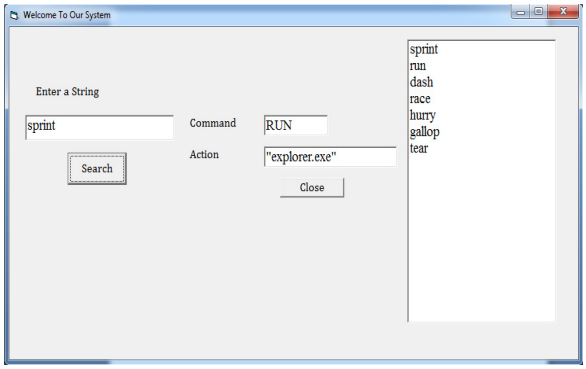


Fig.3.4: 4th Snapshots of SSMA

On the other hand if the user is giving 'tear' as a command, it can be seen from Fig.2 that 'tear' is not there in the system database. When user is clicking on the 'search' button after giving 'tear' as a command, in the right hand side box, all the synonyms of the tear are shown in Fig.3.5. But no synonym is getting matched with any command stored in the system database. That is why a message is shown to the user that *"No Action For This Command…!!!"*. Similarly the same result is found for the 'go' command also shown in the Fig.3.6.

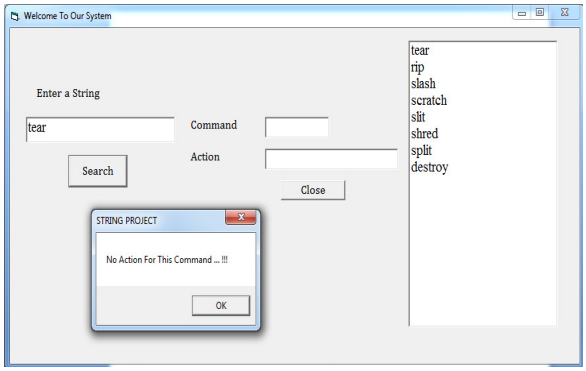


Fig.3.5: 5th Snapshots of SSMA

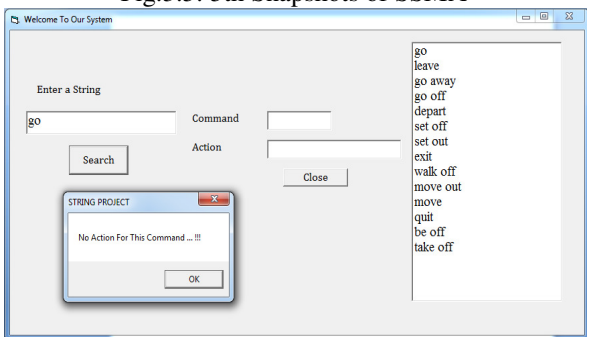


Fig.3.6: 6th Snapshots of SSMA

Fig.3: Snapshots of String and Synonym Matching Algorithm (SSMA)

## VII. ALANYSIS OF SSMA

In the SSMA the user given string along with its all synonyms are compared with all the commands stored in the system database. If the number of system commands is N and the average number of synonyms of N words is M, then the space complexity required in conventional algorithm is O(NxM). But in String and Synonym Matching Algorithm (SSMA) the space complexity is only O(N). The Table I shows the comparison between the SSMA and the conventional algorithm.

TABLE I. ANALYSIS OF SSMA AND THE CONVENTIONAL ALGORITHMS

| | No of System Commands Required | No of Avg. of Synonym of the Commands given by the User | Space Complexity |
|---|---|---|---|
| For SSMA | N | M | N |
| Conventional Algorithm | | | N x M |

In the SSMA Algorithm a database of one thousand words were used to implement. The words are categorized into two types. One is Conventional Words which are frequently used and has many synonyms and the other type is Technical Words which has technical meaning.

TABLE II. SAMPLE OF CONVENTIONAL & TECHNICAL WORDS USED IN SSMA

| Word Type | Words | No of Synonyms | Avg No of Synonym | Reduced Space Complexity |
|---|---|---|---|---|
| Conventional Words | Run | 8 | 7.125 | 87.7% |
| | Restart | 8 | | |
| | Hibernate | 8 | | |
| | Command | 8 | | |
| | Write | 8 | | |
| | User | 5 | | |
| | Click | 8 | | |
| | Folder | 4 | | |
| Total | 8 | 57 | | |
| Technical Words | Shutdown | 3 | 0.455 | 31.25% |
| | Logoff | 0 | | |
| | Notepad | 2 | | |
| | Calculator | 0 | | |
| | Google | 0 | | |
| | Photoshop | 0 | | |
| | Selfie | 0 | | |
| | Cookies | 0 | | |
| | Datacard | 0 | | |
| | Mouse | 0 | | |
| | Leftclick | 0 | | |
| Total | 11 | 5 | | |
| Total Evaluation | 19 | 62 | 3.579 | 76.5% |

The small sample database of Conventional and Technical Words used in Sting and Synonym Matching Algorithm (SSMA) is shown In the Table II. It is reflecting that, there are 8 conventional words and 11 technical words. The average of no of synonyms of the Conventional Words is 7.125 and for the Technical Words is 0.45. If in the system database, all the 19 words along with their all synonyms are stored at the initial stage, then total 81 words were to be stored, but only 19 words were stored in spite of 81 words.

Only for the given nineteen words, total 76.5% space complexity was reduced. For Conventional Words the space complexity reduced to 87.7% and for the Technical Word it reduced to 31.25%. Applying the same methodology in the whole system, 84% of space complexity was reduced where only for the Conventional Words it is about 89% and for technical words it is about 29%. Apart from that if we consider more word and store them in the system database as a command, then we can see that the average no of synonyms of a conventional word are 8-10 and for technical it is 0-2. So it can be concluded that for conventional words 85-87.5% space requirement can be reduced and for technical words 25-35% space complexity can be reduced. Table III shows the reduced time complexity in SSMA.

TABLE III. SPACE COMPLEXITY IN SSMA

| Word Type | Total No. of Words in System Database | Total No of Synonyms | Avg. No of Synonyms | Space Complexity Reduced |
|---|---|---|---|---|
| Conventional Words | 645 | 5160 | 8 | 89 |
| Technical Words | 355 | 142 | 0.4 | 29 |
| Total | 1000 | 5302 | 5.302 | 84 |

From the Table II and Table III, it can be said that the Reduced Space Complexity is inversely proportional to the number of synonyms per word. The relation is given below:

$$\text{ReducedSpaceComplexity} = \frac{1}{\text{NumberofSynonyms}}$$

## VIII. CONCLUSION

The String and Synonym Matching Algorithm (SSMA) is proposed to compare string with the preloaded strings from the system database concatenating the concept of reducing the space complexity. Implant the SSMA into any other system where string comparison is required, it will reduce the space. It will also reduce the cost factor of a system. The further study is required to analyze how the time complexity can be reduced in SSMA. The SSMA can be further used in Human Computer Speech Interaction to increase the user friendliness of the system.

## REFERENCES

[1] "ACM SIGCHI Curricula for Human-Computer Interaction", Definition of HCI, [Available: http://old.sigchi.org/cdg/cdg2.html] [Access Date: May 21, 2014].

[2] "Human-computer Interaction", [Available: https://en.wikipedia.org/wiki/Human–computer_interaction] [Access Date: May 21, 2014].

[3] "Human Computer Interaction (HCI)", [Available: http://www.webopedia.com/TERM/H/HCI.html] [Access Date: May 21, 2014].

[4] F. Karray, M. Alemzadeh, J. A. Saleh and M. N. Arab. "Human-Computer Interaction: Overview on State of the Art". In the proceedings of IJSSIS, Vol. 1, No. 1, pp. 137-159, March 2008.

[5] K. P. Tripathi, "A Study of Interactivity in Human Computer Interaction", International Journal of Computer Applications, ISSN: 0975 – 8887, Vol. 16, Iss. 6, Feb 2011.

[6] P. Sharma, N. Malik, N. Akhtar, Rahul, "Human Computer Interaction", International Journal of Advanced Research in IT and Engineering (IJARITE), ISSN: 2278-6244, Nov 2012.

[7] Brad A. Myers. "A Brief History of Human Computer Interaction Technology." ACM interactions. Vol. 5, no. 2, pp. 44-54, March, 1998.

[8] U. Aickelin and D. Dasgupta, "Artificial Immune System" [Available: http://arxiv.org/ftp/arxiv/papers/0803/0803.3912.pdf] [Access Date: July 5, 2014].

[9] M. Fetaji, S. Loskoska, B. Fetaji, M. Ebibi. "Investing Human Computer Interaction Issues in Designing Efficient Virtual Learning Envioronment". BCI, Sofia, Bulgaria, 2007.

[10] P. Sharma, N. Malik, N. Akhtar, Rahul, "Human Computer Interaction", International Journal of Advanced Research in IT and Engineering (IJARITE), ISSN: 2278-6244, Nov 2012.

[11] A. Britto, R. Sabourin, F. Bartolozzi, C. Y. Suen, "A Two-Stage HMM-Based System for Recognizing Handwritten Numeral Strings", International Conference on Document Analysis and Recognition, pp.396-400, 2001.

[12] L. Yun, C. S. Liu, X. Q. Ding, F. Qiang, "A Recognition Based System for Segmantation of Touching Handwritten Numeral Strings",

[13] S. Gamm, R. H. Umbach, D. Langmann, "Finding with the Design of a Command Based Speech Interface for a Voice Mail System", IEEE Proceedings of Interactive Voice Technologies for Telecommunications Applications, 1996, pp.93-96, India.

[14] A. Rashedi, S. S. Moghaddam, "Appropriate Farsi Speech Recognizes for Commanding Robots: Performance Evolution of Correlation-Based and Model-Based Classifiers for a Farsi Isolated Word Recognition Robotic System", ICSP, pp. 573-576, 2010.

[15] K. Hinckley, J. Hollan, "PapierCraft: A Gesture-Based Command System for Interactive Paper". [Available: http://www.cs.cornell.edu/~francois/Papers/PapierCraft_TOCHI.pdf] [Access Date: 8th Aug, 2015]