An Optimal Human Adaptive Algorithm to Find Action - Reaction Word-Pairs

Arpit Agarwal, Rahul Banerjee, Varun Pandey, and Riya Charaya

Birla Institute of Technology & Science, Pilani
333031 Pilani, India
{arpit,rahul,varun,riya}@touch-lives.org

Abstract. This paper presents an efficient approach for understanding the formation of associations between random sentences spoken by humans over a period of time. The associations formed are mathematical relations (A X B) where the former is called as the "action" and the latter as the "reaction". The voice-to-text converted file is the input to the algorithm. After processing, the algorithm devises a map (Actions X Reactions). The algorithm stops only after the relation becomes surjective. The most important improvement over the previous techniques is the automatic adaptation of the machine to the ever-changing grammar of the user in real-time.

Keywords: adaptive, surjective, relations, grammar, human-behavior.

1 Introduction

Of late, there has been a marked shift in the user acceptance level from desktop to mobile applications and this has opened up several possibilities for technology-assisted approaches aiming to enable those who are often not served by it. Touching the lives of millions using simple, inexpensive technology-driven gadgets has been the driving force of the "*Project Connect*" under the *Touch-Lives Initiative (http://www.touch-lives.org)*. We have attempted to conceptualize and design simple assistive solutions that with the help of voice driven input, makes the mobile device automatically respond to the aid of the elderly user. Part of this work involves making a customized user-interface for easy use by the elderly people. We hope, it would help the user in the following ways: -

1. Respond to user queries without a trigger.
2. Interact with users in real time and intuitively.
3. Call for help in case of emergencies, without requiring explicit call initiation.

2 Related Work

Algorithms specifically designed to interpret a given speaker's real intent have attracted a lot interest in the recent years. Yoshiharu et. al. [1] in their work on the

C. Stephanidis (Ed.): Posters, Part I, HCII 2011, CCIS 173, pp. 255–259, 2011.

MindReader have designed a system where the computer tries to understand the user by repeated questioning. Proposing the view that users cannot always easily express their queries, this algorithm gives multiple queries a score (0/1 or multi-level). Subsequently, the algorithm guesses the appropriate distance function and issues the appropriate query.

Research by Doulamis. al.[2] in *Adaptive Algorithms for Interactive Multimedia*, throws light upon the various feedback algorithms and their performance, which provides important cues on the upper limit of iteration count.

The *Stable Partition Algorithm (SPA)* is an adaptive algorithm, which finds a valid relation between two elements (say ‘a’ and ‘b’). A valid relation is one, where all ‘b’ are preceded by ‘a’ a *sufficient* [1] number of times. The run-time complexity of this algorithm is *O(nLog(n))* if any secondary memory is not available and *O(n)* complexity, if enough primary memory is available.

These pieces of works have helped our algorithm in accurately and reliably classifying the relations between *“action-sentences”* and *“reaction-sentences”* out of large sets of random data.

3 Proposed Methodology

The application developed under the *Project Connect* makes use of innovative algorithms that allow building embedded software capable of efficient and fast response to queries in real time without human-intervention. The mobile device is activated upon the reception of voice queries. Voice recognition algorithms are used to convert the voice-input to text. The text-files thus formed are analyzed forming logical-relations between sentences (stand-alone) and they are termed as *action-reaction pairs*. Once this stage completes, every time an *action-sentence* is encountered, the *reaction-sentence* triggers a function to help the user.

The human-adaptive algorithm, finds a relation between sentences by repeated reduction of the possible target-values by iterative intersection of *Region of Interests (ROI)*. Firstly, a possible *“action-word”* (usually the verb of the sentence) is determined. Since, sentences are logged at random intervals, a lot of *noise* (irrelevant sentences) would be logged as well. The ideal relation between every “*action-sentence*” and “*reaction-sentence*” would eventually take indefinite time. Thus, the limit on the iteration count is essential to balance the optimality and accuracy. It has been found that the method used in our algorithm performs with an accuracy of 91.84% (determined from the following histogram.) and a run-time efficiency of *O(n).* (Where n is the size of input file that has to be analyzed).

The Y axis in Figure 1 (below) is read as: (say for dataset-1) Match:Action::30:70. So, in case there are 70 actions then there are 42.85% (i.e. 30, worst case) matches. The average of the ratio (Match:Action) over these 1000 datasets was found out to be 91.84:100 i.e. the algorithm is 91.84% accurate in order to find the action-reaction word pairs.

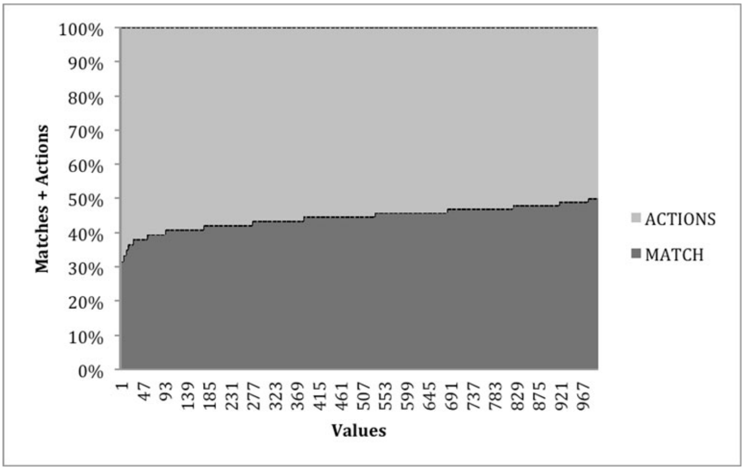

Fig. 1. Histogram showing how different datasets responded to the algorithm. The Y-axis is %age contributions of Actions & Matches.The X-axis refers to each dataset.

The algorithm, which has been explained above, can be summarized as:

```
begin:
          ReadLogFile();
          ChooseSentence();
          FindActionKeyword(); //Within the sentence.
          FormRegionsOfInterest();
//Using the action keyword as delimiters.
          while(!RelationAccuratelyFound) //aRb::a→b
begin:
TakeIntersections();
            //Select common-sentences form ROI's.
          end
end
```

Assumptions: 1) The log-file has infinite length, 2) Number of required iterations [1][6] for each action-sentence <= Number of available ROI's formed (in the log file) for the same action-sentence.

4 Experiments & Results

The most important part of developing this algorithm was identifying the number of iterations to accurately decide the relations (aRb). Earlier works on adaptive algorithms ([3][4][5]) have shown remarkable results but our algorithm has demonstrated (Graphs and explanation below) a significant improvement over them in terms of both *performance* and *reliability*.

Here n is the size of the input file – i.e. the number of actions for which the action-reaction pairs have to be found.

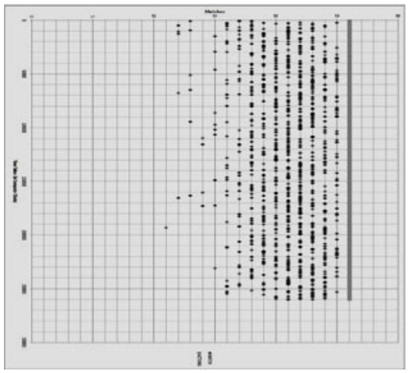

Fig. 2. Number of action-reaction pairs found

In Figure 2, the concentration of the points can be seen near the bold line at the right end that shows the *accuracy* and the *reliability* offered by the presented algorithm. A similar approach was tried with a neural network based algorithm to solve the problem but that turned out to be quite memory intensive. Hence, this algorithm suits fine, as it offers a *space complexity* of *O(n)*.

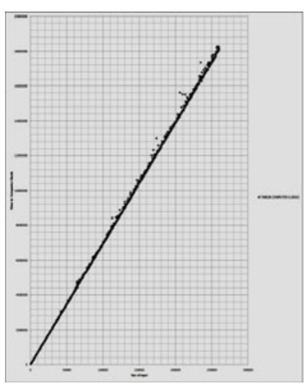

Fig. 3. Size of input Vs. Time

It can be clearly seen from the graphs that the time complexity of the algorithm implemented is *O(n)* Here n is the size of the input. Figure 3 is plotted for *time-taken* versus the *size-of-input*. Here, graph shows a straight-line depicting a linear relation between time and size-of-input.

Also, the sample size for the graph is of the order of 10^6 values, making the results reliable for future studies.

5 Conclusions

The presented algorithm has been designed for mobile communication devices and therefore the principal effort was to optimally use the storage and processing power. We have shown that our algorithm is memory efficient. The time and space complexities have been found to be acceptable for the target platforms and have been successfully tested on the same.

Also, this work represents a significant improvement over other known algorithms [1][2][3][4][5].

References

1. Ishikawa, Y., Subramanya, R., Faloutsos, C.: MindReader: Query Databases through Multiple Examples. In: Proc. 24th VLDB Conf. Morgan Kaufmann, San Francisco (1998)
2. Doulamis, N.D., Doulamis, A.D., Varvarigou, T.A.: Varvarigou, Adaptive Algorithms for Interactive Multimedia. IEEE Multimedia 10(4), 38–47 (2003), doi:10.1109/MMUL.2003.1237549
3. Peter Lepage, G.: A new algorithm for adaptive multidimensional integration. Journal of Computational Physics 27(2), 192–203 (1978) ISSN 0021-9991, doi:10.1016/0021-9991(78)90004-9
4. Duan, J., Bressan, M., Dance, C., Qiu, G.: Tone-mapping high dynamic range images by novel histogram adjustment. Pattern Recognition 43(5), 1847–1862 (2010) ISSN 0031-3203, doi:10.1016/j.patcog.2009.12.006
5. Li, X., Lam, K.M., Shen, L.: An adaptive algorithm for the display of high-dynamic range images. Journal of Visual Communication and Image Representation, Special issue on High Dynamic Range Imaging 18(5), 397–405 (2007) ISSN 1047-3203, doi:10.1016/j.jvcir.2007.06.005
6. Abramowitz, M., Stegun, I.A. (eds.): Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables, ch. 26, p. 948. Dover, New York (1965) ISBN 978-0486612720, MR0167642