Collaborative Form Filling With Predicting Techniques

M. Raghava Rao
PG Student, Computer Science and Engineering Dept.
Akula Sreeramulu College of Engineering,
Tanuku,A.P(India)

M. Satish Kumar
Prof, Computer Science and Engineering Dept.
Akula Sreeramulu College of Engineering,
Tanuku,A.P(India)

Abstract

Modern databases focus on data quality using automatic detection methods. Through this technique we are detect the errors and overcome the errors. Through this system we are increases the data quality and show the better results of information. Previously we are works on probabilistic models creation process. It can identify the results as a refinement questions process. It can show the problems identification in the form complex. Some kinds of situations we are miss some kind of features of information. We are introduces some kind of new detection techniques for finding the dynamic form creation methods. Dynamic form creation shows the results as reformulation techniques. Reformulation nothing but re-asking procedure we are creating inside the form generation. It can provide the results as a good feedback applications identification process.

Keywords: *Form filling, CCFU, MFU, MRU, Bayesian network, Hidden Web.*

1.INTRODUCTION

In many number of applications we are use the form filling environment. Every day for verifying all the values are filled correctly or not we are checks inside the environment process. Same form we are filling in different number of times under the present environment process. Furthermore, people frequently have to fill out the *same* form, over and over. Example, a salesperson may have to fill out a purchase order every time his client makes an order. It can represent an opportunity to use predictive algorithms to improve productivity. Many are a variety of approaches by which predictive algorithms can be used to aid a user when he is filling out a form. The most common approach is to generate a list of likely values for the field the user is attempting to fill out, called a *suggestion list*. The suggestion list for a given field is presented to the user when he navigates to that field. A variety of user interfaces can be used to display a suggestion list to the user; currently, the most popular is to use a drop-down list, for example in Figure 1. The user can then navigate through the items in a suggestion list by scrolling or typing, although other input modalities, such as speaking or touching, could be considered.

Much number of approaches are present under implementation like dynamic form filling. Current approaches do not consider the values of other fields on the form when predicting the value of the field the user is currently filling out. This means that these approaches cannot model the naturally occurring dependencies between fields, such as that between a field intended to capture the name of the form filler and a field intended to capture the filler's address. In addition, current approaches, while typically considering previously entered values from the current user, do not consider values for fields previously entered by other form fillers when making predictions.

Prediction techniques are contains number results. These results are displayed as modified results. We introduce a novel model of predictive form filling named the *Collaborative quorum based environment Contextually Frequently Used* model to utilize these two information sources. We demonstrate that the CCFU improves over the standard models of form filling with a real-world dataset. We also show how incorporating the values used by other form fillers can improve the performance of existing models.

2. Dynamic models of form filling environment process

Different models are creates with automatic specification and prediction techniques. This kind of classification techniques are provides the results as a good decision making environment process. Form filling process shows the faults identification. All the errors are mitigate inside the implementation process.

2.1 Predictive Models of Form Filling

As discussed earlier, our models will build a suggestion list that contains ranked predictions for the user's current target field.More formally, a predictive model of form filling is required to be able to answer queries about the probability of values for a field given the values of other fields, history, and some initial text entered by the user in the target field.

2.2 Form-filling System

The focus of this paper is on the intermediate processing stage of form filling, so we assume the existence of an electronically reproduced, on-screen form an *electronic form*.

The electronic form is created to visually resemble its paper counterpart for two reasons: so users can easily accept the new technology, and so the daily work flow of the user is not adversely affected by the system. Figure 1 shows an electronic version of a Leave Report form used at Washington State University. Although the electronic form appears to be optically scanned image of a paper form, it is actually quite different. Each box, or *field*, on the form is editable; which means a user can type into a text box (e.g., name, address, or social security number), or select a check-box (for selection items) by using the computer's mouse and keyboard. The example form shown in Figure 1 consists of over 300 fields for information input, using both editable text boxes and check-boxes. The user has random access to any of the displayed fields. The control buttons at the bottom of the form labeled Next, Print, Quit, and Reset are part of the user interface to the form-filling and learning system and are not part of a printed Leave Report form. Figure 2 shows a block diagram of the form-filling system used in conjunction with the control buttons. The thick-lined box in the center of the diagram is the core form filling process that combines the electronic form, the user input, and prediction feedback. When a user completes a form field (by typing into an editable box or by clicking a check box), that information is passed as form field data to three modules Data is presented to the printing module so the user may generate a paper copy of the electronic form. The learning module uses the form data to construct predictive functions; these are used in turn by the predictor module to provide default values for other fields on the form. After each form field is edited by the user, a default value is predicted for each field; after each form is completed, the predictive functions are updated by the learning module. Although some of the form-filling functions shown in Figure 2 are commonly available in commercial form design packages, our system has an additional component — the learning module. But before we can describe how the learning module plays a role in our system, it is important to first understand how the user interacts with the electronic form. When the electronic form system is started for the first time, the form fields are blank. To access a field, the user moves the mouse input device to position the screen cursor over a text box or check box. Clicking on a check box will toggle an unchecked box to checked and vice versa. A click on a text box will illuminate a text-edit cursor which indicates that the user may type information into the field. If a field must be changed, the user can employ typical editing commands to delete or change a field's contents. When user has completed the form, they may click on the Print control button to print a paper copy of the form, click Quit to end the session, or click Next begin working on a new instantiation of the same form. Once a new instantiation of the form is shown on the screen, all of the fields are blank again. When a form is being shown on the screen, it is the *current form*. Once the Next button has been clicked, a new instantiation is displayed, and the form that was most recently displayed becomes the *previous form*.

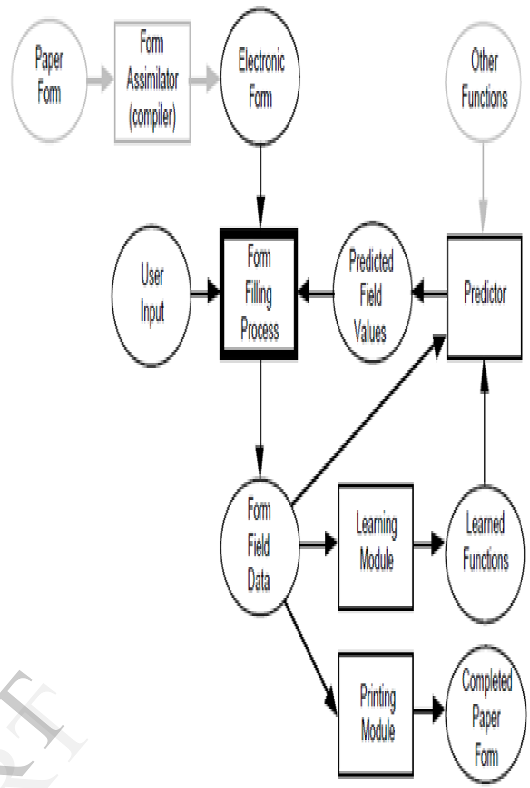

Fig 1: Proposed Architecture

3. . STUDY METHODOLOGY

The user study took place in the Surat and Bharuch districts of the Indian state of Gujarat during July and August of 2008.

3.1. Participants

As detailed in Table III, the study participants consisted of six community health workers and seven hospital paramedical staff. The community health workers were associated with the Dahej public health center; five of the paramedical staff were at the Reliance Tuberculosis hospital; and the remaining two paramedical staff were at the dispensary of the Sardar Vallabhbhai National Institute of Technology. The study participants were recruited through contacts of the first author.Initially, we had hoped to perform the study entirely with community health workers, as they are often the primary agents of remote data collection (including in our upcoming tuberculosis treatment program). However, this turned out to be infeasible because some community health workers were unable to travel to the Dahej public health center for training nand testing, and it was not feasible for us to travel to each worker's home. This prompted us to recruit participants from two other centers. There were also some logistical challenges in performing the studies due to adverse weather conditions and the bomb blasts occurring in July 2008 in the Surat area. The education level of the health workers ranged from 10 to 12 years,while the education of the hospital staff ranged from 10 years to a B.A. degree. The average age of the study participants was 26.4 years (range 19-35). Seven participants owned a cell phone, four participants had used but did not own a cell phone, and two participants had never used a cell phone previously. Eleven of the participants were native Gujarati speakers and all spoke Hindi.

3.2. Training

Participants were trained by at least two trainers in small groups of at least two. Initially, examples were presented on a whiteboard and participants were instructed to practice entering in the data on either electronic forms or as an SMS using the cue card. After this stage, a paper with a set of example patients was handed out, and participants were instructed to practice entering in this data. In the final stage, participants were instructed to practice role playing patient– worker interactions with each other. Participants received variable amounts of training, ranging from 45 minutes to 8 hours, depending on their experience and availability. The longer training sessions were not necessarily more effective, as they were performed in larger groups. While it would have been desirable to achieve more uniform training, this was difficult given the logistics of transportation and worker schedules. Prior to the completion of training, all participants had completed at least two perfect interactions on both electronic forms and SMS, and at least one perfect interaction on the live operator mode. Throughout the user study, we employed Motorola L6i cell phones for training and testing. This is the cheapest Java-enabled phone from Motorola (the source of our current development tools) that is available in India; see Appendix A-1 for a cost analysis. All interfaces and related tools (cue cards, etc.) were presented in Hindi, and the mobile phones used had dual Hindi menus.

3.3. Testing

Participants were tested in pairs, alternating who was being tested on data entry, and who was playing the fake patient for that data point. The order of the interfaces was randomized: for a given participant pairing,the order of voice, SMS, and electronic forms was alternated. For the voice interface, the first author acted as the operator and was located outside of the room testing was being conducted in; however, there was always an additional person associated with the experiment inside the room at all times with the participants. During testing, each participant performed two complete patient–worker interactions (in the role of the worker) for each of the forms and SMS interfaces. For the voice interface, the six community health workers completed only one interaction, while others completed two interactions (we did not anticipate that voice would become a focal point of this study until halfway through our experiments). The lag time between training and testing was exactly one day for seven of the participants, and ranged between half a day and two days for the remaining participants. All participants received a brief refresher and supervised entry session immediately prior to testing.

3.4 Error Model

To formally model the notion of error, we extend our Bayesian network from Section 4 to a more sophisticated representation that ties together intended and actual question responses. We call the Bayesian network augmented with these additional random variables the error model. Specifically, we posit a network where each question is augmented with additional nodes to capture a probabilistic view of entry error. For question i, we have the following set of random and observed variables:. Fi: the correct value for the question, which is unknown to the system, and thus a hidden variable. . Di: the question response provided by the data-entry worker, an observed variable. . _i: the observed variable representing the parameters of the probability distribution of mistakes across possible answers, which is fixed per question.

4.CONCLUSION

We are shows the results are identified good accurate and efficient results identification. In this study, we provide a quantitative evaluation of data entry accuracy on mobile phones using electronic forms, SMS, and voice interfaces in a resource-poor setting. Our results indicate that, within the context of our study, the error rates for electronic forms (4.2% of entries wrong) and SMS (4.5% of entries wrong) may be too high to deploy these solutions in a critical application. In contrast, the accuracy of the voice interface was an order of magnitude better (0.45% of entries wrong), with only a single error observed across all trials. This result has influenced us to overhaul our plans for an upcoming tuberculosis program in Bihar, India, to switch to a voice-only interface. Employing a voice interface requires the employment of an operator, and may not be cost-effective in all countries. However, in India, the cost of this operator is more than compensated by the lower cost of voice-only handsets, voice-only cellular plans.

5.REFERENCES

[1]. T. Dasu and T. Johnson, Exploratory Data Mining and Data Cleaning. Wiley,2011
[2] C. Batini and M. Scannapieco, Data Quality: Concepts, Methodologies and Techniques. Springer, 2009
[3] R.M. Groves, F.J. Fowler, M.P. Couper, J.M. Lepkowski, E. Singer, and R. Tourangeau, Survey Methodology. Wiley-Interscience, 2010.
[4] J. Lee, “Address to WHO Staff,” http://www.who.int/dg/lee/speeches/2003/21_07/en, 2008.
[5] J.V.D. Broeck, M. Mackay, N. Mpontshane, A.K.K. Luabeya, M. Chhagan, and M.L. Bennish, “Maintaining Data Integrity in a Rural Clinical Trial,” ClinicalTrials, vol. 4, pp. 572-582, 2007.
[6] R. McCarthy and T. Piazza, “Personal Interview,” Univ. of California at Berkeley Survey Research Center, 2009.