Semantic Pen - A Personal Information Management System for Pen Based Devices (Extended Abstract)

Akila Varadarajan, Nilesh Patel and William Grosky

The University of Michigan - Dearborn, Dept. of Computer Science,
4901, Evergreen Road, Dearborn, MI 48080, USA
{akilav, patelnv, wgrosky }@umich.edu

Abstract. The Onset of Semantic Web technology have promised a new vision of Personal Information Management (PIM). With the advent of Pen-based computing, PIM faces new challenges: usability and flexibility are important constraints in the pen based environment. We present our system of Semantic Pen - an augmented pen based PIM system that merges the efficiency of semantic web with the usability of pen based devices. The architecture consists of an intuitive user interface which can capture digital ink, a Hidden Markov model (HMM) to extract personal information and a data model of Resource Description Framework(RDF) for flexible organization and semantic querying of data.

1 Introduction

Personal Information Managers (PIM) have become increasingly common these days. The usage model of PIM systems have gone beyond scheduling reminders and simple record maintenance. Semantic Web, through the introduction of ontological reasoning by means of Resource Description Framework(RDF)[1] have proven to be an efficient solution for PIM . The Haystack Project [2] is well known for applying semantic web technologies to create a fully flexible and customizable PIM portal for organizing the germane information. The Gnowsis Semantic desktop [3] targets data integration including data from 3rd party applications. Semex[4] focuses on personalized desktop search. Chandler[5] is an Interpersonal Information Manager that supports data sharing besides managing email, calendar and other general information. Retsina Calendar Agent[6], is a distributed meeting scheduling agent which works in conjunction with Microsoft Outlook 2000 and Semantic Web.

While most of the research in PIM using Semantic Web is centered around desktop and notebooks, there is a need to extend such concepts in context of pen-based computing. The pen-based systems have empowered users by providing the most natural form of input modality known as *Digital Ink*. Since its introduction, researchers have shown increased interest to ease the user interface centric tasks. Wilcox et al. designed a system Dynomite [7] for organizing telephone numbers and other tasks by applying properties for ink words. Scribbler [8] is another tool that enables searching ink words, symbols or simple sketches by matching raw strokes instead of recognized text. Marquee [9]is a logging tool where users can correlate their personal notes and keywords with a videotape during recording. Microsoft's products One Note 2003 and Journal helps to capture, customize and organize ink documents suitably.

We present *Semantic Pen* that aims to combine the efficacy of semantic web with the usability of pen based devices to provide a next generation highly intuitive and intelligent PIM system.

2 Semantic Pen

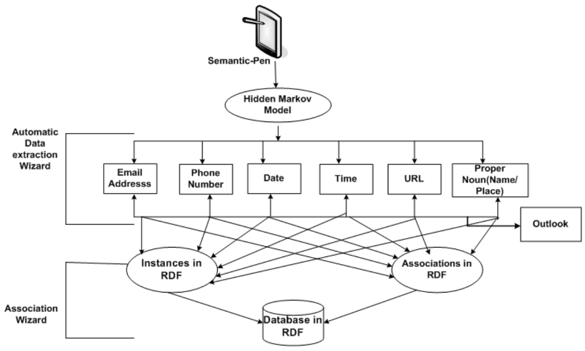

Fig. 1. Architecture of Semantic-Pen

Semantic Pen has a simple and attractive user interface comparable to leading note taking tools. In addition, our system is composed of two core modules; (1) an Automatic Data Extraction (ADE) wizard and (2) an Association wizard. The system Architecture of Semantic Pen is shown in figure.1. ADE is the heart of the system which extracts the data via Hidden Markov Models(HMM)[10]. Additional details such as name of a person for an extracted email address can be semi-automatically included through the ADE wizard. This wizard displays a *smart name/place list* generated by our intelligent noun filter algorithm. The user can either choose a name from the list or enter his own. This extracted personal information is then automatically stored in the commercial information management tool such as Microsoft Outlook. Once the personal information is extracted, an *Association Wizard* helps associating the data with the existing data repository items. Our approach uses the popular RDF framework Jena [11] to store and retrieve the data.

2.1 Automatic Information Extraction using Hidden Markov Model(HMM)

HMM is a finite state automation that implements stochastic state transitions and symbol emissions. We use the model of Freitag and MacCallum [12, 10] to extract personal data from the ink notes. Once the states for the HMM (Prefix, Target,Suffix and Background states) is decided, the document is parsed and taxonomized to obtain the emission vocabulary of the HMM. We generate a set of intuitive term by feature pairs t, f where t is the intuitive term and f is an identified feature that creates the appropriate intuition on that term [13].

The possible formats and constraints for the $Intuitive\ term\ features$ such as *Email ID*, *Phone No*, *Date*, *Proper Noun* are identified and defined in a database. Then we calculate WFM to classify the intuitive terms. For a term t, $WMF(t)$ is computed as follows:

$$WFM(t) = \frac{Nc(t)}{Nc(f)}$$

Where, $Nc(f)$ represents the total number of constraints for the word feature f. For example, the '@ 'symbol and a domain name are some constraints for an email address. $Nc(t)$ is defined as:

$$\sum_{x=0}^{Nc(f)} M(t, c_x)$$

where $M(t, c_x)$, the matching function, equals 1 if the term t contains a matching constraint c_x

$WFM(t)$ is calculated by varying f in $Nc(f)$. If $WFM(t)$ equals 1 for some value of f in $Nc(f)$, it means the term t is of the suspected word feature type f. If $WFM(t)$ is less than 1 for all values of f in $Nc(f)$, it means the term is not of any type of suspected word feature.

Table.1 describes how we define the emission vocabulary for the HMM by means of WFM and Bikel's classification of word features [13].

Table 1. Emission vocabulary for HMM

Intuitive Word Feature	Example formats
Email ID	bob@umich.edu, bob@yahoo.com, bob@xyz.org
Phone No.	(586)-779-6320, 586-779-6320, 779-6320
Date	09/01/06, 09-01-06, 09/01/2006,Sep-1-06
Time	12.30 pm, 12:30 a.m, 12.30 AM
Proper Noun(Name or Place)	Bob, Michigan
URL	www.umich.edu

Once the emission vocabulary by means of the intuitive word features is obtained, the Viterbi algorithm [12] is used to accurately identify the most likely state sequences of a particular document. Finally, the HMM outputs the strings which are likely to be the personal data that need to be stored.

2.2 Personal Information Association using RDF

The next step is to create suitable associations of the new data with the existing elements in the database. We are currently in the development stage of this algorithm. In this, we define two components namely *instances* and *associations*. The instances are the actual objects that need to be associated such as *email address of Bob* or *web page of an institution "XYZ"*. The associations are the relationship that might exist between two instances. Consider, "Bob *works at* XYZ". In this case *works at* is an association that exists between instances Bob and XYZ that binds them together. Similarly there might be another association existing such as "Steve *works at* XYZ". Now an automatic link gets associated between Bob and Steve. However, the user is prompted to obtain a suitable association between these two instances.

Initially all possible instances such as contact information, task list and web page links will be extracted by the system. Associations among these instances will be obtained semi-automatically by running the *association wizard*. The instances and associations are then stored in a separate database and represented by means of Ontolgies using Resource Description Framework (RDF). The RDF framework Jena [11] is chosen to store and retrieve the RDF data. Also, when a new item is added to the database externally, our system will alert the user to run the *association wizard* to form suitable associations.

Our interface will identify the associated instances by querying the RDF database and generate associations such as;(i) a calendar entry is related with a file which is modified at that date and time,(ii) a book marked web page consists of information about a workshop in the task item, (iii) a contact is the author of a particular document. The user will be allowed to choose an association from an existing list or to define his own.

3 Experimental Results

A NEC Versa Lite Pad Tablet was used to test our system. The note taking interface is developed using Agilix infinotes [14] .NET component. To test our Automatic Data Extraction (ADE) wizard, we collected meeting notes from 25 people. Each collected note was about 250-500 words in length, containing a mixture of email address, phone number, date-time information, proper nouns, and hyper-links. The data extraction results were analyzed off-line via the *Automatic Data Extraction(ADE)* wizard. Our application uses the recognized ASCII text from the meeting notes for all manipulations. The built-in Microsoft Hand Writing Recognizer that comes with the tablet is used to translate ink data to the ASCII text. Since the *Association Wizard* is still under development, the experimental results pertaining to only ADE is presented in this paper.

The ink to text recognition accuracy plays a major role in performance of ADE wizard. In past, due to its least individuality the numbers have been reported with higher recognition accuracy [15]. Our analysis also supports the previous research in this regard. The recognition rate of numbers in our experiment, was found to be as good as 88.9% compared to the recognition rate of the letters which was found to be just 58.8%. Similarly, we also found that the non-cursive handwriting had the highest recognition rate of 82.2% compared to the cursive handwriting recognition rate of about 73.2%. The printed handwriting had the worst recognition rate of 21.1%.

In addition to the recognizer's inaccuracy, we also found that the emission vocabulary symbols failed to get identified due to unnecessary white spaces inserted by the recognizer. Our system intelligently handles these white spaces to improve an overall precision of the data extraction system.

The results of our phase I activity in extracting key personal information using HMM is measured using standard precision and recall measures, defined as:

$Precision = \frac{R}{R+R_i} * 100$; $Recall = \frac{R}{R+R_m} * 100$
where, R=Relevant records retrieved, R_i= Irrelevant records retrieved and R_m=Missed relevant records.

Table 2. Precision Vs Recall for automatic data extraction

Data Extracted	Precision	Recall
Email address	90.15	89.34
Phone Number	96.57	96.87
Schedule Information (Date and Time)	88.26	89.75
URL	91.12	92.34
Proper Noun(Name or place)	93.23	89.23

References

1. Guha, R., Brickley., D.: Rdf vocabulary description language 1.0: Rdf schema. W3C recommendation 10 february 2004. (2004)
2. Quan, D., Huynh, D., Karger, D.R.: Haystack: A platform for authoring end user semantic web applications. (ISWC 2003) 738753
3. Sauermann, L.: The gnowsis semantic desktop for information integration. (Proceedings of WM 2005)
4. Cai, Y., Dong, X.L., Halevy, A., Liu, J.M., Madhavan, J.: Personal information management with semex, Baltimore, Maryland USA, ACM (2005)
5. OSAF: Chandler. "http://www.osafoundation.org/Chandler_Compellin_Vision.htm" (2004)

6. Payne, T.R., Singh, R., Sycara, K.: Calendar agents on the semantic web. IEEE INTELLIGENT SYSTEMS. (2002) 84–86
7. D.Wilcox, L., N.Schilit, B., Sawhney, N.: Dynomite: A dynamically organized ink and audio notebook. SIGCHI 1997, (ACM) 186–193
8. poon, A., Weber, K., Cass, T.: Scribbler: A tool for searching digital ink. CHI '95, ACM Press (1995) 252–253
9. Weber, K., Poon, A.: Marquee: a tool for real-time video logging. SIGCHI' 94, ACM (1994) 58–64
10. Taghva, K., Coombs, J., Pereda, R., Nartker, T.: Address extraction using hidden markov models. IS&TSPIE 2005 (January 2005)
11. Jena: A semantic web framework for java. (http://jena.sourceforge.net)
12. Freitag, D., McCallum, A.: Information extraction with hmm structures learning by stochastic optimization. 17th National Conference AI (2000) 584–589
13. Bikel, D., Miller, S., , Weischedel, R.: Nymble: a high-performance learning namefnder. ANLP-97 (1997) 194–201
14. Agilix: Infinotes. (http://www.agilix.com/www/notecontrol.aspx?pid=14.)
15. Zhan, B., Sargur.N.Srihari, Lee, S.: Individuality of handwritten characters. ICDAR 2003, IEEE (2003)