Editorial

This issue is the first one that reflects the renewed editorial strategy and image of the research journal "Polibits". The purpose of our journal is to publish research papers in the field of computer science and engineering with emphasis on the applied research.

The journal subject is really very broad being the computer science a vast camp of research; still we are sure that the journal will be of interest for the scientific community that works in this field and for prospective users of computer technologies. When possible, we will try to group the papers related to the same area of computer science in special issues or in special sections, for example, current issue has a special section on natural language processing.

The papers for the journal are selected on the basis of the strict double-blind reviewing process taking into account their originality and scientific contribution. The journal has an international editorial board formed by 21 distinguished scientists from 13 countries.

Current issue contains special section devoted to natural language processing that consists in 5 papers and 6 regular papers. The papers in the special section are the following.

The paper "*Iterative Feedback based Manifold-Ranking for Update Summary*" is devoted to the recently appeared concept of update summary, i.e., the summary that presents only the new information as compared to a set of previously given documents.

Issues that are important for information retrieval technologies related with morphological enrichment of queries are presented in the paper "*Improvement of queries using a rule based procedure for inflection of compounds and phrases*"

The paper "*Web-based Bengali News Corpus for Lexicon Development and POS Tagging*" presents the approach and experiments for rapid lexicon development and POS tagging for languages with less computational resources. The problem is important in the actuality since there are thousands of languages in the world and only dozens of them have enough resources.

In the paper "*Methods for handling spontaneous e-commerce Arabic SMS: CATS, an operational proof of concept*" an approach to natural language driven e-commerce is discussed.

The modern view on the example based machine translation is exemplified in the paper "*Study of Example Based English to Sanskrit Machine Translation*".

The following papers that appear in this issue are regular papers.

The paper "*Aberración óptica* (Optic aberration)" presents analytical equations in approximate form that describe the front of a spherical wave with aberration.

The idea of applying association mining techniques for improvement of dealing with information overload in a web oriented retailing is presented in "*Applying dynamic causal mining in retailing*".

The expert system that is expected to help to the diabetes patients is described in "*Base de Conocimientos del Monitoreo de Parámetros Sanguíneos* (Knowledge base for monitoring of the blood parameters)". It uses large database and fuzzy inference engine.

Interaction between semantically annotated Web services for health care is the theme of the paper "*Supporting the Continuity of Home Care and the bidirectional Exchange of Data among various Points of Care by Semantically annotated Web Services*".

The system that allows for development of projects related to immersion in virtual reality based on the endless walking and multipersonal cabin is described in the paper "*Desarrollo de un sistema inmersivo de realidad virtual basado en cabina multipersonal y camino sin fin* (Development of the system for immersing in virtual reality based on the endless walking and multipersonal cabin)".

Design and implementation of digital filters, as well as corresponding experiments are presented in the paper "*Implentación de filtros digitales tipo FIR in FPGA* (Implementation of digital filters of FIR type in FPGA)"

I hope that the readers will find this issue interesting and useful for many of their needs.

Grigori Sidorov
Editor-in-Chief

Iterative Feedback Based Manifold-Ranking for Update Summary

He Ruifang, Qin Bing, Liu Ting, Liu Yang, and Li Sheng

***Abstract*—The update summary as defined for the DUC2007 new task aims to capture evolving information of a single topic over time. It delivers *focused* information to a user who has already read a set of older documents covering the same topic. This paper presents a novel manifold-ranking frame based on iterative feedback mechanism to this summary task. The topic set is extended by using the summarization of previous timeslices and the first sentences of documents in current timeslice. Iterative feedback mechanism is applied to model the dynamically evolving characteristic and represent the relay propagation of information in temporally evolving data. Modified manifold-ranking process also can naturally make use of both the relationships among all the sentences in the documents and relationships between the topic and the sentences. The ranking score for each sentence obtained in the manifold-ranking process denotes the importance of sentence biased towards topic, and then the greedy algorithm is employed to rerank the sentences for removing the redundant information. The summary is produced by choosing the sentences with high ranking score. Experiments on dataset of DUC2007 update task demonstrate the encouraging performance of the proposed approach.**

***Index Terms*—Temporal multi-document summarization, update summary, iterative feedback based manifold-ranking.**

I. Introduction

MULTI-DOCUMENT summarization is the process of automatically producing a summary delivering the main information content from a set of documents about an explicit or implicit topic, which has drawn much attention in recent years and exhibits the practicability in document management and search systems. For example, a number of news services, such as Google[1], NewsBlaster[2], and Sina News[3], have been developed to group news articles into news topics, and then produce a short summary for each news topic so as to facilitate users to browse the results and improve users' search experience. News portals usually provide concise headline news describing hot news topic each day and they also produce weekly news review to save user's time and improve service quality.

Temporal multi-document summarization (TMDS) is the natural extension of multi-document summarization, which captures evolving information of a single topic over time. It is assumed that a user has access to a stream of news stories that are on the same topic, but that the stream flows rapidly enough that no one has the time to look at every story. In this situation, a person would prefer to read the update information at a certain time interval under the assumption that the user has already read a number of previous documents. The update summary as defined for the DUC2007 new task just faces this goal, which is a kind of TMDS. For the DUC2007 update task, 100-word summaries has to be generated for three consecutive document subsets sorted by their publication dates, tracking the new development of a single topic through time.

The key problem of summarization is how to identify important content and remove redundant content. The common problem for summarization is that the information in different documents inevitably overlaps with each other, and therefore effective summarization methods are needed to contrast their similarities and differences. However, the above application scenarios, where the objects to be summarized face to some special topics and evolve with time, raise new challenges to traditional summarization algorithms. The first challenge for update summary task is that the information in the summary must be biased to the given topic, and the second is that the information in summary must contain the evolving content. So we need to take into account effectively this topic-biased and temporally evolving characteristics during the summarization process. Thus a good update summary must include information as much as possible, keeping information as novel as possible, and moreover, the information must be biased to the given topic.

In [23], an extractive approach based on manifold-ranking of sentences to topic-focused multi-document summarization by using the underlying manifold structure in data points is proposed without modeling the temporally evolving characteristic. Inspired by this, for the DUC2007 update task, we propose a new manifold-ranking frame based on iterative feedback mechanism, which has the temporally adaptive characteristic. We assume that the data points evolving over time have the long and narrow manifold structure. However,

Manuscript received May 10, 2008. Manuscript accepted for publication June 20, 2008.

He Ruifang, Qin Bing, Liu Ting, Liu Yang and Li Sheng are all with the Information Retrieval Lab, School of Computer Science and Technology, Harbin Institute of Technology, Harbin, 15001, China (phone: +86-451-86413683-801; fax: +86-451-86413683-812; e-mail: rfhe@ir.hit.edu.cn).

[1] http://news.google.com
[2] http://www1.cs.columbia.edu/nlp/projects.cgi\#newsblaster
[3] http://news.sina.com.cn

the common topic for three consecutive document subsets is a static query, which cannot represent the dynamically evolving information. Therefore, we use the iterative feedback mechanism to extend the topic by using the summarization of previous timeslices and the first sentences of documents in current timeslice. We believe this topic extension can represent the relay propagation of information in temporally evolving data and improve the ranking score. The proposed approach employs iterative feedback based manifold-ranking process to compute the ranking score for each sentence that denotes the biased information richness of sentence. Then the sentences highly overlapping with other informative ones are penalized by the greedy algorithm. The summary is produced by choosing the sentences with highest overall scores, which are considered informative, novel and evolving. In this improved manifold-ranking algorithm, the intra-document and inter-document relationships between sentences are differentiated with different weights. Experiments on datasets of DUC2007 update task demonstrate the competitive performance of the proposed approach.

The rest of this paper is organized as follows: Section 2 introduces related work. The details of the proposed approach are described in Section 3. Section 4 presents and discusses the evaluation results. We conclude this paper and discuss future work in Section 5.

II. Related Work

In recent years, a series of workshops and conferences on automatic text summarization (e.g. DUC[4] and NTCIR[5]), special topic sessions in ACL, COLING, and SIGIR have advanced the technology and produced a couple of experimental online systems.

Update summary is a new challenge in the field of summarization. It aims to capture evolving information of a single topic over time, and has the characteristics of the topic-focused and temporal summary. It hopes to extract the new information over time, and also must be biased to a certain topic. Generally speaking, the summarization methods can be either extractive summarization or abstractive summarization. Extractive summarization assigns salience scores to some units (e.g. sentences, paragraphs) of the documents and extracts the sentences with highest scores, while abstractive summarization usually needs sentence compression and reformulation. In this paper, we focus on extractive summarization.

The centroid-based method [20] is one of the most popular extractive summarization methods. The clustering based method [3] is also widely used, including term, sentence and sub-topic clustering. Most recently, the graph-ranking based methods, including TextRank [17] and LexRank [6], have been proposed for document summarization. Similar to PageRank [4] or HITS [11], these methods first build a graph based on the similarity relationships between the sentences in documents and then the importance of a sentence is determined by taking into account the global information on the graph recursively, rather than relying only on the local sentence-specific information. The basic idea underlying the graph-based ranking algorithm is that of ``voting'' or ``recommendation''. When a sentence links to another one, it is basically casting a vote for the linked sentence. The higher the number of votes that are cast for a sentence, the more important the sentence is. Moreover, the importance of the sentence casting the vote determines how important the vote itself is. The computation of sentence importance is usually based on a recursive form, which can be transformed into the problem of solving the principal eigenvector of the transition matrix.

Most topic-focused document summarization methods incorporate the information of the given topic or query into generic summarizers and extract sentences suiting the user's declared information need [21], [8], [5], [9], [7]. Very recently, Wan *et al.* [23] proposed an approach based on manifold-ranking. Their method tried to make use of relationships among all the sentences in the documents and the relationships between the given topic and the sentences. The ranking score is obtained for each sentence in the manifold-ranking process based on graph to denote the biased information richness of the sentence. Then the greedy algorithm is employed to impose diversity penalty on each sentence. The sentences with high ranking score are then selected as the output summary. More related work can be found on DUC2003 and DUC2005 publications.

Temporal summary originates from text summarization and topic detection and tracking (TDT), and is also related to time line construction techniques. Alan *et al*. [1] firstly put forward the concept of temporal summary inspired by TDT in SIGIR2001. Given a sequence of news reports on certain topic, they extract useful and novel sentences to monitor the changes over time. Usefulness is captured by considering whether a sentence can be generated by a language model created from the sentences seen to date. Novelty is captured by comparing a sentence with prior sentences. They report that it is difficult to combine the two factors successfully. Other researchers exploit distribution of events and extract the hot topics on time line by statistical measures. Swan and Allan [22] employ χ^2 statistics to measure the strength that a term is associated with a specified date, and then extract and group important terms to generate ``topics'' defined by TDT. In [12], Chen *et al*. import the aging theory to measure the ``hotness'' of a topic by analyzing the temporal characteristic of news report. The aging theory implies that a news event can be considered as a life form that goes through a life cycle of birth, growth, decay, and death, reflecting its popularity over time. Then hot topics are selected according to energy function defined by aging theory. Lim *et al*.[14] anchor documents on time line by the publication dates, and then extract sentences from each document based on surface features. Sentence weight is adjusted by local high frequency words in each time slot and global high frequency words from all topic sentences. They evaluate the system on Korean documents and report that time can help to raise the

[4] http://duc.nist.gov
[5] http://research.nii.ac.jp/ntcir/index-en.html

percentage of model sentences contained in machine generated summaries. Jatowt and Ishizuka [10] investigate the approaches to monitor the trends of dynamic web documents, which mean different versions of the same web documents. They employ a simple regression analysis on word frequency and time to identify whether terms are popular and active. The importance of a term is measured by its slope, intercept and variance. The weight of a sentence is measured by the sum of the weights of the terms inside the sentence. The sentences with highest scores are extracted into a summary. However, they do not report any quantitative evaluation results. In [16], Mani is devoted to temporal information extraction, knowledge representation and reasoning, and try to apply them to multi-document summarization. In [13], Li *et al*. explore whether the temporal distribution information helps to enhance event-based summarization based on corpus of DUC2001.

In DUC2007, the top performing systems of update summary task adopted the extractive methods. LCC's GISTexter [2] used Machine Reading mechanism with textual inference information to create new and coherent information. Textual entailment and textual contradiction are recognized to construct representations of knowledge coded in a text collection. Update summary is produced by comparing the entailment and contradiction of sentences. This method preferably fused the deep linguistic knowledge, however, which is difficult to be reconstructed. IIIT Hyderabad's system [19] estimated a sentence prior by a term clustering approach, which incorporated the query independent score and query dependent score in a linear combination way. Sentence reduction and entity dereferencing is also used in the algorithm. NUS [26] proposed a timestamped graph model motivated by human writing and reading processes, which is used to model the dynamic and evolutionary characteristic of information. It assumed that writers write articles from the first sentence to the last, and readers read articles from the first sentence to the last. These two processes are similar to evolution of citation networks and the web. Though the parameters of this model are very complex, the method is an interesting attempt.

Due to different tasks, the above researches do not uniformly fuse the information in the topic and the documents or just incorporate the temporal characteristics. While iterative feedback based manifold-ranking approach to the DUC2007 new update summary task can naturally and simultaneously take into account topic information and the relay propagation of information in temporally evolving data.

III. Iterative Feedback based Manifold-Ranking Approach

The iterative feedback based manifold-ranking approach for update summary consists of three steps: (1) iterative feedback mechanism is used to extend the topic; (2) manifold-ranking score is computed for each sentence in the iterative feedback based manifold-ranking process; (3) based on the manifold-ranking scores, the diversity penalty is imposed on each sentence. Overall ranking score of each sentence is obtained to measure both the importance degree of the sentence relevant to the sentence collection and topic and the novelty degree of information contained in the sentence with respect to all sentences in the summary. The sentences with high overall ranking scores are chosen for the summary.

A. Basic Definitions

The manifold-ranking method [24], [25] is a universal ranking algorithm and it is initially used to rank data points along their underlying manifold structure. However, this method cannot model the temporally evolving characteristic, say, which is not temporally adaptively. For the DUC2007 update task, we assume that the data points evolving over time have the long and narrow manifold-structure. However, the common topic for three consecutive document subsets is a static query, which cannot represent the dynamically evolving information. Therefore, we apply the iterative feedback mechanism to extend the topic by using the summarization of previous timeslices and the first sentences of documents in the current timeslice.

Iterative Feedback mechanism: Given a set of timeslices $TS = \{timeslice_i \mid 1 \le i \le m\}$ and a topic $T = \{topic_i \mid 1 \le i \le m\}$, every $timeslice_i = \{d_j \mid 1 \le j \le n\}$ consists of documents, every document consists of sentences. Let s_{ij} denotes the first sentence of document d_j in $timeslice_i$, then first sentences of all documents in $timeslice_i$ $s_{first}(i) = \{s_{ij} \mid 1 \le i \le m, 1 \le j \le n\}$. The timeslices are ordered chronologically. Every timeslice corresponds to an update summary. When summarizing, the current $timeslice_i$ just can refer to the previous timeslices from 1 to *i*-1, but cannot refer to the ones from *i*+1 to m. Let $updateSum_i$ denotes the update summary of the current $timeslice_i$, and then $topic_i$ is extended as follows:

$$topic_i = \{PubTopic \cup \bigcup_{k=1}^{i-1} updateSum_k \cup s_{first}(i) \mid 1 \le i \le m\}$$

PubTopic denotes the public topic description of all timeslices.

We assume this topic extension can represent the relay propagation of information in temporally evolving data and help to capture the changes of a single topic over time.

B. Modified Manifold-Ranking Process

Given a query and a set of data points, the task of manifold-ranking is to rank the data points according to their relevance to the query [24]. The key to manifold-ranking is the prior assumption of consistency, which means: (1) nearby points are likely to have the same ranking scores; (2) points on the same structure (typically referred to as a cluster or a manifold) are likely to have the same ranking scores.

In our context, the data points are denoted by the topic description and all the sentences in the documents, where topic description dynamically evolves over time. The iterative feedback based manifold-ranking process in our context can be formalized as follows:

For $timeslice_i$, given a set of data points $X=\{x_1,...,x_t,x_{t+1},...,x_n\}\subset R^m$, the first t data points are the topic description and the rest data points are the sentences in the documents. According to the iterative feedback mechanism, x_1 denotes the *PubTopic*, $x_2 \ldots x_p$ denotes the and $x_{p+1} \ldots x_t$ denotes the $s_{first}(i)$. Note that because the *PubTopic* is usually short in our experiments, we treat it as a pseudo-sentence. Then it can be processed in the same way as other sentences. Let $f:X\to R$ denotes a ranking function which assigns to each point $x_q(1 \le q \le n)$ a ranking value f_q . We can view f as a vector $f=[f_1,..., f_n]^T$. We also define three vectors, $Y_1=[y_1,..., y_n]^T$, in which $y_1=1$ because x_1 is the *PubTopic* and $y_q=0(2 \le q \le n)$ for all the sentences in the documents; similarly, $Y_2=[y_1,..., y_n]^T$, in which $y_2 \ldots y_p=1$ because $x_2 \ldots x_p$ denotes $\bigcup_{k=1}^{i-1} updateSum_k$ and $y_q=0(q=1, p+1 \le q \le n)$; $Y_3=[y_1,..., y_n]^T$, in which $y_{p+1} \ldots y_t=1$ because $x_{p+1} \ldots x_t=1$ denotes the $s_{first}(i)$ and $y_q=0(1 \le q \le p,\ t+1 \le q \le n)$. The iterative feedback based manifold-ranking algorithm goes as follows:

In the first step of the algorithm, a connected network is formed. We remove the stop words in each sentence, and stem the remaining words. The weight associated with term *t* is calculated with the $tf_t * isf_t$ formula, where tf_t is the frequency of term *t* in the sentence and isf_t is the inverse sentence frequency of term *t*, i.e. $1+\log(N/n_t)$, where *N* is the total number of sentences and n_t is the number of the sentences containing term *t*. Then $sim(x_i, x_j)$ is computed according to the normalized inner product of the corresponding term vectors. The network is weighted in the second step and the weight is symmetrically normalized in the third step. The normalization in the third step is necessary to prove the algorithm's convergence. The fourth step is the key step of the algorithm, where all points spread their ranking score to their neighbors via the weighted network. The spread process is repeated until a global stable state is achieved, and we get the ranking score in the fifth step. The parameter α specifies the relative contributions to the ranking scores from neighbors and the initial ranking scores, and the parameter β , γ , η denotes the relative contribution to ranking scores from the *PubTopic*, the update summary in the previous timeslices and the first sentences of all documents in the current timeslice, respectively. Note that self-reinforcement is avoided since the diagonal elements of the affinity matrix are set to zero.

Algorithm 1. Iterative feedback based manifold-ranking

Input: $X=\{x_1,...,x_n\}$

Output: $f=\{f_i^* \mid i=1...n\}$

1: Compute the pair-wise similarity values between sentences (data points) using the standard Cosine measure. Given two sentences x_i and x_j , the Cosine similarity is denoted as $sim(x_i, x_j)$, computed as the normalized inner product of the corresponding term vectors;

2: Connect any two points with an edge if their similarity value exceeds 0. We define the affinity matrix *W* by $W_{ij}=sim(x_i, x_j)$ if there is an edge linking x_i and x_j . Note that we let $W_{ii}=0$ to avoid loops in the graph built in next step;

3: Normalize *W* by $S=D^{-1}W$ in which *D* is the diagonal matrix with (*i*,*i*)-element equal to the sum of the *i*-th row of *W*;

4: Iterate $f(t+1)=\alpha Sf(t)+(\beta Y_1+\gamma Y_2+\eta Y_3)$ until convergence, where α, β, η are parameters in (0,1);

5: Let f_i^* denote the limit of the sequence $\{f_i(t)\}$. Each sentence x_i gets its ranking score f_i^* ;

For the original manifold-ranking, the iterative formula of the fourth step is $f(t+1)=\alpha Sf(t)+(1-\alpha)Y$. The theorem in [24] guarantees that the sequence $f(t)$ converges to

$$f^*=(I-\alpha S)^{-1}Y \quad (1)$$

Without loss of the generality, we can extend the vector *Y*. Since $(I-\alpha S)$ is invertible, we have

$$f^*=(I-\alpha S)^{-1}(\beta Y_1+\gamma Y_2+\eta Y_3) \quad (2)$$

For real-world problems, the iteration algorithm is preferable due to high computational efficiency. Usually when the difference between the scores computed at two successive iterations for any point falls below a given threshold (0.0001 in this paper), the iteration algorithm will converge.

Wan *et al*. [23] proposed and proved an intuition that intra-document links and inter-document links have unequal contributions in the manifold-ranking algorithm. Given a link between a sentence pair of x_i and x_j , if x_i and x_j come from the same document, the link is an intra-document link; if x_i and x_j come from different documents, the link is an inter-document link. The links between the topic sentences and any other sentences are all inter-document links. In our context, distinct weights are assigned to the intra-document links and the inter-document links respectively. In the second step of the above algorithm, the affinity matrix *W* can be decomposed as

$$W = W_{intra} + W_{inter} \quad (3)$$

where W_{intra} W_{intra} is the affinity matrix containing only the intra-document links (the entries of inter-document links are set to 0) and W_{inter} is the affinity matrix containing only the inter-document links (the entries of intra-document links are set to 0). $RankScore(x_i) = f_i^*(i = 1,...,n)$

We differentiate the intra-document links and inter-document links as follows:

$$W' = \lambda_1 W_{intra} + \lambda_2 W_{inter} \quad (4)$$

We let $\lambda_1, \lambda_2 \in [0,1]$ in the experiments. If $\lambda_1 \le \lambda_2$, the inter-document links are more important than the intra-document links and vice versa. Note that if $\lambda_1 = \lambda_2 = 1$, then Equation(4) reduces to Equation(3). In the iterative feedback based manifold-ranking algorithm, W' is normalized into S' in the third step and the fourth step uses the following iteration form: $f(t+1) = \alpha S' f(t) + (\beta Y_1 + \gamma Y_2 + \eta Y_3)$. The iteration process is shown in Algorithm 2:

Algorithm 2. Power method for computing the stable state of iterative feedback based manifold-ranking

Input: Normalized similarity matrix S'
Input: Matrix size *N*, error tolerance ε
Output: Eigenvector *f*

1: $f(0) = \frac{1}{N}$;
2: t=0;
3: **repeat**;
4: $f(t+1) = \alpha S'^T f(t) + (\beta Y_1 + \gamma Y_2 + \eta Y_3)$
5: $t = t + 1$;
6: $\delta = \| f(t+1) - f(t) \|$;
7: **until** $\delta < \varepsilon$;
8: return $f(t+1)$;

C. Redundancy Removing in Sentence Selection

Based on the normalized original affinity matrix, we apply the greedy algorithm to impose the diversity penalty and compute the final overall ranking scores, representing the importance and relevance to topic and the information novelty of the sentences. For each $timeslice_i$, the algorithm is shown in Algorithm 3:

The algorithm is based on the idea that the overall ranking score of less informative sentences overlapping with the sentences in update summary is decreased. In the second step, where $\omega > 0$ is the penalty degree factor. The larger ω is, the greater penalty is imposed to the overall ranking score. If $\omega = 0$, no diversity penalty is imposed at all. The sentence with highest ranking score is chosen to produce the summary until satisfying the summary length limit.

Algorithm 3. Redundancy removing

Input: Initialize Summary sentences set $A = \phi$, $B = \{x_i \mid i = 1,...,n\}$
Input: $RankScore(x_i) = f_i^*(i = 1,...,n)$, each sentence's overall ranking score is its manifold-ranking score
Output: A

1: Sort the sentences in *B* by their current overall ranking scores in descending order;
2: Suppose x_i is the highest ranked sentence, i.e. the first sentence in the ranked list. Move sentence x_i from *B* to *A*, and then the diversity penalty is imposed to the overall ranking score of each sentence linked with $x_i \in B$ as follows: for each sentence $x_j \in B$,
$RankScore(x_i) = RankScore(x_j) - \omega * S_{ji} f_i^*$
3: Go to step 2 and iterate until $B = \phi$ or exceed the summary length limit;

IV. Experiments

A. Data Set

The dataset of the DUC2007 update summary task is used in our experiments. The update summary task is the first evaluation about TMDS. This task includes a gold standard dataset consisting of document cluster and reference summaries. Ten documents clusters are selected from the 45 clusters of the main task for preparation of the update summary task, and each cluster has 25 documents. Each of these ten clusters is divided into three smaller clusters, A, B, C, where the time stamps on all the documents in each set are ordered such that time(A)<time(B)<time(C). There are approximately 10 documents in A, 8 in B, and 7 in C. The three smaller clusters have the same query as the original larger cluster. The goal of the update summary task is to create short (100-word) multi-document summaries for each smaller clusters under the assumption that the reader has already read a number of previous documents.

B. Evaluation Metric

In order to evaluate the performance and the stability of the proposed approach, we used two kinds of evaluation metrics.

ROUGE [15] is used as the evaluation metric, which has been widely adopted by DUC for automatic summarization evaluation. It measured summary quality by counting overlapping units such as the n-gram, word sequences and word pairs between the candidate summary and the reference summary. ROUGE toolkit reported separate scores for 1, 2, 3 and 4-gram, and also for longest common subsequence co-occurrences and so on. Among these different scores, unigram-based ROUGE score (ROUGE-1) has been shown to agree with human judgment most. The evaluation results of DUC2007 update summary just gave the ROUGE-2 and ROUGE-SU4 scores. Accordingly, we also showed corresponding ROUGE metrics in the experimental results at a

confidence level of 95%, which were computed by running ROUGE-1.5.5[6] with stemming but no removal of stopwords. The input file implemented jackknifing so that scores of systems and humans could be compared.

Pyramid method [18] is also used to evaluate our proposed approach, which is the latest evaluation metric. It incorporates the idea that no single best model summary for a collection of documents exists. The analysis of summary content is based on Summarization Content Units (SCUs). The reference summary is annotated as the set of the SCUs. If the SCUs is contained in more reference summary, it will have the higher weight. After the annotation procedure is completed, the final SCUs can be partitioned in a pyramid. The partition is based on the weight of the SCUs; each tier contains all and only the SCUs with the same weight.

C. Experimental Results and Analysis

ROUGE Metric. We designed seven baselines in addition to the lead baseline (RUNID=35) and the CLASSYO4 baseline (RUNID=58) employed in the update task of DUC2007. We also compared our system with top five systems with highest ROUGE scores, chosen from the performing systems on update task of DUC2007. The comparison results are showed in Table I.

TABLE I
SYSTEM COMPARISON AND RANK ON UPDATE TASK OF DUC 2007
(RECALL SCORE)

System	ROUGE-2	Rank	ROUGE-SU4	Rank
40	0.11189	1	0.14306	1
IFM-ranking	0.09963	2	0.13176	5
55	0.09851	3	0.13509	3
45	0.09622	4	0.13245	4
IFM-ranking-γ	0.09404	5	0.12705	8
IFM-ranking-ω	0.09389	6	0.12985	7
47	0.09387	7	0.13052	6
44	0.0937	8	0.13607	2
IFM-ranking-$\lambda_1 : \lambda_2$	0.09206	9	0.12638	9
IFM-ranking-β	0.09019	10	0.12402	10
IFM-ranking-$\gamma - \eta$	0.0872	11	0.12342	11
IFM-ranking-η	0.08503	12	0.1231	12
CLASSY04(58)}	0.08501	13	0.12247	13
IFM-ranking-α	0.07852	14	0.11523	14
Lead Baseline(35)}	0.04543	15	0.08247	15

The Lead Baseline returns all the leading sentences (up to 100 words) of the most recent document. CLASSY04 Baseline ignores the topic narrative, but which had the highest mean SEE coverage score in Task 2 of DUC2004, a multi-document summarization task. The system uses the CLASSY04 HMM[7] terms as observables and the pivoted QR method for redundancy removal. The sentences are chosen only from the most recent collection of documents. For example, the summary for D0703A-B selects sentences only from the 8 articles in this cluster; however, it uses D0703A-A in the computation of signature terms. Likewise, the summary for D0703A-C selects sentences from only the 7 documents in this cluster and only uses D0703A-A and D0703A-B in the computation of signature terms. S40, S55, S45, S47 and S44 are the system IDs of the top performing systems, whose details are described in DUC publications.

IFM-ranking (Iterative Feedback based Manifold-ranking) is our system, which adopts the proposed approach described in Section 3. IFM-ranking-α, IFM-ranking-β, IFM-ranking-γ, IFM-ranking-η, IFM-ranking-ω, IFM-ranking-($\lambda_1 : \lambda_2$) and IFM-ranking-$\gamma - \eta$ are seven other baselines. IFM-ranking-α ignores spreading the data points' ranking score to their nearby neighbors via the weighted network. IFM-ranking-β, IFM-ranking-γ, IFM-ranking-η ignores the common topic, the update summary of previous timeslices and first sentences of all document in current timeslice when extending the topic, respectively. IFM-ranking-$\gamma - \eta$ ignores the iterative feedback mechanism, which just considers the common topic in manifold-ranking process. IFM-ranking-$\lambda_1 : \lambda_2$ doesn't differentiate the link between the sentences, say $\lambda_1 : \lambda_2 = 1$. IFM-ranking-$\omega$ just computes the ranking score of each sentence without the step of imposing diversity penalty. These baselines are all simplified versions of IFM-ranking.

We conduct experiments to focus on the following research questions, which are related to 7 IFM-ranking parameters α, β, γ, η, λ_1, λ_2, ω.

Q1: Is the modified manifold-ranking process useful?
Q2: Is the iterative feedback mechanism effective?
Q3: Does the update summary in previous timeslice or the first sentences of documents in current timeslice help to extend the information richness of topic?
Q4: How does the intra-document or inter-document link affect the performance?
Q5: Is redundancy removing necessary?

The parameters of the IFM-ranking are set as follows: α=0.8, β=0.7, γ=0.3, η=0.4, λ_1=0.3, λ_2=1, ω=8.5.

Seen from Table I, our system ranks 2th and 5th on ROUGE-2 and ROUGE-SU4, respectively, and outperforms all baseline systems.

In comparison with IFM-ranking, ROUGE-2 and ROUGE-SU4 scores of IFM-ranking-α decrease by 0.02111 and 0.01653. Therefore, modified manifold-ranking process affects the update task and parameter α is very important.

It is shown in IFM-ranking-β, IFM-ranking-γ and IFM-ranking-η that the topic description helps to improve the performance, and both the update summary in previous timeslice and the first sentences of documents in current timeslice are beneficial to extend the information richness of topic. At the same time, parameter η brings the highest contribution on performance, β takes the second place, and γ takes third place. It also shows that the first sentence can availably generalize the topic in news field.

[6] http://www-nlpir.nist.gov/projects/duc/duc2007/tasks.html
[7] http://duc.nist.gov/pubs/2004papers/ida.conroy.ps

ROUGE-2 and ROUGE-SU4 scores of IFM-ranking-$\gamma-\eta$ decrease by 0.01143 and 0.00834 in comparison with IFM-ranking. This result verifies that iterative feedback mechanism is effective, which models the dynamically evolving characteristic, and represents the relay propagation of information in temporally evolving data.

If IFM-ranking-$\lambda_1:\lambda_2$ doesn't differentiate the links between the sentences (where $\lambda_1:\lambda_2=1$), its ROUGE scores will slightly decrease by 0.00757 and 0.00538 than that of IFM-ranking, respectively. Thus intra/inter-document link differentiation affects the update task.

Without the step of imposing diversity penalty, ROUGE scores of IFM-ranking-ω will decrease by 0.00574 and 0.00191, respectively. Therefore, redundancy removing is necessary.

Comparing with the performing system (RUNID=40) [2] with the highest ROUGE scores respectively on the sub dataset A, B, C of DUC2007 update task, it is shown in table II and table III that our ROUGE scores on A and B are lower than that of the performing system 40. However, ROUGE scores on C are both higher than that of ones by 0.009514 and 0.00555. The performing system 40 adopted much linguistic knowledge and discourse understanding techniques. Knowledge base and coreference resolution are used to evaluate whether a particular extracted commitment is a textual entailment or textual contradiction. However, we just used the shallow sentence-level feature. This further validates that our proposed approach is effective in capturing the update information.

TABLE II
ROUGE-2 RECALL SCORES FOR THREE SUBSETS A, B, C ON UPDATE TASK OF DUC2007

System	A	B	C
40	0.125132	0.105644	0.104285
IFM-ranking	0.0983582	0.086997	0.113799

TABLE III
ROUGE-SU4 RECALL SCORES FOR THREE SUBSETS A,B,C ON UPDATE TASK OF DUC2007

System	A	B	C
40	0.155344	0.134188	0.139419
IFM-ranking	0.130028	0.120542	0.144969

Pyramid Metric. Altogether, there are in total 30 standard pyramids created by annotators. Figure 1 shows the average score, maximum score and our system's score for each pyramid set. IFM-ranking outperforms the average scores in 22 out of 30 sets. Note that for dataset C, the proposed IMF-ranking approach performs better than average performance in 7 out of 10 sets, which shows that iterative feedback mechanism is effective. The average scores over all pyramid sets are show in Figure 2, the best system has the average score of 0.3403, whereas our system obtains 0.29855 on average, which is ranked 4th among all 24 systems. This further shows our approach is stable.

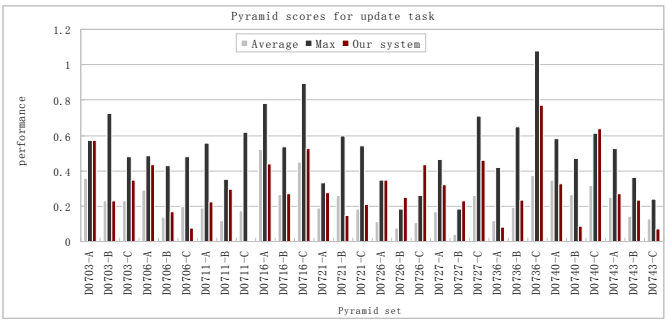


Fig. 1. Pyramid scores for update task.

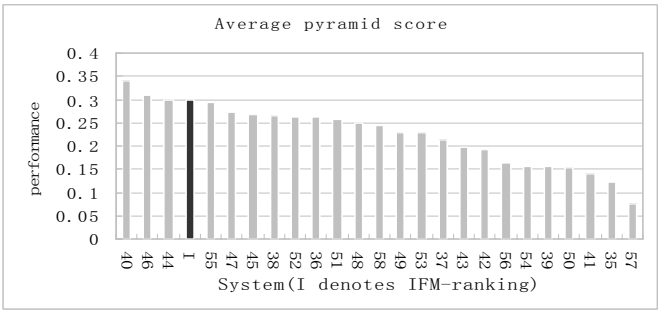


Fig. 2. Average pyramid scores for update task.

Since the update summary task is firstly evaluated in 2007, and we have no other training corpus, thus we cannot directly compare with the top performing system. However, we just use the shallow sentence-level features to achieve the encouraging performance; it will have some instruction on the future participation.

The experiment results suggested that the encouraging performance achieved by IFM-ranking benefits from the following factors: 1) Modified manifold-ranking process; 2) Iterative feedback mechanism; 3) Intra/Inter-document link differentiation; 4) Diversity penalty imposition.

D. Parameter Tuning

As the parameter space is too large to test all possible IFM-ranking algorithms, we adopt the greedy strategy to find the proper parameters value based on ROUGE metric, however, which are impossible optimal. Figures from 3 to 8 show the process of parameter tuning.

When we tune a parameter, the other parameters are set to be the optimal values selected by greedy strategy. Figure 3 demonstrates the influence of the manifold weight α in the proposed approach on performance when β =0.7, γ =0.3, η =0.4, λ_1 =0.3, λ_2 =1, ω =8.5.

Figure 4, Figure 5 and Figure 6 demonstrate the influence of the common topic (β), the update summary in previous timeslices (γ), and the first sentences of documents in current timeslice (η), respectively. From these three figures, it could be observed that both ignoring and excessively depending on the topic description would deteriorate the performance.

Figure 7 demonstrates the influence of the intra/inter-document relationship differentiating weight $\lambda_1:\lambda_2$. It could be observed that the performance curve in field

($\lambda_1 : \lambda_2 < 0.9$) is averagely higher than that in field (λ_1=1 and $\lambda_2 < 0.9$). It shows that inter-document relationships are more important than intra-document relationships for the update task.

Figure 8 demonstrates the influence of the penalty factor ω. It shows that imposing diversity penalty is necessary.

V. Conclusions and Future Work

This paper proposes the iterative feedback based manifold-ranking for update task of DUC2007. Feedback mechanism is used to model the dynamically evolving characteristic, which reveals the relay propagation of information in temporally evolving data. The proposed approach also makes full use of the relationships among sentences and relationships between the topic and the sentences.

However, our approach just used the shallow sentence-level feature, and adopted the greedy strategy to estimate the parameter values, which may be not optimal. In the future, we will mine the deeper level features including temporal event and semantic information, and also explore the parameter optimization algorithm.

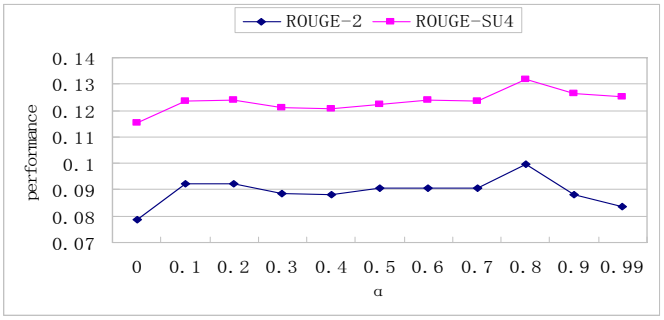


Fig. 3. α vs ROUGE recall scores.

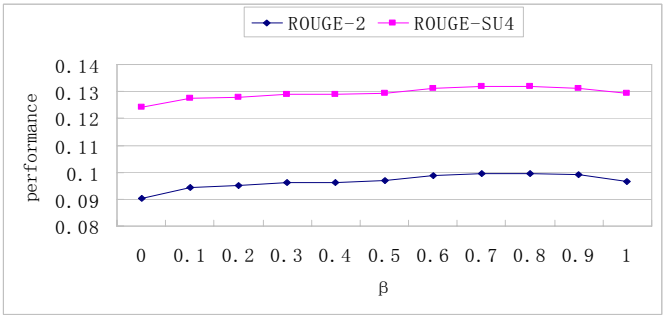


Fig. 4. β vs ROUGE recall scores.

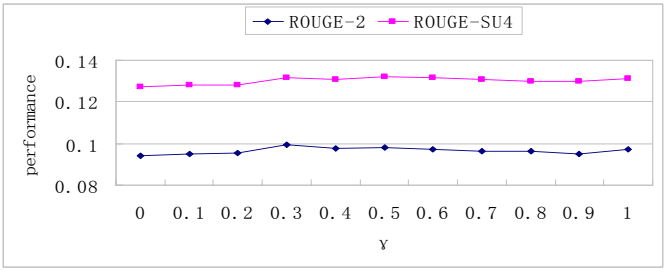


Fig. 5. γ vs ROUGE recall scores.

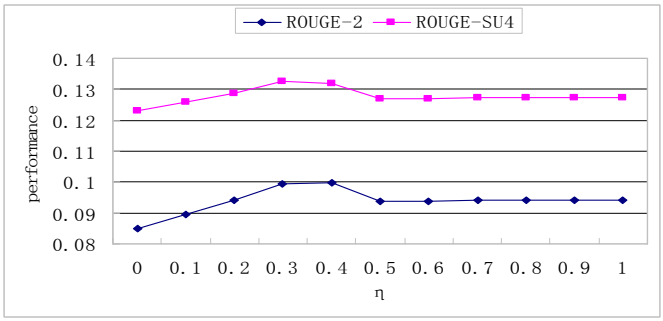


Fig. 6. η vs ROUGE recall scores.

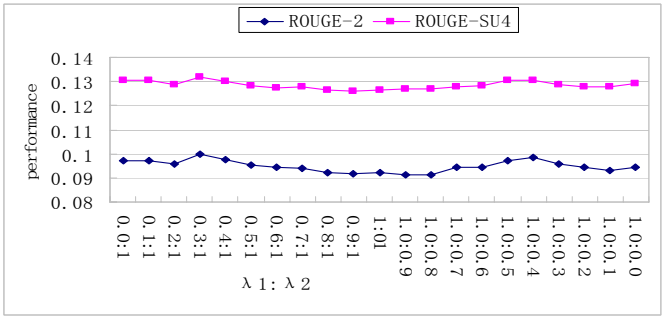


Fig. 7. $\lambda_1 : \lambda_2$ vs ROUGE recall scores.

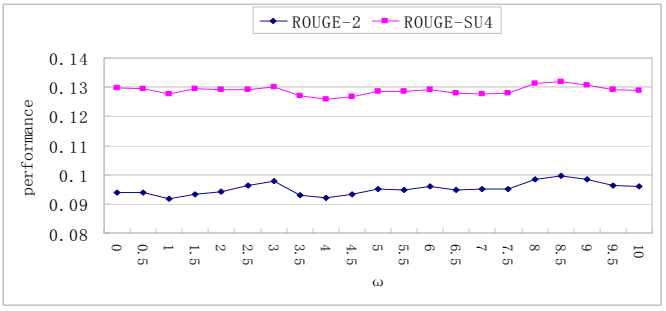


Fig. 8. ω vs ROUGE recall scores.

References

[1] Allan, R. Gupta, and V. Khandelwal. Temporal summaries of new topics. *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 10–18, 2001. K. R. Andrew Hickl and F. Lacatusu. LCC's GISTexter at DUC 2007: Machine Reading for Update Summarization. Proceedings of the DUC2007.

[2] K. R. Andrew Hickl and F. Lacatusu. LCC's GISTexter at DUC 2007: Machine Reading for Update Summarization. *Proceedings of the DUC2007*.

[3] Q. Bing, L. Ting, C. Shanglin, and L. Sheng. Sentences Optimum Selection for Multi-Document Summarization. Journal of Computer Research and Development, 43(6):1129–1134, 2006.

[4] S. Brin and L. Page. The anatomy of a large-scale hypertextual Web search engine. Computer Networks and ISDN Systems, 30(1-7):107–117, 1998.W.-K. Chen, *Linear Networks and Systems* (Book style). Belmont, CA: Wadsworth, 1993, pp. 123–135.

[5] J. Conroy, J. Schlesinger, and J. Stewart. CLASSY query based multi-document summarization. Proceedings of the 2005 Document Understanding Workshop, Boston, 2005.

[6] G. Erkan and D. Radev. LexRank: Graph-based Lexical Centrality as Salience in Text Summarization. Journal of Artificial Intelligence Research, 22:457–479, 2004.

[7] A. Farzindar, F. Rozon, and G. Lapalme. CATS a topic oriented multi-document summarization system at DUC 2005. Proceedings of the 2005 Document Understanding Workshop.

[8] J. Ge, X. Huang, and L. Wu. Approaches to event-focused summarization based on named entities and query words. Proceedings of the 2003 Document Understanding Workshop.

[9] E. Hovy, C. Lin, and L. Zhou. A BE-based multi-document summarizer with query interpretation. Proceedings of the DUC2005.

[10] A. Jatowt and M. Ishizuka. Temporal Web Page Summarization. 5th International Conference On Web Information Systems Engineering, Brisbane, Australia, November 22-24, 2004.

[11] J. Kleinberg. Authoritative sources in a hyperlinked environment. Journal of the ACM (JACM), 46(5):604–632,1999.

[12] C. Kuan-Yu, L. Luesukprasert, and T. Seng-cho. Hot Topic Extraction Based on Timeline Analysis and Multidimensional Sentence Modeling. IEEE Transactions on Knowledge and Data Engineering 19, 8 (Aug. 2007), pages1016–1025, 2007.

[13] M. L. Q. W. K. Li, W.J.and Wu. Integrating temporal distribution information into event-based summarization. International Journal of Computer Processing of Oriental Languages, 19:201–222, 2006.

[14] J. Lim, I. Kang, J. J.Bae, and J. Lee. Sentence extraction using time features in multi-document summarization. In Proceedings of the Asia Information Retrieval Symposium 2004, pages 82–93.

[15] C. Lin. ROUGE: A Package for Automatic Evaluation of Summaries. Proceedings of the Workshop on Text Summarization Branches Out, pages 25–26, 2004.

[16] I. Mani. Recent Developments in Temporal Information Extraction (Draft). Nicolov, N., and Mitkov, R. Proceedings of RANLP, 3, 2004.

[17] R. Mihalcea and P. Tarau. TextRank: Bringing Order into Texts. In In Proceedings of Empirical Methods in Natural Language Processing 2004.

[18] A. Nenkova, R. Passonneau, and K. McKeown. The pyramid method: Incorporating human content selection variation in summarization evaluation. ACM Trans. Speech Lang. Process., 4(2):4, 2007.

[19] R. K. Prasad Pingali and V. Varma. IIIT Hyderabad at DUC 2007. Proceedings of the DUC2007.

[20] D. Radev, H. Jing, M. Sty´s, and D. Tam. Centroid based summarization of multiple documents. Information Processing and Management, 40(6):919–938, 2004.

[21] H. Saggion, K. Bontcheva, and H. Cunningham. Robust Generic and Query based Summarization. 10th Conference of the European Chapter of the Association for Computational Linguistics, EACL-2003.

[22] R. Swan and D. Jensen. Constructing Topic-Specific Timelines with Statistical Models of Word Usage. Proceedings of the 6th ACM Conference on Knowledge Discovery and Data Mining (SIGKDD), pages 73–80, 2000.

[23] X. Wan, J. Yang, and J. Xiao. Manifold-ranking based topic-focused multi-document summarization. In IJCAI, pages 2903–2908, 2007.

[24] D. Zhou, O. Bousquet, T. Lai, J. Weston, and B. Scholkopf. Learning with Local and Global Consistency. In Proceedings of NIPS2003, 2003.

[25] D. Zhou, J. Weston, A. Gretton, O. Bousquet, and B. Scholkopf. Ranking on Data Manifolds. In Proceedings of NIPS2003, 2003.

[26] M.-Y. K. W. S. L. L. Q. Ziheng Lin, Tat-Seng Chua and S. Ye. NUS at DUC 2007: Using Evolutionary Models of Text. Proceedings of the DUC2007.

Improvement of Queries using a Rule Based Procedure for Inflection of Compounds and Phrases

Ranka M. Stanković

Abstract—**The selection of words chosen for a query, crucial for the quality of results obtained by the query, can be substantially improved by using various lexical resources. Thus, for example, morphological dictionaries enable morphological expansion of queries, which is very important in highly inflective languages, such as Serbian. This paper discusses issues related to improvement of queries using a rule based procedure implemented in WS4LR, a workstation for manipulating heterogeneous lexical resources developed by the Human Language Technology Group at the University of Belgrade. The procedure is used for automatic production of lemmas for a morphological dictionary from a given list of compounds, and its evaluation on several different sets of data is given. Several examples illustrate how this procedure can be used for improvement of queries for web search engines. Results obtained for these examples show that the number of documents obtained through a query by using our approach can be remarkably increased.**

Index Terms—**Electronic dictionary, inflection, compounds, query expansion.**

I. Introduction

The Human Language Technology group from University of Belgrade (HLT) has been developing various lexical resources over quite a long period, reaching a considerable volume to date. HLT group has produced an integrated and easily adjustable tool, a workstation for language resources, labeled WS4LR, which greatly enhances the potential of manipulating each particular resource as well as several resources simultaneously [1]. This tool has already been successfully used for various language processing related tasks including query expansion.

Dictionaries are one of the most important resources in various phases of the automatic analysis of text [2]. The system of morphological electronic dictionaries of Serbian follows the methodology and format (known as DELAS/DELAF) presented in [3]. E-dictionaries of simple word forms have reached a considerable size: approximately 120,000 entries in total [4].

In recent years the interest for multi-word units and compounds is growing rapidly, and this paper focuses on the morphological description of compounds compatible with the methodology used for simple words. At present, the dictionary of compounds has 2633 lemmas covering different parts of speech.

Development of the dictionary of compounds is not an easy task, so automated creation of lemmas for such a dictionary for a given list of compounds is of great importance. Such a procedure, which is based on rules and relies on data from e-dictionaries of simple words is described in Section II. The developed procedure has been evaluated on several different data sets and afterwards included in WS4LR.

Section III of this paper demonstrates how the described procedure can be used for query improvement. WS4LR architecture is described with special attention to compound management system. Usages of various lexical resources for query improvement are given, with integrated module for automatic detection of structure and inflectional characteristics of compounds. The application of the procedure presented is demonstrated on several examples of morphological expansion of key phrases for web search engines.

II. A Rule Based Procedure for Inflection of Compounds and Phrases

A. Compounds Dictionary

Morphological description of compounds, compatible with the methodology used for simple words, relies on the usage of Finite-State Technology [5]. The final aim is to produce the counterpart of DELAS/DELAF dictionaries of simple words for compounds – DELAC/DELACF.

The following example illustrates the content of compound dictionaries and some problems in their development. For example, the compound *beli medved* 'polar bear' should be entered in the DELAC dictionary of compounds [6], as follows:

beli(beo.A38:adms1g) medved(medved.N2:ms1v),
NC_AXN+N+Comp+Zool

Information contained in this entry should provide for automatic creation of all inflected forms for the DELACF dictionary, such as:

beloga medveda,beli medved.NC_AXN:ms4v

Manuscript received on May 9, 2008. Manuscript accepted for publication June 20, 2008. The presented work was done within the Human Language Technology group, University of Belgrade, Serbia.

Ranka M. Stanković is with the Faculty of Mining and Geology, University of Belgrade, Đušina 7, 11000 Belgrade, Serbia (phone: +381 11 3219-148; fax: +381 11 3243 978; e-mail: ranka@rgf.bg.ac.yu).

beli medvede,beli medved.NC_AXN:ms5v
belim medvedom,beli medved.NC_AXN:ms6v

The production of a lemma in the DELAC dictionary for a given compound proceeds in several steps:

1) For each compound component determine its lemma in DELAS dictionary with inflectional class code, and grammatical categories from the DELAF dictionary. For instance, for *beli* the lemma is *beo*, its inflectional class code is A38, and grammatical categories of the form *beli* are :adms1g;
2) Determine the inflectional class code for the compound (e.g. NC_AXN in the above example);
3) Determine the syntactic and semantic markers for the compound (e.g. +N+Comp+Zool in the above example).

In order to facilitate this task, a special tool within WS4LR [7] has been developed that assists in obtaining some of the necessary information from existing DELAS/DELAF dictionaries. Even with the help of this tool (for instance by reducing the number of errors in DELAC entries), the development of DELAC dictionary for Serbian is very time consuming. This led to the decision to develop a procedure for automatic (or semiautomatic) construction of DELAC type dictionary from a given list of compounds.

B. Rules Design

The procedure for automatic construction of DELAC type dictionary is based on a set of rules. The rule design strategy is a result of expert knowledge on morphology and the analysis of an existing manually created compound dictionary. The task of the rule based procedure is to generate the complete compound lemma for the dictionary of DELAC type based on the strategy. However, the strategy and the procedure are independent, and changes in the strategy, in general do not affect the procedure itself. This system design made experiments with various rule strategies possible – the final strategy used to evaluate the procedure is a result of several iterations.

The rule based strategy presently consists of 53 rules: 19 rules for compounds with 2 components, 20 rules for compounds with 3 components, 8 rules for compounds with 4 components, and 3 rules for compounds with 5 and 6 components. Each rule defines conditions components of a particular compound and/or separators between them must fulfill in order to get a particular inflectional class assigned to them. The rules are applied in the order they are listed.

Conditions defined for each rule are of two types: the first type specifies grammatical categories of compound components and they usually apply to the components that inflect, while the second type specifies additional conditions like semantic and/or syntactic markers. This can best be illustrated by the example of rule number 43, as shown in the table I.

This rule is applied as follows: if the first component satisfies (according to the dictionary of simple words) the grammatical conditions (which imply that the first component has to be a noun), and if the second and the third component and the separator between them satisfy one of the remaining additional conditions, then the rule class will be suggested for the given compound. The frequency column gives the number of compounds in the existing DELAC dictionary that satisfy the particular rule line. Examples of additional conditions are:

TABLE I
EXAMPLE OF RULE NUMBER 43, CLASS NC_N6X

Class	Gramm. condition	Frequency	Additional conditions
NC_ N6X	_:fs1q__	3	(The first component is a noun)
	_:ms1q__	2	AND
	_:ms1v__	2	((The second, the third and fourth
	_:ns1q__	1	component are in genitive) OR
	_:fs1v__	0	(The second word is a preposition
	_:ns1v__	0	and the third word agrees with it))

1) *tehnolog održavanja poljoprivredne mehanizacije* 'agricultural equipment maintenance technologist' where *tehnolog* 'technologist' is a noun that satisfies the condition :ms1v, *održavanja* is in the genitive case (from *održavanje* 'maintenance'), *poljoprivredne* is in the genitive case (from *poljoprivredni* 'agricultural')and *mehanizacije* is in the genitive case (from *mehanizacija* 'equipment');
2) *motor sa unutrašnjim sagorevanjem* 'engine with combustion chamber' where *motor* 'engine' satisfies the condition :ms1q, *sa* 'with' is a preposition that requires the instrumental case, *unutrašnjim* is in the instrumental case (from *unutrašnji* 'combustion');

All rule lines are ordered according to the listed frequency in order to prioritize some conditions in case of multiple choices. The total number of lines for 53 rules is 1014. Most inflectional classes have only one rule, with multiple rules defined only for a minority. These rules model different conditions, and they have different order in the strategy, which reflects the probability of their application.

Figure 1 depicts the XSD scheme of rules for automatic detection of the structure and inflectional characteristics of compounds. As an example, the XML form of rule number 43, for inflectional class NC_N6X, is presented in table II.

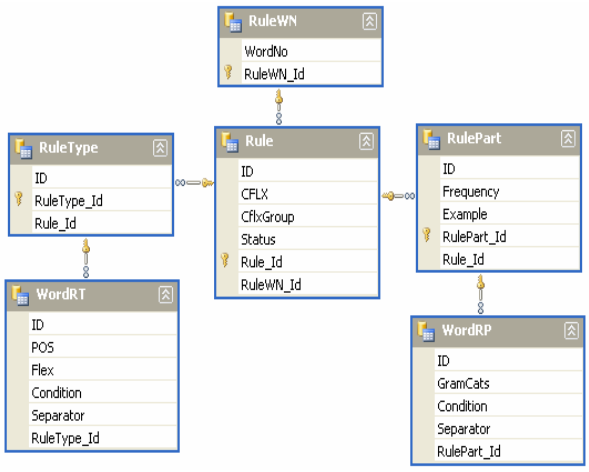


Fig. 1. XSD scheme of rules for the automatic detection of structure and inflectional characteristics of compounds

The pseudo code for automatic construction of a compound lemma goes as follows:

```
predictCFlexLema(Compound)
    // 1) lexical analysis of compound components
   generateDlf(Compund)
   foreach Component in Compound.Components
      Component.findPosLemasFromDlf
      Component.findGramCatsFromDlf
      Component.findLemasFlxCodeFromDelas
      Compund.DataSet.Add(Candidate)
   // 2) Selection of possible rules to be applied
   Rls=Rules.SelectByNumberOfComponents
   Rls.FilterByPosOfComponents
   foreach R in Rls
      dsRt=R.getDataSetRuleType(C.DataSet))
      dsRp=R.getDataSetRulePart(DsRt)
      //3) Construction of compound lemma
      foreach dr in dsRp
         Rule.GenerateCLema(dr)
```

TABLE II
XML FORM OF RULE NUMBER 43, CLASS NC_N6X

```
<Rule ID="43" CFLX="NC_N6X" Status="true">
   <RuleType ID="1">
      <WordRT ID="1" POS="N" Flex="true" />
      <WordRT ID="2" POS="*" Flex="false" Condition="GramCats,2"/>
      <WordRT ID="3" POS="*" Flex="false" Condition="GramCats,2"/>
      <WordRT ID="4" POS="*" Flex="false" Condition="GramCats,2"/>
   </RuleType>
   <RuleType ID="2">
      <WordRT ID="1" POS="N" Flex="true" />
      <WordRT ID="2" POS="PREP" Flex="false" />
      <WordRT ID="3" POS="*" Flex="false" Condition="PrepAgr,2" />
      <WordRT ID="4" POS="*" Flex="false" />
   </RuleType>
   <RulePart ID="1" Frequency="3" Example="princ na belom konju">
      <WordRP ID="1" GramCats="ms1v" />
   </RulePart>
   <RulePart ID="2" Frequency="2"
      <WordRP ID="1" GramCats="ms1q" />
   </RulePart>
   <RulePart ID="3" Frequency="2" >
      <WordRP ID="1" GramCats="ns1q" />
   </RulePart>
   <RulePart ID="4" Frequency="1" >
      <WordRP ID="1" GramCats="fs1q" />
   </RulePart>
   <RulePart ID="5" Frequency="0">
                <WordRP ID="1" GramCats="ns1v" />
   </RulePart>
   <RulePart ID="6" Frequency="0">
      <WordRP ID="1" GramCats="fs1v" />
   </RulePart>
</Rule>
```

The first part of pseudocode relates to step one of the production of a lemma for DELAC dictionary described in part II section A. The second and third part of pseudocode are related to step two of the production of lemma for DELAC dictionary described in part II section A. Examples of the XML structure of RuleType and RulePart of part two of the pseudocode are given in table II.

C. System Evaluation

The first evaluations of the strategy have been performed using the DELAC dictionary, by comparing results from automatic processing with manually created compound lemmas. Figure 2 shows the success statistics: out of 2135 compound lemmas 219 (10.27%) either couldn't be solved, or the offered solution was incorrect. Further analysis showed that the reason for failure in some cases was the absence of some compound components from DELAS dictionary.

Only for 19 (0.89%) compound lemmas the strategy has not been properly defined, while in 4 (0.19%) cases a rule was missing. Totally or "conditionally" correct results were obtained for 1892 (88.67%) compound lemmas, where "conditionally" correct means that the inflective class code was correctly determined, which is good enough for query expansion.

Since the strategy has been designed to produce all possible lemmas by applying all the rules that meet the criteria defined, in some cases several possibilities have been offered and sorted by previously defined rule priority. Figure 2 shows that out of 1892 correct results, as many as 1667 have been offered as the first answer, 137 as second, 53 as third, 33 as fourth and 2 as fifth.

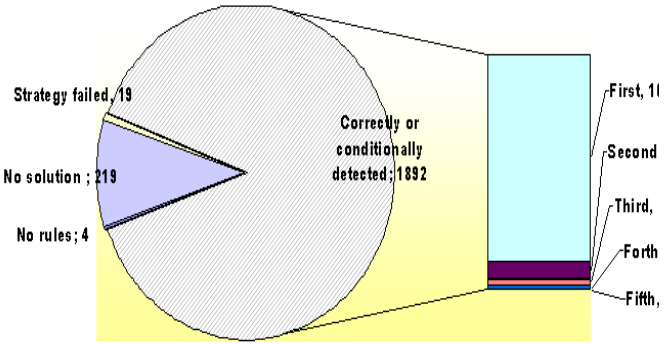


Fig. 2. The Implementation of the Strategy on the test data

The system has been evaluated on three separate sets of data that differ both in content and in structure: compound toponyms, formal names of professions and queries from a search engine. Figure 3 depicts success evaluation for defined strategy. Results have confirmed that the developed strategy can be integrated in morphological query expansion mechanism for compounds and phrases which do not exist in the compounds dictionary.

The evaluation set with queries from search engine was selected from a log file of one of Serbian professional journals that deals with economic issues. The log file used thus gives a good insight in users' queries.

Some of the multi word queries from the log file represent simple lists of key words, for instance *izvoz, uvoz, Beograd, Srbija, 2002* 'import, export, Belgrade, Serbia, 2002'. It is not to be expected that the user would be interested for inflections of such a list as a whole. For many free phrases, especially those with fewer components, the structure was correctly

detected and their inflected forms produced, e.g. *udio izvoza u domaćem proizvodu* 'export quota in domestic product'. As a by-product, the analysis of the log file detected some compounds that were not yet in the dictionary of compounds and which were subsequently added to it (the most frequent one being *kursna lista* 'the exchange rate list'). In order to be able to correctly inflect more free phrases some new inflectional transducers had been created.

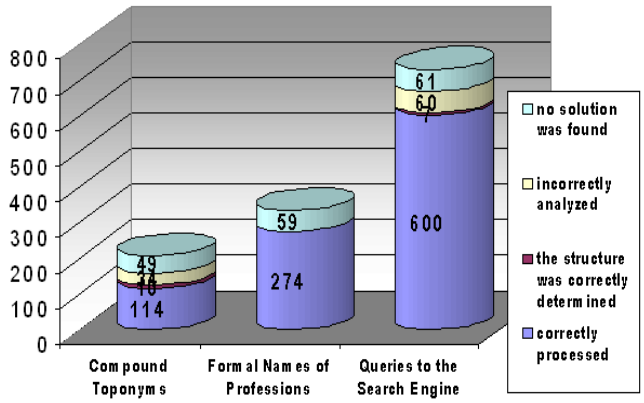


Fig. 3. The implementation of the Strategy on the evaluation data

III. Query Improvement

A. WS4LR

WS4LR handles simultaneously several types of resources, one of them being the system of morphological dictionaries of Serbian simple words and compounds in LADL format. Morphological dictionaries in the same format exist for many other languages, including French, English, Greek, Portuguese, Russian, Thai, Korean, Italian, Spanish, Norwegian, Arabic, German, Polish and Bulgarian.

The system enables concurrent manipulation of a set of dictionaries of lemmas, simple words (DELAS) or compounds (DELAC), distributed in several files. Working with dictionaries of word forms (DELAF, DELACF) type files is not directly supported since this type of files should in general be produced automatically from DELAS and DELAC by applying the appropriate transducers. The organization of dictionaries in separate files is important from the practical point of view since smaller files are easier to manipulate.

An important feature of this system is the ability of retrieving efficiently a subset of lemmas by matching the lemmas, their part of speech (PoS), inflectional class code, syntactic and semantic markers or their Boolean combination. For instance, one can look for all the dictionary entries starting or ending with a search string.

Another important resource handled by WS4LR is the Serbian Wordnet [8]. A Wordnet is composed of synsets, or sets of synonymous words representing a concept, with basic semantic relations between them forming a semantic network. Each synset word or "literal" is denoted by a "literal string" followed by a "sense tag" which represents the specific sense of the literal string in that synset, while interlingual index (ILI) enables the connection of the same concepts in different languages, a feature that can be used, among others, for cross-language information retrieval.

For expansion of queries with proper names WS4LR is using Prolex, a multilingual database of proper names which represents the implementation of an elaborate four-layered ontology of proper names [9] organized around a conceptual proper name that represents the same concept in different languages is used.

WS4LR also handles aligned texts. A pair of semantically equivalent texts in different languages, such as an original text and its translation, that are aligned on a structural level (paragraph, sentence, phrase, etc.) is known as an aligned text or bitext. The standard format for representing aligned texts is the Translation Memory eXchange format (TMX) that is XML-compliant [10].

WS4LR, written in C#, is organized in modules which perform different functions. A Component diagram (Fig. 4.) illustrates the pieces of software that make up the WS4LR system. The diagram on figure 4 demonstrates some components and their inter-relationships. The core of the system *WS4LR_Core* comprises four .Net libraries: *CommonRes.dll*, *NlpQuery.dll, VisualTMX.dll* and *WNDictAuto.dll*. A dependency relationship maps *NlpQuery.dll* to the handled lexical resources.

WS4LR_Core is used by two components: the stand-alone windows application *WS4LR.exe* and the web service *wsQueryExpand.asm*. Web application *WS4QE.asp* manages user query request, than uses web service in order to expand user query, submits the expanded query to Google search engine and finally presents retrieved result.

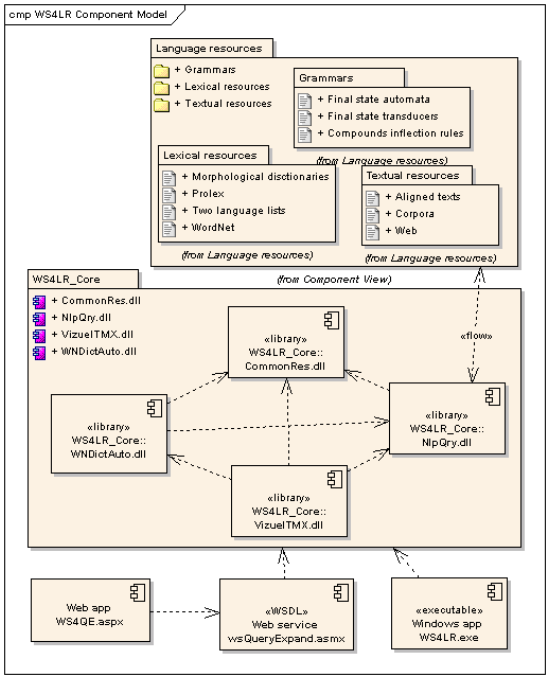


Fig. 4. The components that make up the WS4LR system and their inter-relationships

B. Usage of Various Lexical Resources and Tools for Improvement of Queries

Selection of words chosen for a query, which are of paramount importance for the quality of results obtained by the query, can be substantially improved by using various lexical resources. Morphological dictionaries enable morphological expansion of the query, very important in highly inflective languages, such as Serbian. Wordnets and Prolex support semantic and multilingual expansion of the query.

The WS4LR system for query expansion allows the user to decide how his query will be expanded by choosing one or several of the offered options:

1. Alternate alphabet usage – for instance, the user can submit a keyword in Latin alphabet: *lekar opšte prakse* 'general practitioner' which will be expanded automatically by adding the keyword in Cyrillic: *лекар опште праксе*.

2. The inclusion of inflectional forms, for instance, *lekara opšte prakse, lekaru opšte prakse, lekarima opšte prakse, etc* with support of morphological dictionaries, inflectional transducers and the rule based procedure for Serbian.

3. The addition of synonyms – for instance, the synonym *lekar opšte medicine* 'GP' can be added to the keyword *lekar opšte prakse*. Synonyms are added on basis of the Serbian Wordnet (SWN). All other relations included in SWN can also be used for query expansion, for instance related hyponyms: *porodični lekar, kućni lekar, seoski lekar* 'family doctor, country doctor'.

4. The expansion of proper names using Prolex which offers to the user the option of adding proper name aliases, its synonyms, but also other proper names which are semantically related to the initial proper name through holonym and meronym relations. Thus a query with the word Meksiko 'Mexico' can be expanded with derivation *Meksikanac* '*Mexican man'*, *Meksikanka*, 'Mexican woman' but also with meronymes *Mexico-City* and *Puebla*.

5. The inflection of free phrases by predicting their syntactic structure. Presumption is that many free phrases used for search will have the same syntactic structure as a compound, and that the inflectional transducers for compounds that have already been developed can be applied to inflect them correctly. This type of expansion is implemented with the rule based system described in the second section of this paper. An example is the phrase *prosečna plata u Srbiji* 'average salary in Serbia' which, according to the dictionaries can be analyzed as a phrase of the form adjective+noun followed by any two words. In this particular case the rule 47 for NC_AXN4X is applied for query expansion.

6. The bilingual search – for instance, to the keyword *lekar opšte prakse* and its Serbian synonym keyword *lekar opšte medicine* a corresponding English set of synonyms can be added: {general practitioner, GP}. The bilingual search is, however, done separately and the results are presented in two columns.

C. WS4QE

The developed web application receives the user query, and subsequently uses the local web service WS4QE to expand the query and forward it to the Google search engine using the Google AJAX Search API. Google AJAX Search API is a Java script library which enables the embedding of Google searches into personal web pages or web applications. This library is composed of simple web objects which perform "inline" search using numerous Google services (Web Search, Local Search, Video Search, Blog Search, News Search and Book Search).

The web service returns the required information in XML form, which is being received and converted to appropriate application structures (string, array, table, etc.). Some of the typical calls are: *getObliciLeme(lema)*, which retrieves all inflective forms of a lemma, *getSinonimiWN_WithFlex(lema)* which retrieves all wordnet synonyms with inflective forms, *getSinonimiWN_NoFlex(lema)* which retrieves all wordnet synonyms without inflective forms, *getProlexTable(rec, jezikSearch, Inflect, ExpandWith)* which retrieves all chosen proper name expansions according to the request specified by the user.

WS4QE also offers functions for aligned text manipulation and search with expanded queries, but some of WS4QE features related to query expansion will be illustrated in web search.

Query expansion is implemented with different possibilities and levels of detail, so the web user can choose from several options (from simple query expansion to complex wordnet advanced search). Figure 5 shows the page with the keyword *lekar opšte prakse* chosen as the initial search string. As semantic expansion was chosen, the appropriate synset was retrieved and synonym the *lekar opšte medicine* appeared in the list of words that can be used for composing the query. In this case morphological expansion was selected, and the query is further expanded only by including both chosen words in all inflected forms.

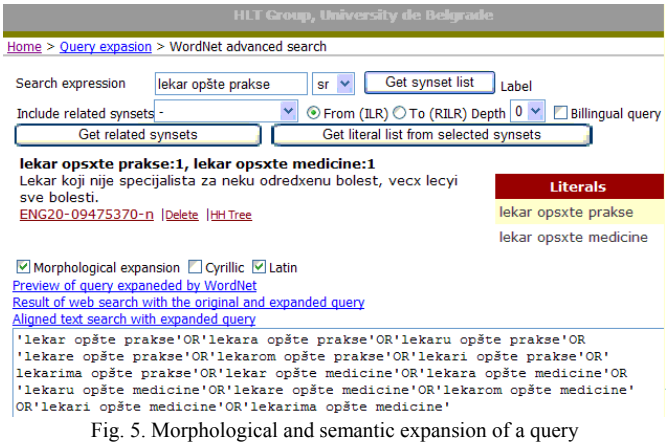


Fig. 5. Morphological and semantic expansion of a query

The query, now composed of two Latin and two Cyrillic strings was then submitted by WS4QE to Google and, as a result, documents with different forms of both synonymous compounds were obtained. A thorough inspection of all

documents was not performed, for obvious reasons, but it is safe to say that it is most unlikely that any of the documents obtained is irrelevant because all words used are specific in that they are neither homonymous nor polysemous. Part of the results of the expanded query is depicted in Figure 6.

For illustration of recall purposes, three query expansions where performed using the word *političko opredeljenje 'political preference'* and all results were compared. First query expansion included semantic expansion with synonym *ideologija 'ideology'*. The expanded query *"ideologija "OR" političko opredeljenje"* was then submitted by WS4QE to Google and, as a result, a total of 245,000 documents were obtained. The same query submitted directly to Google with only the initial string *političko opredeljenje* returned a total of 24,700. Thus the expanded query, without the morphological expansion, obtained almost ten times more documents. In the second case, semantic expansion remained and the query was improved additionally by including all words in Cyrillic alphabet. The result of the expanded query was a total of 320,000 documents. The expanded query once again remarkably increased the number of documents obtained. The third query was performed with morphological and semantic expansion, but the extension to Cyrillic alphabet was omitted. As a result 609,000 documents were obtained, which means that the recall has been extremely improved. Thus it can be concluded that a considerable increase of recall was obtained in all three examples.

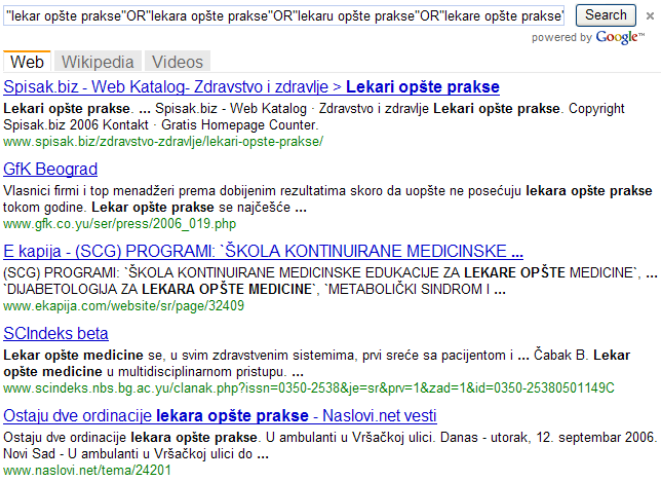


Fig. 6. Results for expanded query for *lekar opšte prakse*

On the other side, speaking of precision, unexpanded query with compounds and phrases can obtain unrelated results. For example document with "*... srpske politike kroz istoriju **političkog** ekumenizma,... u kontekstu istorijskog **opredeljenja** za etiku, etičnost i karakternost, ...*" is obtained, but is not relevant, because the adjective *političkog* is related to the noun *ekumenizma* instead of *opredeljenja*.

References

[1] Krstev, C., Stanković, R., Vitas, D., Obradović, I. (2006). "WS4LR: A Workstation for Lexical Resources". *In Proceedings of the 5th International Conference on Language Resources and Evaluation, LREC 2006*, Genoa, Italy, May 2006, pp. 1692-1697.

[2] Gelbukh, A., Sidorov G. "Approach to construction of automatic morphological analysis systems for inflective languages with little effort". *LNCS 2588*, 2003, pp. 215-220.

[3] Courtois, B., Silberztein, M. (eds.): *Dictionnaires électroniques du français. Langue française*. 87, Larousse, Paris, 1990.

[4] Krstev C.: *Processing of Serbian – Automata, Texts and Electronic Dictionarie*. Faculty of Philology, University of Belgrade, Belgrade, 2008.

[5] Savary, A., Krstev, C., Vitas, D.: "Inflectional non compositionality and variation of compounds in French, Polish and Serbian, and their automatic processing". *Bulag - Bulletin de Linguistique Appliquée et Générale*. 32, 73-94, 2007.

[6] Krstev, C., Vitas, D., Savary, A.: "Prerequisites for a Comprehensive Dictionary of Serbian Compounds". *In: Salakosi, T., Ginter, F., Pyysalo, S., Pahikkala, T. (eds.) FinTAL 2006. LNAI*, vol. 4139, pp. 552--564. Springer, Heidelberg, 2006.

[7] Krstev, C. Stanković, R., Vitas, D., Obradović, I..: "The Usage of Various Lexical Resources and Tools to Improve the Performance of Web Search Engines". *In: 6th LREC International Conference on Language Resources and Evaluation*, Marrakech, Marocco, 2008.

[8] Krstev C., Pavlović-Lažetić G., Vitas D., Obradović I.: "Using Textual and Lexical Resources in Developing Serbian Wordnet." *In Romanian Journal of Information Science and Technology*, Romanian Academy, Publishing House of the Romanian Academy, vol. 7, No. 1-2, pp. 147-161, (2004).

[9] Krstev, C., Vitas, D., Maurel, D., Tran, M. (2005). "Multilingual Ontology of Proper Names". *In Proc. of Second Language & Technology Conference*, Poznań, Poland, April 21-23, Wydawnictwo Poznańskie Sp. z o.o, Poznań.

[10] TMX 1.4b specification, http://www.lisa.org/standards/tmx/tmx.html

Web-based Bengali News Corpus for Lexicon Development and POS Tagging

Asif Ekbal and Sivaji Bandyopadhyay

***Abstract*—Lexicon development and Part of Speech (POS) tagging are very important for almost all Natural Language Processing (NLP) applications. The rapid development of these resources and tools using machine learning techniques for less computerized languages requires appropriately tagged corpus. We have used a Bengali news corpus, developed from the web archive of a widely read Bengali newspaper. The corpus contains approximately 34 million wordforms. This corpus is used for lexicon development without employing extensive knowledge of the language. We have developed the POS taggers using Hidden Markov Model (HMM) and Support Vector Machine (SVM). The lexicon contains around 128 thousand entries and a manual check yields the accuracy of 79.6%. Initially, the POS taggers have been developed for Bengali and shown the accuracies of 85.56%, and 91.23% for HMM, and SVM, respectively. Based on the Bengali news corpus, we identify various word-level orthographic features to use in the POS taggers. The lexicon and a Named Entity Recognition (NER) system, developed using this corpus, are also used in POS tagging. The POS taggers are then evaluated with Hindi and Telugu data. Evaluation results demonstrates the fact that SVM performs better than HMM for all the three Indian languages.**

***Index Terms*—Web based corpus, lexicon, part of speech (POS) tagging, hidden Markov model(HMM), support vector machine (SVM), Bengali, Hindi, Telugu.**

I. Introduction

The mode of language technology work has changed dramatically since the last few years with the web being used as a data source in wide range of research activities. The web is anarchic, and its use is not in the familiar territory of computational linguistics. The web walked in to the ACL meetings started in 1999. The use of the web as a corpus for teaching and research on language has been proposed a number of times [1], [2], [3], [4]. There has been a special issue of the Computational Linguistics journal on Web as Corpus [5]. Several studies have used different methods to mine web data.

There is a long history of creating a standard for western language resources, such as EAGLES [1], PROLE/SIMPLE [6], ISLE/MILE [7], [8]. On the other hand, instead of having great linguistic and cultural diversities, Asian language resources have received much less attention than their western counterparts. An initiative [9] has started to create a common standard for Asian language resources.

Manuscript received May 4, 2008. Manuscript accepted for publication June 12, 2008.

Authors are with the Department of Computer Science and Engineering, Jadavpur University, Kolkata, India 700032, e-mail: asif.ekbal@gmail.com, sivaji_cse_ju@yahoo.com.

[1]http://www.ilc.cnr.it/Eagles96/home.html

Part of Speech (POS) tagging is the task of labeling each word in a sentence with its appropriate syntactic category called part of speech. Part of speech tagging is a very important preprocessing task for language processing activities. This helps in doing deep parsing of text and in developing Information extraction systems, semantic processing etc. Part of speech tagging for natural language texts are developed using linguistic rules, stochastic models and a combination of both. Stochastic models [10] [11] [12] have been widely used in POS tagging task for simplicity and language independence of the models. Among stochastic models, Hidden Markov Models (HMMs) are quite popular. Development of a stochastic tagger requires large amount of annotated corpus. Stochastic taggers with more than 95% word-level accuracy have been developed for English, German and other European languages, for which large labeled data are available. The problem is difficult for Indian languages (ILs) due to the lack of such annotated large corpus.

Simple HMMs do not work well when small amount of labeled data are used to estimate the model parameters. Incorporating diverse features in an HMM-based tagger is also difficult and complicates the smoothing typically used in such taggers. In contrast, a Maximum Entropy (ME) based method [13] or a Conditional Random (CRF) Field based method [14] or a SVM based system [15] can deal with diverse and overlapping features of the Indian languages. A POS tagger has been proposed in [16] for Hindi, which uses an annotated corpus of 15,562 words collected from the BBC news site, exhaustive morphological analysis backed by high coverage lexicon and a decision tree based learning algorithm (CN2). The accuracy was 93.45% for Hindi with a tagset of 23 POS tags.

International Institute of Information Technology (IIIT), Hyderabad, India initiated a POS tagging contest, NLPAI ML[2] for the Indian languages in 2006. Several teams came up with various approaches and the highest accuracies were 82.22% for Hindi, 84.34% for Bengali and 81.59% for Telugu. As part of the SPSAL Workshop[3] in IJCAI-07, a competition on POS tagging and chunking for south Asian languages was conducted by IIIT, Hyderabad. The best accuracies reported were 78.66% for Hindi [17], 77.61% for Bengali [18] and 77.37% for Telugu [17]. Other works for POS tagging in Bengali can be found in [19] with a ME approach and in [20] with a CRF approach.

Newspaper is a huge source of readily available documents.

[2]http://ltrc.iiitnet/nlpai_contest06/proceedings.php

[3]http://shiva.iiit.ac.in/SPSAL2007/SPSAL-Proceedings.pdf

In the present work, we have used the corpus that has been developed from the web archive of a very well known and widely read Bengali newspaper. Bengali is the seventh popular language in the world, second in India and the national language in Bangladesh. Various types of news (International, National, State, Sports, Business etc.) are collected in the corpus and so a variety of linguistics features of Bengali are covered. We have developed a lexicon in an unsupervised way using this news corpus without using extensive knowledge of the language. We have developed POS taggers using HMM and SVM. The news corpus has been used to identify several orthographic word-level features to be used in POS tagging, particularly in the SVM model. We have used the lexicon and a NER system [21] as the features in the SVM-based POS tagger. These are also used as the means to handle the unknown words in order to improve the performance in both the models.

The paper is organized as follows. Section II briefly reports about the Bengali news corpus generation from the web. Section III discusses about the use of language resources particularly in lexicon development. Section IV describes the POS tagset used in the present work. Section V reports the development of POS tagger using HMM. Section VI deals with the development of POS tagger using SVM. Unknown word handling techniques are described in Section VII. Evaluation results of the POS tagger for Bengali, Hindi and Telugu are reported in Section VIII. Finally, Section IX concludes the paper.

II. Development of the Tagged Bengali News Corpus from the Web

The development of the Bengali news corpus is a sequence of language resource acquisition using a web crawler, language resource creation that includes HTML file cleaning, code conversion and language resource annotation that involves defining a tagset and subsequent tagging of the news corpus.

A web crawler has been developed for acquisition of language resources from the web archive of a leading Bengali newspaper. The web crawler retrieves the web pages in Hyper Text Markup Language (HTML) format from the news archive of a leading Bengali news paper within a range of dates provided as input. The news documents in the archive are stored in a particular fashion. The user has to give the range of dates as starting yy-mm-dd and ending yy-mm-dd format. The crawler generates the Universal Resource Locator (URL) address for the index (first) page of any particular date. The index page contains actual news page links and links to some other pages (e.g., Advertisement, TV schedule, Tender, Comics and Weather etc.) that do not contribute to the corpus generation. The HTML files that contain news documents are identified and the rest of the HTML files are not considered further.

The HTML files that contain news documents are identified by the web crawler and require cleaning to extract the Bengali text to be stored in the corpus along with relevant details. An HTML file consists of a set of tagged data that includes Bengali and English texts. The HTML file is scanned from the beginning to look for tags like <fontFACE = "Bengali Font Name"> ... , where the "Bengali Font Name" is the name of one of the Bengali font faces as defined in the news archive. The Bengali texts in the archive are written in dynamic fonts and the Bengali pages are generated on the screen on the fly, i.e., only when the system is online and connected to the web. Moreover, the newspaper archive uses graphemic coding whereas orthographic coding is required for text processing tasks. Hence, Bengali texts, written in dynamic fonts are not suitable for text processing activities. In graphemic coding, a word is coded according to the constituent graphemes. But in orthographic coding the word is coded according to the constituent characters. In graphemic coding conjuncts have separate codes. But in orthographic coding it is coded in terms of the constituent consonants. A code conversion routine has been written to convert the dynamic codes used in the HTML files to represent Bengali text to ISCII codes. A separate code conversion routine has been developed for converting ISCII codes to UTF-8 codes.

TABLE I
News Corpus Tagset

Tag	Definition	Tag	Definition
header	Header of the news document	reporter	Reporter name
title	Headline of the news document	agency	Agency providing news
t1	1st headline of the title	location	The news location
t2	2nd headline of the title	body	Body of the news document
date	Date of the news document	p	Paragraph
bd	Bengali date	table	Information in tabular form
day	Day	tc	Table Column
ed	English date	tr	Table row

The Bengali news corpus developed from the web is annotated using a tagset that includes the *type* and *subtype* of the news, title, date, reporter or agency name, news location and the body of the news. A news corpus, whether in Bengali or in any other language has different parts like title, date, reporter, location, body etc. A news document is stored in the corpus in XML format using the tagset, mentioned in Table I. The *type* and *subtype* of the news item are stored as attributes of the *header*. The news items have been classified on geographic domain (International, National, State, District, Metro) as well as on topic domain (Politics, Sports, Business).

The news corpus contains 108,305 number of news documents with about five years (2001-2005) of news data collection. Some statistics about the tagged news corpus are presented in Table II. Details of corpus development are reported in [22].

III. Lexicon Development from the Corpus

An unsupervised machine learning method has been used for lexicon development from the Bengali news corpus. No extensive knowledge about the language is required except the knowledge of the different inflections that can appear with the different words in Bengali.

In Bengali, there are five different POS namely, noun, pronoun, verb, adjective, and indeclinable (postpositions, conjunctions, and interjections). Noun, verb and adjective belong

TABLE II
CORPUS STATISTICS

Total no. of news documents in the corpus	108,305
Total no. of sentences in the corpus	2,822,737
Average no. of sentences in a document	27
Total no. of wordforms in the corpus	33,836,736
Average no. of wordforms in a document	313
Total no. of distinct wordforms in the corpus	467,858

TABLE III
LEXICON STATISTICS

Iteration	1	2	3	4	5
News Documents	9737	19929	39924	69951	99651
Sentences	0.22	0.49	1.02	1.79	2.55
Wordforms	2.77	5.98	12.53	21.53	30.61
Distinct Wordforms	0.10	0.15	0.23	0.37	0.526
Root words	0.03	0.04	0.065	0.09	0.128

to the open class of POS in Bengali. Initially, all the words (inflected and uninflected) are extracted from the corpus and added to a database. A list of inflections that may appear with noun words is kept and it has 27 entries. In Bengali, verbs can be categorized into 20 different groups according to their spelling patterns and the different inflections that can be attached to them. Original wordform of a verb word often changes when any suffix is attached to it. At present, there are 214 different entries in the verb inflection list. Noun and verb words are tagged by looking at their inflections. Some inflections may be common to both nouns and verbs. In these cases, more than one root word will be generated for a wordform. The POS ambiguity is resolved by checking the number of occurrences of these possible root words along with the POS tags as derived from other wordforms. Pronoun and indeclinable are basically closed class of POS in Bengali and these are added to the lexicon manually. It has been observed that adjectives in Bengali generally occur in four different forms based on the suffixes attached. The first type of adjectives can form comparative and superlative degree by attaching the suffixes *-tara* and *-tamo* to the adjective word. These adjective stems are stored in the lexicon with adjective POS. The second set of suffixes (e.g. *-gato, -karo* etc.) identifies the POS of the wordform as adjective if only there is a noun entry of the desuffixed word in the lexicon. The third group of suffixes (e.g. *-janok, -sulav* etc.) identifies the POS of the wordform as adjective and the desuffixed word is included in the lexicon with noun POS. The last set of suffixes identifies the POS of the wordform as adjective.

The system retrieves the words from the corpus and creates a database of distinct wordforms. Each distinct wordform in the database is checked for pronoun and indeclinable. If the wordform is neither a pronoun nor an indeclinable, it is analyzed to identify the possible root word along with the POS tag obtained from inflection analysis. Different suffixes are compared with the end of a word. If any match is found then the remaining part of that word from the beginning is stored as a candidate root word for that inflected word along with the appropriate POS information. So, one or more [root word, POS] pairs are obtained after suffix analysis of a wordform. It may happen that wordform itself is a root word, so the [wordform, {all possible POS}] is also added to the previous candidate root word list. Two intermediate databases have been kept. A wordform along with the candidate [root word, POS] pairs is stored in one database. The other database keeps track of the distinct candidate [root word, POS] pairs along with its frequency of occurrence over the entire corpus. After suffix analysis of all distinct wordforms, the [root word, POS] pair that has highest frequency of occurrence over the entire corpus is selected from the candidate [root word, POS] pairs for the wordform. If the frequency of occurrences for two or more [root word, POS] pairs are same, the root word with the maximum number of characters is chosen as the possible root.

The corpus has been used in the unsupervised lexicon development. Table III shows the results using the corpus. Except news documents, the number of sentences, wordforms, distinct wordforms and root words are mentioned in millions. The lexicon has been checked manually for correctness and it has been observed that the accuracy is approximately 79.6%. The list of rootwords are automatically corrected to a large degree by using the named entity recognizer for Bengali [21] to identify the named entities in the corpus in order to exclude them from the lexicon. The number of root words increases as more and more news documents are considered in the lexicon development.

IV. POS TAGSET USED IN THE WORK

We have used a POS tagset of 26 POS tags, defined for the Indian languages. All the tags used in this tagset (IIIT, Hyderabad, India tag set) are broadly classified into three categories. The first category contains 10 tags that have been adopted with minor changes from the Penn tagset. The second category that contains 8 tags is a modification of similar tags in the Penn tagset. They have been designed to cater to some phenomena that are specific to Indian languages. The third category consists of 8 tags and has been designed exclusively for Indian languages.

- Group 1: NN-Noun, NNP-Proper noun, PRP-Pronoun, VAUX-Verb auxillary, JJ-Adjective, RB-Adverb, RP-Particle, CC-Conjunction, UH-Interjection, SYM-Special symbol.
- Group 2: PREP-Postposition, QF-Quantifiers, QFNUM-Quantifiers number, VFM-Verb finite main, VJJ-Verb non-finite adjectival, VRB-Verb non-finite adverbial, VNN-Verb non-finite nominal, QW-Question words.
- Group 3: NLOC-Noun location, INTF-Intensifier, NEG-Negative, NNC-Compound nouns, NNPC-Compound proper nouns, NVB-Noun in kriyamula, JVB-Adjective in kriyamula, RBVB-Adverb in kriyamula.

V. POS TAGGING USING HIDDEN MARKOV MODEL

A POS tagger based on Hidden Markov Model (HMM) [23] assigns the best sequence of tags to an entire sentence. Generally, the most probable tag sequence is assigned to each sentence following the Viterbi algorithm [24]. The task of POS tagging is to find the sequence of POS tags $T = t_1, t_2, t_3, \ldots t_n$ that is optimal for a word sequence $W =$

$w_1, w_2, w_3 \ldots w_n$. The tagging problem becomes equivalent to searching for $argmax_T P(T) * P(W|T)$, by the application of Bayes' law.

We have used trigram model, i.e., the probability of a tag depends on two previous tags, and then we have,
$P(T) = P(t_1|\$) \times P(t_2|\$, t_1) \times P(t_3|t_1, t_2) \times P(t_4|t_2, t_3) \times \ldots \times P(t_n|t_{n-2}, t - n - 1)$, where, an additional tag '$' (dummy tag) has been introduced to represent the beginning of a sentence.

Due to sparse data problem, the linear interpolation method has been used to smooth the trigram probabilities as follows: $P'(t_n|t_{n-2}, t_{n-1}) = \lambda_1 P(t_n) + \lambda_2 P(t_n|t_{n-1}) + \lambda_3 P(t_n|t_{n-2}, t_{n-1})$ such that the λs sum to 1.
The values of λs have been calculated by the method given in [12].

To make the Markov model more powerful, ***additional context dependent features*** have been introduced to the emission probability in this work that specifies the probability of the current word depends on the tag of the previous word and the tag to be assigned to the current word. Now, we calculate $P(W|T)$ by the following equation:

$$P(W|T) \approx P(w_1|\$, t_1) \times P(w_2|t_1, t_2) \times \ldots \times P(w_n|t_{n-1}, t_n)$$

So, the emission probability can be calculated as

$$P(w_i|t_{i-1}, t_i) = \frac{freq(t_{i-1}, t_i, w_i)}{freq(t_{i-1}, t_i)}$$

Here also the smoothing technique is applied rather than using the emission probability directly. The emission probability is calculated as:
$P'(w_i|t_{i-1}, t_i) = \theta_1 P(w_i|t_i) + \theta_2 P(w_i|t_{i-1}, t_i)$, where θ_1, θ_2 are two constants such that all θs sum to 1.

The values of θs should be different for different words. But the calculation of θs for every word takes a considerable time and hence θs are calculated for the entire training corpus. In general, the values of θs can be calculated by the same method that was adopted in calculating λs.

VI. POS Tagging using Support Vector Machine

We have developed a POS tagger using Support Vector Machine (SVM). We identify the features from the news corpus to use in the SVM model. Performance of the POS tagger is improved significantly by adopting the various techniques for handling the unknown words. These include word suffixes, identified by observing the various wordforms of the Bengali news corpus. We have also used the lexicon and a NER system [21], developed with the help of news corpus.

A. Support Vector Machine

Support Vector Machines (SVMs), first introduced by Vapnik [25] [26], are relatively new machine learning approaches for solving two-class pattern recognition problems. SVMs are well-known for their good generalization performance, and have been applied to many pattern recognition problems. In the field of Natural Language Processing(NLP), SVMs are applied to text categorization, and are reported to have achieved high accuracy without falling into over-fitting even though with a large number of words taken as the features [27] [28]. Suppose, we have a set of training data for a two-class problem: $\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\}$, where $\mathbf{x}_i \in R^D$ is a feature vector of the i-th sample in the training data and $y \in \{+1, -1\}$ is the class to which $\mathbf{x}_i$ belongs. In their basic form, a SVM learns a linear hyperplane that separates the set of positive examples from the set of negative examples with *maximal margin* (the margin is defined as the distance of the hyperplane to the nearest of the positive and negative examples). In basic SVM framework, we try to separate the positive and negative examples by the hyperplane written as:

$$(\mathbf{w}.\mathbf{x}) + b = 0 \qquad \mathbf{w} \in \mathbf{R}^n, b \in \mathbf{R}.$$

SVMs find the "optimal" hyperplane (optimal parameter $\overline{w}, b$) which separates the training data into two classes precisely. The linear separator is defined by two elements: a weight

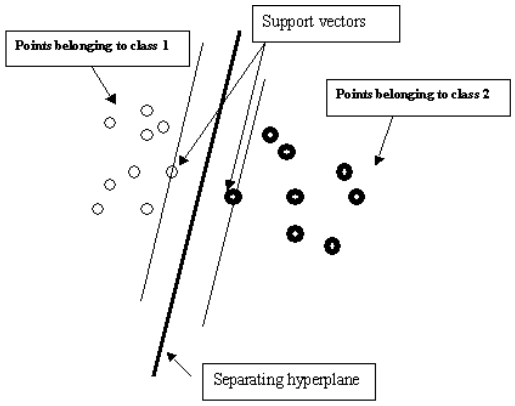


Fig. 1. Example of a 2-dimensional SVM

vector $\mathbf{w}$ (with one component for each feature), and a bias b which stands for the distance of the hyperplane to the origin. The classification rule of a SVM is:

$$sgn(f(\mathbf{x}, \mathbf{w}, b)) \tag{1}$$

$$f(\mathbf{x}, \mathbf{w}, b) = < \mathbf{w}.\mathbf{x} > + b \tag{2}$$

being $\mathbf{x}$ the example to be classified. In the linearly separable case, learning the maximal margin hyperplane $(\mathbf{w}, b)$ can be stated as a convex quadratic optimization problem with a unique solution: *minimize* $||\mathbf{w}||$, *subject to the constraints* (one for each training example):

$$y_i(< \mathbf{w}.x_i > + b) \geq 1 \tag{3}$$

See an example of a 2-dimensional SVM in Figure 1.

The SVM model has an equivalent dual formulation, characterized by a weight vector α and a bias b. In this case, α contains one weight for each training vector, indicating the importance of this vector in the solution. Vectors with non null weights are called support vectors. The dual classification rule is:

$$f(\mathbf{x}, \alpha, b) = \sum_{i=1}^{N} y_i \alpha_i < \mathbf{x}_i.\mathbf{x} > + b \tag{4}$$

The α vector can be calculated also as a quadratic optimization problem. Given the optimal α^* vector of the dual quadratic optimization problem, the weight vector $\mathbf{w}^*$ that realizes the maximal margin hyperplane is calculated as:

$$\mathbf{w}^* = \sum_{i=1}^{N} y_i \alpha_i^* \mathbf{x}_i \quad (5)$$

The b^* has also a simple expression in terms of $\mathbf{w}^*$ and the training examples $(\mathbf{x}_i, y_i)_{i=1}^N$.

The advantage of the dual formulation is that efficient learning of non-linear SVM separators, by introducing *kernel functions*. Technically, a *kernel function* calculates a dot product between two vectors that have been (non linearly) mapped into a high dimensional feature space. Since there is no need to perform this mapping explicitly, the training is still feasible although the dimension of the real feature space can be very high or even infinite.

By simply substituting every dot product of $\mathbf{x}_i$ and $\mathbf{x}_j$ in dual form with any *kernel function* $K(\mathbf{x}_i, \mathbf{x}_j)$, SVMs can handle non-linear hypotheses. Among the many kinds of *kernel functions* available, we will focus on the d-th polynomial kernel:

$$K(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i.\mathbf{x}_j + 1)^d$$

Use of d-th polynomial kernel function allows us to build an optimal separating hyperplane which takes into account all combination of features up to d.

The SVMs have advantage over conventional statistical learning algorithms, such as Decision Tree, Hidden Markov Models, Maximum Entropy Models from the following two aspects:

1) SVMs have high generalization performance independent of dimension of feature vectors. Conventional algorithms require careful feature selection, which is usually optimized heuristically, to avoid overfitting. So, it can more effectively handle the diverse, overlapping and morphologically complex Indian languages.
2) SVMs can carry out their learning with all combinations of given features without increasing computational complexity by introducing the *Kernel function*. Conventional algorithms cannot handle these combinations efficiently, thus, we usually select "important" combinations heuristically with taking the trade-off between accuracy and computational complexity into consideration.

We have developed our system using SVM [27] [25], which perform classification by constructing an N-dimensional hyperplane that optimally separates data into two categories. Our general POS tagging system includes two main phases: training and classification. The training process was carried out by YamCha[4] toolkit, an SVM based tool for detecting classes in documents and formulating the POS tagging task as a sequential labeling problem. We have used TinySVM-0.07 [5] classifier that seems to be the best optimized among publicly available SVM toolkits. Here, the pairwise multi-class decision method and *second degree polynomial kernel function* have been used. In pairwise classification, we constructed K(K-1)/2 classifiers (here, K=26, no. of POS tags) considering all pairs of classes, and the final decision is given by their weighted voting.

[4]http://chasen-org/ taku/software/yamcha/

[5]http://cl.aist-nara.ac.jp/ taku-ku/software/TinySVM

B. Features for POS Tagging

Following are the details of the set of features that have been applied for POS tagging in Bengali.

• Context word feature: Preceding and following words of a particular word are used as features.

• Word suffix:Word suffix information is helpful to identify POS class. One way to use this feature is to consider a fixed length (say, *n*) word suffix of the current and/or the surrounding word(s). If the length of the corresponding word is less than or equal to *n-1* then the feature values are not defined and denoted by ND. The feature value is also not defined (ND) if the token itself is a punctuation symbol or contains any special symbol or digit. The second and the more helpful approach is to modify the feature as binary valued. Variable length suffixes of a word can be matched with the predefined lists of useful suffixes for different classes. This second type of suffixes include the noun, verb and adjective inflections. We have used both type of suffixes as the features.

• Word prefix: Prefix information of a word is also helpful. A fixed length (say, *n*) prefix of the current and/or the surrounding word(s) can be considered as features. This feature value is not defined (ND) if the length of the corresponding word is less than or equal to *n-1* or the word is a punctuation symbol or the word contains any special symbol or digit.

• Part of Speech (POS) Information: POS information of the previous word(s) might be used as a feature. This is the only dynamic feature in the experiment.

• Named Entity Information: The named entity (NE) information of the current and/or the surrounding word(s) plays an important role in the overall accuracy of the POS tagger. In order to use this feature, a CRF-based NER system [21] has been used. The NER system uses the NE classes namely, *Person name*, *Location name*, *Organization name* and *Miscellaneous name*. Date, time, percentages, numbers and monetary expressions belong to the *Miscellaneous name* category. The NER system was developed using a portion of the Bengali news corpus. This NER system has demonstrated 90.7% f-score value during 10-fold cross validation test with a training corpus of 150K wordforms.

The NE information can be used in two different ways. The first one is to use the NE tag(s) of the current and/or the surrounding word(s) as the features of SVM. The second way is to use this NE information at the time of testing. In order to do this, the test set is passed through the NER system. Outputs of the NER system are given more priorities than the outputs of the POS tagger for the unknown words in the test set. The NE tags are then replaced appropriately by the POS tags (NNPC: Compound proper noun, NNP: Proper noun and QFNUM: Quantifier number).

• Lexicon Feature: The lexicon has been used to improve the performance of the POS tagger. One way is to use this lexicon as the features of the SVM model. To apply this, five different features are defined for the open class of words as follows:

1) If the current word is found to appear in the lexicon with the 'noun' POS, then the feature 'Lexicon' is set to 1.
2) If the current word is found to appear in the lexicon with the 'verb' POS, then the feature 'Lexicon' is set to 2.
3) If the current word is found to appear in the lexicon with the 'adjective' POS, then the feature 'Lexicon' is set to 3.
4) If the current word is found to appear in the lexicon with the 'pronoun' POS, then the feature 'Lexicon' is set to 4.
5) If the current word is found to appear in the lexicon with the 'indeclinable' POS, then the feature 'Lexicon' is set to 5.

The second or the alternative way is to use this lexicon during testing. For an unknown word, the POS information extracted from the lexicon is given more priority than the POS information assigned to that word by the SVM model. An appropriate mapping has been defined from these five basic POS tags to the 26 POS tags. This is also used for handling the unknown words in the HMM model.

•Made up of digits: For a token if all the characters are digits then the feature "Digit" is set to 1; otherwise, it is set to 0. It helps to identify QFNUM (Quantifier number) tag.

•Contains symbol: If the current token contains special symbol (e.g., %, $ etc.) then the feature "ContainsSymbol" is set to 1; otherwise, it is set to 0. This helps to recognize SYM (Symbols) and QFNUM (Quantifier number) tags.

•Length of a word: Length of a word might be used as an effective feature of POS tagging. If the length of the current token is more than three then the feature 'LengthWord' is set to 1; otherwise, it is set to 0. The motivation of using this feature is to distinguish proper nouns from the other words. We have observed that very short words are rarely proper nouns.

•Frequent word list: A list of most frequently occurring words in the training corpus has been prepared. The words that occur more than 10 times in the entire training corpus are considered to be the frequent words. The feature 'FrequentWord' is set to 1 for those words that are in this list; otherwise, it is set to 0.

•Function words: A list of function words has been prepared manually. This list has 743 number of entries. The feature 'FunctionWord' is set to 1 for those words that are in this list; otherwise, the feature is set to 0.

•Inflection Lists: Various inflection lists were created manually by analyzing the various classes of words in the Bengali news corpus during lexicon development. A simple approach of using these inflection lists is to check whether the current word contains any inflection of these lists and to take decision accordingly. A feature 'Inflection' is defined in the following way:

1) If the current word contains any noun inflection then the feature 'Inflection' is set to 1.
2) If the current word contains any verb inflection then the value of 'Inflection' is set to 2.
3) If the current word contains any adjective inflection, then the feature 'Inflection' is set to 3.
4) The value of the feature is set to 0 if the current word does not contain any noun, adjective or verb inflection.

VII. Unknown Word Handling Techniques for POS Tagging using HMM and SVM

Handling of unknown word is an important issue in POS tagging. For words, which were not seen in the training set, $P(t_i|w_i)$ is estimated based on the features of the unknown words, such as whether the word contains any particular suffix. The list of suffixes include mostly the noun, verb and adjective inflections. This list has 435 suffixes. The probability distribution of a particular suffix with respect to any specific POS tag is calculated from all words in the training set that share the same suffix.

In addition to the unknown word suffixes, the CRF-based NER system [21] and the lexicon have been used to tackle the unknown word problems. Details of the procedure is given below:

1) Step 1: Find the unknown words in the test set.
2) Step 2: The system assigns the POS tags, obtained from the lexicon, to those unknown words that are found in the lexicon. For noun, verb and adjective words of the lexicon, the system assigns the NN (Common noun), VFM (Verb finite main) and the JJ (Adjective) POS tags, respectively.
   Else
3) Step 3: The system considers the NE tags for those unknown words that are not found in the lexicon
   a) Step 2.1: The system replaces the NE tags by the appropriate POS tags (NNPC [Compound proper noun] and NNP [Proper noun]).
   Else
4) Step 4: The remaining words are tagged using the unknown word features accordingly.

VIII. Evaluation of Results of the POS Taggers

The HMM-based and SVM-based POS taggers are evaluated with the same data sets. Initially, the POS taggers are evaluated with Bengali by including the unknown word handling techniques, discussed earlier. We then evaluate the POS taggers with Hindi and Telugu data. The SVM-based system uses only the language independent features that are applicable to both Hindi and Telugu. Also, we have not used any unknown word handling techniques for Hindi and Telugu.

A. Data Sets

The POS tagger has been trained on a corpus of 72,341 tokens tagged with the 26 POS tags, defined for the Indian languages. This 26-POS tagged training corpus was obtained from the NLPAI ML Contest-2006[6] and SPSAL-2007[7] contest data. The NLPAI ML 2006 contest data was tagged with the 27 different POS tags and had 46,923 tokens. This POS tagged data was converted into the 26-POS [8]tagged data by defining an appropriate mapping. The SPSAL-2007 contest data was

[6]http://ltrc.iiitnet/nlpai_ contest06/data2
[7]http://shiva.iiit.ac.in/SPSAL2007
[8]http://shiva.iiit.ac.in/SPSAL2007/iiit_ tagset_ guidelines.pdf

TABLE IV
TRAINING, DEVELOPMENT AND TEST SET STATISTICS

Language	TRNT	NTD	TST	UTST	UTST (%)
Bengali1	72,341	15,000	35,000	8,890	25.4
Hindi	21,470	5,125	5,681	1,132	19.93
Telugu	27,513	6,129	5,193	2,375	45.74

TABLE VI
EXPERIMENTAL RESULTS OF HINDI AND TELUGU IN HMM

Language	Model	Accuracy (in %)
Hindi	Baseline	51.2
Hindi	HMM	73.75
Telugu	Baseline	40.87
Telugu	HMM	64.09

tagged with 26 POS tags and had 25,418 tokens. Out of 72,341 tokens, around 15K tokens are selected as the development set and the rest has been used as the training set. The systems are tested with a gold standard test set of 35K tokens. We collect the data sets of Hindi and Telugu from the SPSAL-2007 contest. Gold standard test sets are used to report the evaluation results.

Statistics of the training, development and test set are presented in able IV. Following abbreviations are used in the table:
TRNT: No. of tokens in the training set
TST: No. of tokens in the test set
NTD: No. of tokens in the development set
UTST: No. of unknown tokens in the test set

B. Baseline Model

We define the *baseline* model as the one where the POS tag probabilities depend only on the current word:

$$P(t_1, t_2, \ldots, t_n | w_1, w_2, \ldots, w_n) = \prod_{i=1,\ldots,n} P(t_i, w_i).$$

In this model, each word in the test data will be assigned the POS tag, which occurred most frequently for that word in the training data. The unknown word is assigned the POS tag with the help of lexicon, named entity recognizer [21] and word suffixes for Bengali. For unknown words in Hindi and Telugu, some default POS tags are assigned.

C. Evaluation of Results of the HMM-based Tagger

Initially, the HMM based POS tagger has demonstrated an accuracy of 79.06% for the Bengali test set. The accuracy increases upto 85.56% with the inclusion of the different techniques, adopted for handling the unknown words. The results have been presented in Table V.

The POS tagger is then evaluated with Hindi and Telugu data. Evaluation results are presented in Table VI for the test sets.

It is observed from Table V- Table VI that the POS tagger performs best for the Bengali test set. The key to this higher accuracy, compared to Hindi and Telugu, is the mechanism of handling of unknown words. Unknown word features, NER system and lexicon features are used to deal with the unknown words in the Bengali test data. On the other hand, the system cannot efficiently handle the unknown words problem in Hindi and Telugu. Comparison between the performance of Hindi and Telugu shows that the POS tagger performs better with Hindi. One possible reason is the presence of large number of unknown words in the Telugu test set. Agglutinative nature of the Telugu language might be the another possible behind the fall in accuracy. The presence of the large number of unknown words in the Telugu test set. Agglutinative nature of the Telugu language might be the other possible reason behind the fall in accuracy.

D. Evaluation Results of the SVM-based POS Tagger

We conduct a number of experiments in order to identify the best set of features for POS tagging in the SVM model by testing with the development set. We have also conducted several experiments by considering the various polynomial *kernel functions* and found that the system performs best for the polynomial *kernel function* of degree two. Also, it has been observed that the pairwise multi-class decision strategy performs better than the one-vs-rest strategy. The meanings of the notations, used in the experiments, are defined below:
pw, cw, nw: Previous, current and the next word
pwi, nwi: Previous and the next ith word
pre, suf: Prefix and suffix of the current word
ppre, psuf: Prefix and suffix of the previous word
pp: POS tag of the previous word
ppi: POS tag of the previous ith word
pn, cn, nn: NE tags of the previous, current and the next word
pni: NE tag of the previous ith word
$[i, j]$: Window of words spanning from the ith left position to the jth right position, where $i, j > 0$ indicates the words to the right of the current word, $i, j < 0$ indicates the words to the left of the current word, current word is at 0th position.

Evaluation results of the system for the development set are presented in Tables VII- VIII.

Evaluation results (3rd row) of Table VII show that word window $[-2, +2]$ gives the best result with the context window of size five, i.e., previous two and next two words along with the current word. Results also show the fact that further increase (4th and 5th rows) or decrease (2nd row) in window size reduces the accuracy of the POS tagger. Experimental results (6th and 7th rows) show that the accuracy of the POS tagger can be improved by including the dynamic POS information of the previous word(s). Clearly, it is seen that POS information of the previous two words are more effective and increases the accuracy of the POS tagger to 66.93%. Experimental results (8th-10th rows) show the effectiveness of prefixes and suffixes upto a particular length for the highly inflective Indian languages as like Bengali. The prefixes and suffixes of length upto three characters are more effective. Results (10th row) suggest that inclusion of surrounding word suffixes and/or prefixes reduces the accuracy.

It can be decided from the results (2nd-5th rows) of Table VIII that the named entity (NE) information of the current and/or the surrounding word(s) improves the overall accuracy of the POS tagger. It is also indicative from this results (3rd

TABLE V
EXPERIMENTAL RESULTS OF THE TEST SET FOR BENGALI IN HMM

Model	Accuracy (in %)
Baseline	55.9
HMM	79.06
HMM + Lexicon (Unknown word handling technique)	81.87
HMM +Lexicon (Unknown word handling) + NER (Unknown word handling technique)	83.09
HMM+ Lexicon (Unknown word handling) + NER (Unknown word handling) + Unknown word features	85.56

TABLE VII
RESULTS OF THE DEVELOPMENT SET FOR BENGALI IN SVM

Feature (word, tag)	Accuracy (in %)
pw, cw, nw	63.27
pw2, pw, cw, nw, nw2	64.32
pw3, pw2, pw, cw, nw, nw2, nw3	63.53
pw3, pw2, pw, cw, nw, nw2	64.16
pw2, pw, cw, nw, nw2, pp	66.08
pw2, pw, cw, nw, nw2, pp, pp2	66.93
pw2, pw, cw, nw, nw2, pp, pp2, $\|pre\| \leq 4$, $\|suf\| \leq 4$	70.97
pw2, pw, cw, nw, nw2, pp, pp2, $\|pre\| \leq 3$, $\|suf\| \leq 3$	71.34
pw2, pw, cw, nw, nw2, pp, pp2, $\|pre\| \leq 3$, $\|suf\| \leq 3$, $\|ppre\| \leq 3$, $\|psuf\| \leq 3$	70.23

TABLE VIII
RESULTS OF THE DEVELOPMENT SET FOR BENGALI IN SVM

Feature (word, tag)	Accuracy (in %)
pw2, pw, cw, nw, nw2, pp, pp2, $\|pre\| \leq 3$, $\|suf\| \leq 3$, pn, cn, nn	73.31
pw2, pw, cw, nw, nw2, pp, pp2, $\|pre\| \leq 3$, $\|suf\| \leq 3$, pn, cn	74.03
pw2, pw, cw, nw, nw2, pp, pp2, $\|pre\| \leq 3$, $\|suf\| \leq 3$, cn, nn	73.86
pw2, pw, cw, nw, nw2, pp, pp2, $\|pre\| \leq 3$, $\|suf\| \leq 3$, cn	73.08
pw2, pw, cw, nw, nw2, pp, pp2, $\|pre\| \leq 3$, $\|suf\| \leq 3$, pn, cn, Digit, Symbol, Length, FrequentWord, FunctionWord	77.43
pw2, pw, cw, nw, nw2, pp, pp2, $\|pre\| \leq 3$, $\|suf\| \leq 3$, pn, cn, Digit, Symbol, Length, FrequentWord, FunctionWord, Lexicon	82.82
pw2, pw, cw, nw, nw2, pp, pp2, $\|pre\| \leq 3$, $\|suf\| \leq 3$, pn, cn, Digit, Symbol, Length, FrequentWord, FunctionWord, Lexicon, Inflection	86.08

row) that the NE information of the previous and current words, i.e, within the window $[-1, 0]$ is more effective than the NE information of the windows $[-1, +1]$, $[0, +1]$ or the current word alone. An improvement of 3.4% in the overall accuracy is observed with the use of 'Symbol', 'Length', 'FrequentWord', 'FunctionWord' and 'Digit' features. The use of lexicon as the features of SVM model further improves the accuracy by 5.39% (7th row). Accuracy of the POS tagger rises to 86.08% (8th row), an improvement of 3.26%, by including the noun, verb and adjective inflections.

Evaluation results of the POS tagger by including the various mechanisms for handling the unknown words are presented in Table IX for the development set. The table also shows the result of the *baseline* model. Results demonstrate the effectiveness of the use of various techniques for handling the unknown words. Accuracy of the POS tagger increases by 5.44% with the use of lexicon, named entity recognizer [21] and unknown word features.

A gold standard test set of 35K tokens are used to report the evaluation results of the system. Experimental results of the system along with the *baseline* model are presented in Table X for the test set. The SVM-based POS tagger has demonstrated an accuracy of 85.46% with the various contextual and orthographic word-level features. Finally, the POS tagger has shown the overall accuracy of 91.23%, which is an improvement of 5.77% by using the various techniques for handling the unknown words.

In order to evaluate the POS tagger with Hindi and Telugu, we retrain the SVM model with the following language independent features that are applicable to both the languages.

1) Context words: Preceding two and following two words.
2) Word suffix: Suffixes of length upto three characters of the current word.
3) Word prefix: Prefixes of length upto three characters of the current word.
4) Dynamic POS information: POS tags of the current and previous word.
5) Made up of digits: Check whether current word consists of digits.
6) Contains symbol: Check whether the current word contains any symbol.
7) Frequent words: a feature is set appropriately for the most frequently occurring words in the training set.
8) Length: Check whether the length of the current word is less than three.

Experimental results are presented in Table XI for Hindi and Telugu. Results show that the system performs better for Hindi with an accuracy of 77.08%. Accuracy of the system for Telugu is 68.15%, which is less than 19.93% compared to Hindi. The *baseline* model has demonstrated the accuracies of 53.89%, and 42.12% for Hindi, and Telugu, respectively.

TABLE IX
RESULTS OF THE DEVELOPMENT SET FOR BENGALI WITH UNKNOWN WORD HANDLING MECHANISMS IN SVM

Feature (word, tag)	Accuracy (in %)
Baseline	55.9
SVM	86.08
SVM + Lexicon (Unknown word handling technique)	88.27
SVM +Lexicon (Unknown word handling) + NER (Unknown word handling technique)	89.13
SVM+ Lexicon (Unknown word handling) + NER (Unknown word handling) + Unknown word features	91.52

TABLE X
EXPERIMENTAL RESULTS OF THE TEST SET FOR BENGALI IN SVM

Feature (word, tag)	Accuracy (in %)
Baseline	54.7
SVM	85.46
SVM + Lexicon (Unknown word handling technique)	88.15
SVM +Lexicon (Unknown word handling) + NER (Unknown word handling technique)	90.04
SVM+ Lexicon (Unknown word handling) + NER (Unknown word handling) + Unknown word features	91.23

TABLE XI
EXPERIMENTAL RESULTS OF HINDI AND TELUGU IN SVM

Language	Set	Accuracy (in %)
Hindi	Development	78.16
Hindi	Test	77.08
Telugu	Development	68.81
Telugu	Test	68.15

E. Error Analysis

For Bengali gold standard test set, we conducted error analysis for each of the models (HMM and SVM) of the POS tagger with the help of confusion matrix. A close scrutiny of the confusion matrix suggests that some of the probable tagging errors facing the current POS tagger are NNC vs NN, JJ vs NN, JJ vs JVB, VFM vs VAUX and VRB vs NN. A multiword extraction unit for Bengali would have taken care of the NNC vs NN problem. The other ambiguities can be taken care of with the use of linguistic rules.

IX. CONCLUSION

In this work, we have used a Bengali news corpus, developed from the web-archive of leading Bengali newspaper, for lexicon development and POS tagging. Lexicon has been developed in an unsupervised way and contains approximately 0.128 million entries. Manual check of the lexicon has shown an accuracy of 79.6%. We have developed POS taggers using HMM and SVM. The POS tagger has shown the highest accuracy of 91.23% for Bengali in the SVM model. This is an improvement of 5.67% over the HMM-based POS tagger. Evaluation results of the POS taggers for Hindi and Telugu have also shown better performance in the SVM model. The SVM-based POS tagger has demonstrated the accuracies of 77.08%, and 68.81% for Hindi, and Telugu, respectively. Thus, it can be decided that SVM is more effective than HMM to handle the highly inflective Indian languages.

REFERENCES

[1] M. Rundell, "The Biggest Corpus of All," *Humanising Language Teaching*, vol. 2, no. 3, 2000.

[2] W. H. Fletcher, "Concordancing the Web with KWiCFinder," in *Proceedings of the Third North American Symposium on Corpus Linguistics and Language Teaching*, 23-25 March 2001.

[3] T. Robb, "Google as a Corpus Tool?," *ETJ Journal*, vol. 4, no. 1, Spring 2003.

[4] W. H. Fletcher, "Making the Web More Use-ful as Source for Linguists Corpora," *In Ulla Conor and Thomas A. Upton (eds.), Applied Corpus Linguists: A Multidimensional Perspective*, pp. 191–205, 2004.

[5] A. Kilgarriff and G. Grefenstette, "Introduction to the Special Issue on the Web as Corpus," *Computational Linguistics*, vol. 29, no. 3, pp. 333–347, 2003.

[6] A. Lenci, N. Bel, F. Busa, N. Calzolari, E. Gola, M. Monachini, A. Ogonowsky, I. Peters, W. Peters, N. Ruimy, M. Villegas, and A. Zampolli, "Simple: A General Framework for the Development of Multilingual Lexicons," *International Journal of Lexicography, Special Issue, Dictionaries, Thesauri and Lexical-Semantic Relations*, vol. XIII, no. 4, pp. 249–263, 2000.

[7] N. Calzolari, F. Bertagna, A. Lenci, and M. Monachini, "Standards and Best Practice for Multilingual Computational Lexicons, mile (the multilingual isle lexical entry)," *ISLE Deliverable D2.2 & 3.2*, 2003.

[8] F. Bertagna, A.Lenci, M. Monachini, and N. Calzolari, "Content interoperability of lexical resources, open issues and 'mile' perspectives," in *Proceedings of the LREC 2004*, pp. 131–134, 2004.

[9] T. Takenobou, V. Sornlertlamvanich, T. Charoenporn, N. Calzolari, M. Monachini, C. Soria, C. Huang, X. YingJu, Y. Hao, L. Prevot, and S. Kiyoaki, "Infrastructure for Standardization of Asian Languages Resources," in *Proceedings of the COLING/ACL 2006*, pp. 827–834, 2006.

[10] D. Cutting, J. Kupiec, J. Pedersen, and P. Sibun, "A Practical Part-of-Speech Tagger," in *Proceedings of the Third Conference on Applied Natural Language Processing*, pp. 133–140, 1992.

[11] B. Merialdo, "Tagging English Text with a Probabilistic Model," *Computational Linguistics*, vol. 20, no. 2, pp. 155–171, 1994.

[12] T. Brants, "TnT: A Statistical Part-of-Speech Tagger," in *Proceedings of the sixth International Conference on Applied Natural Language Processing ANLP-2000*, pp. 224–231, 2000.

[13] A. Ratnaparkhi, "A maximum entropy part-of -speech tagger," in *Proc. of EMNLP'96.*, 1996.

[14] J. Laffertey, A. McCallum, and F. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *Proceedings of the 18th International Conference on Machine Learning*, 2001.

[15] T. Kudo and Y. Matsumoto, "Chunking with Support Vector Machines," in *Proceedings of NAACL*, pp. 192–199, 2001.

[16] S. Singh, K. Gupta, M. Shrivastava, and P. Bhattacharyya, "Morphological richness offsets resource demand-experiences in constructing a pos tagger for hindi," in *Proceedings of the COLING/ACL 2006*, pp. 779–786, 2006.

[17] P. Avinesh and G. Karthik, "Part Of Speech Tagging and Chunking using Conditional Random Fields and Transformation Based Learning," in *Proceedings of IJCAI Workshop on Shallow Parsing for South Asian Languages*, pp. 21–24, 2007.

[18] S. Dandapat, "Part Of Specch Tagging and Chunking with Maximum Entropy Model," in *Proceedings of the IJCAI Workshop on Shallow Parsing for South Asian Languages*, (Hyderabad, India), pp. 29–32, 2007.
[19] A. Ekbal, R. Haque, and S. Bandyopadhyay, "Maximum Entropy based Bengali Part of Speech Tagging," in *A. Gelbukh (Ed.), Advances in Natural Language Processing and Applications, Research in Computing Science (RCS) Journal*, vol. 33, pp. 67–78.
[20] A. Ekbal, R. Haque, and S. Bandyopadhyay, "Bengali Part of Speech Tagging using Conditional Random Field," in *Proceedings of the seventh International Symposium on Natural Language Processing, SNLP-2007*, 2007.
[21] A. Ekbal, R. Haque, and S. Bandyopadhyay, "Named Entity Recognition in Bengali: A Conditional Random Field Approach," in *Proceedings of 3rd International Joint Conference Natural Language Processing (IJCNLP-08)*, pp. 589–594, 2008.
[22] A. Ekbal and S. Bandyopadhyay, "A Web-based Bengali News Corpus for Named Entity Recognition," *Language Resources and Evaluation Journal*, vol. 40, pp. 10.1007/s10579–008–9064–x, 2008.
[23] D. Jurafsky and J. H. Martin, *Speech and Language Processing*. Prentice-Hall, 2000.
[24] A. J. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *IEEE Transaction on Information Theory*, vol. 13, no. 2, pp. 260–267, 1967.
[25] V. N. Vapnik, *The Nature of Statistical Learning Theory*. New York, NY, USA: Springer-Verlag New York, Inc., 1995.
[26] C. C and V. N. Vapnik, "Support Vector Networks," *Machine Learning*, vol. 20, pp. 273–297, 1995.
[27] T. Joachims, *Making Large Scale SVM Learning Practical*, pp. 169–184. Cambridge, MA, USA: MIT Press, 1999.
[28] H. Taira and M. Haruno, "Feature Selection in SVM Text Categorization," in *Proceedings of AAAI-99*, 1999.

Methods for Handling Spontaneous E-commerce Arabic SMS: CATS, an Operational Proof of Concept

Maher Daoud and Christian Boitet

***Abstract*—The purpose of this paper is to show that it is necessary and possible to build (multilingual) NL-based e-commerce systems with mixed sublanguage and content-oriented methods. The analysis of the sublanguage and the integration of content-oriented methods will definitely increase the accuracy and robustness of the processing. To verify this assumption, we built an experimental system as a proof of concept. The system is a SMS-based classified ads selling and buying platform. To analyze the sublanguage, we first used a web based corpus to build the basic system. A content representation language is defined to capture the meaning of a classified ad post. The semantic grammars of content extraction are coded using the EnCo. Response generation is based on semantic matching ("looking for" and "sell" posts) and reasoning and is able to handle "no answer situations". CATS is currently deployed in Jordan by Fastlink (the largest mobile operator). Testing the content extraction component with a real noisy free texts shows a 90% F-measure.**

***Index Terms*—Spontaneous NL interface, SMS services, sublanguages, content extraction, classified ads, Arabic processing.**

I. Introduction

A natural language interface accepts users' inputs in natural language interacting with typically retrieval systems, which then results in appropriate responses to the commands or query statements. Hence, a natural language (NL) interface should be able to transform unrestrained natural language statements into proper actions for the system.

This type of unrestricted NL interface is an interesting choice because, if it could be built, it would offer many advantages. Firstly, it does not involve any learning and training, because its syntax and vocabulary are already familiar to the user. Secondly, natural language enables users to encode complex meanings. Thirdly, this type of interface is text-based, making it suitable for all types of devices and medium. In contrast, form-based or graphical user interfaces need more sophisticated and specific resources.

Incorporating a NL interface requires translating ambiguous user's inputs into clear intermediate representations. Two main problems are associated with building such systems: handling linguistic knowledge, and handling domain knowledge.

The study of the current scene shows that deployed or operational e-commerce NL interface systems are rare and most of them are only prototypes. This problem is not related to the openness or restrictedness of the domain. Although most e-commerce activities are domain-specific, we did not yet find any e-commerce operational system offering an interface based on a restricted but natural *sublanguage*.

NL-based systems have the reputations of high development cost and low quality. Our goal in this paper is to show that the most important factor in building NL-based systems is the selection of adequate methods for the development, regardless of the targeted language, in terms of richness of resources, or type or complexity of the domain, or even cleanliness of the input text. If this is approach is combined with treating a NLP project as an engineering problem, and not only as a traditional linguistic problem, it is almost guaranteed to produce a system with industrial quality and high extensibility, with the minimum resources possible.

Hence, we built an experimental system as a proof of concept. The system is a SMS-based classified ads selling and buying platform. It allows users to send classified ads describing the articles/goods they would like to sell or to search for, using full natural language interface. The system extracts content from both "sell" and "looking for" posts and transforms the natural language text into a corresponding content representation. For a "sell" post, the content representation is mapped into database records and stored into a RDMS. For a "looking for" type of posts, the content representation is used to build a SQL query to retrieve information from the data that has previously been processed and stored in the RDMS.

This paper is divided into three parts. The first describes the current scene concerning our assumptions and our proposed solution. In this part, we describe the main requirements of the proposed system, its main components, and its internal and external data specifications.

Manuscript received May 2, 2008. Manuscript accepted for publication June 18, 2008.

Daoud Maher Daoud is with Amman University, PoBox 141009, Zip Code 11814, Amman Jordan (Daoud@batelco.jo, Daoud.Daoud@imag.fr)

Christian Boitet is with GETALP, LIG, Université Joseph Fourier, 385 rue de la Bibliothèque, BP n° 53, 38041 Grenoble, cedex 9, France (e-mail: Christian.Boitet@imag.fr)

In the second part, we focus on the Content Extraction process. We describe the programming language used, our lingware engineering methodology, and our approach to the extraction of content from Arabic spontaneous and noisy text.

In the final part, we describe some operational aspects of the CATS system and its current status, before evaluating and comparing it with other systems. We also discuss issues related to porting the system to other languages and other domains.

I. The Scene and Problems of Current Approaches

The study of the current e-commerce systems shows that no e-commerce system available today is able to handle spontaneous users' requests online. Those projects avoid this hard problem by simplifying the user interface either by using controlled languages, form filling, or NLDI.

For example, the failure of MKBEEM [1] [2] to provide full spontaneous NL interface is due the use of methods and tools which are too complicated for the task. When we trace the project back to the beginning we find that one of its main objectives was providing unrestricted NL interface. However, we could not find any evidence in the literature that this goal was ever achieved or demonstrated. The methodology used to extract content is very complicated. Initially, the input text is processed syntactically and several dependency parse trees are produced by WEBTRAN [3]. Those dependency trees are then processed and mapped into semantic representations, which are finally transformed into CARIN (an ontological representation). Apparently, MKBEEM used these long and complicated steps of transforming one representation into another to meet the requirements of multilingualism which are provided by WEBTRAN. WEBTRAN is a machine translation system that analyses input texts syntactically [4, 5]. The developers of this project decided to transform the syntactic representations into semantic ones which led to these complicated, long, and possibly error-prone processing steps.

- As for MIETTA [6], it is also a multilingual system. However, it avoided the use of full natural language interface and only was used form filling interface and keywords processing.
- Similarly, TREE [7] avoided the use of full natural language interfaces and used form filling to interact with users in different languages.
- The HappyAssistant [8] prototype used a very limited NL processing for noun phrases only to provide NLDI.
- CASA [9] had also a form filling interaction style with keywords-based processing.
- Finally, GOOGLE SMS is uses a very restricted language (close to a command language) to interact with users.

On the document processing side, we have seen that some systems had a processing component for this task. CASA, TREE and MIETTA provided a shallow parsing for the semi-structured documents they processed. MKBEEM used full parsing to process controlled-language documents.

Looking carefully at the above systems, we see that many of their authors realized the importance of having internal representations for more precise processing. As an example, MIETTA and TREE used language-independent templates to store extracted information from documents. On the other hand, MKBEEM used several internal representations for mapping and the inferring.

A. Proposed Methodology

Thus, if the free natural language style is the best method for interactions with end users, why is it that most of the above systems avoided implementing it, or failed in delivering it in a robust way? There are different possible reasons:

- All of the above systems are Web-based. Hence, form filling and other graphical user interfaces are viable options, imposing only slightly more constraints on the users than a full NL interface.
- The developers of these systems did not take into account the restricted nature of their systems and the associated sublanguage that can be exploited in building a high quality system without settling for less interesting alternatives.
- Building a "production system" requires to take into consideration many constraints (concurrency, short response time, etc.) that are neglected when building a prototype. Therefore, transforming a prototype into a real system is often unfeasible because it requires major changes that may be impossible to perform.
- The use of inadequate techniques. This was manifested by MKBEEM project which imposed a controlled language on users' inputs, but with inadequate methods and techniques.

In total, we think that using inadequate techniques is the main source of this failure. As an example, using deep syntactic parsing for telegraphic ungrammatical sentences will certainly be unsuccessful. Similarly, using tools and techniques suitable for rigid word order languages will not certainly produce good results if applied on languages with free word order. Another example of inadequate technique is the use of open domain techniques for domain-dependent systems. It is necessary for such systems to take advantage of the narrow scope both linguistically and semantically for such restricted domains.

It is assumed that any applied system will be oriented toward the particular variety of natural language associated with a single knowledge domain. This follows from the now widely accepted fact that such systems require rather tight, primarily semantic, constraints to obtain a correct analysis, and that such constraints can at present be stated only for sublanguages, not for a whole natural language [10]. In that sense, incorporating the accurate linguistic description of a sublanguage into a natural language system will definitely increase the accuracy and robustness of processing.

On the other hand, knowledge representations and content-oriented methods are necessary for building accurate NL-based transactional systems such as e-commerce systems, because they provide the necessary mechanisms for

normalization, unification, transformation, abstraction and compensation of information that exist in human language processing.

Therefore, our paper will show that it is necessary and possible to build (multilingual) NL-based e-commerce systems for limited domains with mixed sublanguage and content-oriented methods.

II. A Corpus-based Development

A corpus-based approach will certainly lead to a better understanding of the sublanguage used and the way people encode their thoughts in this domain. In turn, this will help in selecting the right approach for development. As an example, systems developed for semi-structured text are not appropriate for free text and vice-versa. The assumption that SMS-based classified ads are semi-structured or free text needs to be verified. Developing information systems that depend on natural, spontaneous and unprocessed text requires techniques and approaches different from those used for edited text. Most of the current systems that process users queries and generate responses use shallow text processing techniques based on pattern extraction or information retrieval techniques [11]. However, systems such as CATS require deeper text understanding methods [12].

A. *The Scarcity of Data*

The shortage of data is one of the main obstacles in developing natural language systems. It is not easy to collect corpuses for restricted domain, especially if they must come from a very private medium of communication such as SMS.

We could not find any references that discuss the features of Arabic SMS messages in any domain. Additionally, mobile operators refused to provide us with any excerpt of real SMS messages, to maintain the privacy of their customers.

B. *Choice of a Web-based E-commerce Corpus*

In [13], it is found in experiment with different domains, that the best parsing performance was obtained for the same domain (religion, romance and love stories, etc.), followed by the same class (fiction or non-fiction), and the worst was obtained on domains within a different class.

In selecting a similar corpus, the main condition to consider is the spontaneous and unedited nature of the text. Therefore, texts from printed material were excluded. The only possibility we had was to look for a web site providing unedited Arabic classified ads services. Fortunately, we found a Jordanian one (http://www.almumtaz.com) that provides this service in Arabic for the Cars and Real Estate domains.

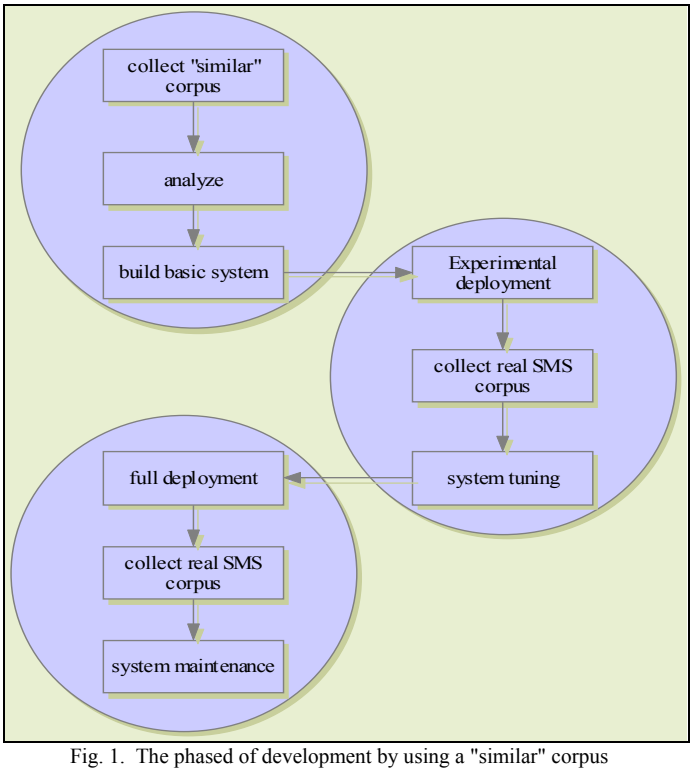


Fig. 1. The phased of development by using a "similar" corpus

As shown in figure 1, we can distinguish between 3 phases in a corpus-based development:

- Design phase and basic implementation: in this phase, we study the corpus with the aim of assigning semantic classes, specifying most frequent words, and depict the lexicon, styles and types of queries that interest users. We also made decisions on what is relevant and what is not relevant to a particular domain. The two outputs of this phase are the design of the knowledge representation and the design of the dictionary. Consequently, we build the basic NL system which consists of the extraction rules and the dictionary. Lexical items are added to the dictionary based on most frequent words. For the encoding of the rules, we use iterative procedures. We manually extract a first set of relevant patterns of the domain. These patterns are then encoded into extraction rules that are applied on the corpus. The coverage of the rules is increasingly expanded until good performance is achieved on the corpus.
- Experimental deployment phase: in this phase, we put the system into full operation, but for testing purposes. Each processed post is evaluated manually. Accordingly, corrective/updating measures are taken in the rules and/or the dictionary. When the number of maintenance tasks becomes smaller and smaller, we move to the full deployment phase.
- Full deployment phase: in this phase, the system is fully operational. Maintenance tasks are based on users' feedback and internal quality assurance procedures.

III. Sublanguage Analysis

It is noticeable that in restricted domains of knowledge, among certain groups of people and in particular types of texts, people have their own way of encoding their thoughts. Such restrictions can be said to reduce the degree of lexical and syntactic variation in text [14]. These specific languages are called either sublanguages or restricted or specialized languages.

As presented in figure 2, the analysis of the linguistic aspects and features of a sublanguage is needed to specify the sublanguage grammar (with the incorporation of the domain knowledge). Then general linguistic knowledge and sublanguage grammar can be used to determine the best NL technique to use. Similarly, the sublanguage grammar and the domain knowledge are both indispensable in selecting the best content representation.

A. *Typology of SMS-based Task-oriented Sublanguages*

To measure the lexical complexity of SMS-based classified ads sublanguage, we use the type-token ratio (TTR). This ratio increases with the lexical complexity and richness of the text and decreases if more words repeat themselves and the lexical complexity is lower. We calculated the TTR for different corpuses for the sake of comparison.

We measure the language complexity by the length of the sentence in words. Finally, finding the words frequency in a corpus identifies the nature of text (telegraphic or normal), in particular the less the percentage of function words in a corpus, the more fragmentary is its style.

The analysis of the sublanguage also includes the manual study of lexico-semantic patterns found in the posts. Our objective is extracting classes of objects that specify the domain knowledge described by the sublanguage.

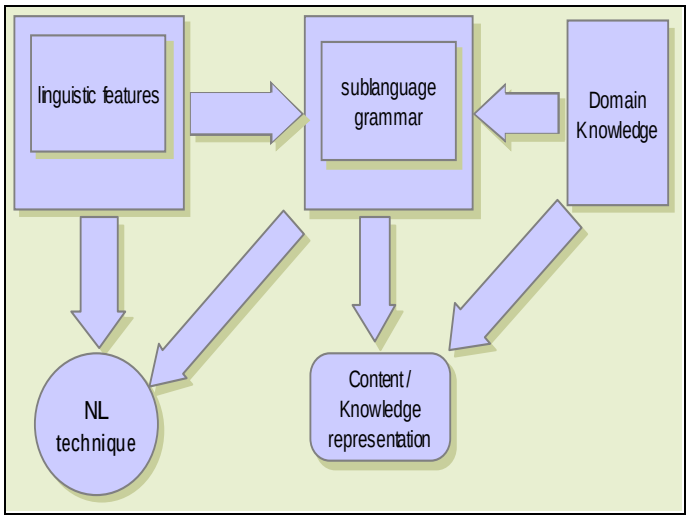


Fig. 2. NL development using sublanguage study

B. *General Corpus Statistics*

The SMS-based corpus consists of posts from Cars and Real Estate domains collected during a limited experimental period of CATS operation.

TABLE I
EXAMINED SMS-BASED CORPUS

Domain	Number of sentences	Sentence average length (words)	Types	Tokens	TTR
Cars	771	9	1181	5875	.201
Real Estate	641	12.5	1441	6182	.233

As it is shown in table I, the length of sentences in the Cars domain is less than that of the Real Estate domain, compared to 7.3 words for TREC questions. In other words, the user needs a lesser amount of words to encode his thoughts in the Cars domain than in the Real Estate domain.

When we compare SMS-based posts with Web-based posts, we find that the first are generally smaller than the second.

The findings also show that the least TTR value was for Cars at 0.201, then for Real Estate at 0.233.

The TTR values of Web-based posts were even lower compared to SMS based ones, suggesting a higher lexical complexity and diversity in the SMS-based text.

The TTR of general Arabic corpus of nearly the same text length (number of tokens) is 0.539 as calculated in [15], suggesting a more topical diversity than that found in classified ads.

Additionally, the top 50 most frequent used words percentage in SMS-based Cars and Real Estate are 53.77%, 45.76% respectively. These findings suggest that as we move from Cars to Real Estate, the percentage of function words (such as prepositions) increases. This finding can be correlated with the TTR of each sub-domain, indicating a less telegraphic text as we move from the Cars domain to the Real Estate domain.

C. *Lexical Characteristics*

Although the vocabulary used is narrow and limited, posters use different words to express the same concept. For example, to express the concept “more”, users use around 30 words (including spelling variations).

We observe that some words in the Cars and Real Estate domains can have different meanings than in the open domain. Therefore, specialized dictionaries are required to process the text. For example, in the Cars domain ‘*duck*’ denotes a Mercedes model, and a ‘*piece*’ in the Real Estate domain means a land.

Multi-word concepts and terms are also very frequent to the extent that they appear in the topmost frequent words list.

In the Cars domain, named entities are references to Car Makes and Models. In the Real Estate domain, they are references to Locations. The study of the corpus of classified ads shows that Named Entities consist of one or more words. As Arabic is not like English in distinguishing named entities

by capitalizing the first character, and sentences are very short, recognition of named entities is impossible without using lexical lookup.

The dataset under study is full of numerical values. In the Car domain, they represent price, year, motor size and sometime models for some car makes. In the Real Estate domain, they represent the price, area, number of bedrooms, etc. The posters encode numerical values differently. Some of them use non-Arabic numerals such as "three thousands". Others use Arabic numerals such as "3000". Finally, some posters combine the two approaches and write expressions such as "3 thousands". Usually, numerical values are preceded by hint words and/or followed by unit words. But, it becomes problematic when users fail to write both hints words and unit words, as demonstrated by the post:

"For sale Mercedes 200 1999"

There are many variations of spelling of the Arabic text in the studied corpus. For example, people write the Alef letter "ا", or with Hamza (ء) over it "أ" or under it "إ.". Also, we find confusions between the Ha' "ه" and Ta' "ة", and between Ya' "ي" and Alef-Maqsoura "ى".

Another problem is the wrong insertions of spaces. In Arabic, spaces are normally used to separate words. After some Arabic letters, people tend to wrongly insert a space, or to (also wrongly) omit it (e.g., "أبـو بكـر" or "أبوبكـــر" {Abu-Baker}).

The inconsistency of the Arabic spelling of transliterated proper nouns is also detected in the classified ads text where many of the proper names (car make and model as an example) are transliterated from other languages.

D. Syntactic Characteristics

The studied posts can have different syntactic structures caused by different word orders and grouping patterns of their constituents.

In some posts, we find that some constituents are not present because they do not interest the poster or are irrelevant for him, in cases such as "looking for a car above 2001". In this post, the user omits all other criteria that can restrict his query and mentions only one.

Other causes of omissions arise when information is supposed to be implicitly known, such as "looking for a Clio" in which "car" is omitted, or "for sale 500 square meter", in which "land" is omitted.

In some posts, we don't find any indication of the type ("sell" or "looking for"): "a Toyota Corolla above 99 and with less than 7000 dinar" because the poster thinks it can be known from the context of the post.

E. Semantic Characteristics

We have shown that the syntactic structure for different posts which express the same information can vary enormously.

Some posters encode the knowledge but at different levels of detail. For example: "looking for a CIVIC" or "A Japanese Honda Civic car for sale".

The use of generalization in the query is also presented in the studied corpus. For example, the use of a generalization concept for searching is quite frequent such "looking for a French car", "looking for a villa in West Amman" or "looking for economical car". Usually these words ("French", "West Amman" and "economical") do not appear in the "sell" post since they are implicitly known.

F. The Main Outcome of Sublanguage Analysis

The data that we studied contains many alternative surface structures for the same utterance. We believe this phenomenon reflects the diversity of the posters. It was evident from looking at the posts that there was no unique underlying syntactic structure in the sublanguage used. Some posts consist of fragmented phrases (telegraphic) rather than fully-formed sentences. Other posts are more cohesive and some are full sentences. Obviously, syntax-based parsing based methods would not prove very useful in dealing with the given data. As an example, a traditional parser looking for object and subject will fail in analyzing the following post:

"*Opel Astra station color red (power sunroof Center Electrical windows and mirrors check for sale*"

Similarly, techniques used for semi-structured text relying on position, layout and format of text are bound to fail on the given data.

Therefore we can view a classified ads post as sequence of properties restricting the main domain object (i.e. car, apartment). This statement is true for both Real Estate and Cars and for both "sell" and "looking for" posts. This information model is more efficient than relying on syntactic structures for the description of the SMS.

This approach of describing sentences semantically achieves better results than using a pure syntactic description. It is also part of our engineering methodology, which allows semantic knowledge to be easily included in the system [16].

The study suggests also the need for a lexical lookup able to handle spelling variations as well as to store a concepts hierarchy.

Because of the information structure attached to this sublanguage, it is also necessary to have a content representation able to model the post, and to normalize the knowledge in a post regardless of its original surface structure.

Hence, what is required is an additional level of abstraction that represents the underlying meaning of a post.

Formulating correct responses for users' queries is another motivation for defining a unique knowledge representation for both types of posts. Suppose we have the following "sell" post:

للبيـــع مـــنزل مســـتقل فـــي خلـــدا "For sale an independent house in Khalda" and that somebody sends the following query:

مطلـوب فيـلا فـي غـرب عمـان "Wanted *a villa in the West of Amman"*

Relying only on bag of words for finding answers is insufficient, and of course will lead to totally unacceptable

results, since none of the tokens in the "looking for" post matches any of those in the "sell" post. This example shows clearly the need to transform both posts into a language-independent structure that captures the meaning. This will enable the system to correctly find matches, because posts with similar meaning will be recognized, regardless of how they are structured grammatically and which particular terms are used.

IV. The CATS Architecture

The CATS is a C2C based e-commerce system that uses content extraction technology based on sublanguage analysis and knowledge representation to enable SMS users to post and search for classified ads in Arabic. It has two main functionalities: the submission for selling items and the answering of users' queries through interaction in spontaneous natural language. The system receives an entry in full text without any pre-specified layout, recognizes the various relevant bits of information, and produces a knowledge representation for further processing. We have two types of users' requests:

- "Sell" post: in which the user is a potential seller.
- "Looking for" post: in which the user is a potential buyer.

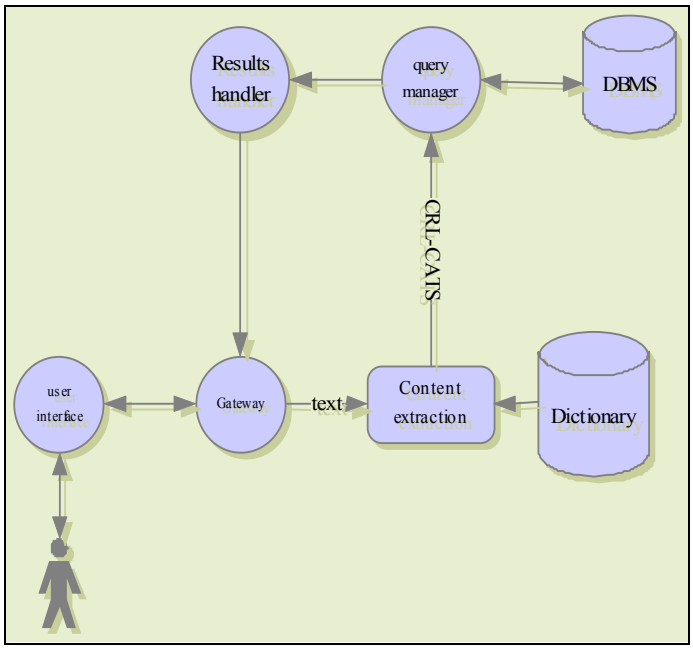


Fig. 3. Overall architecture of the CATS system

A. Overall Architecture

The overall structure of the CATS reflects both the corpus analysis and the adopted knowledge representation. The CATS system consists of a content extraction (CE) component and a query manager (QM) component.

The CE component receives SMS text and decodes it into the corresponding knowledge representation using a domain-specific lexicon. The system is able to extract knowledge from both types of messages.

The QM component takes the KR and converts it into SQL statements. It then issues the SQL statements (query or insert), and checks, validates and formats the results. It also handles situations where no answer found.

One important aspect of this design is that both questions and postings (documents) are processed by the same engine, using the same knowledge representation, leading to accurate matching of questions with answers.

B. The Content Representation Language for CATS

We have chosen a minimal but sufficient formalism to express the content of SMS used in posting or querying classified ads.

In CRL-CATS (Content Representation Language for CATS), a posted SMS is represented as a set of binary relations between objects. There are no variables, but the dictionary is used as a type lattice allowing specialization and generalization.

There is big advantage for us to use such a restricted formalism: as it is formally very near to the UNL formalism, we can use the same tool for CE as the tool we used a few years ago for writing the first Arabic-UNL enconverter, namely the EnCo specialized programming language.

The basic data model of CRL-CATS consists of three object types:

Main Domain Object (MDO). The central notion in CRL-CATS is that there are things that we wish to make assertions about. Examples of such things in the Cars domain are "Saloon" and "Pickup" and in the Real Estate domain are "Apartment" and "Villa".

Properties. A property is a specific aspect, feature, attribute, or relation used to describe a MDO. A "property" and its value are pieces of information that may be attached to things, but which are not sufficiently important in the specific domain to be considered things in their own right.

Some examples of properties of the thing "Red" is: the color of my car. In CRL-CATS, color is simply a property of the MDO "Saloon" and is encoded using the following statement:

Col (saloon, red)

Statement. A specific MDO together with a named property plus the value of that property for that MDO is a CRL-CATS statement:

mak(bus:06, HYUNDAI(country<korea):0R)

Here is a CRL-CATS expression encoding one classified ad post contains one or more CRL-CATS statements.[1]

[S]
wan(saloon:06, wanted:00)
mak(saloon:06, KIA(country<Korea):0C)
yea(saloon:06, 95:0L)
[/S]

[1] The labels ':00', ':06', ':0C', ':0U', etc. are identifiers associated by the DeCo engine to the "nodes" of the graphical representation, while the symbols 'sal', 'mak', etc. are labels on the arcs, created by the *grammar*.

For example, consider the following "sell" post:

للبيـــع ســـياره هونـــدا موديـــل 1997 جـــير اوتوماتيـــك مكيـــف ســـنتر بســـــعر 7750 دينـــار.

For sale Honda year 1997 automatic transmission air condition center lock price 7750 dinar

The CRL-CATS expression extracted from it is:

[S]
sal(saloon:06, sale:00)
mak(saloon:06, HONDA(country<japan):0C)
yea(saloon:06, 1997:0U)
fea(saloon:06, automatic gear:0Z)
fea(saloon:06, air condition:1D)
fea(saloon:06, center lock:1I)
pri(saloon:06, 7750:1S)
[/S]

In the above example, *mak* (make), *sal* (Sale), *pri* (price), *fea* (feature) and *yea* (year) are property labels. The nodes *saloon, sale, HONDA (country<japan), automatic gear, air condition* and *center lock* are CATS Words (CWs). The CW (CATS word) *saloon* represents the MDO; other CWs represent the values of the properties. The label *country<japan* is the semantic label for HONDA, providing information about the country of the manufacturer.

Note that a property such as *fea* (feature) can have multiple values ("air condition", "automatic", "center lock"). In other formalisms, we might have:

fea(saloon, [*air condition, automatic, center lock*]),

where [ ] stands for "and". Here, we simply allow any number of arcs with the same label going out of a node in the graphical representation.

V. Content Extraction in Arabic

CE from Arabic SMS presents not only the usual problems encountered when handling western languages, due to several characteristics:

1. People usually don't write the "small vowels", an orthographic word is much more ambiguous than in English, French, Italian, etc.
2. In some domains, such as Cars, there are many foreign words, which are transliterated in many different ways in the Arabic script by posters.

The main difficulty for us was the absence of freely usable lexical and syntactic resources and tools: Arabic is still a "pi-language" (poorly informatized). The other difficulties concern the treatment of named entities, the problem posed by spelling variations (dictionary size, need to handle "unknown" forms of known words), the free word order, and the presence of unpredictable long compound words.

A. CE CATS Structure

We conclude from our review of the literature that the rule-based approach is more suitable for building CATS. An automatically trainable approach cannot be as accurate as a rule-based approach and requires a huge set of structured or semi-structured data as training corpus, and is not available in our case.

We have chosen to write our CE in EnCo [17] because it was available and we could reuse and adapt to this new context (CE) what we had already developed while writing an Arabic-UNL enconverter (development methodology, dictionary and rules).

The task is different: we are not trying to translate the classified posts into another language, but we want to transform the posts into a higher abstraction that captures the meaning of the sentence, regardless of the original surface form.

In this way, it is possible to use EnCo to parse SMS Arabic language with the intention of producing a CRL-CATS expression, and not a UNL graph. To do this, we cannot use the full analysis rules and the associated dictionary. We have to develop a new rules based on the analysis of the classified ads sublanguage and to collect a new dictionary (or adopt the existing dictionary) to reflect the semantic classes of the domain.

B. Structure of the Dictionary

The dictionary of CATS is manually constructed for the Cars and Real Estate domains. It is the backbone of CATS since it drives the CE process, compensates for lexical inconsistency by providing synonym relations and by connecting words to concepts (CWs), and finally provides the semantic information needed for reasoning.

Different word forms are connected to one concept. A concept is a meaning pointed to by the CW. In a sense, a CW denotes a unique meaning while an unrestricted UW can denote different word senses [18].

This structure minimizes the effect of the alternative representations of text (including different orthographic forms, spelling errors, and abbreviations) on the overall performance of the system, specifically in the searching process.

The number of CWs in the dictionary for both domains is 10828, while the total number of lexical forms is 30982. On average around 3 forms point to the same CW.

The entries for the dictionary are collected from the corpus and many are generated automatically as we will see in the coming sections.

C. Extraction Rules

To perform the CE task, we have written 710 rules for both the Cars and Real Estate domains. The rules were written based on our analysis of the sublanguage used for the classified ads. The study of those posts in the corpus enabled us to design the CRL-CATS as a higher abstraction of knowledge. In the same manner, the EnCo rules are the outcome of sublanguage analysis, in which we collected all structures and patterns used by users.

A Car post consists of components: *make, model, color, sale, want, year, price, feature, country and motor size* in addition to the MDO which is a vehicle.

A Real Estate post consists of the following components: *sale, want, purpose, location, area, number of bedrooms, consist of, price, type, floor* and *feature* in addition to the MDO.

For example, identifying relations between the MDO and the property values is an essential part of CE engine. This is performed by identifying the MDO, linking it to the property values found in the text, and finally producing the CRL-CATS expressions. This is achieved by the DeCo rule:

<{vech:color_add::}{color::col:}()P70;

If the MDO is any type of vehicle and the right window contains a word representing *color value*, a *col* relation is built between *vech* and *color value*.

Similarly, the following rule will fire if the left window is a real estate MDO and the right window contains a node indicating "for sale". A *sal* relation is built connecting the MDO to the *sale* node.

<{flat:sale_add::}{sale::sal:}()P70;

VI. The QA Component: Database Design, Semantic Matching and Response Generations

CE handled mismatches at the local level or within the post "sell" or "looking for" only. On the other hand, CATS should also formulate responses (from previously processed and stored "sell" posts) to users' "looking for" posts. In a sense, variations between the two types are handled by using semantic matching. This will trigger another question: what type of storage is needed? Is it necessary to use storage with very general inference capabilities? Or we can perform the task with a light-weight inference storage that has other features such as reliability and concurrency?

A. Basic Implementation

During the past two decades, relational databases have been developed to a level that cannot be emulated by other storage means, semantic or non-semantic. This is because they accumulated essential and critical features such as scalability, reliability and concurrency, needed in building robust applications in various sectors.

In relational database systems, data objects are normally stored using a horizontal scheme [19]. A data object is represented as a row of a table. There are as many columns in the table as the number of attributes the objects have. Generally, CRL-CATS expressions are the source of the columns.

Additionally, the DB has to be designed to identify related concepts and to contain an inference mechanisms for deduction of information not explicitly asserted.

For example, when a "sell" post is received saying "for sale LANCER 1999", the system recognizes that it is a car, it is a Japanese car, and that the maker is Mitsubishi. Therefore, the system is capable of detecting and compensating for missing information in both types of messages. As a result, the above record would be one of the answers of the following post: "looking for a Japanese car".

B. Implementing Semantic Matching

In this design schema, we don't allocate any table for the ontology, but we use the semantic labels embedded within the CWs to fill concerned columns values, and to ensure that there are no null values in them.

Cars table

id	msgcaller	maincat	make	model	Country	MsgTxT
1	079667999	saloon	Renault	Clio	France	for sale a Clio
2	07989999	saloon	Renault	Clio	France	For sale a Renault Clio
3	07988856	saloon	Renault	Megan	France	For sale a Megan
4	079777	saloon	Peugeot	Null	France	for sale a Peugeot
5	078666	Saloon	Honda	Civic	Japan	for sale a Honda Civic
.....						

Fig. 4. Scenario 2 implementation

As shown in figure 4, the system inserts values for "make" and "country", regardless of their presence in the original "sell" post. In CATS, we used this design, because it performs the semantic matching with simpler queries and consequently with a higher performance.

C. Storing "Sell" Posts

For a "sell" post, the extracted information from the CRL-CATS and from other sources is passed to a stored procedure to generate the insert SQL statement.

To demonstrate the process of transformation, consider the following "sell" post *"for sale a Lancer 99 at 5000 dinar"*

The CRL-CATS for the above post is:

```
[S]
sal(saloon:00, sale:00)
mod(saloon:00,
Lancer(country<japan,make<MITSUBISHI):06)
yea(saloon:00, 99:0I)
pri(saloon:00, 5000:0L)
[/S]
```

Since it is a "sell" post, the system issues an insert SQL statement (as we have shown, this is performed in reality by using a stored procedure and involves more parameters) to populate the database with this post:

```
Insert into cars (maincat, model, year, price, country, make)
Values('saloon','lancer','99','5000','japan','mitsubishi')
```

Each property value in CRL-CATS fills the corresponding column in the Cars table in the database. Note that the semantic information (country and make) is extracted and mapped into prespecified columns to facilitate further semantic matching.

D. Processing of "Looking for" Posts

For example, the following CRL-CATS which corresponds to the query "*looking for a Mitsubishi Lancer*":

```
[S]
wan(saloon:00, wanted:00)
mak(saloon:00, MITSUBISHI(country<japan):06)
mod(saloon:00,
Lancer(country<japan,make<MITSUBISHI):0G)
[/S]
```

is converted to the following SQL query:

```
select MsgCaller from Cars where
 make ='mitsubishi'
and model ='lancer'
and maincat ='saloon'
```

Hence, the method of extracting semantic relations and storing them in the corresponding columns, regardless of their existence in the original "sell" post, makes possible the generation of that kind of simple and efficient queries.

E. *No Answer Situations*

We first try to answer a user's query as it is asked. If it has no answers, we relax it to a more general one, and try again [20]. For example, if no answer is found for the above query "*looking for a Mitsubishi Lancer*", the following SQL query will be issued:

```
select MsgCaller from Cars where
(
   make ='mitsubishi' or
   model ='lancer'
)
and maincat ='saloon'
```

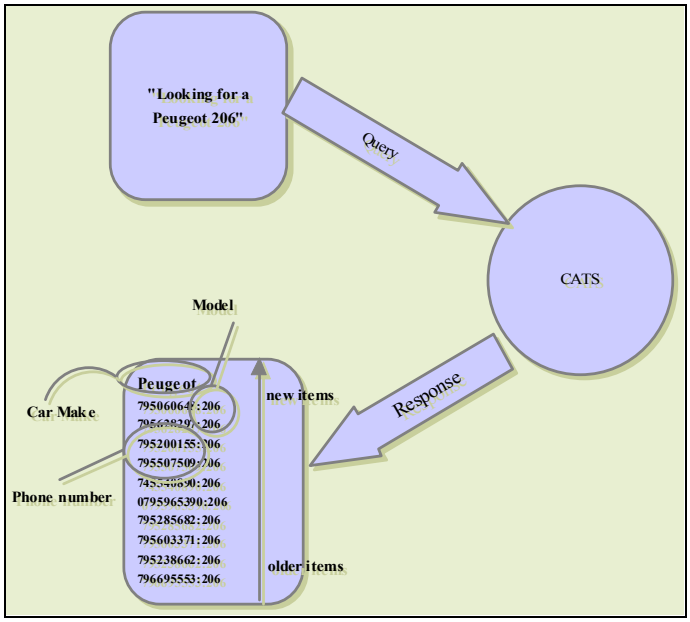


Fig. 5. Example of response in Cars

As for processing "sell" posts, a stored procedure is used within the DB to dynamically generate those queries. At the beginning, it will generate a query based on conjunctive conditions. If no answer retrieved, it will issue another query but this time with the "or" operator connecting these conditions.

In cases where no answer is found, even with this relaxation, the query is marked as *unanswered* by setting its SendFlag in the main table. A service (an agent) will periodically check at predefined time intervals the availability of any answer. As soon as an answer is found, it is then sent to the poster.

F. *Generating Responses*

Given the length constraint put by SMS, we used a tabular form to display the results as shown in figure 5.

Adding more information to the response, such as year, and price would reduce the number of displayed items. Also, many of the "sell" post lack information about year or price, which would cause irregularity in the response format.

The items within a response are ordered according to the sell post's time: the most recent one appears at the top of the list.

VII. Operational Experimentation, Evaluations and Discussion

A. *Status of the System*

This service is currently available in Jordan, where thousands of people have already used it to sell or buy cars or properties. The number of posts received depends on many factors such as the season or the marketing campaign by the mobile operators. Usually, after some marketing, we get on average 1000 posts per day, otherwise we get 20 ~30 posts per day.

B. *Evaluation*

Because the CATS system is targeting end users, we performed an end-to-end evaluation of the system by surveying users directly. We first explained the system to a sample of around 200 users from different backgrounds, and then asked them to test the system by posting "sell" and "looking for" SMS messages.

Generally, the feedback was positive: 95% of the participants said that results were accurate. The rest said that the results should be more precise. We have noticed that 70% of the messages are of the "looking for" type.

However, it is important to provide a quantitative metrics to measure performance and accuracy. As a restricted domain information system, CATS is a task-oriented system, and that should be considered in the evaluation. [21] specifies different user evaluation dimensions for this type of systems. In our case, CATS is a multi-component system and the CE component is the most important in evaluating the completeness, relevance, and accuracy of the responses. Additionally, it is also important to measure the performance of CATS in terms of the time it takes to respond to the users.

We used precision and recall rates to measure the quality of our answers. They were calculated as follows [22, 23]:

$$precision = \frac{\text{number of correct entities identified by the system}}{\text{number of entities identified by the system}}$$

$$recall = \frac{\text{number of correct entities identified by the system}}{\text{number of entities identified by a human}}$$

$$F - Measure = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

C. Experiment and Results

We designed and conducted an experiment to evaluate the usefulness and performance of our content extractor. A set of real posts was used as the testbed. It consisted of 100 posts per type per domain, not used in the development of the systems, and randomly selected from the posts received during the real operation of CATS.

A human experimenter manually processed these posts to identify all entities of interest. Theses posts contained a significant amount of typos, spelling errors, and grammatical mistakes. This added difficulty to the entity extraction process. On the other hand, however, it allowed us to test our system's robustness for noisy data sets.

TABLE II
RESULTS FOR THE TWO DOMAINS

Precision	92.7
Recall	87.2
F-measure	90

Table II shows the precision, recall and F-measures values for the Cars and Real Estate Domains.

We also remark that "looking for" posts show higher F-measure that "sell" posts. On the other hand, the Cars domain has a higher F-measure than the Real Estate domain, reflecting its higher complexity. We also observe that numerical entities have lower F-measure than textual entities, suggesting that numerical entities are harder to detect or to identify correctly.

D. Assessment

In general, the results indicate that our content extractor performs well in identifying different parts of information. Considering that the spontaneous free posts collected to conduct this evaluation were much noisier than the news articles used in MUC evaluations, CATS has a higher recall and precision than the results reported by MUC (unrestricted text: 60-70% R, 65-75% P, Semi-structured text: 90% R/P) [24].

For further assessment of our system, we compared our results with other more recent systems that use English: Phoebus [25], SimpleTagger [26], and AmilCare [27]. Phoebus uses semantic annotation for handling ungrammatical and unstructured text. SimpleTagger is a suite of text processing tools that is an implementation of Conditional Random Fields (CRF) which has been used in information extraction. Amilcare uses shallow natural language processing for information extraction. Unfortunately, we could not use any of the above directly for comparisons. Therefore, we use the comparison study conducted by [25] in two domains: hotel postings and comics books.

For the *price* entity CATS, scored a F-Measure (for all types and domains) of 81%, higher than the three systems in the comics books domain. For the hotel postings, it is better than Simpletagger and Amilcare but worse than Phoebus. For the *year* entity, in the Cars domain, CATS scores 89% higher than all other systems under consideration. For the *location* entity (in the Real Estate domain) which corresponds to the *area* in the hotel domain, CATS scored 91%, again higher than all other systems.

Hence, CATS despite the free, spontaneous and noisy nature of its input, has surpassed other systems in quality.

As to the performance of the system in terms of capacity and time to respond, it has shown high performance. CATS was tested for one post per second and it has performed well. We also noted that during some times it was able to process more than 10 posts/minute efficiently (including response generation). The average response time is around 10~30 seconds. It is much better in comparison this with the 12 minutes to process 100 messages using FASTUS (8 posts/minute, and 36 hours to process 100 messages (more than 3 hours/post) using TACITUS [28].

VIII. CONCLUSION

We have shown in this paper, by surveying some e-commerce systems, that none of them handles spontaneous users' requests online. The hypothesis that it is necessary and possible to build (multilingual) NL-based e-commerce systems with mixed sublanguage and content-oriented methods has been verified by building CATS. We first studied the classified ads sublanguage to determine the linguistic features and the domain knowledge, both are essential in determining the adequate NL processing method.

To enable semantic processing, CRL-CATS was defined to capture the meaning of a classified ad post. The semantic grammars of content extraction are coded using the EnCo. Alight-weight ontology was implemented in the QA

We have shown that CATS is not like other experimental NL systems, because it was designed from the beginning to be a *production system*.

CATS is currently deployed in Jordan by the largest mobile operator (Fastlink) after passing intensive testing by its services. Testing the content extraction component with a real noisy free text shows a 90% F-measure. The average response time is around 10~30 seconds calculated during peak time (10 posts/minute).

The corpus produced by CATS is unique and can be exploited in building spontaneous NLP systems. Additionally, we can explore different methods to build similar systems. We can also explore other techniques for enhancing the quality of CATS. As an example, we can test the use of spell-checkers to handle spelling variations in these types of spontaneous inputs and measure the effects of this approach on quality and to

check for any performance tradeoff. Additionally, we would like to enhance CE in general. We think this can be achieved with the help of the corpus produced by CATS.

We also plan to port CATS to other domains and other languages. Furthermore, CATS is being investigated for multilinguality by exploring different approaches the localization of similar applications. This work is part of research currently conducted at the GETALP group of LIG.

REFERENCES

[1] MKBEEM, "(web site)," 2005.

[2] J. Heinecke and F. Toumani, "A Natural Language Mediation System for E-Commerce applications: an ontology-based approach," presented at Workshop on Human Language Technology for the Semantic Web and Web Services,2nd International Semantic Web Conference, Sanibel Island, Florida, 2003.

[3] A. Lehtola, Y. KÄPYLÄ, C. BOUNSAYTHIP, and M. TALLGREN, "Multilingual and Ontological Product Cataloguing Tool – User Experiences," presented at eChallenges-2003 - Building the Knowledge Economy: Issues, Applications, Case Studies., Bologna, Italy, 2003.

[4] A. Lehtola, C. Bounsaythip, and J. Tenni, "Controlled Language Technology in Multilingual User Interfaces," presented at 4th ERCIM Workshop on "User Interfaces for All" Special Theme "Towards an Accessible Web", Stockholm, Sweden, 1998.

[5] A. Lehtola, J. Tenni, C. Bounsaythip, and a. K. Jaaranen, "WEBTRAN: A Controlled Language Machine Translation System for Building Multilingual Services on Internet ", vol. 2006, 1999.

[6] P. Buitelaar, K. Netter, and F. Xu, "Integrating Different Strategies for Cross-Language Information Retrieval in the MIETTA Project," presented at TWLT14, Enschede, Netherlands, 1998.

[7] H. Somers, B. Black, J. Nivre, T. Lager, A. Multari, L. Gilardoni, J. Ellman, and A. Rogers, "Multilingual Generation and Summarization of Job Adverts: the TREE Project," presented at Fifth Conference on Applied Natural Language Processing, Washington, DC, 1997.

[8] J. Chai, V. Horvath, N. Nicolov, M. Stys, N. Kambhatla, W. Zadrozny, and P. Melville, "Natural Language Assistant - A Dialog System for Online Product Recommendation," AI Magazine, vol. 23, pp. 63–75, 2002.

[9] X. Gao and L. Sterling, "Classified Advertisement Search Agent (CASA): a knowledge-based information agent for searching semi-structured text." presented at Third International Conference on the Practical Application of Intelligent Agents and Multi-Agent Technology, 1998.

[10] R. I. Kittredge:, "Sublanguages," American Journal of Computational Linguistics, vol. 8, pp. 79-84, 1982.

[11] F. Benamara, "Cooperative Question Answering in Restricted Domains: the WEBCOOP Experiment," presented at ACL'04 Workshop on Question Answering in Restricted Domains, Barcelona, 2004.

[12] D. Moldovan, C. Clark, S. Harabagiu, and S. Maiorano, "COGEX: A Logic Prover for Question Answering," presented at HLT-NAACL 2003, Edmonton, 2003.

[13] S. Sekine, "The Domain Dependence of Parsing", presented at Applied Natural Language Processing (ANLP'97), Washington D.C., USA, 1997.

[14] J. Lehrberger, "Automatic Translation and the Concept of Sublanguage," in Sublanguage: Studies of Language in Restricted Semantic Domains, R. Kittredge and J. Lehrberger, Eds. Berlin & New York: Walter de Gruyter, 1982, pp. 81-106.

[15] A. Goweder and A. De Roeck, "Assessment of a significant Arabic corpus," presented at Arabic NLP Workshop at ACL/EACL 2001, Toulouse, France, 2001

.

Study of Example Based English to Sanskrit Machine Translation

Vimal Mishra and R. B. Mishra

***Abstract*—Example based machine translation (EBMT) has emerged as one of the most versatile, computationally simple and accurate approaches for machine translation in comparison to rule based machine translation (RBMT) and statistical based machine translation (SBMT). In this paper, a comparative view of EBMT and RBMT is presented on the basis of some specific features. This paper describes the various research efforts on Example based machine translation and shows the various approaches and problems of EBMT. Salient features of Sanskrit grammar and the comparative view of Sanskrit and English are presented. The basic objective of this paper is to show with illustrative examples the divergence between Sanskrit and English languages which can be considered as representing the divergences between the order free and SVO (Subject-Verb-Object) classes of languages. Another aspect is to illustrate the different types of adaptation mechanism.**

***Index Terms*—Example based machine translation, Devnagari, language divergence, matching.**

I. Introduction

THE Example Based Machine Translation (EBMT) is one of the most popular machine translation mechanisms which retrieve similar examples with their translation from the example data base and adapting the examples to translate a new source text. The origin of EBMT can be dated precisely to a paper by Nagao (1984). He has called the method "Translation by Analogy". The basic units of EBMT are sequences of words (phrases) and the basic techniques are the matching of input sentence (or phrases) with source example; phrase from the data base and the extraction of corresponding phrase from the data base and the extraction of corresponding translation (translation phrase) and the "recombination" of the phrases as acceptable translation sentences. It is defined on the basis of data used in translation process, and it is not enough to say that EBMT is "data driven" in contrast to "theory-driven" RBMT and that EBMT is "symbolic" in contrast to "non symbolic" SMT (John Hutchins, 2005). The emphasis is not on what matters but it is .how the data are used in translation operations (Turcato & Popowich, 1999).

Knowledge Driven Generalized EBMT system has been used which translates short single paragraph from English to Bengali (S. Bandyopadhyay, 2001). Headlines are translated using knowledge bases and example structures, while the sentences in the news body are translated by analysis and synthesis. In translation of news headlines, the various phrases in the source language and their corresponding translation in the target language are stored. The translations for the headlines are first searched in the table, organized under each headlines structure, containing specific source and target language pairs. If the headlines still can not be translated, syntax directed translation technique are applied. It matches with any phrase of a sentence structure and the bilingual dictionaries. Otherwise, word by word translation is attempted. The knowledge bases includes the suffix table for morphological analysis of English surface level words, parsing table for syntactic analysis of English, bilingual dictionaries for different classes of proper nouns, different dictionaries, different tables for synthesis in the target language.

One of the most remarkable basis that differentiate EBMT among RBMT and SMT is that the basic processes of EBMT are analogy-based , that is the search for phrases in the data base which are similar to input source language (SL) strings, their adaptation and recombination as target language (TL) phrases and sentences (Sumita *et al*.,1990). Neither RBMT nor SMT seek "similar" strings; both search for "exact" matches of input words and strings and produce sequence of words and strings as output. Thus, EBMT is analogy based MT while SMT is correlation based MT.

We have divided this paper into seven sections. Apart from introduction in section 1 the remaining sections are as follows. Section 2 discusses different approaches of EBMT like Foundation based approach, Run time approach, Template-Driven approach and Derivation based approach and then, we compare EBMT and RBMT (Rule Based Machine Translation) on basis of computational cost, improvement cost, system building cost, context-sensitive translation, robustness, measurement of reliability factor and example independency. Section 3 describes Sanskrit grammar, gives comparative view of English and Sanskrit language and discusses some previous work done on Sanskrit. Section 4 discusses different problems that occur in English to Sanskrit translation using EBMT. Section 5 covers different types of language divergences between English and Sanskrit. Section 6 discusses different adaptation technique used in EBMT

Manuscript received February 28, 2008. Manuscript accepted for publication June 13, 2008.

Vimal Mishra is with the Department of Computer Engineering, Institute of Technology, Banaras Hindu University (I.T.-BHU), Varanasi, India-221005 (Phone: +91-9415457592; e-mail: vimal.mishra.upte@gmail.com, vimal.mishra.cse07@itbhu.ac.in)

R. B. Mishra is with the Department of Computer Engineering, Institute of Technology, Banaras Hindu University (I.T.-BHU), Varanasi, India-221005 (e-mail: ravibm@bhu.ac.in).

system. Section 7 gives implementation steps to achieve the translation from English to Sanskrit. Section 8 draws the conclusions.

II. Approaches using EBMT and their Comparison

The existing EBMT system uses different approaches like Foundation based approach, Run time approach, Template-Driven approach and Derivation based approach. Then, we compare EBMT with RBMT as both are close to each other on some issues.

A. Approaches using EBMT

We can classify approaches that use EBMT into four categories as shown in figure 1 that are presented from the least rule based to the most rule based approach (John Hutchins, 2005).

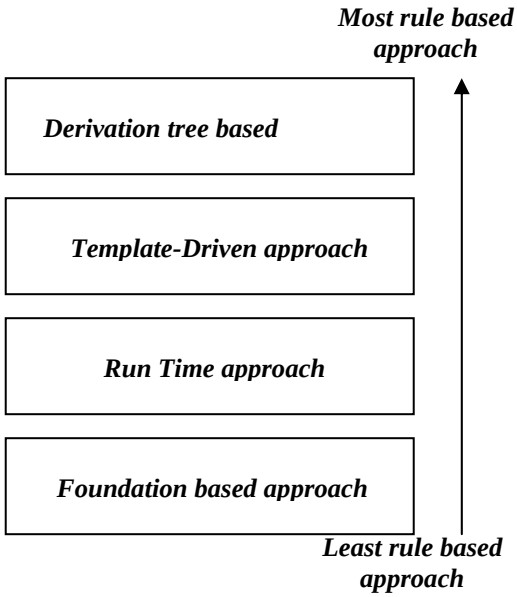


Fig. 1. Approaches based on EBMT

In Foundation based approach based on EBMT, the true EBMT systems are those where the information is not preprocessed, it is available and unanalyzed throughout the matching and execution processes.

In Run time approach using EBMT, (Planas & Furuse, 1999) EBMT uses a method of fuzzy matching involving superficial lemmatization and shallow parsing while E.Sumita *et al*. (1990) describe a full run time EBMT system that uses dynamic programming matching and thesauri for calculating semantic distances and illustrated by Japanese-English translation (at ATR in Japan).

In Template-Driven EBMT, methods of building templates from bilingual example corpora in advance of translation processes are used. Ilyas Cicekli & Altay Guvenir (1996) use templates in the form of words or lemmas with POS tags for a system with English as SL and Turkish as TL while Ralf Brown (2005) describes the induction of transfer rules in the form of templates of word strings, which are then either interpreted as rules of a transfer grammar or added as new examples to the original corpus.

Derivation trees approach of EBMT are devoted to the precompiled preparation of templates with more structure. Kaory Yamamoto & Yuji Matsumoto (1995) describe two studies extracting knowledge from an English-Japanese parallel corpus of business texts. The first study describes that word and phrase correspondences are derived using a statistical dependency parser and three variants are evaluated. The second study compares the statistical dependency model with methods using word segmentation (plain n-gram) and "chunk" boundaries; it is concluded that this method is most useful for preparing bilingual dictionaries in new domains(particularly for identifying compound nouns) while statistical dependency is most useful for disambiguation.

B. Comparison between EBMT and RBMT

We compare EBMT with RBMT on the different basis that shows the feature of EBMT which RBMT lacks as below in table I.

TABLE I
Comparison between EBMT and RBMT

Basis	EBMT	RBMT
Computational Cost	Low	High
Improvement Cost	Low	High
System Building Cost	Low	High
Context-Sensitive Translation	General architecture incorporating contextual information into example representation provides a way to translate context sensitively.	Needs another understanding device in order to translate context sensitively.
Robustness	Low; EBMT works on best match reasoning.	High; works on exact match reasoning.
Measurement of reliability factor	Yes; a reliability factor is assigned to the translation result according to the distance between input and retrieved similar example.	No; RBMT has no device to compute the reliability of the result.
Example Independency	Yes; knowledge is completely independent of the system, is usable in other system.	No; specific to a particular system.

III. Comparison of English and Sanskrit Grammar

English is well known language so we illustrate Sanskrit grammar and its salient features. The English sentence always has an order of Subject-Verb-Object, while Sanskrit sentence has a free word order. A free order language is a natural language which does not lead to any absurdity or ambiguity, thereby maintaining a grammatical and semantic meaning for every sentence obtained by the change in the ordering of the words in the original sentence. For example, the order of English sentence (ES) and its equivalent translation in Sanskrit sentence (SS) is given as below.

ES:	*Ram*	*reads*	*book.*
	(Subject)	*(Verb)*	*(Object)*
SS:	*Raamah*	*pustakam*	*pathati.*
	(Subject)	*(Object)*	*(Verb) ;* or
	Pustakam	*raamah*	*pathati.*
	(Object)	*(Subject)*	*(Verb) ;* or
	Pathati	*pustakam*	*raamah*
	(Verb)	*(Object)*	*(Subject)*

Thus Sanskrit sentence can be written using SVO, SOV and VOS order.

A. Alphabet

The alphabet, in which Sanskrit is written, is called Devnagari. The English language has twenty-six characters in its alphabet while Sanskrit has forty-two character or *varanas* in its alphabet. The English have five vowels (a, e, i, o and u) and twenty one consonants while Sanskrit have nine vowels or *swaras* (a, aa, i, ii, u, uu, re, ree and le) and thirty three consonants or *vyanjanas*. These express nearly every gradation of sound and every letter stands for a particular and invariable sound. The nine primary vowel consists of five simple vowel viz. a, i, u, re and le. The vowels are divided into two groups; short vowels: a, i, u, re and le and long vowels: aa, ii, uu, ree, lee, e, ai, o and au. Thus the vowels are usually given as thirteen. Each of these vowels may be again of two kinds: *anunasik* or nasalized and *ananunasik* or without a nasal sound. Vowels are also further discriminated into *udanta* or acute, *anudanta* or grave and *swarit* or circumflex. *Udanta* is that which proceeds from the upper part of the vocal organs. *Anudanta* is that which proceeds from their lower part while *Swarit* arises out of a mixture of these two. The consonants are divided into *sparsa* or mutes (those involving a complete closure or contact and not an approximate one of the organs of pronunciation), *antasuna* or intermediate (the semivowels) and *ilshman* or sibilants. The Consonants are represented by thirty three syllabic signs with five classes arranged as below.

(a) Mutes: (1) Kavarga: k, kh, g, gh, nn.
(2) Chavarga: ca, ch, j, jh, ni.
(3) Tavarga: t, th, d, dh, ne.
(4) Tavarga: t, th, d, dh, n.
(5) Pavarga: p, ph, b, bh, m.
(b) Semivowels: y, r, l, v.
(c) Sibilants: ss, sh, s.

The first two letters of the five classes and the sibilants are called surds or hard consonants. The rest are called sonants or soft consonants.

In Sanskrit, there are two nasal sounds: the one called *anuswara* and the other called *anunasika*. A sort of hard breathing is known as *visarga*. It is denoted by a sepcial sign: *a swara* or vowel is that which can be pronounced without the help of any other letter. A *vyanjana* or consonant is that which is pronounced with the help of a vowel.

B. Noun

According to Paninian grammar, declension or the inflections of the nouns, substantive and adjectives are derived using well defined principles and rules. The crude form of a noun (any declinable word) not yet inflected is technically called a *pratipadikai*.

C. *Gender*

Any noun has three genders: masculine, feminine, and neuter; three numbers: singular, dual, and plural. The singular number denotes one, the dual two and the plural three or more. The English language has two numbers: singular and plural, where singular denotes one and plural denotes two or more. There exist eight classifications in each number (grammar cases): nominative, vocative, accusative, instrumental, dative, ablative, genitive and locative. These express nearly all the relations between the words in a sentence, which in English are expressed using prepositions. Noun has various forms: *akAranta, AkAranta, ikAranta, IkAranta, nkAranta* and *makAranta*. Each of these *kaarakas*, have different inflections arising from which gender they correspond to. Thus, *akAranta* has different masculine and neuter declensions, *AkAranta* has masculine and feminine declensions, *ikAranta* has masculine, feminine and neuter declensions and *IkAranta* has masculine and feminine forms.

D. Pronoun

According to Paninian Grammar and investigations of M. R. Kale, Sanskrit has 35 pronouns. These pronouns have been classified into nine classes. Each of these pronouns has different classes as personal, demonstrative, relative, interrogative, reflexive, indefinitive, correlative, reciprocal and possessive. Each of these pronouns has different inflectional forms arising from different declensions of the masculine and the feminine form.

E. Adverb

Adverbs are either primitive or derived from noun, pronouns or numerals.

F. Particle

The particles are either used as expletives or intensive. In Sanskrit, particles do not possess any inflectional suffix, for example, *trata saa pathati*. Here, the word *trata* is a particle which has no suffix, yet the word *trata* implies the meaning of the seventh inflection.

G. Verb

There are two kinds of verbs in Sanskrit: primitive and derivative. There are six tenses (*Kaalaa*) and four moods

(*Arthaa*). The tenses are as present, aorist, imperfect, perfect, first future, and second future. The moods are as imperative, potential, benedictive and conditional. The ten tenses and moods are technically called the ten *Lakaras* in Sanskrit grammar.

H. Voice

There are three voices: the active voice, the passive voice and the impersonal construction. Each verb in Sanskrit, whether it is primitive or derivative, may be conjugated in the ten tenses and moods. Transitive verbs are conjugated in the active and passive voices and intransitive verbs in the active and the impersonal form. In each tense and mood, there are three numbers: singular, dual and plural with three persons in each.

I. Comparative View of English and Sanskrit

We describe comparative views of English and Sanskrit on different basis as below in table II.

TABLE II
COMPARATIVE VIEWS OF ENGLISH AND SANSKRIT

Basis	English	Sanskrit
Alphabet	26 character	42 character
Number of vowel	Five vowels	Nine vowels
Number of consonant	Twenty one consonant	Thirty three consonant
Number	Two: singular and plural	Three: singular, dual and plural
Sentence Order	SVO (Subject-Verb-Object)	Free word order
Tenses	Three: present, past and future	Six: present, aorist, imperfect, perfect. 1st future and 2nd future
Verb Mood	Five: indicative, imperative, interrogative, conditional and subjunctive	Four: imperative, potential, benedictive and conditional

Some previous works on Sanskrit are described below.

P. Ramanujan (1992) discusses the computer processing of the Sanskrit. Automatic morphological analysis should be performed. He also discusses syntactic, semantic and contextual analyses of Sanskrit sentence. In Sanskrit, words are composed of two parts: a fixed base part and a variable affix part. The variable part modifies the meaning of the word base, depending on a set of given relationships. The processes of declensions are properly defined. The Sanskrit is based on nominal stems, verbal stems and affixes. All available verbal stems are divided into ten specific classes (the Gana patha record groups of nominal stems, which undergo specific grammatical operations). There are 21 archetypal affixes for nominal declensions (denoted by '*sup'*) and 18 for verbs (denoted by '*tin'*). This is devised for the ending, gender etc. for noun (*subantas*) and for class (*gana*) a usage (*padi*). A nominal lexicon is then chosen, to cover all the allomorphic forms. The *Dhatupatha* is codified as verbal root lexicon. In semantic analysis, there are six functors, viz. Agent (5 types: independent doer causative agent, object agent (reflexive), expressed and unexpressed), object (7 types: accomplished, evolved, attained, desired, undesired, desired-undesired and agent-object), Instrument (2 types: internal and external), Recipient or Beneficiary (3 types: impelled, ascenting and non-refusing object. P. Ramanujan has developed a Sanskrit parser 'DESIKA', which is the analysis program based on Paninian grammar. DESIKA includes Vedic processing as well. In DESIKA, these are separate modules for the three functions of the system: generation, analysis and reference. Generation of nominal or verbal class of word is carried out by the user specifying the word and the applicable rules being activated. In analysis, the syntactic identification and assignment of functional roles for every word is carried out using the *Karaka-vibhakti* mappings. In the reference module, a complete 'trace' of the process of generation or analysis is planned to be provided, besides information or help. The DESIKA parser can be used by taking from the web http://www.tdil.mit.gov.in/download/desika.htm.

Rick Briggs (1985) uses semantic nets (knowledge representation scheme) to analyze sentences unambiguously. He compares the similarity between English and Sanskrit and the theoretical implications of this equivalence are given. In semantic nets, presentation of natural language object and subject is described in form of nodes, while relationship between them is described by edges. The meaning of the verb is said to be both *Vyapara* (action, activity, cause) and *Phulu* (fruit, result, effect). Syntactically, its meaning is invariably linked with the meaning of the verb "to do". All verbs have certain suffixes that express either the tense or mode or both, the person(s) engaged in the "action" and the number of persons or items so engaged.

IV. PROBLEMS IN ENGLISH TO SANSKRIT TRANSLATION USING EBMT

There are the following problems when we use the example based approaches to machine translation (Somer, 1999).

A. Parallel Corpora

EBMT is a corpus based MT, so this requires a parallel aligned corpus. The sources of machine readable parallel corpora are own parallel corpus of researchers, public domain parallel corpora. The EBMT system is generally to be best suited to a sublanguage approach and an existing corpus of translations can serve to define implicitly the sublanguage which the system can handle. When we use parallel aligned corpus from public domain, then the problem of sublanguage can arise. The parallel corpus, which is good enough, is quite difficult to get, especially for typologically different languages or for those languages that do not share the same writing system, such as English and Sanskrit. The alignment problem

of parallel corpus can be avoided by building the example database manually.

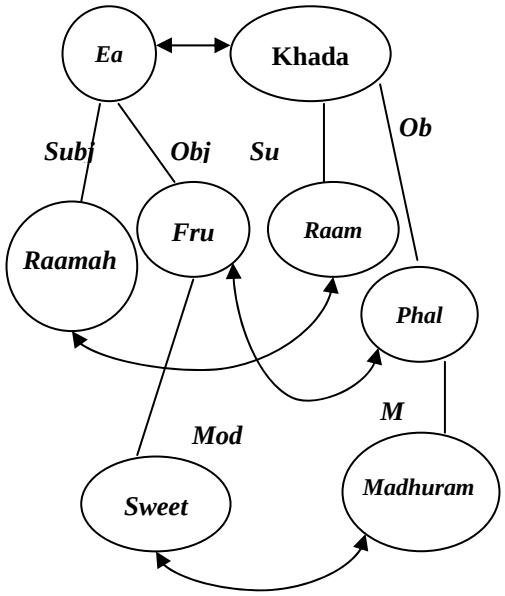


Fig. 2. Representation for English (E) and Sanskrit (S)

B. Granularity of Examples

The longer the matched passes, the probability of a complete match is the lower and the shorter the matched passes, the greater the probability of a complete match (Nirenburg *et al.*, 1993). The obvious and intuitive "grain size" for examples should be the sentence. Although the sentence as a unit for translation, offers the advantage such as sentence boundaries, are for the most part easy to determine.

C. Size of Example Database

There is a question: How many examples are needed in the example database to achieve the best translation result? According to Mima *et al.* (1998) the quality of translation is improved as more examples are added to the database. There is some limit after which further examples do not improve the quality of translation.

D. Suitability of Examples

According to Carl and Hansen (1999), a large corpus of naturally occurring text will contain overlapping examples of two types: (a) some examples will mutually reinforce each other, either by being identical, or by exemplifying the same translation phenomenon. (b) Other examples will be in conflict; the same or similar phrase in one language may have two different translations for no other reasons than inconsistency. According to Murata *et al*. (1999), the suitability of examples are taken by similarity metric, which is sensitive to frequency, so that a large number of similar examples will increase the score given to certain matches.

E. Structure of Examples Database

The structure of database with examples is concerned with storage of examples in the database, which is needed for searching the matches. In the simplest case, the examples may be stored as pairs of strings, with no additional information associated with them. As Somers and Jones (1992) point out, the examples might actually be stored with some kind of contextual manner. There is several structure of examples database of existing EBMT systems such as follows.

F. Annotated Tree Structures

In early EBMT systems, the examples are stored as fully annotated tree structures with explicit links. Figure 2 shows how the English example in E and Sanskrit Translation in S is represented. Similar ideas are found in Watanable (1992), Sato and Nagao (1990), Sadler (1991), Matsumoto *et al*. (1993), Sato (1995), Matsumoto and Kitamura (1995) and Meyers *et al*. (1998).

ES: Ram eats sweet fruits.
SS: Raamah madhuram phalam khaadati.
(Ram) (sweet) (fruits) (eats)

(Al-Adhaileh and Kong, 1999) examples are represented as dependency structures with links at the structural and lexical level expressed by indexes. Figure 3 shows the representation for the English-Sanskrit pair and figure 4 shows translation scheme for "*Shyam runs faster*".

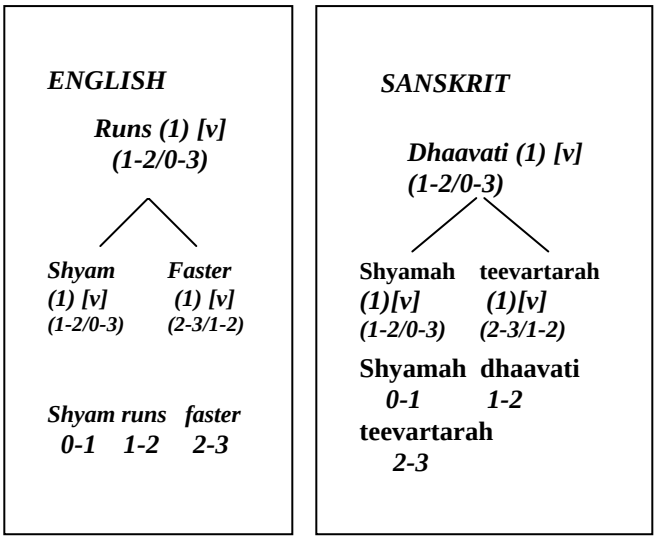


Fig. 3. Representation scheme for "*Shyam runs faster*"

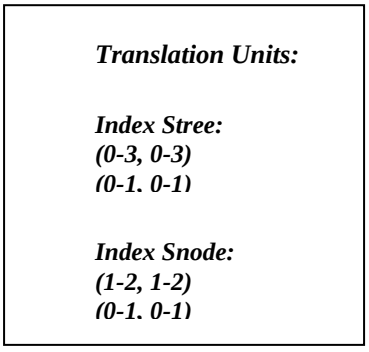


Fig. 4. Translation scheme for "*Shyam runs faster*"

ES: Shyam runs faster.
SS: Shyamah teevartarah dhaavati.
(Shyam) (faster) (runs)

The nodes in the trees are indexed to show the lexical head and span of the tree of which that item is head: so for the example the node labeled "*runs*" (1) [v] (1-2/0-3) indicates that the subtree headed by *runs*, which is the word spanning nodes 1 to 2 (i.e. the second word) is the head of the sub tree spanning nodes 0 to 3, i.e. *Shyam faster*. The labeled "Translation Units" gives the links between the two trees, divided into "Stree" links, identifying subtree correspondences (e.g. the English subtree 1-2 *runs* corresponds to the Sanskrit subtree *dhaavati* 1-2) and "Snode" links, identifying lexical correspondences (e.g. English word 1-2 *runs* corresponds to Sanskrit word 1-2 *dhaavati*).

G. Generalized Examples

In some systems, similar examples are combined and stored as a single "generalized" example. Brown (1999,) for instance, tokenizes the examples to show equivalence classes such as "person's name", "date", "city name" and also linguistic information such as gender and number. In Generalized Examples approach, phrases in the examples are replaced by these tokens, thereby making the examples more general.

H. Statistical Approach

In the statistical approach for structure of examples database, the examples are not stored at all, except in as much as they occur in the corpus on which the system is based (Somers, 1999).

I. Matching

The matching is a process that retrieves the similar examples from example data base. We describe some popular matching approaches below.

J. Character based Matching

The input sentence is matched with example sentence. The matching process involves a distance or similarity measure. When the examples are stored as strings, the measure may be a character-based pattern matching. In the earliest MT systems (ALPS "Repetitions processing" cf. Weaver, 1988), only exact matches of the alphanumeric strings were possible.

K. Word based Matching

Nagao (1984) proposed to use thesauri for indication of words similarity on the basis of meaning or usage. A thesaurus provides a listing of synonyms, allowing examples to match the input, on condition that they can be classified as synonyms based on a measurement of similarity. The examples in (1) and their translations in (2) (Nagao, 1984) show how this technique can be used successfully in choosing between conflicting examples.

(1) *(a) ES: A man eats vegetables.*
SS: Narah shaakam khaadati.
(A) (man) (vegetables) (eats)
(b) ES: Acids eats metal.
SS: Aambat dhaatum nashyati.
(Acids) (metal) (eats)

(2) *(a) ES: He eats potatoes.*
SS: Sah sukantham khaadati.
(He) (potatoes) (eats)
(b) ES: Sulphuric acid eats iron.
SS: Gandhak lauham nashyati.
(Sulphuric acid) (iron) (eats)

In 2 (a), the correct translation of *eats* (from Sanskrit translation SS) is chosen. This is correct in this instance as it refers to food and is chosen because of the relative similarity or distance between potatoes and vegetables.

L. Structure based Matching

In the earlier proposals for EBMT, it is assumed that the examples would be stored as structured objects, so the process involves a rather more complex tree-matching (e.g., Maruyama and Watanable 1992, Matsumoto *et al*. 1993, Watanable 1995, Al-Adhaileh and Tang 1999).

M. Annotated Word-based Matching

When we analyze both the input sentence and the examples to measure the similarity among them, then Annotated Word-Based Matching can be applied. Cranias *et al*. (1994, 1997) takes the function words for similarity measurement and makes use of POS tags. Veale and Way (1997) use sets of closed-class words to segment the examples which is said to be based on the "Marker Hypothesis" from psycholinguistics (Green, 1979).

N. Carroll's "Angle of similarity"

Carroll (1990) suggests the concept of an angle of similarity as a measure of distance between input sentence and the example sentence. This angle is calculated using a triangle whose three points represent the two sentences being compared and a 'null sentence'. The length of sides from this null point to the points representing the two sentences are the respective sizes of those sentences and the length of the third size is the difference between the two. The size of a sentence is calculated by costing the addition, deletion and replace operations necessary to derive one sentence from the other using costs from a set of 'rules' embodied in the system. We compare the given sentence with examples in the database looking for similar words and taking account of three basic operations. The relevance of particular mismatches is referred as "cost".

O. Partial Matching for Coverage

In most of the matching process, the aim is to find a single example or a set of individual examples that provide the best match for the input. In Nirenburg *et al* (1993), Somers *et al*. (1994) and Collins (1998), the matching process decomposes the cases and makes a collection of using terminology as "substring", "fragments" or "chunks" of the matched material. In these matching processes, the recombination process is needed for generating the target text (Jones, 1992: 165). If the dataset of examples is regarded as not a static set of discrete entities but a permutable and flexible interactive set of process modules, we can envisage a control architecture, where each process (example) attempts to close itself with respect to (parts of) the input.

P. Dynamic Programming Matching

Sumita (2003) applies an algorithm based on dynamic programming (DP) matching between word sequences for a speech to speech translation system. DP technique provides optimal solutions to specific problems by making decision at discrete time stages. At each stage, a small number of finite options are possible. Decisions are made, based on obtaining the optimal path from the input sentence to an example sentence. In Summit's approach, retrieval of examples is based on the calculation of a distance measure between the input and the example sentences. This distance measure is a normalized score of the sum of substitution, deletion and insertion operations. Once a similar example has been detected, the next step is to formulate a translation pattern from this example. These patterns are created dynamically and are not retained or stored for use in future translation.

Gelbukh and Sidorov (2006) show that dynamic programming gives least-cost hyper graph to formalize the paragraph alignment task in bilingual text such as English and Spanish. In formalization of the task, they select the optimal hyper graph out of hyper graphs with different number of arcs. Their algorithm prefers a smaller number of hyper arcs. It uses a (NE+1) (NS+1) chart, where NE and NS are the number of paragraphs in the text of the language English and Spanish, respectively. This algorithm has the complexity O (N^4), where N= NE = NS is the size of the text to be aligned.

Quirk and Menezes (2006) use dynamic programming for the dependency tree let translation that shows the convergence of statistical and example based machine translation. They have scored the head-relative positions of the tree as well as the root elements of the existing candidates. For the target-language model, we must multiply the probabilities of the neighbor words of each candidate. These additional probabilities depend only on a very small amount of information of the candidate. They have shown that dynamic programming does the search space savings, but it is not sufficient to produce a real-time translation system.

Q. Case Based Reasoning Matching

Case Based Reasoning (CBR) applies past cases to solve new problems. Each case contains a description of the problem and a possible solution. The Case–based ReVerb system (Collins, 1998) applies CBR technique to EBMT. In this approach, candidate examples are initially selected on condition that they share n words with the input. From this set, a parsed representation of each example is compared against a parsed representation of the input. This is an attempt to locate a match based on syntactic function. Syntactic function is combined with the additional parameters of sentence position and lexical equivalences. Where more than one match has been retrieved at this stage, matches are scored in terms of adaptability.

R. Boundary Friction Problem

The boundary friction is the problem of MT, when the same fragment of sentences needs inflections to indicate the grammatical case, such as determiner, adjective or noun. The boundary friction problem is difficult, in the case of language like Sanskrit, due to the fact that there is more than one grammatical inflection to indicate the syntactic function. So, for example, the translation associated with *the handsome boy* extracted, say, from (3), is equally reusable in the sentence (4,a), but it is not equally reusable in the sentence (4,b).

(3) ES: The handsome boy entered the room.
SS: Sundarah baalakah prakoshtam pravesham akarot.
(The) (handsome) (boy) (the) (room) (entered)

(4. a) ES: The handsome boy ate his breakfast.
SS: Sundarah baalakah svalapaahaaram agarhanaat.
(The) (handsome) (boy) (his) (breakfast) (ate)

(4. b) ES: I saw the handsome boy.
SS: Aaham sundaram baalakam apashyam.
(I) (the) (handsome) (boy) (saw)

S. Computational Problem

All the approaches of EBMT systems have to be implemented as software and significant computational factors influence many of them. One problem of such approaches, which stores the examples as complex annotated structures, is the huge computational cost in terms of creation, storage and matching or retrieval algorithms. This situation is problematic if such resources are difficult to obtain for one or both of the languages, as Guvenir and Cicekli (1998) report. Another problem of EBMT comes in picture when we extend the system's linguistic knowledge by increasing the size of example set (cf. Sato and Nagao, 1990:252). Adding more examples to the existing example database involves a significant overhead if these examples must be parsed and the resulting representations possibly checked by human. The next problem of EBMT is computational speed, especially for those of the EBMT systems that are used for real-time speech translation, which is solved by using "massively parallel processors".

V. Language Divergence between English and Sanskrit

Divergence is a common problem in translation between two natural languages. Language divergence (Dorr, 1993; Dave *et al*, 2001) occurs, when lexically and syntactically similar sentences of the source language are not translated into sentences that are similar in lexical and syntactic structure in the target language.

For example, consider the following English sentences and their Sanskrit translations:

(A) ES: She is in love.
SS: Saa madanesu asti.
(She) (love)(in) (is)

(B) ES: She is in train.
SS: Saa vaashpshakateshu asti.
(she) (train)(in) (is)

(C) ES: She is in fear.
SS: Saa vibheti.
(She) (is in fear)

Items (A) and (B) are examples of normal translation pattern. The prepositional phrases (PP) of the English sentences are similar to PP in Sanskrit though the prepositions occur after the corresponding noun in accordance with the Sanskrit syntax. Still example (C) has a structural variation.

The prepositional phrase "is in fear" is translated by the verb "*vibheti*". This is an instance of a translation divergence.

We have considered that if the English sentence in (A) is given as the input to English to Sanskrit Example Base Machine Translation (EBMT) system, then two cases may arise:

1. The retrieved example is B, i.e., "She is in train". In this case, the correct Sanskrit translation may be generated simply by using word replacement operation to replace "*vaashpshakateshu*" with "*madanesu*".
2. If example (C) is retrieved for adaptation, the generated translation may be "*Saa* (she) *madaneshati* (love) (in) (is)", which is syntactically incorrect Sanskrit sentence. So, the output of the system will depend entirely on the sentence (B), which will be retrieved to generate the translation of the input (A). We see that when we take example C to generate the translation of the input A, which gives us a syntactically incorrect Sanskrit sentence. This is due to the presence of divergence in the translation of example (C). Identification of divergence must be considered paramount for an EBMT system. So, an algorithm must be used in partitioning the example base into two parts: (i) divergence example base and (ii) normal example base.

This will help in efficient retrieval of past examples which improves the performance of an EBMT system.

VI. Divergence and its Identification: Some Relevant Previous Work

There are several approaches that deal translation divergence. We discuss some of them below.

A. Transfer Approach

In the transfer approach of translation divergence, there is transfer rule for transforming a source language (SL) sentence into target language (TL), by performing lexical and structural manipulations. These transfer rules are formed in several ways:

(i) With manual encoding (Han *et al*., 2000) and
(ii) With analysis of parsed aligned bilingual corpora (Watanable *et al*, 2000).

B. Interlingua Approach

In the interlingua approach, the identification and resolution of divergence are based on two mappings GLR (Generalized Linking Routine), CSR (Canonical Syntactic Realization) and a set of LCS (Lexical Conceptual Structure) parameters. The translation divergence occurs, when there is an exception either to GLR or to CSR (or to both) in one of the languages. This situation permits one to formally define a classification of all possible lexical-semantic divergences that could arise during translation. This approach has been used in the UNITRON system (Dorr, 1993) that performs translation from English to Spanish and English to German.

C. Generation Heavy Machine Translation (GHMT) Approach

The MATADOR System (Habash, 2003) uses this approach for translation between Spanish and English. In this approach, a symbolic overgeneration is created for a target glossed syntactic dependency representation of SL sentences, which uses rich target language resources, such as word-lexical semantics, categorical variations and sub-categorization frames for generating multiple structural variations. This is constrained by a statistical TL model that accounts for possible translation divergences. Then, a statistical extractor is used for extracting a preferred sentence from the word lattice of possibilities. This approach bypasses explicit identification of divergence, and generates translations, which may include divergence sentences otherwise.

D. Universal Networking Language based Approach

In Universal Networking Language (UNL), sentences are represented using hypergraphs with concepts as nodes and relations as directed arcs. A dictionary of UW (Universal Word) is maintained. A divergence is said to occur if the UNL expression generated from the both source and target language analyzer differ in structure. Dave *et al* (2002) proposed UNL approach for English to Hindi machine translation.

Each of the above approaches has problems, when we apply them in English to Sanskrit machine translation. For example, GHMT (Generation Heavy Machine Translation) approach requires rich resources for the target language (here, Sanskrit), which is not available for Sanskrit nowadays. The Interlingua approach requires deep semantic analysis of the sentences and creation of exhaustive set of rules to capture all the lexical and syntactic variation may be problem in English to Sanskrit translation. While in case of UNL based approach, each UW of the dictionary contains deep syntactic, semantic and morphological knowledge about the word. Creation of such UW dictionary for a restricted domain is difficult and rarely happens.

With respect to Sanskrit, the major problem in applying the above approach is that linguistic resources are very scarce for Sanskrit.

We proposes an approach that uses only the functional tags (FT) and syntactic phrasal annotated chunk (SPAC) structures of the source language (SL) and target language (TL) sentences for identification of divergences. In a translation example, a translation divergence occurs when some particular FT upon translation is realized with the help of some other FT in the target language. The occurrence of divergence is identified by comparing different constraints of words in the source and target language sentence.

VII. Divergences and its Identification in English to Sanskrit Translation

Divergence is a language dependent phenomenon, it is not expected that the same set of divergences will occur across all languages. Dorr (1993) classifies divergence in seven broad types, which is lexical-semantic divergences for translating among the European languages, as below.

(i) Structural divergence
(ii) Conflational divergence
(iii) Categorial divergence
(iv) Promotional divergence
(v) Demotional divergence
(vi) Thematic divergence

(vii) Lexical divergence

A. Structural Divergence

A structural divergence is said to have occurred if the object of the English sentence is realized as a noun phrase (NP) but upon translation in Sanskrit it is realized as a prepositional phrase (PP). The following examples illustrate this.

(a) ES: Ram will attend this meeting.
SS : Ramah asyaam sabhaayaam anuvartishyate.
(Ram) (this) (meeting in) (will attend)

(b) ES: Ram married Sita.
SS: Ramah Sitayaa sahpaanigrahanam akarot.
(Ram) (Sita)(with) (married)

(c) ES: Ram will challenge Mohan.
SS: Ramah Mohanam aahanyashyate.
(Ram) (Mohan) (will challenge).

Analysis of above examples gives us the following points with respect to structural divergence, which we use to design the algorithm for identification of structural divergence.

(i) If the main verb of an English sentence is a declension of "be" verb, then the structural divergence cannot occur.
(ii) Structural divergence deals with the objects of both the English sentence and its Sanskrit translation. So, if any one of the two sentences has no objects then structural divergence cannot occur.
(iii) If both sentences have objects, and then SPAC structures are same then also structural divergence does not occur.
(iv) In this situation, structural divergence may occur only if the SPAC of the object of the English sentence is an NP, and the SPAC of the object of the Sanskrit sentence is a PP.

B. Categorial Divergence

If English sentence has subjective complement (SC) or predictive adjustment (PA), then categorical divergence occurs. In the categorical divergence, the SC or PA of the English sentence, upon translation, is realized as the main verb of the Sanskrit sentence. The SC may be noun phrase (NP) or adjective phrase (AdjP) and PA may be prepositional phrase (PP) or adverb in the English sentence. The categorial divergence is concerned with adjectival SCs which upon translation map into noun, verb or PP. In English to Sanskrit translation, depending upon the nature of the SC or PA, the following subtypes of categorial divergence have been identified, which are given below.

(i) Categorial Subtype 1

When the SC of the English sentence is used as an adjective, but upon translation, it is realized as the main verb of the Sanskrit sentence, then this divergence occurs. For example, consider the following sentences given below.

ES: Ram is afraid of lion.
SS: Ramah singhaat vibheti.
(Ram) (of) (lion) (afraid)

The adjective of the English sentence "afraid" is realized in Sanskrit by the verb "*vibh*" meaning "afraid" and "*vibheti*" is it's conjugate form for present indefinite tense, when the subject is first person, singular and masculine in Sanskrit.

(ii) Categorial subtype 2

When the SC is an NP in the English sentence, then after translation the noun part corresponds to the verb of the Sanskrit sentence. This part is realized as an adverb upon translation.

Consider the following sentences given below.

ES: Ram is a regular user of the library.
SS: Ramah pustakaalayasya aharvisham prayogam karoti.
(Ram) (library)(of) (regular) (user)

The word "user", which is a noun, has been used as an SC in the English sentence above. This provides the main verb "*prayogam karoti*" (meaning "to use") of the Sanskrit sentence. The adjective "regular" of the noun "user" is realized as the adverb "*aharvisham*".

(iii) Categorial subtype 3

The adverbial PA of an English sentence is realized as the main verb of the Sanskrit sentence, for example,

ES: The fan is on.
SS: Vyajanam chalati.
(fan) (move) (ing) (is)

The main verb of the Sanskrit is "*chal*" i.e. "to move". Its sense comes from the adverbial PA "on" of the English sentence. The present continuous form of this verb is "*chalati*", when the subject is third person, singular and masculine in Sanskrit.

(iv) Categorial subtype 4

The PA that is realized in English as PP, but PA is realized in Sanskrit as the main verb. For example, consider the following sentences given below.

ES: The train is in motion.
SS: Railyaanam chalatii.
(train) (move) (ing) (is)

The PA "in motion" is a preposition phrase which sense is realized by the verb "*chal*". In Sanskrit translation, the present continuous form of this verb is "*chalati*", because the subject of the sentence is feminine and singular. After the analysis of these translation examples, we get the following cases related to above mentioned ones..

(i) Categorial divergence occurs if the main verb of the English sentence is a declension of "be" but the main verb of the Sanskrit translation is not the "be" verb.
(ii) Categorial divergence occurs if the Sanskrit translation does not have any subjective complement or PA.
(iii) If SPAC structure of the SC of English sentence is an AdjP or NP then categorial divergence will be of subtype 1 or 2, respectively.
(iv) If SPAC structure of PA of English sentence is AdvP or PP then categorial divergence will be of subtype 3 or 4, respectively.

C. Nominal Divergence

Nominal divergence is concerned with the subject of the English sentence. After translation, the subject of the English sentence becomes the object or verb complement. This nominal divergence is similar to thematic divergence of Dorr (1993).

The subject of the English sentence is realized in Sanskrit with the help of a prepositional phrase. We define two subtypes of nominal divergence as below.

(i) Nominal subtype 1

The subject of the English sentence becomes object upon Translation. For example, consider the following sentences.

ES: Ram is feeling hungry.
SS: Raamen ksudhitaa anubhuuyate.
(To Ram) (hunger) (feeling) (is)

The adjective "hungry" is an SC. Its sense is realized in Sanskrit by the word "*ksudhita*" that acts as the subject of the Sanskrit sentence. The subject "Ram" of the English sentence becomes the object "*Raamen*" (to Ram) of the Sanskrit translation.

(ii) Nominal subtype 2

The subject of the English sentence provides a verb complement (VC) in the Sanskrit translation. For example, consider the following sentences below.

ES: This gutter smells foul.
SS: Asmaat jalanirgamaat malinam jighrati.
(This) (gutter)(from) (foul) (smells)

The subject of the English sentence "This gutter" is realized as the modifier "*Asmaat jalanirgamaat*" of the verb "*anubhavati*".

The analysis of above examples gives the following points.

(i) If the English sentence does not have an SC or declension of the "be" verb, then divergence is to be nominal.

(ii) If the SC of English sentence is null and the object is not null in Sanskrit then it is the instance of nominal divergence of subtype 1. If verb complement (VC) is present in Sanskrit then it nominal divergence of subtype 2.

D. Pronominal Divergence

Pronominal divergence occurs if the pronoun "it" is used as the subject in English sentences. The Sanskrit equivalent of "it" is "*edam*". So, the Sanskrit translation of such a sentence should have "*edam*" as the subject of the sentence. For example, consider the following sentences.

ES: It is crying.
SS: Edam krandati.
(It) (is crying)
ES: It is small.
SS: Edam laghu asti
(It) (is small)

E. Demotional Divergence

When the main verb of the English sentence upon translation is demoted to the subjective complement or predicative adjunct of the Sanskrit sentence and the main verb of Sanskrit translation are realized as "be" verb, then demotional divergence occurs. For example, consider the following sentences.

ES: This house belongs to a doctor.
SS: Edam griham ekasya chikitsakasya asti.
(This) (house) (one) (doctor) (of) (is)
ES: This dish feeds four people.
SS: Edam bhojanam chaturthajanebhyah asti.
(this) (dish) (four) (people) (for) (is)

F. Conflational Divergence

The conflational divergence pertains to the main verb of the source language sentence. According to Dorr (1993), the conflational divergence occurs, when some new words are required to be incorporated in the target language sentence in order to convey the proper sense of a verb of the input.

G. Possessional Divergence

The possessional divergence occurs when the verb "have" in the English sentence is used as the main verb. For example, consider the following sentences given below.

ES: Mohan has many enemies.
SS: Mohanasya anekaah shatrvah santi.
(Mohan) (many) (enemies) (has)

VIII. ADAPTATION

After matching and retrieval of a set of examples, with associated translations, the next step in the EBMT systems is to extract from the translations, the appropriate fragments ("alignment" or "adaptation") and combine these fragments so as to produce a grammatical target output, which is called as recombination. These processes are carried out as twofold that is identifying which fragment of the associated translation corresponds to the matched fragments of the source text and recombining these fragments in an appropriate manner. We can illustrate the problem by considering English to Sanskrit translation below.

1. *(a) ES: He buys a notebook.*
   SS: Sah ekaah panjikam krinaati.
   (b) ES: He read a book on Hindi.
   SS: Sah ekam Hindyaam pustakam pathati.
   (c) ES: He buys a book on Hindi.
   SS: Sah ekam Hindyaam pustakam krinaati.

To understand how the relevant elements of (1: a, b) are combined to give (1, c), we must assume a mechanism to extract from them the common elements (underlined here). Then, we have to make the further assumption that they can be simply pasted together as in (1, c) and that this recombination will be appropriate and grammatical.

The need for an efficient systematic adaptation scheme is required for modifying a retrieved example and thus, generating the required translation. Some of major adaptation approaches of an EBMT system are described below.

(1) Veale *et al*. (1997) proposed adaptation in Gaijian via two categories: high-level grafting and key hole surgery. The phrases are handled with high level grafting. In the high level grafting, an entire phrasal segment of the target sentence is replaced with another phrasal segment from a different example. The key hole surgery deals with individual words in an existing target segment of an example. Under the key hole surgery operation, words are replaced to fit the current translation task. For example, suppose the input sentence is "The girl is playing in the lawn", and in the example base, we have the following examples.

(a) The child is playing.
(b) Sita knows that girl.
(c) It is a big lawn.
(d) Shyam studies in the school.

The sentences (a) and (d) will be used for high level grafting. Then key hole surgery will be applied for putting in the translations of the words "lawn" and "girl". These translations will be extracted from (b) and (c).

(2) In Shiri *et al*. (1997), adaptation procedure is based on three steps: finding the difference, replacing the difference and smoothing the output. The differing segments of the input sentence and the source template are identified. The translations of these different segments in the input sentence are produced by rule-based methods and these translated segments are fitted into a translation template. The resulting sentence is then smoothed over by checking for person, and number agreement and inflection mismatches. For example, assume the input sentence and selected templates as below.

SI : A very efficient lady doctor is busy.

ST : A lady doctor is busy.

TT: Ekaa mahilaa chikitsaka kaaryavyagrah asti.

The parsing process shows that "*A very efficient lady doctor*" is a noun phrase and so matches it with "*A lady doctor*" ("*Ekaa mahilaa chikitsaka*"). "*A very efficient lady doctor*" is translated as "*Ekaa bahut yogyah mahilaa chikitsaka*", by rule based noun phrase translation system. This is inserted into TT giving the following TT: *Ekaa bahuyogyah mahilaa chikitsaka kaaryavyagrah asti.*

(3) Collins (1998) proposed the adaptation scheme as ReVerb system. In this, two different cases are considered: Full case adaptation and Partial case adaptation. Full case adaptation is used when a problem is fully covered by the retrieved example and desired translation is created by substitution alone. In Full case adaptation, five scenarios are possible that are SAME, ADAPT, IGNORE, ADAPT_ZERO and IGNORE_ZERO. Partial case adaptation is used when a single unifying example does not exist. In this case, three more operations are required on the top of the above five. These three operations are ADD, DELETE and DELETE_ZERO.

(4) Somers (2001) proposed adaptation scheme that uses case based reasoning (CBR). The simplest of the CBR adaptation method is null adaptation, where no changes are recommended. In a more general situation, various substitution methods (e.g. Reinstantiation, Parameter Adjustment), transformation methods (e.g. Commonsense transformation and model-guided repair) may be applied. For example, suppose the input sentence (I) and the retrieve examples (R).

I: That old woman has come.

R: That old man has come. (vrddhah aagacchat.)

To generate the desired translation of the word "*man*" ("*vrddhah*") is first replaced with the translation of "*woman*" ("*vrddhaah*") in R. This operation is called reinstitution. At this stage, an intermediate translation "*vrddhah aagacchat*" is obtained.

(5) Jain (1995) proposed HEBMT system, in which examples are stored in an abstracted form for determining the structural similarity between the input sentence and the example sentences. The target language sentence is generated using the target pattern of the sentence that has lesser distance with the input sentence. The system substitutes the corresponding translations of syntactic units identified by a finite state machine in the target pattern. Variation in tense of verb and variations due to number, gender etc. are taken care at this stage for generating the appropriate translation. The HEBMT system translates from Hindi to Sanskrit.

Thus in our view, the adaptation procedures employed in different EBMT systems primarily consists of four operations that are given as below.

(i) Copy: Where the same chunk of the retrieved translation example is used in the generated translation.

(ii) Add: Where a new chunk is added in the retrieved translation example.

(iii) Delete: When some chunk of the retrieved example is deleted and

(iv) Replace: Where some chunk of the retrieved example is replaced with a new one to meet the requirement of the current input.

IX. Implementation

Each translation example record in our example base contains morpho-functional tag information for each of the constituent word of the source language (English) sentence, its Sanskrit translation and the root word correspondence. These tags are obtained by the ENGCG parser (http://www.lingsoft.fi/cgi-bin/engcg) for English sentences. The Sanskrit parser is obtained from the Sanskrit heritage site (http://sanskrit.inria.fr/) which is developed by Gerard Huet. The English to Sanskrit on line dictionary is taken from the site (www.dicts.info/dictionary.php?l1=English&l2=Sanskrit).

X. Conclusions

The EBMT is "data driven" in contrast to "theory driven" RBMT, which retrieves similar examples (pairs of source sentences and their translations), adapting the examples to translate a new source sentence. The Example-Based Machine Translation is used in situations, where on-line resources (such as parser, morphological analyzer, rich bilingual dictionary, rich parallel corpora, etc) are scarce. The Sanskrit is free word order language. Thus, we maintain a grammatical and semantic meaning for every sentence obtained by the change in the ordering of the words in the original sentence. The language divergence significantly occurs between English and Sanskrit translation. Suitable illustrations through examples for some popular adaptation approaches have been given. The adaptation processes select the best match of example sentences and suggests the adaptation procedures employed in different EBMT systems primarily consists of four operations: copy, add, delete and replace. The basic objective of the paper is to illustrate with examples the divergence and adaptation mechanism in English to Sanskrit.

References

[1] M. H. Al-Adhaileh and E. K. Kong, "Example-Based Machine Translation Based on the Synchronous SSTC Annotation Schema", Machine Translation Summit VII, Singapore, pp. 244–249, 1999.

[2] S. Bandyopadhyay, "An Example Based MT System in News Items Domain from English to Indian Languages", Machine Translation Review 12, pp 7-10, 2001.

[3] Rick. Briggs, "Knowledge Representation in Sanskrit and Artificial Intelligence", The AI Magazine, pp 33–39, 1985.

[4] R. D. Brown, "Adding Linguistic Knowledge to a Lexical Example-based Translation System", in TMI, pp. 22–32, 1999.

[5] R. Brown, "Context-sensitive retrieval for example-based translation", MT Summit X, Phuket, Thailand, September 16, Proceedings of Second Workshop on Example-Based Machine Translation, pp. 9-15, 2005.

[6] M. Carl, “Inducing Translation Templates for Example-Based Machine Translation”, Machine Translation Summit VII, Singapore, pp. 250–258, 1999.
[7] J. J. Carroll, “Repetitions Processing Using a Metric Space and the Angle of Similarity”, Report No. 90/3, Centre for Computational Linguistics, UMIST, Manchester, England, 1990.
[8] Cicekli and H. A. Güvenir, “Learning Translation Rules From A Bilingual Corpus”, NeMLaP-2: Proceedings of the Second International Conference on New Methods in Language Processing, Ankara, Turkey, pp. 90–97, 1996.
[9] B. Collins, “Example-Based Machine Translation: An Adaptation-Guided Retrieval Approach”, PhD thesis, Trinity College, Dublin, 1998.
[10] L. Cranias, H. Papageorgiou and S. Piperidis, “A Matching Technique in Example-Based Machine Translation”, in Coling, pp. 100–104, 1994.
[11] L.Cranias, H. Papageorgiou and S. Piperidis, “Example Retrieval from a Translation Memory”, Natural Language Engineering 3, 255–277, 1997.
[12] Dave, Sachi, Parikh, Jignashu, Pushpak Bhattacharya, “Interlingua-based English–Hindi Machine Translation and Language Divergence”, Machine Translation, Vol. 16, pp. 251–304, Kluwer Academic Publishers, 2001.
[13] B. J. Dorr, “Machine Translation: A View from the Lexicon”, MIT Press, Cambridge, MA, 1993.
[14] Bonnie J. Dorr, “Machine Translation Divergences: A Formal Description and Proposed Solution”, Association for Computational Linguistics, pp 597-633, 1994.
[15] Nano Gough, “Example-Based Machine Translation using the Marker Hypothesis”, PhD thesis, School of Computing, Dublin, 2005.
[16] T. R. G. Green, “The Necessity of Syntax Markers: Two Experiments with Artificial Languages”, Journal of Verbal Learning and Verbal Behavior 18, 481–496, 1979.
[17] H. A. Guvenir and I. Cicekli, “Learning translation templates from examples”, Information System 23, 353-363, 1998.
[18] Deepa Gupta, “Contributions to English to Hindi Machine Translation Using Example-Based Approach”, PhD thesis, IIT Delhi, 2005.
[19] N. Habash, “Generation-Heavy Hybrid Machine Translation”, PhD thesis, University of Maryland, College Park, 2003.
[20] C. Han, L. Benoit, P. Martha, R. Owen, R. Kittredge, T. Korelsky, N. Kim and M.Kim, “Handling structural divergences and recovering dropped arguments in a Korean to English machine translation system”, Proceedings of the Fourth Conference of the Association for Machine Translation in the Americas, AMTA-2000, Cuernavaca, Mexico, 2000.
[21] J. Hutchins, “Towards a Definition of Example-Based Machine Translation”, In Machine Translation Summit X, Second Workshop on Example-Based Machine Translation, pages 63–70, Phuket, Thailand 2005.
[22] J. Hutchins, “Example-Based Machine Translation: A Review and Commentary”, Machine Translation, Vol. 19, pp 197-211, 2005.
[23] R. Jain, “HEBMT: A Hybrid Example-Based Approach for Machine Translation (Design and Implementation for Hindi to English)”, PhD thesis, I.I.T. Kanpur, 1995.
[24] D. Jones, “Non-hybrid example-based machine translation architectures”, In TMI, pp. 163-171, 1992.
[25] M. R. Kale, “A Higher Sanskrit Grammar”, 4th Ed, Motilal Banarasidas Publishers Pvt. Ltd., 2005.
[26] Macdonnel, “A Sanskrit Grammar for Students”, 3rd Ed, Motilal Banarasidas Publishers Pvt. Ltd, 2003.
[27] Maruyama and H. Watanabe, “Tree Cover Search Algorithm for Example-Based Translation”, in TMI, pp. 173–184, 1992.
[28] Y. Matsumoto, H. Ishimoto and T. Utsuro, “Structural Matching of Parallel Texts”, 31st Annual Meeting of the Association for Computational Linguistics, Columbus, Ohio, pp. 23–30, 1993.
[29] Y. Matsumoto and M. Kitamura, “Acquisition of Translation Rules from Parallel Corpora”, in Mitkov & Nicolov, pp. 405–416, 1995.
[30] Meyers, R. Yangarber, R. Grishman, C. Macleod and A. Moreno-Sandeval, “Deriving Transfer Rules from Dominance-Preserving Alignments”, in Coling-ACL, pp. 843–847, 1998.
[31] H. Mima, H. Iida and O. Furuse, “Simultaneous Interpretation Utilizing Example-based Incremental Transfer”, in Coling-ACL, pp. 855–861, 1998.
[32] M. Murata, Q. Ma, K. Uchimoto and H. Isahara, “An Example-Based Approach to Japanese to English Translation of Tense, Aspect, and Modality”, in TMI, pp. 66–76, 1999.
[33] M. Nagao, “A Framework of a Mechanical Translation between Japanese and English by Analogy Principle”, in A. Elithorn and R. Banerji (eds), Artificial and Human Intelligence, Amsterdam: North-Holland, pp. 173–180, 1984.
[34] Chakradhar Nautiyal, “Vrihad Anuvaad Chandrika”, 4th Ed., Motilal Banarasidas Publishers Pvt. Ltd, 1997.
[35] S. Nirenburg, C. Domashnev and D. J. Grannes, “Two Approaches to Matching in Example-Based Machine Translation”, in TMI, pp. 47–57, 1993.
[36] E. Planas and O. Furuse, “Formalizing Translation Memories”, Machine Tran Summit VII, Singapore, pp. 331–339, 1999.
[37] P. Ramanujan, “Computer Processing Of Sanskrit”, Computer Processing Of Asian Languages CALP-2, IIT Kanpur, 1992.
[38] V. Sadler, “The Textual Knowledge Bank: Design, Construction, Applications”, International Workshop on Fundamental Research for the Future Generation of Natural Language Processing (FGNLP), Kyoto, Japan, pp. 17–32., 1991.
[39] S. Sato and M. Nagao, “Toward Memory-Based Translation”, in Coling, Vol. 3, pp. 247–252, 1990.
[40] S. Sato, “MBT2: A Method for Combining Fragments of Examples in Example-Based Machine Translation”, Artificial Intelligence 75, 31–49, 1995.
[41] S. Shiri, F. Bond and Y. Takhashi, “A Hybrid Rule and Example-Based Method for Machine Translation”, Proceedings of the 4th Natural Language Processing Pacific Rim Symposium: NLPRS-97, Phuket, Thailand, pp. 49-54, 1997.
[42] H. Somers, “Review article: example-based machine translation”, Machine Translation 14 (2), 113-157, 1999.
[43] H. Somers, and D. Jones, “Machine Translation Seen as Interactive Multilingual Text Generation”, Translating and the Computer 13: The Theory and Practice of Machine Translation – A Marriage of Convenience?, London, Aslib, pp. 153–165, 1992.
[44] H. Somers, I. McLean and D. Jones, “Experiments in Multilingual Example-Based Generation”, CSNLP 1994: 3rd Conference on the Cognitive Science of Natural Language Processing, Dublin, Ireland, 1994.
[45] H. Somers, “EBMT seen as case-based reasoning”, MT Summit VIII Workshop on Example-Based Machine Translation, Santiago de Compostela, Spain, pp. 56-65, 2001.
[46] E. Sumita, H. Iida and H. Kohyama, “Translating with Examples: A New Approach to Machine Translation”, in TMI, pp. 203–212, 1990.
[47] E. Sumita, “EBMT Using DP-matching Between Word Sequences”, In Recent Advances in Example-Based Machine Translation, M. Carl and A. Way, Ed., Kluwer Academic Publishers, Dordrecht, The Netherlands, 2003, pp. 189–209.
[48] T. Veale and A. Way, “Gaijin: A Bootstrapping Approach to Example-Based Machine Translation”, International Conference, Recent Advances in Natural Language Processing, Tzigov Chark, Bulgaria, pp. 239–244, 1997.
[49] H. Watanabe, “A Similarity-Driven Transfer System”, in Coling, pp. 770–776, 1992.
[50] H. Watanabe, “A Model of a Bi-Directional Transfer Mechanism Using Rule Combinations”, Machine Translation 10, 269–291, 1995.
[51] H. Watanabe, S. Kurohashi and E. Aramaki, “Finding structural correspondences from bilingual parsed corpus for Corpus-Based Translation”, Proceedings of COLING, Saarbrucken, Germany, 2000.
[52] Weaver, “Two Aspects of Interactive Machine Translation”, in Technology as Translation Strategy, M. Vasconcellos Ed., Binghamton, NY: State University of New York at Binghamton (SUNY), 1988, pp. 116–123.
[53] Alexander Gelbukh and Grigori Sidorov, “Alignment of Paragraphs in Bilingual Texts using Bilingual Dictionaries and Dynamic Programming”, in Lecture Notes in Computer Science, N 4225, ISSN 0302-9743, Springer-Verlag, 2006, pp 824-833.
[54] C. Quirk and A. Menezes, “Dependency tree let translation: the convergence of statistical and example-based machine-translation?”, Journal of Machine Translation, Vol. 20, pp. 43–65, Kluwer Academic Publishers, 2006.

Aberración Óptica

Magdalena Marciano Melchor, María Aurora Molina Vilchis, Juan Carlos Herrera Lozada

***Resumen*—El estudio de las aberraciones ópticas radica en la evaluación de las imágenes que produce un sistema óptico. Este fenómeno se debe a la geometría del sistema. En este artículo se tiene la finalidad de presentar en forma aproximada las ecuaciones analíticas que describen a un frente de onda esférico afectado por aberración "coma" en un sistema óptico con simetría.**

***Palabras clave*—Sistema óptico, aberración.**

OPTIC ABERRATION

***Abstract*—The study of optic aberrations is related to evaluation of the images produced by an optic system. This phenomenon is related to the geometry of the system. In this paper, we present approximate analytical equations that describe the front of the spherical wave affected by the aberration in an optic system with symmetry.**

***Index Terms*—Optic system, aberration.**

I. INTRODUCCIÓN

En la mejora de la calidad de las imágenes de un sistema óptico se ha estudiado el fenómeno de las aberraciones, este se debe a las leyes físicas que producen los rayos de luz y a las imperfecciones geométricas de los sistemas ópticos, entre algunas otras características [1-3]. En este trabajo nos centraremos en una de las "*aberraciones de Seidel"* denominada "*coma*", con la intención de mostrar en forma analítica el efecto que tiene sobre un frente de onda esférico. La expresión analítica en coordenadas cartesianas para la función de aberración "*coma*" está dada por

$$\Delta(x_0, y_0) = C_2 y_0 \left(x_0^2 + y_0^2\right) \tag{1}$$

donde C_2 es una constante.

Manuscrito recibido el 5 de marzo del 2008. Manuscrito aceptado para su publicación el 15 de junio del 2008.

M. Marciano Melchor, Centro de Innovación y Desarrollo Tecnológico en Cómputo del Instituto Politécnico Nacional, México, D. F. (teléfono: 57296000 Ext. 52529; e-mail: mmarciano@ipn.mx).

M. A. Molina Vilchis, Centro de Innovación y Desarrollo Tecnológico en Cómputo del Instituto Politécnico Nacional, México, D. F. (teléfono: 57296000 Ext. 52531; e-mail: mamvilchis@ipn.mx).

J. C. Herrera Lozada, Centro de Innovación y Desarrollo Tecnológico en Cómputo del Instituto Politécnico Nacional, México, D. F. (teléfono: 57296000 Ext. 52512; e-mail: jlozada@ipn.mx).

II. TEORÍA

Supongamos que trabajamos en un sistema óptico simétrico y que inicialmente se tiene un frente de onda esférico de radio R_0 en un tiempo inicial $t = 0$, además un punto arbitrario sobre el frente de onda con coordenadas $x = x_0$, $y = y_0$, $z = z_0 = \pm\sqrt{R_0^2 - x_0^2 - y_0^2}$. Debido a que el caso de aberración esférica ya fue tratado con detalle anteriormente por GSO y MMM [4] se considera que la forma del frente de onda inicial después de un tiempo determinado t y afectado por alguna aberración tiene la expresión de las siguientes ecuaciones:

$$\begin{aligned} x(x_0, y_0, z) &= x_0 + \left(\frac{z - z_0}{z_0}\right)\left\{x_0 - \left[R_0 + \Delta(x_0, y_0)\right]\frac{\partial\Delta(x_0, y_0)}{\partial x_0}\right\} \\ y(x_0, y_0, z) &= y_0 + \left(\frac{z - z_0}{z_0}\right)\left\{y_0 - \left[R_0 + \Delta(x_0, y_0)\right]\frac{\partial\Delta(x_0 y_0)}{\partial y_0}\right\} \\ z(x_0, y_0, z) &= z. \end{aligned} \tag{2}$$

Las ecuaciones (2) son exactas dentro del límite de la óptica geométrica, pero en la mayoría de las aplicaciones se asume que [5]

$$\left|\Delta(x_0, y_0)\right| << R_0, \quad z_0 \cong -R_0, \quad \frac{z}{R_0} << 1. \tag{3}$$

Utilizando estas aproximaciones las ecuaciones (2) se reducen al mapeo

$$\begin{aligned} x(x_0, y_0, z) &= -\frac{z x_0}{R_0} + R_0 \frac{\partial\Delta(x_0, y_0)}{\partial x_0} \\ y(x_0, y_0, z) &= -\frac{z y_0}{R_0} + R_0 \frac{\partial\Delta(x_0, y_0)}{\partial y_0} \\ z(x_0, y_0, z) &= z, \end{aligned} \tag{4}$$

que tienen mayor simplicidad que en la representación de la evolución de un frente de onda esférico en el caso exacto.

III. CÁLCULOS

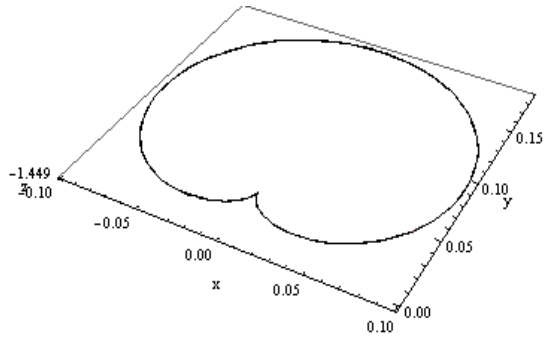
A diferencia del caso esférico descrito en coordenadas cartesianas, aquí haremos un cambio a coordenadas polares en la función de aberración

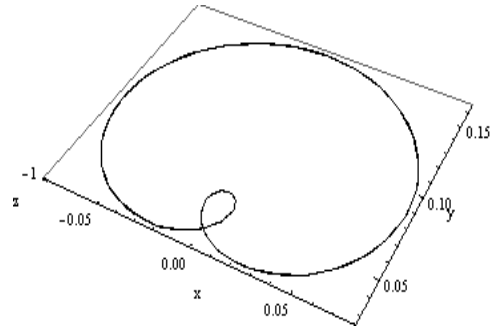
$$\Delta(\rho, \varphi) = C_2 \rho^3 Cos\,\varphi \tag{5}$$

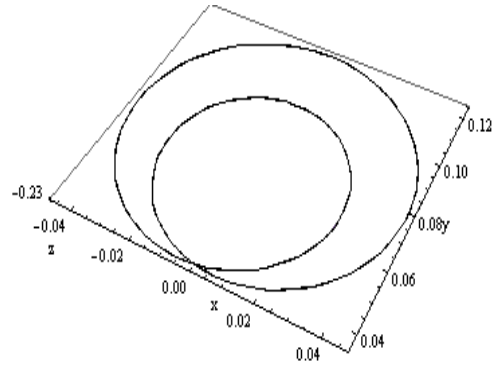
y en las expresiones del mapeo (4) representante de la evolución del frente de onda obteniendo:

$$x(\rho,\varphi,z) = -\left(\frac{z}{R_0}\right)\rho\, Sen\,\varphi + C_2 R_0 \rho^2 Sen\, 2\varphi$$
$$y(\rho,\varphi,z) = -\left(\frac{z}{R_0}\right)\rho\, Cos\,\varphi + C_2 R_0 \rho^2 \left(2 + Cos\, 2\varphi\right) \quad (6)$$
$$z(\rho,\varphi,z) = z,$$

A continuación se representan gráficas de la intersección con planos $z = ons\ tan\ te$ para la superficie del frente de onda en su evolución. Las gráficas fueron elaboradas en un programa de *Mathematica*.







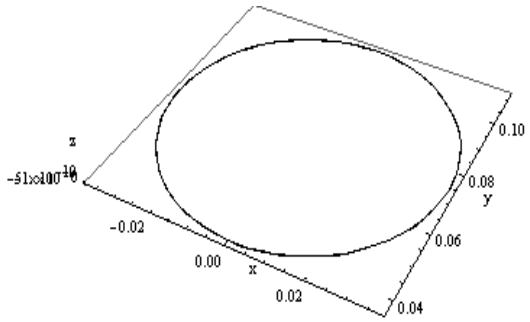


Figura 1. La superficie dada por las ecuaciones (6), cuando $a = 0.124224\ \ m, R_0 = 2.415\ m, C_2 = 1/m^3$ y z variando.

IV. Conclusiones

En este trabajo presentamos la evolución de un frente de onda esférico afectado por *"coma"*, este procedimiento se quiere aplicar para el resto de las *"aberraciones de Seidel"* y con motivaciones importantes, la elaboración de la mejor imagen formada por el sistema óptico. La necesidad de tener una calidad de imagen en los sistemas ópticos hace de este trabajo un desarrollo importante para sentar las bases en los próximos estudios de los sistemas ópticos no simétricos tales como el ojo humano.

Agradecimientos

M. Melchor agradece el apoyo de CIDETEC-IPN y de la SIP-IPN mediante el registro de su proyecto 2008161.

Referencias

[1] M. Born and E. Wolf, *Principles of Optics*. New York: Cambridge, 2006, capítulo 5.
[2] L. N. Thibos, Representation of Wavefront Aberrations, disponible en: http://research.opt.indiana.edu/Library/wavefronts/index.htm
[3] J. C. Wyant and K. Creath, "BasicWavefront Aberration Theory for Optical Metrology," *Applied Optics and Optical Engineering*, vol XI, pp. 1-53.
[4] G. Silva Ortigoza, M. Marciano Melchor, O. Carvente Muñoz and R. Silva Ortigoza, "Exact computation of the caustic associated with the evolution of an aberrated wavefront," *Journal of Optics A: Pure and Applied Optics*, vol. 4, pp. 358-365, 2002.
[5] E. L. O'Neil, "Introduction to Statistical Optics," (Addison-Wesley, Massachusetts, 1963).

Applying Dynamic Causal Mining in Retailing

Yi Wang

***Abstract*—With the fast development of information technology, retailers are suffering from the excess of information. Too much information can be a problem. However, more information creates more opportunity. In retailing, information is the key issue to maximizing revenue. It is now hard to make timely or effective decisions and to the right content to the right place, at the right time and in the right form. This paper is about managing the information so that the user can gain more clear insight. It is about integrating and inventing methods and techniques. The Semantic Web will provide a foundation for such a solution. However, semantics only provide a way of mapping the content of a web to user defined annotations. Not many companies have fully utilized the power of Internet retailing due to the various technical obstacles have yet to be overcome. The existing research in e-retailing focuses only on the traditional retailing including direct and indirect retailing approaches. This paper suggests that applying association mining techniques can further improve the dealing of information overload in a web oriented retailing environment.**

***Index terms*—Semantic web, online retailing, data mining, formal concept, Protégé, triple store, Sparql.**

I. Introduction

Information is all around us, easy to collect, store and access. It consists of the useful data that is needed to solve our problems. But as information is more and more overloaded, managers, researchers or retailers have to spend more time to process the information before making their decisions, due to the fact that the stored information is unstructured.

One way of solving this problem is to turn the important and useful information into knowledge and filter away the less important information. This requires understanding of when to use information, how to find it, and how to present it to the target customer. It is imperatives that enterprises will need to exploit the knowledge and information available on the WWW as far as possible, so its static nature will tend to increase the information overload. This paper suggests using Semantic Web to solve the information overload problem. The technological basis of the Semantic Web provides a unified framework within which many approaches to the problem of information overload can be integrated.

Manuscript received May 10, 2008. Manuscript accepted for publication June 20, 2008.

Yi Wang is with Nottingham Business School, Park Row, floor 2, Nottingham Trent University Burton Street, Nottingham, NG1 4BU, U.K.

The major components in Semantic Web are shared conceptualizations and terminologies to describe customers, operation, content of web page, etc. These conceptualizations and terminologies are called ontologies. They may refer to agreed ways of describing customer's preference and demand, operational capacity and constraints, retailing brand. Ontologies can be used as meaningful enrichment for the content. They provide common base framework within which information can be properly shared, modeled and filtered. The next part of paper reviews some existing literature regarding online retailing and information overload. The third part of the paper explains semantic web as a mean to solve information overload. The fourth part gives some technical background for the usage of semantic web in retailing. The fifth part suggests some improvement based on existing technologies. The sixth part will discuss in detail about dynamic causal mining as a specific data mining tool. The seventh part will illustrate a practical problem. The final part concludes the whole paper.

II. Literature Review

On-line retailing offers more choices and flexibility [Lamoureux, 97] and, at the same time, eliminates huge inventories, storage costs, utilities, space rental, etc. [Avery, 97]. Companies can design and personalize advertising for each customer [Peterson, *et al*., 1997]. The Internet can provide timely information to customers because of its ability for instant communication [Lane, 1996]. This means more interaction [Rosenspan, 2001] and quicker responses [Isaac, 1998]. The information can be used to assist new product development and introduction [Higgins, 2001]. The communication also helps with identifying prospects [Ebling, 2001], sales and relationship building [Ginovsky, 2001], and deepening customer loyalty [Kranzley, 2001]. Perlow [1999] describes a software company characterized by an environment. It also allows for easy follow-up on customers' needs [Marks, 1998]. Retailing activity occurs through three types of channels: communication, transaction, and distribution channels [Peterson, 1997].

Studies of the semantics web were initiated by Tim Berners-Lee, the creator of the World Wide Web [Berners-Lee, 01]. The Web is referred to as the "semantic Web", where information will be retrieve from intelligent network services such as information brokers and search agents [Decker & Melnik, 00, Decker *et al*., 00].

The World Wide Web has evolved into a dynamic, distributed, heterogeneous, complex network, which is hard to

control [Albert *et al*., 99, Huberman & Adamic, 99]. It is important to have consistent understanding and interpretation [Helbing *et al.*, 00,]. in the World Wide Web [Huberman & Adamic, 99, Huberman *et al.*, 97, Barabasi & Albert, 99].

When more information arrives than individuals can process, an information overload occurs [Simon, 1971]. Much research has done in dealing with information overload. Abiteboul *et al*. [Abiteboul, 00] systematically investigated the data on the Web and the features of semistructured data. Zhong studied text mining on the Web including automatic construction of ontology and filtering system [Zhong, 00; Zhong *et al.*, 00b]. Liu *et al*. worked on e-commerce agents [Liu & Ye, 01] and KDDA (Knowledge Discovery and Data Mining Agents) [Liu & Zhong, 99, Liu *et al*, 01] to minimize the information overload.

III. A General Framework of Semantic Web

Figure 1 shows the simplified representation of a semantic network where enter prise on one side provides information and the user on the other hands give the queries.

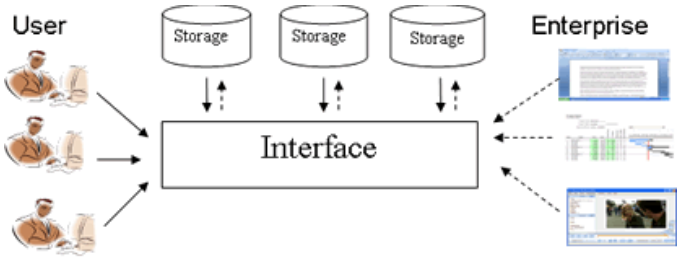


Figure 1 A general frame work

The semantic network interface has three goals:

1. **To provide ontologies for interoperations.** In many cases, the customer and the enterprise does not speak the same language. They have different preference, gain different knowledge and get different information from the product or services. In order to have a more successful relationship, there is need for some ontology which bridges these gaps.
2. **To unify information from different document format**. The enterprises provides the information online in different format, some as word documents, some as excel documents, some as media files., etc. The goal here is to integrate all relevant based on the given ontology. And map all the information to a user friendly representation.
3. **To Store the relevant information and update the ontology.** The interface should be able to retrieve and represent the information based on user's queries. And the interface should be able to update the ontologies for improvement, thus rather storing a large amount of information, the relevant ontologies or relations are stored.

IV. Ontology Development and Storage

An ontology for retailing defines a common vocabulary for any participants in retailing, including customer, manager, retailer, etc. who need to share information in a domain. It includes machine-interpretable definitions of basic concepts in the domain and relations among them. The major goals for ontology development (Natalya and McGuinness, 07) are:

1. To share common understanding of information among user (Musen 1992; Gruber 1993).
2. To enable reuse of domain knowledge was one of the driving forces behind recent surge in ontology research.
3. To change domain assumptions if information about the domain changes.
4. To separate the domain information from the operational information
5. To reuse the existing ontologies and extending them (McGuinness *et al*. 2000).

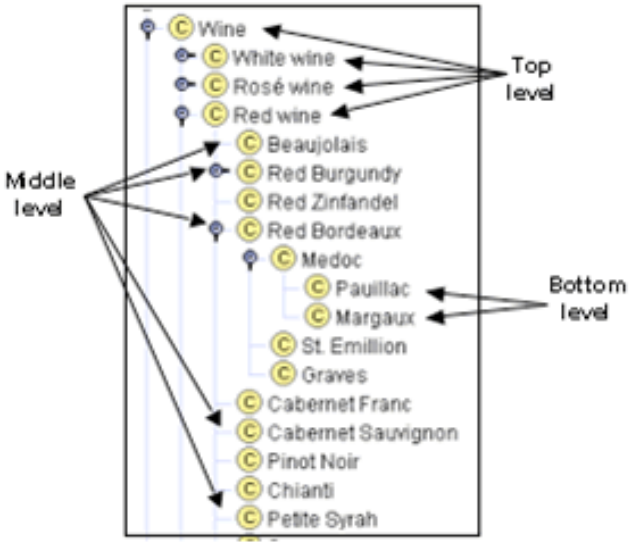


Figure 2 Wine brand taken from(Natalya and McGuinness, 07)

The most common tools for developing ontology are Protégé-2000 (Protege, 2000), Ontolingua (Ontolingua, 1997), and Chimaera (Chimaera, 2000) as ontology-editing environments.

Figure 2 shows a sample of the protégé interface for wine ontology. It is typically based on graphical class hierarchical development (Uschold and Gruninger, 96). An ontology can be seen as a triplets (Subject, relation, object). An typical example is "Wine", which is subject, "has brand name", which is a relation, "Chinati" which is an object. Instead of normal Sql technology, the ontology can be store in a triple store, which is a specific type of data storage. It is designed to store and retrieve identities that are constructed from triplex collections of *strings* (sequences of letters) and can be queried with Sparql. Sparql query consists of three parts. The pattern matching part, which focus on matching patterns of graphs, like optional parts, union of patterns, nesting, The solution modifiers part , which allows to modify values applying classical operators like projection, distinct, order, limit, and offset. And the output part which consists of different types: yes/no solution based on descriptions of resources.

V. MINING TECHNIQUES

The ontologies developed for semantic web are based on user experience. This requires the need for the developers to have an increased understanding of the complex issues involved in the ontology. And sometimes it is difficult to make a universal accepted ontology. And a lot of hidden relations are not modeled at all. This section suggests combining classical ontology development with data mining for identifying hidden information and expanding the application area of both techniques. This gives an improved description of the target system represented by a database; it can also improve strategy selection and other forms of decision making.

Data mining techniques to automatically discover and extract information from Web documents and services [Kosala & Blockeel, 00, Srivastava *et al*., 00, Zhong, 01]. Zhong *et al*. proposed a way of mining peculiar data and peculiarity rules that can be used for Web-log mining [Zhong, 99]. They also proposed ways for targeted retailing by mining classification rules and retail value functions [Yao and Zhong, 01, Zhong *et al*., 00]. Data mining is the systematic refining of information resources on the Web for business intelligence [Hackathorn, 00].

This paper suggests using associative formal concept analysis as the base tools for data mining (Wolff, 94) and develops it further using Dynamic causal mining as the technique for mining the relations. The DCM algorithm was discovered in 2005 [Pham *et al.*, 2005] using only counting algorithm to integrate with Game theory. It was extended in 2006 [Pham *et al.*, 2006] with delay and feedback analysis, and was further improved for the analysis in Game theory with Formal Concept analysis [Wang, 2007]. DCM enables the generation of dynamic causal rules from data sets by integrating the concepts of Systems Thinking [Senge *et al*., 1994] and System dynamics [Forrester, 1961] with Association mining [Agrewal *et al*., 1996]. The algorithm can process data sets with both categorical and numerical attributes. Compared with other Association mining algorithms, DCM rule sets are smaller and more dynamically focused. The pruning is carried out based on polarities. This reduces the size of the pruned data set and still maintains the accuracy of the generated rule sets. The rules extracted can be joined to create dynamic policy, which can be simulated through software for future decision making. The rest of this section gives a brief review of Association mining and System Dynamics.

Association mining was discovered by Agrawal [Agrawal *et al*., 1996]. It was further improved in various ways, such as in speed [Agrawal *et al*., 1996 and Cheung *et al*., 1996] and with parallelism [Zaki *et al*., 1997] to find interesting associations and/or correlation relationships among large sets of data items. It shows attributes value conditions that occur frequently together in a given dataset. It generates the candidate itemsets by joining the large itemsets of the previous pass and deleting those subsets which are small in the previous pass without considering the transactions in the database. By only considering large itemsets of the previous pass, the number of candidate large itemsets is significantly reduced.

Systems thinking is about the interrelated actions which provide a conceptual framework or a body of knowledge that makes the pattern clearer [Senge *et al*., 1994]. It is a combination of many theories such as soft systems approach and system theory [Coyle, 1996]. Systems thinking seeks to explore things as wholes, through patterns of interrelated actions. *System dynamics* [Sterman 1994 & Coyle, 1996], is a tool to visualize and understand such patterns of dynamic complexity, which is build up from a set of system archetypes based on principles in *System thinking* [Sterman, 2000]. *System dynamics* visualizes complex systems through causal loop diagrams. A causal loop diagram consists of a few basic shapes that together describe the action modeled.

System dynamics addresses two types of behavior, *sympathetic* and *antipathetic* [Pham *et al*., 2005]. *Sympathetic behavior* indicates an initial quantity of a target attributes starts to grow, and the rate of growth increases. *Antipathetic behavior* indicates an initial quantity of a target attributes starts either above or below a goal level and over time moves toward the goal.

VI. MINING ALGORITHM

A. Problem Formulation

Let D denote a database which contains a set of n records with attributes {A1, A2, A3,…, An.}, where each attribute is of a unique type (sale price, production quantity, inventory volume, etc). Each attribute is linked to a time stamp t. To apply DCM, the records are arranged in a temporal sequence (t = 1, 2,…, n). The causality between attributes in D can be identified by examining the polarities of corresponding changes in attribute values. Let D_{new} be a new data set constructed from D. A generalized dynamic association rule is an implication of the form A1 $\rightarrow$p A2, where A1 $\subset$ D, A1 $\subset$ D, A1 $\cap$ A2= ϕ and p is the polarity.

The implementation of the *DCM* algorithm must support the following operations:

(1) To add new attributes.

(2) To maintain a counter for each polarity with respect to every dynamic value set. While making a pass, one dynamic set is read at a time and the polarity count of candidates supported by the dynamic sets is incremented. The counting process must be very fast as it is the bottleneck of the whole process.

B. Algorithm Description

DCM makes two passes over the data as shown in Figure 3. In the first pass, the *support* of individual attributes is counted and the frequent attributes are determined. The dynamic values are used for generating new potentially frequent sets and the actual *support* of these sets is counted during the pass over the data. In subsequent passes, the algorithm initializes with dynamic value sets based on dynamic values found to be frequent in the previous pass. After the second of the passes, the *causal rules* are determined and they become the candidates for the dynamic policy. In the *DCM* process, the main goal is to find the strong dynamic causal rule in order to form a policy. It also represents a filtering process that prunes away static attributes, which reduces the size of the data set for further mining.

Part 1: – Preprocessing: Removal of the "least" causal data from database
Part 2: – Mining: Formation of a rule set that covers all training examples with minimum number of rules
Part 3: – Checking: Check if an attribute pair is self contradicting (sympathetic and antipathetic at the same time)
Input: The original database, the values of the pruning threshold for the neutral, sympathetic and antipathetic supports.
Output: Dynamic sets
Step 1: Check the nature of the attributes in the original database (numerical or categorical). Initialize a new database with dynamic attributes based on the attributes and time stamps from original database.
Step 2: Initialize a counter for each of the three polarities.
Step3: Prune away all the dynamic attributes with supports above the input thresholds.
Step 4. Check weather a rule is self-contradictory (a rule is both sympathetic and antipathetic).
Step 5. If step 1 returns true then
Retrieve the attribute pair form the preprocessed database
Step 6. Initialize a counter that includes polarity combination
Step 7. For the pair of attributes
Count the occurrence of polarity combination with two records each time.
Prune away the pairs if the counted support is below the input threshold.

Figure 3. The Steps of DCM

Table I
Pruned results

Data set	Single Support						
	0.05	0.10	0.15	0.20	0.25	0.30	0.35
Adult	5%	20%	27%	74%	100%	100%	100%
Bank	11%	20%	60%	94%	100%	100%	100%
Cystine	5%	33%	70%	100%	100%	100%	100%
Market basket	6%	10%	50%	86%	100%	100%	100%
Mclosom	1%	13%	38%	72%	90%	100%	100%
ASW	1%	8%	40%	68%	70%	97%	100%
Weka-base	6%	25%	52%	86%	100%	100%	100%

VII. EXPERIMENT

A. Data preparation

The overall aim is to identify hidden dynamic changes. The original data was given as shown in Table 2. The only data of interest are the data with changes, for example sale amounts of a product, the time stamp, etc, The rest of the static data, such as the weight and the cost of the product can be removed.

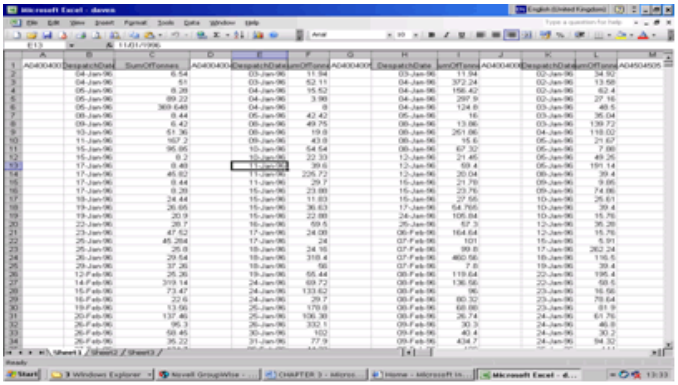


Table 2 Original data sets

After cleaning the data, the dynamic attributes are found as shown in Table 3. The dynamic attribute is calculated by finding the difference between sales amounts in one month and sales amounts in the previous month.

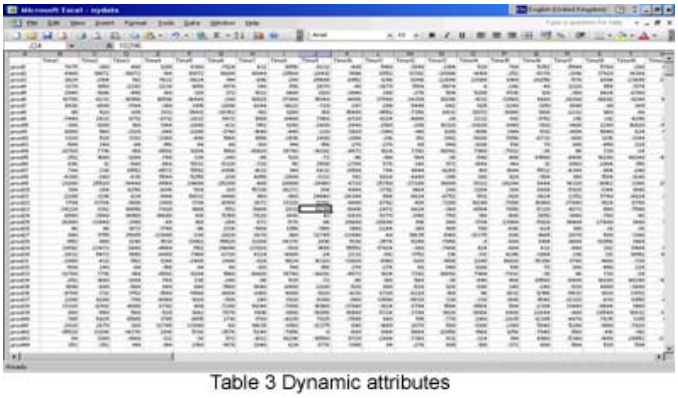


Table 3 Dynamic attributes

In the next step, the neutral attributes are pruned. The idea of pruning is to remove redundant dynamic attributes; thus fewer sets of attributes are required when generating rules. The first pruning is based on the single attribute support. In this case, the single attribute support is defined to be 0.5, which means that if an attribute with polarity +, -, or 0 occurs in more than half of total time stamps, it will be pruned. In this case, 429 attributes remain for the rule generation.

In this experiment, dynamic sets are compared based on a simultaneous time stamp. Then the support of sympathetic and antipathetic rules for each dynamic set is calculated. The support is used as the threshold to eliminate unsatisfactory dynamic sets and to obtain the rules from the satisfactory sets.

B. Evaluation and Results

The algorithm was run based on the procedures described in previous sections. Figure 4 shows the plot of sympathetic and antipathetic support. The x –axis represents the support and the y-axis represents the number of rules. This database shows that there are more sympathetic rules than antipathetic rules. The figure shows that increasing support will lead to exponential growth of the rules. As the support reaches 0.05 or 5, as it indicates on the figure, the number of rules is 630. Most of these rules are redundant and have no meaning due to the low support. Figure 5 shows the rule plot with support equal to average value, where the "+support" = "the average of all

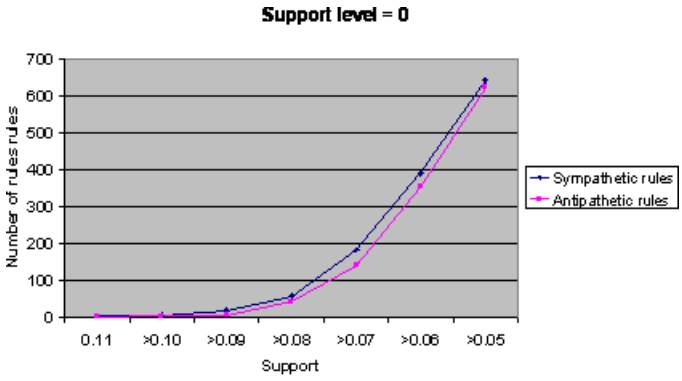


Figure 4 Rule plot with support level = 0

positive records" and "–support" = "the average of all negative records". The number of rules has decreased by applying the support level.

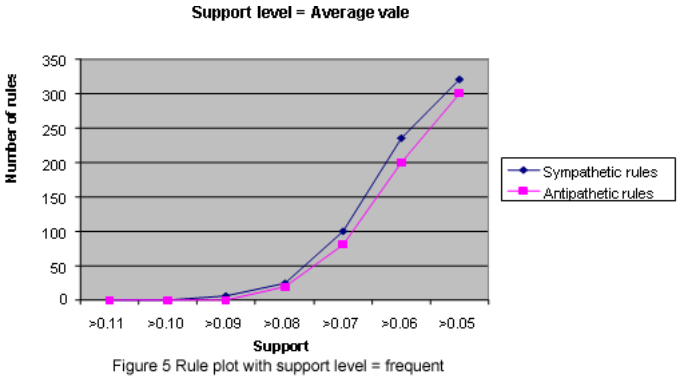


Figure 5 Rule plot with support level = frequent

Table 2 shows the extracted strong rules with support level equal to average value and support larger than 0.08. There are only dynamic pairs so there is no need to do the simulation. The connection can then be put onto concept relational software as

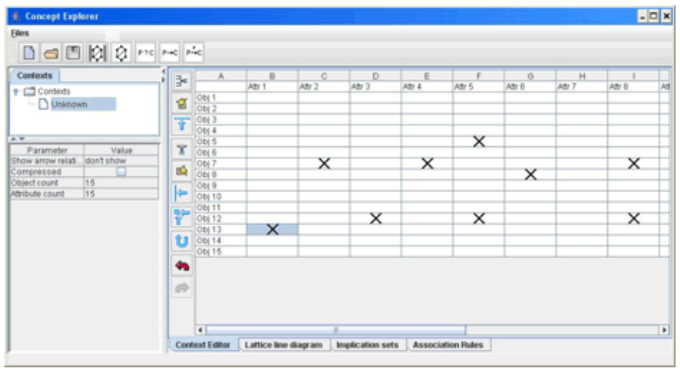


Figure 6. Pictorial scan for ConExp

ConExp (Conceptual explorer), and can be represented by a lattice as shown in Figure 7.

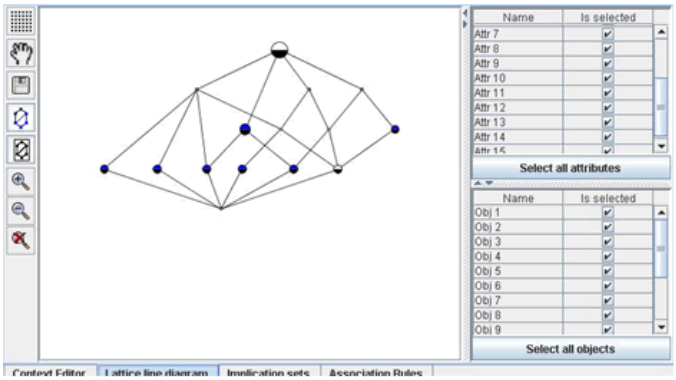


Figure 7. Lattice generated

Table II
Result generated by the algorithm

Strong rules	**Support**
Sympathetic	
{C15276179, F030008}	0,093
{J08008008, F060010}	0,089
{A04004004, A05005005}	0,086
{A05005006, C10251104}	0,084
{A04004004, F100020}	0,082
Antipathetic	
{A05005008, C15276179}	0.092
{C10251104, F070010}	0.083
{A05005008, F030008}	0.082

C. Discussion

A priori it is provided some form of causal information, i.e. suggesting a possible direction of causation between two attributes, but there is no basis to conclude that the arrow indicates direct or even indirect causation. The DCM algorithm, on the other hand, shows causality between attributes. Thus, where association rule generation techniques find surface associations, causal inference algorithms identify the structure underlying such associations.

Each type of relationship generated by the *DCM* algorithm provides additional information. The *DCM* algorithm finds four kinds of relationships, each of which deepens the user's understanding of their target system by constructing the possible models. For example, $A_1 \rightarrow^{+} A_2$ provides more information than $A_1 \rightarrow A_2$ because the latter indicates that A_1 coexists with A_2. The condition of the rule is not stated (whether sympathetic or antipathetic). A genuine causality such as $A_1 \rightarrow^{+} A_2$ provides useful information because it indicates that the relationship from A_1 to A_2 is strictly sympathetic causal.

The rules extracted by DCM can be simulated by using software to model the future behavior. The rules extracted by association algorithm cannot be simulated.

VIII. Conclusion

This paper has considered the most fundamental ways to tackle the problems caused by information overload and complexity in retailing. The information system would be available always and everywhere, reacting immediately to any request for guidance or any change in the situation. It would constantly be fed with new information thus cause the

overload. Using semantic web is a popular way dealing with such problems; however the semantic web requires improvement. This paper suggests integration of semantic web and association formal concept analysis method to improve the analysis.

REFERENCES

[1] Abiteboul, S., Buneman, P., and Suciu, D. Data on the Web, Morgan Kaufmann, 2000.

[2] Agrawal, R., Mannila, H., Srikant, R., Toivonen, H., & Inkeri. A. (1996). Fast Discovery of Association Rules, Advances in Knowledge Discovery and Data Mining. The Association for the Advancement of Artificial Intelligence and The MIT Press, 307–328.

[3] Albert, R., Jeong, H. and Barabasi, A.-L., Diameter of the World-Wide Web, Nature, 410, 130-131, 1999.

[4] Andrews, Jonathan; Trites, Gerald, "Net Sales." CA Magazine, Vol.130, No.6:12-15, Aug 1997.

[5] Barabasi, A.-L. and Albert, R., Emergence of scaling in random networks, Science, 286, 509-512, 1999.

[6] Berners-Lee, T., Hendler, J., and Lassila, O. The semantic Web, Scientific American, 29-37, May 2001.

[7] Cheung, D., Han, J., Ng, V. T. & Fu, A.W. (1996).Fast distributed algorithm for mining association rules. In International Conference on Parallel and Distributed Information Systems, Tokyo, Japan, 31–42.

[8] Chimaera (2000). Chimaera Ontology Environment.www.ksl.stanford.edu/software/chimaera

[9] Coyle, R. G. (1996). *System dynamics Modelling: A Practical Approach*, London, Chapman and Hall.

[10] Decker, S.& Melnik, S.. The semantic web: the roles of XML and RDF, IEEE Internet Computing, 4:5, 63-74, 2000.

[11] Decker, S., Mitra, P., and Melnik, S. Framework for the semantic web: an RDF tutorial, IEEE Internet Computing, 4:6, 68-73, 2000.

[12] Ebling, Tom, "The economics of online banking," Target Retailing; Philadelphia; Vol. 24, Issue 2:67-78, Feb. 2001.

[13] Flood, R. (1999). *Rethinking the fifth discipline. Learning within the unknowable*, Ruthledge, London

[14] Forrester, J. W. (1961). Industrial Dynamics: A Major Breakthrough for Decision Makers. *Harvard Business Review*, July-August, 37-66.Fountain, J. E. (2001).

[15] Ginovsky, John, "Bricks can excel at clicks," ABA Bankers News; Washington; Vol. 9, Issue 8, page 8, Apr. 17, 2001.

[16] Gruber, T.R. (1993). A Translation Approach to Portable Ontology Specification. *Knowledge Acquisition* **5**: 199-220.

[17] Hackathorn, R.D.Web Farming for the Data Warehouse, Morgan Kaufmann, 2000

[18] Hawn, Matthew, "Stay on the Web", MacWorld, Vol.13, No.4:94-98, Apr. 1996.

[19] Helbing, D., Huberman, B. A., and Maurer, S. M, Optimizing traffic in virtual and real space, in: Traffic and Granular Flow '99: Social, Traffic, and Granular Dynamics, Helbing, D., Herrmann, H. J., Schreckenberg, M., and Wolf, D. E. (Eds.), Springer-Verlag, 2000.

[20] Huberman, B. A. and Adamic, L. A., Growth dynamics of the World-Wide Web, Nature, 410, 131, 1999.

[21] Huberman, B. A., Pirolli, P. L. T., Pitkow, J. E., and Lukose, R. M., Strong regularities in World Wide Web surfing, Science, 280, 96-97, 1997.

[22] Kranzley, Arthur, "Lifestyles fuel Internet banking," Credit Card Management; New York, Vol. 14, Issue 2, page 72, May 2001.

[23] Kosala, R. and Blockeel, H. Web mining research: a survey, ACM SIGKDD Explorations Newsletter, 2, 1-15, 2000.

[24] Higgins, Amy, "Designing with the new Internet," Machine Design; Cleveland; Vol. 73, Issue 14:90-94, July 26, 2001.

[25] Isaac, Peter, "Electronic commerce benefits is frictionless trading," New Zealand Manufacturer, pp.38-39, Feb 1998.

[26] Lane, Andrea, "Success in sight… or site?" Australian Accountant, Vol.66, No.10:22-25 , Nov 1996.

[27] Ling, C.X. and Li, C. Data mining for direct retailing: problems and solutions,

[28] Proceedings of KDD'98, 73-79, 1998.

[29] Liu, J. and Ye, Y. (Eds.) Advances in E-commerce Agents: Brokerage, Negotiation,

[30] Security, and Mobility, Springer-Verlag, 2001.

[31] Liu, J. and Zhong, N. (Eds.) Intelligent Agent Technology: Systems, Methodologies, and Tools, World Scientific, 1999.

[32] Liu, J., Zhong, N., Tang, Y.Y., and Wang, P.S.P. (Eds.) Agent Engineering, World Scientific, 2001.

[33] Long, Johnny, " E-COMMERCE: Doing What's Best for Business," Data Communications, Vol.26, No.16:77-80, Nov 21, 1997.

[34] Malazdrewicz, Michael A., "Navigating on the Net," CA Magazine, Vol.129, No.6:22-26, Aug. 1996.

[35] Marks, Michael, "The Internet: Rewriting the rules of business." Supply House Times, Vol.40, No.11, page 63, Jan 1998.

[36] McGuinness, D.L. and Wright, J. (1998). Conceptual Modeling for Configuration: A Description Logic-based Approach. *Artificial Intelligence for Engineering Design, Analysis, and Manufacturing - special issue on Configuration*.

[37] McKim, Robert, "Dollars and sense on the Web", Target retailing, Vol.20, No.7:30 31, July 1997.

[38] Musen, M.A. (1992). Dimensions of knowledge sharing and reuse.*Computers and Biomedical Research***25**: 435-467.

[39] Natalya F. Noy and Deborah L. McGuinness Ontology Development 101: A Guide to Creating Your First Ontology.

[40] http://protege.stanford.edu/publications/ontology_development/ontology 101-noy-mcguinness.html

[41] Ontolingua (1997). Ontolingua System Reference Manual. http://www-ksl svc.stanford.edu: 5915 /doc/frame-editor/index.html

[42] Perlow, L. A. (1999) "The Time Famine: Toward a Sociology of Work Time",

[43] *Administrative Science Quarterly,* (44)1, pp. 57-81.

[44] Peterson, R.A., Balasubramanian, S. and Bronnenberg B.J., "Exploring the Implications of the Internet for Consumer Retailing," Journal of the Academy of Retailing Science, Vol. 25, No. 4:329-346, 1997.

[45] Pham, D.T., Wang, Y. & Dimov, S. (2005). Intelligent Manufacturing strategy selection. *Proc 1st Int Virtual Conf on Intelligent Production Machines and Systems* Oxford: Elsevier. 312-318.

[46] Pham, D.T., Wang, Y., & Dimov, S. (2006). Incorporating delay and feedback in intelligent manufacturing strategy selection. *Proc 2nd Int Virtual Conf on Intelligent Production Machines and Systems*. Oxford: Elsevier. 246-252.

[47] Protege (2000). The Protege Project.http://protege.stanford.edu

[48] Rendleman, John, "Customer data means money," Informationweek;, Manhasset; Issue 851:49-50, Aug. 20, 2001.

[49] Richardson, G. P.(1996) System Dynamics. In: Gass, S.I., Harris, C.M. (eds.): Encyclopedia of Operations Research and Management Science. Kluwer Academic Publishers. Boston 656-660.

[50] Richardson, G.P. and Andersen, D.F. Teamwork in group model building. System Dynamics Review, 11 2. (1995), 113-137

[51] Rosenspan, Alan, "The art of the questionnaire," Target Retailing, Philadelphia, Vol.24, Issue 8:42-44, Aug. 2001.

[52] Sandilands, Ben, "The Internet: A tool of the trade?" Australian Accountant, Vol.67, No.11:14-17, Dec 1997.

[53] Senge, P., Kleiner, A., Roberts, C., Ross, R., & Smith, B. (1994). The fifth discipline fieldbook. New York: Doubleday

[54] Simon, H. (1971) "Designing Organizations for an Information-Rich World" in M.

[55] Greenberger (ed.) Computers, Communications, and the Public Interest, Baltimore, MD: The Johns Hopkins Press, pp. 38-52.

[56] Srivastava, J. Cooley, R., Deshpande, M. and Tan, P. Web usage mining: discovery and applications of usage patterns from web data, SIGKDD Explorations, Newsletter of SIGKDD, 1, 12-23, 2000.

[57] Sterman, J. D. (1994). Learning in and about Complex Systems. System dynamics Review. Volume 10 (2-3), 291-330.

[58] Sterman, J. D. (2000). Business Dynamics: Systems Thinking and Modelling for a Complex World, Irwin McGraw-Hill, Boston,

[59] Uschold, M. and Gruninger, M. (1996). Ontologies: Principles,Methods and Applications. Knowledge Engineering Review 11(2).

[60] Wolff, K. E. (1994). "A first course in Formal Concept Analysis". F. Faulbaum StatSoft '93: 429–438, Gustav Fischer Verlag.

[61] Yao, Y.Y. and Zhong, N. Mining retail value functions for targeted retailing, Proceedings of the 25th IEEE Computer Society International Computer Software and Applications Conference (COMPSAC 2001), 2001.
[62] Wang, Y. (2007). Integration of data mining with Game theory, Journal of International Federation for Information Processing, Volume 207/2006, Boston: Springer, pp 275-280
[63] Zhong, N., Yao, Y.Y., and Ohsuga, S. Peculiarity oriented multi-database mining, J. Zytkow and Jan Rauch (eds.) Principles of Data Mining and Knowledge Discovery, LNAI 1704, Springer-Verlag, 136-146, 1999.
[64] Zaki, M. J., Parthasarathy, S., Ogihara, M. & Li, W. (1997). New Parallel Algorithms for Fast Discovery of Association Rules, Data Mining and Knowledge Discovery, Vol. 1, No. 4, 343-37.
[65] Zhong, N., Dong, J.Z., and Ohsuga, S. Rule discovery by soft induction techniques,
[66] Neurocomputing, An International Journal, 36:1-4, 171-204, Elsevier, 2000.
[67] Zhong, N. A Study on E-mail Filtering by Uncertainty Sampling and Relation Learning, Technical Report, Yamaguchi University, 2000.
[68] Zhong, N., Yao, Y.Y., and Kakemoto, Y. Automatic construction of ontology from text databases, N. Ebecken and C.A. Brebbia (Eds.) Data Mining, 2, WIT Press, 173-180, 2000.(b)
[69] Zhong, N. Knowledge discovery and data mining, The Encyclopedia of Microcomputers, 27, Supplement 6,235-285, Marcel Dekker, 2001.

The databases were contributed by many researchers, mostly from the field of machine learning and data mining and collected by the machine learning group at the University of California, Irvine. Two of the datasets, MCslom and ASW, are taken from real life. These data sets are described briefly below

Cystine Database: This data arises from a large study to examine EEG correlates of genetic predisposition to alcoholism. It contains measurements from 64 electrodes placed on the scalp sampled at 256 Hz (3.9-msec epoch) for 1 second.

Weka base: This dataset contains time series sensor readings of the Pioneer-1 mobile robot. The data is time series, multivariate. The few are binary coded 0.0 and 1.0. Two categorical variables are included to delineate the trials within the datasets. The data is broken into "experiences", in which the robot takes action for some period of time and experiences a controlled interaction with its environment.

Market basket: Classical association data mining, used in WeKa analysis. It consists of 100 different transactions.

Bank data: Includes 600 instances of bank transactions and 12 attributes in each instance.

ASW: This data consists of real life data. This dataset contains 65536 attributes of metal manufacturing, with 8 records in each attribute.

Mclosom: Manufacturing database for logistics. 72 time stamps and 50 attributes for 5 different classes.

Base de Conocimientos del Monitoreo de Parámetros Sanguíneos

Israel Rivera Zarate, Patricia Pérez Romero, Jesús Pimentel Cruz

***Resumen*— Se propone un sistema capaz de brindar un apoyo al paciente diabético dado el gran desconocimiento que la población tiene respecto a esta enfermedad. La base de conocimientos se ha tomado gracias a la asesoría de médicos y laboratorista clínicos. Esta primera versión del sistema inteligente utiliza como motor de inferencia lógica difusa dadas sus características de manejo de incertidumbre. Este proyecto permitirá llevar un registro preciso de los niveles de diferentes parámetros sanguíneos de un paciente así como generar representaciones gráficas y estadísticas de control de forma que permita apoyar en la prevención y toma de decisiones oportunas de la diabetes.**

***Palabras clave*—Base de conocimiento, parámetros sanguíneos, monitoreo.**

KNOWLEDGE BASE FOR MONITORING OF THE BLOOD PARAMETERS

***Abstract*—We propose a system capable to help a patient with diabetes taking into account that in general the persons have little knowledge about this disease. This knowledge base was developed in cooperation with medic personnel. The system uses a fuzzy logic inference engine and, thus, is capable of managing uncertainty. This project allows keeping the records of values of various blood parameters, graphic representation of data and statistic information, and it is used in prevention and decision making for patients with diabetes.**

***Index Terms*—Knowledge base, blood parameters, monitoring.**

I. INTRODUCCIÓN

No hace mucho tiempo, se creía que algunos problemas como la demostración de teoremas, el reconocimiento de la voz y el de patrones, ciertos juegos (como el ajedrez o las damas), y sistemas altamente complejos de tipo determinista o estocástico, debían ser resueltos por personas, dado que su formulación y resolución requieren ciertas habilidades que sólo se encuentran en los seres humanos (por ejemplo, la habilidad de pensar, observar, memorizar, aprender, ver, oler, etc.). Sin embargo, el trabajo realizado en las tres últimas décadas por investigadores procedentes de varios campos, muestra que muchos de estos problemas pueden ser formulados y resueltos por máquinas. El amplio campo que se conoce como inteligencia artificial (IA) trata de estos problemas, que en un principio parecían imposibles, intratables y difíciles de formular utilizando computadoras[1].

Hoy en día, el campo de la IA engloba varias subáreas tales como los sistemas expertos, la demostración automática de teoremas, el juego automático, el reconocimiento de la voz y de patrones, el procesamiento del lenguaje natural, la visión artificial, la robótica, las redes neuronales, etc. (una revisión de los campos que componen la IA se puede encontrar en Castillo, Gutiérrez y Hadi, 1997). Ver Fig. 1. Aunque los sistemas expertos constituyen una de las áreas de investigación en el campo de la IA, la mayor parte de las restantes áreas, si no todas, disponen de una componente de sistemas expertos formando parte de ellas.

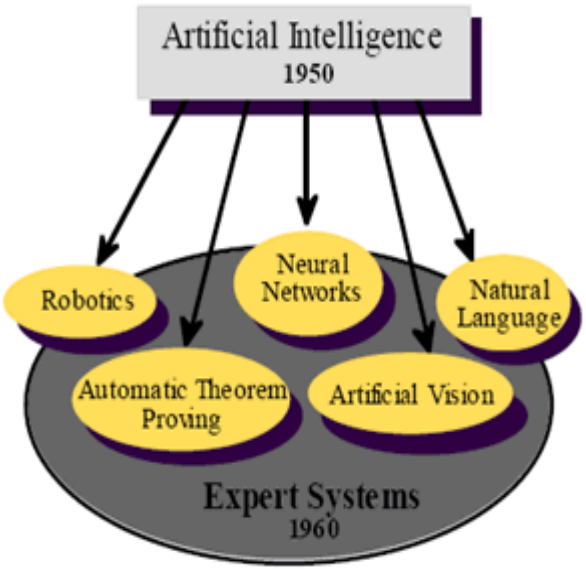


Fig. 1. Areas de la Inteligencia Artificial

Manuscrito recibido el 30 de marzo del 2008. Manuscrito aceptado para su publicación el 15 de junio del 2008.

I. Rivera Zarate, Centro de Innovación y Desarrollo Tecnológico en Cómputo del Instituto Politécnico Nacional, México, D. F. (teléfono: 57296000 Ext. 52535; e-mail: irivera@ipn.mx).

P. Pérez Romero, Centro de Innovación y Desarrollo Tecnológico en Cómputo del Instituto Politécnico Nacional, México, D. F. (teléfono: 57296000 Ext. 52536; e-mail: promerop@ipn.mx).

J. Pimentel Cruz, Centro de Innovación y Desarrollo Tecnológico en Cómputo del Instituto Politécnico Nacional, México, D. F. (teléfono: 57296000 Ext. 52535; e-mail:jpimente@ipn.mx).

II. Componentes de un Sistema Experto

Los sistemas expertos son máquinas que piensan y razonan como un experto lo haría en una cierta especialidad o campo. Por ejemplo, un sistema experto en diagnóstico médico requeriría como datos los síntomas del paciente, los resultados de análisis clínicos y otros hechos relevantes, y, utilizando éstos, buscaría en una base de datos la información necesaria para poder identificar la correspondiente enfermedad.

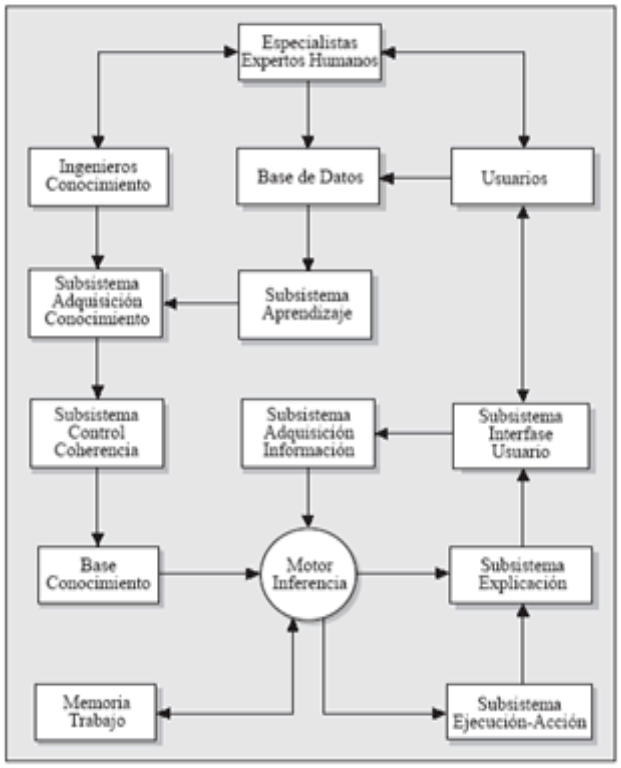


Fig. 2. Componentes de un Sistema Experto

Un Sistema Experto de verdad, no sólo realiza las funciones tradicionales de manejar grandes cantidades de datos, sino que también manipula esos datos de forma tal que el resultado sea inteligible y tenga significado para responder a preguntas incluso no completamente especificadas. La Fig. 2, ilustra los distintos componentes de un sistema experto

III. Tipos de Sistemas Expertos

Los problemas con los que pueden tratar los sistemas expertos pueden clasificarse en dos tipos: problemas esencialmente deterministas y problemas esencialmente estocásticos. Por ejemplo, en el campo médico las relaciones entre síntomas y enfermedades se conocen sólo con un cierto grado de certeza (la presencia de un conjunto de síntomas no siempre implica la presencia de una enfermedad). Estos tipos de problemas pueden también incluir algunos elementos deterministas, pero se trata fundamentalmente de problemas estocásticos [2].

Consecuentemente, los sistemas expertos pueden clasificarse en dos tipos principales según la naturaleza de problemas para los que están diseñados: deterministas y estocásticos. Los problemas de tipo determinista pueden ser formulados usando un conjunto de reglas que relacionen varios objetos bien definidos. Los sistemas expertos que tratan problemas deterministas son conocidos como sistemas basados en reglas.

En situaciones inciertas, es necesario introducir algunos medios para tratar la incertidumbre. Por ejemplo, algunos sistemas expertos usan la misma estructura de los sistemas basados en reglas, pero introducen una medida asociada a la incertidumbre de las reglas y a la de sus premisas. En este caso se pueden utilizar algunas fórmulas de propagación para calcular la incertidumbre asociada a las conclusiones. Durante las últimas décadas han sido propuestas algunas medidas de incertidumbre.

Algunos ejemplos de estas medidas son los factores de certeza, usados en el conjunto de reglas para generar sistemas expertos tales como el sistema experto MYCIN; la lógica difusa, etc. [3].Otra medida intuitiva de incertidumbre es la probabilidad, en la que la distribución de un conjunto de variables se usa para describir las relaciones de dependencia entre ellas, y se sacan conclusiones usando fórmulas conocidas en la teoría de probabilidad. Este es el caso del sistema experto PROSPECTOR, que emplea el teorema de Bayes para la exploración de mineral [2].

Los sistemas expertos que utilizan la probabilidad como medida de incertidumbre se conocen como sistemas expertos probabilístico y la estrategia de razonamiento que usan se

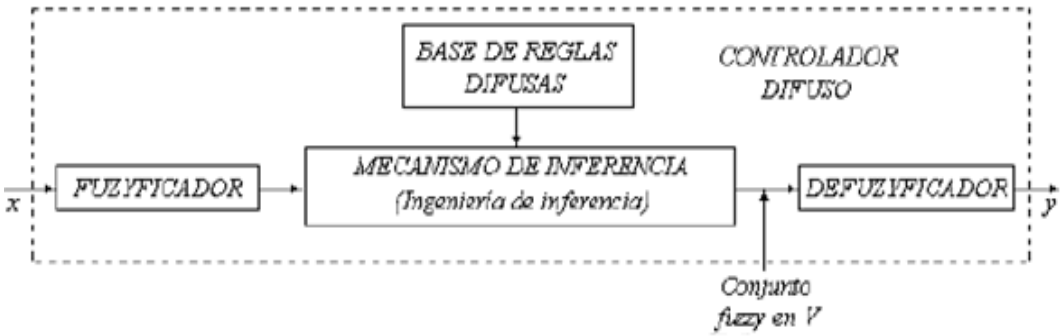


Fig. 3. Componentes de una Máquina de inferencia Difusa

conoce como razonamiento probabilístico, o inferencia probabilística.

IV. Propuesta de Sistema Experto de apoyo al Paciente Diabético

Este sistema propone brindar un apoyo al paciente diabético, que desconozca algo referente a su enfermedad. La base de conocimientos se genero con la asesoría de médicos y laboratorista del Laboratorio de Análisis Clínicos y Microbiológicos Montecristo de Chalco, Edo. de México.

Esta primera versión del sistema inteligente utiliza como motor de inferencia Lógica Difusa dadas sus características de manejo de incertidumbre. Las etapas de una máquina de inferencia difusa se ilustran en la Fig. 3.

La Fuzificacion es la etapa que transforma los valores de las variables de entrada al rango de valores difusos [0 -> 1] o grados de verdad donde 0 es falso y 1 es cierto.

El Mecanismo De Inferencia Difusa son las reglas que se construyen de la forma

Si ….Antecedente 1 Y Antecedente 2
Entonces…Consecuente

Donde los antecedentes corresponden con las variables de entrada y los consecuentes corresponden a su vez con las variables de salida.

Las Variables de Entrada en nuestro sistema son los parámetros que el médico observa mediante una serie de preguntas específicas practicadas al paciente; esta serie de preguntas son tomadas de la norma oficial mexicana para el manejo de la diabetes (Ver apéndice). Por ejemplo:

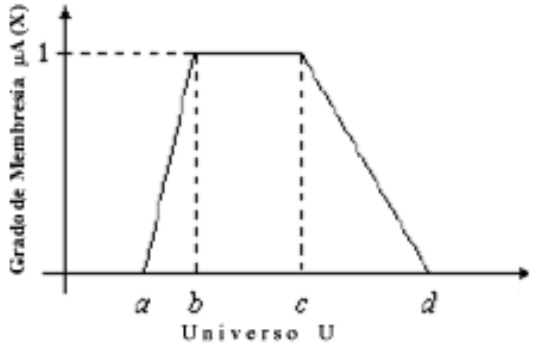
- Raciones o porciones diarias de los grupos alimenticios,
- Historia clínica del paciente,
- Escala de factores de riesgo,
- Niveles de glucosa, etc.

Las Variables de Salida son las recomendaciones a seguir por el paciente para cuidar de su salud y también son específicas y aparecen en la norma oficial mexicana para el manejo de la diabetes Por ejemplo:

- Reducción del consumo habitual de kilocalorías,
- Limitar consumo de grasas,
- Aumento o disminución de la dosis de insulina, etc.

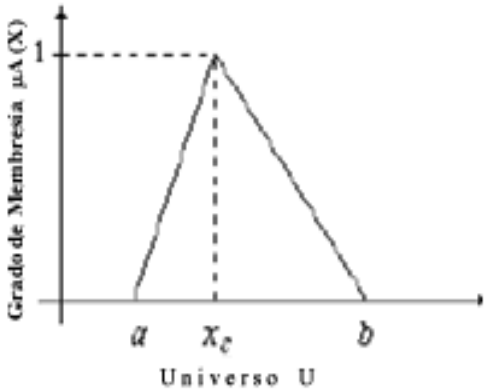
Esta primera versión utiliza tres Funciones de Membresía (tipo trapezoide y triangular) por variable de entrada y asimismo tres Funciones de Membresía (tipo singleton) por variable de salida.

Para las funciones trapezoidales se aplicará:



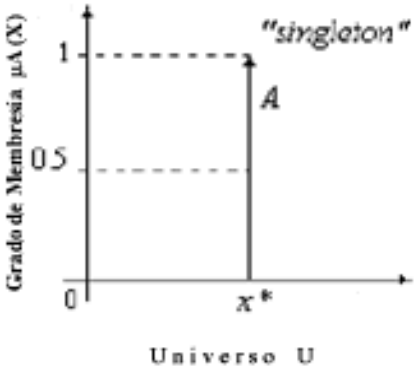
$$\mu_A(x) = \begin{cases} 1, & \text{si } b \le x^* \le c \\ \left(1 - \dfrac{|b - x^*|}{b - a}\right), & \text{si } a < x^* < b \\ \left(1 - \dfrac{|x^* - c|}{d - c}\right), & \text{si } c < x^* < d \\ 0, & \text{de otra manera} \end{cases}$$

Para las funciones triangulares:



$$\mu_A(x) = \begin{cases} 1, & \text{si } x^* = x_c \\ 1 - \dfrac{|x_c - x^*|}{x_c - a}, & \text{si } a < x^* < x_c \\ 1 - \dfrac{|x_c - x^*|}{b - x_c}, & \text{si } x_c < x^* < b \\ 0, & \text{de otra manera} \end{cases}$$

Para el caso de las salidas:



$$\mu_A(x) = \begin{cases} 1, & x = x^* \\ 0, & x \neq x^* \end{cases}$$

El MÉTODO DE INFERENCIA para el manejo de las reglas es el conocido como el MÉTODO MIN MAX, donde se toman los valores mínimos de verdad de los antecedentes y el valor máximo en los consecuentes.

Para determinar el valor de salida se utilizó El MÉTODO DEL CENTROIDE o del CENTRO DE GRAVEDAD dada su salida como un promedio del peso de los consecuentes o variables de salida. Como se ve en la ecuación (1):

$$Z^* = \frac{\sum_{i=1}^{M} \overline{Z}^i \omega_i}{\sum_{i=1}^{M} \omega_i} \qquad (1)$$

En relación a la base de conocimientos, las reglas se realizaron tomando como referencia las recomendaciones citadas por la norma mexicana y la experiencia de los médicos y laboratorista mencionados anteriormente

De manera común se elige trabajar por pares de variables de entrada contra una variable de salida lo cual permite construir lo que en lógica difusa se conoce como una MATRIZ DE INFERENCIA DIFUSA. (FAM.) Ver Tabla I.

TABLA I.
TABLA DE INFERENCIA DIFUSA

Gp / Gd	poco	regular	mucho
baja	B	B	M
media	B	M	A
alta	M	A	A

Gp: glucosa postpandrial Gd: grasas diarias
Cf: Dieta Fraccionada < 1500 Kcal.
(A = alta M= media B= baja)

A partir de la FAM es posible visualizar fácilmente cada una de las reglas. Tenemos por ejemplo:

Si *EL NIVEL DE GLUCOSA POSTPANDRIAL ES ALTO (> 240 mg/dl)*

Y *EL CONSUMO DE GRASAS DIARIAS ES ALTO (> 4 porciones diarias)*

ENTONCES *EL CONSUMO CALORICO DIARIO SERA FRACCIONADO ALTO (6 comidas diarias a < 1500 kilocalorías por día)*

Como se puede apreciar, el método propuesto en este trabajo consiste en relacionar a través de varias FAM conjuntos de pares de variables de entrada y posibles variables de salida; en el entendido que dicha salida mantiene una estrecha relación con la entrada.

V. USO DEL SISTEMA EXPERTO

Para usar el programa se deben plantear las variables de entrada y de salida sobre las cuales de desea hacer el análisis así como las reglas que rigen su relación. El programa aparece listado en un anexo y es el esqueleto general para cualquier relación del conjunto de variables de entrada y salida indicadas en el apéndice.

El programa solicita los rangos de operación de las variables los cuales son aquellos recomendados por la norma y los médicos especialistas, de igual modo las reglas deben almacenarse previamente en la base de conocimientos. Esto corresponde con el arreglo de datos: char reglas[9] que aparece indicado en el listado del programa en la sección titulada como Inferencia Difusa. Observar que las reglas se codifican mediante caracteres tales como: **A** para indicar un valor "alto", **M** para indicar un valor "medio" y **B** para uno "bajo".

El programa entrega como resultado lo que es conocido en la Lógica Difusa como VALOR REAL y es obtenido por el método del CENTROIDE, lo cual produce un consecuente expresado en el rango predefinido por la norma para dicha salida.

VI. PRUEBAS Y RESULTADOS

Para probar el funcionamiento del programa se establecieron dos variables de entrada y una de salida:

Entrada 1. Gp: Nivel de glucosa postpandrial. Es el nivel de la concentración en mg/dl de glucosa capilar medido 2 HR después reingerir alimentos. Sus rangos aparecen definidos en la siguiente gráfica y corresponden con lo especificado por la norma mexicana. Ver Fig. 4.

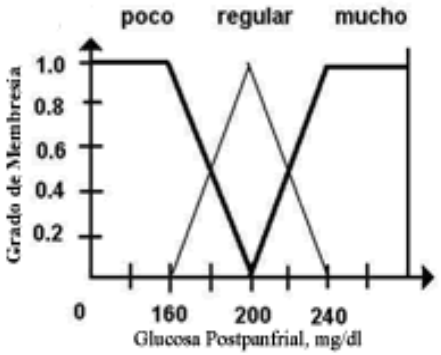


Fig. 4. Funciones de Membresia para la entrada1: Glucosa Postpandrial

Entrada 2. Gd: Nivel de consumo de grasas diariamente. Es el número de porciones diarias de grasas saturadas (origen animal). Sus rangos aparecen definidos en la siguiente gráfica y corresponden con lo especificado por la norma mexicana. Ver Fig. 5.

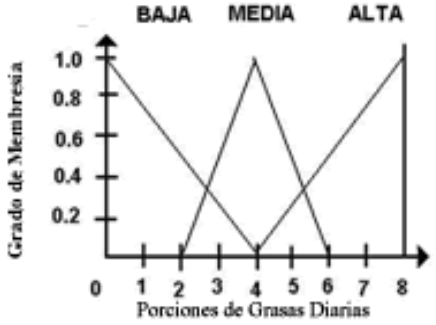


Fig. 5. Funciones de Membresia para la entrada 2: Grasas Diarias

Salida. Cf: Consumo fraccionado. Es el número de comidas al día que deben realizarse para garantizar una modificación en el metabolismo obteniendo como consecuencia una disminución en el sobrepeso. Sus rangos aparecen definidos en la siguiente gráfica y corresponden con lo especificado por la norma mexicana. Ver Fig. 6.

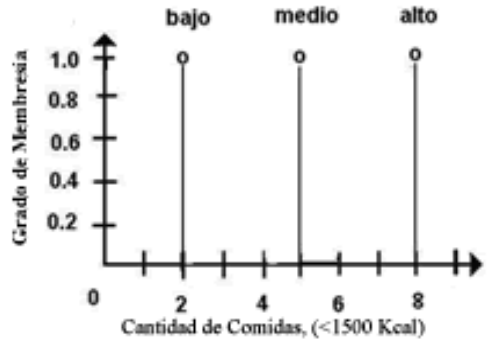


Fig. 6. Funciones de Membresia para la salida: Consumo Fraccionado

Por medio de las definiciones anteriores se puede construir la FAM correspondiente, ver Tabla II.

TABLA II
TABLA DE INFERENCIA DIFUSA

Gp / Gd	poco	regular	mucho
baja	B	B	M
media	B	M	A
alta	M	A	A

Gp: glucosa postpandrial Gd: grasas diarias
Cf: Dieta Fraccionada < 1500 Kcal.
(A = alta M= media B= baja)

Observando la FAM resulta evidente la formación de 9 reglas difusas que se pueden enunciar como sigue:

1. Si *El Nivel de Glucosa Postpandrial es Mucho Y El Consumo de Grasas Diarias es Alto* ENTONCES *el Consumo Calórico Diario Será Fraccionado Alto*

2. Si *El Nivel de Glucosa Postpandrial es Mucho Y el Consumo de Grasas Diarias es Medio* ENTONCES *el Consumo Calórico Diario Será Fraccionado Alto*

3. *Si El Nivel de Glucosa Postpandrial Es Mucho Y el Consumo de Grasas Diarias Es Bajo* ENTONCES *eL CONSUMO Calórico Diario Será Fraccionado Medio*

4. Si *El Nivel de Glucosa Postpandrial Es Regular Y el Consumo de Grasas Diarias es Alto* ENTONCES *el Consumo Calórico Diario Será Fraccionado Alto*

5. Si *El Nivel de Glucosa Postpandrial es Regular Y el Consumo de Grasas Diarias es Medio* ENTONCES *el Consumo Calórico Diario Será Fraccionado Medio*

6. Si *El Nivel de Glucosa Postpandrial es Regular* Y *el Consumo de Grasas Diarias es Bajo* ENTONCES *el Consumo Calórico Diario Será Fraccionado Bajo*

7. Si *El Nivel de Glucosa Postpandrial Es Poco Y el Consumo de Grasas Diarias es Alto* ENTONCES *el Consumo Calórico Diario Será Fraccionado Medio*

8. Si *El Nivel de Glucosa Postpandrial es Poco Y el Consumo de Grasas Diarias Es Medio* ENTONCES *el Consumo Calórico Diario Será Fraccionado Bajo*

9. Si *El Nivel de Glucosa Postpandrial es Poco* Y *el Consumo de Grasas Diarias es Bajo* ENTONCES *el Consumo Calórico Diario Será Fraccionado Bajo*

Por ejemplo, si se considera el caso de un paciente cuyo nivel de glucosa postpandrial es del orden de 200 mg/dl y el número de porciones en su consumo de grasas de origen animal es de 6 se observa que dichos valores corresponden a una Gp media y a una Gd media por lo que la única regla que se dispara es:

4. Si *El Nivel de Glucosa Postpandrial es Regular* Y *El Consumo de Grasas Diarias es Alto* ENTONCES *El Consumo Calórico Diario Será Fraccionado Alto*

De acuerdo con las expresiones correspondientes a las funciones de membresía de entrada se tienen los siguientes valores de verdad:

4. Si *El Nivel de Glucosa Postpandrial es* 1.0 *Regular Y El Consumo de Grasas Diarias es* 0.45 ***Alto*** ENTONCES *El Consumo Calórico Diario* 0.45 *Será Fraccionado Alto*

Por lo que la salida real de acuerdo con la ecuación (1) es:

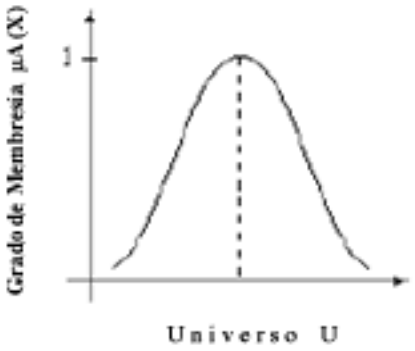
$$Gf = (2 * 0 + 5 * 0 + 8 * 0.45) / (0 + 0 + 0.45) = 8$$

La recomendación será indicar al paciente que realice 8 comidas al día sin rebasar las 1500 Kcal. por día.

VII. Conclusiones

En el campo médico las relaciones entre síntomas y enfermedades se conocen sólo con un cierto grado de certeza (la presencia de un conjunto de síntomas no siempre implica la presencia de una enfermedad). Estos tipos de problemas pueden también incluir algunos elementos deterministas, pero se trata fundamentalmente de problemas estocásticos.

Por lo anterior se sugiere una segunda versión del sistema en el cual se proponen funciones de membresía del tipo gaussiano como se indica a continuación.



Que opera bajo la expresión siguiente:

$$\mu_A(X) = e^{\frac{-(x^* - x_c)^2}{2\sigma^2}} \qquad (2)$$

Donde X^* : es el valor de la variable de entrada
X_c : es el valor de X donde la función Gaussiana es máxima.
σ : es la desviación estándar.

Este tipo de función de membresía permite un mejor manejo de la incertidumbre en cuanto a aquellas variables tanto de entrada como de salida donde sus rangos y relación no puede ser definida de forma determinística.

Sin embargo para el conjunto de casos determinísticos las aproximaciones mostradas en la sección de pruebas fueron satisfactorias y poseen la ventaja de poderse ajustar dinámicamente en rangos así como en las posibles modificaciones sobre la FAM que se persigue a fin de brindar un manejo adecuado del paciente diabético.

Con este sistema, el médico facilita mediante unos cuantos datos una fácil ruta a seguir por parte del paciente lo cual resulta en una disminución del gran desconocimiento que existe respecto al tratamiento de esta enfermedad.

Apéndice

Norma Oficial Mexicana, Nom-015-Ssa2-1994, "Para La Prevención, Tratamiento Y Control De La Diabetes Mellitus En La Atención Primaria".

Al margen un sello con el Escudo Nacional, que dice: Estados Unidos Mexicanos.- Secretaría de Salud .JOSE RODRIGUEZ DOMINGUEZ, Director General de Medicina Preventiva, por acuerdo del Comité Consultivo Nacional de Normalización de Servicios de Salud, con fundamento en los artículos 39 de la Ley Orgánica de la Administración Pública Federal; 3o. fracción XV, 13 apartado A) fracción I y III 158, 159, 160 y 161 de la Ley General de Salud, los artículos 38 fracción II , 46 fracción XI, 41, 43 y 47 de la Ley Federal sobre Metrología y Normalización y en el artículo 19 fracción II del Reglamento Interior de la Secretaría de Salud.

ÍNDICE

REFERENCIAS

[1] Castillo, E, Gutiérrez, J.M. and Hadi, A.S. (1997) Expert Systems and Probabilistic Network Models. Springer Verlag, New York. Versión castellana publicada por la Academia de Ingeniería (1998).

[2] Pearl, J. (1988) Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Morgan Kaufmann, San Mateo, CA.

[3] Jensen, F.V. (1996) An Introduction to Bayesian Networks. Springer-Verlag, New York.

Supporting the Continuity of Home Care and the Bidirectional Exchange of Data among Various Points of Care by Semantically Annotated Web Services

Maria Botsivaly and Basile Spyropoulos

***Abstract*—In this paper we report, first, the conceptualization and initial design of a system that creates a structured subset of data, concerning the most relevant facts about a patient's healthcare, organized and transportable, in order to be employed during the post-discharge homecare period, enabling simultaneously the planning and the optimal documentation of the provided homecare. Second, we present the actual development and implementation of the system according to the ASTM Continuity of Care Record (CCR) Specification. Finally, we present the implementation of a semantic-web-based system, which aims to facilitate the exchange of Clinical Information among various points of care, and we also present a solution that provides for the shared understanding of Medical Data between diverge information systems, and overcomes, both, the problems of incompatible formats in messages and of the use of diverse vocabularies.**

***Index Terms*—Home care, continuity of care record, semantically annotated Web services, data exchange among various points of care.**

I. Introduction

IT is highly anticipated that the continuous evolution of Information Technology during the last decades, in combination with the increase of mean life expectance and the hospital care cost avalanche, will eventually alter the way that health care is going to be delivered, and a significant proportion of health care will be provided in the near future in outpatient, community and homecare schemas. Nevertheless, as we move towards to this decentralized model, well argued concerns are raising about the fragmentation of patient's relevant information and the discontinuity in the delivered care [1]. Furthermore, especially in transitions from hospital to homecare, crucial questions emerge concerning the way this specific kind of care will be medically supervised and financially reimbursed.

It is generally expected that the Electronic Health Record will facilitate and simplify the exchange of information between different care providers and agencies, improving the quality and continuity of care. Nevertheless, a number of questions arise concerning the scope and the level of detail of information that should be exchanged when a patient is transmitted to a different care provider, especially in the case of transition from hospital to homecare. ASTM, an American National Standards Institute (ANSI) standard development organization, has recently approved the E2369-05, Standard Specification for Continuity of Care Record (CCR) [2]. CCR is intended to assure at least a minimum standard of health information transportability when a patient is discharged, referred or transferred, fostering thus and improving continuity in care.

In this article we report, first, the conceptualization and initial design of a system that creates a structured subset of data, concerning the most relevant facts about a patient's healthcare, organized and transportable, in order to be employed during the post-discharge homecare period, enabling simultaneously the planning and the optimal documentation of the provided homecare, and second, we present the actual development and implementation of the system according to the ASTM-CCR Specification.

Finally, an additional purpose of this study is the implementation of a semantic-web-based system, which aims to facilitate the exchange of Clinical Information among various points of care, and the presentation of a solution that provides for the shared understanding of Medical Data between diverge information systems, and overcomes, both, the problems of incompatible formats in messages and of the use of diverse vocabularies.

II. The Developed Continuity of Care Record

The CCR could be described as a proposed standard for an electronic form for patient transfer, referral, and discharge. Rather than a complete patient record, the CCR is designed to

Manuscript received May 13, 2008. Manuscript accepted for publication June 20, 2008

Maria Botsivaly is with Medical Instrumentation Technology Department, Technological Education Institute of Athens, 12210 Athens, Greece.

Basile Spyropoulos is with Medical Instrumentation Technology Department, Technological Education Institute of Athens, 12210 Athens, Greece (phone: +302109811964; fax: +302109811964; e-mail: basile@teiath.gr).

provide a snapshot in time containing the pertinent clinical, demographic, and administrative data for a specific patient. It is a way to create flexible documents that contain the most relevant core clinical information about a patient, and to send these electronically from one provider to another or to provide them directly to patients.

The CCR consists of three core components, the header, the body and the footer, each one consisting by a number of sections, covering the most important aspects of a patient's health condition. The sections consisting the CCR include: Patient and provider information; Insurance information; Patient's health status (allergies and other alerts, medications, medical equipment / external medical devices used by the patient, immunizations, vital signs, results, and recent procedures); Recent care provided and Recommendations for future care (care plan).

The CCR is designed to be technology and vendor neutral for maximum applicability. It must be developed on the extensible markup language (XML) platform in order to offer multiple options for its presentation, modification, and transmission. Through XML, CCR can be prepared, transmitted, and viewed in a browser, in an HL7 - CDA compliant document, in a secure email and in any XML-enabled application. The widespread use of the CCR will improve continuity of patient care, enhance patient safety, reduce medical errors, reduce costs, enhance communication and exchange of health information and standardize patient care information across healthcare providers.

It is actually anticipated that CCR will facilitate and stimulate more rapid EHR development, as an essential and simple building block.

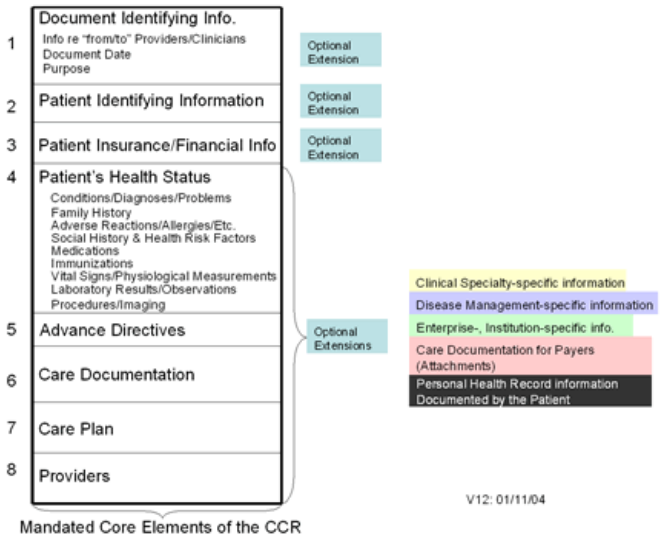


Fig. 1. Core elements of the CCR [3].

The developed system consists of two modules. The first module is responsible for the creation of a typical CCR that contains the appropriate demographic and administrative data, as well as the relevant clinical information, while the second module is responsible for the creation of a homecare plan which will be included in the Care Plan section of the CCR. The system is intended to be used upon the transition of a patient from hospital to homecare, although the first module alone could actually be used in any case of transition or referral.

The typical–CCR module can either collect the necessary data from an already installed EHR system or allow the user to enter the data manually by filling special forms. In any case, the user decides which parts of the patient's medical record (electronic or paper) are the most significant ones or are the necessary ones for the description of the current health status of the patient and should be included in the CCR.

The second module is responsible for the creation of the homecare plan by creating a structured subset of data, containing the diagnostic, monitoring, treatment, and nursing activities that should be employed during the post-discharge home-care period. The actual flow diagram of the developed system is illustrated in Figure 2. The developed model allows for every Hospital Department or Medical/Nursing group, to individually assign an appropriate set of homecare activities to specific diagnoses codes that are coded according to Diagnosis Related Group (DRG) codification.

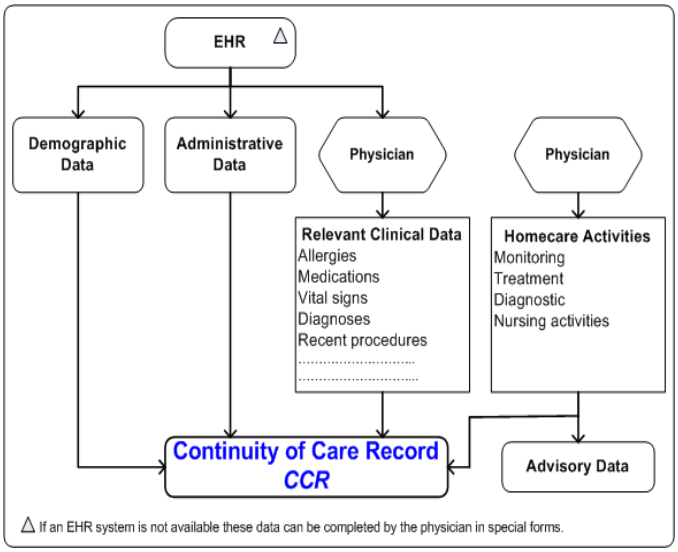


Fig. 2. Flow-Chart of the developed system.

These activity sets consist of diagnostic, monitoring and treatment activities that can be actually performed in home-environment, together with an appropriate nursing – activity treatment plan. These profiles of home-care activities are custom-made and every user, i.e. every physician responsible for discharging a patient from hospital, is actually allowed to set up his own profiles.

During the formation of these profiles the user can attach to each activity a set of nominal fees. This set of fees consists firstly of the official Insurance Agencies reimbursement amount, and, secondly, by a currently valid financial rate. This later is estimated by a software tool that we have already developed and allows for a rational approximation of the effective mean cost for several elementary medical activities, over different medical specialties [4], [5]. Thus, the developed system ignites, when relevant, the corresponding revision of an implicitly associated latent financial record that allows for an approximation of the individual case-cost.

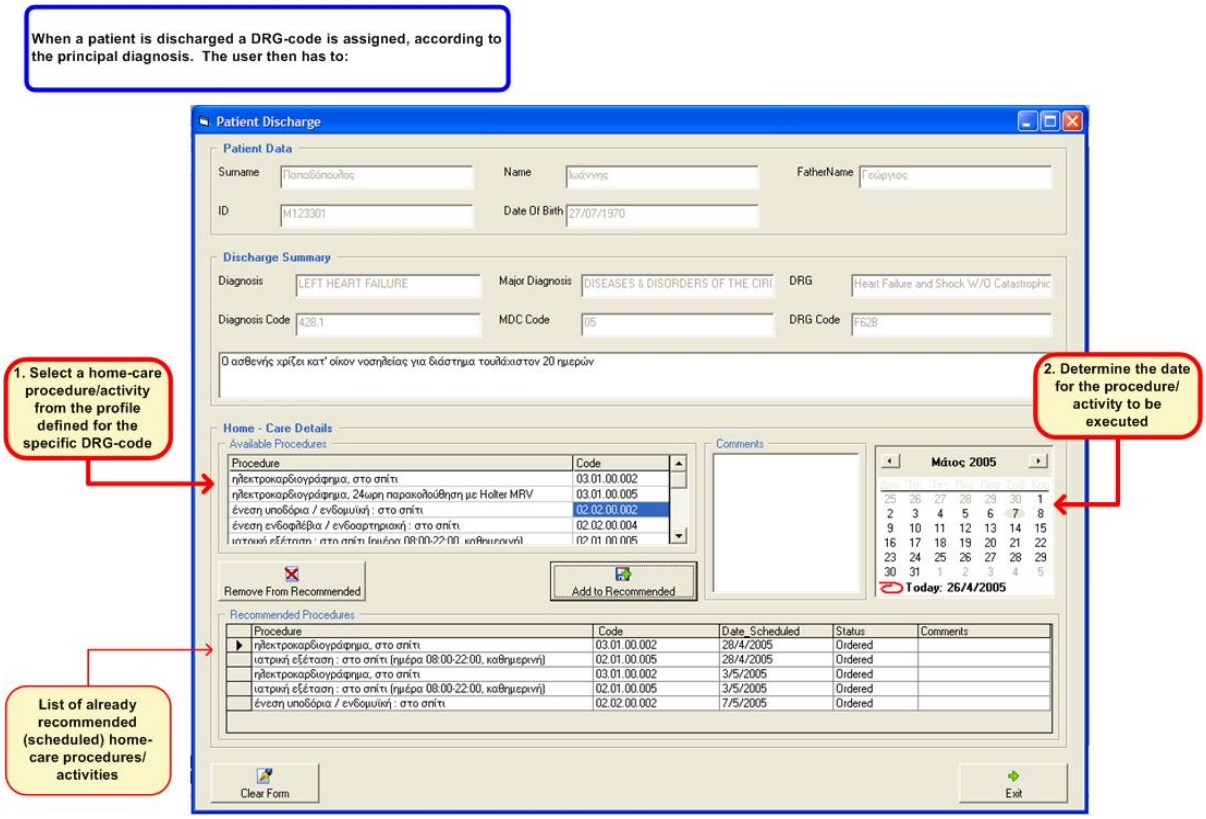


Fig. 3. Homecare activities selection for a specific patient.

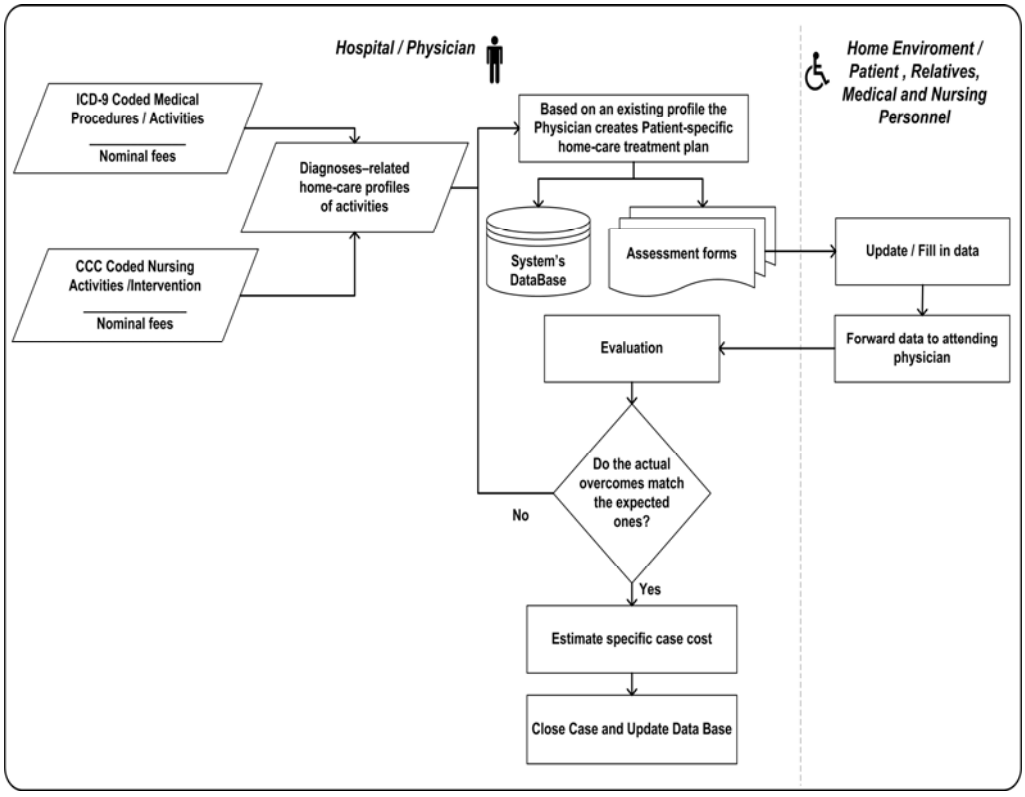


Fig. 4. Details of system's flow chart.

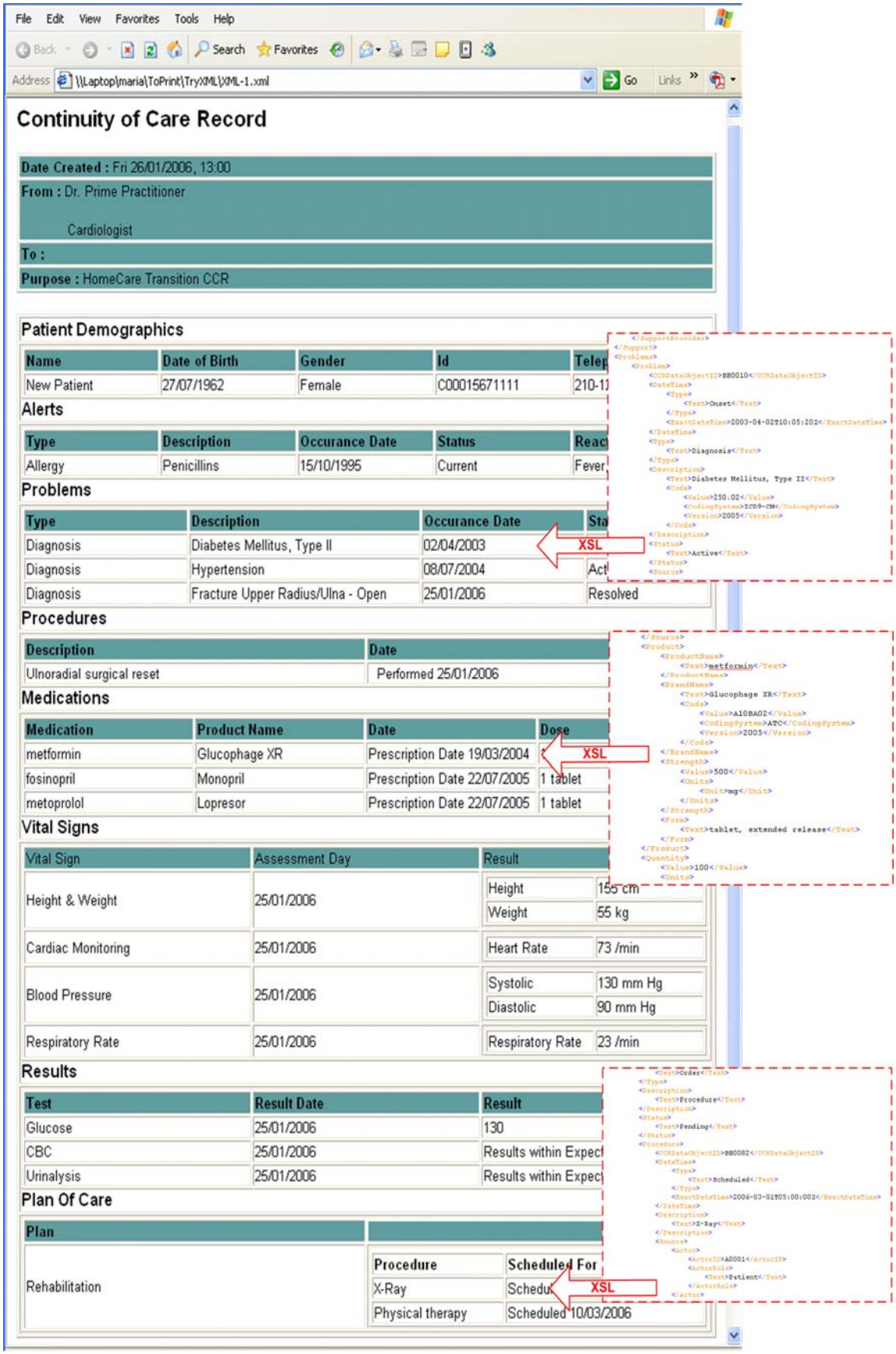


Fig. 5. Details of system's flow chart.

Upon the actual discharge of a patient the physician can use one of the predefined profiles, create a new one or modify an existing one in order to adapt his home-care profiles to specific instances and to emerging new demands. The scheduled procedures are automatically inserted in the CCR in the section of Care Plan. However, the system, apart from producing, electronically or in paper – format, the CCR, also produces a number of additional forms, including advisory and informational notes for the patient himself or for his relatives and diagrams of physiologic measurements, such as glucose, blood pressure etc. that the patient should monitor.

The system also provides for the production of forms that will be filled by the nursing personnel during the care visits in order to document their activities. The filled forms, both the ones regarding the nursing activities and interventions and the ones regarding the monitoring of physiological parameters, are returned to the responsible physician who evaluates them and, depending on his evaluation, can modify the care – plan of the specific patient in any suitable way.

The structure and data of the produced CCR are complying with the ASTM E2369-05 Specification for Continuity of Care Record, while XML is used for the representation of the data. The XML representation is made according to the W3C XML schema proposed by ASTM [6]. The CCR that is produced by the system is currently automatically transformed to HTML format, using the Extensive Stylesheet Language (XSL), in order to be viewable and printable.

It should be mentioned here that the diagnostic and treatment activities are classified according to International Classification of Diseases Version 9 (ICD9), while the Australian Refined DRGs (AR-DRGs) have served for the case codification, and the Nursing Interventions taxonomy of the Clinical Care Classification (CCC) system [7] was used for the documentation of nursing activities.

III. The Semantically Annotated Web Service

Interoperability of health care information systems has become one of the most crucial and challenging aspects in the healthcare domain [8], [9]. Medical data integration is currently a difficult task since the existing health information systems still operate in an isolated mode. Each information system employees currently its own vocabularies and knowledge bases and represents the data in different formats [1], [10].

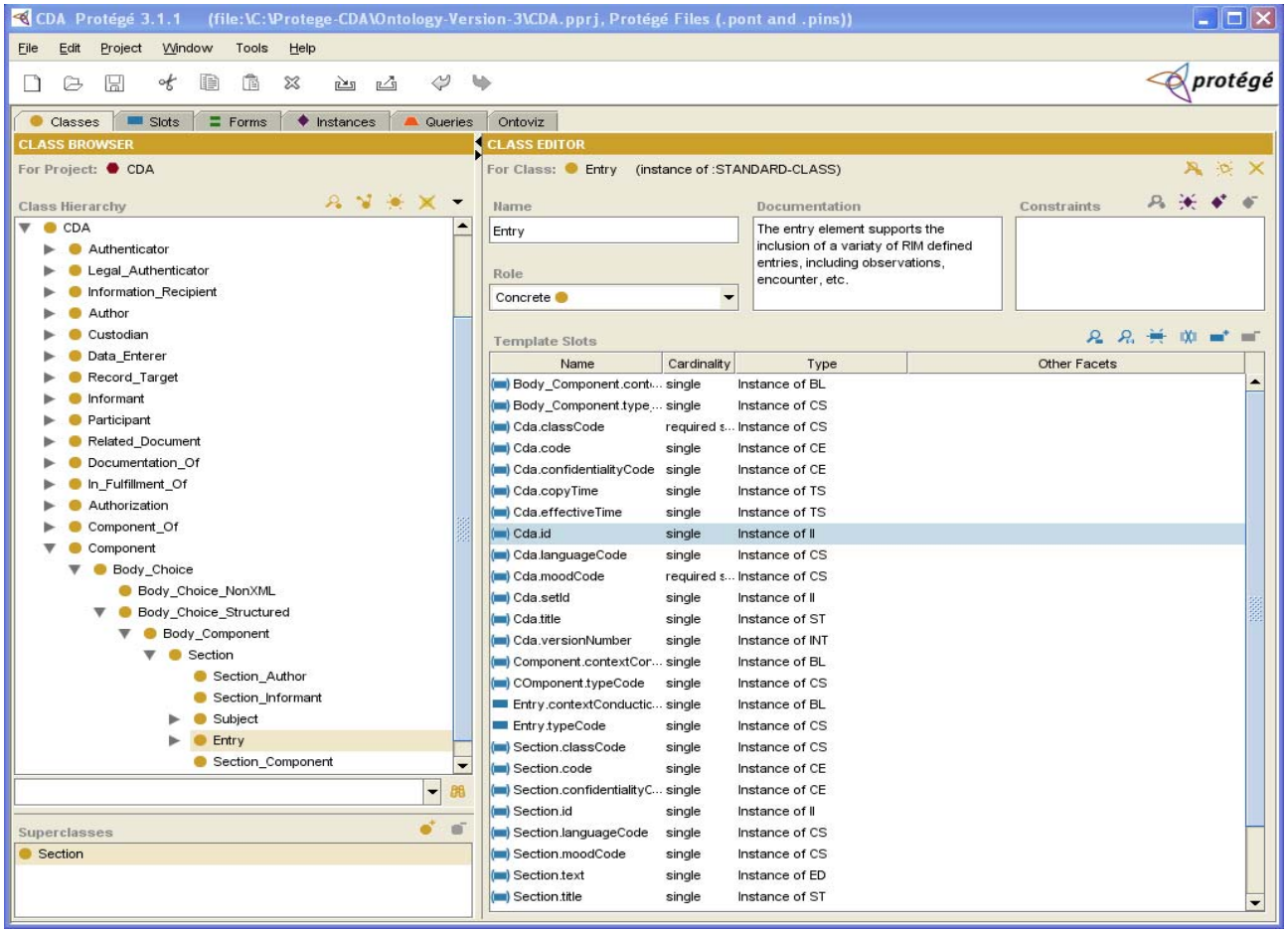


Fig. 6. The CDA class of the ontology

Clinical terminologies and vocabularies, such as SNOMED, ICD-9, ICD-10, and LOINC are already in use for several years and they provide a well established description of the medical domain knowledge. Other healthcare standards,

like the HL7 and the openEHR address the problem of structuring the formats of electronic data exchange among different information systems and of defining the contents of patients' records. These standards provide a certain degree of interoperability and are already in use by numerous healthcare organizations, with the HL7 version 2 being today the most widely implemented medical information standard worldwide [1], [9].

The HL7 Clinical Document Architecture (CDA) [11], [12] in particular is already in use by several countries [13], [14], since it provides for a common representation of clinical documents, enables the clinical document exchange and facilitates document management [15]. HL7-CDA contributes significantly to semantic interoperability, by allowing the structured use of controlled terminologies and promotes the sharing understanding of both the structure and the semantics of clinical documents that are created by diverse information systems. Nevertheless, HL7-CDA cannot by itself be a solution to the interoperability problem since it is unrealistic to expect that all care providers will agree on adopting a single standard and there is currently lack of a globally accepted terminology.

The solution for medical data integration appears to be the employment of computing technologies that are able to comprehend the semantics of the underlying data [1], [8]-[10], [16]. The emerging Semantic Web, which will employee semantically annotated Web Services and in which information will have a well defined machine – interpretable meaning, appears currently to be the most appealing approach towards this direction [16], [17]. At the same time, well established standardization efforts like the clinical vocabularies and the healthcare standards should not be ignored.

The developed system approaches the CDA-documents as domains of knowledge, which describe specific events of a case, such as, for example, a coronary angiography referral. The proper representation of the concepts of these documents, in terms of an ontology, provides for the shared understanding of the document, and allows for the creation of appropriately designed semantic Web Services, exceeding the problems of, both, incompatible formats in messages, and that of the use of diverse vocabularies.

The designed system consists of, first, a prototype ontology based upon the HL7 – CDA, and second, an application that converts the referral documents into a CDA – compliant format and the contents of the CDA – compliant documents into ontology instances. An appropriately designed semantically annotated Web service is responsible for the distribution of the documents over the network, by discovering existing instances of the ontology upon demand.

For the purposes of this paper, Referral ontology was designed, incorporating the HL7 – CDA healthcare standard. The hierarchy of the ontology was defined using the HL7 – RIM entities, the HL7 data types and vocabularies and the HL7-CDA R2 Hierarchical Description. These four concepts constitute the top-classes of the ontology and are further analyzed into a hierarchy of sub-classes, which describe the concepts that belong to these main categories. The vocabularies used were the ICD-9 and LOINC, which were inserted in the ontology as instances of the corresponding vocabularies' classes. However, any clinical vocabulary, such as SNOMED etc., may be employed.

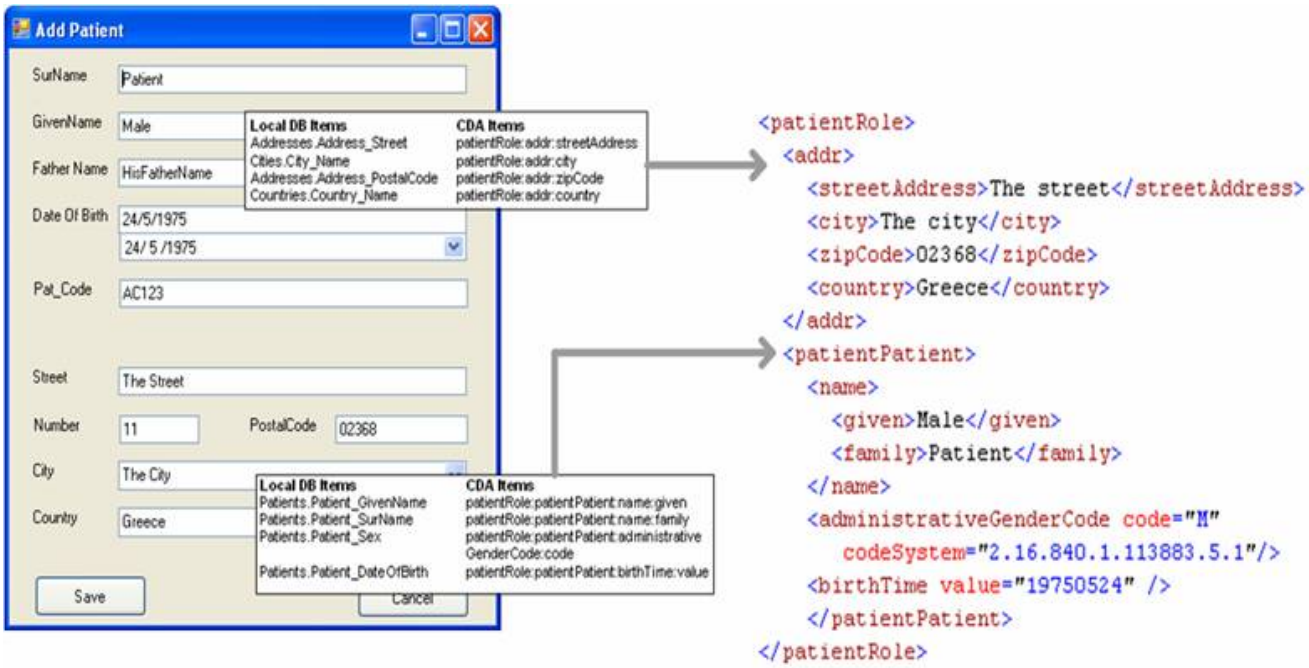


Fig. 7. The CDA – Compliant referral documents.

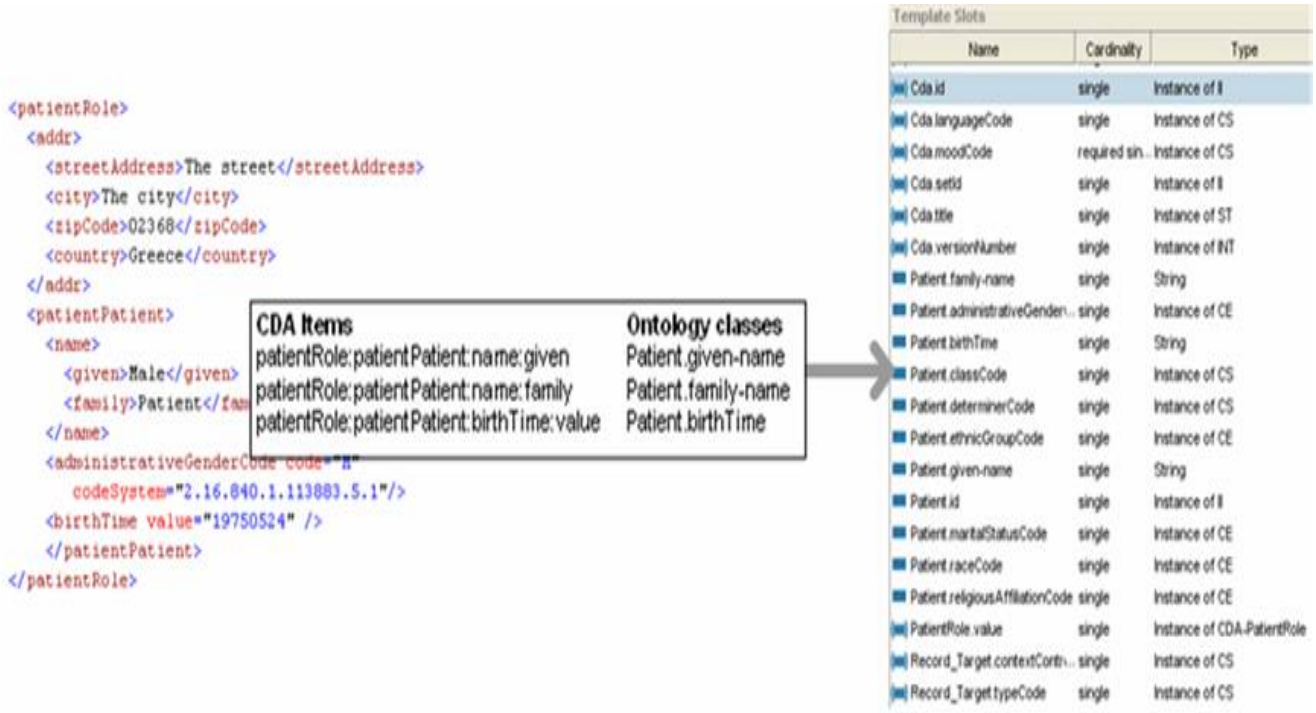


Fig. 8. The addition of ontology instances.

The referral documents, usually containing also static and/or dynamic images are created in a CDA – compliant format. The contents of the CDA – compliant referral documents are converted into ontology instances. An appropriately designed semantically annotated Web service is responsible for the distribution of the documents over the network, by discovering existing instances of the ontology upon demand. The developed service is currently a quite simple one, which enables the discovery of existing instances of the ontology upon the query of the appropriate PatientId.

The service is composed by two processes. The first process accepts as input the PatientId and gives an output that consists of a comprehensive list of the corresponding ontology instances (i.e. the documents that correspond to the specific patient) that were found in the ontology. This list provides for a general definition of the documents (creation date, healthcare provider, etc.). This list serves as the input for the second process whose output is the actual content of the Referral document.

IV. Discussion

The developed system of a Continuity of Care Record combined with a semantically annotated Web Service, is currently being laboratory tested with an EHR system that has been developed by our team. The laboratory implementation indicates that the system, whether interfaced to an EHR or not, is stable enough for practical use and it actually provides a simple, effective and easily expanded tool for the formation of both a CCR and a homecare plan, offering at the same time a good approximation of the individual case cost and a flexible HTML format for data representation, as it is illustrated earlier in Figure 5.

The implementation of the ASTM-CCR Specification Standard confirms that the specific protocol ensures indeed easy document production and manipulation while, at the same time, it assures at least a minimum standard of health information transportability. XML has proven to be the appropriate technology for such an application, since it renders the presentation of information flexible and generic enough to adapt to various users and various software platforms, with minimal custom programming.

Nevertheless, there are some issues concerning the actual use of the CCR that should be taken into consideration, the main one being the fact that, since the physician in charge is actually responsible for the selection of the appropriate / relevant clinical data that should be included in the record, there is always the possibility for the record to become information–polluted by unnecessary data. We believe that the establishment of diagnosis–specific pathways for the formation of special profiles that will support the physicians upon selecting the appropriate data could facilitate the use of CCR and prevent its main characteristic which is its summarized schema.

In the developed semantically annotated Web Service application, the parts of a referral document, usually containing also static and/or dynamic images, can be mapped in a corresponding taxonomy hierarchy, as defined in ICD9 and LOINC. However, any clinical vocabulary, such as SNOMED etc., may be employed. Although the developed service supports currently the discovery of complete documents, the next step will be the selective discovery of specific parts of the summaries, a goal supported by the implemented architecture.

The use of semantic web technologies and ontologies, together with the employment of well established healthcare standards and vocabularies are vital for the promotion of the

interoperability among diverge healthcare information systems. The flexible design concept and the adaptable retrieval mechanism of the proposed system allows for, first, any conceptualization of a continuity of care data exchange procedure, and second, the integration of the structured Referral and Medical Data, in any Electronic or Paper Patient Record System.

REFERENCES

[1] Orgun B, Vu J, "HL7 ontology and mobile agents for interoperability in heterogeneous medical information systems", Comput Biol Med., Vol 36 (7-8), pp 817-36, 2006.

[2] ASTM www.astm.org: E2369-05, Standard Specification for Continuity of Care Record.

[3] Tessier C, Continuity of Care Record, ASTM E31- WG on CCR, 21st TEPR 2004 (Towards an Electronic Patient Record), May 16-18, 2005, Salt Lake City, USA

[4] Botsivaly M, Spyropoulos B, Marinis M., Tzavaras A., Koutsourakis K., Sarantakis P. A Software-Tool allowing for Departmental Hospital Operational Cost Estimation. Proc. of EMBEC 2005 - 3rd European Conf. on Medical & Biological Engineering, Prague, Czech Republic, 2005.

[5] Spyropoulos B, Botsivaly M, Tzavaras A, Nikoloudakis G, Balabanis I, Karagiannis N, Limnou I Acquisition of cost-data for Surgery and Intensive medicine to be employed for drafting a DRGs hospital reimbursement system in Greece. Proc. of EMBEC 2005 - 3rd European Conf. on Medical & Biological Engineering, Prague, Czech Republic, 2005.

[6] ASTM www.astm.org: ADJE2369- Adjunct to E 2369 Continuity of Care Record (CCR).

[7] Saba V K, Home Health Care Classification of Nursing Diagnoses and Interventions. Washington, DC: Georgetown Univesity, 1994.

[8] Dogac A., Laleci G., Kirbas S., Kabak Y., Sinir S., Yildiz A., Gurcan Y., "Artemis: Deploying Semantically Enriched Web Services in the Healthcare Domain", Information Systems Journal (Elsevier), to appear [Available from : http//www.srdc.metu.edu.tr/webpage/ projects/artemis/publications/_Dogac_InfSys04.pdf].

[9] Bicer V., Laleci G., Dogac A., Kabak Y., "Providing Semantic Interoperability in the Healthcare Domain through Ontology Mapping", eChallenges 2005, Ljubljana, Slovenia.

[10] Eccher C., Purin B., Pisanelli D.M., Battaglia M., Apolloni I., Forti S., "Ontologies supporting continuity of care: The case of heart failure", Comput Biol Med., Vol 36, No 7-8, pp 789-801, 2006.

[11] Dolin R. H. "Clinical Document Architecture", HL7 International Affiliates Joint Meeting. August 2000, Dresden, Germany.

[12] Dolin R.H., Alschuler L., Boyer S., Beebe C., Behlen F.M., Biron P.V., Shabo A., "The HL7 Clinical Document Architecture", J Am Med Inform Assoc.; Vol 8 No 6, pp 552–561, 2001.

[13] Dolin R.H., Alschuler L., Boyer S., Beebe C., Behlen F.M., Biron P.V., Shabo A., "HL7 Clinical Document Architecture, Release 2", J Am Med Inform Assoc.; Vol 13, pp 30–39, 2006.

[14] Müller M., Ückert F., Bürkle T., Prokosch H., "Cross-institutional data exchange using the clinical document architecture (CDA)", International Journal of Medical Informatics, Vol 74, pp 245-256, 2005.

[15] Paterson G. I., Shepherd M., Wang X., Watters C., Zitner D., "Using the XML-based Clinical Document Architecture for Exchange of Structured Discharge Summaries", Proceedings of the 35th Hawaii International Conference on System Sciences, 2002.

[16] Della Valle E., Corizza D., Bicer V., Kabak Y., Laleci G. B., Lausen H., "The Need for Semantic Web Service in the eHealth", W3C Workshop on Frameworks for Semantics in Web Services, 2005. [Available from: http://www.w3.org/2005/04/FSWS/accepted-papers.html].

[17] Kamel Boulos M.N., Roudsari A.V., Carson E.R., "Towards a semantic medical Web: HealthCyberMap's tool for building an RDF metadata base of health information resources based on the qualified Dublin Core Metadata Set", Med Sci Monit, 2002; 8(7): MT 124-136.

Desarrollo de un Sistema Inmersivo de Realidad Virtual basado en Cabina Multipersonal y Camino sin Fin

Mauricio Olguín Carbajal, Israel Rivera Zarate, Oliver Pozas Quiteria

***Resumen*—El presente trabajo reporta los avances del desarrollo de un sistema inmersivo de realidad virtual que actualmente se esta desarrollando en el CIDETEC del IPN. El objetivo principal es generar un sistema de realidad virtual para el desarrollo de proyectos de realidad virtual de parte de estudiantes así como de profesores e investigadores. También se tiene como objetivo básico el que el CIDETEC pueda contar con un área para la enseñanza de la realidad virtual en un ambiente inmersivo.**

***Palabras clave*—Inmersión en realidad virtual, camino sin fin, cabina multipersonal.**

DEVELOPMENT OF THE SYSTEM FOR IMMERSING IN VIRTUAL REALITY BASED ON THE ENDLESS WALKING AND MULTIPERSONAL CABIN

***Abstract*—The present document reports the advances of the development for a Virtual Reality Inmersive System based on multipersonal cabin. This project is actually under development in the CIDETEC of the IPN. The main objective is to build a Virtual Reality Lab for the use in projects for researchers and students in the IPN. Also one of the basic goals of the project is development of the platform for development and teaching of virtual reality applications.**

***Index Terms*—Virtual reality immersion, endless walking, multipersonal cabin.**

Manuscrito recibido el 2 de febrero del 2008. Manuscrito aceptado para su publicación el 15 de junio del 2008.

M. Olguín Carbajal, Centro de Innovación y Desarrollo Tecnológico en Cómputo del Instituto Politécnico Nacional, México, D. F. (teléfono: 57296000 Ext. 52536; e-mail: molguin@ipn.mx).

I. Rivera Zarate, Centro de Innovación y Desarrollo Tecnológico en Cómputo del Instituto Politécnico Nacional, México, D. F. (teléfono: 57296000 Ext. 52535; e-mail: irivera@ipn.mx).

Oliver Pozas Quiteria, Centro de Innovación y Desarrollo Tecnológico en Cómputo del Instituto Politécnico Nacional, México, D. F. (teléfono: 57296000 Ext. 52536; e-mail: deltax17@hotmail.com).

I. INTRODUCCIÓN

Una forma de simulación sin precedentes es la realidad virtual, en ella la simulación toma una nueva dimensión, ya que no son solo simulaciones planas de datos, ni imágenes estáticas o video preempacado (como es el caso de la multimedia). Son simulaciones de audio y video en tiempo real.

Un sistema inmersivo logra que el usuario se sienta dentro del mundo virtual, en el caso de las cabinas de inmersión se pueden tener básicamente dos tipos de cabinas, las unipersonales y las multipersonales.

Entre las cabinas unipersonales mas usadas se encuentran las que simulan vehículos de conducción, como pueden ser simuladores de automotores, de aviones, naves espaciales o de barcos o submarinos. Estas cabinas generalmente sustituyen las ventanas del vehiculo por pantallas de computadora de alta resolución, de forma que se cubra todo el ángulo visual del usuario. Generalmente estos sistemas usan una sola computadora para calcular todas las vistas del entorno virtual.

A principios de los 80, Thomas A. Furness, científico de la fuerza aérea norteamericana, comenzó a desarrollar una cabina individual para entrenar a los pilotos de la base Wright-Patterson, en Ohio. La cabina contaba con un ángulo de visión de 120 grados lo cual proporciono una sensación de inmersión sin precedentes, ya que los sistemas existentes hasta ese momento solo contaban con 60 grados de campo de visión. Thomas A. Furnes dirige el Laboratorio de Tecnología de Interfaz Humana [1].

Actualmente dicha tecnología es básica para el entrenamiento de los pilotos de la fuerza aéreas norteamericana, así como para una gran parte de pilotos civiles en todo el mundo.

En las cabinas de inmersión multipersonales, se usan pantallas de proyección de gran tamaño, para una mayor sensación de inmersión, y proyectores posteriores para presentar las imágenes en las pantallas.

En 1992 la Universidad de Chicago demostró la Caverna (CAVE, Automatic Virtual Environment), en la conferencia SIGGRAPH. La Caverna es un sistema de proyección de realidad virtual multipersonal, basado en cabina de inmersión

desarrollado por el Laboratorio de Visualización Electrónica (Electronic Visualization Lab) [2].

Aquí en México la Universidad Autónoma de México cuenta con una sala inmersita de realidad virtual basada en una pantalla curva y en proyección para la Realidad Virtual [3].

El desarrollo que se propone en el presente trabajo es un sistema de inmersión multipersonal basado en cabina pero con un camino sin fin el cual debe proporcionar al usuario de la cabina una sensación de inmersión aun más grande que desarrollos anteriores.

A. *Sistemas de proyección*

Un sistema de proyección sencillo solo coloca la visión de un mundo virtual en una pantalla de proyección. El tamaño de la pantalla sirve para incrementar la sensación de inmersión, tal y como lo hace el cine.

Un sistema de proyección con cabina o "Caverna" involucra el uso de múltiples proyectores y pantallas que rodean al usuario en tres o cuatro lados, figura 3.

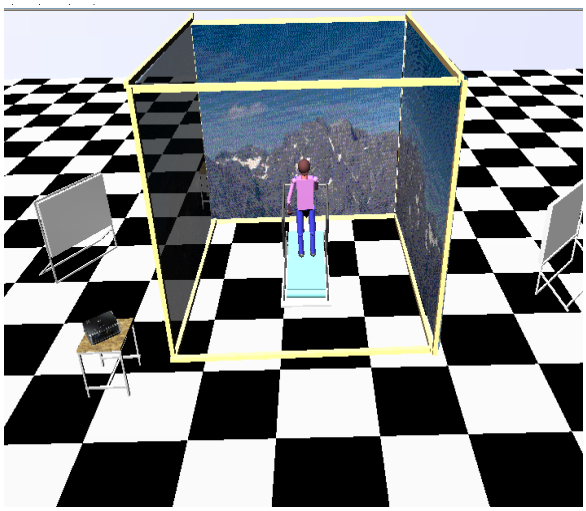


Figura 3. Sistema de cabina

Existe un proyector por cada pantalla, de forma que el usuario se sienta rodeado por el mundo. Los sistemas de cabina son muy útiles para pequeños grupos de usuarios ya que cada uno puede ver al mundo de forma simultánea, sin embargo tienen ciertas desventajas: requieren múltiples sistemas de proyección y muy grandes cantidades de poder de cómputo para generar todas esas imágenes al mismo tiempo, así como mucho espacio de suelo para el sistema en general. Estas limitaciones hacen poco practicas a las cabinas para el uso casero, pero para museos, escuelas, industria y otros lugares son ideales. Las cabinas son estereoscópicas por medio del uso de lentes para visión estereoscópica.

B. *Uso de la cámara de inmersión*

Los usos se pueden dar en cualquier campo, pero especialmente en la capacitación y la investigación, por ejemplo los sistemas de realidad virtual basados en cabinas se usan para entrenar a pilotos y astronautas.

Es posible usar un sistema de cabina para capacitación de técnicos o usuarios en herramienta y equipo especializado como sistemas de bombeo de presas, los cuales no pueden ser desplazados de su lugar y con los cuales no se pueden estar haciendo pruebas mientras el nuevo usuario aprende.

En la investigación un sistema de cabina tiene múltiples usos desde la investigación en física atómica, en biología celular, hasta la arquitectura y el diseño, pasando por la electrónica, las matemáticas, la informática, etc. Realmente todas las disciplinas que usan simulaciones son susceptibles de tener y desarrollar aplicaciones para un sistema como este.

II. DESARROLLO

A. *Motivación*

¿Por que construir una cabina?

La cabina se propone como una herramienta para la visualización científica.

Objetivos

- Desarrollar un sistema inmersivo de RV de bajo costo para un uso multidisciplinario dentro del Instituto.
- Crear un sistema de despliegue para la RV para el desarrollo de aplicaciones novedosas en un ambiente inmersivo.
- Llamar la atención de los estudiantes y profesores para que usen la RV en sus investigaciones y desarrollos.

B. *¿Como es una cabina?*

Una cabina de inmersión es un ambiente multipersonal, del tamaño de un cuarto, con imágenes de alta resolución, con audio y video en tres dimensiones. En la configuración propuesta las graficas son proyectadas en formato estereoscopio en las tres paredes y vistas con lentes para una visualización estereoscópica.

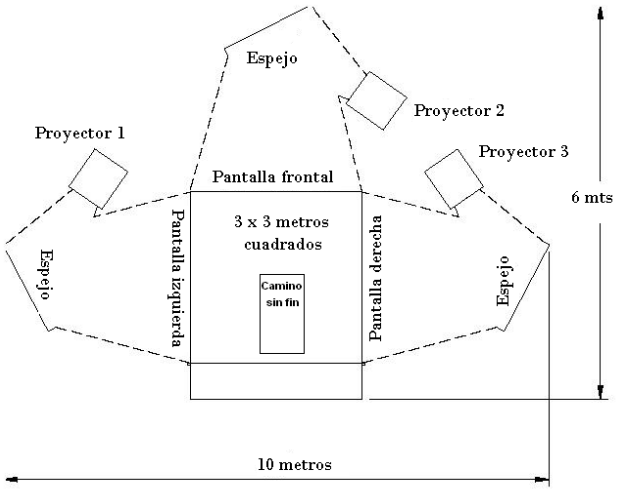


Figura 4. Esquema de cabina

En el diseño se incluye un camino sin fin, compuesto por una banda sin fin. De forma que el usuario pueda caminar dentro

del mundo virtual y al tiempo que avanza el mundo se mueva y se tenga una sensación de inmersión aun mayor. La propuesta de distribución de la cabina se muestra en la figura 4.

El camino sin fin es una propuesta del presente proyecto de investigación, y tiene el objetivo de solucionar algunos problemas de inmersión dentro de la cabina, uno de ellos es la movilidad del usuario y como al desplazarse pueden producir accidentes, tales como que al usuario se le "olvide" que esta dentro de un recinto cerrado y trate de caminar a través del mundo virtual rompiendo alguna de las pantallas de la cabina o, peor aun, causándose a si mismo un daño. Se tiene la hipótesis de que al permitirle al usuario "caminar" dentro del mundo virtual la sensación de inmersión será aun mayor pero sin riesgo para él o el equipo, figura 5.

Para otro tipo de entornos donde no sea necesario caminar, por ejemplo en desarrollos químicos, moleculares o modelado matemático el camino sin fin puede ser retirado para que la cabina sea usada como una cabina de inmersión convencional.

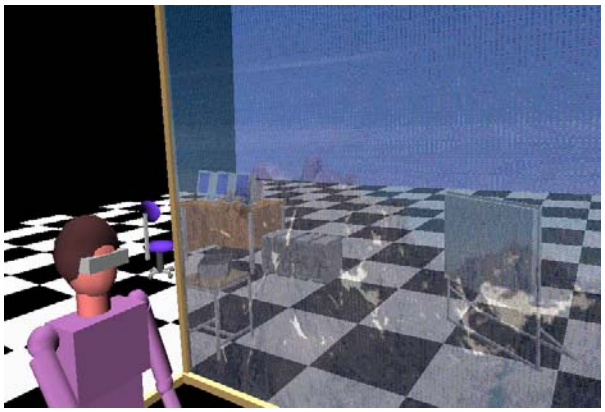


Figura 5. Cabina en uso

C. *Características*

Las proyecciones estereo y la correcta perspectiva del ambiente serán calculadas y actualizadas por un motor de realidad virtual formado por un cluster de computadoras. Las imágenes se moverán de forma sincronizada rodeando al usuario proyectando imágenes estereo de modelos 3D. Para el observador portando los lentes las pantallas de proyección se vuelven transparentes y la imagen 3D parece extenderse hasta el infinito.

D. *Funcionamiento*

En la cabina será posible simular, por ejemplo, un patrón de losetas que sean proyectadas en el suelo y paredes de tal forma que parezca un piso que se extiende al infinito, fuera de los límites del cuarto de proyección, al caminar sobre el suelo se desplazara acorde con el desplazamiento del usuario en el camino sin fin. Objetos tridimensionales como mesas y sillas que aparenten presentarse dentro y fuera del cuarto de proyección. Para el observador estos objetos estarán ahí para el mientras no intente tocarlos. Los sistemas basados en proyección, a diferencia de los de cascos, son ideales para presentaciones multipersonales, el único equipo adicional necesario son lentes estereo para cada usuario. Los participantes pueden compartir la experiencia virtual, mantener contacto visual, y comunicarse entre ellos de forma natural. Aunque solo un usuario controlará la cabina.

E. *Elementos*

- Se propone una cabina de 3x3x2.7 metros con pantallas translucidas, ver figura 4.
- Tres proyectores con una resolución de 1024 x 768 píxeles, los cuales proporcionaran la imagen periférica compuesta.
- Lentes estereo serán usados para separar las imágenes para cada ojo de forma alternativa.
- Un cluster de computadoras que será usado para calcular las imágenes proyectadas y para el sistema de rastreo así como la sincronización de todos los elementos.
-

F. *Motor*

El motor se puede desarrollar de dos formas básicamente:

1) Una sola computadora con mucho poder de cálculo y a una gran velocidad, con tres tarjetas de video, una para cada vista.

2) Un grupo de computadoras de capacidad media, encargada cada una de ellas de calcular la vista de una sola pantalla, sincronizadas todas por una cuarta computadora servidor, conectadas entre ellas por medio de una red local.

Se eligió la segunda opción para el presente desarrollo por dos motivos: Realizar una cabina de bajo presupuesto y usar equipo ya en existencia.

El motor consta de 4 computadoras Celeron:

- Velocidad 2.8 GHz
- Con memoria de RAM de 512 Mb
- Disco Duro de 20 Gb
- Tarjeta de sonido
- Tarjeta de red Ethernet 10/100

- API de Java 3D.

Tres computadoras son clientes del servidor de sincronización. Cada uno de los clientes se encarga de calcular una imagen de manera independiente, el servidor de sincronización se tiene la responsabilidad de informarle a cada cliente las características de cada imagen, de forma que todas tengan continuidad entre si. Los clientes y el servidor están conectados en una red dedicada independiente.

G. Objetivos

- Obtener la habilidad para mezclar imágenes y elementos virtuales y dispositivos reales (por ejemplo el desplazamiento y la mano del usuario)
- La necesidad de guiar y enseñar a otros de una forma razonable en mundos artificiales, figura 5.
- El deseo de unir la supercomputación y fuentes de datos para su mutuo crecimiento.

III. Conclusiones

La cabina de inmersión es un proyecto de investigación con las siguientes características:

- Titulo: Sistema inmersivo de realidad virtual basado en cabina y camino sin fin.
- Clasificación CONACyT:
  - Sector: Sector Educación
  - Subsector: Infraestructura

Se desarrollo un software de comunicación utilizando el modelo cliente-servidor, el servidor enviando los datos de la posición del usuario a los tres clientes encargados de calcular las imágenes para cada vista (izquierda, derecha y frente).

El software cliente servidor instalado en las cuatro PC constituyen el motor una parte básica del motor de realidad virtual formando un sistema de procesamiento distribuido.

El motor ya esta desplegando imágenes, sencillas, sincronizadas en tres monitores de PC.

Del desarrollo del motor se esta realizando una tesis de licenciatura y además de obtener una base para el sistema de inmersión se están generando recursos humanos capacitados para el uso y desarrollo del motor.

Se están desarrollando 2 tesis de Maestría que serán probadas en la cámara de inmersión. Una de ellas es una aplicación de aprendizaje matemático del cálculo diferencial y la otra es un desarrollo para la interfaz del guante P5 que permitirá a los usuarios manipular objetos dentro de la cabina de una forma más natural.

Se desarrollo un prototipo de camino sin fin el cual le permite al usuario caminar dentro de un mundo virtual.

El sistema de proyección se ha probado únicamente con materiales opacos y aun faltan pruebas en diferentes materiales translucidos.

Actualmente se han desarrollado algunos mundos a modos de prueba por investigadores participantes en el proyecto, un sistema solar y un estacionamiento y edificio de graduados de UPIICSA, que posteriormente servirán para las primeras pruebas.

El presente proyecto pretende sentar las bases en la investigación y desarrollo de herramientas para la realidad virtual en el CIDETEC, como parte de la materia y la línea de investigación de de RV. Hasta el momento los avances son alentadores y se espera que muy pronto se tenga la cabina armada en su totalidad para impartir clases apoyándonos en ella, así como desarrollos en investigaciones tanto propias como de otros investigadores y de alumnos.

Referencias

[1] http://www.hitl.washington.edu/home
[2] http://www.evl.uic.edu
[3] http://www.ixtli.unam.mx
[4] Stephen Matsuba & Bernie Roehl.Using VRML, Ed. QUE.

Implementación de Filtros Digitales Tipo FIR en FPGA

Jesús Antonio Álvarez Cedillo, Klauss Michael Lindig Bos, Gustavo Martínez Romero

***Resumen*—En este artículo se hace la descripción del diseño de un filtro digital tipo FIR con ocho bits de ancho de datos. Este sistema ha sido implementado en un FPGA (SPARTAN 3E de XILINX) y posee un software que realiza el cálculo de los coeficientes del filtro y la reconfiguración del hardware. Las pruebas se realizaron usando el programa MATHLAB para verificar su funcionamiento.**

***Palabras clave*—Filtros digitales, tratamiento digital de señales, FPGA, VHDL, FIR.**

IMPLEMENTATION OF DIGITAL FILTERS OF FIR TYPE IN FPGA

***Abstract*—This paper presents the description of development of digital filter of FIR type with eight bits data transmission. This system was implemented in FPGA (SPARTAN 3E by XILINX) and includes the software for calculation of filter coefficients and hardware reconfiguration. The experiments were conducted using simulation in MATHLAB.**

***Index Terms*—Digital filter, digital signal processing, FPGA, VHDL, FIR.**

I. INTRODUCCIÓN

Un filtro es un sistema que, dependiendo de algunos parámetros, realiza un proceso de discriminación de una señal de entrada obteniendo variaciones en su salida. Los filtros digitales tienen como entrada una señal analógica o digital y a su salida tienen otra señal analógica o digital, pudiendo haber cambiado en amplitud, frecuencia o fase dependiendo de las características del filtro.

El filtrado digital es parte del procesado de señal digital. Se le da la denominación de digital más por su funcionamiento interno que por su dependencia del tipo de señal a filtrar, así podríamos llamar filtro digital tanto a un filtro que realiza el procesado de señales digitales como a otro que lo haga de señales analógicas.

El filtrado digital consiste en la realización interna de un procesado de datos de entrada. El valor de la muestra de la entrada actual y algunas muestras anteriores (que previamente habían sido almacenadas) son multiplicadas por unos coeficientes definidos. También podría tomar valores de la salida en instantes pasados y multiplicarlos por otros coeficientes. Finalmente todos los resultados de todas estas multiplicaciones son sumados, dando una salida para el instante actual. Esto implica que internamente tanto la salida como la entrada del filtro serán digitales, por lo que puede ser necesario una conversión analógico-digital o digital-analógico para uso de filtros digitales en señales analógicas. Un elemento de prueba de estos circuitos típicamente es el ruido blanco (Figura 1).

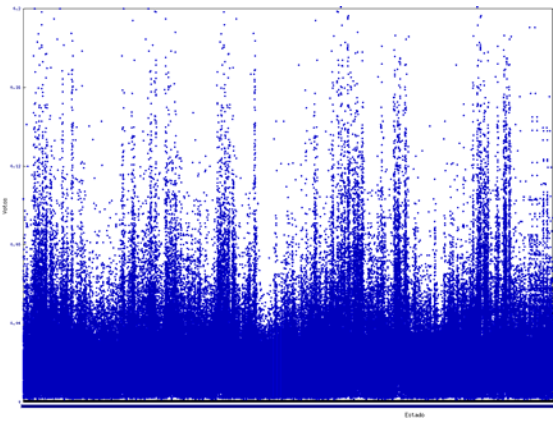


Fig. 1. Gráfica de ruido blanco

II. FILTROS FIR

Los filtros digitales se usan frecuentemente para tratamiento digital de la imagen o para tratamiento del sonido digital.

FIR es un acrónimo en inglés para *Finite Impulse Response* o *Respuesta finita al impulso*. Se trata de un tipo de filtros digitales en el que, como su nombre indica, si la entrada es una señal impulso, la salida tendrá un número finito de términos no nulos.

Para obtener la salida sólo se basan en entradas actuales y anteriores. Su expresión en el dominio **n** es:

Manuscrito recibido el 13 de marzo del 2008. Manuscrito aceptado para su publicación el 16 de junio del 2008.

J. A. Alvarez Cedillo, Centro de Innovación y Desarrollo Tecnológico en Cómputo del Instituto Politécnico Nacional, México, D. F. (teléfono: 57296000 Ext. 52535; e-mail: jaalvarez@ipn.mx).

K. M. Lindig Bos, Dirección de Cómputo y Telecomunicaciones del Instituto Politécnico Nacional, México, D. F. (teléfono: 57296000 Ext. 52536; e-mail: mlindig@ipn.mx).

G. Martínez Romero, Centro de Investigación e Innovación Tecnológica Unidad Azcapotzalco del Instituto Politécnico Nacional, México, D. F. (teléfono: 57296000 Ext. 52535; e-mail: gumartinezr@ipn.mx).

$$y_n = \sum_{k=0}^{N-1} b_k x(n-k)$$

En la expresión anterior **N** es el orden del filtro, que también coincide con el número de términos no nulos y con el número de coeficientes del filtro. Los coeficientes son **bk.**

La salida también puede expresarse como la convolución de la señal de entrada *x(n)* con la respuesta impulsional *h(n)*:

$$y_n = \sum_{k=0}^{N-1} h_k x_{n-k}$$

Aplicando la transformada Z a la expresión anterior:

$$H(z) = \sum_{k=0}^{N-1} h_k z^{-k} = h_0 + h_1 z^{-1} + \cdots + h_{N-1} z^{-(N-1)}$$

La estructura básica de un FIR se presenta en la Fig. 2.

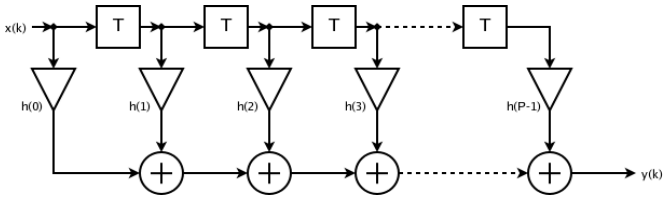


Fig. 2. Estructura básica de un FIR

En la figura 2 los términos **β** son los coeficientes y los **T** son retardos.

Pueden hacerse multitud de variaciones de esta estructura. Hacerlo como varios filtros en serie, en cascada, etc.

Hay tres métodos básicos para diseñar este tipo de filtros:

- Método de las ventanas. Las más habituales son:
  - Ventana rectangular
  - Ventana de Barlett
  - Ventana de Hanning
  - Ventana de Hamming
  - Ventana de Blackman
  - Ventana de Kaiser
- Muestreo en frecuencia.
- Rizado constante,

III. Antecedentes

Como ya se mencionó en líneas anteriores, es gracias al avance de las computadoras modernas que el tratamiento digital de señales se ha expandido y se ha hecho cada vez más fuerte. Han contribuido con su velocidad al montaje de algoritmos de procesamiento para lograr entregar respuestas casi instantáneas, siempre dentro de los límites exigidos por el desarrollo en las diferentes aplicaciones en que han sido utilizadas.

Las computadoras no sólo han influido en el montaje de algoritmos, además se han convertido en una herramienta fundamental para los diseñadores, ya que gracias a éstos se ha conseguido elaborar mejores y variados algoritmos para el diseño de filtros y herramientas de proceso. Una herramienta muy potente para el análisis de imágenes es el MATHLAB.

El diseño de filtros se ha apoyado en los dispositivos lógicos programables, los cuales han jugado un papel muy importante en el montaje de los filtros digitales, puesto que gracias a ellos se ha logrado un adecuado funcionamiento en tiempo real. El FPGA es uno de estos dispositivos, que posee la cualidad de la re-configuración, lo que permite realizar cambios en la arquitectura sin necesidad de producir variaciones en el montaje o en el software que se está operando.

IV. Factores de Implementación

La implementación de estos filtros está determinada por algunos factores que ayudan a la calificación de dichos sistemas, tales como:

1. Complejidad computacional, requisitos de memoria y longitud de palabra.
2. La Complejidad computacional: está determinada por el número de operaciones aritméticas necesarias para el cálculo de la salida, como sumas, multiplicaciones y divisiones.
3. Requisitos de memoria: hacen referencia a la cantidad de posiciones de memoria que son necesarias para almacenar elementos, tales como los coeficientes del sistema, entradas retrasadas, salidas retrasadas y algunos valores internos necesarios para el cálculo de la salida.
4. Longitud de palabra: se refiere a un efecto de precisión que se encuentra dado por la cuantificación, tanto de los coeficientes del filtro como de la señal de entrada. Este elemento se hace presente en filtros implementados en hardware y en software.

Las operaciones realizadas deben ser redondeadas o truncadas para poder ajustarse a las restricciones de operación del ordenador, en el caso del software, o a las características definidas por el diseñador del hardware digital.

V. Tipos de Filtros

Existen dos tipos básicos de filtros digitales:

- no recursivos
- recursivos.

Para los filtros no recursivos la función de transferencia contiene un número finito de elementos, cuya ecuación en diferencias es:

$$H(z) = \sum_{k=0}^{M-1} b_k z^{-k}$$

Y su equivalente en función de transferencia es:

$$H(z) = \frac{\sum_{k=0}^{M} a_k z^{-k}}{1 - \sum_{k=1}^{N} b_k z^{-k}} = \frac{a_0 + a_1 z^{-1} + a_2 z^{-2} + \ldots + a_n z^{-n}}{1 - b_1 z^{-1} + b_2 z^{-2} + \ldots + b_n z^{-n}}$$

Esta clase de sistemas se caracteriza por no poseer realimentaciones, de lo cual se observa que la salida se encuentra dada en función de la entrada y de sus respectivos retrasos.

Para los filtros recursivos la ecuación en diferencias se encuentra expresada en función de dos formas polinomiales:

$$y(n) = -\sum_{k=1}^{N} a_k y(n-k) + \sum_{k=0}^{M} b_k x(n-k)$$

Esta ecuación nos lleva a encontrar una función de transferencia de la forma:

$$H(z) = \frac{\sum_{k=0}^{M} a_k z^{-k}}{1 - \sum_{k=1}^{N} b_k z^{-k}} = \frac{a_0 + a_1 z^{-1} + a_2 z^{-2} + \ldots + a_n z^{-n}}{1 - b_1 z^{-1} + b_2 z^{-2} + \ldots + b_n z^{-n}}$$

A los primeros pertenecen los filtros tipo FIR, caracterizados por no poseer realimentación, y a los segundos los filtros tipo IIR, en donde la salida se encuentra dada en función de la entrada y de las salidas en instantes previos.

VI. Implementación en el FPGA

Se creó una herramienta en hardware como filtro tipo FIR, esta fue probada en una tarjeta XILINX SPARTAN 3E. El kit de desarrollo se muestra en la figura 3.

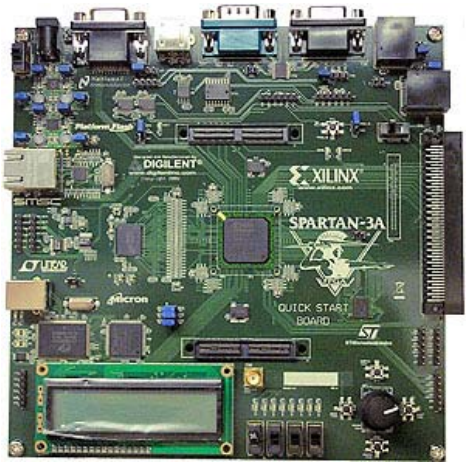


Fig. 3. Kit de desarrollo XILINX SPARTAN 3E

La entidad fue definida de la siguiente forma:

```
ENTITY FIR IS   GENERIC (N : natural := 16  );
PORT (   SIGNAL CLK : IN std_logic;
SIGNAL RES_n : IN std_logic;
SIGNAL X : IN std_logic_vector( N-1 DOWNTO 0 );
SIGNAL Y : OUT std_logic_vector( N-1 DOWNTO 0 )   );
END ENTITY FIR ;
```

Como es posible observar se mantiene la señal de reloj CLK como entrada, dos valores de ingreso donde X es dato de 2 bits y RES_n es un reset activado por pulso. La salida se encuentra definida por un valor de dato de 2 bits. Tal como se muestra en la figura 4.

Fig. 4. Entidad de un filtro Fir en FPGA XILINX

La arquitectura se define de la siguiente manera:

```
ARCHITECTURE RTL OF FIR IS
  TYPE t_operacion IS ARRAY (7 DOWNTO 1) OF std_logic_vector(N-1
DOWNTO 0);

  SIGNAL operacion : t_operacion;
  SIGNAL add_01 : std_logic_vector(N DOWNTO 0);
  SIGNAL add_23 : std_logic_vector(N DOWNTO 0);
  SIGNAL add_45 : std_logic_vector(N DOWNTO 0);
  SIGNAL add_67 : std_logic_vector(N DOWNTO 0);
  SIGNAL add_0123 : std_logic_vector(N+1 DOWNTO 0);
  SIGNAL add_4567 : std_logic_vector(N+1 DOWNTO 0);
  SIGNAL add_all : std_logic_vector(N+2 DOWNTO 0);
 SIGNAL vystup : std_logic_vector(N+2 DOWNTO 0);
 SIGNAL pom    : std_logic_vector(N+2 DOWNTO 0);
 SIGNAL pipe_0123 : std_logic_vector(N+1 DOWNTO 0);
SIGNAL pipe_4567: std_logic_vector(N+1 DOWNTO 0);
BEGIN
```

Como es posible observar las señales de operación son diversas, la idea principal es mostrar el manejo del pipeline, el circuito presentará el esquema de la figura 5.

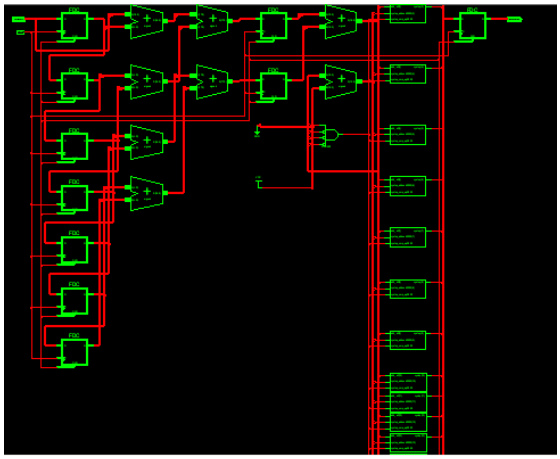


Fig. 5. Esquema RTL

El pipeline característico se codifico de la siguiente manera:
Si es presionado el reset RES_n el pipeline inicia su operación, si existe el evento del reloj y es un pulso positivo, entonces realiza las operaciones retardas y secuenciales.

```
 pipeline: PROCESS (CLK, RES_n)
 BEGIN
 IF RES_n='0' THEN
   pipe_0123 <= (OTHERS => '0');
```

```
    pipe_4567 <= (OTHERS => '0');

   ELSIF CLK='1' AND CLK'EVENT THEN
     pipe_0123 <= add_0123;
     pipe_4567 <= add_4567;
  END IF;
  END PROCESS pipeline;
```

Las siguientes instrucciones muestran el bloque de cada procedimiento u operación del pipeline, cabe resaltar el manejo de los índices en el filtro.

```
    add_01 <= (X(N-1) & X) + (operacion(1)(N-1) & operacion(1));
    add_23 <= (operacion(2)(N-1) & operacion(2)) + (operacion(3)(N-1) &
operacion(3));
    add_45 <= (operacion(4)(N-1) & operacion(4)) + (operacion(5)(N-1) &
operacion(5));
    add_67 <= (operacion(6)(N-1) & operacion(6)) + (operacion(7)(N-1) &
operacion(7));

    add_0123 <= (add_01(N) & add_01) + (add_23(N) & add_23);
    add_4567 <= (add_45(N) & add_45) + (add_67(N) & add_67);
  add_all <= (pipe_0123(N+1) & pipe_0123) + (pipe_4567(N+1) &
pipe_4567);

    pom(3 DOWNTO 0) <= "0100";
    pom(N+2 DOWNTO 4) <= (OTHERS => '0');

    vystup <= add_all WHEN add_all(3 DOWNTO 0) = "0100" else  add_all
+ pom;
```

VII. Pruebas y Resultados

Se creó una herramienta de software que se encarga de brindar una conexión sencilla y amable entre el usuario del filtro y el hardware.

La interfase se encuentra desarrollada sobre Matlab 6.0 y al inicio presenta 3 posibilidades de filtros tipo FIR para escoger con cuál se desea trabajar:

- Filtro con coeficientes estáticos.
- Filtro con coeficientes dinámicos.
- Filtro adaptativo.

Para efectos de este artículo sólo se trabaja con la opción de filtro con coeficientes dinámicos.

Esta sección escogida presenta dos bloques principales así:

Diseño del filtro: esta es la primera sección, le permite al usuario manejar las especificaciones del filtro tales como:

- Cantidad de coeficientes.
- Ancho de banda.
- Atenuación y ganancia del filtro para diferentes frecuencias, como se muestra en la figura 6.

Una vez se han escogido los diferentes parámetros del filtro, tanto para su arquitectura como para su funcionamiento, se procede a diseñar dicho filtro. Proceso que se lleva a cabo mediante la utilización del algoritmo FIR.

La verificación e implementación: se realizan una vez que se han determinado las especificaciones del filtro y se ha diseñado. Se observa una nueva ventana (figura 7), la cual muestra la respuesta en frecuencia del filtro. En esta ventana se presentan tres respuestas espectrales diferentes así:

- La deseada por el usuario.
- La lograda por el algoritmo
- La obtenida luego de redondear los valores de los coeficientes a los valores permitidos por la resolución de 8 bits.

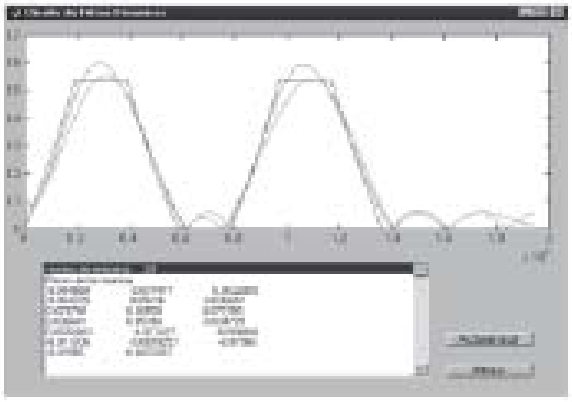


Fig. 6. Muestra de la interfase para el diseño de los filtros

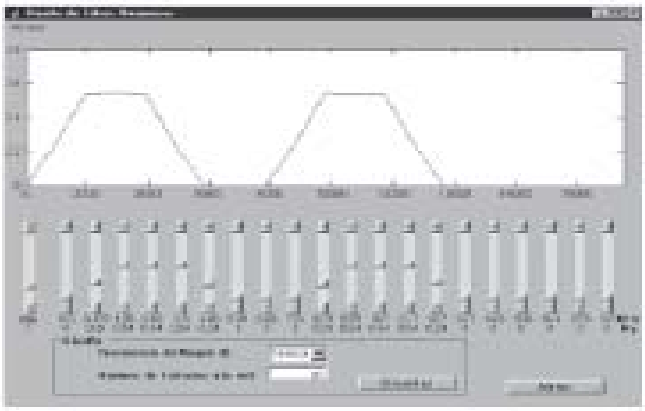


Fig. 7. Ventana de verificación e implementación del filtro

VIII. Conclusiones

El diseño se realizó en forma jerárquica, de tal manera que se dividió en diferentes etapas.

Antes de iniciar el estudio de cada una de estas entidades deberá de ser necesario tener en cuenta algunos aspectos del funcionamiento que presenta este diseño:

- El ancho de palabra es de 8 bits, lo que obliga a las memorias, al sumador y al acumulador a trabajar con esta especificación.
- Debido a que el ancho de palabra es de 8 bits, se toma como mínimo intervalo 1/128, de tal forma que el formato varía desde 0.996 hasta –1. De esta forma se logra reducir el problema de truncamiento que se experimenta cuando se trabaja con números de formato entero.
- Para el reloj del sistema se utilizó el oscilador interno del FPGA, que es de 50 Mhz, al que se le acondiciona un contador para poder dividir la frecuencia y así obtener diferentes posibilidades en el ancho de banda del filtro, como: 77.8 Khz, 39Khz, 19.5 Khz, 9.7 Khz, 4.8 Khz, 1.2Hhz, 600Hz y 150Hz. Estas frecuencias son

seleccionadas por el usuario desde el exterior de la arquitectura.

– La arquitectura presenta una etapa que se encarga de comunicarse con el software y, de esta forma, mediante una interfase de puerto paralelo a la PC.

REFERENCIAS

[1] Willis J. Tompkins. Biomedical digital signal procesing.Prentice Hall, may, 1993.

[2] Xilinx The progamable logic databook. SPARTAN3 , marzo 2 de 2007.

[3] J. G., Proakis, Dimitris G. Manolakis. Tratamiento digital de señales. Prentice Hall, 1997.

[4] Samuel Stearms, Ruth A. David. Signal procesing algoritms". Prentice Hall, 1997.

[5] Binary numbering systems. Altera Corporation, 1997

[6] R. W. Hamming. Digital filters. Prentice Hall, 1989.

[7] Using select-RAM memory in XC4000 series FPGA's". Xilinx aplication note, july 7, 1996.

[8] A CPLD VHDL introduction. Xilinx application note, january 12, 1998.

[9] Digital signal processing toolbox. The MathWorks Inc., 1992-2001.

Journal Information and Instructions for Authors

I. Journal Information

"*Polibits*" is a half-yearly research journal published since 1989 by the Center for Technological Design and Development in Computer Science (CIDETEC) of the National Polytechnic Institute (IPN) in Mexico City, Mexico. The journal solicits original research papers in all areas of computer science and computer engineering, with emphasis on applied research.

The journal has double-blind review procedure. It publishes papers in English and Spanish.

Publication has no cost for the authors.

A. Main topics of interest

The journal publishes research papers in all areas of computer science and computer engineering, with emphasis on applied research.

More specifically, the main topics of interest include, though are not limited to, the following:

- Artificial Intelligence
- Natural Language Processing
- Fuzzy Logic
- Computer Vision
- Multiagent Systems
- Bioinformatics
- Neural Networks
- Evolutionary algorithms
- Knowledge Representation
- Expert Systems
- Intelligent Interfaces: Multimedia, Virtual Reality
- Machine Learning
- Pattern Recognition
- Intelligent Tutoring Systems
- Semantic Web
- Database Systems
- Data Mining
- Software Engineering
- Web Design
- Compilers
- Formal Languages
- Operating Systems
- Distributed Systems
- Parallelism
- Real Time Systems
- Algorithm Theory
- Scientific Computing
- High-Performance Computing
- Geo-processing
- Networks and Connectivity
- Cryptography
- Informatics Security
- Digital Systems Design
- Digital Signal Processing
- Control Systems
- Robotics
- Virtual Instrumentation
- Computer Architecture
- other.

B. Indexing

The journal indexing is in process.

II. Instructions for Authors

A. Submission

Papers ready to review are received through the Web submission system www.easychair.org/polibits

The papers can be written in English or Spanish.

Since the review procedure is double-blind, the full text of the papers should be submitted without names and affiliations of the authors and without any other data that reveals the authors' identity.

For review, a file in one of the following formats is to be submitted: PDF (preferred), PS, Word. In case of acceptance, you will need to upload your source file in Word (for the moment, we do not accept TeX files, if you are interested to submit a paper in TeX, please, contact the editor). We will send you further instructions on uploading your camera-ready source files upon acceptance notification.

Deadline for the nearest issue (January-June 2009): January 15, 2009. Papers received after this date will be considered for the next issue (July-December 2009).

B. Format

Please, use IEEE format[1], see section "Template for all Transactions (except IEEE Transactions on Magnetics)". The editors keep the right to modify the format of the final version of the paper if necessary.

We do not have any specific page limit: we welcome both short and long papers, provided the quality and novelty of the paper adequately justifies the length.

Submissions in another format can be reviewed, but the use of the recommended format is encouraged.

In case of being written in Spanish, the paper should also contain the title, abstract, and keywords in English.

[1] www.ieee.org/web/publications/authors/transjnl/index.html