A Comparative Evaluation of an Ontological Medical Decision Support System (OMeD) for Critical Environments*

John A. Doucette
David R. Cheriton School of Computer Science, University of Waterloo
200 University Avenue West
Waterloo, ON, Canada
j3doucet@cs.uwaterloo.ca

Atif Khan
David R. Cheriton School of Computer Science, University of Waterloo
200 University Avenue West
Waterloo, ON, Canada
atif.khan@uwaterloo.ca

Robin Cohen
David R. Cheriton School of Computer Science, University of Waterloo
200 University Avenue West
Waterloo, ON, Canada
rcohen@cs.uwaterloo.ca

ABSTRACT

Modern medical decision making systems require users to manually collect and process information from distributed and heterogeneous repositories to facilitate the decision making process. There are many factors (such as time, volume of information and technical ability) that can potentially compromise the quality of decisions made for patients. In this work we demonstrate and evaluate a new medical decision making support system, called OMeD, which automatically answers medical queries in real time, by collecting and processing medical information. OMeD utilizes a natural-language-like user interface (for querying) and semantic web techniques (for knowledge representation and reasoning) to answer queries. We compare OMeD to a set of standard machine learning techniques across a series of benchmarks based on simulated patient data. The conventional techniques attempt to learn the answer to a query by analyzing simulated patient records. The sparsity of the simulated data leads conventional techniques to frequently misidentify the relationships between medical concepts. In contrast, OMeD is able to reliably provide correct answers to queries. Unlike conventional automated decision support systems, OMeD also generates independently verifiable proofs for its answers, providing healthcare workers with confidence in the system's recommendations.

Categories and Subject Descriptors

J.2 [**Life and Medical Sciences**]: Health; I.2.4 [**Knowledge Representation Formalisms and Methods**]: Semantic networks; H.3.4 [**Systems and Software**]: Question-answering (fact retrieval) systems

General Terms

Experimentation, Performance

Keywords

Evidence based medical decision support system, Semantic knowledge representation and reasoning, Question answering, Health information systems, Machine Learning

*This research was funded in part by the National Science and Engineering Research Council of Canada (NSERC).

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
IHI'12, January 28–30, 2012, Miami, Florida, USA.
Copyright 2012 ACM 978-1-4503-0781-9/12/01 ...$10.00.

1. INTRODUCTION

Electronic medical data is becoming increasingly important in medical decision making. Consequently, health information systems have begun to play a dominant role in providing proper care for patients. These systems store large amounts of medical data in various formats across distributed heterogeneous *information repositories*. Unfortunately, this leads to the creation of isolated information, or "silos". When medical decisions require information from multiple systems, healthcare providers must resort to manual aggregation of knowledge from selected individual information silos. The information is then locally processed to facilitate medical decision making. Limiting factors such as the availability of time or resources may prohibit utilization of health information systems to their full capabilities. Clearly information silos are, on their own, of limited usefulness in a domain like healthcare.

In this work, we evaluate the performance of a recently developed medical decision support framework called the **O**ntological **Me**dical **D**ecision Support System, or **OMeD**. OMeD provides accurate and reliable medical decision support in real-time, and can work with data from distributed heterogeneous information repositories.

Our recent work [8] proposed a framework (see Fig. 1) for medical decision making support based on semantic web techniques, which is realized in OMeD. The framework allows healthcare providers to pose specific questions regarding a patient in the context of a healthcare scenario. For example, an emergency response team member, who is trying to stabilize a patient, may ask whether a certain drug can be safely administered to the patient. OMeD avoids the local aggregation issues discussed above by automatically collecting and *reasoning* about the relevant medical information. The framework proposed in [8] made use of semantic web techniques such as ontology based knowledge represen-

tation and reasoning to facilitate this process. Furthermore, the semantic approach was contrasted, but not compared, with other methodologies for medical decision making. The framework focused on producing a patient specific answer, rather the most probable answer (as with machine learning techniques).

This paper is an extension of [8] and directly compares OMeD with conventional (machine learning) techniques. For this investigation we have defined a simple (yet realistic) model (see Fig. 2) containing various key entities, such as patients, drugs and medical conditions. The model also captures the important relationships between these entities (e.g. drug-to-drug interactions). The model was translated into an N3 [2] representation, converting the entities and associations into their respective ontological concepts and relationships. Simulated data was generated for various scenarios and then a series of *patient specific* queries of the form "Should patient Y be given drug X?" were carried out. We confirmed that, as expected, machine learning models performed poorly in the presence of highly sparse data. In contrast, OMeD was able to accurately answer queries even when presented with limited data.

The remainder of the paper is organized as follows: § 2 reviews the OMeD framework, and the semantic technologies which underpin it. § 3 details our experiments, the generation of simulated patient data, comparative techniques and performance metrics. § 4 presents the results of our experiments, and our final section is devoted to conclusions and a discussion of future work.

2. BACKGROUND

The core components of OMeD are briefly reviewed here. For a detailed description of the framework components please refer to [8]. Fig 1 highlights the following components: (i) An *interface component*, which receives queries and fetches contextual information. (ii) A *knowledge representation component*, which aggregates and translates scenario specific information into a semantic representation suitable for use with OMeD. (iii) A *semantic reasoning component*, which derives the answer to a user query by reasoning over relevant medical knowledge.

The main advantages of using semantic technologies are as follows: (i) Semantic knowledge translation provides the ability to share data across heterogeneous medical information systems. (ii) The inference rules that provide the reasoning capabilities are *data agnostic*. That is, the rules can be defined without recourse to sparse or erroneous medical records, and can be used across many different scenarios. (iii) A by-product of using a semantic reasoner to answer queries is that a formal verifiable proof is constructed by the semantic reasoning component. This proof explains how the system arrived at its answer. (iv) The effectiveness of the semantic approach relies on an enriched semantic model and the available facts. The number of *individuals* (i.e. patients, drugs etc.) generally does not contribute towards increasing the accuracy of a system. Therefore, OMeD's ability to answer a user query can be improved by creating richer ontological models and inference rules. In contrast a machine-learning algorithm would require an increasing amount of data (about the individuals) to achieve a similar level of accuracy.

The current *proof-of-concept* implementation of OMeD was created with the intention of comparing the semantic

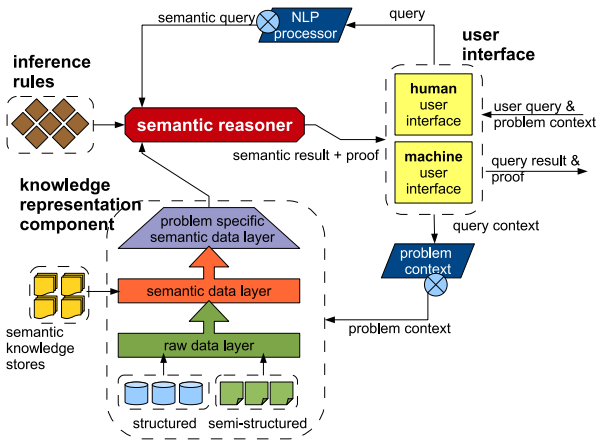

Figure 1: OMeD Architecture.

web technology based approach, against corresponding machine learning approaches. OMeD realization included the semantic knowledge component, the inference rules component and the decision making component of the proposed framework. The user interface component and the data aggregation component were deemed out of scope for this particular study.

3. EXPERIMENT

We constructed an implementation of OMeD based on the framework suggested in [8], and compared its performance against several conventional machine learning techniques by creating random data (simulating health domain). This section describes the composition of the experiments in detail, with results appearing in the subsequent section.

3.1 Setup

We created a minimalistic ontology to represent the core concepts (patient, drug, disease, condition and so on) and their various relationships (see Fig. 2). Using this ontology, we generated a *knowledge-store* consisting of patients, drugs, diseases and conditions with their attributes (§ 3.2). The model and the entire knowledge-store was represented in N3 [2]. Inference rules were generated to answer our **object query i.e. which patients can not be given what drugs**. A sample inference rule is as follows:

Inference Rule: *If a patient has a prescribed drug, and the drug is contra-indictive to the new drug (being given to the patient), then the patient should not be given the new drug*:

```
{?PAT  a :Patient. ?PD   a :Drug. ?DRUG a :Drug.
 ?PAT :hasPrescribedDrug ?PD.
 ?PD  :isContraIndictive ?DRUG.
}=>{?PAT :cannotBeGiven ?DRUG}.
```

We generated all variants of the ***cannotBeGiven*** inference rule, corresponding to all the different entity relationships defined in the semantic model. In order to answer our query, we utilized a semantic reasoner [12]. Based on the inference rules, the reasoner provided the answer to our query for each patient in the knowledge-base covering all drugs, diseases and conditions. Furthermore, for each answer, the

reasoner also generated a ***semantic proof***. A sample proof is as follows:

Semantic Proof: *Patient_0* ***cannotBeGiven*** *Drug17*:
Patient_0 is a patient, who has been prescribed drug Drug0. Drug0 has a contraIndictive relationship with Drug17. Therefore based on the inference rule, Patient_0 can not be given Drug17.

```
{{:Patient_0 a :Patient} e:evidence <ks.n3#_1>.
 {:Drug0 a :Drug} e:evidence <ks.n3#_85>.
 {:Drug17 a :Drug} e:evidence <ks.n3#_124>.
 {:Patient_0 :hasPrescribedDrug :Drug0}
                        e:evidence <ks.n3#_4>.
 {:Drug0 :isContraIndictive :Drug17}
                        e:evidence <ks.n3#_86>}
=> {{:Patient_0 :cannotBeGiven :Drug17}
                        e:evidence <ks.n3#_14>}.
```

In our experiments, we contrast the OMeD with 5 machine learning techniques. The first, and simplest, is the infamous "decision stump". This classifier consists of a single layer decision tree (i.e. a stump), and has been previously demonstrated to perform well across a broad selection of real-world datasets[7]. It can also serve as an indication of the complexity of a machine learning task. The second algorithm is the well known C4.5 decision tree algorithm, which generates comparatively transparent classifiers, and which remains a competative and widely used system despite its age[10]. The third classifier is Naive Bayes, a model which builds a Bayesian classifier under the assumption that the attributes of the data are statistically independent[9]. Like the Decision Stump, Naive Bayes can serve as an indication of the difficulty of a machine learning problem. We also considered two state-of-the-art techniques: Bootstrap Aggregating (Bagging)[3] and AdaBoost[5]. Bagging consists of making a collection of decision trees, each of which is trained on a random sample of the dataset drawn *with replacement* from the full set. The resulting collection's outputs are averaged when engaging in prediction. Boosting, in contrast, involves training a large number of simple classifiers and combining their outputs as an average weighted by a function of their accuracy. AdaBoost dynamically adjusts the importance of learning particular patterns in the data to maximize the total coverage of the resulting collection.

We used the Weka [6] machine learning toolkit's implementations of the five algorithms to ensure that they were correct. Default parameters were used both for simplicity, and because none of the selected algorithms is especially sensitive to parameter selection (unlike, for example, a support vector machine [4]).

3.2 Data

Due to the difficulty inherent in obtaining real-world medical data for use in the evaluation of systems such as this one, we elected to perform our preliminary evaluations with randomly generated simulated patient data. Our data is engineered with the intention of facilitating queries asking whether particular patients should or should not be given particular drugs. We generated a series of hypothetical drugs and maladies, and relationships between them (see Fig. 2). We then generated a dataset of patients, each of which has a randomly selected subset of maladies and is taking a randomly selected subset of drugs. For the semantic system, we represented the data as a N3 knowledge-base [2]. For the machine learning systems, we generated an ARFF[6] formatted data file, containing one line for each patient record.

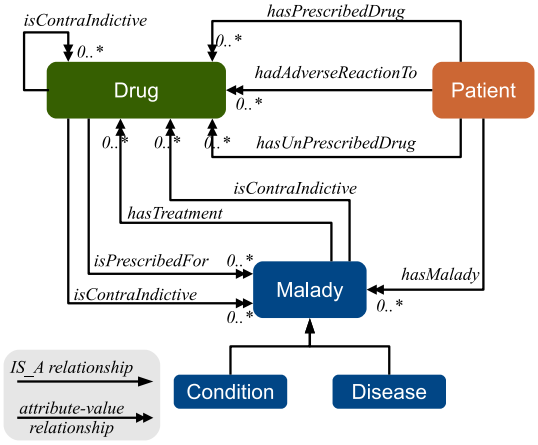

Figure 2: A semantic model depicting the core concepts and their relationships. For example a Patient can have many Maladies and can have many prescribed drugs. A drug may have contraindications to other drugs or maladies.

A patient record contains information about any changes in the patient's status occurring as a result of combining a particular pair of drugs, or taking any drugs on their own, and a list of the maladies a patient has. The goal is to correctly respond to a series of ad hoc queries using only the initial (training) data and any information contained in the query.

We conducted two experiments. First, we varied the number of patient records present in a dataset, while holding the number of different drugs and conditions constant. Second, we held the number of patient records per dataset constant while varying the number of drugs. The data generation process is described in algorithm 1.

3.3 Evaluation Criteria

OMeD is expected to give consistently correct answers (based on the assumption that it is given only accurate data), but it will be useful to determine the extent to which more conventional approaches to question answering might fail to produce such an answer. The assumption behind our experiments is that, because a single patient will interact with only a small fraction of all available drugs and conditions, a very large amount of data will be required to facilitate conventional machine learning techniques. We used a somewhat liberal parameterization, generating data which we expect would be easier for machine learning methods to process than real-world patient data. For example, we assume that a typical patient takes 25% of all drugs in the system, when in fact this rate would probably be much lower. This is done in part to reduce the total amount of data which needs to be generated, and in part to provide comparatively dense data to the machine learning algorithms. The parameters which were fixed across experiments were: $M = 20$,$p_D = 0.05$,$p_T = 0.5$, $p_{pd} = 0.25$, and $p_{pc} = 0.225$ (See Algorithm 1).

For experiment one, we generated 4 data sets using the parameters described above, with 20 drugs and 10, 100, 1,000, and 10,000 patient records respectively. In each set, the first half of the data was used to train the machine learning al-

Algorithm 1 Generates a set of random data with P patient records, D drugs and M maladies. The probability of two drugs interacting is p_D, the probability of a drug treating a particular condition is p_T, and the probabilities of a patient taking a particular drug or having a particular condition are p_{pd} and p_{pc} respectively.

Let *drugs* be a list of hypothetical drugs, of cardinality D.
Let *maladies* be a set of hypothetical maladies, of cardinality M
{Generate the drug <-> drug interactions}
for all *drugs* d1 **do**
  for all *drugs* d2 **do**
    With probability p_D, d1 has a negative interaction with d2, with the severity of the reaction distributed as $\Gamma(2,3)-c$, where c is a random draw from $\Gamma(2,1)$. Otherwise, they have interaction effect $\Gamma(2,2)$.
  end for
end for
{Generate the drug -> malady treatment effects}
for all *drugs* d1 **do**
  for all *maladies* m **do**
    With probability p_T, d1 has a strong treatment effect on m, distributed as $\Gamma(2,3)$. Otherwise, a placebo effect is present, with efficacy distributed as $\Gamma(2,2)$.
  end for
end for
{Output the relationships, and generate and output the patient records.}
Output all the generated relationships in n3 format.
for all *drugs* d **do**
  for p = 1 to P **do**
    Let pat be a new patient record, created with: Generate_Patient(*drugs*, *maladies*,p_{pd}, p_{pc})
    Output pat.
  end for
end for

Algorithm 2 Subroutine: Generate_Patient(*drugs*, *maladies*,p_{pd}, p_{pc})

Let P be a new patient. {Determine which drugs P is taking.}
for all *drugs* d **do**
  With probability p_{pd}, P is taking d.
end for{Determine which conditions P has.}
for all *maladies* m **do**
  With probability p_{pc}, P has m.
end for{Determine whether P experienced any adverse reactions, and which drugs were effective for P.}
for all $(d1, d2) \in drugs \times drugs$ **do**
  if P is taking both d1 and d2 **then**
    Associate with P a random draw from the interaction distribution for (d1,d2).
  end if
end for{Determine whether and of P's conditions were effectively treated by the drugs they were taking.}
for all $(d, m) \in drugs \times maladies$ **do**
  if P has m and is taking d **then**
    Associate with P a random draw from the treatment distributions for (d,m).
  end if
end for
return P.

gorithms, while the second half were used as queries to both OMeD and the models produced using machine learning. We issued a query for each test record and each drug asking the following question: "Is it safe to administer this drug to this patient?", and tracked the correctness of all tested systems. We expected to observe OMeD correctly answering all queries, and the performance of the machine learning algorithms improving as a function of the number of training exemplars. This is because the machine learning algorithms are essentially attempting to fit a number of parameters to the set of patient records provided for training. When the number of records is small relative to the number of parameters, the system is expected to *under-fit* the data, and may perform poorly when generalizing to previously unseen data.

For experiment two, we generated 5 data sets using the parameters described above, with 1,000 patient records and 20, 30, 40, 50 and 60 drugs respectively. As in experiment one, the first half of each set was used for training data, with the remainder being reserved for test queries. We posed the same set of queries as in experiment one, and expected to observe that OMeD's accuracy was invariant to changes in the number of drugs present in the system, and that the performance of the various machine learning algorithms was not (due to under-fitting as the number of parameters in the data increases).

4. RESULTS

The results of our experiments strongly support the use of OMeD's semantic based approach in the medical domain. OMeD produced the correct response to 100% of queries on all datasets.

We analyzed the performance of the various machine learning algorithms in terms of F1-Measure [13], a commonly used information retrieval metric. Our analysis consisted of two steps. First, we determined whether the distribution of F1-Measure for each drug in the data set was consistent with a normal distribution, using a Shapiro-Wilk test[11]. Second, if the data were found inconsistent with a normal distribution, we used non-parametric hypothesis tests (Wilcoxon Rank-Sum and Kruskal-Wallace tests[11]) to determine the effects of algorithm choice and dataset on F1-Measure. Otherwise, we used parametric tests (Student T-test and ANOVA [11]) to determine the impact of those factors. We adjusted p-values for multiple hypothesis testing using Holm's method[11]. The results are presented using violin plots [1].

For experiment 1, we evaluated our results in terms of F1-Measure tracked across queries relating to each of the 20 drugs in each dataset. The distribution of F-Measure was found to be inconsistent with data drawn from a normal distribution. No significant differences in performance were found between the different machine learning algorithms under this measure, but a Kruskal-Wallice test found that the number of patient records used for training had a significant impact on the the performance of machine learning algorithms.This indicates that the machine learning methods we tested require large amounts of data to correctly identify the relationships in simulated medical data. In contrast, OMeD's performance is not dependent on the number of patient records. Further, the exact choice of machine learning algorithm does not matter, indicating that they are all approximately equally effected. Figure 3 visualizes this result for the best performing algorithm (C4.5).

For experiment 2, we continued to use F1-Measure. The data were once again found to be non-normally distributed.

[1]Violin plots show the full distribution information by depicting a rotated density plot and boxplot simultaneously. These violin plots were created using R[11][1].

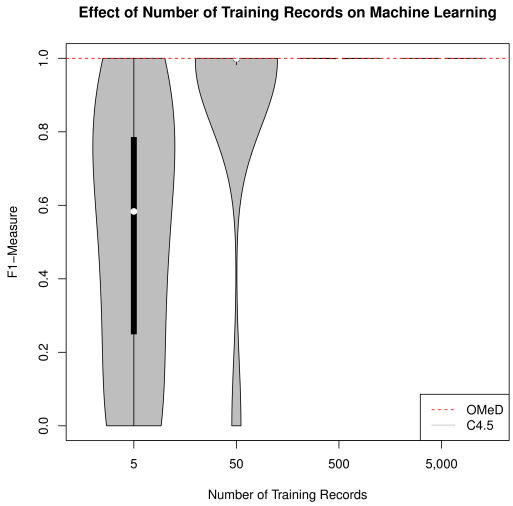

Figure 3: Violin plots depicting the distribution of F1-Measure produced by the C4.5 algorithm across queries pertaining to each drug in the data set, on data sets with increasing numbers of patient records. OMeD's F1-Measure is shown with the dashed line.

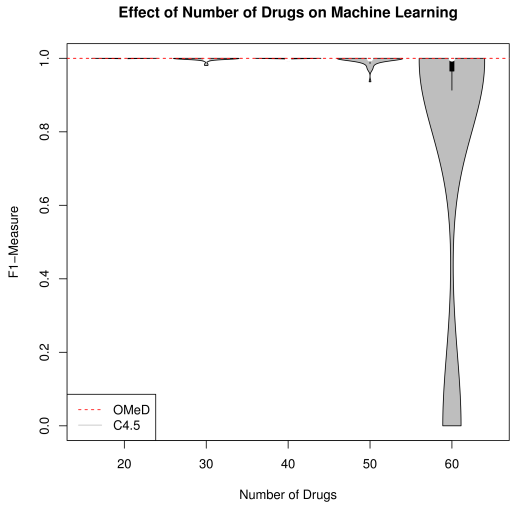

Figure 4: Violin plots depicting the distribution of F1-Measure produced by the C4.5 algorithm across queries pertaining to each drug in the data set, on data sets with increasing numbers of drugs. OMeD's F1-Measure is shown with the dashed line.

A pairwise Wilcoxon Rank-Sum test found no significant difference in the performance of the various machine learning methods, although the mean values for Decision Stump and Naive Bayes were markedly lower than for the more modern methods. A Kruskal-Wallace test found that increasing the number of drugs in the system produced a statistically significant decrease in F1-Measure ($p<0.001$) for the machine learning methods. This indicates that the amount of data machine learning methods require to learn the simulated relationships is coupled to the number of relationships. Again, OMeD is not subject to this constraint. Figure 4 offers a visualization of this result, again in terms of the C4.5 algorithm.

5. CONCLUSIONS

We have constructed and evaluated a prototype of the OMeD system described in [8]. The system performed well, producing high quality performance independent of the availability or sparseness of patient data. Our data sets were generated under assumptions which are favorable to conventional question answering techniques based on machine learning. For example, we assumed that patients take 25% of all drugs, when in fact most patients will take far fewer, yielding an even sparser dataset. In spite of this, machine learning techniques required a comparatively large number of patient records (500) to produce useful models for a small number of drugs (20), and require even more data as the number of drugs in the system increase. In future work will involve testing OMeD with real medical data, and on the the possibility of a combined probabilistic/semantic model.

6. REFERENCES

[1] D. Adler. *vioplot: Violin plot*, 2005. R package version 0.2.

[2] T. Berners-Lee. Notation 3 (n3): An readable language for data on the web. `http://www.w3.org/DesignIssues/Notation3`.

[3] L. Breiman. Bagging predictors. *Machine Learning*, 24(2):123–140, 1996.

[4] C. Cortes and V. Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, 1995.

[5] Y. Freund and R. E.Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, (55), 1997.

[6] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. The weka data mining software: An update. *SIGKDD Explorations*, 11(1), 2009.

[7] R. C. Holte. Very simple classification rules perform well on most commonly used datasets. *Machine Learning*, 11:63–91, 1993.

[8] A. Khan, J. Doucette, C. Jin, L. Fu, and R. Cohen. An ontological approach to data mining for emergency medicine. In *2011 Northeast Decision Sciences Institute Conference Proceedings 40th Annual Meeting*, pages 578–594, Montreal, Quebec, Canada, April 2011.

[9] M. E. Maron. Automatic indexing: An experimental inquiry. *Journal of the ACM*, 8(3):404–417, 1961.

[10] J. Quinlan. *C4.5: programs for machine learning*. Morgan Kaufmann Publishers, 1993.

[11] R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2010. ISBN 3-900051-07-0.

[12] J. D. Roo. Euler proof mechanism. `http://eulersharp.sourceforge.net/`.

[13] G. V. C. Stefan Buettcher, Charles L. A. Clarke. *Information Retrieval: Implementing and Evaluating Search Engines*. The MIT Press, 2010.