# Learning semantic rules for intelligent transport scheduling in hospitals

Pieter Bonte, Femke Ongenae, and Filip De Turck

IBCN research group, INTEC department, Ghent University - iMinds
Pieter.Bonte@intec.ugent.be

**Abstract.** The financial pressure on the health care system forces many hospitals to balance budget while struggling to maintain quality. The increase of ICT infrastructure in hospitals allows to optimize various workflows, which offer opportunities for cost reduction.
This work-in-progress paper details how the patient and equipment transports can be optimized by learning semantic rules to avoid future delays in transport time. Since these delays can have multiple causes, semantic clustering is used to divide the data into manageable training sets.

## 1 Introduction

Due to the continuing financial pressure on the health care system in Flanders, many hospitals are struggling to balance budget while maintaining quality. The increasing amount of ICT infrastructure in hospitals enables cost reduction, by optimizing various workflows. In the AORTA project[1], cost is being reduced by optimizing transport logistics of patients and equipment through the use of smart devices, self-learning models and dynamic scheduling to enable flexible task assignments. The introduction of smart wearables and devices allows the tracking of transports and convenient notification of personnel.

This paper presents how semantic rules, in the form of OWL-axioms, can be learned from historical data, to avoid future delays in transport time. These learned axioms are used to provide accurate data to a dynamic transport scheduler, allowing an optimized scheduling accuracy. For example, the system could learn that certain transports during the visiting hour on Friday are often late and more time should be reserved for those transport during that period. Since transport delays can have multiple causes, semantic clustering is performed to divide the data in more manageable training sets. The increasing amount of integrated ICT infrastructure in hospitals allows all facets of these transport to be captured for thorough analysis. To learn accurate rules, a complete overview of the various activities in the hospital is mandatory. Since this data is resulting from various heterogeneous sources, ontologies are utilized that have proven their strengths in data integration [1]. The incorporation of the domain knowledge modeled in the ontology, allows to learn more accurate rules. Furthermore, learning semantic rules allows to understand and validate the learned results.

[1] http://www.iminds.be/en/projects/2015/03/10/aorta

## 2 Related Work

### 2.1 Learning Rules

Learning rules from semantic data can be accomplished through various methods. The most prevalent are association rule mining [8] and Inductive Logic Programming (ILP) [4]. ILP is able to learn rules as OWL-axioms and fully exploits the semantics describing the data. Incorporating this domain knowledge makes this method more accurate. Statistical relational learning is an extension of ILP that incorporates probabilistic data and can handle observations that may be missing, partially observed, or noisy [2]. However, since our data is not noisy or possible missing, ILP was used in this research.

DL-Learner [5] is an ILP framework for supervised learning in description logics and OWL. Its Class Expression Learning for Ontology Engineering (CELOE) algorithm [6] is a promising learning algorithm. It is a class expression learning algorithm for supervised machine learning that follows a generate and test methodology. This means that class expressions are generated and tested against the background knowledge to evaluate their relevance. Furthermore, no explicit features need to be defined, since the algorithms uses the structure of the ontology to select its features.

### 2.2 Semantic Similarity

Clustering algorithms use a distance measure to have a notion of how similar two data points are. Traditional distance measures, such as the Euclidean measure, are not applicable to semantic data. Therefore, a semantic similarity measure is used to calculate the semantic distance ($1 - semantic\_similarity$).

Semantic similarity measures defines a degree of closeness or separation of target objects [7]. Various semantic similarity measures exist, e.g. the Link Data Semantic Distance [10] uses the graph information in the RDF resources, however it cannot deal with literal values in the RDF data set.

The closest to our approach is the The Maedche and Zacharias (MZ) [9] similarity measure because it fully exploits the ontology structure. The MZ similarity differentiates three dimensions when comparing two semantic entities (i) the taxonomy similarity, (ii) the relation similarity and (iii) the attribute similarity. However, MZ does not take into account that some relations between instances hold more information than others.

## 3 Data set and Ontology

Real data sets were received from two hospitals describing all transports and related information over a timespan of several months. A tailored ontology has been created to model all transport information. It describes the transports, the hospital layout, the patients, the personnel and their relations.

Based on the characteristics of the received data, a data set was generated to conduct our experiments on. For example, about 25% of the scheduled transports do not arrive on time. The relevant use cases, such as described in Section 5, were provided by the hospitals as well. An elaborate description and example of the ontology and the generated data set can be found on http://users.intec.ugent.be/pieter.bonte/aorta/ontology/.

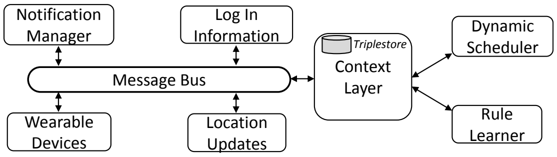

**Fig. 1.** The architecture of the designed AORTA platform

## 4 Architecture

Figure 1 visualizes the conceptual architecture of the dynamic transport planning and execution platform that is currently being built within the AORTA project. Various components can be discerned:

- **Message Bus:** enables the communication between the various components.
- **Notification Manager:** allows the planning of new transports to be communicated and allocated to the staff.
- **Wearable devices:** allows to interact with multiple wearable devices. This enables to communicate personalized tasks and transport to the personnel.
- **Location Updates:** captures the location updates of the executed transports and positioning of personnel.
- **Log in Information:** captures where personnel is logged in and on which wearable device they are reachable.
- **Context Layer:** captures all context changes from the Message Bus and constructs a view on what is happening in the hospital. All knowledge is stored in a Triplestore, using an ontology for easy integration of the data arriving from various sources and incorporation of background knowledge.
- **Dynamic Scheduler:** incorporates sophisticated scheduling algorithms in order to plan the transports in a dynamic fashion. The algorithms receive their input data from the Context Layer, e.g., current location of the staff or average walking speed in particular hallways.
- **Rule Learner:** analyzes the data in the Context Layer and learns why certain transports were late. These learned rules are added to the context layer, enabling this knowledge to be taken into account when planning new transports. This allows the Context Layer to get a better grasp on what is going on in the hospital and provides the Dynamic Scheduler with more accurate input data.

The remainder of this paper focuses on the Rule Learner component.

## 5 Learning relevant rules

The goal of the rule learner, is to learn why certain transports were delayed and use this information to optimize future transport scheduling. Examples of why a transport might be late and their possible learned rules include:

- Popular visiting hours: transports taking place during the visitor hours on Friday or during the weekends have a considerable chance of delay.
  $LateTransport_1 \equiv executedAtDay\ some\ (\{Friday\})$

- Busy passages: transports that use busy passages, such as a very busy elevator, are often late and thus should be avoided on busy moments or more time should be planned when taking this route.
  $LateTransport_2 \equiv hasRoute\ some\ (hasSegment\ some\ (\{BusyElevator\}))$
- Personalized transport times: not all staff members are able to do the transports in the same amount of time.
  $LateTransport_3 \equiv executedBy\ some\ (\{SlowNurse\})$

The CELOE algorithm from DL-Learner was utilized to discover hidden axioms, such as those mentioned above, from the knowledge base. Once the rules are found with a reasonable accuracy, they are added to the *Context Layer*, so they can be taken into consideration when scheduling future transports.

### 5.1 Obstacles

Two obstacles occurred when learning expressions with the CELOE algorithm:

As one can assume, there are various possible explanations why a set of transports were late. The CELOE algorithm should thus result in a Class expression containing multiple unifications to cover all possibilities. However, CELOE gives by design priority to shorter expressions. It thus has difficulties to capture all the causes, even after fine-tuning its parameters. To resolve this matter, we performed semantic clustering on the set of late transports. Each resulting cluster then captures one of the possible causes of the delay. If a detected cluster is sufficient large and represent a subset of data, it is fed to CELOE separately to find a suitable explanation. These explanation are merged afterward. The clustering of semantic data is further discussed in Section 5.2.

Furthermore, learning causes such as in the first example (visitor hours on Friday are busy) require a notion of time. However, days of the week or interesting periods, e.g., vacations, are difficult to derive from the dateTimeStamp data type by the CELOE algorithm. Therefore, we extended our semantic model with various concepts and individuals to incorporate more knowledge about the time points at which the transport was planned and eventually executed. For example, the day of the week, the period of the day (morning, noon, etc.) and the shifts of the staff are modeled.

### 5.2 Semantic Clustering

**Calculating the semantic distance:** Compared to the MZ similarity measure, we propose a semantic similarity measure that takes into account that some relations between certain individuals might hold more information than others. Comparable to Term Frequency-Inverse Document Frequency (TF-IDF), which reflects how important a word is to a document in a collection, some relations from one individual to another, are more important to identify the similarity between two instances in a semantic knowledge base. Since clustering takes place on the subset of transport data that were late, relations to more frequently occurring individuals might hold more information. For example, referring back to the first example, executedOnDay relationships to Friday occur more frequently in the data set of late transports than other days. As such, the executedOnDay relationships to Friday should get more weight. Formally, this means that when two individuals have the same relation to a third individual (of type $C$) which is more frequently referred to than other individuals of type $C$, extra weight is added to the similarity of this relation. The similarity can be measured as:

$$PBSim(i,j) = \sum \frac{taxSim(i,j) + relSim(i,j) + attSim(i,j)}{3}$$

The calculation of taxonomy (taxSim) and attribute similarity (attSim) is similar to those proposed by the MZ measure. The relational similarity is further elaborated upon. To explain this similarity some additional functions and terminology are introduced.
$D$ = document describing all Classes (C), Relations (Rel) and Individuals (Inds).

$$\begin{aligned}
linkFreq(x) &= |\{(s,p,x)|(s,p,x) \in D, p \in Rel, s,x \in Inds\}| \\
typeFreq(x) &= |\{(s,p,y)|(s,p,y) \in D, p \in Rel, s,x,y \in Inds, C(x) = C(y)\}| \\
distinctTypeFreq(x) &= |\{y|x,y \in Inds, C(x) = C(y)\}| \\
R(x) &= \{p|(x,p,y) \in D, p \in Rel, x,y \in Inds\} \\
ER(x,q) &= \{y|(x,q,y) \in D, q \in Rel, x,y \in Inds\}
\end{aligned}$$

The similarity itself can be defined as $min(1, rSim(i,j))$:

$$rSim(i,j) = \begin{cases} \sum_{r \in R(i), r \in R(j)} \sum_{e \in ER(i,r)} \frac{linkFreq(e) * distinctTypeFreq(e)}{typeFreq(e)} & \text{if } e \in ER(j,r) \\ \sum_{r \in R(i), r \in R(j)} \sum_{e \in ER(i,r)} PBSim(e, ER(j,r) & \text{otherwise} \end{cases}$$

It gives more weight to relations to individuals that occur more often than others.

**Clustering semantic data:** To cluster the data in more managable subset, the K-Means clustering algorithm is utilized. It calculates centroids to compute to which cluster the various data points belong to. In the original algorithm, the centroid is an averaged vector of all the data points in that cluster. Since there are no vectorized data points in our cluster, this is not possible. Therefore, a mean individual is calculated for each cluster and used as centroid, similar to the mean schema proposed by Fleischhacker et al. [3]. The type of the centroid is based on the most occurring type of individuals in the cluster. When no consensus can be made, the class hierarchy is taken into account. The most occurring attributes are selected, while the specific value of the literals can be averaged over the occurring values. When selecting the relations, the relations occurring with more than half of the individuals in the cluster are added.

## 6 Results

A data set was generated reflecting the characteristics of a real hospital setting. Various reasons why the transport is late are integrated in the data over various clusters. Since DL-Learner can only discover one of the reason with an accuracy of 80.46%, and a F-measure of 41.77%, clustering was performed. The K-Means algorithm is utilized to compare the MZ similarity measure and our own PB similarity. Table 1 shows the clustering precision and recall and the accuracy (DL-Acc) and F-measure (DL-F) retrieved from DL-learner on the clusters. The precision is calculated as $\frac{TP}{TP+FP}$ and the recall as $\frac{TP}{TP+FN}$. With the true positives (TP), false positives (FP) and false negatives (FN) defines as follows:

**Table 1.** The results utilizing K-Means for the two similarity measures.

| | PB Similarity | | | | MZ Similarity | | | |
|---|---|---|---|---|---|---|---|---|
| | Precision | Recall | DL-Acc | DL-F | Precision | Recall | DL-Acc | DL-F |
| K-Means | 0.95 | 0.94 | 99.94% | 99.96% | 0.43 | 0.45 | 36.7% | 36.7% |

$TP$ = number of transports correctly assigned to $cluster_i$.
$FP$ = number of transports incorrectly assigned to $cluster_i$.
$FN$ = number of transports incorrectly assigned to a $cluster_j$, instead of $cluster_i$.

Note that the results are averages over the various clusters. The combination of K-Means and our own similarity measures allows DL-Learner to learn the expected rules with acceptable accuracy, which was not the case before clustering the data. Using the MZ similarity measure, clustering often fails to identify the correct clusters. This results in an unmanageable subset of data for DL-Learner to perform its rule learning on.

## 7 Conclusion & Future Work

Scheduling transports can be optimized by learning from past delays. The possible high number of various reasons for this delay can be handled by performing semantic clustering on the data set, producing more manageable data sets for learning algorithms such as DL-Learner's CELOE.

In future work, we will investigate how the learned rules can be incorporated to influence the assignment. Furthermore, a statistical relational learning solution will be researched, to be able to handle possible missing and noisy data.

**Acknowledgment:** This research was partly funded by the AORTA project, co-funded by the IWT, iMinds, Xperthis, Televic Healthcare, AZMM and ZNA.

## References

1. Bergamaschi, et al.: Semantic integration of heterogeneous information sources. Data & Knowledge Engineering 36, 215–249 (2001)
2. De Raedt, L., Kersting, K.: Statistical relational learning. In: Encyclopedia of Machine Learning, pp. 916–924. Springer (2011)
3. Fleischhacker, et al.: Mining rdf data for property axioms. In: On the Move to Meaningful Internet Systems: OTM 2012, pp. 718–735. Springer (2012)
4. Konstantopoulos, S., et al.: Formulating description logic learning as an inductive logic programming task. In: FUZZ-IEEE. pp. 1–7 (2010)
5. Lehmann, J.: Dl-learner: Learning concepts in description logics. J. Mach. Learn. Res. 10, 2639–2642 (Dec 2009)
6. Lehmann, J., et al.: Class expression learning for ontology engineering. Web Semantics: Science, Services and Agents on the World Wide Web 9(1), 71–81 (2011)
7. Lin, C., et al.: Sherlock: A Semi-automatic Framework for Quiz Generation Using a Hybrid Semantic Similarity Measure. Cognitive Computation 7(6), 667–679 (2015)
8. Nebot, V., Berlanga, R.: Finding association rules in semantic web data. Knowledge-Based Systems 25(1), 51–62 (2012)
9. Ng, R.T., et al.: Clustering ontology-based metadata in the semantic web. Principles of Data Mining and Knowledge Discovery 1704(February), 262 – 270 (1999)
10. Passant, A.: Measuring semantic distance on linking data and using it for resources recommendations. In: AAAI spring symposium: linked data meets artificial intelligence. vol. 77, p. 123 (2010)