

A Conceptual Graph Description of Medical Data for Brain Tumour Classification

Madalina Croitoru, Bo Hu, Srinandan Dashmapatra, Paul Lewis,
David Dupplaw, and Liang Xiao

University of Southampton

Abstract. HealthAgents proposes an agent-based distributed decision support system for brain tumour diagnosis and prognosis which employs Magnetic Resonance Imaging and Magnetic Resonance Spectroscopy techniques and genomic profiles. From a knowledge representation view point the distributed nature and the heterogeneity of the data to be integrated pose a number of challenging problems. This paper shows how Conceptual Graphs can be employed to describe the data sources in the HealthAgents system. Such knowledge representation based description of data allows for reasoning power when querying and for data modularisation capabilities.

1 Introduction

In this paper we propose a Conceptual Graph [6] based description of the knowledge involved to build a distributed decision system for brain tumour classification (HealthAgents). We present our work formally and demonstrate how a model based semantics description of such highly heterogeneous knowledge, as well as a Conceptual Graph integration of such descriptions can benefit the system by providing modularization and querying power. Our results are theoretical and lay rigorous foundations for future implementation.

HealthAgents [1] is an agent-based, distributed decision-support system (DSS) that employs clinical information, Magnetic Resonance Imaging (MRI) data, Magnetic Resonance Spectroscopy (MRS) data and genomic DNA profile information. The aim of this project is to help improve brain tumour classification by providing alternative, non invasive techniques. A predecessor project, Interpret [7], has shown that MRI and single voxel MRS data can aid in improving brain tumour classification. HealthAgents builds on top of these results and further employs multi voxel MRS data, as well as genomic DNA micro-array information for better classification results. Moreover, HealthAgents is decentralizing the Interpret DSS by building a distributed decision support system (D-DSS). This way, the number of cases to be studied is greatly increased, improving classifier accuracy. Certain differences in patient data, determined by geographic factors, is also easier to identify.

At the moment the data in the HealthAgents system is stored in relational databases at the various participating European clinical centers. A uniform vocabulary needed for interoperability reasons is provided by the means of HADOM

- an ontology containing MRI, MRS and micro-array *domain information* as well as a taxonomy of brain tumours compliant to the WHO(World Health Organisation)¹ classification.

We propose describing the knowledge contained in the sources by the means of Conceptual Graphs. This allows us to build upon the existing ontology while not overcomplicating the ontology with rules to describe data extraction techniques that employ different parameters which greatly influence the outcome data. An immediate advantage of our Conceptual Graphs choice is their graph based reasoning mechanisms which allow versatile querying algorithms [4].

In Section 2 we present the challenges the HealthAgents system poses in terms of knowledge representation and reasoning. This motivates our Conceptual Graph based approach to data description informally introduced in Section 3 and formally presented in Section 4. Section 5 concludes the paper and lays down future work directions.

2 Motivation

From a knowledge representation view point the distributed nature of the HealthAgents system poses a number of problems due to the heterogeneity of the data to be integrated. Once the data acquisition protocols have been agreed upon and the data formats reconciliated, the data has to be managed and queried in an “intelligent” manner. The need – triggered by interoperability issues – for a common vocabulary was already addressed by the HADOM (HealthAgents Domain) ontology which conceptualises the parameters of the employed techniques (MRI, MRS, DNA microarrays etc.), the clinical information needed (age, sex, location etc.) and the known brain tumour classes.²

However this is not expressive enough for versatile querying and data integration purposes. This paper shows how Conceptual Graphs (a graph-based, logical knowledge representation formalism) can be employed to describe the data sources in the HealthAgents system. Since Conceptual Graphs are logically equivalent to the existential, positive fragment of First Order Logic, this knowledge based description allows for reasoning power when querying and for data modularisation capabilities which will lead to complete query answering across incomplete data sources.

We claim that a Conceptual Graph (CG) based description of the data within the HealthAgents system adds expressiveness for knowledge representation and versatility for querying. Our choice of knowledge representation (KR) formalism is motivated by the fact that Conceptual Graphs are:

- Expressive enough to be able to represent the data extraction protocols and the rules associated with them.
- Easy to plug in on top of existing ontologies due to the distinction between ontological knowledge (the support) and factual knowledge (bipartite graph).

¹ Available from Harvard Medical School at:
<http://neurosurgery.mgh.harvard.edu/newwhobt.htm>

² According to the WHO classification.

The work we present here is highly technical, addressing the specific Conceptual Graphs problems that occur when describing such data sources. Our work is evaluated theoretically by the soundness and completeness of the proposed definitions.

3 Example

As mentioned in Section 2, the *(i)* heterogeneity of the data to be represented and the *(ii)* distributed nature of the project make knowledge representation a challenging aspect of HealthAgents.

A first step towards addressing the heterogeneity problem was creating an ontology of the main concepts used in the system. In this way a common ontological background was established. This ontology contains a poset (partially ordered set) of the known brain tumour classes according to the WHO classification, a poset of the techniques used for data acquisition characterized by their parameters and the clinical information needed for the patients (age, sex, location, medication, etc.). The HealthAgents data itself is stored anonymously and securely in a distributed network of datamarts in relational databases.

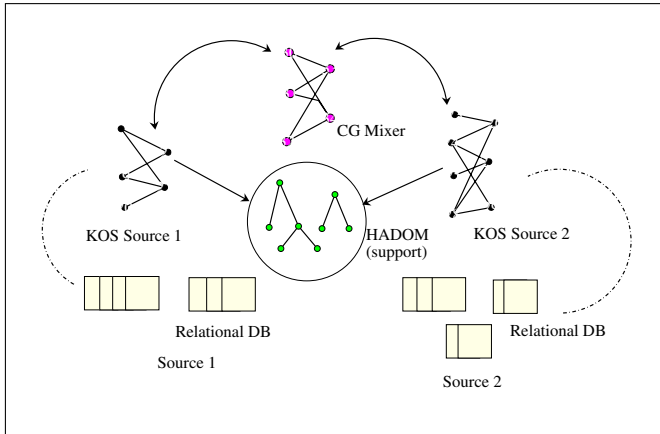


Fig. 1. Conceptual Graph Description of Knowledge

The nature of the acquisition protocols make the output data (the spectra) highly dependent on the parameters employed (for example, multi-voxel MRS techniques require the scanner manufacturer to be known in order for the data to be interpreted in a correct way). Since the data from different clinical centers has to be integrated, a common vocabulary is not enough to represent such knowledge. It is also essential to be able to provide reasoning power between the sources. We propose a “KR annotation” for the relational databases stored at each individual clinical center: Conceptual Graph based descriptions of the data

in the sources. These Conceptual Graphs based on the common support of the extension of the HADOM ontology are called a Knowledge Oriented Specification of the source.³ An example of the Conceptual Graph based approach to data description is shown in Figure 1, while Figure 4 presents a simplified example of two KOSs for MRS and MRI data sources.

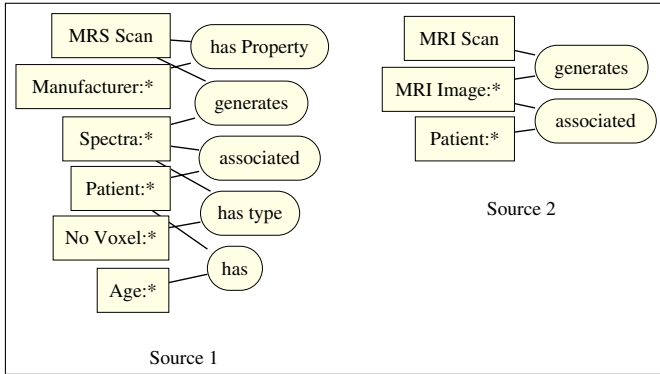


Fig. 2. Knowledge Oriented Specification

In order to query the knowledge Oriented Specification we use query Conceptual Graphs (qCG) [5]. Moreover, once the sources are described with Conceptual Graphs they can be integrated in a CG Mixer. In this “global view” of the system the domain expert specifies exactly what queries can be posed in terms of this integrated schema. Once the query is posed, the relations from the CG Mixer are rewrote to direct the query to the appropriate data sources. Querying a CG Mixer is intuitively depicted in Figure 3.

4 Formalism

In this section we provide the formalization for our approach. The motivation behind such a thorough, step by step rigorous foundation is that, in this way, we benefit from a in depth understanding of the model. This understanding facilitates future implementation.

A couple of definitions are needed to “prepare” linking the proposed Conceptual Graph description to the data sources. We introduce a support model to assign appropriate values from a domain (universe) to each concept type, relation type and marker. An assignment allows to link the concepts of a CG to the domain (universe) of the model defined over its support.

Given a data source, we need to be able to link the information (set of tuples) contained therein with the associated Conceptual Graph and its model. To do this we introduce the notion of a repository.

³ More specifically, we enrich HADOM with a poset of relation types needed for protocol description, patient diagnosis etc.

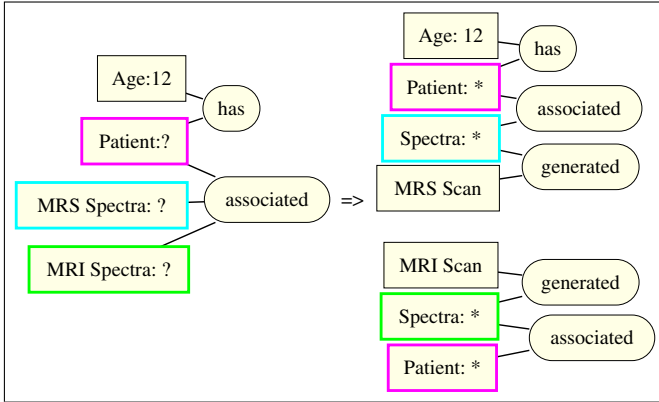


Fig. 3. Query rewriting

A query Conceptual Graph allows one to represent a query SQ over the sources in a Conceptual Graph like notation. However, if SQ has a concept vertex labelled with an individual marker then this vertex can be projected only in a concept vertex of the conceptual graph labelled with the same individual marker, by the definition of a projection. Therefore we introduce a specific querying mechanism, considering legal queries. Lastly, we define an answer to a qCG as the set of all data retrieved from the repository that validate the qCG.

A knowledge oriented specification of an information source is composed by (i) a Conceptual Graph that visually describes what we expect to know from that source, (ii) an interpretation for the support on which the graph is built, (iii) a repository for the graph (that contains all the data tuples), and (iv) a wrapper that ensures the communication between the user queries and the repository.

A CG Mixer depicts the integrated view, by the means of a Conceptual Graph, and provides the rules to allow for the translation of user queries to the appropriate data sources. The rules are defined by the relation vertices from the integrated view. For each relation in the integrated view, the proper translation is provided. This translation has to preserve the order of nodes in the initial relation, for correctness.

4.1 Data Sources

A Knowledge Oriented Specification (KOS) for a source is, informally, a Conceptual Graph that syntactically describes the data along with the data source itself. The specification does not try to exhaustively describe the sources, but provides a description of the data we have access to. More formally, if we issue a query over this specification, we should have access to the answer by the means of a wrapper. The main purpose of a KOS is to simply inform us how materialized views over the data sources can be obtained.

Usually, CGs are given semantics by translating them to existential first order logic formulae (see, for example, [3]). We propose a semantics based on model

theory, adapted for our integration purposes. In order to do this we define what the interpretation of a support is, and how to assign that interpretation to the simple Conceptual Graph defined on that support. This leads to the notion of a repository of a CG on a model. In the following definitions we are building upon the work of [2].

An **interpretation** (or model) for a support is a structure that assigns appropriate values from a domain (universe) to each concept type, relation type and marker. This assignment respects the way the relation /concept types are defined and also preserves their hierarchical order.

Definition 1. (Interpretation)

An interpretation or model \mathcal{M} for the support $S = (T_C, T_R, \mathcal{I}, *)$ is a pair $\mathcal{M} = (D, F)$ where

- D is a set of objects called the domain or universe of \mathcal{M} ,
- F is a function defined on $T_C \cup T_R \cup \mathcal{I}$ such that $F(\mathcal{I}) \subseteq D$, $F(T_C) \subseteq \mathcal{P}(D)$, $F(T_R^i) \subseteq \mathcal{P}(D^i)$ for each $i \in \{1, \dots, k\}$ (k is the maximum arity of a relation type in T_R) satisfying:
 - $\forall t_c, t'_c \in T_C, t_c \leq t'_c \Rightarrow F(t_c) \subseteq F(t'_c)$,
 - $\forall t_r, t'_r \in T_R^i, t_r \leq t'_r \Rightarrow F(t_r) \subseteq F(t'_r)$.

An **assignment** allows to link the concepts of a CG to the domain (universe) of the model defined over its support.

Definition 2. (Assignment)

Let $\mathcal{M} = (D, F)$ be a model for the support $S = (T_C, T_R, \mathcal{I}, *)$, and $SG = [S, G, \lambda]$ be a CG, with $G = (V_C, V_R, N_G)$.

- An assignment for SG in \mathcal{M} is a function $f : V_C \rightarrow D$ such that
- $\forall c \in V_C$, if $\lambda(c) = (t_c, ref_c)$ then $f(c) \in F(t_c)$, and if $ref_c \in \mathcal{I}$ then $f(c) = F(ref_c)$;
 - $\forall r \in V_R$, if $deg_G(r) = i$ then $(f(N_G^1(r)), \dots, f(N_G^i(r))) \in F(\lambda(r))$.
- The set of all assignments for SG in the model \mathcal{M} is denoted $\mathcal{A}(SG, \mathcal{M})$. If $\mathcal{A}(SG, \mathcal{M}) \neq \emptyset$ then SG holds in \mathcal{M} and is denoted $\mathcal{M} \models SG$.

The soundness of projection now follows as a simple observation. Indeed, let SG and SH be two Conceptual Graphs on the same support S such that $SG \geq SH$ and let \mathcal{M} be a model for S .

Each assignment f for SH in \mathcal{M} can be used to construct an assignment f' for SG in \mathcal{M} , by defining $f'(c) = f(\pi(c))$, where π is some projection from SG to SH .

Hence, we have obtained that *if $SG \geq SH$ and $\mathcal{M} \models SH$ then $\mathcal{M} \models SG$.*

Given a data source, we need to be able to link the information (set of tuples) contained therein with the associated Conceptual Graph and its model. To do this we introduce the notion of a **repository**. A repository is a set of tuples, each of which makes the Conceptual Graph true in a given model.

The repository is intentional (as opposed to extensional); one needs to go through the data source to be able to build it. There is no need to materialize the repository in order to make use of it (in the manner of materialized views

for databases). Intuitively, a repository contains all possible interpretations for the generic (marked with “*”) concepts in the graph.

Definition 3. (Repository)

Let $\mathcal{M} = (D, F)$ be a model for the support $S = (T_C, T_R, \mathcal{I}, *)$, and $SG = [S, G, \lambda]$ be a CG, with $G = (V_C, V_R, N_G)$.

We set $V_C := V_C(*) \cup V_C(\mathcal{I})$, where for each $c \in V_C$ with $\lambda(c) = (t_c, ref_c)$ we have $c \in V_C(*)$ if $ref_c = *$, and $c \in V_C(\mathcal{I})$ if $ref_c \in \mathcal{I}$. We also suppose that $V_C(*) \neq \emptyset$ and that an ordering $V_C(*) = \{c_1, \dots, c_p\}$ is fixed.

The repository for SG in the model \mathcal{M} , is the set $\mathcal{R}(SG, \mathcal{M}) \subseteq D^p$ of all tuples $(d_1, \dots, d_p) \in D^p$ with the property that the mapping $f : V_C \rightarrow D$, defined by

$$f(c_i) := d_i, \text{ for } c_i \in V_C(*), \text{ and } f(c) := F(ref_c), \text{ for } c \in V_C(\mathcal{I}),$$

is an assignment for SG in \mathcal{M} .

4.2 Querying the Data Sources

Once the data sources are defined, we need to be able to query and integrate them with other sources. In this section we define the main querying mechanisms for our model and how the results are retrieved. We also formally introduce the notion of a knowledge oriented specification, and we present our integration methodology.

In order to be able to query the data sources, we introduce a structure called a **query Conceptual Graph** (qCG). This structure is similar to that introduced in [5], but in this paper we define it in a new, graph theory oriented, light.

A query Conceptual Graph allows one to represent a query over the sources in a Conceptual Graph like notation. Basically, to find all the information about a generic concept, we mark it by “?”. The “?” symbol stands for all the instances of a given type in the repository, which make the graph hold. The qCG has an associated Simple Conceptual Graph, whose intuitive purpose is to represent the query graph without any “?”. Later on, when defining an answer to a qCG, this graph is important because it helps validate answers.

Definition 4. (Query Conceptual Graph)

A query Conceptual Graph (abbreviated qCG) is quadruple $\mathbb{Q} = [SQ, arity, X, \lambda'_Q]$, where

- $SQ = [S, Q, \lambda_Q]$ is a CG with $Q = (V_C, V_R, N_Q)$,
 - $arity$ is a positive integer,
 - $X \subseteq V_C(*)$, and
 - $\lambda'_Q : X \rightarrow \{?^1, ?^2, \dots, ?^{arity}\}$ is a surjective labelling (with query marks).
- SQ is the Conceptual Graph associated to qCG \mathbb{Q} , $arity$ is the arity of \mathbb{Q} , and X are the query concept vertices of \mathbb{Q} .

Let SG be a CG and \mathbb{Q} be a qCG both defined on the same support S . We could define the answer to \mathbb{Q} over SG as the union of the repositories of all spanned subgraph of SG on which SQ (the the Conceptual Graph associated to \mathbb{Q}) can be projected. However, if SQ has a concept vertex labelled with an individual marker then this vertex can be projected only in a concept vertex of SG labelled with the same individual marker, by the definition of a projection (if $i_1, i_2 \in \mathcal{I}$

then $i_1 \geq i_2$ if and only if $i_1 = i_2$; however, $* \geq i$ for all $i \in \mathcal{I}$. This works if the source represented by SG is a collection of Conceptual Graphs, which is not feasible in an integration scenario. Therefore we introduce a specific querying mechanism, considering *legal queries* defined as follows.

Definition 5. (Legal Query)

Let SG be a CG and \mathbb{Q} be a qCG both defined on the same support S . Let $SQ = [S, Q, \lambda_Q]$ be the Conceptual Graph associated to \mathbb{Q} with $Q = (V_C, V_R, N_Q)$. We denote by SQ^* the CG obtained from SQ by replacing the individual markers with $*$, for all concept nodes belonging to the set $A = V_C \setminus V_C(*)$.

We say that \mathbb{Q} is a **legal query for SG** if the set, $Occ(\mathbb{Q}, SG)$, of the occurrences of \mathbb{Q} in SG is nonempty, where $Occ(\mathbb{Q}, SG) = \{\pi(SG^*) \mid \pi \in \Pi_{SG^* \rightarrow SG} \text{ and for each } v \in A, \text{ if } \lambda_G(\pi(v)) = (type, i), \text{ then } \lambda_Q(v) = (type', i)\}$.

In words, \mathbb{Q} is a legal query for SG if there is a spanned subgraph of SG ($\pi(SQ^*)$) into which SQ can be projected or into which SQ^* (the CG obtained from SQ by transforming all nodes in generic conceptual nodes) can be projected. In the second case, if the spanned subgraph of SG has individual concept nodes, these must be “compatible” with the corresponding individual concept nodes from SQ .

Therefore if $SH \in Occ(\mathbb{Q}, SG)$ then $SH \geq SG$ and either $SQ \geq SH$ or $SQ \not\geq SH$. In the last case however, we have chances to find an assignment in the repository of SG for which SQ holds.

By the above definition we have oriented the role of the SG to the description of querying capability of the source, rather than a “schematically” description of it. This is the first step to our “knowledge oriented specification” of a source. We further need to define the answering mechanism.

An **answer** to a qCG is the set of all data retrieved from the repository that validate the qCG. Intuitively, by taking all the instances from the repository that make the graph associated to the qCG true, one obtain its answer. This notion is very important because it helps us define a knowledge oriented specification for a given source.

Definition 6. (Answer to a qCG)

Let $SG = [S, G, \lambda]$ be a CG with $G = (V_C^1, V_R^1, N_G)$, $V_C^1(*) = \{c_1, \dots, c_p\}$, $\mathcal{M} = (D, F)$ be a model for the support S , and $\mathcal{R}(SG, \mathcal{M})$ be a repository for SG in the model \mathcal{M} .

Let $\mathbb{Q} = [SQ, arity, X, \lambda'_Q]$ be a legal qCG for the CG SG ($Occ(\mathbb{Q}, SG) \neq \emptyset$).

We define the answer to \mathbb{Q} over $\mathcal{R}(SG, \mathcal{M})$ as being the set $Ans(\mathbb{Q}, \mathcal{R}(SG, \mathcal{M}))$ obtained with the following algorithm:

```

Ans( $\mathbb{Q}, \mathcal{R}(SG, \mathcal{M})$ )  $\leftarrow \emptyset$ 
for each  $\pi(SQ^*) \in Occ(\mathbb{Q}, SG)$  do
  for each  $(d_1^1, \dots, d_p^1) \in \mathcal{R}(SG, \mathcal{M})$  do {
    compatible  $\leftarrow true$ 
    for  $j = \overline{1, p}$  do
      if compatible then
        if  $\exists c \in V_C(SQ^*)$  s.t. [ $\pi(c) = c_j$  and  $\lambda_Q(c) = (type, i)$ ] then

```



```

        if  $F(i) \neq d_j^1$  then compatible  $\leftarrow$  false
    if compatible then {
        for  $i = \overline{1, \text{arity}}$  do {
            find  $c \in \lambda'^{-1}(i)$  and  $j$  such that  $\pi(c) = c_j$ 
             $d_i \leftarrow d_j^1$ 
        }
        add  $(d_1, \dots, d^{\text{arity}})$  to  $\text{Ans}(\mathbb{Q}, \mathcal{R}(SG, \mathcal{M}))$ 
    }
}

```

Theorem 1. Let $SG = [S, G, \lambda]$ be a CG with $G = (V_C^1, V_R^1, N_G)$, $V_C^1(*) = \{c_1, \dots, c_p\}$, $\mathcal{M} = (D, F)$ be a model for the support S , and $\mathcal{R}(SG, \mathcal{M})$ be a repository for SG in the model \mathcal{M} . If $\mathbb{Q} = [SQ, \text{arity}, X, \lambda_Q]$ is a legal qCG for the CG SG , then

(i) if $SQ \geq SG$ then $\text{Ans}(\mathbb{Q}, \mathcal{R}(SG, \mathcal{M})) \neq \emptyset$.

(ii) if $SQ \not\geq SG$ but in the $\mathcal{R}(SG, \mathcal{M})$ there is a tuple which gives rise to an assignment of a CG SG' obtained from SG by replacing some generic concept nodes by individual concept nodes and having the property that $SQ \geq SG'$, then $\text{Ans}(\mathbb{Q}, \mathcal{R}(SG, \mathcal{M})) \neq \emptyset$.

Proof. Part (i) is a trivial corollary of the soundness of projection and part (ii) follows from the algorithm for the construction of the set $\text{Ans}(\mathbb{Q}, \mathcal{R}(SG, \mathcal{M}))$.

All notions introduced above lead now to the formal definition of a **knowledge oriented specification**.

A knowledge oriented specification of an information source is composed by

- (i) a Conceptual Graph that visually describes what we expect to know from that source,
- (ii) an interpretation for the support on which the graph is built,
- (iii) a repository for the graph (that contains all the data tuples), and
- (iv) a wrapper that ensures the communication between the user queries and the repository.

Definition 7. (Knowledge Oriented Specification of an Information Source)

Let IS be an information source. A knowledge oriented specification of IS is a quadruple $KOS(IS) = (SG, \mathcal{M}, \mathcal{R}(SG, \mathcal{M}), W)$, where

- $SG = [S, G, \lambda]$ is a CG on the support S , the source support,

- $\mathcal{M} = (D, F)$ is a model for the support S , the source model,

- $\mathcal{R}(SG, \mathcal{M})$ is a repository for SG in the model \mathcal{M} , and

- W is a wrapper, that is a software tool which, for each legal qCG \mathbb{Q} for SG , returns the answer set $\text{Ans}(\mathbb{Q}, \mathcal{R}(SG, \mathcal{M}))$.

Figure 4 integrates all the above definitions and sketches a Knowledge Oriented Specification.

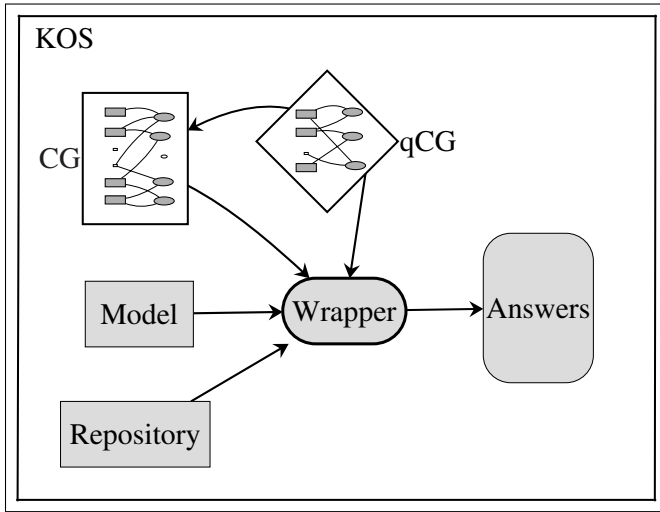


Fig. 4. Knowledge Oriented Specification

4.3 Data Integration

A CG Mixer depicts the integrated view, by the means of a Conceptual Graph, and provides the rules to allow for the translation of user queries to the appropriate data sources. The rules are defined by the relation vertices from the integrated view.

For each relation in the integrated view, the proper translation is provided. This translation has to preserve the order of nodes in the initial relation, hence the extra labelling of concepts (as depicted in greyed out rectangles).

Definition 8. (CG Mixer)

Let IS^1, \dots, IS^n be a set of information sources, and their knowledge oriented specifications $KOS(IS^i) = (SG^i, \mathcal{M}^i, \mathcal{R}^i(SG^i, \mathcal{M}^i), W^i), i = 1, n$.

A CG Mixer over the information sources IS^1, \dots, IS^n is a pair $\mathbb{M}(IS^1, \dots, IS^n) := (SG^0, \mathbb{R})$, where

- $SG^0 = [S^0, G^0, \lambda^0]$ is a CG with $G^0 = (V_C^0, V_R^0, N_{G^0})$, and
- \mathbb{R} is a mapping which specifies for each $r^0 \in V_R^0$ a set $\mathbb{R}(r^0)$ of rules providing descriptions of the relation vertex r^0 in (some of) information sources. Each rule in $\mathbb{R}(r^0)$ is a triple (IS^k, A, w) , where

- IS^k is an information source specified by $KOS(IS^k)$
- $A \subseteq V_R^k$ (the relation vertices set of SG^k)
- $w \in V_C^{\dagger}([A]_{G^k})$ is a sequence of $d_{G^0}(r^0)$ concept vertices of the subgraph $[A]_{G^k}$ spanned in G^k by the relation vertices in A .

A rule $(IS^i, A, w) \in \mathbb{R}(r^0)$ means that the star graph $G^0[r^0]$, is translated in the source IS^i as $[A]_{G^k}$ and if $w = w_1 \dots w_k$ ($k = d_{G^0}(r^0)$), then w_j corresponds to $N_{G^0}^j(r^0)$ ($j = 1, k$).

In other words, a rule interprets each relation vertex in the CG Mixer via a subgraph of the CG describing the appropriate local source. This is done by means of an ordered sequence of concept vertices (the relations' vertex neighbors).

Let $\mathbb{M}(IS^1, \dots, IS^n) = (SG^0, \mathbb{R})$ be a CG Mixer. A *legal query* over $\mathbb{M}(IS^1, \dots, IS^n)$ is any legal qCG for SG^0 . Let $\mathbb{Q} = [SQ, \text{arity}, X, \lambda_{\mathbb{Q}}]$, be a legal qCG for SG^0 , with $SQ = [S, Q, \lambda_Q]$, $Q = (V_C, V_R, N_Q)$, and $X \subseteq V_C(*)$. Consider also $Occ(\mathbb{Q}, SG)$, the set of the occurrences of \mathbb{Q} in SG (see definition 5).

Let $V_R = \{r_1^0, \dots, r_m^0\}$ and $H = [\{r_1^0, \dots, r_m^0\}]_{G^0}$ (the spanned subgraph of G^0 from which is obtained SQ by specialization).

From SQ and (SG^0, \mathbb{R}) a set $\mathbb{R}(SQ)$ of graphs is constructed as follows.

For each $H \in Occ(\mathbb{Q}, SG)$ consider $\{r_1^0, \dots, r_m^0\}$ its set of relation nodes ($H = [\{r_1^0, \dots, r_m^0\}]_{G^0}$).

For each m -uple of rules $((IS^{k_1}, A^1, w^1), \dots, (IS^{k_m}, A^m, w^m)) \in \mathbb{R}(r_1^0) \times \dots \times \mathbb{R}(r_m^0)$ a graph RH is added to $\mathbb{R}(SQ)$.

The graph RH is constructed by considering first the union $RH = F^1 \cup \dots \cup F^m$. Here, the graph F^i , ($i = 1, m$), is obtained from $[A^i]_{G^{k_i}}$ in the following way: if the concept vertex w_j^i , ($j = 1, d_{SQ}(r_i^0)$), has a generic marker in SG^{k_i} and in SQ the j -neighbor of r_i^0 has been replaced by an individual marker, then the generic marker of w_j^i is replaced by this individual marker. Note that in the above union, the subgraphs coming from distinct sources are disjoint.

The final graph RH is obtained by adding, to the above obtained graph, a special set of new relation vertices in order to describe the neighborhood structure of the original graph H . All these vertices have the special label (name) " = " and have exactly two neighbors (with the meaning that the corresponding concept vertices represent the same object).

More precisely, for each $i, j \in \{1, \dots, m\}$ such that $N_H^t(r_i^0) = N_H^s(r_j^0)$ (in H the t -neighbor of r_i^0 is the same concept vertex as the s -neighbor of r_j^0), and $IS^{k_i} \neq IS^{k_j}$ (the two subgraphs in which r_i and r_j are coming from distinct sources), a new equality relation vertex is added to the graph already constructed, with the 1-neighbor the vertex w_i^i of F^i and the 2-neighbor the vertex w_s^j of F^j .

The graphs from the set $\mathbb{R}(SQ)$ can be considered as the set of all possible query rewriting of \mathbb{Q} . Clearly, each of the $|\mathbb{R}(r_1^0)| \times \dots \times |\mathbb{R}(r_m^0)|$ graphs added in $\mathbb{R}(SQ)$ for $H \in Occ(\mathbb{Q}, SG)$, $H = [\{r_1^0, \dots, r_m^0\}]_{G^0}$, is constructed with the above algorithm in polynomial time (with respect to the orders of the subgraphs involved).

By the above construction, each graph $RH \in \mathbb{R}(SQ)$ can be expressed as a disjoint union of source subgraphs, interconnected (as described above) by the equality relation vertices.

Let us denote by RH^j be the (nonempty) subgraph of RH which is also a subgraph of the graph SG^j associated to the source IS^j . If we assign appropriate query marks to the concept vertices corresponding to the vertices of SQ having query marks, we obtain a legal qCG for the source SG^j . This can be obtained with the following algorithm:

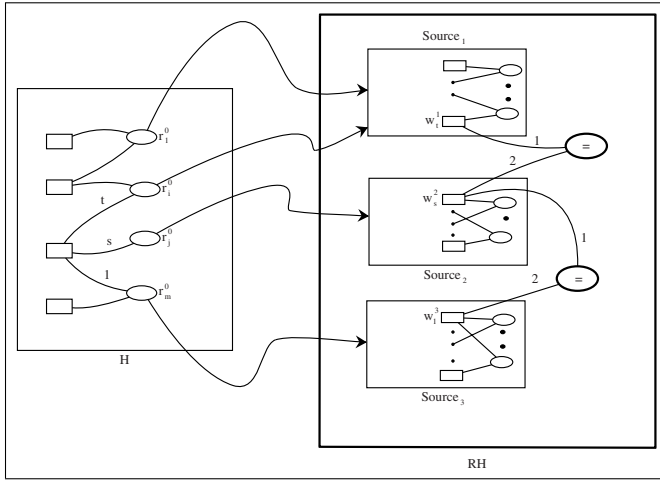


Fig. 5. Query Rewriting

For each concept vertex $c \in V_C(RH^j)$, if $c = w_k^j$, for some k (which means that there is r_i^0 for which a rule (IS^j, A^j, w^j) has been used in the construction of RH), and if $N_{SQ}^k(r_i^0)$ has a query marker, then assign a query marker to w_k^j .

The superscripts of these new query markers can be given such that they form a set $\{1, \dots, arity'\}$ and also respect the meaning in \mathbb{Q} (that is, if two original vertices in SQ have the same query mark, then their surrogates in RH^j have the same new query mark).

Clearly, the above algorithm constructs a legal qCG \mathbb{Q}_{RH}^j for SG^j .

Therefore to each graph $RH \in \mathbb{R}(SQ)$ we have associated a set of legal qCG's for the graphs describing the sources involved in the construction of RH .

If $J \subseteq \{1, \dots, n\}$ is the index set of the sources IS^j involved in the construction of RH , then this set of legal qCG's is $\mathbb{Q}(RH) := \{\mathbb{Q}_{RH}^j | j \in J\}$.

The answer, $Ans(RH)$, to RH over $\mathbb{Q}(RH)$ is constructed with the following algorithm:

- For each $j \in J$ find the set $Ans(\mathbb{Q}_{RH}^j, \mathcal{R}^j(SG^j, \mathcal{M}^j))$ (using the wrapper W^j);
- $Ans(RH) := \emptyset$;
- For each element of the set $\times_{j \in J} Ans(\mathbb{Q}_{RH}^j, \mathcal{R}^j(SG^j, \mathcal{M}^j))$ verify if the values corresponding to the concept vertices which are the two neighbors of some equality relation vertex in RH are equal. If all these tests are successfully add the tuple obtained by concatenating the components of this element, to the set $Ans(RH)$.

The above test depends on the number of equality relation vertices of the graph RH and, clearly, can be implemented in linear time. Also, some optimization of the construction can be considered; for example, if some set $Ans(\mathbb{Q}_{RH}^j, \mathcal{R}^j(SG^j, \mathcal{M}^j))$ is empty, then $Ans(RH)$ is also empty.

Finally, the answer to \mathbb{Q} over the CG mixer $\mathbb{M}(IS^1, \dots, IS^n)$ is the union of the answers to $RH \in \mathbb{R}(SQ)$:

Definition 9. (Answer to a qCG over a CG Mixer)

Let $\mathbb{M}(IS^1, \dots, IS^n) := (SG^0, \mathbb{R})$ be a CG Mixer. If $\mathbb{Q} = [SQ, \text{arity}, X, \lambda'_Q]$ is a legal qCG for SG^0 , then the answer to \mathbb{Q} over $\mathbb{M}(IS^1, \dots, IS^n)$ is

$$Ans(\mathbb{Q}, \mathbb{M}(IS^1, \dots, IS^n)) := \cup_{RH \in \mathbb{R}(SQ)} Ans(RH).$$

Therefore a CG mixer can be viewed as an integrated schema of the individual sources, which directs every user query to the appropriate sources, using a set of rules. Individual query results are then combined and presented to users.

5 Conclusions and Future Work

In this paper we presented a Conceptual Graph approach to describing data in a distributed decision support system for brain tumour diagnosis. Our work is theoretical and it explains how Conceptual Graphs expressivity and easy plug in capabilities benefit such system.

At the moment we do not explicitly represent knowledge regarding problem solving methods. That is to say, our approach captures only the static model rather than the inference procedures. Typical examples of the former are “patient”, “particular type of tumour”, “MRS scans with their parameters”, etc. while examples of the latter are “due to the fact that ... the tumour is malignant” or “peak areas with ... characters suggest ...”. Future work will address extending the KOS (and subsequently the CG Mixer) with rules to address this problem.

On the other hand, a medical diagnosis is normally a complicated process with ambiguity and uncertainty which cannot be entirely and precisely formalised in an inference model good for taxonomic knowledge. This, however, does not deny the merit of building a reasoning system on top of HADOM to provide moderate suggestions and warnings to clinicians instead of replacing them. We are also investigating a Conceptual Graph based case base reasoning approach for HealthAgents.

References

1. Arús, C., Celda, B., Dasmahapatra, S., Dupplaw, D., González-Vélez, H., van Huffel, S., Lewis, P., Lluch, M., i Ariet, M.L., Mier, M., Peet, A., Robles, M.: On the design of a web-based decision support system for brain tumour diagnosis using distributed agents. In: WI-IATW'06. 2006 IEEE/WIC/ACM Int Conf on Web Intelligence & Intelligent Agent Technology, Hong Kong, pp. 208–211. IEEE, Los Alamitos (2006)
2. Chein, M., Mugnier, M.-L.: Conceptual graphs: Fundamental notions. *Revue d'Intelligence Artificielle* 6(4), 365–406 (1992)
3. Chein, M., Mugnier, M.-L., Simonet, G.: Nested graphs: A graph-based knowledge representation model with FOL semantics. In: Proc. of the 6th Int'l Conf. on the Principles of Knowl. Repres. and Reasoning (KR'98), pp. 524–535. Morgan Kaufmann, San Francisco (1998)

4. Croitoru, M., Compatangelo, E.: Conceptual graph projection: a tree decomposition-based approach. In: Doherty, P., Mylopoulos, Welty, C. (eds.) Proc. of the 10th Int'l Conf. on the Principles of Knowledge Representation and Reasoning (KR'2006), pp. 271–276. AAAI, Stanford, California, USA (2006)
5. Dau, F.: Query Graphs with Cuts: Mathematical Foundations. In: Blackwell, A.F., Marriott, K., Shimojima, A. (eds.) Diagrams 2004. LNCS (LNAI), vol. 2980, pp. 32–50. Springer, Heidelberg (2004)
6. Sowa, J.F.: Conceptual Structures: Information Processing in Mind and Machine. Addison-Wesley, London, UK (1984)
7. Tate, A.R., Underwood, J., Acosta, D.M., Julia-Sape, M., Majos, C., Moreno-Torres, A., Howe, F.A., van der Graaf, M., Lefournier, M.M., Murphy, F., Loosemore, A., Ladroue, C., Wesseling, P., Bosson, J.L., Simonetti, A.W., Gajewicz, W., Calvar, J., Capdevila, A., Wilkins, P., Bell, A.C., Remy, C., Heerschap, A., Watson, D., Griffiths, J.R., Arus, C.: Development of a decision support system for diagnosis and grading of brain tumours using in vivo magnetic resonance single voxel spectra. *NMR Biomed* 19, 411–434 (2006)