Conceptual Graphs as a Knowledge Representation Core in a Complex Language Learning Environment

Galia Angelova[1], Ani Nenkova[1],
Svetla Boycheva[2], and Toma Nikolov[2]

[1] Linguistic Modelling Lab, CLPP,
Bulgarian Academy of Sciences,
25A Acad. G. Bonchev Str,
1113 Sofia, Bulgaria
{ani, galja, toma}@lml.bas.bg
[2] Department of Information Technology,
Faculty of Mathematics and Informatics,
Sofia University "St. Kliment Ohridski"
5 J. Bauchier Blvd., 1164 Sofia, Bulgaria
svetla@fmi.uni-sofia.bg

Abstract. This paper describes briefly the application of Conceptual Graphs (CG) in a tutoring environment for teaching English terminology and emphasises on knowledge acquisition results in the domain of finances. The project faces on the one hand the necessity to support an intuitive conceptual representation, providing simple graphical visualisation of domain knowledge to the learner, and on the other hand the necessity to integrate formal techniques for Natural Language Understanding (NLU) allowing analysis of the learners' answers. The paper shows that in a practically situated task-dependent paradigm, most ontological choices — granularity of concept types, conceptual relations, explicit and implicit concept hierarchy, are influenced by the task requirements. We describe the specific aspects of the translation of CG to First Order Logic (FOL), needed in order to use domain knowledge as a background for proving the correctness of learner's utterances and evaluate the complexity of the whole knowledge-based paradigm within this project.

1 Introduction

Although very actively approached, conceptual modelling is far from ultimate solutions concerning principles of Knowledge Acquisition (KA) and Knowledge Engineering (KE) at all levels of building ontologies: upper, middle, and domain models (for discussion of upper models see e.g. [1]). The variety of the available ontologies and the many perspectives to conceptual representations yield some doubts that complex domains can be modelled in a more universal, task-independent way. Meanwhile many researchers continue attempts for aligning

ontologies (see [2] and comments in [3]). Thus, given the current state-of-the-art in ontology engineering, every concrete project, that aims at a particular knowledge-based task in a certain domain, requires some activity for manual acquisition of domain knowledge and proper goal-oriented combination of middle and upper models from well-known knowledge resources like Cyc, WordNet, MikroKosmos, Sensus etc. (see URLs [4]–[7]).

This paper focuses on knowledge usage and knowledge acquisition results in the area of finances in a tutoring environment, assisting non-native English speakers in English terminology learning. Section 2 presents the overall framework where the KA task is performed — i.e. the project LARFLAST , which aims at the development of a holistic learning environment where the student accomplishes three basic tasks (reading teaching materials, performing test drills and discussing his/her own learner model with the system). Section 3 shows that in such a practically situated task-dependent paradigm, like the one we face in this project, most ontological choices are influenced by the task requirements: the granularity of concept types, the conceptual relations, as well as the engineering of the explicit and implicit concept hierarchies. Presenting fragments of our ontology, we provide examples for illustration of the complexity of the whole knowledge-based paradigm within this project. Section 4 describes the specific aspects of the translation CG $\rightarrow$ FOL. The conclusion is given in Section 5.

2 The Project Environment: User Needs vs. Internal Representations

LARFLAST aims at the development of a Web-based learning environment where the student accomplishes three basic tasks (reading teaching materials, performing test drills and discussing his/her own learner model with the system). The project is oriented mostly to learners in Eastern Europe, who need English language competence as well as expertise in correct usage of English financial terms given the fact that finances are somewhat new but increasingly important domain for these users. Thus LARFLAST attempts at finding some balance between the following goals:

- to test students' language and conceptual knowledge (drill checking is partly done by the HPSG-like formal semantic environment PARASITE, see section 2.1;
- to give enough domain knowledge and relevant English terms, and
- to find easy ways of student-system communication and discussion of learner misconceptions by diagrammatic representations, which are considered a powerful expressive language (see the discussion of Open Learner Model technique in section 2.2).

This ambitiously formulated knowledge-base paradigm implies on the one hand the necessity to support an intuitive conceptual representation (providing simple graphical visualisation of domain knowledge and learner model facts to

the user), and on the other hand the necessity to integrate formal techniques for NLU, allowing for analysis of the users' answers to drills where the student is given the opportunity to type in free natural language text. Fig. 1 illustrates the central role of CG as basic knowledge representation formalism. The domain knowledge in finances is encoded in a Knowledge Base (KB), which is the primary knowledge resource in the learning environment. In addition to the manual knowledge acquisition and update in graphical format, the KB-related software supports translation into three CG formats: FOL, CGIF and Prolog representation. The workbench with basic functionality for CG acquisition and representation already exists (see http://www.larflast.bas.bg:8080 and [8], which discusses in detail the www-workbench CGWorld).

2.1 PARASITE (PrAgmatics: Reasoning About the Speaker's attITudEs)

PARASITE is a complex environment for analysis and interpretation of natural language utterances, developed by Allan Ramsay [9]. The general framework for linguistic analysis contains several elements. Lexical processing is provided by (i) a dictionary, (ii) a categorial description of English morphology; and (iii) a set of morphotactic rules which describe the spelling changes at morpheme boundaries. Syntactic description of English grammar is given by a minimal set of HPSG-like schemata. Syntactic processing is performed by a head-corner chart parser. Domain knowledge is not applied at the stages of lexical and syntactic analysis. Additional levels of the input analysis are:

- Semantic analysis: the syntactic analysis underpins the construction of a logical form for the input text (learner's answer to a drill). The parameterised logical form is derived semi- compositionally. This enables PARASITE to produce a range of different interpretations, depending on the requirements of the application. To assure correct semantic analysis of learner's utterances, domain knowledge is translated from CG to PARASITE Meaning Postulates (MP) via FOL.
- Model construction: the information contained in the logical form is anchored and then used for constructing a model. This process, which employs a theorem prover, combines the information that is explicitly presented in the logical form with various kinds of background information. The result of the combination process is a model, which constitutes the speaker's current world model. The CG type hierarchy is used as background knowledge.

Section 4 gives more details concerning the translation of CG to MP. We discuss briefly the contribution of the Bulgarian team, which makes experiments, seeking to find proper ways for integration of the rather complex PARASITE system in language learning tasks.

2.2 Interaction with the User by Open Learner Model Techniques

In contrast to the formal representation necessary for proving the domain correctness of the learner's answer, the system-user interface has to satisfy require-

ments like high expressiveness, simplicity of the output etc. In LARFLAST, the interaction with the learner is provided by Open Learner Model (OLM) [10], [11] — a technique employing diagrammatic representations for carrying out system-user dialogs.

Following some new trends in the development of tutoring systems, OLM externalises the Learner Model and thus tries to engage the student more deeply in the learning process. As suggested in [12], communication by diagrams facilitates learning. The www-workbench CGWorld, developed for the LARFLAST project ([8]), provides a coloured visualisation of different ontological perspectives; the assumption is that this is a better way to explain them especially to a foreigner.

Another important OLM motivation was the conclusion of a user study, performed in 1998 by showing CG drawings to non-specialists and asking them "what does it mean" [10], [11]. Most of the people understood roughly the drawings although they had doubts concerning the arrow direction. Comparing reading of English text and looking at a picture, many people shared that the picture, unlike the text, provided very clear visualisation of terms and relations between them (while in the text it is more difficult to differentiate the units). Such observations explain the LARFLAST choice to show to the learner both domain knowledge and facts from the internal Learner Model as CG.

3 Ontological Choices for Acquisition of Conceptual Graphs

Looking for more universal principles and solutions, after all KA aims at the elaboration of a knowledge base fitting to the specific project goals. We consider the choices described below as task-dependent because there might be other ways to model the same domain; for every KA choice we try to answer the questions: Which concepts, relations and facts are important for the LARFLAST user? (i.e. Why do I acquire these KB elements?) and How to encode knowledge in order to better satisfy the specific LARFLAST requirements?

3.1 Choice of Concept Labels and Organization of the Type Hierarchy

As Fig. 1 shows, one of the reasons to support explicitly domain knowledge is that some CG (or relevant projections of them) will be visualised as an explanation of domain facts when student's misconceptions are diagnostisized. This means that (through the facilities of the www-workbench presented in [8]) the student observes directly the internal KB labels, which substantially differs from the case, say, of NL generation, where the KB labels are hidden. Because of this project-specific aspect,

- We partition the types in the ontology according to the features which seem to imply the most important characteristics and differentiation to be communicated to the learner (a foreigner who studies English financial terms).

So, we omit types that are considered as "insignificant for the learner". Fig. 2 presents a sample type hierarchy for security. Another possible classification of securities can be done according to the issuing authority. But since we consider the distinction bond-stock as the central one to be taught, ISSUING_AUTHORITY is connected to SECURITY only in the type definition.

– We choose labels-terms whenever possible. Most financial terms are noun phrases (NPs) containing more than one word. All concept types in Fig. 2 are real terms in financial dictionaries, which are to be considered in the terminology learning course (but there are also some labels, such as the security supertype PRODUCT_OF_FINANCIAL_MARKET,that are not a real-life term). It might be misleading to arbitrary synthesize "dummy labels" for providing a "more ordered" ontology, because the visualisation to the learner might give rise of "wrong impression" about existing collocations of financial terms (language learning is complex also because many collocations of terms actually originate from other word senses, e.g. military security and collective security come from security in the sense of safety). So, we prefer to synthesise somewhat "explanatory" dummy labels (phrases like ISSUED_BY a COMPANY instead of COMPANY_SECURITY). To summarise, in the hierarchy we place either labels-terms, or explanatory dummy labels.

Fig. 2. Sample ontology of financial terms. Focussing on a single concept, graphical visualisation of different perspectives with different colours is considered as a simple and natural way for system-learner communication. To encode the perspective, in the Prolog representation an additional KB predicate isa_kind/4 is used:
isa_kind(PartitionedType, [Subtype(s)], [PartitionKind(s)], 'PartitionName').
The fourth argument of isa_kind is a text string to be used for generation of NL explanations concerning the partitions.

3.2 Granularity of Concepts and Conceptual Relations

LARFLAST aims at the integration of PARASITE as a drill-checking machine performing formal semantic analysis. So it is necessary to provide a translation from CG to meaning postulates in PARASITE, thus supporting the formal reasoning procedures with domain facts. But PARASITE (and NLU in general) treats the NL semantics compositionally, with basic granularity of meanings, which are defined by word senses. Thus we need a technique for shifting granularity, to assure that the domain semantics of complex CG types will be translated correctly to PARASITE meaning postulates. The shifting technique applies type definitions and type expansions in the appropriate way. For instance: the conceptual relation ISSUED_BY has the type definition

relation ISSUED_BY (x,y) is
[SECURITY: x] ← (THEME) ← [ISSUE] → (AGNT) →
[ISSUING_AUTHORITY: y].

In this way we may obtain KB facts with suitable "cascade" granularity: one encoding to be shown to the learner, for instance the phrasal explanation ISSUED_BY, e.g. in the graph

[BOND] → (ISSUED_BY) → [COMPANY].

and another encoding with the corresponding word-by-word granularity, provided by type expansion (to be further translated to FOL as a PARASITE meaning postulate), e.g.

[BOND] ← (THEME) ← [ISSUE] → (AGNT) → [COMPANY].

Such KA approach requires a very careful and therefore time-consuming elaboration of all KB concepts and conceptual relations.

3.3 Encoding Different Kinds of Partitions in One Hierarchy

There are many ways to partition things, at least because of the different goals and many view-points that might exist. The compact hierarchy at Fig. 2 encodes several kinds of partitions in one lattice. Fig. 3 shows mixed partitions with assigning one isa_kind clause per partition. At present the following ontological perspectives are considered:

Fig. 3. Compact representation of different perspectives in one lattice

Partition into Natural and Role Subtypes Natural subconcepts are defined in [13] as "classified according to unchangeable features", while ROLE is a different perspective for classification. For instance, MAN and WOMAN seem natural subconcepts of PERSON but CHILD and ADULT are roles. A similar example in the domain of finances is a classification of DEALERs as BULLs and/or BEARs. But as far as one dealer can be simultaneously bear for one client and bull for another, the bull/bear distinction is a role-partition. Examples of isa_kind/4 are presented in Fig. 2 and Fig. 3, but here we give one more example concerning the role:

[isa_kind(dealer, [bear, bull], [role], ").

(to be read: The classification of dealer into bear and bull is role without name). Thus the isa_kind predicate provides names and strings for pattern-oriented generation of explanatory texts.

Disjoint and Exhaustive Partitions The default partition at Fig.2 is a joint and unexhaustive classification into natural types. But there are other kinds of partitions in our domain. For instance, one may classify all the securities according to their interests: FIXED or NOT (see Fig. 3 (b), this classification is important for recognition of typical situations of how to deal with securities). This is one possible partition into subtypes, according to the "interests"-perspective. Then, to encode the fact the such a classification is exhaustive and disjoint, we define

```
isa_kind(security, [floating_rate_security, fixed_interest_security],
         [exhaustive, disjoint], 'depending on the interest').
```

This means that

- all securities are either floating_rate_security OR fixed_interest_security (disjoint, i.e. the two subconcepts do not intersect in contrast to say bond and fixed_interest_security);
- there are no other kinds of securities according to this classification perspective (exhaustive).

Our current KA experience shows that in a complex domain like finances one cannot simplify the KA task too much, i.e. one needs mixed perspectives in classification. On the other hand, the learner will have partial and simplified views to small sub-hierarchies, which are relevant to the topics discussed at the particular point. In section 4 below we discuss specific aspects of translation to FOL, which concern the ontological structure presented here.

3.4 Choice of Conceptual Relations and Sample KB Facts

Type definitions of several concepts from the sample type hierarchy are shown in Fig. 4. We illustrate the types BOND and STOCK, as well as the conceptual relations ISSUED_BY, HOLDED_BY, DEPENDING_ON. Since these conceptual relations with granularity "more than one word" are to be translated to "word-by-word" representation, we carefully choose the set of "elementary" conceptual relations like AGNT, THME, OBJ, INST to keep close track to the thematic roles of the verbs-events, where relations like ISSUED_BY are to be translated by type expansion. At the moment, since we are still experimenting with translation of CG to FOL and meaning postulates for further proving in the LARFLAST NLU application, we avoid CG contexts and replace them by concept types with granularity "more than one word", which after proper type expansion are to be turned to CG allowing word-by-word granularity of the further FOL-representation.

Fig. 4. Sample KB in domain of finances

4 Translation to FOL and Meaning Postulates for NLU Analysis

The first step of translating CG to MP in PARASITE is the CG *rightarrow* FOL translation. Applying the standard algorithm in [14], we obtain a FOL predicate for every CG from the KB (but certainly not all CG are applied for proving the correctness of learnerns' answers, so this translation works for selected graphs in order to provide a smaller MP set where the correctness of the learner's input is checked).

There are some specific aspects in the CG$\rightarrow$FOL translation we make. During the translation process, we have to justify the thematic roles to be used in the NLU-analysis; these thematic roles most generally correspond to the KB conceptual relations. However, not all of the conceptual relations are applied in NLU. The most typical difference is in the ATTR-treatment. The graph

[BALL] → (ATTR) → [RED]
is translated to FOL as

```
exists x,y BALL(x) & RED(y) & ATTR(x,y)
```

while in NLU this fact is recorded as

```
exists x BALL(x) & RED(x),
```

since in NLU the adjectives are treated in the logical form differently from nouns. Thus is acquire ATTR for connection of Noun-to-Adjective type labels only. Another very specific aspect concerns the CHAR conceptual relation, which we use as encoding of "has" and "is characterised by". In the manual acquisition special attention is paid, so that such constructions, requiring verb-realization, never appear as ATTR and thus the distinction is preserved in subsequent transition to FOL.

The MP in PARASITE are obtained after one more step in translating the FOL-predicate to the specific MP internal form. The next MP present the internal format of graphs given in section 3.2. The thematic roles — theta-triples — are the MP representation of conceptual roles:

relation ISSUED_BY (x,y) is
[SECURITY: x] ← (THEME) ← [ISSUE] → (AGNT) →
[ISSUING_AUTHORITY: y].
Translated as a MP, the relation definition looks as follows:

```
lexicalMP(
forall(X,forall(Y::{theta(X,$issued_by,Y)},
                    security(X)&issuing_authority(Y)&
              exists(Z::{issue(Z)},theta(Z,$theme,X)&theta(Z,$agent,Y))))
).
```

The graph [BOND] → (ISSUED_BY) → [COMPANY]. is translated to the MP:

```
lexicalMP(
forall(X::{bond(X)},
               exists(Y::{company(Y)},theta(X,$issued_by,Y)))
).
```

The graph [BOND] ← (THEME) ← [ISSUE] → (AGNT) → [COMPANY]. is the MP:

```
lexicalMP(
forall(X::{bond(X)},
                    exists(Z::{issue(Z)}, theta(Z,$theme,X)&
                       exists(Y::{company(Y)},theta(Z,$agent,Y))))
).
```

4.1 Integration of PARASITE in LARFLAST

In addition to the domain KB and the corresponding lexicon with domain words, a necessary elaboration is to develop a prover for checking whether the given user answer "matches in some respect" to the preliminarily defined correct answer(s). Matching in this case means that the semantic derivations of the user's answer include some necessary subset of PARASITE inference of the correct answer. This necessary subset is the intersection of PARASITE inferences of all correct answers (since answers can vary depending of their detailness, use of synonyms and paraphrases etc.). At present we perform experiments with simultaneous (i) acquisition of KB fragments, (ii) construction of suitable and sensible set of drills for testing learner's understanding of the domain facts and perspectives, and (iii) ways of interpreting the mismatches between learner's answers and expectations as mistakes pointing to erroneous learner's knowledge.

4.2 Hidden Hierarchy in MPs

Using only the MPs definitions of the concepts, it is possible to observe the child/parent concepts not only according to the explicit CG type hierarchy, but according to implicit perspectives of classification. We can construct classification tree if we build partitions depending on different values of some of the agents, objects, locations, recipients etc. and to see the semantic of the concepts from different (implicit) points of view. For instance, let us look at the type definitions of SECURITY, BOND and STOCK in Fig. 4. We wish to create their classification tree according to the HOLDER. An especially developed module SORT-CONCEPTS revises all MP, in this case

```
lexicalMP(
forall(X::{security(X)},product_of_financial_market(X)&
           exists(Y::{issuing_authority(Y)},theta(X,$issued_by,Y)&
                exists(Z::{holder(Z)},theta(X,$holded_by,Z))))
).

lexicalMP(
forall(X,security(X)=> exists(Y,of(Y,lambda(Z,issuing_date(Z)),X)))
).


lexicalMP(
forall(X,security(X) => exists(Y,of(Y,lambda(Z,maturity_date(Z)),X)))
).

lexicalMP(
forall(X,security(X)=> exists(Y,of(Y,lambda(Z,currency(Z)),X)))
).

lexicalMP(
forall(X,security(X) => exists(Y,of(Y,lambda(Z,nominal(Z)),X)))
```

```
).

lexicalMP(
 forall(X::{bond(X)}, security(X)&
       exists(Y::{issuing_authority(Y)},theta(X,$issued_by,Y)&
         exists(ZV,of(ZV,lambda(Z,interst(Z)),X))&
            exists(T::{bondholder(T)},theta(X,$holded_by,T)&
             exists(R::{recieve(R)},theta(R,$agent,T)&theta(R,$theme,ZV)&
              exists(U::{give(U)},theta(U,$agent,T)&
                exists(V::{credit(V)},theta(U,$theme,V)&
                  exists(W::{issuing_authority(W)},theta(V,$to,W))))))))
).

lexicalMP(
 forall(X::{stock(X)},security(X)&
       exists(Y::{corporation(Y)},theta(X,$issued_by,Y)&
         exists(ZV,of(ZV,lambda(Z,divident(Z)),X))&
            exists(T::{stockholder(T)},theta(X,$holded_by,T)&
             exists(R::{recieve(R)},theta(R,$agent,T)&theta(R,$theme,ZV)&
               exists(U::{posses(U)},theta(U,$agent,T)&
                exists(V::{share(V)},theta(U,$theme,V)&
                  exists(W::{corporation(W)},theta(V,$of,W))))))))
).
```

First it retrieves all concepts defined as security. After that, those of them having HOLDED- BY-relation are selected. The next step is to sort the resulting set of concepts, depending on the HOLDER. As a result the classification shown in Fig. 3 (a) is obtained.

This simple algorithm allows us to check automatically the consistency of concept and relation definitions in the KB. Since the knowledge-based approach described here leads to an enormous complexity in the internal logical representations, the possibility for performing such visually simple consistency checks is highly desirable and appreciated.

5 Conclusion and Further Work

In this paper we describe the contribution of the Bulgarian team in the project LARFLAST. We deal with acquisition of conceptual graphs and checking the correctness of the learner's answer by mapping it against expectations.

On the one hand, the choice of terms as internal KB labels is a good prerequisite for further ontology standardisation and alignment, since terms are commonly accepted conceptual units; additionally, the choice of conceptual relations close to some approved set of NLU thematic roles is also good for further KB reuse. On the other hand, however, putting together CG and formal NLU requires very precise elaboration of knowledge acquisition decisions and translation algorithms. We are still on the way of building the first project prototype with a realistic drill set.

Acknowledgements: The work reported here is possible only within the framework of project LARFLAST, where OLM and PARASITE are supplied by other project partnets. The authors are also grateful to at least four other team members. Kristina Toutanova developed in Prolog the standard CG→FOL translation module in the workbench CGWorld. Irena Vitanova checked as domain expert the knowledge presented in figures 1 and 4 (as well as many other graphs). Albena Sokolova and Stefan Dimov worked on the sample KB acquisition.

References

1. Fridman, N., Hafner, C.: The State of the Art in Ontology Design. A Survey and Comparative Review. AI Magazine Vol. 18(3) (1997), 53–74.
2. Hovy, E.: Progress on an Automatic Ontology Alignment Methodology, 11/7/1997, see http://www.teknowledge.com/HPKB/meetings/meet102497/hovy/
3. Sowa, J. Knowledge Representation: logical, philosophical, and computational foundations. BROOKS/COLE, 2000.
4. Cyc, see http://www.cyc.com
5. WordNet, see http://www.cogsci.princeton.edu/ wn/.
6. MikroKosmos, see http://crl.nmsu.edu/Research/Projects/mikro/.
7. Sensus, see http://www.isi.edu/natural-language/resources/sensus.html.
8. Dobrev, P., Toutanova, Kr.: CGWorld — A Web-Based Workbench for Conceptual Graphs Management and Applications. Submitted to ICCS-2000.
9. PARASITE, see http://ubatuba.ccl.umist.ac.uk/ for detailed description, demo and numerous relevant papers.
10. Dimitrova, V., Dicheva, D., Brna, P., Self, J.A. A knowledge based approach to support learning technical terminology. In P. Navrat & H. Ueno (eds.) Knowledge-Based Software Engineering, Proc. of the 3rd Joint Conference on Knowledge-Based Software Engineering, 1998, Frontiers in Artificial Intelligence and Applications, vol. 48, IOS Press, pp. 270–277.
11. Dimitrova, V., Self, J.A., Brna, P.: The interactive maintenance of open learner models. In Lajoie S.P. & Vivet M., eds., Artificial Intelligence in Education: Open Learning Environments: New Computational Technologies to Support Learning, Exploration, and Collaboration, 1999. Frontiers in Artificial Intelligence and Applications, vol. 50, IOS Press, pp. 405–412.
12. Brna, P., Cox, R., Good, J.: Learning to think and communicate with diagrams. Artificial Intelligence Review, to appear.
13. Sowa , J. Conceptual Graphs Summary. In: T. Nagle, J. Nagle, L. Gerholz, P. Eklund (Eds.), Conceptual Structures: Current Research and Practice, Ellis Horwood, 1992, pp. 3–52.
14. Sowa, J. Conceptual Structures: Information Processing in Mind and Machine. Addison-Wesley, Reading, MA, 1984.
15. LeARning Foreign LAnguage Scientific Terminology, INCO Copernicus Joint Research Project 977074, funded by the European Commission (Nov.1998–Oct.2001), with 7 partners (see URLs www-it.fmi.uni- sofia.bg/larflast/, www.larflast.bas.bg and the HLT-site www.linglink.lu/hlt/projects/inco-larflast/)