Semantic Analysis in the Automation of ER Modelling through Natural Language Processing

N. Omar[1], P. Hanna[2] and P. Mc Kevitt[3]

[1]Department of Computer Science, Faculty of Information Science and Technology, UKM, 43600, Bangi, Selangor, Malaysia

[2] School of Computing and Mathematics, Faculty of Engineering, Jordanstown Campus, University of Ulster, Newtownabbey BT37 0QB, Northern Ireland, UK

[3] School of Computing and Intelligent Systems, Faculty of Engineering, Magee College, University of Ulster, Derry/Londonderry BT48 7JL, Northern Ireland, UK

Abstract- **This paper deals with the problem of extracting semantic knowledge in the production of ER models from natural language specifications. The application of semantic heuristics is proposed as the strategy to obtain the relevant ER elements such as entities, attributes and relationships from the specifications. Earlier research has shown that syntactic heuristics produced good results in identifying the relevant and correct results of the ER elements in terms of recall and precision. The inclusion of the semantic lexical knowledge is hoped to further improve the results. The semantic heuristics may later be implemented as part of a natural language tool in the generation of the ER models.**

I. Introduction

Database modelling can be a daunting task to both students and designers alike due to its abstract nature and technicality. Much research has attempted to apply natural language processing in extracting knowledge from requirements specifications with the aim to design databases. However, research on the formation and use of heuristics to aid the construction of logical databases from natural language has been scarce.

This paper proposes the use of semantic heuristics in the generation of ER models from natural language specifications. The semantic heuristics will be used to determine the relevant ER elements such as entities, attributes and relationships from the database specifications. The application of the heuristics would be realised through the extension of a developed tool called ER-Converter [12, 13]. Syntactic heuristics have been implemented in ER-Converter. ER-Converter has been evaluated against a set of database problems and achieved 90% recall and 85% precision. In order to further improve the accuracy of the results, semantic heuristics are proposed. Though this is a semi-automatic transformation process, the tool aims to provide minimal human intervention during the process.

II. Background and Previous Work

This section provides a brief summary on data modelling which introduces the concept of ER Model and reviews the previous work that applies natural language processing to Databases. The existing tools, techniques and limitations are discussed. Some of the work like DMG[15] provides a basis for the development of new heuristics applied in ER-Converter.

A. *Overview of Data Modelling*

The first step in designing a database application is to understand what information the database must store. This step is known as requirements analysis. The information gathered in this step is used to develop a high-level description of the data to be stored in the database. This step is referred to as conceptual design, and it is often carried out using the ER model. ER models are built around the basic concepts of entities, attributes, relationships and cardinality. An entity is an object that exists in the real world and is distinguishable from other objects. These are typically derived from nouns. Examples of entities include the following: a “student”, an “employee” and a “book”. A collection of similar entities is called an entity set. An entity is described using a set of attributes. The attributes of an entity reflect the level of detail at which we wish to represent information about entities. Attributes may be derived from adjectives and adverbs. For example, the “Student” entity set may have “ID_number”, “Name”, “Address”, “Course” and “Year” as its attributes. A relationship is an association among two or more entities. Relationships can be typically derived from verbs. For example, we may have a relationship from this sentence: A student may “take” many courses. “take” implies a relationship between the entity “student” and “course”. Cardinality represents the key constraint in a relationship. In the previous example, the cardinality is said to be many-to-many, to indicate that a student can take many courses and a course can be taken by many students. In an ER diagram, an entity is normally represented by a rectangle. An ellipse usually represents an attribute meanwhile a diamond shape shows a relationship. Cardinality is represented by 1 for the one-sided and M for the many-sided.

B. *Applying Natural Language Processing (NLP) to Databases*

Much work [2,5,6,15] has attempted to apply natural language in extracting knowledge from requirements specifications or dialogue sessions with designers with the aim to design databases. Dialogue tool [2] is a knowledge-based tool applied to the German language for producing a skeleton diagram of an Enhanced Entity-Relationship (EER) model. This

1-4244-0220-4/06/$20.00©2006 IEEE

tool is part of a larger database design system known as RADD (Rapid Application and Database Development) which consists of other components that form a complex tool. In order to obtain knowledge from the designer, a moderated dialogue is established during the design process. The transformation of the structure of natural language sentences into EER model structures is a process which is based on heuristic assumptions and pragmatic interpretation. The aim of the pragmatic interpretation is the mapping of the natural language input onto EER model structures using the results of the syntactic and semantic analyses. One major limitation in this system is that the accuracy of the EER model produced depends on the size and complexity of the grammar used and the scope of lexicon.

ANNAPURNA [5] is project aimed to provide a computerized environment for semi-automatic database design from knowledge acquisition up to generating an optimal database schema for a given database management system. ANNAPURNA concentrated on the phases concerned with acquiring the terminological rules. The first step in acquisition of the terminological knowledge involves extracting the knowledge from queries and rules that have the form of natural language expressions. The knowledge obtained would then be put into the form of S-diagrams. An S-diagram is a graphical data model which can be used to specify classes (for example room and door), subclass connections between classes (for example rooms and doors are physical objects) and attributes. The limitation of the above work is that the use of S-diagrams performs best when the complexity is small.

DMG [15] is a rule based design tool which maintains rules and heuristics in several knowledge bases. A parsing algorithm which accesses information of a grammar and a lexicon is designed to meet the requirements of the tool. During the parsing phase, the sentence is parsed by retrieving necessary information from the grammar, represented by syntactic rules and the lexicon. The parsing results are processed further on by rules and heuristics which set up a relationship between linguistic and design knowledge. The DMG has to interact with the user if a word does not exist in the lexicon or the input of the mapping rules is ambiguous. The linguistic structures are then transformed by heuristics into EER concepts. Though DMG proposed a large number of heuristics to be used in the transformation from natural language to EER models, the tool has not yet been developed into a practical system.

E-R generator [6] is another rule-based system that generates E-R models from natural language specifications. The E-R generator consists of two kinds of rules: specific rules linked to semantics of some words in sentences, and generic rules that identify entities and relationships on the basis of the logical form of the sentence and on the basis of the entities and relationships under construction. The knowledge representation structures are constructed by a natural language understander (NLU) system which uses a semantic interpretation approach. There are situations in which the system needs assistance from the user in order to resolve ambiguities such as the attachment of attributes and resolving anaphoric references.

CM-Builder [9] is a natural language based CASE tool which aims at supporting the analysis stage of software development in an object-oriented framework. The tool uses natural language processing techniques to analyse software requirements documents and produces initial conceptual models represented in Unified Modelling Language. The system uses discourse interpretation and frequency analysis in producing the conceptual models. CM-Builder still has some limitation in the linguistic analysis. For example, attachment of postmodifiers such as prepositional phrases and relative clauses is limited. Other shortcomings include the state of the knowledge bases which are static and not easily updateable nor adaptive.

Heuristics, based on linguistic rules, are reported to be utilized in many of the systems like ANNAPURNA [5], DMG [15] and RADD [2]. However, only DMG [15] presents a precise set of heuristics used in deriving an EER model. The heuristics presented, however, are mainly based on syntax. This research aims to fill in the gap by proposing a new set of semantic heuristics.

III. SYNTACTIC HEURISTICS TO IDENTIFY ER ELEMENTS

Heuristics represent an indefinite assumption [15], often guided by common sense, to provide good but not necessarily optimal solutions to difficult problems, easily and quickly [16]. Research on the formation and use of heuristics to aid the construction of logical database structures from natural language has been scarce. The only existing work that proposes a large number of heuristics to be used in the transformation from natural language to ER models is DMG [15]. However the work has not been implemented. The authors of DMG proposed both syntactic and semantic heuristics to be applied in extracting knowledge from requirements specifications. Although E-R Generator [6] and RADD [2] utilized heuristics in their work, they do not detail a precise set of heuristics in their approach. Chen [3] suggested that the basic constructs of English sentences could be mapped into ER schemas in a natural way and presented a set of rules to put forward the ideas. Though the set are referred to as “rules”, Chen mentioned that they are better viewed as “guidelines” as it is possible to find counter examples to them. Here we regard Chen’s “rules” as heuristics as they are largely “rules-of-thumb” based on observations rather than theoretically derived. Only heuristics for language syntax are considered and proposed at this stage.

Here, a selection of the syntactic heuristics applied in the transformation from database specifications to the data modeling constructs is presented. These heuristics are gathered from past work [3,14,15] and some are newly formed [12,13]. A complete set of these heuristics can be found in [13]. Some examples in terms of sentences are provided to illustrate the application of heuristics which are context dependent.

1-4244-0220-4/06/$20.00©2006 IEEE

Heuristics to determine entities:

1. Heuristic HE2: A common noun may indicate an entity type [3,15].
2. Heuristic HE3: A proper noun may indicate an entity [3,15].
3. Heuristic HE7: If consecutive nouns are present, check the last noun. If it is not one of the words in set S where S={number, no, code, date, type, volume, birth, id, address, name}, most likely it is an entity type. Else it may indicate an attribute type.

Heuristics to exclude non-potential entity types candidates:

1. Heuristic HEX: A noun such as “record”, “database”, “company”, “system”, “information” and “organization” may not be a suitable candidate for an entity type. For example, “company” may indicate the business environment and should not be included as part of the entity types. Examples:
   a. “An insurance company wishes to create a database to keep track of its operations.”
   b. “An organization purchases items from a number of suppliers.”

Heuristics to determine attributes:

1. Heuristic HA6: Genitive case in the noun phrase may indicate an attributive function [15].
2. Heuristic HA8: If a noun is followed directly by another noun and the latter belongs to set S where S={number, no, code, date, type, volume, birth, id, address, name}, this may indicate that both words are an attribute. Else it is most likely to be an entity.

Heuristics to determine relationships:

1. Heuristic HR1: An adverb can indicate an attribute for relationship [3].
2. Heuristic HR4: A verb followed by a preposition such as “on”, “in”, “by” and “to” may indicate a relationship type. For example: “Persons work on projects.” Other examples include “assigned to” and “managed by”.

Heuristics to determine cardinalities:

1. Heuristic HC2: The adjective “many” or “any” may suggest a maximum cardinality. For example:
   a. “A surgeon can perform many operations.”
   b. “Each diet may be made of any number of servings.”
2. Heuristic HC3: A comparative adjective “more” followed by the preposition “than” and a cardinal number may indicate the degree of the cardinality between two entities. For example: “Each patient could have more than one operation.”

The syntactic heuristics were tested through the implementation of ER-Converter. The approach in the evaluation uses methods for evaluating Information Extraction systems, primarily Message Understanding Conferences (MUC) [8] evaluations i.e. recall and precision. Recall is percentage of all the possible correct answers produced by the system. Precision is the percentage of answers that are correctly identified by the system. In any system, both precision and recall should be as close to 100% as possible. However, in general, an increase in precision tends to decrease recall and vice versa. In the context of this research, the definition of recall and precision below are adopted as used by CM-Builder [9] and new measures are defined. The evaluation results obtained from applying the syntactic heuristics in ER-Converter is 90% recall and 85% precision. In order to improve the accuracy of the results, semantic analysis of the natural language specifications is deemed as a promising approach.

IV. SEMANTIC ANALYSIS

In order to resolve a wider range of problems related to ambiguities in requirements’ specifications such as anaphoric references or nominalization, without pre-processing text or using restricted language, semantic analysis of the sentences may be necessary to handle such issues. Semantic analysis involves a process whereby meaning representations are created and assigned to linguistic inputs [10]. The ‘understanding’ of the results of the parsing, lexical information, context and common sense reasoning is referred to as the semantic interpretation of the text. More expressive power can be added when semantic interpretation is used.

Semantic roles in objects like agent, instrument, source and location [7] may be helpful in interpreting possible elements of the ER model. Semantic roles or sometimes known as thematic roles are conceptual notions which provide a shallow semantic language for characterizing certain arguments of verbs [10]. Table 1 shows some commonly used semantic roles and their definitions.

TABLE 1
SEMANTIC ROLES AND THEIR DEFINITIONS [10]

Semantic role	*Definition*
AGENT	The volitional causer of an event
EXPERIENCER	The experiencer of an event
FORCE	The non-volitional causer of the event
THEME	The participant most directly affected by an event
RESULT	The end product of an event
CONTENT	The proposition or content of a prepositional event
INSTRUMENT	An instrument used in an event
BENEFICIARY	The beneficiary of an event
SOURCE	The origin of the object of a transfer event
GOAL	The destination of an object of a transfer event

The following example illustrates the concept of semantic roles:

“The purchaser (AGENT) sends an order form (THEME) to the supplier (GOAL).”

From the example, the subject, i.e. ‘the purchaser’ acts as an agent as the causer of the event. The object, ‘order form’, has the semantic role ‘THEME’ as it is

1-4244-0220-4/06/$20.00©2006 IEEE

directly affected by the event. 'Supplier' represents the GOAL where it is the destination of the transfer event. Other examples are as follows:

1. They (AGENT) may send their good (THEME) to market (GOAL) through suppliers (INSTRUMENT)
2. The goods (THEME) are sent by truck (INSTRUMENT) to collection centres (GOAL) in larger cities (LOCATION)

In example (1) and (2), different roles are assigned to the subject and object of the sentences depending on the context and the semantic cues. These semantic roles may assist in the pragmatic interpretation of the natural language input to an ER model. For example, the semantic roles agent, goal and instrument may indicate entity types, depending on the context. By deriving suitable heuristics from the semantic roles, it is envisaged that these could assist in deriving ER elements which syntactic heuristics fail to identify. In addition, the semantic heuristics may also add extra evidence on some of the elements that has been identified through syntactic clues. DMG [15] and [11] may provide a basis for such heuristics.

V. CURRENT WORK

At present, the development of the semantic heuristics is still in early stage. Once the heuristics are developed, a manual test will be carried out to test the usability of the heuristics across different domains using a training dataset. Once an optimal set of the heuristics are determined and selected, they will be implemented as part of an extension to the existing tool, ER-Converter[12]. Figure 1 shows the architecture of ER-Converter.

The process begins by reading a plain input text file containing a requirements specification of a database problem in English. For this purpose, a parser is required to parse the English sentences to obtain their part-of-speech (POS) tags before further processing. Part of speech tagging assigns each word in an input sentence its proper part of speech such as noun, verb and determiner to reflect the word's syntactic category [1]. The parser used here is Memory-Based Shallow Parser (MBSP) [4,17]. The results produced is then parsed through a semantic analyser for the semantic analysis. The parsed text is then be fed into ER-Converter to identify suitable data modeling elements from the specification. The task requires several steps to be carried out in order to achieve the desired ER model from the natural language input, each of which is listed as follows:

- Step 1: Part of speech tagging using Memory-based Shallow Parser
- Step 2: Semantic roles assignment using semantic analyser
- Step 3: Read natural language input text into system
- Step 4: Apply syntactic and semantic heuristics and assign weights
- Step 5: Human intervention
- Step 6: Attachment of attributes to their corresponding entity
- Step 7: Attachment of entities to their corresponding relationship
- Step 8: Attachment of entities to their corresponding cardinality
- Step 9: Produce final result

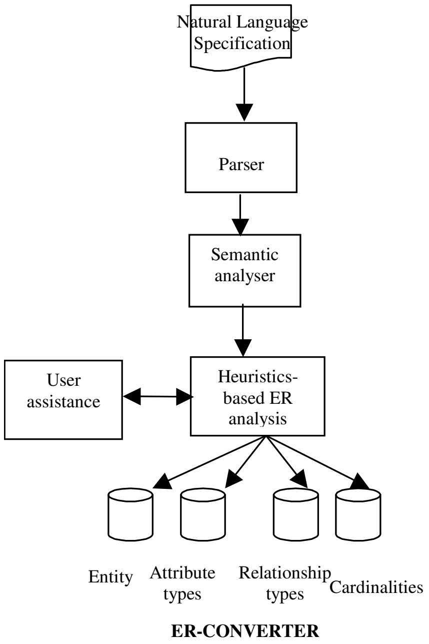

Fig. 1. Architecture of the ER-Converter tool

In this research, weights are used to represent uncertainty when dealing with heuristics. Each of the semantic heuristics will be assigned a specific weight ranging from 0 up to 0.9. The heuristics' weights are assigned according to the confidence level that the event is true. For example, HE2 (one of the syntactic heuristics to determine entity type) states that a common noun may indicate an entity type. It has been given a weight of 0.5. This basically means that 50% of the time this heuristic may produce the correct result, as not all nouns are entity types. Though the assignment of the weights is mainly based on intuition, these weights will be compared and reflected against the results obtained from training set.

VI. CONCLUSION AND FUTURE WORK

We have described an approach of generating ER elements automatically from natural language specifications using a heuristics-based approach. Semantic heuristics are proposed to be utilized in conjunction with the syntactic heuristics to improve the accuracy of the results in producing the ER elements from natural language specifications. The contribution made can be applied in areas such as part of the domain model of an intelligent tutoring system, designed to assist in the learning and teaching of databases and other applications of NLP for database design.

REFERENCES

[1] Brill, E.: A Simple Rule-Based Part of Speech Tagger. In: Proceedings of the Third Conference on Applied Natural Language Processing, ACL, Trento, Italy (1992) pp. 152-155

[2] Buchholz, E., Cyriaks, H., Dusterhoft, A., Mehlan, H., and B. Thalheim.: Applying a Natural Language Dialogue Tool for Designing Databases. In: Proceedings of the First Workshop on Applications of Natural Language to Databases (NLDB'95), Versailles, France (1995) 119- 133.

[3] Chen, P.P.: English Sentence Structure and Entity-Relationship Diagram, Information Sciences, Vol.1, No. 1, Elsevier (1983) 127-149

[4] Daelemans, W., Zavrel, J., Berck, P. and Gillis, S.: MBT: A memory-based part of speech tagger generator. In: Ejerhed, E. and Dagan, I. (eds.), Proc. Of Fourth Workshop on Very Large Corpora, Philadelphia, USA (1996) 14-27

[5] Eick, C. F. and Lockemann, P.C.: Acquisition of Terminology Knowledge Using Database Design Techniques. Proceedings ACM SIGMOD Conference, Austin, USA (1985) 84-94

[6] Gomez, F., Segami, C. and Delaune, C.: A system for the semiautomatic generation of E-R models from natural language specifications. Data and Knowledge Engineering 29 (1) (1999) 57-81

[7] Fillmore, C.J. (1971) Some Problems for Case Grammar. In R.J. O'Brien (ed), 22nd Annual Round Table. Linguistics: Developments of the Sixties-viewpoints of the Seventies, Vol. 24 of Monograph Series on Language and Linguistics. Georgetown University Press, Washington D.C., pp. 35-36.

[8] Grishman, R. and Sundheim, B.: Message Understanding Conference-6: A Brief History. Proceedings of the 16th International Conference on Computational Linguistics, Copenhagen, Denmark (1996) 466-471

[9] Harmain, H.M. and Gaizauskas, R. CM-Builder: A Natural Language-Based CASE Tool for Object-Oriented Analysis. Automated Software Engineering 10 (2) (2003) 157-181

[10] Jurafsky, D. and Martin, J. H. Speech and Language Processing, (2000) Prentice-Hall, New Jersey.

[11] Martinez, P. and Garcia-Serrano, A. (2001) On the Automatization of Database Conceptual Modelling through Linguistic Engineering. Lecture Notes in Computer Science, No. 1959, pp. 276-287.

[12] Omar, N., Hanna, P. and Mc Kevitt, P. Acquisition of Entity-Relationship Models from Natural Language Specifications Using Heuristics, 3rd International Conference on IT and Multimedia, UNITEN, Malaysia (2005) CD-ROM.

[13] Omar, N. Heuristics-Based Entity Relationship Modelling Through Natural Language Processing. PhD Thesis (2004). University of Ulster, UK.

[14] Storey, V.C. and Goldstein, R.C.: A Methodology for Creating user Views in Database Design. ACM Transactions on Database Systems 13 (3) (1988) 305-338

[15] Tjoa, A.M and Berger, L.: Transformations of Requirements Specifications Expressed in Natural Language into an EER Model. Proceeding of the 12th International Conference on Approach, Airlington, Texas, USA (1993) 206-217

[16] Zanakis, S.H. and Evans, J.R.: Hueristic 'Optimization': Why, When and How to use it. Interfaces 11(5) (1981) 84-91

[17] Zavrel, J. and Daelemans, W.: Recent Advances in Memory-Based Part-of-Speech-Tagging In: Actas del VI Simposio Internacional de Communicacion Social, Santiago de Cuba, Cuba (1999) 590-597

1-4244-0220-4/06/$20.00©2006 IEEE