ONTOLOGY-DRIVEN INFORMATION-PROVIDING DIALOGUE SYSTEMS

Annika Flycht-Eriksson, Arne Jönsson, Magnus Merkel and Håkan Sundblad
Department of Computer and Information Science
Linköpings universitet, S-58183 Linköping, SWEDEN
{annfl,arnjo,magme,hakjo}@ida.liu.se

Abstract

In this paper we propose an architecture for dialogue systems that incorporate information extraction, thereby providing natural and efficient access to unstructured information sources. A key feature of the architecture is the use of ontologies as shared domain knowledge sources. We discuss how ontologies can be used for various tasks. with focus on question analysis, dialogue interaction and information extraction.

Introduction

As more and more information has been made available in unstructured and semi-structured text formats the demands on support for users to seek and retrieve information have increased. Moving from Information Retrieval (IR) to Information Extraction (IE) and Question Answering (Q&A) gives users vastly improved means to express queries and techniques for processing, retrieving and presenting documents and answers. For example, with today's Q&A-systems a user can state a (factual) question and expect the relevant portion of a text document containing the answer to the question in return.

A future direction in improving and facilitating the interaction is to allow for dialogue instead of isolated questions. The challenge is to combine the IE techniques and some of the features of Q&A approaches with the dialogue capabilities of dialogue systems (Burger et al. 2001). By a successful combination of these techniques, users would be allowed to access information derived from a large set of, initially unstructured, documents, using dialogue functionalities, such as referring expressions and clarification requests.

One crucial challenge when bringing the two research areas together is the kind of shared knowledge sources such a combined system would utilise to represent linguistic and domain knowledge. A good candidate for the latter is domain ontologies. Ontologies as means of representing and supporting reasoning about domain knowledge are becoming increasingly common, they are for example used in several Q&A systems (cf. Harabagiu et al. 2000).

In this paper we will present an architecture for such a combined system where domain ontologies have a central role. A first version of an application, BIRDQUEST, which supports dialogue interaction to access textual data in a bird encyclopaedia, has been implemented based on this architecture. It will be used to illustrate how domain ontologies can be used for various tasks in such a system, with the focus on the dialogue interaction.

System architecture

One possible way of merging dialogue interaction with information extraction in a practical system is to have two components, one information processing component and one interaction component, which as a basis for their tasks use a set of shared knowledge sources that define the scope of the language and domain, as shown in Figure 1.

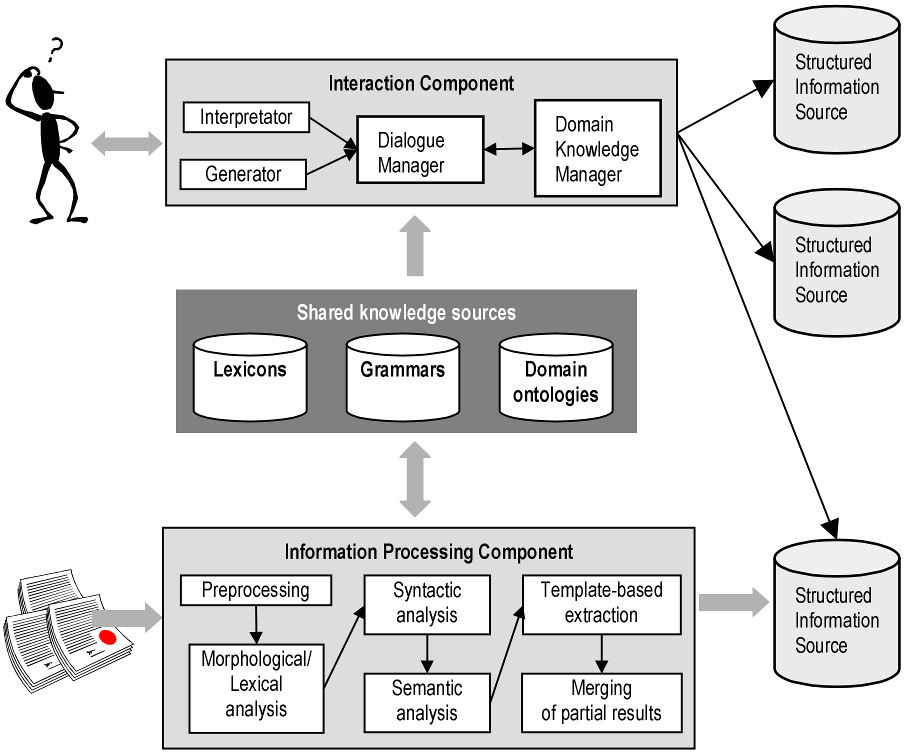

Interaction Component
Interpretator
Generator
Dialogue Manager
Domain Knowledge Manager
Structured Information Source
Structured Information Source
Shared knowledge sources
Lexicons
Grammars
Domain ontologies
Information Processing Component
Preprocessing
Morphological/ Lexical analysis
Syntactic analysis
Semantic analysis
Template-based extraction
Merging of partial results
Structured Information Source

Figure 1. Overview of system architecture. The figure shows the different components and the shared knowledge sources.

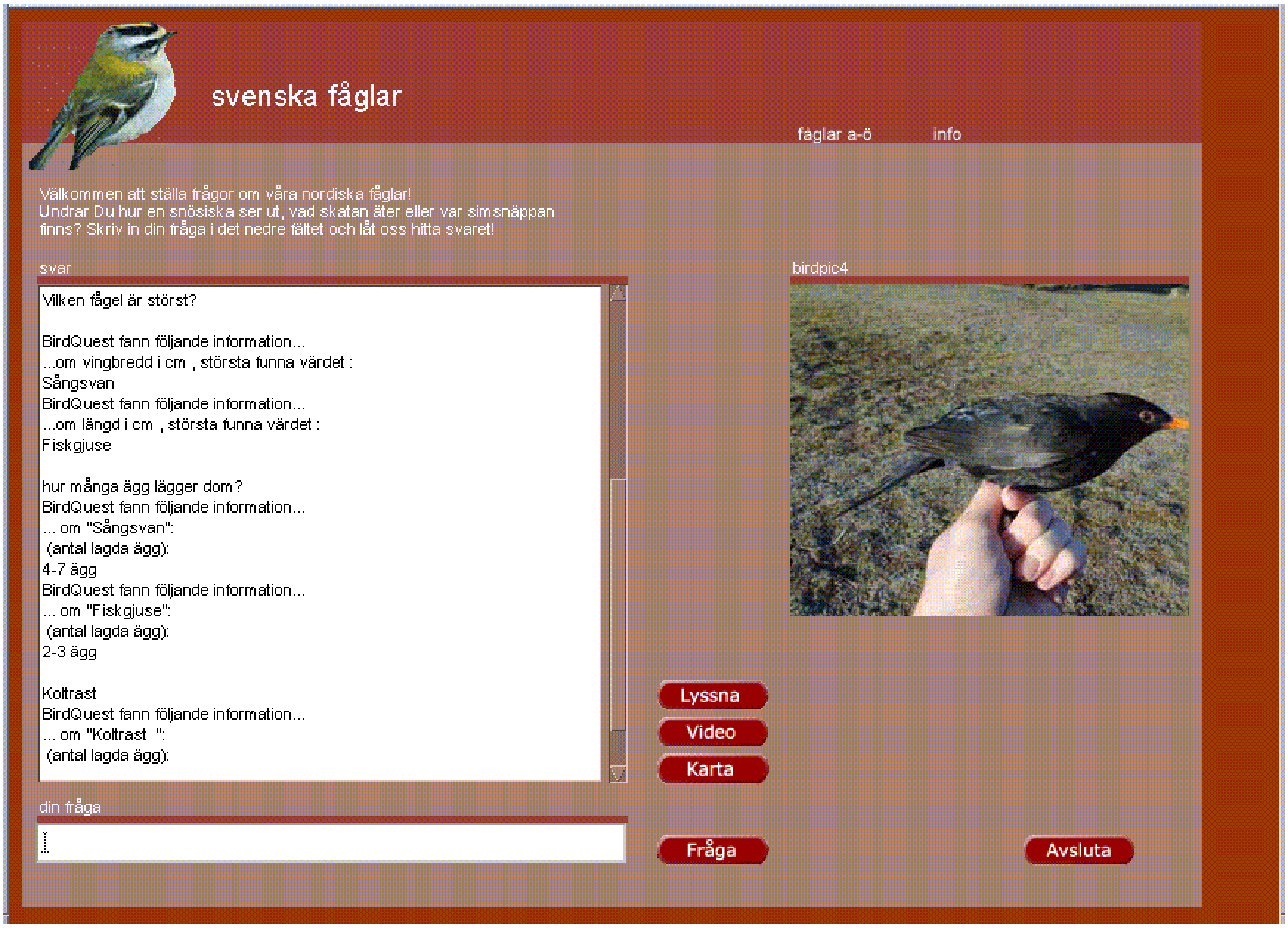
The Information Processing Component takes collections of unstructured or semi-structured documents and transforms them into structured information that can be used by the Interaction Component during interaction with the user. The transformation is based on information extraction techniques, and the documents are analysed in several stages going through lexical and morphological, syntactical, and semantic analysis, in each step adding more structure to the documents. A wide variety of pattern extraction rules are then applied to the documents. The objective is to fill the database with relevant information and ignore text segments that do not meet the information needs of the users.

The Interaction Component is responsible for the dialogue interaction with the user. It collaborates with the user to produce a query and access the structured information sources to retrieve an answer. It consists of four components, the Interpreter, the Dialogue Manager, the Domain Knowledge Manager, and the Generator. The Interpreter takes a user request as input and produces a representation of the question type, expected answer type and a representation of the content. The Dialogue Manager controls the flow of the dialogue by deciding how the system should respond to the user request. The Domain Knowledge Manager retrieves and co-ordinates knowledge from the different information sources. The Generator produces the system utterances based on information from the Dialogue Manager. In this type of dialogue system this means either presenting information texts retrieved from the database, clarification requests or error messages.

The shared knowledge sources comprise lexicons, grammars and domain ontologies. The term ontology is used very differently in various areas of computer science. A general and commonly used definition given by Gruber (1993) is that "*An ontology is a formal explicit specification of a shared conceptualisation*." A more practical view is to consider an ontology as "*a world model used as a computational resource for solving a particular set of problems*" (Mahesh and Nirenburg, 1995), i.e. a database with information about what categories (or concepts) exist in the world/domain, what properties they have, and how they are related to one another. Constructing an ontology that can be shared by the Information Processing Component and the Interaction Component provides a mean for bridging user requests to the information in the unstructured documents.

BirdQuest

Based on the proposed architecture an application, BIRDQUEST, was developed for a web site where people, watching nature programs on TV, can ask questions related to the topic of the TV program, in this case Nordic birds. The information was extracted from a bird encyclopaedia (Staav and Fransson 1991) and a corpus of 329 information requests that was collected from users asking questions on a web page hosted by the Swedish National Television. In BIRDQUEST users can ask questions in natural language and also use menus to enter bird names. The interface is shown in Figure 2.



Välkommen att ställa frågor om våra nordiska fåglar!
Undrar Du hur en snösiska ser ut, vad skatan äter eller var simsnäppan finns? Skriv in din fråga i det nedre fältet och låt oss hitta svaret!

```
Vilken fågel är störst?

BirdQuest fann följande information...
...om vingbredd i cm , största funna värdet :
Sångsvan
BirdQuest fann följande information...
...om längd i cm , största funna värdet :
Fiskgjuse

hur många ägg lägger dom?
BirdQuest fann följande information...
... om "Sångsvan":
 (antal lagda ägg):
4-7 ägg
BirdQuest fann följande information...
... om "Fiskgjuse":
 (antal lagda ägg):
2-3 ägg

Koltrast
BirdQuest fann följande information...
... om "Koltrast  ":
 (antal lagda ägg):
```

Figure 2. The interface of the BIRDQUEST application.

Figure 3 depicts a longer dialogue using BIRDQUEST, which will be used throughout this paper. It is assembled from dialogues with novice users of BIRDQUEST (Flycht-Eriksson and Jönsson, 2003), i.e. although the whole interaction can be realised in BIRDQUEST, the dialogue in Figure 3 is not from one user, but bits and pieces from different users, in order to capture a variety of phenomena.

On important feature of BIRDQUEST is its ability to handle dialogue. A dialogue history allows for fragmentary input that can be interpreted in the context of previous utterances, as seen for instance in U21 in Figure 3. Clarification subdialogues are common when users interact with BIRDQUEST, and play a crucial role since they in many cases are necessary to guide the user to formulate a request that can be answered. Utterances U1-S6 illustrate the use of, and need for, such subdialogues. Utterance U1 "*What do waterfowl look like*" is under-specified since the object, the bird family "Waterfowl", can refer to a number of different species of birds, resulting in a too long answer. Furthermore, the property "Appearance" is vague. Thus, BirdQuest asks for clarification as seen in S2, based on information about which species belong to the given family as gathered from the ontology, and in S4, where the ontology is accessed to retrieve the sub-properties of "Appearance". When the user has chosen a specific property (U5) the request is sufficiently specified.

U1: What do waterfowl look like?
Hur ser en and ut?

S2: Choose one of the following waterfowl: tufted duck, bean goose, whooper swan, gadwall,...
Välj en av följande andfåglar: vigg, sädgås, mindre sångsvan, snatterand,...

U3: gadwall
snatterand

S4: You can recieve information about the size and plumage of a gadwall. Please specify what you are interested in.
Du kan få information om en snatterands storlek och dräkt. Vad vill du ha information om?

U5: size
storlek

S6: A gadwall has a length of 56 cm from beak tip to tail
Information about wingspan is missing for gadwall
En snatterand är omkring 56 cm lång från näbben till stjärtspetsen.
Information om vingbredd saknas för fasan.

U7: What does an ostrich look like
Hur ser en struts ut

S8: Information about non-Nordic birds is missing. The available birds are listed in the menu “Birds a-ö”. Please try again
Information saknas om ickenordiska fåglar. De fåglar som du kan få information om finner du i menyn “Fåglar a-ö”. Försök igen!

U9: What colour are swans?
vilken färg har svanar?

S10: The system failed to interpret the question. Please try again.
Systemet förstod inte frågan, försök igen!

U11: Where do swans live?
Var lever svanar?

S12: Choose one of the following swans: song swan, whooper swan, mute swan.
Välj en av följande svanar: sångsvan, mindre sångsvan och knölsvan.

U13: Where do mute swans live?
Var lever knölsvanar?

S14: Information about a mute swans range:
The mute swan can be found nesting mainly in western Europe, but also in southeast Europe and Tureky.
Information om en knölsvans utbredning:
Knölsvanen förekommer som häckfågel framför allt i västra Europa men också i sydöstra Europa och i Turkiet.

U15: How much does it weigh?
Hur mycket väger den

S16: Failed to interpret the question. Please try again.
Misslyckades tolka meningen. Försök igen!

U17: How long does a mute swan live?
Hur länge lever en knölsvan?

S18: Information about highest age of a mute swan:
Highest age in the wild: 24 years and 3 months
Information om en knölsvans högsta ålder:
Högsta ålder i vilt tillstånd: 24 år och 3 månader

U19: what does a mute swan look like
hur ser en knölsvan ut

S20: You can recieve information about the size and plumage of a mute swan. Please specify what you are interested in.
Du kan få information om en knölsvans storlek och dräkt. Vad vill du ha information om?

U21: plumage
dräkt

S22: Information about eclipse plumage is missing for Mute Swan.
Information about winter plumage is missing for Mute Swan.
Information about summer plumage is missing for Mute Swan.
Information about breeding plumage is missing for Mute Swan.
Information om eklipsdräkt saknas för knölsvan.
Information om vinterdräkt saknas för knölsvan.
Information om praktdräkt saknas för knölsvan.
Information om sommardräkt saknas för knölsvan.

U23: Which birds can I see in Sweden in December
Vilka fåglar kan jag se i Sverige i december

Figure 3. A dialogue with BirdQuest. Pieced together from various real dialogues between BirdQuest and users for illustrative purposes.

Since information extraction is used to populate the database with information the database will have gaps, i.e. for some properties values are missing due to errors made by the information processing component or lack of information in the original text source. This causes unnecessary clarifications, e.g. S20. A related problem is how partial and empty answers should be presented to the user when a vague property has been mapped to several sub-properties, for example "Size" in S6. In such cases BIRDQUEST presents all sub-properties as it helps the user to learn what type of information the system can provide.

BIRDQUEST's knowledge is limited to the type of information found in the bird encyclopaedia. Since users do not know exactly what these restrictions are, questions outside the coverage of the database occur and in most cases the response to these are "*Failed to interpret the question. Please try again*" as in S16. If the system could spot some of the most common properties that are not covered by the database but users often ask about, for example "Weight" and "Flight-speed" as in U15 "*How much does it weigh?*", it could give more informative error messages and explain to the user that it cannot answer questions about these topics, as is done in S8.

The BIRDQUEST Ontology

A challenge when constructing a shared domain ontology lies in capturing and including two different conceptualisations of the domain, the one present in the information sources and the one that users have. The shared ontology for the BIRDQUEST application was developed based on the analysis of two different types of empirical material, a bird encyclopaedia and a question corpus. For more details see Flycht-Eriksson (2003).

The organisation and structure of the encyclopaedia were taken as a starting point for identification of ontology concepts. It used the K H Voous system for dividing birds into orders, families and species. For each of the categories information about certain properties were presented, most of which were species-specific. The book was manually analysed, to identify the objects, properties and relations relevant for the purpose of information extraction, some of which are presented in Figure 4.

The analysis of the questions in the corpus revealed that the users' view of the domain in most cases correspond to the one found in the encyclopaedia, but a small number of new categories and properties were identified. These new concepts were of three types:

- Users sometimes utilised other ways of categorising birds from the biologically oriented taxonomy in the reference book, talking about "Spring birds", "Small birds", "Migratory birds", and "Birds of prey", etc.
- In many cases the properties of the birds were more general than the terms used in the book, for example questions about size which includes both wingspan and length.
- A number of properties were not present in the bird encyclopaedia but closely related to them, such as weight.

From the analysis of the encyclopaedia a conceptualisation underlying the structure and presentation of information that were to be extracted by the Information Processing Component was constructed. The result was a system-oriented domain ontology representing experts' (the book authors) view of the domain. The non-expert view of the domain useful for dialogue interaction as provided by the question corpus were then integrated in the following manner:

- By allowing multiple inheritance new links between existing categories and new categories were added. Note, for example, how the category "Small bird" is introduced and that "Finches" is multiple linked to both this category and "Families" in Figure 4.
- In a similar manner the vague properties were introduced and linked to the existing properties. Figure 4 illustrates how two new levels are introduced, "Wingspan" and "Length" are sub-properties of "Size", which in turn is a sub-property of "Appearance".

The ontological knowledge is represented in two different knowledge bases, one containing the concepts and their taxonomical relations, and one holding the facts, i.e. object instances and their taxonomical relations. The factual part is generated semi-automatically by extraction of instances from the bird encyclopaedia. Each concept has a unique name, a definition in natural language, and a tag stating if it was system or user derived, i.e. if it came from the bird encyclopaedia conceptualisation or the question corpus. Properties and Relations also had domain and range restrictions, which state what type of objects they are applicable to. There are also cardinality restriction for the properties and relations.

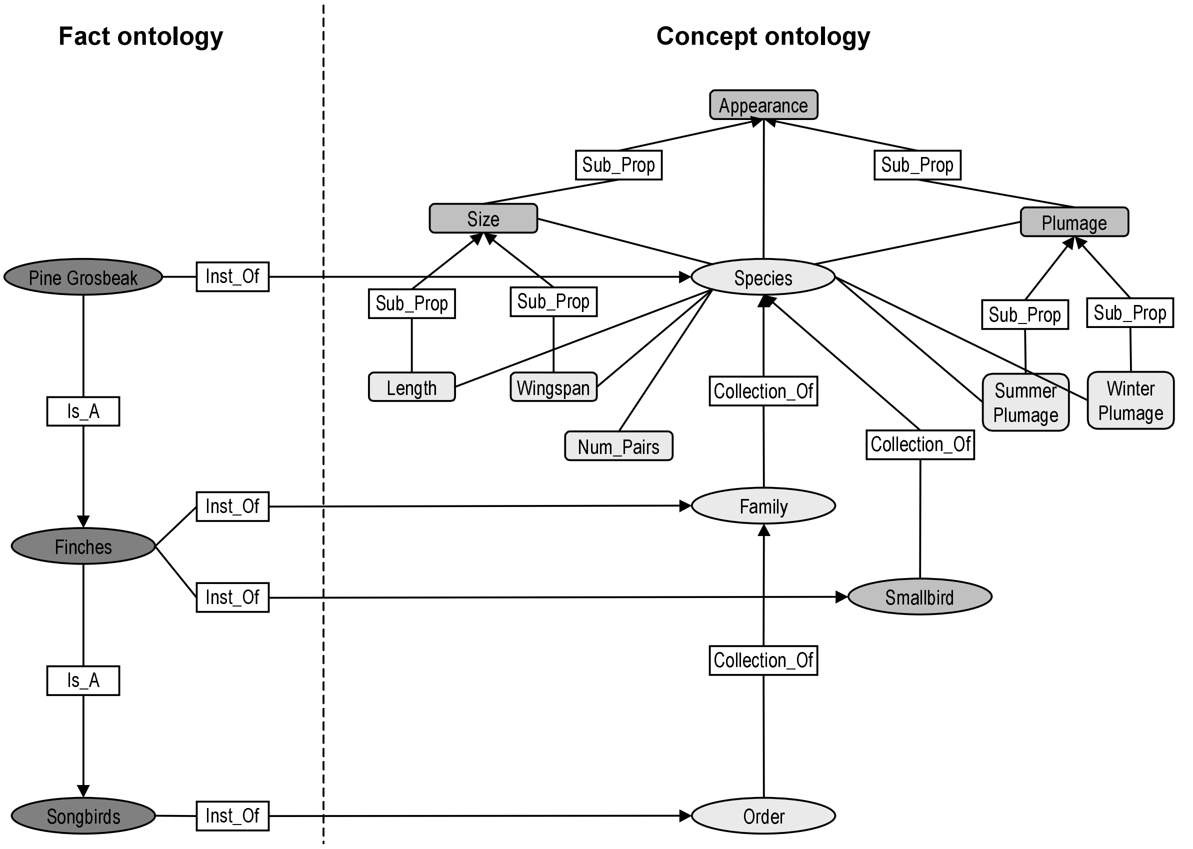


Figure 4. Part of the BIRDQUEST ontology. The figure shows some of the objects and properties and how these are related in the ontology.

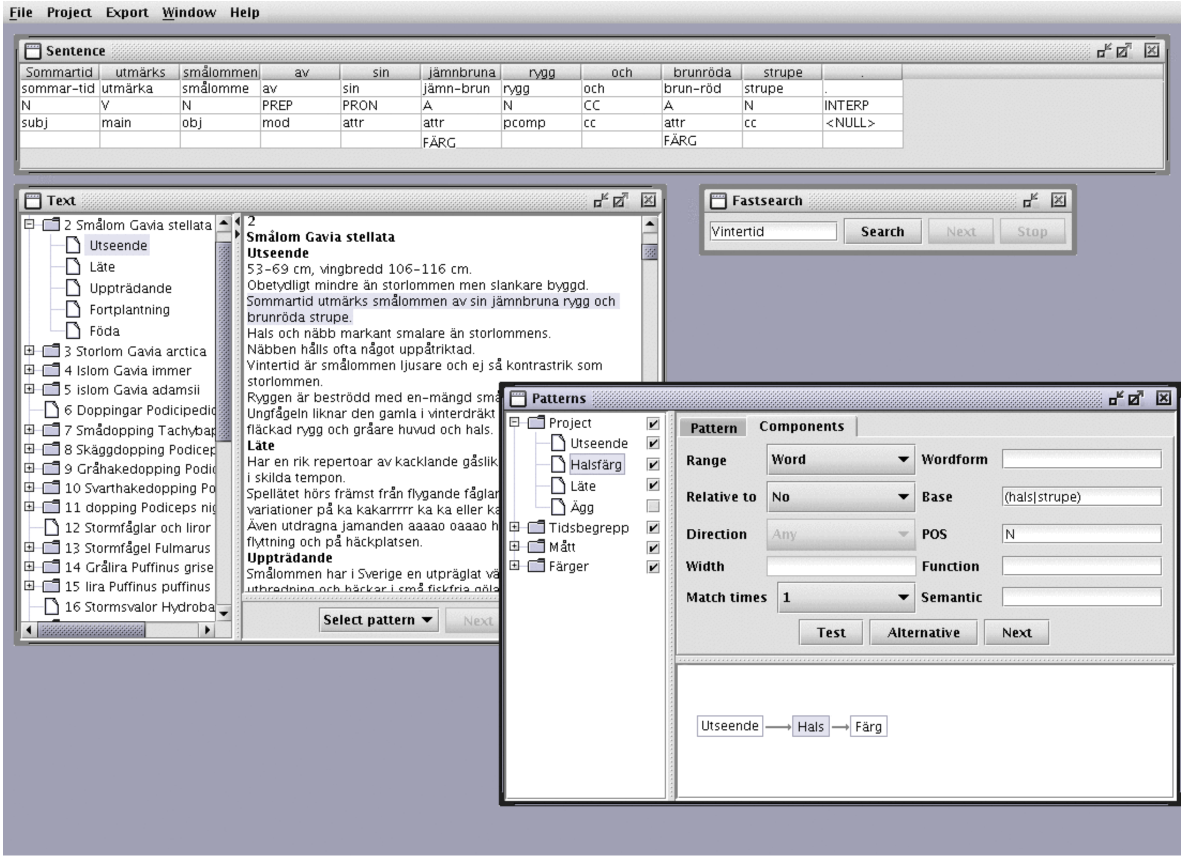
Information Extraction – use and construction of ontology

The task for information extraction is, as stated earlier, to transform the information inherent in text to a structured, semantically enriched representation that can be used as an information source for various applications, such as a dialogue system. While the first stages of IE rely mainly on linguistic knowledge, the semantic analysis, template-based extraction, and merging of partial results can be based on domain knowledge from an ontology. For example, in the information extraction system LaSIE (Gazauskas et al. 1997) the ontology organises the entities of the world relevant for the information extraction task. It provides presuppositions and support for co-reference resolution and inferences during the template-based extraction, and supports the construction of a template by merging of extracted information pieces.

An ontology can also contain procedural attachments for the objects, properties and relations, which can be used for information extraction. For example, in the information extraction system presented by Wee et al (1999) demon attributes of three different types, TO-FILL, NORMALIZE and WHEN-FILLED, are used during different stages of template filling. Ontological knowledge of domain and range restrictions on properties and relations, and their cardinality constraints can also be used to help the information extraction populate a database (Embley et al. 1998).

In our approach information extraction is mainly user-driven through a tool called I*PEX (see Figure 5) that lets the user construct pattern extraction rules in an interactive and incremental fashion in a way similar to Yangarber and Grishman (1997). Patterns can be built and tested in the semantic analysis stage on the document source that has undergone the first stages of IE (namely pre-processing, morphological and syntactic analysis). Within I*PEX the user can construct patterns of different complexity, starting from basic named-entity patterns that identifies atomic concepts such as instances of colours and names. The basic concepts can then be used to build more complex patterns that for example couple a body part of a bird to a specific colour or a geographical name to a habitat for a specific bird. All patterns are stored and can be reused later in the same IE project or exported to and re-used in other tasks. The user can perform different kinds of searches in the document base and test the applicability of the rules during the specification phase. When all necessary patterns have been constructed the user chooses the rules to be applied on the document base and the tool will update the structured information base accordingly.

However, users may find that linguistic items do not have semantic information in the lexicon (i.e., they are not connected to the part of the ontology that holds the domain facts). If this happens the user can update the lexicon, and if necessary, add or revise the domain ontology. In the current version, the revision capabilities are not fully implemented, but implementation is under way.



Sommartid	utmärks	smålommen	av	sin	jämnbruna	rygg	och	brunröda	strupe	.
sommar-tid	utmärka	smålomme	av	sin	jämn-brun	rygg	och	brun-röd	strupe	.
N	V	N	PREP	PRON	A	N	CC	A	N	INTERP
subj	main	obj	mod	attr	attr	pcomp	cc	attr	cc	<NULL>
					FÄRG			FÄRG		

Fastsearch: Vintertid

2
Smålom Gavia stellata
Utseende
53–69 cm, vingbredd 106–116 cm.
Obetydligt mindre än storlommen men slankare byggd.
Sommartid utmärks smålommen av sin jämnbruna rygg och brunröda strupe.
Hals och näbb markant smalare än storlommens.
Näbben hålls ofta något uppåtriktad.
Vintertid är smålommen ljusare och ej så kontrastrik som storlommen.
Ryggen är beströdd med en-mängd små
Ungfågeln liknar den gamla i vinterdräkt
fläckad rygg och gråare huvud och hals.
Läte
Har en rik repertoar av kacklande gåslik
i skilda tempon.
Spellätet hörs främst från flygande fåglar
variationer på ka kakarrrrr ka ka eller ka
Även utdragna jamanden aaaao oaaao h
flyttning och på häckplatsen.
Uppträdande
Smålommen har i Sverige en utpräglat vä

Range: Word
Relative to: No
Direction: Any
Width:
Match times: 1
Wordform:
Base: (hals|strupe)
POS: N
Function:
Semantic:

Figure 5. A screen-shot of the user-interface for the Information Extraction Rules Builder, I*PEX. In the Text window the document source is shown, in the Sentence window (top) a lexical and syntactic analysis of the active sentence is shown. In the Pattern window patterns can be built using multiple description levels, such as word forms, base forms, parts-of-speech, syntactic function and semantic type.

As discussed above the use of information extraction to fill a database can result in information gaps due to problems with extraction of certain types of information or lack of information in the text sources. The problem is of course to identify such gaps in the extracted knowledge bases. However, by making a completeness check in a domain such as the bird encyclopaedia, it is relatively easy to get hold of possible gaps. To determine whether lack of information stems from poor IE or from the fact that some information is not present in the source document is crucial and will have consequences for the dialogue interaction as will be shown in later sections.

Question Interpretation

In Q&A systems, questions are typically processed in a number of different steps, depending on the needs of the different components in the system. For instance, the content words of the question can be extracted and treated as a bag-of-words in order to pose a search query to a search engine, while a detailed analysis of the structure and semantics of the question is performed to find the answer within the retrieved documents (Harabagiu et al. 2000, Elworthy 2000). In dialogue systems, the interpretation of questions and requests is often driven semantically and includes identifying the relevant portions of an utterance and assigning them to slots in a semantic frame (McTear 2002). For example, spotting a city name and assigning it as a departure or arrival location in a travel information system is a typical frame-filling task.

We have constructed a generic module for question analysis that can be used as a component in question answering systems and dialogue systems. Figure 6 shows the architecture of the module and the knowledge sources it uses. Three types of information are extracted: the question type, the expected answer type, and the relevant question content. The question type discerns between yes/no, factual, explanation, and definition questions. Examples of different answer types are boolean, number, string, or an ontological object like “Spatial_Object” or “Species”. For the latter type, the expected answer type also contains information about cardinality. Regarding question content, details are given below. The content modelling can be seen as an elaboration of Jönsson (1997).

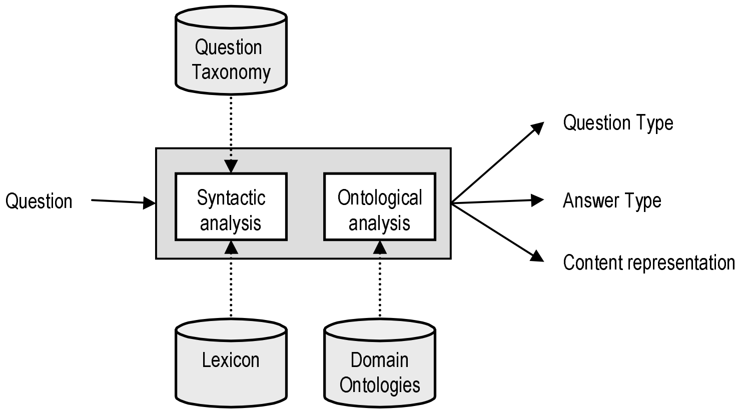


Figure 6. Question Analysis module. The question analysis module takes a question or information request as input and tries to determine the question type, answer type and content.

In the following section we present four examples drawn from the dialogue in Figure 3, and in detail explain how the question analysis operates, and how the ontological analysis contributes to the creation of a knowledge-rich and unambiguous representation of question contents with explicit semantics. Knowledge of domain and range restriction for properties and relations are utilised to filter out erroneous links between objects, values, properties and relations, and to add missing objects or values.

The figures in this section have two parts. The left-hand side presents the output from the parser used for syntactical analysis, a commercial functional dependency grammar (FDG) developed by Connexor (Tapanainen and Järvinen 1997). The right hand side presents the output from the question analysis module as a whole where black text denote information produced by the syntactical analysis, grey text information added by the ontological analysis, and strike-through text depicts information retracted by the ontological analysis.

```
Fel!Output from the FDG parser                               Output from the question analysis
1  Hur   hur   man:>2    %AH ADV                             4:     O:     #Family:$Waterfowl
2  ser   se    main:>0   %MV V PRES                          2, 5:  P:     #Appearance
3  en    en    det:>4    %>N DET SG NOM
4  and   and   subj:>2   %NH N SG NOM                        Links:                {{5, 2}, {4, 2}}
5  ut    ut    advl:>2   %AH ADV                             Question type:        Fact
6  ?     ?     %INTERP INTERP QuestionMark                   Answer type:          String
```

Figure 7: Analysis of “Hur ser en and ut?” (*What does a waterfowl look like?*)

The first example, presented in Figure 7, shows how U1 in the example dialogue is interpreted. The output from the FDG parser basically consists of six different parts, some separated by tabs, and other by white spaces. The leftmost column contains numerical references to the constituents of the question interpreted. The second column contains the actual word and the third holds the word's stem. The fourth column represents the syntactic function of the constituent, as well as a reference to the constituent that it is dependent on. The last column has two different parts. The leftmost part contains the surface syntactic tag, while the rest of the column holds morphological information about the word.

The information provided from the FDG parser is then used to make a semantic analysis of the question, which can be found to the right in Figure 7. Here, the first column refers to the constituents that give rise to the semantic entities; the second contains either an O, P, or R for objects, properties, and relations respectively; the third contains the semantics proper.

First of all, the syntactic representation receives a content representation, by looking up the words in a semantic lexicon. This lexicon is basically a reflection of the ontology, i.e., they use the same basic categories to describe objects, properties and relations in the world. Here, the fourth constituent in the question ('and') is mapped to the object "Waterfowl". The second and fifth constituent is mapped to the property "Appearance". In Swedish, 'se ut' (*look like*) is a particle verb, and in the current question, the particle 'ut' is located three constituents to the right of the verb 'se'. However, since 'ut' is dependent on 'se' and no constituents are dependent on 'ut', we can conclude that this is essentially a single unit. The dependencies are also relabelled to *Links* and their direction is removed. They are kept since they are later used for ontological disambiguation. In this case the interpretation is unambiguous since it contains one property and one object where the object is of the property's domain. The ontological analysis only adds the expected answer type *String* based on the range restriction of the property.

```
Fel!Output from the FDG parser                                 Output from the question analysis
1  Var         var        loc:>2     %AH ADV                   1:     O:     #Spatial_Object:@EXIST
2  lever       leva       main:>0    %MV V PRES                2:     R:     #Distribution
3  knölsvanar  knölsvan   subj:>2    %NH N PL NOM              2:     P:     ~~#Oldest_Age~~
4  ?           ?          %INTERP INTERP QuestionMark          3:     O:     #Species:$Mute_Swan

                                                               Links:           {{1, 2}, {3, 2}}
                                                               Question type:   Fact
                                                               Answer type:     #Spatial_Object: @EXIST
```

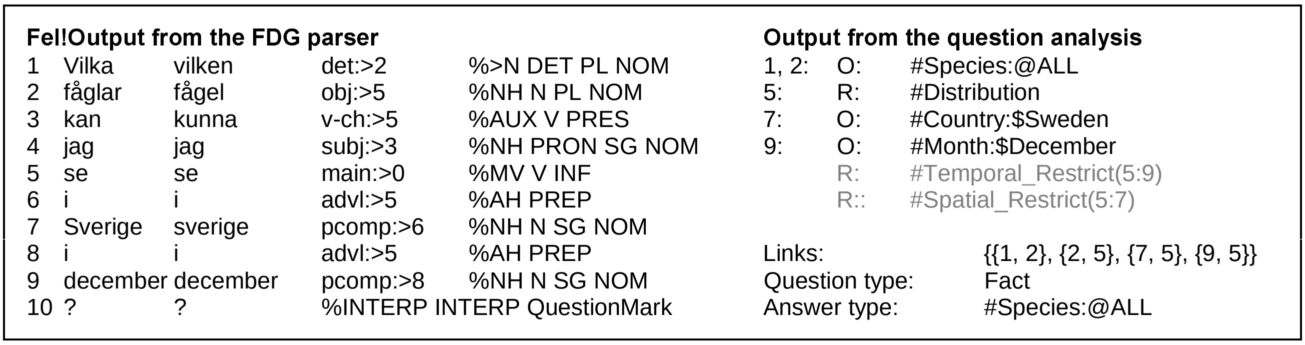
Figure 8: Analysis of "Var lever knölsvanar?" (*Where do mute swans live*?)

Figure 8 illustrates how utterance U13 in the dialogue is analysed. Here we have an ambiguity because of the word 'lever' (*live*), which can refer to both the distribution of a bird, as well as its life span. This is reflected above as the second constituent receives two different readings. Disambiguation is performed using the ontology to compare the given objects with the property and relation. Since the relation "Distribution" has domain "Species" and range "Spatial_Object" this is the preferred interpretation and the property "Oldest_Age" is ignored. However, this particular ambiguity can be avoided since 'var lever' can be treated as a single unit, i.e., a question phrase as illustrated in the example in figure 9.

```
Fel!Output from the FDG parser                                 Output from the question analysis
1  Hur         hur        ad:>2      %>A ADV                   1, 2, 3:  P:     #Oldest_Age
2  länge       länge      advl:>3    %AH ADV                   5:        O:     #Species:$Mute_Swan
3  lever       leva       main:>0    %MV V PRES
4  en          en         det:>5     %>N DET SG NOM            Links:           {{1, 2}, {2, 3}, {5, 3}}
5  knölsvan    knölsvan   subj:>3    %NH N SG NOM              Question type:   Fact
6  ?           ?          %INTERP INTERP QuestionMark          Answer type:     Number
```

Figure 9: Analysis of "Hur länge lever en knölsvan?" (*How long does a mute swan live?*)

In Figure 9 (U17 in the example dialogue) we once again find the ambiguous word 'lever'. In this case, however, no ambiguity arises since 'Hur länge lever' (H*ow long ... live?*) is treated as a single question phrase. There are a number of different question phrases, or generally multi-word units, that can be stored as single constituents in the semantic lexicon for more efficient processing.



```
Fel!Output from the FDG parser                                      Output from the question analysis
1  Vilka     vilken    det:>2     %>N DET PL NOM                   1, 2:  O:    #Species:@ALL
2  fåglar    fågel     obj:>5     %NH N PL NOM                     5:     R:    #Distribution
3  kan       kunna     v-ch:>5    %AUX V PRES                      7:     O:    #Country:$Sweden
4  jag       jag       subj:>3    %NH PRON SG NOM                  9:     O:    #Month:$December
5  se        se        main:>0    %MV V INF                               R:    #Temporal_Restrict(5:9)
6  i         i         advl:>5    %AH PREP                                R::   #Spatial_Restrict(5:7)
7  Sverige   sverige   pcomp:>6   %NH N SG NOM
8  i         i         advl:>5    %AH PREP                         Links:            {{1, 2}, {2, 5}, {7, 5}, {9, 5}}
9  december december   pcomp:>8   %NH N SG NOM                     Question type:    Fact
10 ?         ?         %INTERP INTERP QuestionMark                 Answer type:      #Species:@ALL
```

Figure 10: Analysis of "Vilka fåglar kan jag se i Sverige i december?" (*Which birds can I see in Sweden in december?*)

The last example, presented in Figure 10 (U23), shows a more complex question. Here, we conclude that the user is interested in a number of birds matching certain criteria, hence the answer type Species:@ALL. The ontological analysis use the ontology to reason about the proper interpretation of the spatial and temporal information. Temporal and spatial aspects are treated using two special types of relations, "Temporal_Restrict" and "Spatial_Restrict" that can have objects, properties or other relations as domain and "Temporal_Object or "Spatial_Object" as range, respectively. In this case they are attached to the relation "Distribution" since this has both temporal and spatial aspects, and a full interpretation of the question is derived.

Dialogue and Domain Knowledge Management

The dialogue and domain knowledge management components are based on the MALIN architecture (Jönsson, 1997), which were initially developed for traditional information sources and did not use ontological knowledge. For BIRDQUEST an ontology were introduced, which were primarily used for clarifications and database access. The MALIN Dialogue Manager classifies the discourse segments by general speech act categories, such as *question* (Q) and *answer* (A) and utilise the focal parameters to control interaction. In MALIN dialogue knowledge is represented in dialogue objects with a parameter termed Objects, which identify a set of primary referents, and the parameter Properties, which denote a complex predicate ascribed to this set. Examples of Objects in BIRDQUEST are mainly birds and Properties model information about the birds, such as appearance, number of eggs and feed The dialogue is represented in a dialogue tree structured on three levels, dialogue (D), initiative-response units (IR) and moves. An example of a dialogue tree is seen in Figure 7, which shows a part of the dialogue tree created from the dialogue in Figure 3.

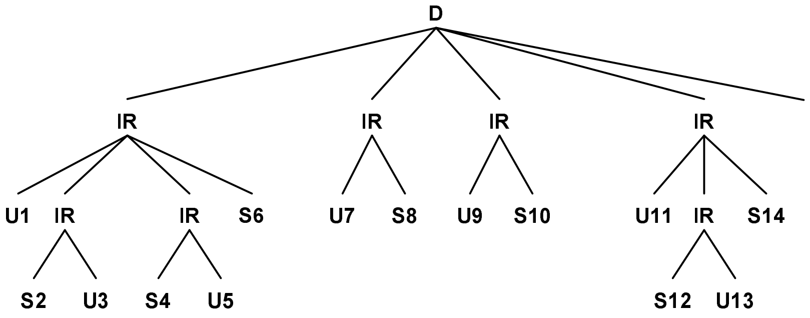


Figure 11. Dialogue tree. from the first part of the dialogue in Figure 3.

The Domain Knowledge Manager receives requests from the Dialogue Manager, and processes them further using domain knowledge collected from the domain ontology. The most common type of question is request for values of properties, such as U1 and U13, where an object and a property have been given by the user. These are processed in the following way:

1. Verify that the objects and properties in the request match. This means that the DKM accesses the ontology and checks that the domain restrictions of the properties are of the same type as the given objects. If they are not, instances of the sub- and super- classes of the given objects are collected from the ontology and new requests are created with these if they match the properties. If too many or no alternatives can be found a clarification or error message is sent to the DM.
2. Verify that the properties are suitable for database access. The DKM must access the ontology to decide if a property is vague and have to be disambiguated. In such cases the sub-properties are collected and a clarification message is sent to the DM.
3. Access the available information sources. If necessary, the request is partitioned in several object and property pairs, and each is used to collect the requested information from various data and knowledge bases. The partial results are then integrated into one answer, which is returned to the DM.

To illustrate the work of the DM and DKM consider the utterance U1 "*What do waterfowl look like*". It is an example of a request for a property value, which results in the interpretation presented in Figure 7. Since it contains both an object and a property it is considered fully specified by the Dialogue Manager, and is therefore sent to the Domain Knowledge Manager for retrieval of information.

The DKM checks if the property "Appearance" is applicable to the object of type "Family". This is not the case, as can be seen in the domain ontology in Figure 4, but since the type "Species" is, the instances of species are collected from the ontology. As there are many bird species that belong to the family "Waterfowl", the DKM decides to send a clarification message to the DM instead of creating new requests. Thus, the DM initiate a clarification subdialogue, see Figure 11, asking for a more specific object "*Choose one of the following waterfowl: tufted duck,…*" (S2). When the user has made her choice (U3) the request is once again sent to the DKM.

This time the object "Gadwall" and the property "Appearance" match. However, when checking the property in the ontology, the DKM finds that it is too vague to be used for database access since it has no immediate sub-properties that can be used for database access. Therefore a clarification message is sent to the DM, and a new clarification subdialogue takes place (S4-U5).

When the user has specified the property the request is sent back to the DKM. This time the DKM can map the property to several sub-properties suitable for access of the structured information source, utilising the ontology. The DKM translates the request into two database questions and retrieves the information. The results for "Length" and "Wingspan" is then sent to the DM.

Conclusions and future work

In this paper we have presented an architecture for ontology-driven dialogue systems that incorporate information extraction to create structured information sources from unstructured text. Examples drawn from the BIRDQUEST application developed based on the architecture has shown the use of ontologies as domain knowledge sources for such systems.

There are several directions that could be pursued in order to improve systems of this kind. One is to extend the question analysis with respect to both question and answer types as well as the ontological framework, for example, with events. A more sophisticated generator that uses ontological knowledge is also desirable. Another issue is to generalise the knowledge sources, e.g. creating a general question type taxonomy and incorporating other general ontologies like WordNet. Furthermore, using information extraction to update domain ontologies is a priority in the future.

Finally, BIRDQUEST was developed for a closed domain and future research issue concerns the move to multi-domain and then to open domain applications. It is still very much an open question to what extent the techniques for IE, shared knowledge sources and dialogue management presented in this paper can be applied for such applications

Acknowledgement

Many thanks to Frida Andén, Lars Degerstedt and Sara Norberg for interesting discussions and work on the development and implementation of the BIRDQUEST application. Thanks also goes to Andreas Lange who has been working with I*PEX. This research is financed by Vinnova, Swedish Agency for Innovation Systems

References

Burger, J., Cardie, C., Chaudhri, V., Gaizauskas, R., Harabagiu, S., Israel, D., Jacquemin, C., Lin, C. Y., Maiorano, S., Miller, S., Moldovan, D., Ogden, B., Prager, J., Rilo, E., Singhal, A., Shrihari, R., Strzalkowski, T., Voorhees, E. and Weishedel. R. Issues, tasks and program structures to roadmap research in question & answering (q&a).
URL: wwwnlpir.nist.gov/projects/duc/papers/qa.Roadmap-paper_v2.doc, 2001.

Elworthy, E. Question answering using a large NLP system. In *Proceedings of the Ninth Text REtrieval Conference (TREC-9),* Gaithersburg, Maryland, USA, 2000.

Embley, D. W., Campbell, D. M., Liddle, S. W. and Smith, R. D. Ontology-based extraction and structuring of information from data-rich unstructured documents. In *Proceedings of CIKM'98*, Washington, D.C., USA, 1998.

Flycht-Eriksson, A. Design of Ontologies for Dialogue Interaction and Information Extraction. In *Proceedings of IJCAI'03 workshop on Knowledge and reasoning in practical dialogue systems*, Acapulco, Mexico. 2003.

Flycht-Eriksson, A. and Jönsson, A. Dialogue and domain knowledge management in dialogue systems. In *Proceedings of 1st SIGdial Workshop on Discourse and Dialogue*, Hong Kong, 2000.

Flycht-Eriksson, A. and Jönsson, A. Some Empirical Findings on Dialogue Management and Domain Ontologies in Dialogue System - Implications from an Evaluation of BirdQuest. In *Proceedings of 4th SIGdial Workshop on Discourse and Dialogue*, Sapporo, Japan. 2003.

Gazauskas, R., Humphreys, K., Azzam, S. and Wilks, Y. Concepticons vs. Lexicons: An Architecture for Multilingual Information Extraction, in *Information Extraction: A Multidisciplinary Approach to an Emerging Information Technology*, volume 1299 of Lecture Notes in Articial Intelligence, chapter 3, pages 28-43. Springer,1997.

Gruber, T. R. Towards principles for the design of ontologies used for knowledge sharing. Technical report, Stanford University, Palo Alto, CA, 1993.

Harabagiu, S. Moldovan, D., Pasca, M., Mihalcea, R., Surdeanu, M., Bunescu, R., Girju, R., Rus, V.and Morarescu, P. FALCON: Boosting knowledge for answer engines. *In Proceedings of the Ninth Text REtrieval Conference (TREC-9)*, Gaithersburg, Maryland, USA, 2000.

Jönsson, A. A model for habitable and efficient dialogue management for natural language interaction. *Natural Language Engineering*, 3(2/3):103-122, 1997.

Mahesh, K. and Nirenburg, S. A Situated Ontology for Practical NLP. *Proceedings of IJCA'95 Workshop on Basic Ontological Issues in Knowledge Sharing*, Montreal, Canada, 1995.

McTear, M. F. Spoken dialogue technology: Enabling the conversational user interface*. ACM Computing Surveys*, 34(1), March 2002.

Staav, R. and Fransson, T. *Nordens fåglar*. Stockhom: Norstedt, 1991

Tapanainen, P. and Järvinen, T. A non-projective dependency parser. In *Proceedings of the 5th Conference on Applied Natural Language Processing*, pages 64-71, 1997.

Wee, L. K. A., Tong, C. and Tan, C. L. Textual information extraction in the face of information deluge. In *Proceedings of IJCAI'99 Workshop Text Mining, Foundations, Techniques and Applications*, Stockholm, Sweden, 1999.

Yangarber, R. and Grishman, R. Customization of Information Extraction Systems. In *Proceedings of the International Workshop on Lexically Driven Information Extraction*, Frascati, Italy., 1997.