published as: Serge Linckels, Stephan Repp, Naouel Karam, Christoph Meinel: The Virtual Tele-TASK Professor---Semantic Search in Recorded Lectures;
In Proceedings of the 38th technical symposium on Computer science education (SIGCSE 2007);
Covington, Kentucky, USA, March 2007. pp. 50 - 54, ISBN: 1-59593-361-1.

The Virtual Tele-TASK Professor—Semantic Search in Recorded Lectures

Serge Linckels, Stephan Repp, Naouel Karam, Christoph Meinel
Hasso-Plattner-Institut (HPI), University of Potsdam
P.O. Box 900460, D-14440 Potsdam
{linckels,repp,karam,meinel}@hpi.uni-potsdam.de

ABSTRACT

This paper describes our *e-librarian service* that understands students' complete questions in natural language and retrieves very few but pertinent *learning objects*, i.e., short multimedia documents. The system is based on three key components: the formal representation of a domain ontology, a mechanism to automatically identify learning objects out of a knowledge source, and a semantic search engine that yields only pertinent results based on the freely formulated questions in natural language.

We report on experiments about students' acceptance to enter complete questions instead of only keywords, and about the benefits of such a virtual personal teacher in an educational environment.

Categories and Subject Descriptors

H.3.7 [**Information Storage and Retrieval**]: Digital Libraries; K.3.1 [**Computers and Education**]: Computer Uses in Education—*Computer-Assisted Instruction*

General Terms

Algorithms, Management, Performance, Reliability

Keywords

Autonomous and exploratory learning, multimedia retrieval, multimedia knowledge base, speech indexing, semantic indexing

1. INTRODUCTION

This paper describes a working system that fosters autonomous and exploratory learning. It is the first time that we summarize different projects of our research (segmentation of recorded lectures, translation of natural language (NL) into a semantic query, and the study of a student oriented learning software) in order to conceive and discuss a pragmatic and useful *e-librarian service* that can be used in school and at home. The system understands the students' questions in NL, and returns only a few, but semantically relevant, results in a multimedia form.

The common part between the "user world" and the "machine world" is a shared *ontology* (section 3) to which the learning objects from the knowledge source and the users' NL questions are mapped. This allows us to have equal semantics between both worlds.

We bypass the enormous problem of content production by using existing knowledge sources. We show a mechanism to identify learning objects in a knowledge source (section 4).

Most people who are not expert users of modern information retrieval systems have difficulties in formulating their queries in a machine optimized way, e.g., by combining search terms with Boolean operators. Furthermore, they may not use the right domain expressions or make spelling errors. Therefore, the NL interface to our e-librarian service is a key part of the system. The user can freely formulate questions in NL which are then translated in a machine readable query (section 5).

Experiments confirmed our e-librarian service as an efficient e-learning tool, a kind of virtual personal teacher that can be used in school and at home (section 6). We measured a relevant improvement in the students' results that is mainly attributed to the fact that the students were more motivated by using our system, and therefore put more effort into learning and acquiring new knowledge.

2. RELATED WORK

2.1 Segmentation and Indexing

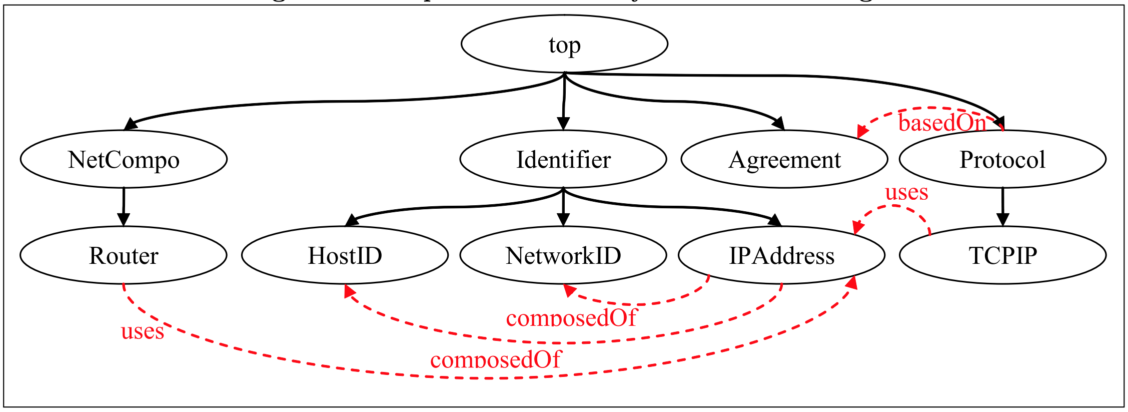
In [21] an automatic video browsing and retrieval system is presented. It is able to extract the texts from key frames, and to construct the textual indexes for the retrieval. However, the system is not adapted to lecture videos, no semantic search is possible, and the quality of the speech recognition is not mentioned.

An automatic speech recognition system to index recorded lectures is evaluated in [7]. Although the accuracy of the speech recognition software is rather low, the recognition accuracy of audio lectures is approximately 22%-60%.

A system to locate and browse lecture segments is presented in [5]. It automatically generates a time-aligned word transcript of a lecture. A browser allows students to locate and download audio segments that are relevant to their query.

A segmentation method of continuous lecture speech into

Figure 1: Sample of a taxonomy about networking.



topics is presented in [20]. It is based on the transcript obtained by spontaneous speech recognition of the audio data that is then associated with the textbook used in the lecture.

2.2 Retrieval and NLDB

START [8] is the first question-answering system available on the Web. However, the NLP is not always sound, e.g., the question "What did Jodie Foster do before she became an actress?" returns "I don't know what Jodie fostered before the actress became an actress".

AquaLog [11] is a portable question-answering system which takes queries expressed in NL and an ontology as input, and returns answers drawn from one or more knowledge bases. User questions are expressed as triples: <subject, predicate, object>. If the various translation mechanisms fail, then the user is asked for disambiguation. The system also uses an interesting learning component to adapt to the user's "jargon". Unfortunately, AquaLog currently has a very limited knowledge space. In a benchmark test dealing with 76 different questions, 37 (48.68%) were handled correctly.

The prototype PRECISE [15] uses ontology technologies to map semantically tractable NL questions to the corresponding SQL query. It was tested on several hundred questions drawn from user studies over three benchmark databases. Over 80% of the questions are semantically tractable questions, which PRECISE answered correctly, and recognized the 20% it could not handle, and requests a paraphrase. The problem of finding a mapping from the tokenization to the database requires that all tokens must be distinct; questions with unknown words are not semantically tractable and cannot be handled.

FALCON [6] is an answer engine that handles questions in NL. When the question concept indicating the answer type is identified, it is mapped into an answer taxonomy. The top categories are connected to several word classes from WordNet. Also, FALCON gives a cached answer if a similar question has already been asked before; a similarity measure is calculated to see if the given question is a reformulation of a previous one. In TREC-9, FALCON generated a score of 58% for short answers and 76% for long answers, which was actually the best score.

LASSO [13] relies on a combination of syntactic and semantic techniques, and lightweight abductive inference to find answers. The search for the answer is based on a form of indexing called paragraph indexing. The extraction and evaluation of the answer correctness is based on empirical abduction. A score of 55.5% for short answers and 64.5% for long answers was achieved in TREC-8.

3. ONTOLOGICAL APPROACH

It has been realized that digital libraries could benefit from having its content understandable and available in a machine processable form, and it is widely agreed that ontologies will play a key role in providing much enabling infrastructure to achieve this goal. A fundamental part of our system is a common domain ontology, which is used on the one hand for the extraction and the description of learning objects in a knowledge source (section 4), and on the other hand for the translation of the users' NL questions into a formal terminology (section 5). An existing ontology can be used, or one can build its own ontology that is optimized for the knowledge sources. In this section, we describe how we built our own ontology about networking.

3.1 Building an Ontology

First, we created a list of semantically relevant words that were used in the knowledge source. Secondly, these words were organized in a hierarchical way to form a taxonomy of concepts (figure 1). Finally, the ontology was formalized in a common language, a language that a human individual can understand and that can be processed by a machine; we use *Description Logics*. Such a representation is called a terminology or TBox (figure 2). The terminology can be serialized as OWL-DL (*Semantic Web Ontology Language*).

3.2 Description Logics

Description logics (DL) [2] are a family of knowledge representation formalisms that allow to represent the knowledge of an application domain in a structured way and to reason about this knowledge. In DL, the conceptual knowledge of an application domain is represented in terms of *concepts* (unary predicates) such as IPAddress, and *roles* (binary predicates) such as composedOf. Concepts denote sets of individuals and roles denote binary relations between individuals. Complex descriptions are built inductively using concept constructors which rely on basic concept and role names. The different DL languages distinguish themselves by the kinds of constructs they allow. Examples of concept

Figure 2: Sample of a terminology about networking.

$$\begin{array}{rcl}
\mathsf{Protocol} & \sqsubseteq & \exists \mathsf{basedOn.Agreement} \\
\mathsf{TCPIP} & \sqsubseteq & \mathsf{Protocol} \sqcap \exists \mathsf{uses.IPAddress} \\
\mathsf{Router} & \sqsubseteq & \mathsf{NetComponent} \sqcap \exists \mathsf{has.IPAddress} \\
\mathsf{HostID} & \sqsubseteq & \mathsf{Identifier} \\
\mathsf{NetworkID} & \sqsubseteq & \mathsf{Identifier} \\
\mathsf{AddressClass} & \sqsubseteq & \mathsf{Identifier} \\
\mathsf{IPAddress} & \sqsubseteq & \mathsf{Identifier} \sqcap \exists \mathsf{composedOf.HostID} \\
 & & \sqcap \exists \mathsf{composedOf.NetworkID} \\
 & & \sqcap \exists \mathsf{partOf.AddressClass}
\end{array}$$

Figure 3: Examples of learning objects.

$$\begin{array}{l}
\mathsf{LO}_1 \equiv \mathsf{IPAddress} \\
\mathsf{LO}_2 \equiv \mathsf{TCPIP} \sqcap \exists \mathsf{uses.IPAddress} \\
\mathsf{LO}_3 \equiv \mathsf{IPAddress} \sqcap \exists \mathsf{composedOf.HostID} \\
\mathsf{LO}_4 \equiv \mathsf{IPAddress} \sqcap \exists \mathsf{composedOf.NetworkID} \\
\mathsf{LO}_5 \equiv \mathsf{TCPIP}
\end{array}$$

constructs are the following:

- top-concept $\top$ and bottom-concept $\bot$ denoting all the individuals in the domain and the empty set respectively,
- conjunction $\sqcap$,
- existential restriction $\exists r.C$ e.g., $\mathsf{IPAddress} \sqcap \exists \mathsf{composedOf.HostID}$ says that an IP address is composed of a host ID.

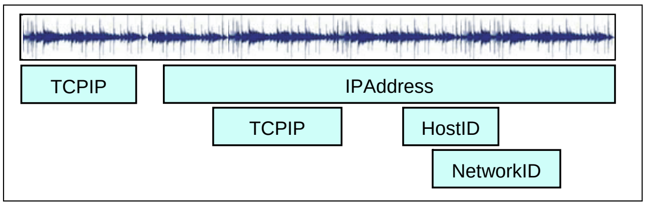
Concept descriptions are used to specify terminologies that define the intentional knowledge of an application domain. Terminologies are composed of *inclusion assertions* and *definitions*. The first impose necessary conditions for an individual to belong to a concept, e.g., to impose that a router is a network component that uses at least one IP address, one can use the inclusion assertion: $\mathsf{Router} \sqsubseteq \mathsf{NetComp} \sqcap \exists \mathsf{uses.IPAddress}$. Definitions allow us to give meaningful names to concept descriptions such as $\mathsf{LO}_1 \equiv \mathsf{IPAdress} \sqcap \exists \mathsf{composedOf.HostID}$.

DL systems provide various reasoning services; one of the most important is the computation of the *subsumption* relation standing for the more specific relations among concepts and forming the basis of the concept hierarchy. Formally, a concept description D subsumes a description C (noted $C \sqsubseteq D$) if every interpretation assigns to C a set of individuals included in the one assigned to D. Subsumption over a TBox is noted $\mathcal{T} \models C \sqsubseteq D$ and it occurs if the subsumption holds for every model of $\mathcal{T}$.

4. LEARNING OBJECTS

The creation of new learning material is an awkward and time consuming task if it is done from sketch. Therefore, we decided to use the online tele-TASK archive (`http://www.tele-task.de`) that contains hundreds of university lectures. We selected the lecture series about networking in computer-science, which is a set of 30 units with a total of 38 hours of lectures.

Figure 4: Example of 4 identified chains inside a lecture part about IP addressing.



Because we want the search engine to yield more precise and concise results than a complete lecture of 90 minutes, we created a mechanism to automatically identify short learning objects inside the lectures.

The segmentation of recorded lectures into smaller items (*chains*) is based on the imperfect transcripts of the audio data. Our algorithm [17, 18] works in four steps. Firstly, a speech recognition software converts the audio stream into a transcript with an accuracy of 65 - 80%. Secondly, the extracted words are transformed into their canonical form using a part-of-speech (POS) tagger. Thirdly, all stop words—words that are not important to understand the sense of a sentence, e.g., articles—are automatically removed. Finally, cohesive areas—segments (chains) of accumulated appearance of an equal word—in the transcript are identified.

Definition 1 (Chain). *A chain c_w is a sequence in a data stream, identified by its start and end time, which represents the longest repetition of the same word w, and where the gap between two identical words w_i and w_{i+1} is not greater than a given threshold.*

A chain is always about one specific word. The semantics are given to a chain by mapping it to an ontology and by eventually enriching it with further assertions, e.g., role restrictions; we call the result a *learning object*. At the current state of the algorithm, the enrichment must be done by hand (figure 3). A learning object is an entity about a precise concept that can be processed by the semantic search engine. The mapping function uses a list of synonyms to associate similar words (in their canonical form) to a concept in the ontology, e.g., expressions like "IP-address" and "IP addr." refer to the same concept IPAddress in the ontology. This simple solution gives sufficiently good results. 88% of the chains identified in this way were exact matches—with a tolerance of 30 seconds—of the topics in the lecture.

Chains overlap when the speaker uses different relevant words several time during the same time interval (figure 4). The overlapping is detected by comparing the start and end time of the different chains. The resulting granularity of the segmentation depends on the allowed gap (threshold) between two identical words. For illustration, we chose a random sequence in a lecture that covered 7 slides about IP addressing with a total duration of 13 minutes. The system identified 22 chains in this sequence with a threshold set to 180 seconds. Each chain had an average length of 2 minutes, and contained in average 3 times the chain specific word (chain relevancy).

The length (duration) of a chain depends on the number of occurrences of the chain specific word. The higher

Figure 5: Examples of NL questions and their semantic interpretation.

Question	DL definition	Yielded learning object
What is an IP-address?	$\mathsf{IPAddress}$	LO_1
What is an IP-address composed of?	$\mathsf{IPAddress} \sqcap \exists \mathsf{composedOf}.\top$	$\mathsf{LO}_3+\mathsf{LO}_4$
Where are IP-address' used?	$\exists \mathsf{uses}.\mathsf{IPAddress}$	LO_2
Is the host-id part of an IP-address?	$\mathsf{IPAddress} \sqcap \exists \mathsf{composedOf}.\mathsf{HostID}$	LO_3

the frequency of that word inside a chain—that is the more the speaker used the same word during a relatively short interval—the greater is the semantic relevance of the chain.

5. RETRIEVAL

The representation of context-independent meaning is called the *logical form*, and the process of mapping a sentence to its logical form is called *semantic interpretation* [1]. The logical form is expressed in a certain knowledge representation language; we use *Description Logics* (DL). First of all, DL have the advantage that they come with well defined semantics and optimized algorithms. Furthermore, the link between DL and NL has already been established [4, 19]. Finally, translating the user question into DL allows direct reasoning over the learning objects because both—the user query in its logical form, and the knowledge base described with OWL-DL—respect the same common language and terminology.

In a former version of the system, the translation algorithm [10] worked by mapping each word to a concept in the ontology, and by creating a conjunctive query of the identified concepts and roles. The major weakness of this algorithm is that the translation is performed without considering the possible answers available in the knowledge base.

The semantic interpretation is improved by searching the best possible definition available in the terminology (section 3) to which the complete user question can be mapped. The new algorithm performs the semantic interpretation of a NL user question in three steps. Firstly, there is the linguistic pre-processing, which transforms a stream of symbols in one or more linguistic clauses depending on the complexity of the user's question. Secondly, each word from the linguistic clauses is mapped to a concept or a role in the ontology by considering linguistic information. Normally, nouns, question words (*w-words*) and proper names are mapped to concepts, and verbs, adjectives and adverbs are mapped to roles. In general, a clause has up to two concepts and not more than one role. Thirdly, among the various available axioms in the terminology the most accurate is selected in order to be the best possible match for the identified concepts and roles of the user question. Missing items in a matching definition are replaced by wildcards, i.e., top concept ($\top$). In the current state of the algorithm, number restrictions and negations are not considered.

Once the formal representation of the user question is generated, a standard DL reasoner is used to infer over the knowledge base, and to retrieve learning objects. The returned results are logical consequences of the inference rather than of keyword matchings. Various examples are shown in figure 5.

6. DISCUSSION

We made two different experiments with students to collect empirical data about the quality of our e-librarian service.

6.1 Description of the Experiments

In a first experiment—detailed in [16]—60 students (aged 17 - 22) compared our semantic search engine to a keyword search engine. Both tools had the same interface and accessed the same knowledge source (about the topic of computer history). We gave the students the opportunity to test both search engines during 20 minutes and to state their impressions.

In a second experiment—detailed in [9]—we had a different approach. A class of 22 students (aged 12 - 14) used the semantic search engine—with a knowledge base about fractions in mathematics—during 5 weeks for 6 hours a week. The teacher used the first two lessons to introduce the tool and its semantic search engine, explained how they had to use it, and why it is so important to always enter complete questions.

6.2 Entering Complete Sentences

In the first experiment, 22% of the students answered that they would have no problem entering complete questions instead of keywords, 69% preferred to enter complete questions instead of keywords if this yielded better results, and 8% disliked this option.

The second experiment showed that students had problems with entering complete questions at the beginning, but it became generally accepted after a few lessons. We witnessed that most of the students entered questions very quickly. It seemed that they had a lot of experience typing on a computer (possibly by chatting on the Internet). At the end of the experiment, no student stated that this was awkward, 7 students (31.8%) answered that they accept having to enter a complete question but that they did not like it, and 15 students (68.2%) answered that this was no problem at all.

6.3 Impact on School Results

We learned from the first experiment that users need training and domain knowledge before they are able to successfully use search engines. Secondly, when using search engines, the students are relatively free to act as they like, which is quite unusual for most. As confirmed by [12, 3, 14], users need guidance in how to formulate effective queries, and in how to use a computer tool efficiently.

The second experiment covered a period of 5 weeks of intensive use of our e-librarian service. There was no classical mathematics lesson—i.e., teacher centered lesson—where the teacher gave explanations, but the students had to learn in an autonomous and exploratory way. They had to ask questions to the e-librarian service just the way they would if there was a human teacher.

The students did not perceive the e-librarian service as a game, but as a helpful educational tool, a kind of virtual personal teacher. We measured an overall improvement of 5% in the students' results on fractions, compared to their past results on geometry. 11 students had better results in fractions than in geometry. 9 of them progressed very much (at least 6 marks with a maximum of 60 marks for a test). There is even one student whose progression is 21 marks. 8 students regressed, 3 of them very much (at least 6 marks). 3 students stayed constant. In total the 11 students progressed by 111 marks against the 8 students that regressed by 50 marks.

One of the main reasons for these positive results is that the students were more motivated and therefore willing to put more effort into learning and acquiring new knowledge. The students also stated that the tool explained better and that they understood more easily. This is certainly due to positive effect of using multimedia video sequences to explain a topic. Finally, students said that they always found the right answers to their questions quickly, which confirms the performance and the reliability of the semantic search engine.

7. CONCLUSION

This paper summarized several research projects around our *e-librarian service* that understands the users' complete questions in NL and retrieves very few but pertinent *learning objects*. The system is based on three key components: the formal representation of a domain ontology, a mechanism to automatically identify learning objects in a knowledge source, and a semantic search engine that yields only pertinent results based on the users NL question.

The advantage of such a modular architecture is that each module can be improved and evaluated separately. Currently, we are working on the improvement of the translation from NL into a DL definition to get a higher expressivity, especially concerning negations, number restrictions, and quantifiers. We are also working on the segmentation of the lecture videos in order to automatically generate a complete semantic table of contents for a given lecture.

8. REFERENCES

[1] J. Allen. *Natural Language Understanding.* Addison Wesley, 1994.

[2] F. Baader, D. Calvanese, D. L. McGuinness, D. Nardi, and P. F. Patel-Schneider, editors. *The Description Logic Handbook: Theory, Implementation, and Applications.* Cambridge University Press, 2003.

[3] R. Fidel, R. K. Davies, M. H. Douglass, J. K. Holder, C. J. Hopkins, E. J. Kushner, B. K. Miyagishima, and C. D. Toney. A visit to the information mall: Web searching behavior of high school students. *Journal of the American Society for Information Science*, 50(1):24–37, 1999.

[4] E. Franconi. *The Description Logic Handbook: Theory, Implementation, and Applications*, chapter Natural Language Processing, pages 450–461. Cambridge University Press, 2003.

[5] J. R. Glass, T. J. Hazen, D. S. Cyphers, K. Schutte, and A. Park. The MIT spoken lecture processing project. In *HLT/EMNLP Interactive Demonstrations*, pages 28–29, 2005.

[6] S. M. Harabagiu, D. I. Moldovan, M. Pasca, R. Mihalcea, M. Surdeanu, R. C. Bunescu, R. Girju, V. Rus, and P. Morarescu. Falcon: Boosting knowledge for answer engines. In *Text Retrieval Conference (TREC)*, 2000.

[7] W. Hürst, T. Kreuzer, and M. Wiesenhütter. A qualitative study towards using large vocabulary automatic speech recognition to index recorded presentations for search and access over the web. In *IADIS WWW/Internet (ICWI)*, pages 135–143, 2002.

[8] B. Katz. Annotating the world wide web using natural language. In *Computer Assisted Information Searching on the Internet (RIO)*, 1997.

[9] S. Linckels, C. Dording, and C. Meinel. Better results in mathematics lessons with a virtual personal teacher. In *ACM SIGUCCS*, pages 201–209, 2006.

[10] S. Linckels and C. Meinel. Resolving ambiguities in the semantic interpretation of natural language questions. In *Intelligent Data Engineering and Automated Learning (IDEAL)*, volume 4224 of *Lecture Notes in Computer Science*, pages 612–619, 2006.

[11] V. Lopez, M. Pasin, and E. Motta. Aqualog: An ontology-portable question answering system for the semantic web. In *European Semantic Web Conference (ESWC)*, pages 546–562, 2005.

[12] P. Martin. *Web Intelligence*, chapter Knowledge Representation, Sharing and Retrieval on the Web, pages 263–297. Springer-Verlag, 2003.

[13] D. I. Moldovan, S. M. Harabagiu, M. Pasca, R. Mihalcea, R. Goodrum, R. Girju, and V. Rus. Lasso: A tool for surfing the answer net. In *Text Retrieval Conference (TREC)*, 1999.

[14] R. Navarro-Prieto, M. Scaife, and Y. Rogers. Cognitive strategies in web searching. In *Conference on Human Factors & the Web*, 1999.

[15] A.-M. Popescu, O. Etzioni, and H. A. Kautz. Towards a theory of natural language interfaces to databases. In *Intelligent User Interfaces*, pages 149–157, 2003.

[16] M. Reichert, S. Linckels, and C. Meinel. Student's perception of a semantic search engine. In *Cognition and Exploratory Learning in Digital Age (CELDA)*, pages 139–147, 2005.

[17] S. Repp and C. Meinel. Segmenting of recorded lecture videos - the algorithm voiceseg. In *Signal Processing and Multimedia Applications (SIGMAP)*, pages 317–322, 2006.

[18] S. Repp and C. Meinel. Semantic indexing for recorded educational lecture videos. In *Pervasive Computing and Communications Workshops (PerCom)*, pages 240–245, 2006.

[19] R. A. Schmidt. Terminological representation, natural language & relation algebra. In *German AI Conference (GWAI)*, volume 671 of *Lecture Notes in Artificial Intelligence*, pages 357–371, 1993.

[20] N. Yamamoto, O. Jun, and Y. Ariki. Topic segmentation and retrieval system for lecture videos based on spontaneous speech recognition. In *Eurospeech*, pages 961–964, 2003.

[21] Y. Zhu and D. Zhou. Video browsing and retrieval based on multimodal integration. In *IEEE/WIC International Conference on Web Intelligence (WI)*, pages 650–653, 2003.