ISNLP on AI for Education
Exploratory action 2019
Exploratory experiments combining
Information Systems and
Natural Language Processing

Objectives

We proposed experiments to enrich texts (in various formats) that serve as pedagogical support.

This comes in two major ways:

- extraction of major parts from full text, automatic or interactive.
- completion of lacunary texts (eg, slides or program comments)

To help learners through enriched pedagogical documents/txts in comp. sc., languages, NLP.

Major terms are to be connected with referring or enlarged information (terminology, definition, formula, symbols, etc.).

The targeted complexity may go from a simple glossary to a rich semantic network and may include interactive technology.

Learning context: between courses, before exams, self-learning.

Texts/concepts:
- enrichments, avoid confusions, self-correction via semantics & new exercises, precise terminology, explanations.

These features are not topic-specific (science, LVL, ...)

Informal to Informal

Workflow on Terminology

SemL2

The new workflow goes from tests or pdf to a selected terminology.

It has been tested on the L2 and L3 level and can be run on a terminal or as a web interface. It can deal with multiple expressions (such as "Diagramme de Venn") and symbols (such as set operations) in a pdf file.

This workflow prepares the links to definitions and concepts for relevant terms/symbols.

Informal to Formal

Evolution of concepts in term occurrence graph

SemL3

We worked on enriching learning resources to analyse and to understand the evolution of involved concepts: in particular, in scientific articles, from a large corpus of scientific publication (arXiv) workshop LEOS2019.

This work is based on the analysis of the term co-occurrence graph, where we applied spectral techniques.

In a similar way, we could extract terms from two widely used textbooks (Deep learning and Android programming) in PDF format.

Annotated Corpus for Modeling

SemL3

With the aim of assisting the engineer to design a system, the evaluation and validation of a processing chain around system design remains a challenge.

We thus defined study cases and started an annotated corpus to prepare a semantic extraction from specifications.

Experiments overview

The experiments considered students in computer science, in particular their learning of formal topics and of query languages.

Pitfalls examples: definitions-conventions and variability
- partial vs total (for a function, for an order)
- symmetric vs not symmetric (for a relation, for a matrix)
- r = 0 vs r = 

For Course Companions in computer science

Questionary analysis by LOUSTIC

<table>
<thead>
<tr>
<th>Attribute des utilisateurs</th>
<th>Description des utilisateurs</th>
<th>Rôle des utilisateurs</th>
<th>Action de l'utilisateur</th>
<th>Action de l'utilisateur</th>
<th>Action de l'utilisateur</th>
</tr>
</thead>
<tbody>
<tr>
<td>Utilisation, facilité d'utilisation</td>
<td>Facilité d'utilisation</td>
<td>Facilité d'utilisation</td>
<td>Utilisation, facilité d'utilisation</td>
<td>Utilisation, facilité d'utilisation</td>
<td>Utilisation, facilité d'utilisation</td>
</tr>
<tr>
<td>Accessibilité et souplesse d'utilisation</td>
<td>Accessibilité et souplesse d'utilisation</td>
<td>Accessibilité et souplesse d'utilisation</td>
<td>Accessibilité et souplesse d'utilisation</td>
<td>Accessibilité et souplesse d'utilisation</td>
<td>Accessibilité et souplesse d'utilisation</td>
</tr>
<tr>
<td>Possibilité d'évaluer l'outil (évaluation continue)</td>
<td>Possibilité d'évaluer l'outil (évaluation continue)</td>
<td>Possibilité d'évaluer l'outil (évaluation continue)</td>
<td>Possibilité d'évaluer l'outil (évaluation continue)</td>
<td>Possibilité d'évaluer l'outil (évaluation continue)</td>
<td>Possibilité d'évaluer l'outil (évaluation continue)</td>
</tr>
</tbody>
</table>

Improving a tool for learning modal logic

Hirinckx's Word can be used for learning modal (dynamic epistemic) logic with examples. Epistemic Situations are described by Kripke models. For card games the number of possible worlds is gigantic. This is why symbolic models based on ROGs are used. Our internship has proposed a new software architecture, for better GPU performances. We used the asynchronous programming paradigm.

Learning formal grammars and the writing of syntax analyzer

The created educational software proposes to study formal grammars and the writing of syntax analyser by hand. From a given formal grammar (generated randomly), the learner must answer several questions that lead to a final program (written in C) that can analyse this grammar.

The questionnaire shows a demand for more training exercises.