The Readability Checker Delite

Technical Report

Tim vor der Brück, Hermann Helbig, Johannes Leveling
{tim.vorderbrueck, hermann.helbig, johannes.leveling}@fernuni-hagen.de
FernUniversität in Hagen
Intelligente Informations- und Kommunikationssysteme
Informatikzentrum
Universitätsstraße 1
58084 Hagen

Keywords: Readability, Simple Language, Natural Language Analysis, Machine Learning, Robust Regression, Iterative Linear Regression, Language Model

This report describes the DeLite readability checker which automatically assesses the linguistic accessibility of Web documents. The system computes readability scores for an arbitrary German text and highlights those parts of the text causing difficulties with regard to readability. The highlighting is done at different linguistic levels, beginning with surface effects closely connected to morphology (like complex words) down to deep semantic phenomena (like semantic ambiguity). DeLite uses advanced NLP technology realized as Web services and accessed via a clearly defined interface. The system has been trained and evaluated with 315 users validating a corpus of 500 texts (6135 sentences). The results of the human judgments regarding the readability of the texts have been used as a basis for automatically learning the parameter settings of the DeLite component which computes the readability scores. To demonstrate the transfer of this approach to another language (in this case to English), a feasibility study has been carried out on the basis of a core lexicon for English, and the parser has been adapted to the most important linguistic phenomena of English. Finally, recommendations for further guidelines regarding the linguistic aspects of accessibility to the Web are derived.

Contents

1. Introduction

The amount of information and services which are offered over the Internet (e.g., Wikipedia, booking of train and concert tickets, renting DVDs) is continually growing.

At the moment, texts of Web pages are frequently not written using simple language. Instead, Web pages usually contain a lot of rare abbreviations, technical terms and also spelling or grammar errors. This is caused by the fact that, in a lot of cases, texts are not written carefully or difficult terminology is used (e.g., a lot of legal terms) without regard to the intended audience. Sometimes it also occurs that texts are created by automatic machine translation programs without any further manual verifications.

The readability of Web pages can have a major impact on the financial success of companies. On the Web, a consultation and discussion with a salesman or vendor has usually vanished completely. For this reason, people have to rely totally on the information they get over the Internet. Thus, it is very important that the corresponding Web pages are provided in an easy-to-understand way. Otherwise, potential customers are moving to competitors or they will cause additional costs by consulting the support. Also from the point of view of the customer, the usage of easy-to-understand language is important because it can save him time or — in the case of mental handicaps — it allows him to actually use that service at all.

The usage of simple language is not only important for Web pages of commercial enterprises but also for Web pages of governments and administrations. In this context, the European Council has made a recommendation to its member states to make their Web pages accessible, which also includes several linguistic criteria (see for instance the recommendations of the WCAG[1] [CCGV07]). Thus, several countries (including Germany) have committed themselves to fulfill this demand.

For the formulation of an easy-to-understand text, a tool for checking readability can be very helpful. The readability checker DeLite, developed by the IICS[2], is able to rate a given text concerning its readability as well as to identify difficult text passages.

Current readability formulas usually employ surface-based indicators (see Section 3.2 for a definition of *indicator*) for assessing readability, like sentence length, word length or the number of occurrences of words in a collection containing words which are assumed to be easy-to-understand [CD95, Fle48]. However, such indicators are often not adequate to realistically approximate the cognitive difficulties a person can have to understand a text. In contrast to that the DeLite readability checker is investigating texts on all linguistic levels, including the semantic level.

The development of the DeLite system relies on the NLP techniques of the IICS.

[1] WCAG is the abbreviation of **W**eb **C**ontent **A**ccessibility **G**uidelines and is developed by the W3C Consortium.
[2] IICS is the acronym for **I**ntelligent **I**nformation and **C**ommunication **S**ystems and denotes a subdivision of the computer science department of the University at Hagen.

In particular, the readability analysis is based on the syntactic, semantic and morphologic information derived by the deep linguistic parser WOCADI[Har03, HH97][3]. The communication between DeLite and WOCADI is realized over Web services with a clearly defined formal language interface. This clear division makes it possible to automatically improve the quality of the readability checking if the coverage of the parser is increased, even after the end of the DeLite project.

The DeLite system has been developed within the framework of the EU-project *Benchmark Tools and Methods for the Web* (BenToWeb, URL: `http://www.bentoweb.org`). The aim of this project was to investigate the accessibility of Web pages. Other aspects, besides readability, which were examined in this project included for instance color contrasts or the design of user interfaces [SHVV06, PPK+07].

The rest of this document is organized as follows: Section 2 gives an overview of related work and Section 3 introduces the DeLite readability checker, including its architecture and user interface. The readability criteria employed by DeLite are described in Section 4. Section 5 gives a description of DeLite's readability formula. Section 6 characterizes the methods for determining the parameters of this formula by the application of machine learning techniques. The evaluation of these methods is discussed in Section 7. Section 8 deals with the adaptation to other languages like English and contains a description of the English prototype of DeLite. Section 9 contains the recommendations for writing readable texts, which are based on the linguistic readability indicators that have been found to be important. Section 10 gives a conclusion and an outlook to future work. Finally, the appendix contains a description of all readability indicators employed by DeLite, as well as a list of data structures and formal definitions concerning of these indicators.

2. Related Work

Various methods to derive numerical values corresponding to text readability have been proposed. One of the most popular readability formulas, the Flesch Reading Ease score, was developed already in 1948 [Fle48]. For judging readability, this formula uses the average sentence length and the average number of syllables per word. The average sentence length is intended to roughly approximate the complexity of a sentence, while the number of syllables is related to word frequency since long words are usually used less often. The Flesch readability formula is defined as follows:

$$P = 206.835 - (1.015 \times ASL) - (84.6 \times AWL)$$

The variables have the following meaning:

P : readability score (scores around zero correspond to simple texts, scores around 100 to difficult texts)

ASL : average sentence length (measured in number of words)

AWL : average word length (measured in syllables)

Later, this formula was also adapted to German, resulting in the so-called Amstad Readability index [Ams78]. Despite of its age, the Flesch formula is still widely used. Moreover, its indicators "sentence/word length" are employed in various other readability formulas [Kla63].

[3] WOCADI is the abbreviation of **Wo**rd **Cla**ss based **Di**sambiguation.

The revised Dale-Chall readability index [CD95] depends on surface-type indicators as well. This index, analogous to Flesch, also employs the average sentence length for computing readability scores. But instead of recognizing difficult-to-understand words by counting the number of syllables it looks up each word in a list. In case a word occurs in this list, the word is considered as easy to understand. Thus, the average word complexity is determined by the percentage of words in a text which this list does not contain. To keep this list small, it contains only lemmas. Therefore, a lemmatization has to be done before lookup. This allows for instance to find the word *sleeps* if the list only contains the lemma *sleep*. Although readability formulas usually focus on surface-oriented type indicators, there exist several reformulation tools which also check for syntactic complexity [CS96].

The Coh-Metrix-Project is dealing with a special aspect of text readability of English texts, i.e., the text coherence [MLDM06]. The text coherence is determined by identifying several referential constructs like anaphora, temporal and spatial relations.

3. Basic Functionality and Architecture of DeLite

3.1. Overview

The DeLite system consists of several software components and interacts with various other NLP components developed by IICS (see Figure 1).

For entering the text to be checked and for displaying the results of the readability analysis, a *GUI-based Web Server Application* is used. The calculations related to readability checking are done by the *DeLite engine*. To investigate the structure of texts at different linguistic levels, the DeLite system relies on the syntactico-semantic analysis of the *WOCADI parser*. For that WOCADI employs both a knowledge base and the semantically oriented lexicon *HaGenLex*. The knowledge base contains a large number of semantic relations between concepts like hypernymy, hyponymy, synonymy, and antonymy , as well as meaning postulates, expressing more complicated relationships between concepts. HaGenLex can be divided into three parts, the *General lexicon DE*, the English *Lexicon EN* and the *Domain-specific lexicon DE*. A workbench for the computer lexicographer *LIA* is provided to comfortably add and modify lexical entries. The work of DeLite can be controlled by setting different parameter weights in the readability formula. These weights have been determined by the *Parameter learner*. The training data for the parameter learning were obtained by a readability study with more than 300 participants. For this study, a user interface and an associated Web server application were developed.

The readability checking of DeLite is done in the following way:

- The user enters the text he wants to be analyzed in a text field of the DeLite user interface.
- The Web browser calls the DeLite Engine to start a readability check of this text.
- The text is analyzed by the syntactico-semantic parser WOCADI. The result of this analysis consists of morphological information, a syntactic dependency tree, and a semantic network conforming to the MultiNet formalism [Hel06] (see Figure 8). An example for such a dependency tree is shown in Figure 2, the associated semantic network in Figure 3.
- On the basis of this analysis the text is investigated with regard to possible readability violations. Whereas the dependency tree returned by WOCADI is used

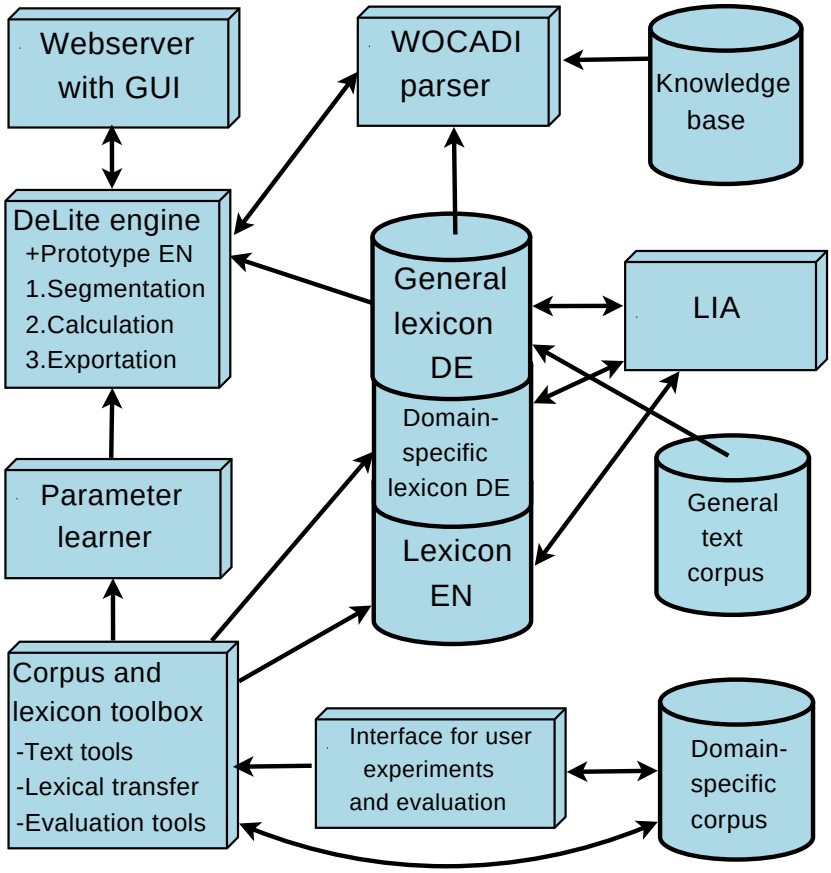

Figure 1: Interaction of DeLite with other NLP components. Arrows indicate data flow.

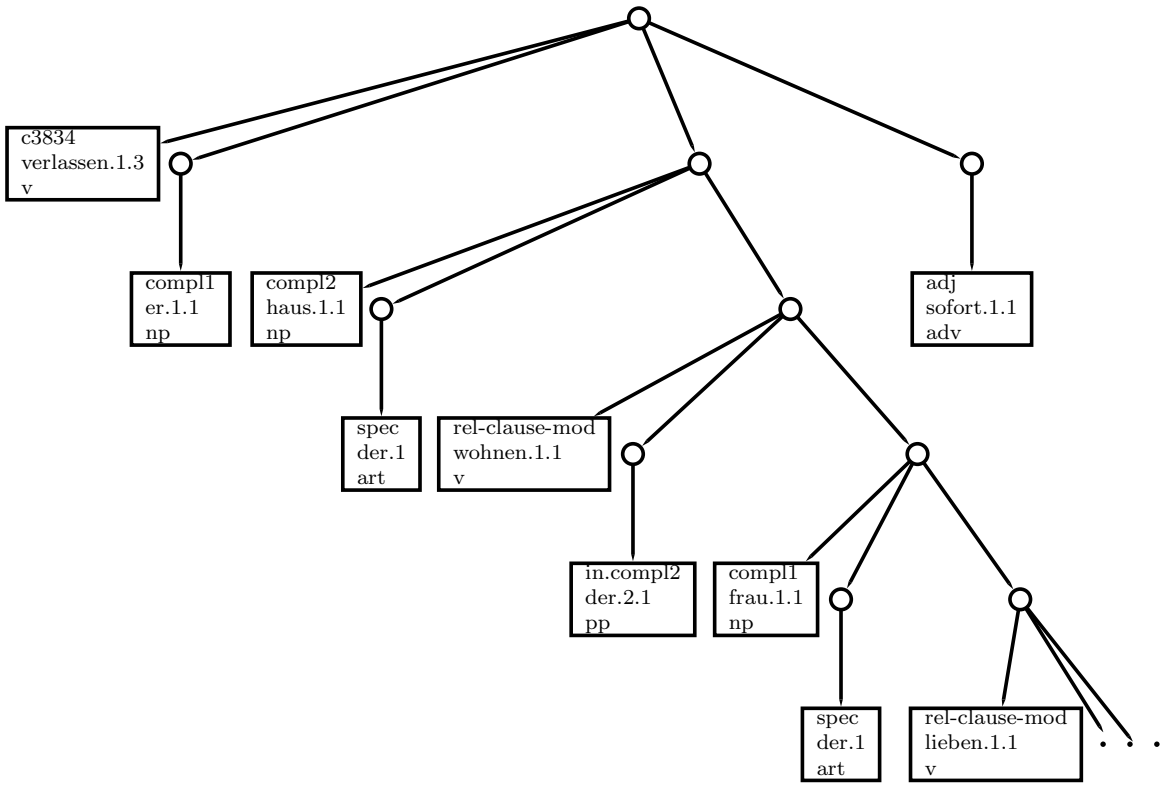

Figure 2: Dependency tree for the Sentence: *Er verließ das Haus, in dem die Frau, die er liebte, wohnte, sofort. (He left the house where the woman he loved lived immediately.)*

to recognize syntactic readability problems, like deeply embedded sentences or syntactic ambiguities, the semantic network is employed by the modules computing the semantic indicators, e.g., in counting the number of network nodes or the number of propositions per sentence. Morphological information and token information are required to identify compounds. Based on all the indicator values, a global readability score is calculated and difficult-to-read text passages are identified and passed to the Web-based user interface.

- All text passages with potential readability problems are highlighted in the Web browser. In addition, the global readability score is also displayed at the top of the Web page.
- The user can re-analyze the text after manual editing. This allows to iteratively improve the text until no further readability violations can be found.

The input document is processed by the DeLite Engine supervised by a controller and working in the following way (see Figure 4):

- A deep syntactico-semantic analysis of the text given is carried out by WOCADI.
- The Preparation Layer segments the input text into words, phrases and sentences.
- The individual indicator values are determined by the Calculation Layer using the results of WOCADI. Each indicator is attached to a certain module depending on whether the indicator requires information about words, phrases, sentences or the entire document. The corresponding module iterates over all text segments of its associated segment type and triggers the calculation of the corresponding indicators. While lexical and morphological indicators are applied to words, semantic and syntactic indicators usually operate on the sentence level.

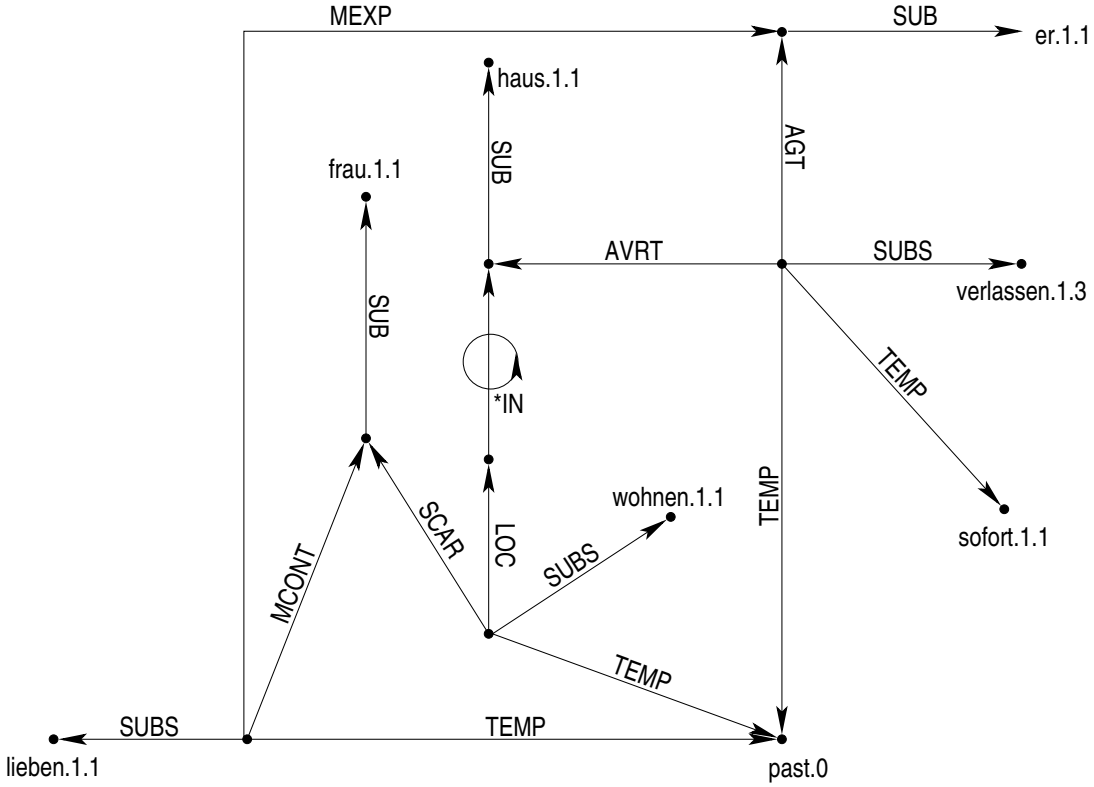

Figure 3: Semantic Network for the sentence: *Er verließ das Haus, in dem die Frau, die er liebte, wohnte, sofort. (He left the house where the woman he loved lived immediately.)* Only the top level concepts and semantic relations are shown.

Since each indicator is dealt with by a separate subcomponent, it is quite easy to exchange, remove or add indicators. As a result of this calculation, the text segments are associated to indicator values.

- In the Evaluation Layer, the individual indicator values are aggregated, normalized and combined into a single readability score. Furthermore, text segments are identified, for which an indicator value exceeds a predefined threshold, thus indicating text passages which are difficult-to-read.
- Finally, all this information is marked up in XML (see Figure 5, 6 and 7) as well as in a HTML format which is intended for graphical presentation. The resulting files are returned to the calling process by the Exportation Layer.

DeLite provides a GUI to support the comfortable checking of texts with regard to their readability (see Figure 9). The types of readability problems are categorized on the following five levels: morphological, lexical, syntactic, semantic, and discourse level (described in Section 4 in detail). If the user selects a name of a readability problem (on the right side) the associated critical passages are highlighted in the text field with the corresponding color. When moving the mouse pointer over the highlighted text passage, a short description of the readability problem is displayed as a so-called tooltip. Furthermore, DeLite shows a total readability score (upper right part), readability scores on each linguistic level and some statistical information (on the left side). The sentence displayed in Figure 9 contains a pronominal ambiguity. The pronoun *er (he)* can either refer to *Mr. Müller* or to *Dr. Peters*. When the user selects the pronoun *er*, both possible antecedents are displayed in bold face by DeLite for a better recognition of the readability problem.

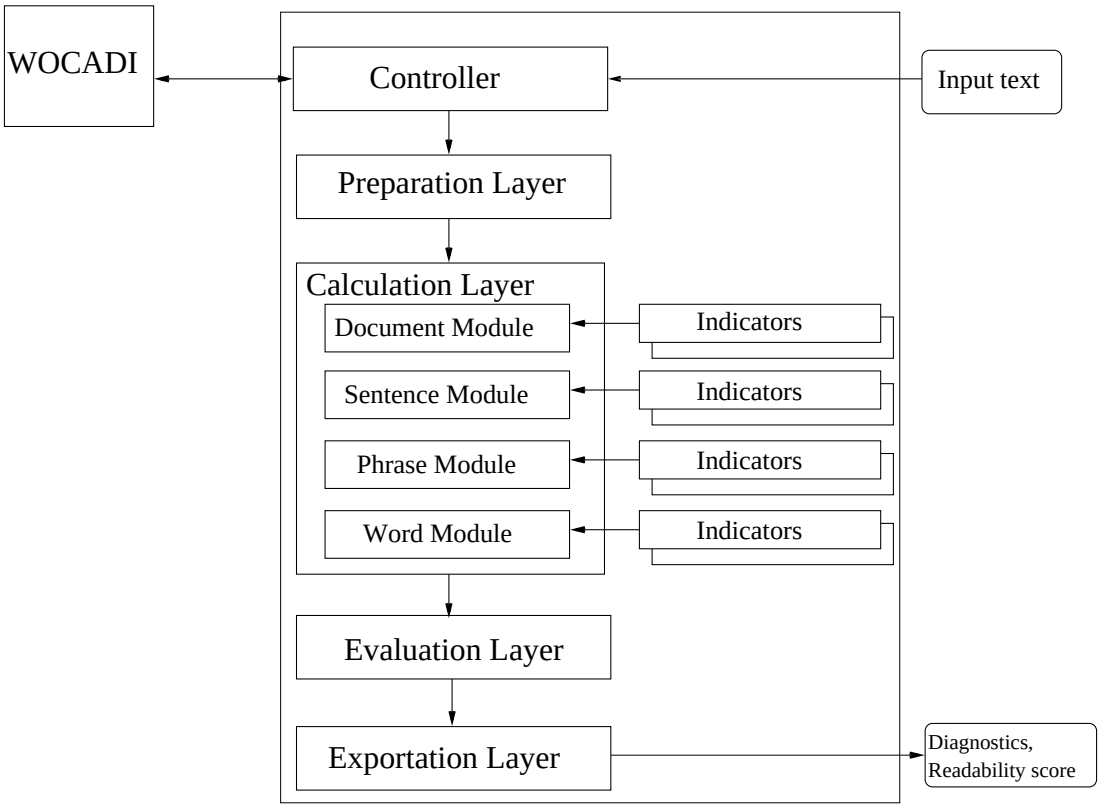

Figure 4: The internal architecture of DeLite.

Figure 10 shows a second example with a sentence containing a complex syntactic structure. DeLite indicates that this sentence contains a deeply embedded sub-clause (indicator value for *center embedding depth*: 2).

3.2. Readability Criteria versus Readability Indicators

In order to be treatable by automatic computation, the readability criteria under consideration must be based on attributes and their values which can be determined algorithmically, that is, by the methods of natural language processing (NLP). To this end, the DeLite system relies on a variety of basic numerical readability indicators that can be automatically extracted from the results of the linguistic analysis provided by the WOCADI server. These indicators mostly encode numeric information such as the number of word tokens in a text or the number of propositions expressed in a sentence. Note that indicators can be associated with all levels of linguistic analysis. The violation of a given readability criterion is then defined by means of a function over a certain subset of these indicators. Similarly, the overall readability scoring of texts or text passages is defined as a function over readability indicators (see also Section 5).

4. Readability Criteria Employed by DeLite

The readability criteria of DeLite are grouped into several readability criteria: Morphological Level, Lexical Level, Syntactic Level, Semantic Level and Discourse Level.

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<doc id="d0" start="0" end="67" length="68" type="text">
Er verlie&#223; das Haus, in dem die Frau, die er liebte, wohnte, sofort.
<sentence id="d0s0" start="0" end="68" length="68" type="declarative-sentence">
Er verlie&#223; das Haus, in dem die Frau, die er liebte, wohnte, sofort.

<word id="d0s0w0" start="0" end="2" length="2" type="simplicium">
Er
</word>
<word id="d0s0w1" start="3" end="10" length="7" type="simplicium">
verlie&#223;
</word>
<word id="d0s0w2" start="11" end="14" length="3" type="simplicium">
das
</word>
<word id="d0s0w3" start="15" end="19" length="4" type="simplicium">
Haus
</word>
<word id="d0s0w4" start="19" end="20" length="1" type="punctuation">
,
</word>
<word id="d0s0w5" start="21" end="23" length="2" type="simplicium">
in
</word>
<word id="d0s0w6" start="24" end="27" length="3" type="simplicium">
dem
</word>
<word id="d0s0w7" start="28" end="31" length="3" type="simplicium">
die
</word>
<word id="d0s0w8" start="32" end="36" length="4" type="simplicium">
Frau
</word>
<word id="d0s0w9" start="36" end="37" length="1" type="punctuation">
,
</word>
<word id="d0s0w10" start="38" end="41" length="3" type="simplicium">
die
</word>
<word id="d0s0w11" start="42" end="44" length="2" type="simplicium">
er
</word>
<word id="d0s0w12" start="45" end="51" length="6" type="simplicium">
liebte
</word>
...
<phrase id="d0s0p0" start="0" end="2" length="2" type="maximal-np">
Er
</phrase>
...

</sentence>

</doc>
```

Figure 5: Simplified format example of an XML Report R1.

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<doc id="d0" start="0" end="67" length="68" type="text"
 abbreviation_type_token_ratio="0"
 num_sentences="1"
 num_words="18"
 avg_sentence_length="18"
 num_characters="51"
 num_syllables="17"
 num_simplicia="13"
 num_compounds="0"
 num_nouns="2"
 num_acronym_types="0"
 num_acronym_tokens="0"
 num_wordform_types="14"
 num_wordform_tokens="18"
 num_lemma_types="11"
 num_lemma_tokens="18"
...
>
Er verlie&#223; das Haus, in dem die Frau, die er liebte, wohnte, sofort.
<sentence id="d0s0" start="0" end="68" length="68" type="declarative-sentence"
 num_sentence_constituents="9"
 num_words="18"
 analysis_passes="0.367"
 longest_path_sym="11"
 max_path="3"
 num_connections="2.25"
 num_concept_nodes="12"
 num_propositions="3"
 num_introduced_concepts="0"
...
>
Er verlie&#223; das Haus, in dem die Frau, die er liebte, wohnte, sofort.

<word id="d0s0w0" start="0" end="2" length="2" type="simplicium"
 parse_lemma="er"
 pos="perspro"
 num_characters="2"
 frequency_class="4"
 inverse_lemma_frequency="3.255865441593e-6"
 lemma_frequency="307138"
 pronoun_without_antecedent="1"
 distance_verb_complement="0">
Er
</word>
<word id="d0s0w1" start="3" end="10" length="7" type="simplicium"
 parse_lemma="verlassen"
 pos="v"
 num_characters="7"
 num_syllables="2"
 ...
 >
verlie&#223;
</word>
...
</sentence>
</doc>
```

Figure 6: Simplified format example of an XML Report R2.

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<doc id="d0" start="0" end="67" length="68" type="text"
 dis_all_score="1.0"
 sem_all_score="0.63222455684782"
 syn_all_score="0.63606437965051"
 lex_all_score="0.8749890002853"
 mor_all_score="0.9921360576734"
 readability_all_score="0.76078020763843"
 mor_weight="0.12963086437241"
...
 syn2_4238_score="0.99745589633223"
 syn2_4238b_score="0.99745589633223"
 syn2_4238c_score="0.9806703996899"
 syn4_4239a_score="0.7753618651709"
 syn4_4239b_score="0.0770984641946"
 sem1_4241_score="0.12427192556533"
 sem1_4242a_score="0.92700465925699"
 sem1_4242c_score="0.93387505909682"
 sem1_4243_score="0.0018858210882751"
 sem2_4244a_score="0.5343492271672"
 sem2_4244b_score="0.50174999232086"
 sem2_4244c_score="0.50249997840496"
 sem3_4245_score="0.64955179261608"
 sem3_4246_score="0.8423012847464"
 sem3_4247_score="0.43624564716013"
 dis1_4251_score="0.82429705366716"
 syn2_4233b_score="0.45234587941059"
 syn2_4237_score="9.2833749469989e-4"
 syn6_42310b_score="0.034835845492951"
 mor2_4211b_score="0.64873950263849"
 mor3_4213_score="0.99736438694611"
 mor3_4214_score="0.99727631136146"
 lex2_4222_score="0.0054027389687058"
 lex2_4223_score="0.02699485065661"
 lex4_4224_score="0.26894148129203"
 lex8_4225_score="0.99552770397011"
 mor5_4217a_score="0.99490457751781"
 syn2_4233a_score="0.53635498940877"
 syn7_42312_score="0.87468792934374"
 syn8_42313_score="2.6655495369954e-5"
 syn8_42314_score="5.3530285210046e-5"
 mor5_4217b_score="0.9921360576734"
 lex1_4221a_score="0.98706434192462"
 lex1_4221b_score="0.96890563173919"
 syn6_42310a_score="0.37471789964436"
 dis1_4252_score="0.11531743509851">
Er verlie&#223; das Haus, in dem die Frau, die er liebte, wohnte, sofort.
<sentence id="d0s0" start="0" end="68" length="68" type="declarative-sentence"
 dis1_4251_score="0"
 sem3_4247_score="11"
 sem3_4246_score="3"
...
</sentence>
</doc>
```

Figure 7: Simplified format example of an XML Report R3.

```
;; dependency tree
(dep-tree (("" "c434" "verlie" "verlassen.1.3" v) (("compl1" "c433" "Er" "er.1.1"
np)) (("compl2" "c438" "Haus" "haus.1.2" np) (("spec" "c437" "das" "der.1" art))
(("rel-clause-mod" "c601" "wohnte" "wohnen.1.1" v) (("in.ctxt" "c584" "dem" "der.2.1"
pp)) (("compl1" "c588" "Frau" "frau.1.1" np) (("spec" "c587" "die" "der.1" art))
(("rel-clause-mod" "c598" "liebte" "lieben.1.1" v) (("compl2" "c592" "die" "der.2.1"
np)) (("compl1" "c597" "er" "er.1.1" np)))))) (("adj" "c607" "sofort" "sofort.1.1"
adv)))) ;; semantic network
(net (
;; semantic relations
(sub ‘‘c433’’ ‘‘er.1.1’’ categ situa) (agt ‘‘c434’’ ‘‘c433’’ categ situa) (avrt
‘‘c434’’ ‘‘c438’’ categ situa) (temp ‘‘c434’’ ‘‘past.0’’ categ situa) (temp ‘‘c434’’
‘‘sofort.1.1’’ categ situa) (subs ‘‘c434’’ ‘‘verlassen.1.3’’ categ situa) (sub ‘‘c438’’
‘‘haus.1.2’’ categ situa) (sub ‘‘c588’’ ‘‘frau.1.1’’ categ situa) (sub ‘‘c597’’
‘‘er.1.1’’ categ situa) (mcont ‘‘c598’’ ‘‘c588’’ categ restr) (mexp ‘‘c598’’ ‘‘c597’’
categ situa) (subs ‘‘c598’’ ‘‘lieben.1.1’’ categ situa) (temp ‘‘c598’’ ‘‘past.0’’ categ
situa) (ctxt ‘‘c601’’ ‘‘c438’’ restr situa) (scar ‘‘c601’’ ‘‘c588’’ categ situa) (temp
‘‘c601’’ ‘‘past.0’’ categ situa) (subs ‘‘c601’’ ‘‘wohnen.1.1’’ categ situa)
;; layer features
(sort ‘‘sofort.1.1’’ t) (base ‘‘sofort.1.1’’ ‘‘sofort’’) (sort ‘‘c438’’ (dis d io))
(card ‘‘c438’’ 1) (etype ‘‘c438’’ 0) (fact ‘‘c438’’ real) (gener ‘‘c438’’ sp) (quant
‘‘c438’’ one) (refer ‘‘c438’’ det) (varia ‘‘c438’’ con) (base ‘‘c438’’ ‘‘Haus’’) (orth
‘‘c438’’ ‘‘Haus’’) (sort ‘‘c434’’ da) (gener ‘‘c434’’ sp) (base ‘‘c434’’ ‘‘verlassen’’)
(orth ‘‘c434’’ ‘‘verließ’’) (mod ‘‘c434’’ ind) (tem ‘‘c434’’ past) (v-form ‘‘c434’’
finite) (v-gend ‘‘c434’’ act) (sort ‘‘c433’’ o) (gener ‘‘c433’’ sp) (refer ‘‘c433’’
det) (base ‘‘c433’’ ‘‘er’’) (orth ‘‘c433’’ ‘‘Er’’) (sort ‘‘haus.1.2’’ (dis d io))
(etype ‘‘haus.1.2’’ 0) (gener ‘‘haus.1.2’’ ge) (base ‘‘haus.1.2’’ ‘‘Haus’’) (sort
‘‘past.0’’ t) (etype ‘‘past.0’’ 0) (base ‘‘past.0’’ ‘‘past’’) (sort ‘‘verlassen.1.3’’
da) (gener ‘‘verlassen.1.3’’ ge) (base ‘‘verlassen.1.3’’ ‘‘verlassen’’) (sort ‘‘c601’’
st) (gener ‘‘c601’’ sp) (base ‘‘c601’’ ‘‘wohnen’’) (orth ‘‘c601’’ ‘‘wohnte’’) (tem
‘‘c601’’ past) (v-form ‘‘c601’’ finite) (v-gend ‘‘c601’’ act) (sort ‘‘c588’’ d) (card
‘‘c588’’ 1) (etype ‘‘c588’’ 0) (fact ‘‘c588’’ real) (gener ‘‘c588’’ sp) (quant ‘‘c588’’
one) (refer ‘‘c588’’ det) (varia ‘‘c588’’ con) (base ‘‘c588’’ ‘‘Frau’’) (orth ‘‘c588’’
‘‘Frau’’) (sort ‘‘wohnen.1.1’’ st) (gener ‘‘wohnen.1.1’’ ge) (base ‘‘wohnen.1.1’’
‘‘wohnen’’) (sort ‘‘frau.1.1’’ d) (etype ‘‘frau.1.1’’ 0) (gener ‘‘frau.1.1’’ ge)
(base ‘‘frau.1.1’’ ‘‘Frau’’) (sort ‘‘c598’’ st) (gener ‘‘c598’’ sp) (base ‘‘c598’’
‘‘lieben’’) (orth ‘‘c598’’ ‘‘liebte’’) (tem ‘‘c598’’ past) (v-form ‘‘c598’’ finite)
(v-gend ‘‘c598’’ act) (sort ‘‘c597’’ d) (gener ‘‘c597’’ sp) (refer ‘‘c597’’ det) (base
‘‘c597’’ ‘‘er’’) (orth ‘‘c597’’ ‘‘er’’) (sort ‘‘lieben.1.1’’ st) (gener ‘‘lieben.1.1’’
ge) (base ‘‘lieben.1.1’’ ‘‘lieben’’) (sort ‘‘er.1.1’’ o) (gener ‘‘er.1.1’’ sp) (refer
‘‘er.1.1’’ det) (base ‘‘er.1.1’’ ‘‘er’’)))
```

Figure 8: Dependency tree and semantic network as returned by WOCADI.

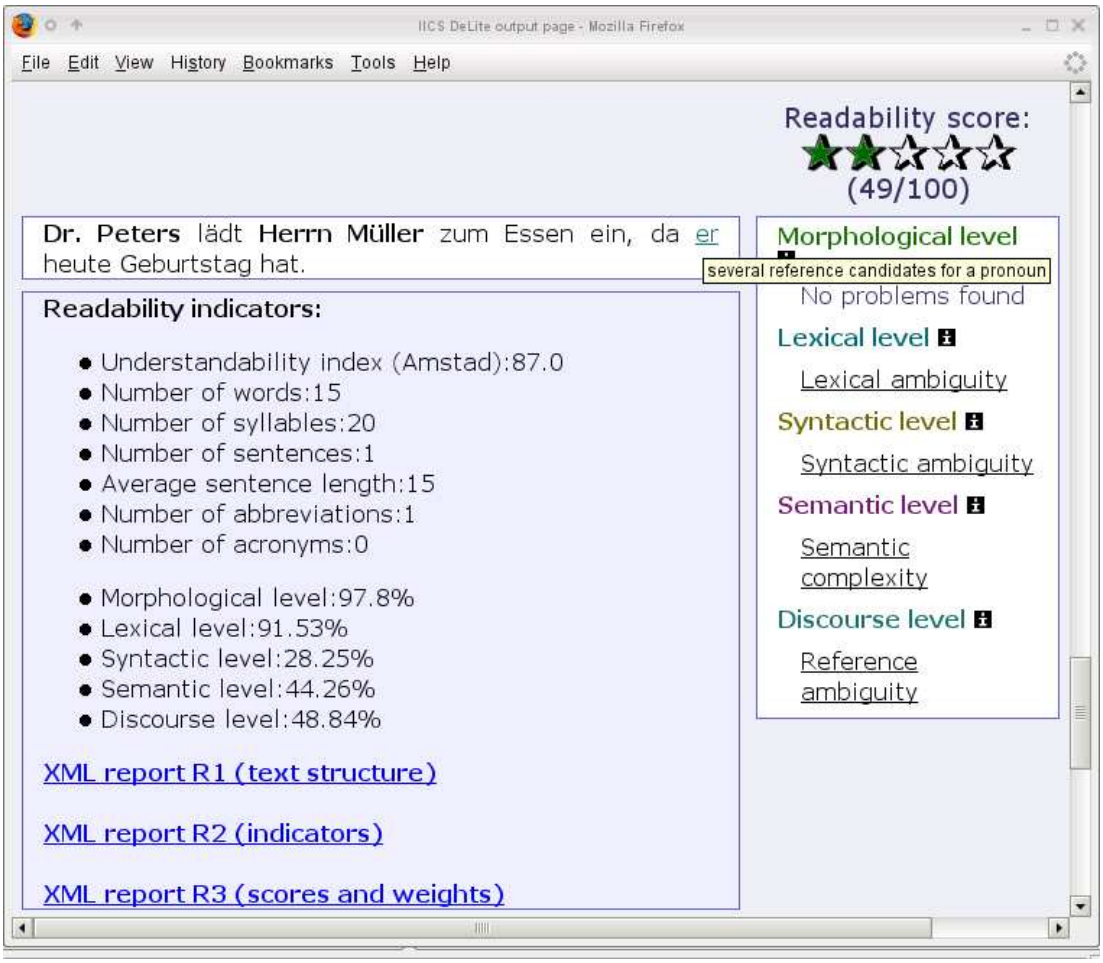

Figure 9: Screenshot of the user interface of DeLite, where a pronoun reference ambiguity is indicated. (English translation of the example sentence: *Dr. Peters invites Mr. Müller for dinner because it's his birthday today.* (literally: *Dr. Peters invites Mr. Müller for dinner since he has birthday today.*))

4.1. Morphological Level

1. *Derivation.* For the calculation of this indicator, derivational information for deverbal and deadjectival nouns is exploited. More specifically, nouns which are semantically connected to a verb or an adjective by the semantic relation CHEA or CHPA, respectively, are considered as being derived.
Indicators: is-deverbal-noun, is-deadjectival-noun

2. *Compound complexity.* Compounds are very frequent in German and can be quite complex. WOCADI automatically analyzes compound words. In contrast to shallower approaches, the complexity is not only determined on morphological grounds (i.e., just counting morphemes) but also on the semantic level (i.e., how many conceptual constituents the compound consists of). Such a deeper approach more precisely identifies those compounds which are problematic for human readers.
Indicators: num-compound-simplicia (number of words forming the compound), num-compound-concepts (number of concepts building the semantics of the compound)

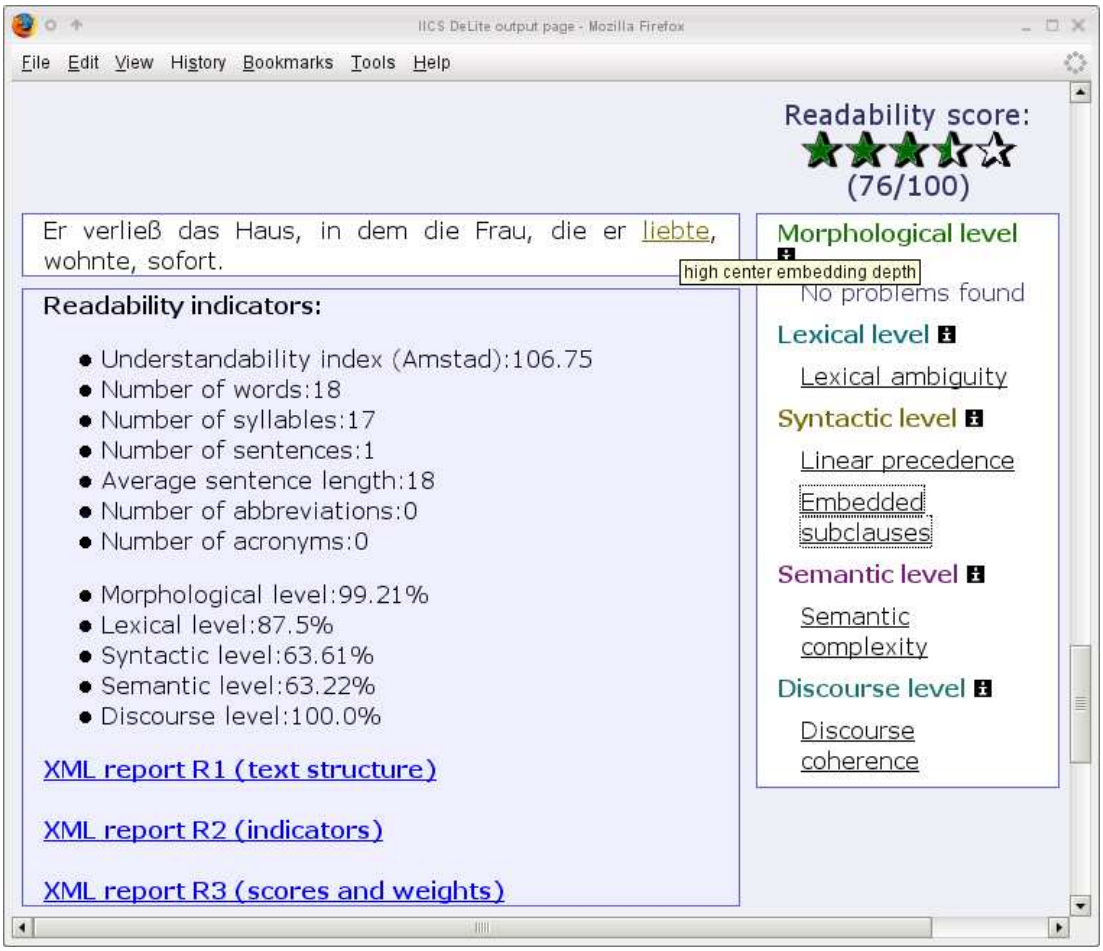

Figure 10: Screenshot of the user interface of DeLite, where a syntactic complexity is indicated.

3. *Abbreviations and acronyms.* Excessive use of abbreviated words increases the requirements to keep both short and long forms in mind. In particular, the usage of many *different* abbreviations is critical, which can be measured by dividing the number of different abbreviations by the number of all abbreviations.
Indicators: abbreviation-type-token-ratio, acronym-type-token-ratio

4. *Word length.* Long words can consume a large part of the reader's attention.
Indicators: num-syllables, num-characters

4.2. Lexical Level

1. *Word frequency.* Rare words are unfamiliar to many readers and thus can impede reading ease. The indicators for this criterion rely on frequency lists derived from large corpora. Especially problematic are foreign words that are not frequent for a given domain and audience. The choice of corpora (and thereby the derived frequency list) is a subtle issue; one might even devise mechanisms so that user groups can supply personalized frequency lists.
Indicators: frequency-class (word forms are divided into several classes occurring to their frequency), inverse-lemma-frequency

2. *Lexical ambiguity.* These ambiguities arise when a word can have more than

one meaning in a given context. The most difficult part is to distinguish ambiguities that exist only for the machine (so-called spurious or parasitic ambiguities, which are artifacts of the linguistic analysis) from those ambiguities that are also relevant to human readers.
Indicators:
num-readings-from-lookup, num-readings-from-parse

3. *Naming consistency.* Synonymy relations (SYNO) from HaGenLex form the basis for determining naming consistency in a text. The synonymy relation is an equivalence relation which induces so-called synsets as equivalence classes of words.
Indicator: synset-size (number of synonyms used for the same concept)

4. *Lexical abstractness.* Abstract nouns are considered more difficult to read than concrete ones (especially if they occur infrequently).
Indicator: is-abstract-noun

5. *Vocabulary complexity.* This criterion mirrors the richness of the vocabulary, often in relation to the text length.
Indicators: lemma-type-token-ratio (ratio of *different* to *all* lemmas),
wordform-type-token-ratio (ratio of *different* to *all* word forms)

4.3. Syntactic Level

1. *Syntactic ambiguity.* Ambiguities on the syntactic level come in various types. Most frequent and most irritating to readers are scope and attachment ambiguities, which the respective DeLite module tries to locate (Example: *Peter saw the man with the telescope.* In this sentence, the telescope can be the instrument of seeing or be attached to the man. Similarly to lexical ambiguities, parsers and human readers may be confused by different sets of ambiguities and have different problems in resolving ambiguities.
Indicators: num-complement-ambiguities, num-pp-attachment-candidates

2. *Syntactic complexity.* Complexity on the syntactic level is a core criterion towards readability [Gro92]. Sentences having too much constituents can be a major obstacle to understanding a text. Different complexity measures can be determined from the syntactic structure (dependency graph) of a sentence.
Indicators: num-dependents-per-verb, num-dependents-per-np, num-constituents-per-coordination (especially problematic because coordinations often involve ellipsis), num-np-words, num-ap-words

3. *Sentence length.* Though this is admittedly a simple criterion, it has proven its worth in the past as one indicator for the understandability of a sentence. Sentence length is a valuable measure in combination with other metrics on syntactic complexity and is the perfect indicator for fall-back strategies in case the deep linguistic analysis fails.
Indicators: num-sentence-words, num-sentence-constituents

4. *Linear precedence complexity.* A long distance between a verb and its separated

prefix[4], its complements, or its adjuncts is problematic for ease of understanding.
Indicators: distance-verb-prefix, distance-verb-complement, distance-verb-adjunct, distance-verb-group-parts

5. *Passive form.* For the purpose of good readability passive sentence constructions should be avoided [Gro92].
Indicator: is-passive

6. *Deeply embedded subclauses*: Sentences with deeply embedded subclauses can make a sentence difficult to understand [Gro92]. The difficulty can be further increased if the subordinate clause is embedded into the middle of a clause since the reader has to memorize the interrupted superior clause until its continuation after the termination of the subordinate clause.
Indicators: clause-embedding-depth, clause-center-embedding-depth

4.4. Semantic Level

1. *Semantic complexity.* Complexity on the semantic level can be assessed by investigating the semantic representation of sentences. For example, the number of conceptual nodes in the semantic network or the number and type of relations in the semantic representation can lead to good indicators.
Indicators: num-propositions-per-sentence, num-propositions-per-concept, num-relations-in-cluster (this indicator checks for complicated semantic subnetworks, e.g., relational chains built of reasons (REAS), causes (CAUS), justifications (JUST), and concessions (CONC)), num-concept-nodes-per-sentence (number of semantic concept nodes created for a sentence)

2. *Negations.* The usage of too many negations can make a text more difficult to read [Gro92]. DeLite distinguishes indicators for concept negations, negated adjectives and multiple negations.
Indicators: num-negations, num-negated-adjectives, num-negated-concepts

4.5. Discourse Level

1. *Discourse coherence.* A text should be as coherent as possible.
Indicators: num-introduced-concepts-per-sentence, num-pronouns-without-antecedents.

2. *Coreference Ambiguity.* A pronoun which can refer to more than one preceding constituent in the text is a common and often irritating phenomenon. WOCADI's coreference module identifies such instances.
Indicator: num-reference-candidates

3. *Distance of Pronoun and Antecedent.* A large distance between a pronoun and its antecedent can make it difficult for the reader to relate the pronoun to its antecedent. The distance is measured as the number of words or the number of sentences occurring between the pronoun and its antecedent.
Indicators: reference-distance-in-words, reference-distance-in-sentences

[4] A phenomenon often occurring in German.

5. DeLite Readability Formula

5.1. The DeLite Approach for Computing a Global Readability Score

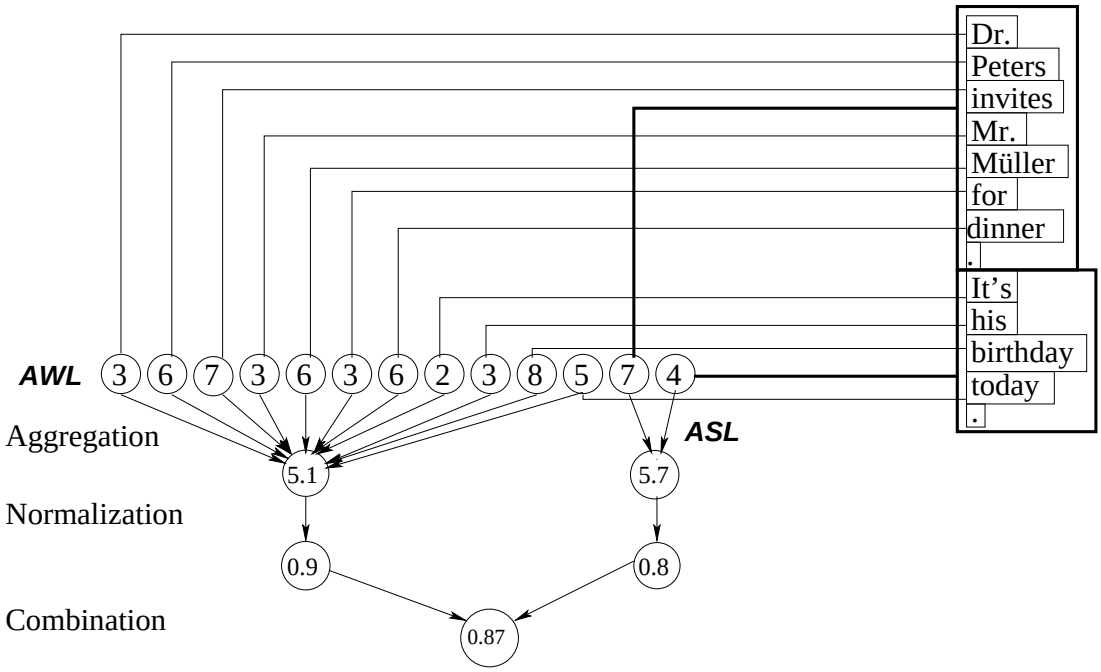

Figure 11: Calculation of a global readability score with the two indicators *average word length (AWL)* and *average sentence length (ASL)*.

In DeLite the calculation of the global readability score is done in several steps (see Figure 11):

- Segmentation: In the first step the entire document is segmented into words, phrases and sentences.
- (Basic) Calculation: Indicator values are calculated for each segment the indicators are associated to, e.g., the indicator *num-compound-concepts* calculates one value for every word, the indicator *average sentence length* for every sentence.
- Aggregation: For each indicator its values associated to text segments are averaged. This average is called the aggregated indicator value.
- Normalization: The aggregated indicator values are normalized, i.e., mapped to the interval from zero to one.
- Combination: In the last step, a global readability score is determined by calculating a weighted sum of all aggregated and normalized indicator values. All weights are non-negative and sum up to one.

5.2. Comparison with Other Readability Formulas

Most readability formulas do not do any normalization but combine the indicator values directly. However, a normalization has some important advantages.

Since all indicators have the same value range after the normalization, the combination of the individual indicators can be carried out by using a weighted sum with normalized and non-negative weights. The weights, which are determined automatically by an optimization algorithm, serve not only the purpose of combining the

indicators but also of filtering out irrelevant indicators. Indicators with a weight of zero can be removed automatically. Furthermore, it is guaranteed that unimportant indicators have only a very small influence on the readability score. This would not be true, if the indicators were combined directly, i.e., without using a normalization, which is done by a lot of existing readability formalisms [Fle48, Ams78]. In this case, additional work is necessary to determine a minimal set of indicators to avoid overfitting to the data. A further advantage of the usage of normalized weights consists in the fact that the importance of each indicator becomes immediately obvious.

However, two problems have to be dealt with by following the normalization approach. First, ordinary linear regression does not allow the usage of inequality constraints which would be necessary to ensure non-negative weights. Second, parameters for the normalization of the indicator values have to be determined additionally. Therefore, in the next subsections, we describe the treatment of these problems.

5.3. Indicator Normalization

Non-normalized indicators have very different probability distributions, mean values and variances (consider, for example, the sentence length versus the number of syllables in a word). To avoid this, indicator values have to be normalized, mapping them into the interval $[0, 1]$.

For simplicity, we presume non-negative indicator values, and that higher indicator values correspond to a worse readability. The normalization is achieved by applying a variation $N(x)$ of the Fermi-function to the non-normalized indicator values. $N(x)$ is defined by the following equation:

$$N(x) = 1 - \frac{1}{1 + e^{-\frac{x-\mu}{\delta}}}$$

This formula is based on two additional parameters, μ and δ, which have to be determined for every individual indicator. The parameter μ is the location of the 0.5-intercept ($N(x) = 0.5$), δ specifies the gradient of the function. The form of this function for an arbitrarily chosen indicator is shown in Figure 12.

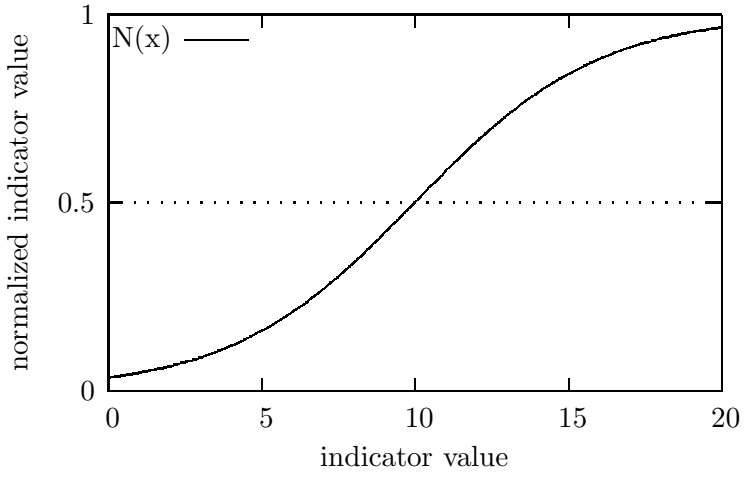

Figure 12: Normalization function N(x) for $\mu = 10$ and $\delta = 3$.

5.4. Combining Readability Indicators

The readability score $R(i)$ for a text d_i is calculated as a weighted sum, combining all indicator values x_{ij}. The function is given as:

$$R(i) = \sum_{j=1}^{m} w_j x_{ij}$$

In the remainder of this paper, weights are assumed to be normalized, i.e.,

$$\sum_{j=1}^{m} w_j \quad = \quad 1 \quad \text{and} \quad w_j \geq 0 \tag{1}$$

In summary, to compute the readability score $R(i)$, for each indicator I_j, its weight w_j and its normalization parameters μ_j and δ_j have to be determined.

6. Machine Learning Algorithms

The parameters for the normalization function and the weights for combining the normalized indicator values are computed by applying methods from machine learning (see also [vL07]).

6.1. Determining Parameters of the Normalization Function

The parameter μ_j of the normalization function $N_j(x)$ for a given indicator I_j determines the 0.5-intercept. It usually corresponds to some point near the center of the distribution of the indicator values. Several methods for calculating the parameters μ and δ of the normalization function were tested, including techniques based on analyzing conditional probabilities.

Selecting the mean value of the indicator-value distribution for μ yielded the smallest error in comparison to the user ratings. The parameter δ was obtained by computing the arithmetic means for solutions of $N(x)$ for given values of μ and maximum and minimum values of the indicator value under consideration.

6.2. Determining Indicator Weights

Basically, two types of machine learning algorithms to determine the parameters in a weighted sum are applicable: Algorithms that depend on a specific probability distribution and algorithms which do not. A method of the first type is, for instance, the expectation maximization algorithm [DLR77]. Note that this algorithm cannot be applied directly to a data set if the indicators are highly correlated among each other. In this case a transformation technique like principal component analysis [Jol86] is necessary to create a new dataset with independent indicators. Since there are a lot of different indicators with varying probability distributions, methods of the second type were preferred. More specifically, two types of regression algorithms were investigated. Since regression can also be used on highly correlated data, no data transformation is necessary by following such an approach. However, the indicator values still have to be linearly independent of each other. The aim is to minimize the square (or absolute) error between the DeLite readability function and the user

ratings. In the case of the square error, the solution for the optimization problem given by the Equation 2.

$$\mathbf{w_{opt}} = arg \min_{w_1,\ldots,w_m} \sum_{i=1}^{n} (y_i - \sum_{j=1}^{m} x_{ij} w_j)^2 \tag{2}$$

The variables specified above have the following meanings:

- y_i: average user rating for text d_i. This value is determined from the global readability judgment of test persons. Values of the discrete seven-point Likert scale are converted into a value between zero and one by a linear transformation. A value of one represents optimal, a value of zero worst readability.
- w_j: weight for indicator j (to be determined)
- x_{ij}: value between 0 and 1 for text d_i and indicator I_j

Using vector notation with:

- $\mathbf{X_i} : (x_{i1} \ldots x_{im})^T$
- $\mathbf{w} : (w_1 \ldots w_m)^T$

this equation can be rewritten as follows:

$$\mathbf{w_{opt}} = arg \min_{\mathbf{w}} \sum_{i=1}^{n} (y_i - \mathbf{X_i w})^2 \tag{3}$$

Because all weights must be non-negative, an ordinary linear regression cannot be used. Instead of that, this problem could be solved by quadratic programming which would raise the complexity enormously. Thus, two alternatives were investigated, one exact robust regression method (see Section 6.3) and an approximative method based on linear regression (see Section 6.4).

6.3. Robust Regression with Linear Optimization

The minimization of the error is done using linear optimization which is a robust regression method. In this case the parameters are estimated by minimizing the sum of the absolute errors instead of the square errors. This method is called robust since it is not as sensitive to outliers as linear regression. The minimization problem for determining the weights of the DeLite readability function can be defined as follows:

$$\mathbf{w_{opt}} = arg \min_{\mathbf{w}} \sum_{i=1}^{n} |y_i - \mathbf{X_i w}| \tag{4}$$

This optimization problem can be transformed in the following way by introducing additional variables $z_1, \ldots, z_n$:

$$arg \min_{\mathbf{w}} \quad \textstyle\sum_{i=1}^{n} z_i \quad with \quad z_i \quad \geq |y_i - \mathbf{X_i w}| \tag{5}$$

This problem is equivalent to the original optimization problem since the solutions for z_i are the lowest numbers which are greater than $|y_i - \mathbf{X_i w}|$ [BT97]. Since

$$z_i \geq |y_i - \mathbf{X_i w}| \Leftrightarrow (z_i \geq y_i - \mathbf{X_i w} \wedge z_i \geq -(y_i - \mathbf{X_i w})) \tag{6}$$

the constraints can be changed to

$$z_i \geq (y_i - \mathbf{X_i w}) \tag{7}$$
$$z_i \geq -(y_i - \mathbf{X_i w}) \tag{8}$$

This problem can be solved by common linear optimization algorithms. A popular and efficient algorithm to solve this problem is the simplex algorithm. It reduces the costs continually by walking on the vertices of the polygon which constrains the solution space. Since the cost vector and the constraints are both linear, the solution is guaranteed to be located on such a vertex.

6.4. Iterative Linear Regression

It is also possible to get a good approximation of weights minimizing the square error by using a linear regression with Lagrange restriction. Such a regression problem has the general form:

$$\mathbf{w_{opt}} = arg\min_{\mathbf{w}} \sum_{i=1}^{n} (y_i - \mathbf{X_i w})^2$$

with an additional equality constraint: $\mathbf{Lw} = q$, where $\mathbf{L}$ is a matrix with n columns and and the same number m of rows as q. In our case, the equality constraint represents the condition that all weights sum up to one: $\mathbf{L} = (1...1)^T$ and $q = (1)$. According to [Gre93], this optimization problem can be transformed to

$$\mathbf{W} = \begin{bmatrix} \mathbf{X^T X} & \mathbf{L^T} \\ \mathbf{L} & 0 \end{bmatrix} \quad \mathbf{u} = \begin{bmatrix} \mathbf{X^T y} \\ q \end{bmatrix} \tag{9}$$

$$\mathbf{W} \begin{bmatrix} \mathbf{w} \\ \lambda \end{bmatrix} = \mathbf{u} \tag{10}$$

(λ is the Lagrange multiplier and can be ignored in the solution). The solution can be found by:

$$\begin{bmatrix} \mathbf{w} \\ \lambda \end{bmatrix} = \mathbf{W^{-1} u} \tag{11}$$

However, the weights calculated by this method might be negative. Negative weights are due to one of the following reasons: First, they can result if some of the indicators are not correlated with the average user ratings. Second, they can be caused when some of the indicators are strongly correlated among each other. The first problem can be avoided by setting all weights to zero for indicators which are not correlated with the average user ratings $y_1, \ldots, y_n$. The regression as described above is then only applied to the remaining indicators. The second problem, however, cannot be solved so easily. The following iterative algorithm is proposed and applied for DeLite:

- Solve the restricted regression.
- Determine all weights which are negative and remove the associated indicators from the regression model.
- Repeat all steps above as long as any negative weights are found.

As a result we get:

- Indicators not correlated to the user ratings do not contribute to the global readability score.
- Different indicators strongly correlated to each other are usually replaced by a single one in the readability formula.

Instead of removing indicators with negative weight, a further improvement of this method could be to remove those indicators which are most strongly correlated to the former ones, since very highly correlated (normalized) indicators are nearly exchangeable. In this way the square error can possibly be further reduced. In the worst case, however, the performance may exponentially grow with the number of indicators since in every step several alternative solution paths have to be followed.

7. Evaluation of the DeLite Readability Checker

In this section, the evaluation of the DeLite readability checker is described. The evaluation took place in a setting for which the parameters and weights were set according to the results of the machine learning techniques described in the last section. To compare the performance of the machine learning algorithms with human judgements, the mean absolute error (MAE) and the root mean square error (RMSE) were determined [GH62], which give also measures for the quality of the DeLite readability function. The weights of each indicator in the weighted sum were calculated by both robust and linear iterative regression.

Furthermore, the readability function developed in this work is compared with the Amstad Readability Index [Ams78], a German variant of the Flesch Reading ease score.

In order to determine parameters and weights for the readability function an online readability study was performed. In this study the participants rated several texts according to their readability on a seven point Likert scale [Lik32]. The text corpus, which has been used for learning, contained 515 texts mainly originating from the municipal domain. In total, the data consists of about 2800 readability judgments.

7.1. Evaluation Settings

315 users participated in the readability study, 43.1 % of them were female and 56.9 % male. 91.4 % of them were German native speakers. Four people were not native speakers and their German language skills were, according to their own judgment, worse than "Good". Since the aim of this experiment was to test the readability for German native speakers their ratings were filtered out. Readability experiments for non-native speakers were not carried out and left for future work. Almost 70 % of the participants were between 20 and 40 years old; the number of participants over 60 was very small (ca. 3 %). The participants were mainly well-educated. 58 % of them owned a university or college degree. There is none who had no school graduation at all. The participants of the evaluation belonged to a large variety of professions, e.g., software-developers, scientists, physicians, linguists, pharmacists, administrators, psychologists, and musicians.

7.2. Evaluation Results

Table 1 shows the weight for each indicator, determined by robust regression with linear optimization and by iterative linear regression. The average standard deviation

of these weights for a ten-fold cross-validation amounted to 0.006 for robust and 0.008 for iterative linear regression. Table 1 and 2 illustrate the influence of the individual linguistic levels on the total DeLite score.

Indicators are differentiated into surface-oriented and deep indicators. An indicator is considered as deep if it exploits semantic information or relies on the syntactic dependency tree. Concerning the MultiNet paradigm [Hel06], deep indicators employ semantic MultiNet-relations, ontological sorts or semantic concept information which are derived by a deep syntactico-semantic analysis.

Note that the fact of an indicator being surface-based or deep corresponds only roughly to the linguistic level of the indicator. For instance, the syntactic level, which consists mainly of deep indicators, contains the surface-oriented indicator *average sentence length*. The information whether or not an indicator is considered as deep is given in Table 8. The sum of weights for traditional surface-based indicators is 0.498 (0.412) while the indicators requiring a (deep) syntactico-semantic analysis reaches 0.502 (0.588) for absolute regression (iterative linear respectively). Thus, the latter type of indicators have a larger total weight in the DeLite readability formula. In general, only a small part of the 48 indicators is used to compute the readability score which is mainly caused by the fact that several indicators which are strongly correlated to each other and removed by the machine learning approach.

Table 3 shows MAE and RMSE of both validated machine learning methods, robust regression and iterative linear regression. The approximative iterative linear regression method leads to very good results in practice: It always yields a smaller RMSE than computing scores with the weights found by the robust regression algorithm. Furthermore, the RMSE and MAE for between normalized indicator and the average user ratings were determined (see Table 8).

The user ratings were also compared to the scores computed with the Amstad understandability index. The correlation between user ratings and the Amstad index scores amounts to 0.187. This relatively low correlation possibly shows that the Amstad index is not an adequate measure of text understandability. By using DeLite (correlation: 0.417) instead of Amstad, the correlation is increased and, in comparison, the MAE and the RMSE are considerably lower. These improvements are mainly due to a larger number of indicators and to indicators resulting from deep natural language processing methods, i.e., indicators on the semantic and discourse level. One has also to take into account that the Amstad index was developed primarily for non-domain-specific newspaper corpora.

Table 8 in Appendix C shows the mean absolute error (MAE) and the root mean square error (RMSE) between the normalized indicator values, which were calculated by DeLite, and the ratings of the test persons.

Note that the study was made with texts of a municipal domain. Other types of corpora (like newspapers, books or spoken texts) may lead to different results since each type of text corpus shows special linguistic phenomena. For instance, the texts investigated during the readability study contained long sentences with a lot of legal terms and compound words, while only comparatively few pronouns and negations are involved. Thus, the weights of the discourse indicators are rather low (see Table 1).

In Table 4 the runtimes of the machine learning algorithms to determine parameter and weights are displayed. Note, that these algorithms presume that all indicator values are already determined for the text corpus of the readability study, which was really done by DeLite in a batch mode. Also, the indicator values for all text segments have to be already calculated, which is done by the DeLite readability checker. Thus,

Table 1: Indicator weights determined by robust regression or iterative linear regression.

Indicator	Weight	
	Robust regression	Linear regression
Morphological Level		
Number of compound concepts	0.099	0.057
Number of syllables	0.085	0.047
Number of characters	0.020	0.021
Lexical Level		
Inverse lemma frequency	0.110	0.114
Word frequency	0.046	0.053
Syntactic Level		
Average number of words per phrase	0.060	0.099
Clause center embedding depth	0.000	0.006
Sentence length	0.101	0.141
Distance between verb and complement	0.017	0.038
Distance between verb and prefix	0.095	0.155
Distance between verb group parts	0.053	0.000
Passive	0.000	0.028
Semantic Level		
Number of propositions	0.019	0.066
Number of clusters of causal relations in a chain	0.008	0.005
Connections between discourse entities	0.029	0.018
(Double) negations	0.052	0.000
Quality of semantic network	0.184	0.122
Discourse Level		
Number of pronouns without antecedents	0.000	0.014
Number of reference candidates	0.022	0.015

Table 2: Weights of the individual levels determined by robust and iterative linear regression

Level	Weight	
	Robust regression	Iterative linear regression
Morphological Level	0.204	0.125
Lexical Level	0.156	0.167
Syntactic Level	0.325	0.467
Semantic Level	0.292	0.212
Discourse Level	0.022	0.029

Table 3: Mean absolute (MAE) and Root mean square errors using robust regression or iterative linear regression.

Method	MAE	RMSE
Robust regression evaluated on training data	0.127	0.157
Iterative linear regression evaluated on training data	0.126	0.159
Robust regression 10 fold cross-validation	0.130	0.165
Iterative linear regression, 10 fold cross-validation	0.131	0.161
Amstad Index	0.203	0.245

the output of Delite is used by the machine learning algorithms as input. A major part of the time consumed was needed to parse the results of the DeLite readability checker. The calculation was done on a computer with 1 GB memory and the single-core processor AMD Athlon^{tm} XP 2200+ using a clock frequency of 1.8 GHz. The iterative linear regression has been implemented in the programming language Scheme while the linear optimization was carried out by using a highly optimized mathematics library (glpk, see `http://www.gnu.org/software/glpk/`). Note that the runtimes are very short which is mainly due to avoiding non-linear optimization techniques.

Table 4: Runtime for the calculation of weights and parameters.

Algorithm	Runtime (sec.)
Determine weights (robust regression)	112
Determine weights (linear iterative regression)	126
Determine parameters for the normalization function	93

8. The English Prototype and the Language Model of DeLite

One goal of this research was to investigate into the transferability of the DeLite approach to other languages (especially English). It can be stated that, at least for European languages, many of the linguistic phenomena which make a text difficult to read are the same as in German. Thus, in other languages, a text is also usually more difficult to read if it contains both a lot of rare or long words and long sentences. Most of the semantic indicators are usable for other languages too, like the number of propositions per sentence, the number of negations, etc. Basically there are only minor changes, e.g., there exist no long compound words in English, where long compound noun phrases play a similar role (see *Lebensversicherungsgesellschaft* vs. *life insurance company*). In the next section, a brief introduction of the current DeLite system for English is given and it is shown how the indicators had to be adjusted to the English language. In traditional readability checkers only different weights are used to adapt the rating formulas to another language [Fle48, Ams78].

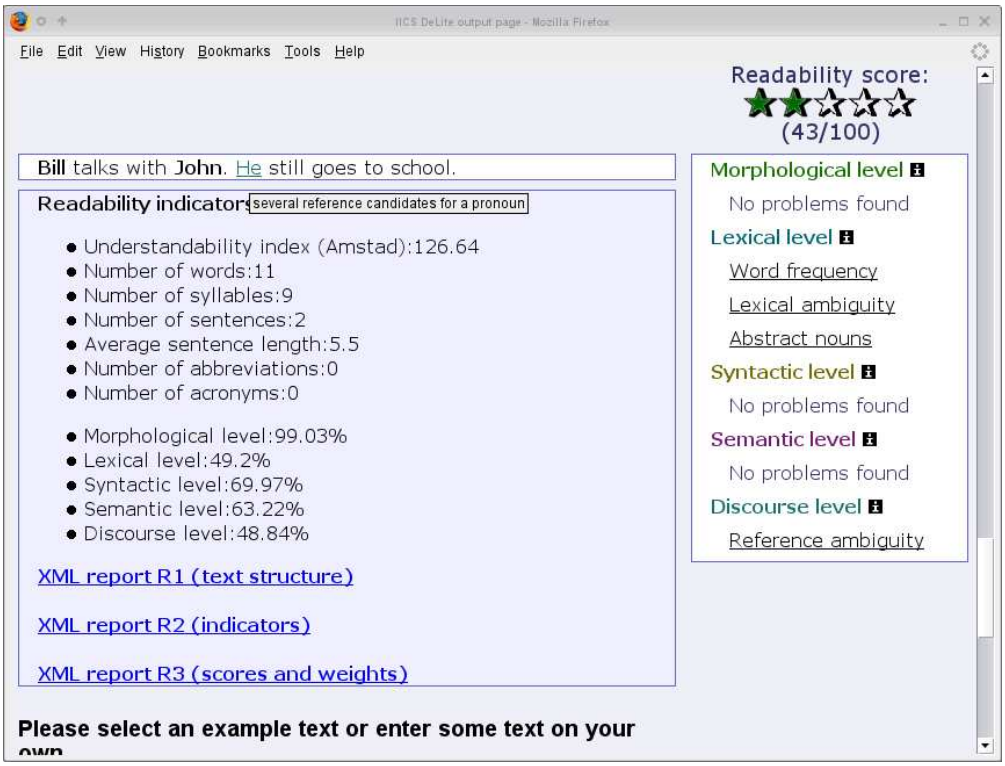

Figure 13: English analysis of two sentences where the latter contains a reference ambiguity.

8.1. Analyzing English Texts with DeLite

The user can select the language to be analyzed using a special checkbox of DeLite. Figure 13 shows an example of the analysis of an English text. The text consists of two sentences where the second sentence contains a pronoun ambiguity: *Peter talks with John. He still goes to school.* The second sentence is ambiguous since the pronoun *He* can either refer to *Peter* or to *John.* A better formulation would be *Peter talks to John, who still goes to school.* or alternatively, if the pronoun has to refer to *Peter*: *Peter, who still goes to school, talks to John.* DeLite highlights the pronoun *He* because of its ambiguity and displays its potential antecedents in bold face, analogously to the analysis of German sentences (see page 8).

8.2. Language Dependent Indicators

Some indicators are strongly language-dependent while others are not. Indicators of the second type constitute the largest part of the semantic indicators, like number of propositions per sentence, number of causal relations in a chain. Also the syntactic and surface type indicators like word or sentence length belong into this group. Indicators of the first type are for instance:

- Word and lemma frequency (frequency distribution is always language-dependent)
- Negations (negations are expressed by specific negation prefixes or expressions)
- Deverbal and deadjectival nouns
- Number of different word readings
- Synset size per word

In the following it is demonstrated how additional manual work for adjusting DeLite to different languages can be reduced if lexical resources (e.g., lexicon and knowledge base), which have to be prepared for the syntactico-semantic analysis anyway, are used systematically. Note that an in-depth description of all indicators can be found in Appendix A.

Word and Lemma Frequency: The inverse lemma frequency is determined for all lemmas appearing in the given text. If a lemma does not occur at all in the training corpus, its inverse lemma frequency is defined to be 1. Besides the lemma frequency, the word form frequency is employed too. All word forms are divided into several classes, where class 1 contains the most frequent and class 100 contains the least frequent words. All word forms occurring in a text which cannot be recognized at all are assigned to class 100 too. The associated frequency tables for word form and lemma frequency can be configured for each language separately.

Negations: Negations appear either in form of special words, like: *not*, *never*, *nowhere* or as negation prefixes (like *un-*). Note that in the latter case the negation prefix does not always invert the meaning, e.g., *unheimlich (weird)* is not the contrary of *heimlich (secret)*. Furthermore, the identification of a negation prefix of an adjective can be misleading, e.g., *unterirdisch* does not contain the negation prefix *un* but instead the prefix *unter*. In both cases the adjective should not be considered containing a negation. This problem is solved by using information from the semantic lexicon HaGenLex. Consider a concept w with a negation prefix n, i.e., $w = nv$. The concept w is considered to have a negative meaning if there exists an antonym relation in the lexicon between w and v.

Sentence negations can be recognized in many cases by checking if the facticity of the sentence node is associated to *nonreal* (the facticity of a concept node of a semantic network specifies whether this node represents a real, nonreal or hypothetical fact [Hel06, Chapt. 8]). Apart from the language dependent definition of negation prefixes, this approach is easy to transfer to other languages since it remains to ensure the knowledge base contains all required antonymy relations.

Deverbal and Deadjectival Nouns: A noun that is derived from a verb is considered to be more complex than the underlying verb itself. This indicator is also related to nominal style, which is (unfortunately) quite frequent in a lot of languages including German and English.
Examples: The word *discussion* is derived from *to discuss*), the word *nominalization* is derived from *nominalize*. The semantically oriented lexicon HaGenLex contains derivational information. In case an abstract noun is derived from a verb both underlying concepts are connected by the semantic relation CHEA (Change of sorts: Event - Abstract Concept). Similarly, if a noun is derived from an adjective, the underlying concepts are connected by the relation CHPA (Change of sorts: Property - Abstractum). Thus, for this indicator, we just need to test if for a given noun one of these relations is present. This approach is easy to transfer to other languages. One only has to assure that the relations CHEA and CHPA are specified for the concepts in the lexicon.

Number of Different Word Readings: A word can have several different readings, e.g., the German noun *Raum* can mean either *room* or *space*. The different word

Table 5: Important indicators ordered according to their weights in the readability formula.

Indicator	Subsection	Weight
Quality of the semantic network	9.3.1	0.183
Inverse lemma frequency	9.2.1	0.110
Average sentence length	9.3.2	0.101
Average distance between verb and prefix	9.3.4	0.094
Number of syllables	9.1.1	0.085
(Multiple) negations	9.4.3	0.051
Word frequency	9.2.1	0.046
Number of characters per word	9.1.1	0.020
Connections between network nodes	9.4.2	0.029
Number of reference candidates for a pronoun	9.5.1	0.021
Number of propositions per sentence	9.4.1	0.018
Average distance between verb and complement	9.3.3	0.017

readings are provided by the WOCADI parser, which itself retrieves this information from the lexicon for the specific language. So no adjustments are needed for DeLite itself.

Synset Size (per Word): Frequently, a concept appearing several times in a text is verbalized by different synonyms. This can degrade readability since the reader has to identify the different words with each other, i.e., he has to infer that those words all refer to the same concept. Thus, for each word the number of synonyms used earlier are counted. Information about synonymy is also provided by HaGenLex (MultiNet relation SYNO). Thus, no work on DeLite side has to be invested to compute this indicator for other languages.

In summary, adopting DeLite to another language mostly consists of adapting lexical resources and knowledge bases (see Figure 1) or in rewriting NLP tools like the WOCADI parser.

9. Recommendations for Ensuring Good Readability

One goal of the work on DeLite has been to derive recommendations for good readability and consequently also the accessibility of texts on the Web. The recommendations given here have been derived on the basis of the human user readability judgments gathered during the evaluation experiments. Table 5 shows the linguistic indicators ordered by their weights found by the robust regression evaluation. Thus, we recommend to judge the readability by the following criteria for English and German in that order. The indicators with the highest weights are described in the next subsections. Note that there exists also an in-depth description of all indicators in Appendix A.

9.1. Important Morphological Indicators

9.1.1. Number of Syllables and Characters per Word

The number of syllables/characters per word length (see Table 5) is employed as an indicator in many readability formulas [Kla63, Fle48]. This indicator is generally

accepted as being important for the judgment of readability. We expect that this indicator has lower weight for English since long compound nouns usually do not occur there. Therefore, for English, we propose to weight the number of nouns per NP stronger, which in corresponds to the higher weighting of long compounds in German.

9.2. Important Lexical Indicators

9.2.1. Word Frequency and Inverse Lemma Frequency

The frequency of words (see Table 5) is also used in various readability formulas (e.g., in the readability formula of Flesch [Fle48]). This indicator is based on the assumption that words which appear rarely are more difficult to understand. This indicator also penalizes misspelled words. We differentiate between word and lemma frequency. For the latter, a lemmatization of words appearing in the analyzed text is necessary.

9.3. Important Syntactic Indicators

9.3.1. Quality of the Semantic Network

The semantic network quality (see Table 5) including failing of the parsing process or recognition of chunks only turned out to be a reliable indicator for readability. The parse can fail because of syntactic complexity or violation of semantic constraints. This indicator was assigned the highest weight in the DeLite readability formula.

9.3.2. Average Sentence Length in Words

Almost all popular readability formulas use the average sentence length as an indicator of text readability (see Section 2). In the DeLite readability formula this indicator reaches a total weight of 0.10 which makes it the most important indicator next to semantic network quality. However, this indicator still has several drawbacks (see Section 9.4.1), which can be overcome by using additional semantic indicators. However, most of the semantic indicators can only be determined if the semantic network was successfully constructed. Thus, this makes them less robust than the indicator *average sentence length* which can be computed for any text.

9.3.3. Average Distance between Verb and Complements

A large distance between a verb and its complement can effect readability since the reader has to attach the complements to the far-off verb.
Example: Peter ***lädt*** *Herrn Meyer, den er gestern in München zufällig in der U-Bahn getroffen hatte und den er schon lange nicht mehr gesehen hatte, sowie* ***Petra zum Abendessen*** *ein. (Peter* ***invites*** *Mr. Meyer, whom he met by chance yesterday in the subway and whom he did not see for a long time, as well as* ***Petra for dinner****.)*, distance: 25 words between *lädt* und *Petra*, including comma and period.

9.3.4. Average Distance between Verb and Prefix

A large distance between a verb and its separable prefix can degrade readability since this effect makes it difficult for the reader to relate a verb and its prefix to each other. Note that this effect does not exist in English. Thus, this indicator is only calculated

for German.
*Example: Peter **lädt** Herrn Müller am Dienstag gegen 18.00 Uhr mit seiner Frau zum Abendessen **ein**. (Peter invites Mr. Müller with his wife at about six o'clock p. m. for dinner.)* Distance: 12 words between *lädt* and *ein*.

9.3.5. Clause Center Embedding Depth

A sentence can be quite difficult to understand if a sub-clause is embedded in the middle of the superior clause since the reader has to memorize the superior clause until it is continued after the termination of the subordinate clause. Not only human readers consider this as difficult. The WOCADI parser, too, failed in many cases to analyze such sentences correctly. It was observed that a large portion of incomplete parses (indicator *semantic network quality*, see Section 9.3.1) was caused by deeply embedded sentences.
Example: Er verließ das Haus, in dem die Frau, die er liebte, wohnte, sofort. (He left the house where the woman he loved lived immediately); depth of embedding: 2.

9.4. Important Semantic Indicators

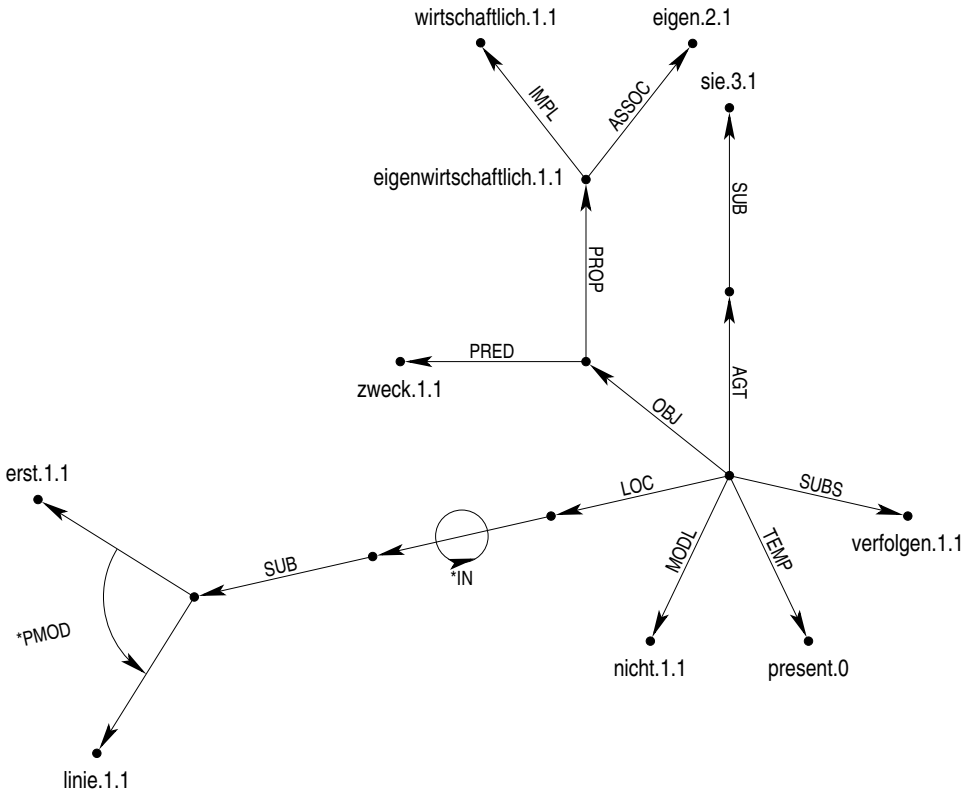

Figure 14: Semantic Network with a small number of connections between network nodes for the example sentence: *Sie verfolgen nicht in erster Linie eigenwirtschaftliche Zwecke. (They do not primarily pursue their own economical interests.)*

9.4.1. Number of Propositions per Sentence

The indicator *sentence length* has several drawbacks for estimating readability. For instance a sentence with a long item list is usually not difficult to understand [LvTT81]

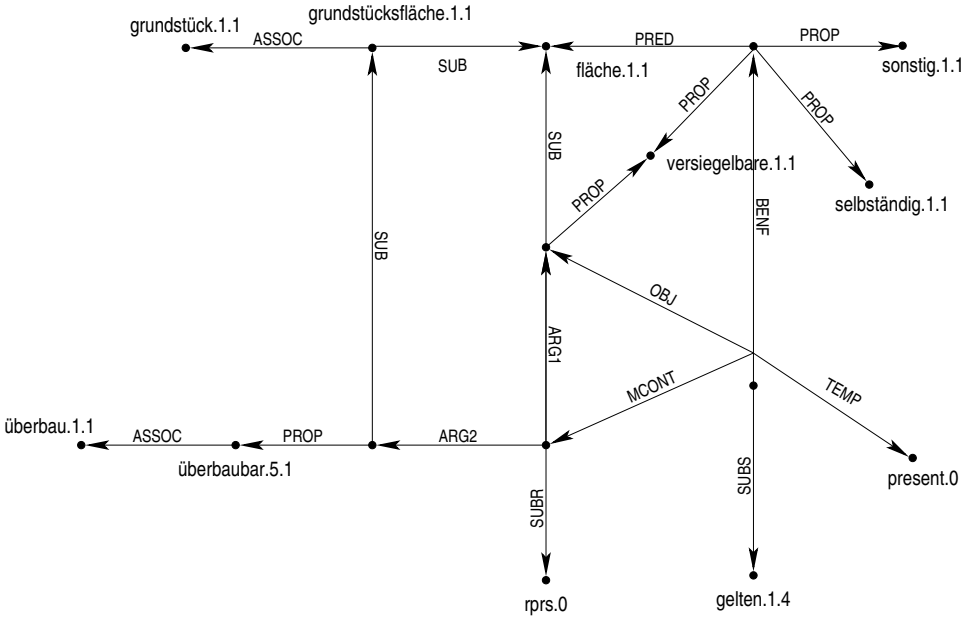

Figure 15: Semantic Network with a large number of connections between network nodes for the example sentence: *Für sonstige selbständige versiegelbare Flächen gilt die versiegelbare Fläche als überbaubare Grundstücksfläche. (For other autonomous sealable areas the sealable area is considered a land area which can be overbuilt.)*

although this sentence consists of a large number of words. However, these words are part of a single proposition. Therefore, the difficulty of such sentences can better be described by the number of propositions per sentence.
Example: *Anwesend waren Herr Müller, Dr. Peters, Herr Franck, ... (Mr. Müller, Dr. Peters, Mr. Franck, ... were present).* Number of propositions: 1

9.4.2. Connections between Network Nodes

The number of connections between network nodes which represent objects (in contrast to actions or properties) turned out to be quite important for assessing readability. A high average number of connections is often an indicator for complex dependencies between concepts.
Example:

- *Sie verfolgen nicht in erster Linie eigenwirtschaftliche Zwecke. (They do not primarily pursue their own economical interests.)* Indicator value: 0.625 (see Figure 14)
- *Für sonstige selbständige versiegelbare Flächen gilt die versiegelbare Fläche als überbaubare Grundstücksfläche. (For other autonomous sealable areas the sealable area is considered as land area which can be overbuilt.)* Indicator value: 1.43 (see Figure 15).

9.4.3. Multiple Negations

Negations in a text are to be avoided if possible [Gro92]. In particular, double or even triple negations make a sentence often quite difficult to understand. Usually such a sentence can be rephrased by dropping most of the negations.
Example: Tom glaubt nicht, dass Bill nicht denkt, dass der Film nicht sehr uninter-

essant war. (Tom does not think that Bill does not think that the movie was not very uninteresting.) This sentence contains four negations, including the negation morpheme *un*. A sentence without any negations and which expresses a similar meaning is for instance:
Tom glaubte eher, dass Bill dachte, dass der Film zumindest etwas interessant war. (Tom believed that Bill thought that the movie was at least a bit interesting.)

9.5. Important Discourse Indicators

9.5.1. Number of Reference Candidates

The antecedent of a pronoun should be uniquely determinable. The existence of more than one antecedent candidate in a text has to be avoided.
Example: Der Bürgermeister und der Pfarrer empfingen die neuen Kirchturmglocken. Anschließend wurden sie aufgehängt. (The mayor and the priest received the new church bells. Afterwards they were hung up.) The meaning of this sentence is ambiguous since the pronoun *they* can either relate to *Der Bürgermeister und der Pfarrer (The mayor and the priest)* or to *Kirchturmglocken (church bells)*. Thus, in the case the pronoun should actually relate to *Kirchturmglocken*, this sentence should better be reformulated to: *Der Bürgermeister und der Pfarrer empfingen die neuen Kirchturmglocken, die anschließend aufgehängt wurden. (The mayor and the priest received the new church bells which were hung up afterwards.)*

9.6. Conclusions for the Recommendation

For making a text better readable,

- avoid long sentences,
- use common and short words,
- make sure that the antecedent of a pronoun is unique,
- avoid multiple negations,
- do not use deeply embedded clauses,
- keep the number of propositions per sentence small, and
- keep the distance between verb and verb prefix small (only for German).

10. Conclusion and Further Work

The evaluation showed that the deep linguistic readability indicators (mainly semantic and syntactic indicators) had a total weight of more than 50 %. This means that this type of indicators are more important than surface oriented indicators like sentence or word length. The weights of the indicators were determined by a robust regression using linear optimization on ratings of 315 test persons. In addition, a method based on linear regression was tested.

By avoiding non-linear algorithms, which have various convergence problems caused mainly by the occurrence of local minima, a high performance of the learning algorithms was obtained. Thus, the weights and parameters of the DeLite readability function were calculated in less than ten minutes, assuming that the indicator values were already calculated.

Furthermore, the effort needed for transferring the readability checker DeLite to other languages than German were investigated. Exemplary, a prototype for the English language has been realized.

The current readability checker is not yet able to actively make suggestions for a better formulation of difficult-to-read text passages. Therefore, DeLite should be extended to become a real authoring tool, which provides such abilities. This could be an important step in the area of text simplification. The major modules for this task do already exist, whereas a good generation component is still lacking. Also, an additional readability study should be carried out with handicapped people which have cognitive impairments. It is generally proposed to adapt the DeLite system to the needs of special user groups. The learning algorithms needed for that do already exist.

Future work could also comprise cover other aspects as the reconsideration of a logical level, which checks whether the text is logically consistent or whether causal/concessive clauses are comprehensible from a logical point of view. For that, an automatic reasoner has to be integrated into DeLite. The preconditions for this work are created in the DFG project LogAnswer[5].

Since the evaluation showed the necessity of including deep semantic methods into DeLite, we believe that semantic and logic readability indicators will play an important part for future readability checkers.

[5]Contract Number: HE 2847/10-1

A. Readability Indicators

A.1. Description of Indicators

Each indicator realized in the DeLite system is systematically described by the general schema below. The full name of the indicator is the subsection name. The indicator description schema has the following general form:

A Symbolic name: This element is a formal name that can also be used in implementations.

B Related criterion: Here, the readability criterion that is related to the indicator is given (see Section 4).

C Indicator definition: This schema element defines how the indicator values are calculated. Indicator values are always non-negative numbers.

D Relevant parser attributes/relations: The schema element lists the attributes that are needed from the parser output. The mentioned attributes are formally defined in Appendix B. If an attribute is shown in parentheses, the information is collected from other resources (and not from the parser output). Also, the relevant MultiNet relations (or functions) are given which are used by the indicator.

E Type of text segment the indicator operates on. The segment type can be either *word*, *sentence*, *phrase* or *document*.

F Value range of this indicator, can be binary (0 or 1), integer or float

G Examples: Natural language examples illustrating the determination of the intended indicator. If the example is a clear violation of the underlying readability criterion with respect to the given indicator it is marked by a leading exclamation mark (!).

H Discussion: The optional discussion element contains issues that might be controversial or open to alternative solutions. In the following indicator descriptions, similar indicators are grouped together so that *only* 35 descriptions are needed for all 48 indicators of the implemented readability criteria. Most indicators can be counted on a per-sentence or per-word basis.

A.2. Morphological Level

A.2.1. Indicator: derived noun

A Symbolic name: a.) is-deverbal-noun, b.) is-deadjectival-noun

B Related criterion: derivation

C Indicator definition: A noun that is derived from a verb is typically more complex than the underlying verb itself. This indicator is also related to nominal style, which is (unfortunately) quite frequent in German. A noun that is derived from an adjective can also be hard to understand. As there are different derivation morphemes, some derivations might be easier to understand than others.

D Relevant MultiNet relations: CHEA, CHPA

E Associated text segment type: word

F Value range: binary

G Examples:

- *discussion* (from *to discuss*)
- *nominalization* (from *nominalize*)
- *Den **Spekul**anten interessierte seine **Heul**erei nicht. (The scalper was not interested in hearing him crying.)* Value: 1 for the deverbal nouns *Speku-*

lanten and *Heulerei*; verb stem parts are marked by bold font style.
- *cleverness* (from *clever*), *!Unverständlichkeit* (*incomprehensibility*), value: 1 for the deadjectival nouns *cleverness* and *!Unverständlichkeit*

H Maybe a more fine-grained classification is needed because some derivations are clearly easier to understand than others. The aggregated indicator value, calculated by averaging (see Section 5.1), specifies the relative frequency of a noun being derived from a verb (or adjective respectively) and ranges from 0 to 1, i.e., the value range after aggregation is float and not binary.

A.2.2. Indicator: number of compound simplicia

A Symbolic name: num-compound-simplicia
B Related criterion: compound complexity
C Indicator definition: The indicator counts the number of simplicia (component words) of a compound.
D Relevant parser attributes/relations: analysis-compounds, (lexicon)
E Associated text segment type: word
F Value range: integer
G Examples:
- *!Mehrwertsteuererhöhungsdiskussion (value-added tax increase discussion)*, value: 5
- *Steuerdiskussion (tax discussion)*, value: 2
- *Stubentiger (cat)*, value: 2

H Discussion: As the number of words in a compound is a better estimate for cognitive load for most people, an indicator counting the number of compound morphemes has been excluded from the system.

A.2.3. Indicator: number of compound concepts

A Symbolic name: num-compound-concepts
B Related criterion: compound complexity
C Indicator definition: This indicator counts the number of concepts involved in the semantics of a compound. In contrast to the indicator num-compound-simplicia, it does not decompose compounds with irregular semantics. For example, the German noun *Stubentiger* (literally: *room tiger*) does not denote a tiger in a room. Therefore, this noun refers to only one concept and not to the two concepts *Tiger (tiger)* and *Stube (room)*.
D Relevant parser attributes: analysis-compounds
E Associated text segment type: word
F Value range: integer
G Examples:
- *Bundestagsdiskussion (parliament discussion)*, value: 2
- *!Mehrwertsteuererhöhungsdiskussion (value-added tax increase discussion)* value: 4
- *Stubentiger (cat)*: Value: 1

A.2.4. Indicator: types/token ratio

A Symbolic name: abbreviation-type-token-ratio
B Related criterion: abbreviations and acronyms

C Indicator definition: An abbreviation typically involves periods, for example *e.g.*, *i.e.*, *vs.* (in German: *z. B.*, *etc.*)
D Relevant parser attributes: analysis-words
E Associated text segment type: document
F Value range: float
G Examples: A text that contains *e.g.* three times and *vs.* four times, but no other abbreviations. Value: $2/7 \approx 0.29$

A.2.5. Indicator: acronym type/token ratio

A Symbolic name: acronym-type-token-ratio
B Related criterion: abbreviations and acronyms
C Indicator definition: An acronym is an artificial word made up of the initial letters from the long form: *USA*, *UNESCO*, *WCAG* [Tho95].
D Relevant parser attributes: analysis-words
E Associated text segment type: document
F Value range: float
G Examples: A text that contains *UNO* two times and *UNESCO* one time (value: $2/3 \approx 0.67$).

A.2.6. Indicator: number of syllables/characters (per word)

A Symbolic name: a.) num-syllables, b.) num-characters
B Related criterion: word length
C Indicator definition: The first indicator is the number of syllables of a word. The second indicator simply counts characters of the written word.
D Relevant parser attributes: analysis-words
E Associated text segment type: word
F Value range: integer
G Examples:
- *Steuererhöhungsdiskussion (tax increase discussion)*, values: num-syllables: 8, num-characters: 25
- *Buch (book)*, values: num-syllables: 1, num-characters: 4

A.3. Lexical Indicators

A.3.1. Indicator: word frequency

A Symbolic name: frequency-class/ inverse-lemma-frequency
B Related criterion: word frequency
C Indicator definition: The frequency class of a word form is calculated from text corpus statistics. Instead of absolute or relative frequency, the frequency class is used in order to reduce the size of number representations (typical word form lists contain several million entries for German). Currently, there exist 100 frequency classes, where class 1 represents the highest frequency and class 100 the lowest frequency. The frequency is also calculated on lemma instead of word form counts. The inverse lemma frequency is defined as the reciprocal of the lemma frequency but at most 1. This means that in the case a lemma does not appear at all, its inverse lemma frequency is defined to be 1.
D Relevant parser attributes: analysis-words, parse-reading
E Associated text segment type: word

F Value range: integer (word frequency), float (inverse lemma frequency)
G Examples:
- *!Steuererhöhungsdiskussionsende (tax increase discussion end)*, value for frequency class: 100, inverse lemma frequency: 1
- *Steuer (tax)*, value for frequency class: 8, inverse lemma frequency ≈ 0

A.3.2. Indicator: number of lexical readings from lookup

A Symbolic name: num-readings-from-lookup
B Related criterion: lexical ambiguity
C Indicator definition: Each word form in a given text is assigned one or more readings by the morpho-lexical analysis of WOCADI. The number of different readings characterizes the lexical ambiguity.
D Relevant parser attributes: analysis-readings
E Associated text segment type: word
F Value range: integer
G Examples:
- *Haus*, senses: *building*, *company location*, value: 2
- *!Raum*, senses: *room*, *space*, *area*, value: 3

H Discussion: Lexical ambiguities are much more problematic for computers than for humans.

A.3.3. Indicator: number of lexical readings from parse

A Symbolic name: num-readings-from-parse
B Related criterion: lexical ambiguity
C Indicator definition: A word form in a given text is assigned one or more readings by WOCADI. In contrast to the preceding indicator (number of lexical readings from lookup), all steps of the parser (not just the initial morpho-lexical analysis) are applied so that the context often leads to exactly one reading per word form. But in some cases, several alternative parses remain with equal scores assigned by the parser so that also different readings of a word form can be left over after parsing.
D Relevant parser attributes: analysis-parse-readings, analysis-alternatives-enet
E Associated text segment type: word
F Value range: integer
G Examples: *Haus (house)*, value: 1, if the best parse and equally scoring parses contain only one of the two readings of the noun *Haus*.

A.3.4. Indicator: synset size (per word)

A Symbolic name: synset-size
B Related criterion: naming consistency
C Indicator definition: The indicator counts the number of synonyms in a synset which occur in the given text. It is only defined if at least two synset elements occur and if they are possibly referring to the same entity. The latter condition has been added to reduce the number of false warnings. It illustrates the interesting fact that indicators on lower levels (here: the lexical level) can profit from information coming from higher levels (here: the text semantic level).
D Relevant parser attributes: analysis-parse-readings, coref-pairs, (synsets from the lexical resources), relevant MultiNet relation: SYNO

E Associated text segment type: word
F Value range: integer
G Examples:
- *Der [Weltraum]$_i$ ist groß. Es befinden sich viele Sterne im [All]$_i$. (The [universe]$_i$ is large. There are a lot of stars in the [space]$_i$.)*
- *![Der Bundestag]$_i$ verabschiedete das Gesetz. [Das Parlament]$_i$ debattierte nicht. (The [German Bundestag]$_i$ passed the law. The [parliament]$_i$ did not debate.)*, value for *Parlament*: 2.

H Discussion: From a stylistic view, variation of words along synonymy relations is often regarded as positive. But from a readability perspective (and from a cohesion perspective as part of coherence), the opposite is true.

A.3.5. Indicator: abstract/concrete concept

A Symbolic name: is-abstract-concept
B Related criterion: lexical abstractness
C Indicator definition: The first indicator determines if a noun is abstract or not which is done by examining the semantic sorts (defined in the semantic network formalism MultiNet) of that noun. An abstract noun belongs to the ontological sort: abstract objects (excluding well-known, measurable attributes like *size* and *height*). If a word has several meaning facets (such words are called semantic molecules or families), all facets must fulfill this criterion. Concrete nouns are nouns that are not abstract.
D Relevant parser attributes: parse-net (only sort information)
E Associated text segment type: word
F Value range: binary
G Examples:
- *Die Frau kaufte zwei Dampfmaschinen. (The woman bought two steam engines.)*, value for *Dampfmaschinen (steam engines)*: 0
- *Philosophie besteht aus vielen Unterdisziplinen. (Philosophy knows many subdisciplines.)*, value for *Philosophie (philosophy)*: 1

H Discussion: The aggregated indicator value, determined by averaging (see Section 5.1), specifies the relative frequency of a rare noun being classified as abstract, and ranges from 0 to 1, i.e., the value range of the aggregated indicator is float and not binary. It is planned to combined this indicator with the word frequency for future work. Thus, only rare abstract nouns would be highlighted as problematic.

A.3.6. Indicator: number of lexical tokens/types

A Symbolic name: a.) lemma-type-token-ratio b.) wordform-type-token-ratio
B Related criterion: vocabulary complexity
C Indicator definition: This indicator determines the ratio between the number of lemma tokens (i.e., text occurrences of lemmata) and lemma types (i.e., the number of different lemmas). This indicator is calculated analogously for word forms.
D Relevant parser attributes: net (= semantic network), analysis-lemmata, analysis-parse-readings
E Associated text segment type: document
F Value range: float

G Example: *Der junge Mann traf zwei alte Männer. (The young man meets two old men.)*, values: number of tokens: 8 (including the period), number of (lemma) types: 7, number of wordform types: 8, lemma-type-token-ratio: $7/8 = 0.875$, wordform-type-token-ratio: $8/8 = 1$
H Discussion: The ratio of number of types and tokens is a measure of vocabulary richness, which is often normalized by text length.

A.4. Syntactic Indicators

A.4.1. Indicator: number of complement ambiguities

A Symbolic name: num-complement-ambiguities
B Related criterion: syntactic ambiguity
C Indicator definition: If a complement position of a head word (verb, noun, or adjective) can be filled by more than one constituent and the resulting parses are equally good, then this complement is counted under this indicator.
D Relevant parser attributes: net, analysis-alternatives-enet
E Associated text segment type: word
F Value range: integer
G Examples:
- *Die Firma schickt Hansen Müller. (The company sends Hansen to Müller (or Müller to Hansen).)*, value for *schickt*: 2. In German, *Hansen* can be the direct object and *Müller* the indirect object or vice versa.)
- *Tiere treten Menschen. (Animals kick humans.)* or *(Humans kick animals.)* Value for *treten*: 2. In German — due to its freer constituent order — *Tiere* can be the subject and *Menschen* the object or vice versa, although humans prefer the first reading, which shows unmarked constituent order.

A.4.2. Indicator: number of PP attachment candidates

A Symbolic name: num-pp-attachment-candidates
B Related criterion: syntactic ambiguity
C Indicator definition: The indicator num-pp-attachment-candidates counts for a prepositional phrase (PP) the number of attachment candidates that are still possible after parsing.
D Relevant parser attributes: net, analysis-alternatives-enet
E Associated segment type: phrase
F Value range: integer
G Example: *Der Junge sah den Mann mit dem Fernrohr. (The boy saw the man with the telescope.)* Value for *mit dem Fernrohr (with the telescope)* is 2 since the PP *mit dem Teleskop (with the telescope)* can either be attached to the NP *den Mann (the man)* or directly to the verb phrase.

A.4.3. Indicator: number of dependents per verb/NP

A Symbolic name: num-dependents-per-verb, num-dependents-per-np
B Related criterion: syntactic complexity
C Indicator definition: The indicator num-dependents counts the direct dependents of a verb or an NP. According to the underlying dependency grammar, dependents can be complements and adjuncts.
D Relevant parser attributes: dep-tree

E Associated segment type: word for the indicator *num-dependents-per-verb*, phrase for *num-dependents-per-np*
F Value range: integer
G Examples:
- *Die Gruppe traf die Politiker aus Spanien im August. (The group met the politicians from Spain in August.)* Value for *treffen (met)*: 3, value for *Politiker (politicians)*: 1
- *das Buch des Klosters von 1990 über seine Geschichte (the book of the monastery from 1990 about its history)*, value for *Buch (book)*: 3
- *Die Reise 1985 von Italien nach Österreich über schmale Straßen zum besseren Verständnis kultureller Unterschiede (the trip in 1985 from Italy to Austria along small roads for better understanding of cultural differences)*, value for *Reise (trip)*: 5

H Discussion: Verbal heads typically can take more dependents without causing reading problems, while the limits for nominal heads are lower.

A.4.4. Indicator: number of constituents per coordination

A Symbolic name: num-constituents-per-coordination
B Related criterion: syntactic complexity
C Indicator definition: The indicator counts the conjuncts or disjuncts in a coordination.
D Relevant parser attributes: dep-tree
E Associated segment type: phrase
F Value range: integer
G Examples:
- *Äpfel und Orangen (apples and oranges)*, value: 2
- *Rom, Venedig, Wien, München, Bonn, Berlin, Düsseldorf und Hamburg sind die Städte, die das Komitee besuchen wird. (Rome, Venice, Vienna, Munich, Bonn, Berlin, Düsseldorf, and Hamburg are the cities that the committee will visit.)* Value: 8

H Discussion: The coordination's position in a sentence influences readability in some cases. For example, long enumerations should go to the end of the sentence.

A.4.5. Indicator: number of NP words

A Symbolic name: num-words
B Related criterion: syntactic complexity
C Indicator definition: The indicator num-words counts the number of words that belong to a noun phrase. As currently only constituents exceeding a fixed threshold will be reported and no averages will be calculated, it suffices to calculate this indicator for maximal NPs only.
D Relevant parser attributes: parse-net
E Associated segment type: phrase
F Value range: integer

G Examples:
- *der kleine Junge aus der alten Stadt (the little boy from the old town)*, value: 7
- *der sehr alte Mann mit der Pfeife, der die Straße von London herunter kam. (the very old man with a pipe that came down the road from London)*, value: 14

H Discussion: PPs are not handled separately because a PP is already covered by the NP that is headed by the PP's preposition.

A.4.6. Indicator: number of sentence words/constituents

A Symbolic name: a.) num-words, b.) num-sentence-constituents
B Related criterion: sentence length
C Indicator definition: The first indicator (num-sentence-words) measures the sentence length in a traditional way: by counting its words (including punctuation marks). The second indicator (num-sentence-constituents) counts constituents (maximal phrases of type v, np or pp) instead.
D Relevant parser attributes: analysis-words, parse-net
E Associated segment type: sentence
F Value range: integer
G Example: *Der Regen fiel am Montag. (The rain fell on Monday.)* Values: num-words: 6, num-sentence-constituents: 3

A.4.7. Indicator: distance between verb and complement/adjunct/prefix

A Symbolic name: a.) distance-verb-complement, b.)distance-verb-adjunct, c.) distance-verb-prefix
B Related criterion: linear precedence complexity
C Indicator definition: The first indicator (distance-verb-complement) measures the smallest distance (in words) between the main verb and a given complement. The second indicator does the same for adjuncts of the verb. The third indicator is only relevant for verbs with a separable prefix where the prefix is actually split off. Punctuation marks count as words, as usual in DeLite.
D Relevant parser attributes: dep-tree, parse-net
E Associated sentence type: word
F Value range: integer
G Example:
- *Das Pferd sprang über den Zaun. (The horse jumped over the fence.)*, value for the complement headed by *Pferd (horse)*: 0, value for the PP adjunct headed by *über (over)*: distance-verb-adjunct: 0
- *Das Kind **lachte** den Freund **aus**. (The child laughed at the friend.)* Value for the verb prefix *aus*: distance-verb-prefix: 2
- *!Peter **lädt** König Ludwig, den er gestern in Köln kennengelernt hatte, zum Abendessen in sein neues Haus **ein**.* Value for verb prefix: distance-verb-prefix: 17
- *Klaus ärgerte sich über Kurts Verhalten enorm. (Klaus was very angry about Kurt's behavior)*, value for *distance-verb-adjunct*: distance-verb-prefix: 4

A.4.8. Indicator: distance between verb group parts

A Symbolic name: distance-verb-group-parts

B Related criterion: linear precedence complexity
C Indicator definition: The verb group can be separated in two parts in German. This indicator counts the number of words between the two parts. The indicator value is attached to the verb which allows for the investigation of more than one verb of the same sentence.
D Relevant parser attributes: dep-tree, parse-net
E Associated sentence type: word
F Value range: integer
G Examples:
- *Das Haus stürzt ein. (The house collapsed.)* Value: 0
- *Das Haus ist gestern eingestürzt. (Yesterday, the house collapsed.)* Value for the German example: 1
- *!Das Haus **ist** am gestrigen Sonntag nach lang anhaltenden Regenschauern, die aus dem Norden des Landes hereingezogen waren, unter großem Getöse **eingestürzt**. (Yesterday (on Sunday), the house collapsed with an enormous rumble after long and heavy rains which had come from the north of the country.)* Value for the German example: 20
- *Where **did** he **go**?* Value: 1

A.4.9. Indicator: passive

A Symbolic name: is-passive
B Related Criterion: Passive form
C Indicator definition: This indicator is calculated for every verb. It is assigned to 1, if the verb is in passive voice, a semantic subject is present and its semantic role is AGT. Otherwise this indicator is assigned to 0.
D Relevant parser attributes: v-gend, relevant MultiNet relations: AGT
E Associated segment type: word
F Value range: binary
G Examples:
- *!Peter wurde vom großen Mann erschossen. (Peter was shot by the big man.)* Value: 1
- *Peter wurde erschossen. (Peter was shot.)* Value: 0
- *Der große Mann erschoss Peter. (The big man shot Peter.)* Value: 0

H Discussion: A passive formulation is usually more difficult to understand. However, there are some exceptions from the rule, e.g., when the semantic subject is missing. The passive sentence *Der Mann wurde erschossen. (The man was shot.)* is not worse readable than the associated active formulation *Jemand erschoss den Mann. (Someone shot the man.)* Also, sometimes the semantic agent is present but is not executing some form of action and still the passive construction is preferred, e.g., the passive formulation *Der Mann wurde vom Baum erschlagen. (The man was struck by a falling tree.)* is not worse readably than *Der Baum erschlug den Mann. (A falling tree struck the man.)* Note that the question if passive or active voice should be used, is often quite difficult to answer. More sophisticated approaches are preferable for future work.

A.4.10. Indicator: clause embedding depth

A Symbolic name: clause-embedding-depth
B Related criterion: embedding depth

C Indicator definition: A sentence can contain main clauses and subclauses. The latter can be nested. This indicator measures how deeply subclauses are embedded.
D Relevant parser attributes: dep-tree
E Associated segment type: word
F Value range: integer
G Examples:
- *Der Zug verließ die Station, die gerade renoviert wurde. (The train left the station, which was recently renovated.)*, value for *renoviert (renovated)*: 1
- *!Der Präsident eröffnete eine Behörde, die dafür gedacht war, die Geldströme, die zwischen verschiedenen Staaten auftreten, zu kontrollieren. (The president opened an agency that was intended to control the financial flows that occur between different countries.)*, value for *auftreten (occur)*: 3

A.4.11. Indicator: clause center embedding depth

A Symbolic name: clause-center-embedding-depth
B Related criterion: embedding depth
C Indicator definition: This indicator concentrates on special cases of clause embedding (see previous indicator) where a clause is in the middle (and not at the beginning or the end) of a higher clause. Such clauses are usually harder to understand than embedded clauses at the border since the reader has to memorize the interrupted sentence until it is continued after the termination of the subordinate clause.
D Relevant parser attributes: dep-tree
E Associated segment type: word
F Value range: integer
G Examples:
- *Der Mann, der die Abhandlung geschrieben hat, wusste nicht genug über das Thema. (The man that wrote the essay did not know enough about the topic.)*, value for *(geschrieben) hat (wrote)*: 0.
- *Die Frau, die er liebte, tötete ihn. (The woman who he loved killed him.)*, value for *liebte (loved)*: 1
- *Er glaubte der Geschichte, die ihm seine Frau, die erst nach Mitternacht nach Hause kam, erzählte, nicht. (He did not believe the story his wife, who came home after midnight, told him.)* Value for *kam*: 2, vaue for *came*: 1
- *Er verließ das Haus, in dem die Frau, die er liebte, wohnte, sofort. (He left the house where the woman he loved lived immediately.)*, value for *liebte (loved)*: 2

H Discussion: Some languages allow center-embedding in more positions than others, e.g. in adjective phrases used as attributes of nouns (for example, in German but not in English).

A.5. Semantic Indicators

A.5.1. Indicator: quality of the semantic network

A Symbolic name: sn-quality
B Related criterion: semantic complexity
C Indicator definition: quality of the semantic network. Three cases are differentiated:

1. Semantic network construction failed - Indicator value: 2
2. Only chunks could be constructed - Indicator value: 1
3. Full Parse - Indicator value: 0

D Relevant parser attributes: net, analysis-quality, relevant MultiNet relations: TUPL*

E Associated segment type: sentence

F Value range: integer

G Examples:

- *Der Junge geht in die Schule. (The boy goes to school.)*: Full Parse, value: 0
- *Der Junge gehrt in die Schule. (The boy gors to school.)*: This sentence is misspelled. Thus, only a chunk parse is available. Value: 1
- *!L'enfant va a l'école.* The parse failed since French is not supported by DeLite. Value: 2
- *Die Birne isst den Apfel. (The pie eats the apple.)* The parse failed since the semantic constraint that the subject of *essen (eat)* has to be human is violated. Value: 2

H Discussion: This indicator is both semantic and syntactic. This means it belongs to the list of semantic indicators too. However, in order to avoid redundancy it is only listed here.

A.5.2. Indicator: number of propositions per sentence

A Symbolic name: num-propositions-per-sentence

B Related criterion: semantic complexity

C Indicator definition: A semantic network node of the semantic sort *situation* (or a subsort) corresponds to a proposition[Hel06]. Counting such nodes leads to the indicator value. Abstracted situations (i.e., nominalized verbs) could also be seen as representing a proposition if complements are uttered too, e.g., *the discussion of the parliament about taxes*. But at the moment, such cases are ignored.

D Relevant parser attributes: net (= semantic network)

E Associated segment type: sentence

F Value range: integer

G Examples:

- *Der Mann stolpert über einen Stein. (The man stumbles over a stone.)* Value: 1
- *Die Gruppe organisierte einen Ausflug zu einem Ort, den keiner kannte. (The group organized a trip to a place that was new to all of them.)* Value: 2
- *!Während die Kinder spielten, rannten die Pferde umher, da der Zug einen fürchterlichen Lärm machte, als er vorbeifuhr. (While the children played, the horses ran around because the train made a terrible noise when it passed the area.)* Value: 4

H Discussion: This semantic indicator is more psycholinguistically motivated than syntactic variants like counting main clauses and subclauses per sentence because propositions might appear in different syntactic forms and disguises. For example in German, these can be participle clauses like *die von den angetrunkenen Brüdern vorgenommene Handlung* (word by word: *the by the drunken brothers committed actions*) and adjective clauses like *das im Winter dunkle Haus* (word by word: *the in the winter dark house*). The case of abstracted situations (see indicator definition) will need further consideration.

A.5.3. Indicator: number of relations in a cluster

A Symbolic name: (a.) num-reas-relations (b.) num-reas-clusters, (c.) max-reas-cluster-size
B Related criterion: semantic complexity
C Indicator definition:
  a) Number of occurrences of relations of the types REAS, CAUS, JUST or CONC
  b) The number of subgraphs with relations of the type REAS, CAUS, JUST and CONC
  c) Maximum cluster size in network nodes which consists only of relations of type REAS, CAUS, JUST and CONC. This indicator follows the assumption that a sentence is difficult to understand if it contains a long chain (path) of causal relationships.
D Relevant parser attributes: net (= semantic network), relevant MulitNet relations: REAS, CAUS, JUST, CONC
E Associated segment type: sentence
F Value range: integer
G Examples:
  - *Weil es keinen Kuchen gab, ging er nicht zur Fete. (Because there were no cake, he did not go to the party.)*
    (a.) num-reas-relations: 1 (relation: REAS)
    (b.) num-reas-clusters: 1 (only one relation of REAS, CAUS, JUST and CONC, hence only one cluster)
    (c.) max-reas-cluster-size: 2 (contains both concepts which are connected by the REAS edge)

A.5.4. Indicator: number of concept nodes per sentence

A Symbolic name: num-concept-nodes
B Related criterion: semantic complexity
C Indicator definition: This indicator counts the conceptual nodes (discourse entities) in the semantic network for a given sentence.
D Relevant parser attributes: net
E Associated segment type: sentence
F Value range: integer
G Examples: *Das Team gewann gegen den Champion. (The team won against the champion.)* Value: 3

A.5.5. Indicator: negations

A Symbolic names: (a.) num-negated-concepts, (b.) num-negated-adjectives, (c.) num-negations
B Related criterion: Negations
C Indicator definition
  (a.) The indicator *num-negated-concepts* counts the number of negated concepts in a sentence.
  (b.) The indicator *num-negated-adjectives* counts the number of adjectives that were derived by other adjectives by adding a negation prefix like *un-*. Examples are *unglücklich (unhappy)*, *unmöglich (impossible)*, *illegal (illegal)* etc.

(c.) The indicator *num-negations* deals with the special phenomena of double (multiple) negations. The number of negations are counted which can possibly cancel each other out. For that, certain relations in the MultiNet graph are followed which are relevant for this phenomenon, i.e., the length of the largest negation chain is counted. Thus, negations which are independent of each other do not increase the value of this indicator.

D Relevant MulitNet relations: ANTO, MODL, MCONT
E Associated segment type: sentence
F Value range: integer
G Examples:
- *Der ungerechte Paul und die uninteressante Laura gehen nicht in die Schule (The unfair Paul and the uninteresting Laura do not go to school.)*, indicator values: (a.): 1, (b.): 2, (c.): 2
- *Ich gehe nicht in die Schule, weil ich nichts lernen will.* (word for word: I do not go to school because I want to learn nothing.) Values: (a.): 2, (b.): 0, (c.): 2
- *!Ich glaube nicht, dass der Mann nicht extrem uninteressant ist. (I do not think that the man is not extremely uninteresting.)* Values: (a.): 2, (b.): 1, (c.): 3

H Discussion: The indicator *num-negations* tries to recognize all kinds of multiple (double, triple, quadruple, etc.) negations. A double negation is a phenomenon describing an expression which is two times negated. In the German language such expressions have usually a (weakened) positive meaning.

A.5.6. Indicator: connections between network nodes

A Symbolic names: num-connections
B Related criterion: Semantic dependencies
C Indicator definition: counts the average number of connections between network nodes. Only nodes are regarded which have the semantic sort *object*.
D Relevant MultiNet relations: all, relevant semantic sorts: o
E Associated segment type: sentence
F Value range: float
G Examples see Section 9.4.2

A.5.7. Indicator: maximum path length

A Symbolic names: max-path-length, max-path-length-sym
B Related criterion: semantic dependencies
C Indicator definition: This indicator counts the length of the longest path occurring in the semantic network. There exist two versions of this indicator. The first version takes into account the direction of the MultiNet edges while the second (max-path-length-sym) ignores them, i.e., in the latter case all relations are treated as symmetric.
D Relevant MultiNet relations: all
E Associated segment type: sentence
F Value range: integer
G Examples:
- *Ich besuche meine Schwiegermutter, meinen Onkel, meine Schwester und meine Cousine. (I visit the mother-in-law, the uncle, the sister, and the*

cousin.) Value for max-path-length: 4, value for max-path-length-sym: 5 (see Figure 16)

- *Ich besuche die Schwiegermutter des Onkels der Schwester meiner Cousine. (I visit the mother-in-law of the uncle of my cousin's sister.)* Value for max-path-length: 6, max-path-length-sym: 8 (see Figure 17)

H Discussion: For the first example, a parallel interpretation is possible since the complements are independent from each other. Therefore, the dependency chain is rather short. However, this is not the case for the second example. In this case, the complements depend on each other and have to be interpreted sequentially, which leads to a large dependency chain.

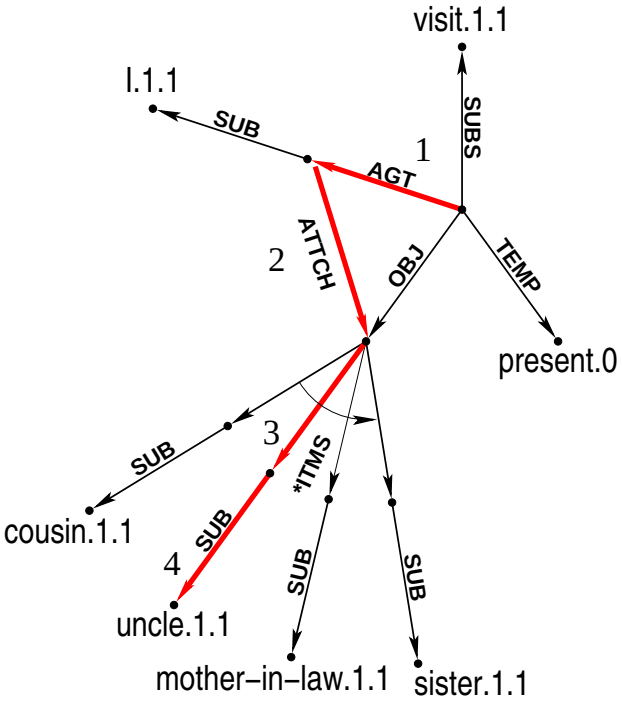

Figure 16: Semantic Network for the sentence: *Ich besuche meinen Vetter, meinen Onkel, meine Schwiegermutter und meine Schwester. (I visit my mother-in-law, my uncle, my sister, and my cousin.)* The longest path, taking into account the direction of the arcs, is emphasized by printing the associated arcs in bold face.

A.6. Discourse Indicators

A.6.1. Indicator: number of introduced concepts per sentence

A Symbolic name: num-introduced-concepts
B Related criterion: discourse coherence
C Indicator definition: The indicator counts the conceptual nodes (discourse entities) that are newly introduced in a sentence.
D Relevant parser attributes: net, parse-net
E Associated segment type: sentence
F Value range: integer
G Examples:

- *Ein Mann betrat einen Laden. (A man visited a shop.)* Value: 2
- *!Ein Mann betrat einen Laden in einer lauten Straße mit einem seltsamen*

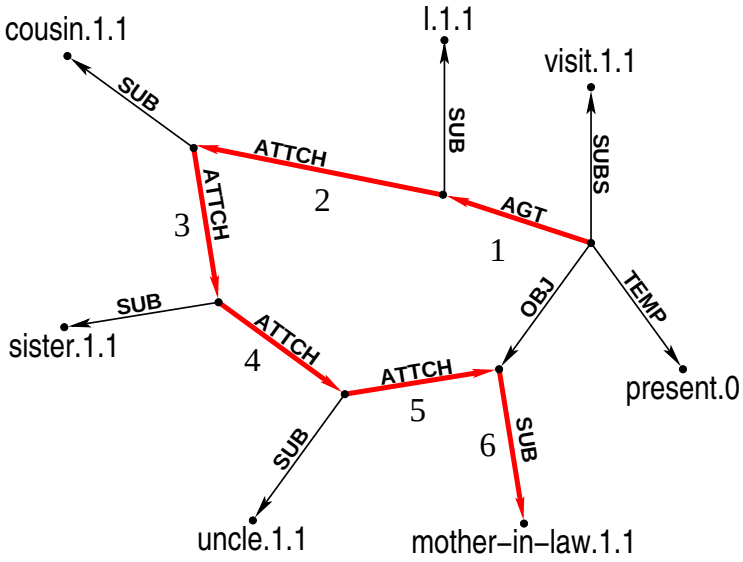

Figure 17: Semantic Network for the Sentence: *Ich besuche die Schwiegermutter des Onkels meiner Cousines Schwester.(I visit the mother-in-law, the uncle, the sister, and the cousin.)* The longest path, taking into account the direction of the arcs, is emphasized by printing the associated arcs in bold face.

Namen, der in einem schmutzigen Stadtteil einer kleinen Stadt lag. (A man visited a shop in a loud street with a strange name, which was in a dirty quarter of a small city.) Value: 6

A.6.2. Indicator: pronoun without antecedents

A Symbolic name: is-pronoun-without-antecedents
B Related criterion: discourse coherence
C Indicator definition: This indicator determines pronouns (personal pronouns, possessive pronouns) without any antecedent. Excluded from counting are pronouns that are deictic like *ich (I)*, *du/Sie (you)*, *wir (we)*, and *ihr (you)*.
D Relevant parser attributes: coref-pairs
E Associated segment type: word
F Value range: binary
G Example: *Es ist das beste Produkt. (It is the best product.)* Value: 1 for *Es (It)*, if the sentence is the first sentence of a text.
H Discussion: In rare cases, cataphoric references may be falsely identified as being without antecedents (postcedents). This small portion of false warnings can be accepted because cataphoric references are in general regarded as being hard to understand as well.

A.6.3. Indicator: number of reference candidates

A Symbolic name: num-reference-candidates
B Related criterion: coreference ambiguity
C Indicator definition: The indicator counts the candidate antecedents for a pronoun.
D Relevant parser attributes: coref-pairs
E Associated segment type: word

F Value range: integer
G Example: *Der Mann schlägt den Jungen. Er weint. (The man beats the boy. He cries.)* Value: 2, assuming that this sentence is the first one in a text.
H Discussion: For a more precise detection of possible coreference ambiguities, only text windows of three sentences are considered. To concentrate on clear cases, DeLite looks only at pronominal noun phrases.

A.6.4. Indicator: reference distance in words/sentences

A Symbolic names: a.) reference-distance-in-words , b.) reference-distance-in-sentences
B Related criterion: reference distance
C Indicator definition: The first indicator, reference-distance-in-words, counts the number of words between a mention and its most likely antecedent. The second indicator measures the distance by counting sentences.
D Relevant parser attributes: coref-pairs
E Associated segment type: word
F Value range: integer
G Examples: ***Der Mann*** *lacht.* ***Er*** *ist glücklich. (****The man*** *laughs.* ***He*** *is happy.)*, values for *Er*: reference-distance-in-words: 2, reference-distance-in-sentences: 1.
H Discussion: Punctuation marks are included in the word count as usual in DeLite.

B. Formal Indicator Definitions

Table 6: WOCADI output format used by DeLite. The output format is formally defined by the extended Backus-Naur form (EBNF in ISO 14977 notation [fS96] with commas omitted where unambiguous).

```
<WOCADI-OUTPUT>               =  <MENTION-PARTITION> <COREF-PAIRS>
                                 <W-OUT-SENT>;
<MENTION-PARTITION>           =  (<PARTITION-ELEMENT> );
<PARTITION-ELEMENT>           =  ( <NODE-NAME> <NODE-NAME> <NODE-NAME> )
<COREF-PAIRS>                 =  ( <COREF-PAIR> );
<COREF-PAIR>                  =  ( <COREF-RULE> <ANAPHOR>
                                 <ANTECEDENT-CANDIDATE> [<PROBABILITY>] );
<W-OUT-SENT>                  =  ( <W-OUT-SENT-ANALYSIS> );
<W-OUT-SENT-ANALYSIS>         =  <SENT-ANALYSIS-ALTERNATIVES>
                                 <SENT-ALTERNATIVES-ENET>
                                 <SENT-ANALYSIS-PASSES>
                                 <SENT-ANALYSIS-PASSES-BEST>
                                 <SENT-ANALYSIS-QUALITY>
                                 <SENT-ANALYSIS-TRIES>
                                 <SENT-ANALYSIS-ML>
                                 <SENT-DEPENDENCY-TREE>
                                 <SENT-FOCUS>
                                 <SENT-NET>
                                 <SENT-PARSE-NET>
                                 <SENT-PARSER-VERSION>
                                 <SENT-SENTENCE-ID>
                                 <SENT-SENTENCE-TYPE>;
<SENT-ANALYSIS-ALTERNATIVES>  =  (analysis-alternatives <INTEGER> );
<SENT-ALTERNATIVES-ENET>      =  (analysis-alternatives-enet <ENETS> );
<SENT-ANALYSIS-PASSES>        =  (analysis-passes <INTEGER> );
<SENT-ANALYSIS-PASSES-BEST>   =  (analysis-passes-best <INTEGER> );
<SENT-ANALYSIS-QUALITY>       =  (analysis-quality <INTEGER> );
<SENT-ANALYSIS-TRIES>         =  (analysis-tries <INTEGER> );
<SENT-ANALYSIS-WORDS>         =  (analysis-words ( <WORD> ));
<SENT-DEPENDENCY-TREE>        =  (dep-tree <DEP-TREE> );
<SENT-FOCUS>                  =  (focus <NODE-NAME> );
<SENT-NET>                    =  (net <NET> );
<SENT-PARSE-NET>              =  (parse-net <PARSE-NET> );
<SENT-PARSER-VERSION>         =  (parser-version <DATE> );
<SENT-SENT-ID>                =  (sentence-id <SENT-ID> );
<SENT-SENT-TYPE>              =  (sentence-type <SENT-TYPE> );
<DEP-TREE>                    =  ( <DEP-NODE-INFO> <DEP-INFO> );
<DEP-NODE-INFO>               =  ( <DEP-RELATION> <NODE-NAME>
                                 <BASE-FORM> <READING> <CATEGORY> );
<DEP-RELATION>                =  adj | compl1 | compl2 | compl3 |
                                 compl4 | compl5 | spec | ...  ;
<DEP-INFO>=<DEP-TREE>;
<ENETS>                       =  ( <ENET> );
<ENET>                        =  ( <SENT-ANALYSIS-QUALITY> <SENT-DEP-TREE>
                                 <SENT-FOCUS> <SENT-NET> <SENT-PARSE-NET>
                                 <SENT-SENT-TYPE> <SENT-SPANS> );
<SENT-SPANS>                  =  (spans ( <SENT-SPAN-ENTRY> ));
<SENT-SPAN-ENTRY>             =  ( <NODE-NAME> <INTEGER> <INTEGER> );
```

Fortsetzung Table 6: EBNF for WOCADI's output format (Continued).

```
<ANALYSIS-ML>                                  =  (analysis-ml (<ANALYSIS-ML-WORD*>));
<ANALYSIS-ML-WORD>                             =  (<ML-WORD> <ML-CHAR-ID>
                                                  <ML-CAT> <ML-LEMMA>
                                                  <ML-READING> <ML-PARSE-LEMMA>
                                                  <ML-PARSE-READING>);
<ML-WORD>                                      =  (word <WORD>);
<ML-CHAR-ID>                                   =  (char-id <INTEGER>);
<ML-CAT>                                       =  (cat (<CATEGORIES>*));
<ML-LEMMA>                                     =  (lemma (<LEMMA>*));
<ML-READING>                                   =  (reading (<READING>*));
<ML-PARSE-LEMMA>                               =  (parse-lemma <LEMMA>);
<ML-PARSE-READING>                             =  (parse-reading <READING>);
(* elements not formally specified here:  *)
<BASE-FORM>                                    =  <STRING>;
<CATEGORY>                                     =  (* part of speech (syntactic category) *);
<DATE>                                         =  <STRING> (* ISO date format *);
<LEMMA>                                        =  <STRING>;
<NET>                                          =  (* see documents on MultiNet *);
<NODE-NAME>=<STRING>;
<PARSE-NET> =(* see MultiNet documentation *)
<READING> =<STRING>;
<SENTENCE-ID>                                  =  <STRING>;
<SENTENCE-TYPE>                                =  declarative-sentence
                                               |  ...  (* see MultiNet documentation *);
<WORD> =<STRING> (* orthographic form *);
```

Table 7: The internal document format which DeLite operates on is defined in ISO EBNF notation.

```
<DOC_S>             =  (DOC <DOC_ATTRS> <SENT_S> ) ;
<DOC_ATTRS>         =  id= <STRING>
                    |  start= <INTEGER>
                    |  end= <INTEGER>
                    |  string=<STRING>
                    |  type=<STRING>
                    |  length=<INTEGER>
                    |  <DOC_INDICATORS>
                    |  <SCORE>
<DOC_INDICATORS>    =  token-type-ratio= <REAL>
                    |  average_sentence_length=<REAL>
                    |  ...  (* other derived indicators *)
                    |  num-abbreviation-tokens=<INTEGER>
                    |  num-abbreviation-types= <INTEGER>
                    |  num-acronym-tokens= <INTEGER>
                    |  num-acronym-types=<INTEGER>
                    |  num-wordform-tokens=<INTEGER>
                    |  num-lemma-tokens=<INTEGER>
                    |  num-wordform-types=<INTEGER>
                    |  num-lemma-types=<INTEGER>
                    |  <SENT_INDICATORS>;
<SENT_S>            =  (SENTENCE <SENT_ATTRS> <MULTI_WORD_S> <WORD_S> );
<SENT_ATTRS>        =  id= <STRING>
                    |  start= <INTEGER>
                    |  end= <INTEGER>
                    |  string= <STRING>
                    |  type= (wh-question | yes-no-question | declarative-sentence | ...)
                    |  length= <INTEGER>
                    |  <SENT_INDICATORS>
                    |  <SCORE>;
<SENT_INDICATORS>   =  abstract-nouns= <INTEGER>
                    |  num-complement-ambiguities= <INTEGER>
                    |  clause-embedding-depth= <INTEGER>
                    |  clause-center-embedding-depth= <INTEGER>
                    |  num-constituents= <INTEGER>
                    |  num-propositions= <INTEGER>
                    |  max-reas-cluster-size= <INTEGER>
                    |  num-reas-relations= <INTEGER>
                    |  num-concept-nodes= <INTEGER>
                    |  num-introduced-concepts= <INTEGER>
                    |  sn-quality= <INTEGER>
                    |  passive= <BOOLEAN>
                    |  longest-path= <INTEGER>
                    |  num-negations= <INTEGER>
                    |  num-negated-concepts=<INTEGER>
                    |  num-negated-adjectives=<INTEGER>
                    |  num-connections=<FLOAT>
```

Fortsetzung Table 7: EBNF for DeLite's internal document format (continued).

```
<PHRASE_S>            =  (MULTI_WORD <MULTI_WORD_ATTRS> <WORD_S> );
<PHRASE_ATTRS>        =
                      |  id= <STRING>
                      |  start= <INTEGER>
                      |  end= <INTEGER>
                      |  string= <STRING>
                      |  type= [np|idiom|support_verb_construction|...]
                      |  <PHRASE_INDICATORS>
                      |  <SCORE>;
<PHRASE_INDICATORS>   =  num-pp-attachment-candidates= <INTEGER>
                      |  num-genitive-np-attachment-candidates=<INTEGER>
                      |  num-constituents-per-coordination= <INTEGER>
                      |  num-words= <INTEGER>;
<WORD_S>              =  (WORD <WORD_ATTRS>);
<WORD_ATTRS>          =
                      |  word_id= <INTEGER>
                      |  start= <INTEGER>
                      |  end= <INTEGER>
                      |  string= <STRING>
                      |  type= [simplicium|compound|abbreviation|punctuation|...]
                      |  length= <INTEGER>
                      |  pos= (n|v|a|adv|...)
                      |  register= (slang|foreign|elevated_speech|...)
                      |  lemma= <STRING>
                      |  <WORD_INDICATORS>
                      |  <SCORE>;
<WORD_INDICATORS>     =  num-compound-simplicia= <INTEGER>
                      |  abstract-noun=<FLOAT>
                      |  deverabel-noun=<BOOLEAN>
                      |  deadjectival-noun=<BOOLEAN>
                      |  num-compound-concepts= <INTEGER>
                      |  num-syllables= <INTEGER>
                      |  num-characters= <INTEGER>
                      |  frequency-class= <INTEGER>
                      |  lemma-frequency= <INTEGER>
                      |  num-readings-from-lookup= <INTEGER>
                      |  num-readings-from-parse= <INTEGER>
                      |  synset-size= <INTEGER>
                      |  num-dependents= <INTEGER>
                      |  distance-verb-complement= <INTEGER>
                      |  distance-verb-adjunct= <INTEGER>
                      |  distance-verb-prefix= <INTEGER>
                      |  distance-verb-group-parts= <INTEGER>
                      |  num-reference-candidates= <INTEGER>
                      |  reference-distance-in-words= <INTEGER>
                      |  reference-distance-in-sentences= <INTEGER>;
                      |  pronoun-without-antecedent= <INTEGER>
<SCORE>               =  <SCORE_NAME> = <SCORE_VALUE>;
<SCORE_NAME>          =  total_score
                      |  mor_score | mor1_score | mor2_score | ...
                      |  lex_score | lex1_score | lex2_score | ...
                      |  sem_score | sem1_score | sem2_score | ...
                      |  dis_score| dis1_score | ...;
<SCORE_VALUE>         =  <REAL>;
```

C. Mean Absolute Error and Root Mean Square Error for each Indicator

Table 8: Mean Absolute (MAE) and Root Mean Square Errors (RMSE) of the normalized indicator values in comparison with the ratings of the test persons.

Indicator	MAE	RMSE	Surface
Morphological Level			
Deverbal noun	0.346	0.388	no
Deadjectival noun	0.371	0.412	no
Number of simplicia in a compound	0.212	0.267	yes
Number of compound concepts	0.218	0.273	no
Abbreviation type token ratio	0.338	0.380	yes
Acronym type token ratio	0.406	0.443	yes
Number of syllables	0.406	0.454	yes
Number of characters	0.226	0.275	yes
Lexical Level			
Inverse lemma frequency	0.201	0.249	yes
Frequency class	0.251	0.306	yes
Number of readings from lookup	0.248	0.303	no
Number of readings from parse	0.237	0.300	no
Synset size	0.167	0.196	no
Number of abstract nouns	0.320	0.382	no
Lemma type token ratio	0.393	0.431	yes
Word form type token ratio	0.397	0.434	yes
Syntactic Level			
Number of complement ambiguities	0.459	0.498	no
Number of dependants per verb	0.284	0.347	no
Number of dependants per NP	0.278	0.342	no
Clause embedding depth	0.380	0.426	no
Clause center embedding depth	0.362	0.406	no
Number of constituents per coordination	0.403	0.440	no
Average number of words per phrase	0.218	0.276	no
Quality of the semantic network	0.213	0.267	no
Average number of words per sentence	0.181	0.233	yes
Average number of constituents per sentence	0.346	0.405	no
Distance between the verb and its complements	0.364	0.409	no
Distance between verb and its adjuncts	0.372	0.416	no
Distance between verb and its prefix	0.408	0.444	no
Distance between verb and verb group parts	0.384	0.425	no
Passive	0.327	0.380	no
Semantic Level			
Number of propositions per sentence	0.335	0.390	no
Number of causal relations in a chain	0.297	0.368	no
Number of causal relation clusters	0.305	0.382	no
Maximum causal relations cluster size	0.303	0.380	no
Number of concept nodes per sentence	0.421	0.426	no
Number of negated concepts	0.392	0.430	no
Number of negated adjectives	0.393	0.430	no
Number of negations	0.312	0.361	no
Number of connection betweens entities	0.329	0.375	no
Maximum path length	0.315	0.368	no
Maximum path length (both directions)	0.290	0.348	no
Lexical Level			
Number of introduced concepts	0.210	0.260	no
Number of pronouns without antecedent	0.412	0.464	no
Number of reference candidates	0.382	0.434	no
Reference distance in words	0.407	0.453	no
Reference distance in sentences	0.393	0.440	no

Acknowledgement

This work was funded by the EU-project Benchmark Tools and Methods for the Web (BenToWeb, FP6-004275, URL: `http://www.bentoweb.org`). We thank all colleagues in our group who contributed to this work, especially Christoph Doppelbauer, Christian Eichhorn, Ingo Glöckner, Rainer Osswald and Sven Hartrumpf.

References

[Ams78] Toni Amstad. *Wie verständlich sind unsere Zeitungen?* PhD thesis, Universität Zürich, Zürich, Switzerland, 1978.

[BT97] Dimitris Bertsimas and John Tsitsiklis. *Introduction to Linear Optimization*. Athena Scientific, Belmont, Massachusetts, USA, 1997.

[CCGV07] Ben Caldwell, Michael Cooper, Loretta Guarino, and Gregg Vanderheiden. Web content accessibility guidelines (WCAG) 2.0, Working draft, 2007.

[CD95] Jeanne Chall and Edgar Dale. *Readability Revisited: The New Dale-Chall Readability Formula*. Brookline Books, Brookline, Massachusetts, USA, 1995.

[CS96] Raman Chandrasekar and Bangalore Srinivas. Automatic induction of rules for text simplification. Technical report, University of Pennsylvania, Philadelphia, Pennsylvania, USA, 1996.

[DLR77] Arthur Dempster, Nan Laird, and Donald Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B*, 39(1), 1977.

[Fle48] Rudolf Flesch. A new readability yardstick. *Journal of Applied Psychology*, 32:221–233, 1948.

[fS96] International Organization for Standardization. ISO/IEC 14977, 1996.

[GH62] J. Arthur Greenwood and H.O. Hartley. *Guide to Tables in Mathematical Statistics*. Princeton University Press, Princeton, New Jersey, USA, 1962.

[Gre93] William Greene. *Econometric Analysis*. Prentice Hall, Englewood Cliffs, New Jersey, USA, 1993.

[Gro92] N. Groeben. *Leserpsychologie: Textverständnis - Textverständlichkeit*. Aschendorff, Münster, Germany, 1992.

[Har03] Sven Hartrumpf. *Hybrid Disambiguation in Natural Language Analysis*. PhD thesis, FernUniversität in Hagen, Hagen, Germany, 2003.

[Hel06] Hermann Helbig. *Knowledge Representation and the Semantics of Natural Language*. Springer, Berlin, Germany, 2006.

[HH97] Hermann Helbig and Sven Hartrumpf. Word class functions for syntactic-semantic analysis. In *Proceedings of the 2nd International Conference on Recent Advances in Natural Language Processing (RANLP'97)*, pages 312–317, Tzigov Chark, Bulgaria, 1997.

[Jol86] Ian T. Jolliffe. *Principle Component Analysis*. Springer, Berlin, Germany, 1986.

[Kla63] George R. Klare. *The Measurement of Readability*. Iowa State University Press, Ames, Iowa, USA, 1963.

[Lik32] Rensis Likert. A technique for the measurement of attitudes. *Archives of Psychology*, 140:1–55, 1932.

[LvTT81] Inghard Langer, Friedemann Schulz von Thun, and Reinhard Tausch. *Sich verständlich ausdrücken*. Reinhardt, Munich, Germany, 1981.

[MLDM06] Philip M. McCarthy, Erin J. Lightman, David F. Dufty, and Danielle S. McNamara. Using coh-metrix to assess distributions of cohesion and difficulty: An investigation of the structure of high-school textbooks. In *Proceedings of the 28th annual conference of the Cognitive Science Society*, Vancouver, Canada, 2006.

[PPK+07] Helen Petrie, Christopher Power, Omar Kheir, David Swallow, Carlos A. Velasco, Henrike Gappa, Gaby Nordbrock, and Dimitar Denev. EU-deliverable BenToWeb 3.4: Evaluation of color vision deficiency module. Technical report, University of York and Fraunhofer Institute for Applied Technology, 2007.

[SHVV06] Christophe Strobbe, Sandor Herramhof, Evangelos Vlachogiannis, and Carlos A. Velasco. Test case description language (TCDL): Test case metadata for conformance evaluation. In *Computers Helping People with Special Needs*, volume 4061 of *LNCS*. Springer, Berlin, Germany, 2006.

[Tho95] Della Thompson. *The Concise Oxford Dictionary of Current English*. Oxford University Press, Oxford, UK, 1995.

[vL07] Tim vor der Brück and Johannes Leveling. Parameter learning for a readability checking tool. In Alexander Hinneburg, editor, *Proceedings of the LWA 2007 (Lernen-Wissen-Adaption), Workshop KDML*, pages 149–153. Gesellschaft für Informatik, Halle/Saale, Germany, 2007.