Volume 30 — Number 1 — 2015 — ISSN 2190-6858

JLCL

Journal for Language Technology
and Computational Linguistics

Herausgegeben von / Edited by
Lothar Lemnitzer

GSCL Gesellschaft für Sprachtechnologie & Computerlinguistik

Contents

Impressum

Herausgeber	Gesellschaft für Sprachtechnologie und Computerlinguistik (GSCL)
Aktuelle Ausgabe	Band 30 – 2015 – Heft 1
Anschrift der Redaktion	Lothar Lemnitzer Berlin-Brandenburgische Akademie der Wissenschaften Jägerstr. 22/23 10117 Berlin lemnitzer@bbaw.de
ISSN	2190-6858
Erscheinungsweise	2 Hefte im Jahr, Publikation nur elektronisch
Online-Präsenz	www.jlcl.org

Editorial

Ich begrüße Sie zur dieser thematisch nicht gebundenen Ausgabe des Journals.

Es ist das erste (und einzige) Heft des 30. Jahrgangs (ein Jubiläumsjahrgang, wenn Sie so wollen). Das Heft ist das Ergebnis einer Ausschreibung und der anschließenden Begutachtung der Einreichungen, für die ich mich herzlich bei allen GutachterInnen bedanken möchte.

Nach dem Begutachtungsprozess blieben fünf Beiträge übrig, die in diesem Heft zusammengestellt sind. In diesem Zusammenhang gilt mein Dank auch den Autorinnen und Autoren, die ihre Zeit und Mühe auf die von den Gutachtern vorgeschlagenen Überarbeitungen aufgewendet haben.

Amine Bayoudhi und Kollegen nehmen ein „multi-domain“ Korpus des Arabischen zum Ausgangspunkt für die Annoation der (emotionalen) Gerichtetheit von Ausagen auf der Ebene von Sätzen und Teilsätzen.

Alexander Magidow stellt in seinem Beitrag ein relationales Datenbankmodell für die Speicherung verschiedenartiger Typen von linguistischen Daten vor. Auch in seiner Arbeit spielt die arabische Sprache eine wichtige Rolle.

Laurent Romary macht in seinem Beitrag einen Vorschlag, Details des sog. „Lexical Markup Framwork“ (LMF) in TEI zu serialisieren. Eines seiner Beispiele ist die Modellierung eines koreanischen Wortnetzes.

Uladzimir Sidarenka und Kollegen schlagen ein Verfahren für die Segmentierung von Texten eines deutschen Korpus in Diskurseinheiten vor und evaluieren dieses Verfahren.

Lucie Flekova und Kollegen nehmen Transkripte audivisueller Aufnahmen aus Unterrichtssituation zum Ausgangspunkt für eine Klassifizierung nach diskurspragmatischen Kategorien. Hier kommen Verfahren des maschinellen Lernens zum Einsatz.

Zum Schluss noch ein Ausblick auf das Jahr 2016. Für dieses Jahr sind wieder 2 Hefte geplant, die jeweils auf Workshops basieren und „Korpora“ in verschiedenen Facetten zum Thema haben werden.

Ich wünsche Ihnen aber zunächst einmal viel Spaß und intellektuellen Gewinn bei der Lektüre der Beiträge dieses Heftes.

Beste Grüße aus Berlin

Lothar Lemnitzer Januar 2016

Amine Bayoudhi, Hatem Ghorbel, Houssem Koubaa, Lamia Hadrich Belguith

Sentiment Classification At Discourse Segment Level: Experiments on multi-domain Arabic corpus

Abstract

Sentiment classification aims to determine whether the semantic orientation of a text is positive, negative or neutral. It can be tackled at several levels of granularity: expression or phrase level, sentence level, and document level. In the scope of this research, we are interested in the sentence and sub-sentential level classification which can provide very useful trends for information retrieval and extraction applications, Question Answering systems and summarization tasks. In the context of our work, we address the problem of Arabic sentiment classification at sub-sentential level by *(i)* building a high coverage sentiment lexicon with semi-automatic approach; *(ii)* creating a large multi-domain annotated sentiment corpus segmented into discourse segments in order to evaluate our sentiment approach; and *(iii)* applying a lexicon-based approach with an aggregation model taking into account advanced linguistic phenomena such as negation and intensification. The results that we obtained are considered good and close to state of the art results in English language.

1 Introduction

Sentiment analysis refers to the computational study and processing of opinions, sentiments and emotions of people found in text (Al-Radaideh et al., 2014). Recently, this domain has significantly evolved and attracted widespread attention especially with the expanding growth of social networks services and user-generated web content. This situation provided a great opportunity to easily access and mine public opinions and sentiments about any subject. Business companies, for example, exploited this information source to discover consumer feedbacks about their products or even to decide future marketing actions.

Sentiment analysis includes several tasks. According to Liu (Liu, 2012), it can be divided into six main tasks: 1) extract and categorize all entity expressions from documents, 2) extract all aspect expressions and categorize them into clusters, 3) extract and categorize opinion holders, 4) extract the times when opinions are given and standardize the time formats for all opinions, 5) determine whether an opinion is positive, negative or neutral, 6) produce all opinion quintuples expressed in a document. Among these tasks, the fifth task, namely sentiment classification, is the one having received the most researcher attention.

Specifically, sentiment classification aims to determine whether the semantic orientation of a text is positive, negative or neutral. It can be tackled at many levels of granularity: expression or phrase level, sentence level, and document level. Expression sentiment classification aims to determine the prior sentiment class or valence of an expression. As for sentence level, the objective is to calculate the contextual polarity of a sentence. Concerning document level, the main goal is to mine the overall polarity of a document with the hypoth-

esis that is expressed by a single author towards a single target. In the scope of this research, we are interested in the sentence and sub-sentential level classification. This level of granularity can provide very useful trends for information retrieval and extraction applications, Question Answering systems and summarization tasks.

Sentence sentiment classification is often processed by applying machine learning techniques, in particular supervised learning which consists basically of two major steps: feature extraction and training the learning model. Though this approach has proved to be successful in producing high accuracy, it suffers from certain shortcomings. It requires building a huge corpus (dataset), which needs to be labeled manually by human experts (Abdulla et al., 2014). The process of manual annotation can be very difficult even for native speakers due to sarcasm and cultural references. It can also be expensive and time-consuming (He and Zhou, 2011). Moreover, the model built could be a domain-biased. That is, it could give low accuracy when applied in a domain, different than the domain from which it was learned (Read and Carroll, 2009). Due to these reasons, many researchers were oriented towards a second approach, namely the lexicon-based one.

In the context of our work, we address the problem of Arabic sentiment classification at sub-sentential level by *(i)* building a high coverage sentiment lexicon with semi-automatic approach; *(ii)* creating a large multi-domain annotated sentiment corpus segmented into discourse segments in order to evaluate our sentiment approach; and *(iii)* applying a lexicon-based approach with an aggregation model taking into account advanced linguistic phenomena such as negation and intensification. In fact, most of the recent works in Arabic language have not yet released their resources and some of them have common weak points such as not handling negation in the statement. In addition, the redundancy in the training data causes an ambiguity in sentiments (Ibrahim et al., 2015).

Compared to related Arabic sentiment classification work, the main contributions of this research are: *(i)* adopting a lexicon-based approach handling negation and intensification by applying state of the art strategies and establishing an extensive list of word and phrase operators, *(ii)* addressing Arabic sentiment classification at discourse segment level (first work according to our knowledge), *(iii)* experimenting Arabic sentiment classification on discussions and debates other than short comments and reviews, which is more difficult.

The rest of the paper is organized as follows. In section 2, we review a selection of key papers related to the sentiment classification for English and Arabic languages. In section 3, we describe our semi-automatic approach to build a sentiment lexicon of opinion words. In section 4, we outline our efforts to annotate two sentiment corpuses at discourse segment level. In section 5, we detail our proposed approach for sub-sentential sentiment classification, and we present and discuss the experiment results. In section 6, we sum up and provide some perspectives for future work.

2 Related work

Sentiment analysis is an emerging domain in natural language processing. Due to the explosion of user-generated web content, the interest in this field is continually increasing. Currently, dozens of research papers are published in this field in each year and many work-

shops are organized about this topic, such as WASSA (Workshop on Computational Approaches to Subjectivity, Sentiment & Social Media Analysis) and SENTIRE (Workshop on Sentiment Elicitation from Natural Text for Information Retrieval and Extraction). Due to the large number of publications and the variety of sentiment analysis tasks, we are interested in this section in only sentence and sub-sentential level and with an emphasis especially in works related to English and Arabic languages.

In the literature, we discern two main approaches: supervised machine learning approach and lexicon-based approach. In machine learning approach, sentiment classification is viewed as a special case of text categorization task. The sentiment classifier is built by extracting discriminative features from manually annotated sentiment data and applying a learning algorithm such as Support Vector Machines, Naïve Bayes and Maximum Entropy. Generally, the best performance is achieved by using n-grams feature, but also Part-of-speech and syntactic information can be important effective features (Pang et al., 2002); (Pak and Paroubek, 2010); (Ghorbel and Jacot, 2011).

At the sentence level, recently researches have started to study more advanced linguistic traits. For instance, Tang et al. (Tang et al., 2014) proposed a joint segmentation and classification framework for sentiment analysis in order to handle the inconsistent sentiment polarity between a phrase and the words it contains. Specifically, they used a log-linear model to score segmentation candidates, and utilize the phrasal information as features to build the sentiment classifier. The effectiveness of the joint model has been verified by applying it on the benchmark dataset of Twitter sentiment classification in SemEval 2013.

Yang et al. (Yang and Cardie, 2014) proposed an approach that allows structured modeling of sentiment while taking into account both local and global contextual information. Specifically, they incorporated intuitive lexical and discourse knowledge as expressive constraints while training the conditional random field model via posterior regularization. According to the experiments, these constraints allow achieving better accuracy than existing supervised models for the sentence-level sentiment classification.

Liu et al. 2014 (Liu et al., 2014) have focused in sentiment analysis of sentences with modality. They presented a linguistic analysis of modality and detailed its types. Then, they proposed some general linguistic features, and specific modality features to train a support vector machine classifier predicting the sentiment orientation in sentences with modalities. The reported experimental results outperformed traditional lexicon-based, unigram-based SVM and Naive Bayes classifiers.

With regard to lexicon-based approach, it typically uses a lexicon of opinion words where a sentiment polarity or intensity is associated to each entry. In addition to detected words, linguistic phenomena such as intensification and negation are often taken into account to aggregate the sentiment polarity of sentences. One of the pioneer researches in the lexicon-based approach is the work of Turney (Turney, 2002) who used a part-of-speech tagger to identify phrases that contain adjectives or adverbs, and then estimated the semantic orientation of each extracted phrase by using Pointwise-Mutual Information (PMI). The sentiment class is finally assigned to the review based on the average semantic orientation of the extracted phrases. Kim and hovy (Kim and hovy, 2004) extended Turney work by using a seed

list enriched by wordnet synonyms. They also proposed two other models to combine sentiment words, which are the product model and the geometric mean model.

Ding et al. (Ding et al., 2008) proposed a holistic lexicon-based approach to solve the problem of opinion words whose semantic orientations are context dependent in reviews by exploiting the review context. They also proposed an effective function based of sum method for aggregating multiple conflicting opinion words in a sentence.

Taboada et al. (Taboada et al., 2011) developed a Semantic Orientation CALculator (SO-CAL) based on some dictionaries where words are annotated with polarity and strength scores. SO-CAL introduced state of the art methods to deal with negation and intensification. The authors used Amazon's Mechanical Turk service to collect validation data to their dictionaries and performed their experiments on four different corpora with equal numbers of positive and negative reviews.

Chardon et al. (Chardon et al., 2013) proposed to compute the opinion orientation at the sub-sentential level using a parabolic model that accounts for the effects of negation and modality on opinion expressions based on several linguistic experiments. The parabolic model represents an opinion expression as a point on a parabola, negation as functions over this parabola and modality as a family of parabolas of different slopes. The reported evaluation of the model showed that it has good agreement with the way in which humans handle negation and modality in opinionated sentences.

Research done on Arabic sentiment analysis is considered very limited compared to other languages like English whether at document-level or sentence-level (Shoukry and Rafea, 2012). Indeed, Ibrahim and Salim demonstrated in their literature review (Ibrahim and Salim, 2013) that there is a lack of studies focusing on multilingual twitter sentiment analysis and especially on Arabic tweet opinion and Arabic tweet subjectivity. They pointed out also that the most features used for twitter SA for Arabic tweets are n-grams features, and the most methods used in twitter SA for Arabic tweets is Naive Bayes (NB) and Support Vector Machines (SVM). For instance, Shoukry and Rafea (Shoukry and Rafea, 2012) compared two machine learning techniques which are SVM and NB classifiers on 1000 collected tweets. The task is considered a sentence-level sentiment classification since tweets length was restricted to 140 characters. The authors used unigrams and bigrams as features and concluded that there is no difference in the results between them. Final classification results showed that SVM outperformed NB in sentiment analysis with an accuracy of 72.6%.

Abdul-Mageed et al. (Abdul-Mageed et al., 2014) developed the SAMAR system for subjectivity and sentiment analysis of Arabic social media using some Arabic morphological features. They used the SVM^{light} as classification algorithm and a multi-genre dataset collected from four different genres of social media websites.

Arafat et al. (Arafat et al., 2014) implemented the Aara' system for polarity classification over informal colloquial Arabic comments. The classification scheme consisted of four categories: strongly positive, positive, negative and strongly negative. Experiments were carried out on 815 comments collected from online newspapers and achieved 82% in terms of accuracy.

Recently, ElSahar and El-Beltagy (ElSahar and El-Beltagy, 2015) conducted an extensive set of experiments for the sake of benchmarking their collected datasets and testing their

viability for both two and three class sentiment classification problems. Yielded results showed that the best performing classifier was SVM and that the best effective feature representations were the combination of the lexicon based features with the other features.

Regarding lexicon-based approach, very few researches relative to Arabic were conducted. Al-Subaihin et al. (Al-Subaihin et al., 2011) introduced a sentiment analysis tool for Arabic social media. The tool is relies in merging human computation with natural language processing. Human computing aims to harness knowledge of humans in a novel way (such as a computer game), and use the gained results to solve certain steps in an otherwise fully automated system expressions. This technique was used by the authors to build sentiment lexicons. Given these lexicons and the set of negative, positive and neutral sentence patterns, user reviews are classified according to their sentiments.

Oraby et al. 2013 (Oraby et al., 2013) proposed a scalable opinion-rating system following a rule-based approach tailored to the Arabic language. The approach takes into account language-specific traits that allows for closer analysis of opinion-bearing queues such as polar words, basic negation words, intensifiers, and conjunction modifiers. The overall document rating were calculated by taking the positive polarity score over the total document polarity score to give an estimate of the document's polarity score as a ratio of polar units.

Abdulla et al. (Abdulla et al., 2014) presented detailed steps of building the main two components of the lexicon-based SA approach: the lexicon and the sentiment analysis tool. In particular, the sentiment tool was designed to take into account negation and intensification. To aggregate the review polarity score, the authors used the sum method.

3 Our approach: building the sentiment lexicon

While there has been a recent progress in the area of Arabic Sentiment Analysis, most of the resources in this area are either of limited size, domain specific or not publicly available (ElSahar and El-Beltagy, 2015). Therefore, we decided to build our own lexicon. However, since building a sentiment lexicon "from scratch" is a relatively expensive task, we have chosen to benefit from the available lexicons, enhance and enrich them in order to build our sentiment lexicon. Thus, we propose a semi-automatic approach exploiting a set of Arabic linguistic tools and resources (i.e. translator, tagger, dictionaries) and exploiting other English or multilingual sentiment lexicons (i.e. MPQA lexicon, SentiStrength lexicon). The approach includes also a manual annotation process of multi-domain collected corpus. The starting lexicon that we used to build our lexicon is MPQA Arabic translated lexicon (Elarnaoty et al., 2012).

Our approach consists of three phases including manual and automatic steps. These phases are: phase of study and cleaning, phase of enrichment, and phase of reforming and revision (Figure 1).

3.1 Phase of study and cleaning

This phase consists in manually reviewing the MPQA translated lexicon in order to detect possible defects. The process allowed us to identify the following anomalies:

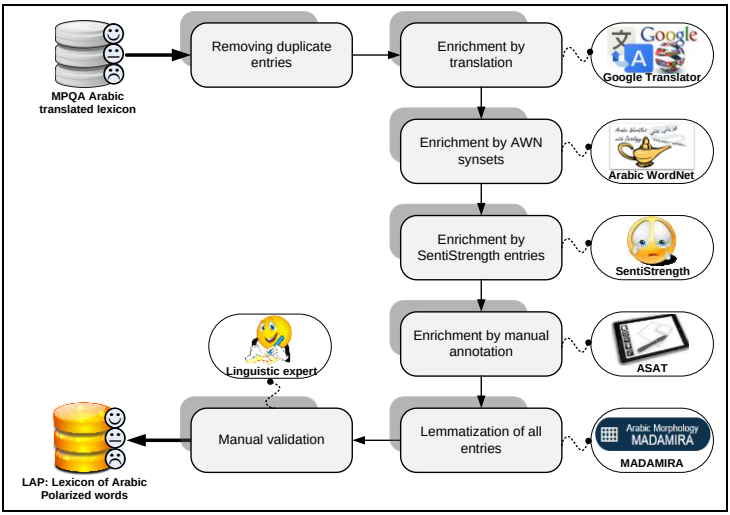


Figure 1: Creation step of the lexicon LAP

- Existence of many duplicate entries in the same class. For example: the words احتقار (contempt), أحمق (stupid), أزعج (discomfort), are found in many entries of the Negative Strong class.
- Existence of objective words (do not have **a priori** polarity). For example: معنى (sense), قرار (decision), يعامل (treat).
- Existence of misclassified words (wrong assigned polarity). For example: الخوف (fear) is classified as Strong Positive word when it should be classified as Strong Negative, ممتاز (excellent) is classified as Weak Positive instead of Strong positive, ألم (pain) is classified as Strong Positive instead of Strong Negative.
- Lack of Part of Speech (POS) tags. Indeed, MPQA translated lexicon is stored in four TXT files representing each sentimental class. POS tags are missing despite the fact that they are very useful in resolving morphological disambiguation (i.e. ذهب can be a name and means "gold", and can be a verb and means "go").

In order to remedy some of these anomalies, we performed a manual cleaning operation by eliminating duplicated words. These words are of two types: (1) duplicated words in the same class (they are unnecessary words that can be deleted without any problem), (2) duplicate words in classes that must be removed to avoid ambiguity. In fact, at this stage, the lexicon has no disambiguation technique.

3.2 Phase of enrichment

This phase consists of four steps, namely, enrichment by translation, enrichment by importing synsets of ArabicWordNet, enrichment by importing entries from the SentiStrength

lexicon and enrichment by manual annotation. In each step, only new words that do not exist in the lexicon are added to it.

- Enrichment by automatic translation: the Arabic version of MPQA does not contain the translation of all the words in the original English version. Indeed, the number of words in the English version of MPQA is 8222 and the number of words in the Arabic version is 7434. Therefore, we used the Google translator to translate the rest of the words. Among the words that have been added in this step, there are the words نزيه (probe), ثأر (revenge), بارز (marking). Note that during the translation process, there are words that can be added to the lexicon because of the existence of several translation possibilities. Similarly, there are words that can be eliminated because their corresponding Arabic words already exist in the lexicon.
- Enrichment by synsets of WordNet: This step is based on the assumption that the words having the same synonyms have the same polarity. Therefore, we have enriched our lexicon by ArabicWordNet synsets (Boudabous et al., 2013) corresponding to each word of Arabic MPQA lexicon. This is performed by exploiting the semantic relation "Near_synonym" of ArabicWordNet. Examples of the words added to the lexicon in this step are شديد (severe) that is the near synonymous of عنيف (violent), منفصل (separate) which is the closest synonym to مفكك (disassembled) and واقعي (realistic) which is the near synonymous حقيقي of (real).
- Enrichment by SentiStrength entries: To enrich our lexicon, we have exploited some other multilingual sentiment lexicons which are freely available for scientific research, such as SentiStrength (Thelwall et al., 2010). The Arabic version of this lexicon has a small size and includes only 1,434 entries. Nevertheless, this step was useful since it has fed our lexicon by new opinion words, for example, أناني (selfish) الأمثل (ideal) and اطمئنان (contentment).
- Enrichment by manual annotation: This step allows enriching the lexicon by annotating sentiment corpora. The annotation allows to mark subjective words of the text and to add them to the lexicon LAP according to their sentimental class. The annotation process is performed using an annotation tool developed for this task and called ASAT tool (Arabic Sentiment Annotation Tool). ASAT offers the possibility to the annotator to mark words with different colors according to their sentimental classes. The annotation is carried out according to the annotation scheme of MPQA translated lexicon. This scheme is composed of four classes: Negative Strong, Weak Negative, Positive Weak and strong Positive. The annotated corpus used for the enrichment of the lexicon is COPARD2 (Corpus of Arabic Opinion Debates), a collected corpus from the political domain. It consists of a set of TV political debates type of programs broadcast by Aljazeera (see section 4.1.2).

3.3 Phase of reforming and revision

To ensure good coverage of our lexicon and a size compression, we saved the opinion words as lemmas using MADAMIRA tool (Pasha et al., 2014). This allows, on the one hand, detecting all morphological variants of the opinion words, and on the other hand, saving memory space and execution time. Finally, to improve the quality of the lexicon, a validation step was conducted by a linguistic expert. The aim is to check that the polarity assigned to each word corresponds to the most frequent context in which it is used. Table 1 illustrates the evolution of the size of the lexicon after accomplishing each phase of our approach.

Table 1: Evolution of the size of the lexicon

Class	Arabic MPQA	Ph1	Ph2	Ph3
Negative Strong	2,860	1,752	2,991	1,544
Negative Weak	2,057	741	3,056	1,719
Positive Strong	1,386	922	1,990	1,278
Positive Weak	1,131	571	1,608	761
Total	7,434	3,986	9,645	5,302

4 Annotation of sentiment corpuses

In this section, we describe our efforts in annotating two sentiment corpuses at discourse segment level. In our context, we will use the annotated corpuses to evaluate the approach that we will propose afterwards (section 5) to classify Arabic discourse segments according to their polarity. However, these corpuses can be as well exploited for machine learning purposes. First, we present our data collection and provide statistics on each corpus. Second, we argue the adoption of discourse segmentation and present the used tool. Third, we explain the annotation process and outline the annotation model and guidelines. Finally, we discuss the obtained results.

4.1 Data collection

4.1.1 OCA (Opinion Corpus for Arabic)

It consists of 500 documents divided equally into positive and negative (Table 2). The corpus was collected by extracting reviews about movies from Arabic web pages and blogs (Rushdi-saleh et al., 2011). After that, many processing steps on each review were carried out in order to obtain a formatted document. The main steps were removing HTML tags and special characters, correcting spelling mistakes, filtering out nonsense and nonrelated comments, fixing Romanized comments and comments in different languages. OCA was annotated at document level by classifying its documents into positive and negative. This classification was automatically performed by exploiting the review rating score given by the user.

4.1.2 COPARD2 (Corpus of OPinion Arabic Debates 2)

It consists of 20 episodes of political debates broadcast on Aljazeera satellite channel. Most of the discussed issues were related to the political situations in the countries of Arab Spring especially in Tunisia and Egypt. All selected debates were multiparty conversations involving speakers of different profiles: politicians and political activists, economists, sociologists, security experts etc. These debates were transcribed and made publicly available in PDF format in Aljazeera website. After downloading the debates, some preliminary preprocessing steps were performed such correcting spelling and conversion mistakes, normalization and standardization of some Arabic letters, and removing thematic header of the debates.

Table 2: specificities of OCA and COPARD2

Property	OCA	COPARD2
Domain	Movie reviews	Politics
Number of documents	500	20
Number of words	209,733	97,172
Avg. of words per document	419	4,858
Number of segments	18,377	8,234
Avg. of segments per document	36	411
Avg. of words per segment	11.41	11.8

4.2 Segmentation into discourse segments

The objective of this work is to investigate sentiment classification at local level which is coarser than expression level and finer than document level. This requires splitting text into several segments of tokens connected by a structural or semantic relation. Most researchers have gone for considering sentences as these portions of text, and a lot of studies in sentiment classification were focused on sentence level. Nevertheless, in our current research, we do not share the opinion that sentences are the appropriate segmentation unit to study sentiment classification at local level, and this is due to many reasons.

First, theoretical definition of a sentence as "a part of a speech or a written discourse that has a complete and independent meaning" (Khalifa et al., 2011) do not offer, in Arabic case, enough cues to estimate sentence boundaries. As a matter of fact, unlike Indo-European languages, in Arabic there is no capitalization, which makes recognizing sentence boundaries a harder task. In addition, there are no strict rules of punctuation. This is leads to a rare use of punctuation marks, and even if they are used, they are not decisive cues to guide the segmentation process (Belguith et al., 2008). Moreover, Arabic discourse tends to use long and complex sentences. The average number of words per sentence is larger than the average in English sentence. For example, we can easily find too long paragraph with only one punctuation mark at the end (Keskes, 2015); (Aliwy, 2012).

Second, given the additional length of Arabic sentence compared to other languages, this sentence contains often more than one opinion expression and opinion target. Or, dealing with more than one opinion target per segmentation unit is a very complicated task in sentiment classification. That's why, most researches adopt an ignorance strategy in this issue and start from the hypothesis that the detected opinion expressions are expressed towards a single target, even if the problem is tackled at a document level (they are expressed throughout a whole document). Hence, it will be more beneficial to adopt a segmentation unit which is shorter than sentence in order to maximize chances to have only one target per segmentation unit.

Third, adopting a minimal segmentation unit shorter than sentence will help to resolve another problem in sentiment classification which is defining the scope of opinion operators. Actually, local polarity is highly affected by opinion operators in particular negation operators. Estimating opinion words affected by these operators is a very challenging task.

Among efficient solutions proposed to this problem is to adopt a minimal segmentation unit and to propagate the negation effect to all opinion terms of this segmentation unit.

Given the reasons explained above, we decided to split our texts into discourse segments and to study sentiment classification at this local level of granularity. Indeed, According to Chardon (Chardon, 2013), after splitting his corpus into discourse segments or Elementary discourse Units (UDE), 90% of these units contain only one opinion expression. An EDU is mainly a sentence or clause in a complex sentence that typically correspond to a verbal clause, as in [*I loved this movie*]$_a$ [*because the actors were great*]$_b$ where the relative clause introduced by the discourse connective because, indicates a cutting point. An EDU can also correspond to other syntactic units describing eventualities, such as prepositional and noun phrases, as in [*After several minutes,*]$_a$ [*we found the keys on the table*]$_b$ (Keskes et al., 2014).

To ensure the best segmentation quality, segmentation process was semi-automatically performed and conducted by an Arabic native speaker annotator using two segmentation tools. The main tool was ATS (Arabic Text Segmenter) developed by Keskes (Keskes, 2015). ATS was designed to divide documents into EDU following the Segmented Discourse Representation Theory (SDRT) principles. It adopts a pattern based approach relying on punctuation marks and discourse connectors as cues. ATS, evaluated in a collected corpus of elementary school books, has achieved good results around 85.5% in terms of F-measure. Nevertheless, in our case, our data collection consists of spontaneous user generated content and transcribed dialogues which are much more difficult to segment than school books written with regular Arabic discourse style. Hence, in order to ensure that the segmented output do not contain too long sentences neither broken sentences or clauses, another segmentation proposition is offered to the annotator by using STAr tool. STAr was developed by Belguith (Belguith et al., 2005) to segment non-vowelled Arabic texts into paragraphs and sentences. The tool adopts an approach based on a contextual analysis of the punctuation marks and a list of particles, such as the coordination conjunctions. Evaluation results on a corpus of elementary school books yields a precision of 80.65%. Given the two segmentation propositions, the annotator intervenes to choose the appropriate segmentation output consisting of minimal segments or clauses holding a complete and independent meaning. Final segmentation results are illustrated in Table 2.

4.3 Annotation process and guidelines

In the literature, many annotation models were proposed in sentiment classification either at expression level or at document level. For instance, in MPQA Project (Wilson et al., 2006), the proposed model to annotate news articles, included three attributes: polarity, intensity with 4 values, and explicit or implicit character. A more detailed model is proposed by Daille et al. (Daille et al., 2011) which contains five semantic categories: opinion, appreciation, agreement/disagreement, judgment and acceptation/refuse.

In Arabic language, the most known annotation work was realized by Abdul-Mageed et al. (Abdul-Mageed et al., 2012) who introduced AWATIF, a multi-genre Arabic corpus labeled at sentence level. The corpus was labeled using both regular and crowdsourcing methods with two types of annotation guidelines: simple and linguistically-motivated. For the simple annotation, two examples of positive, negative and neutral sentences were provided to the

annotators as guidelines, then, they were asked to label the data according to these three categories. As for the linguistically-motivated annotation, before starting the annotation task, annotators were exposed to a linguistics background, and the nuances of the genre to which each dataset belongs were explained to them. The annotated datasets consisted of collections of newswire stories from various domains (e.g., political, economic, sports), 30 Wikipedia Talk Pages, and web forum extracts comprising 2532 threaded conversations from 7 forums. Depending on these datasets, the annotation model was updated by adding "MIXED" or "OBJECTIVE" classes. Annotator Agreement reached good rates that vary from 0.79 to 0.82 depending on the dataset.

In the current research, we have assigned the task of annotating our data collection for sentiment analysis to two students familiar with natural language processing domain. The two students were Arabic native speakers and postgraduate. This profile was selected based on Abdul-Mageed et al. funding's who assert that linguistics background can be very useful for sentiment labeling since the concepts of subjectivity and sentiment are fuzzy. Indeed, the authors reported an achieved improvement of linguistically-motivated annotation when compared to simple annotation guidelines. Our Annotation model and guidelines are described in the two next sections.

4.3.1 Annotation model

Our annotation model (Figure 2) is based on two levels of granularity: expression level and discourse segment level. In each level, we tried to rely only on basic sentiment attributes that are essential for our task. The objective is first to reduce the expansive cost of the annotation task in terms of time consumption and money; and second, to maximize the annotator agreement which can be degraded when the annotation model contains too many categories.

Briefly, our annotation model consists:

- At the expression level of three attributes:
  - Polarity: polarity of the expression which can be positive or negative. Neutral expressions are not labeled.
  - Intensity: intensity of the expression which can be strong or weak.
  - Introducer: term used to introduce an opinion expression. It can be evaluative (polarized) or non-evaluative (non-polarized).

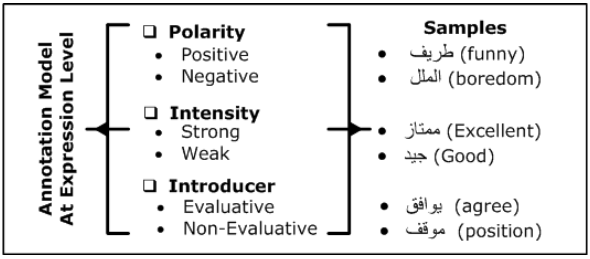


Figure 2: Annotation model at Expression level

- At discourse segment level it only consists of the attribute "Polarity" which allows classifying discourse segments depending on their semantic orientation into positive, negative and neutral. Neutral class includes objective segments that do not express any position or sentiment and also segments that express neutral position or sentiment.

Table 3. Sample of each type of discourse segments

Polarity	Sample
Positive	كما لعب العمل الجماعي دورا هاما في نجاح الفيلم
	Teamwork has also played an important role in the success of the movie
Negative	ولكن ليس هناك اتفاق حول المخرج من هذه الأزمة
	But there is no agreement on the way out of this crisis
Neutral	استفساري الثاني يتعلق بالمغربية سناء موزيان
	The second question is about the Moroccan Sana Mouziane

4.3.2 Annotation guidelines

Annotation guidelines are a set of instructions communicated to the annotators to help them to further understand the task and resolve the difficult encountered cases. Therefore, in our guidelines, we provided several examples to the annotators illustrating each sentiment attribute in the annotation model. In addition, to simplify the task, we made the assumption that each segmentation unit contains only one opinion target, but of course it may contain more than one opinion expression. This will help the annotators to focus more on opinion expressions than extracting the target or the holder, which is beyond the scope of this paper. Moreover, the annotators were asked to label all morphological forms that can bear an opinion or a sentiment: adjectives, adverbs, nouns and verbs. In fact, adjectives are significant indicators of opinion expressions. However, that does not mean that other Part-Of-Speech tags do not contribute to opinion expressions. Indeed, a lot of researchers point out that adjectives and adverbs are better than adjectives alone and certain verbs and nouns are also strong indicators of sentiment (Xia et al., 2011).

Nevertheless, to ensure a good progress of the annotation process, the two annotators were trained separately by annotating, under our supervision, the first 10 documents of OCA corpus and the first political debate of COPARD2 corpus.

4.4 Annotation Results

In order to evaluate the annotation task at discourse segment level, we used the confusion matrix to visualize the numbers of segments in which annotators agree and disagree according to each annotation category. Each column of the matrix represents the instances labeled by the first annotator, while each row represents the instances labeled by the second annotator. Figure 3 and Figure 4 illustrate respectively the confusion matrix related to OCA and the confusion matrix related to COPARD2.

		Annotator 2			
		Positive	**Negative**	**Neutral**	**Total**
Annotator 1	**Positive**	3,579	435	1,203	5,217
	Negative	260	2,966	1,013	4,239
	Neutral	978	1,946	6,485	9,409
	Total	4,817	5,347	8,701	18,865

Figure 3: confusion matrix relative to OCA

Then, to measure the annotator agreement, Cohen's kappa coefficient is computed. The results we obtined were 0.51 and 0.35 respectively for OCA and COPARD2. This agreement rate is considered very poor and subsequently, the annotated data cannot be considered enough homogenous to compare the machine results to it.

		Annotator 2			
		Positive	**Negative**	**Neutral**	**Total**
Annotator 1	**Positive**	408	27	93	528
	Negative	21	393	53	467
	Neutral	809	1,100	5,464	7,373
	Total	1,238	1,520	5,610	8,368

Figure 4: confusion matrix relative to COPARD2

By observing the two confusion matrices, we can easily find the cause of the rather poor agreement rates. Indeed, we can clearly see that the high number of instances where annotators disagree concerns always the "Neutral" category. Therefore, to resolve the problem, we decided to abandon neutral segments and to transform the problem to a binary classification task. Hence, we have removed agreed neutral segments (segments agreed by the two annotators to be neutral) as well as disagreed neutral segments (segments labeled by of one the annotators as neutral). The updated confusions matrices are illustrated in Figure 5 and Figure 6.

		Annotator 2		
		Positive	**Negative**	**Total**
Annotator 1	**Positive**	3,579	435	4,014
	Negative	260	2,968	3,228
	Total	3,839	3,403	7,242

Figure 5: confusion matrix relative to OCA without Neutral category

With these new confusion matrices values, Kappa rates for OCA corpus has reached 0.8 and for COPARD2 corpus 0,89. These new results are considered good enough for the purpose of using the annotated data for the sentiment classification task.

		Annotator 2		
		Positive	**Negative**	**Total**
Annotator 1	**Positive**	408	27	435
	Negative	21	393	414
	Total	429	420	849

Figure 6: confusion matrix relative to COPARD2 without Neutral category

4.5 Creation of Gold Standard

Although removing neutral category has resolved the problem of the poor annotator agreement, it has severely decreased the number of annotated segments in our collected data from 27,233 segments to 8,089 segments. In order to increase the size of our annotated data, we have revised our decision of abandoning neutral category, and we have decided to reject only agreed neutral segments. For the disagreed neutral segments, an adjudication operation is applied by a senior annotator (me) to add them to our final Gold standard version. The adjudication process included also disagreed polarized segments which are 695 segments in OCA corpus and 48 segments in COPARD2 corpus. The final properties of the Gold Standard versions of OCA and COPARD2 are presented in Table 4.

Table 4. Statistics on the Gold Standard versions of OCA and COPARD2

	OCA	COPARD2
Positive segments	7,455	1,794
Negative segments	4,931	1,110
Total	12,386	2,904

4.6 Discussion

Improving researches in sentiment analysis relies basically on availability of linguistic resources, in particular sentiment corpus. Such resource is required to conduct the linguistic study of the problem, to carry out a machine learning technique, or to evaluate the implemented proposed solution. In spite of that, sentiment corpuses annotated at local level are very rare. This is due to the high and expansive cost necessary to build them. Actually, according to our knowledge, Abdul-Mageed et al. (Abdul-Mageed et al., 2012) work is the only consistent attempt to create a sentiment corpus annotated at sentence level for Arabic language, and the annotated data are not yet released. In comparison to their work, we tried in this research to perform sentiment classification at a finer level, which is discourse segment. The strength of using this level of granularity was explained in section 4.3. In addition, similarly to Abdul-Mageed et al. work, our data collection is multi-domain including movie

review and political discussions. The final number of annotated segments in the standard Gold version is over than 15,000 segments.

5 Our approach for Sentiment classification

As seen in the literature survey, sentiment classification can be tackled by adopting a machine learning approach or by setting up a lexicon-based model. Our proposed approach (Figure 7) to classify discourse segments according to their polarity is based on a rule model and a sentiment lexicon to detect opinion expressions. It consists of three phases: preprocessing steps, detection of opinion expression, and computation of polarity score of the discourse segment. The first and second phases exploit Arabic linguistic resources, while, the third phase is language-free.

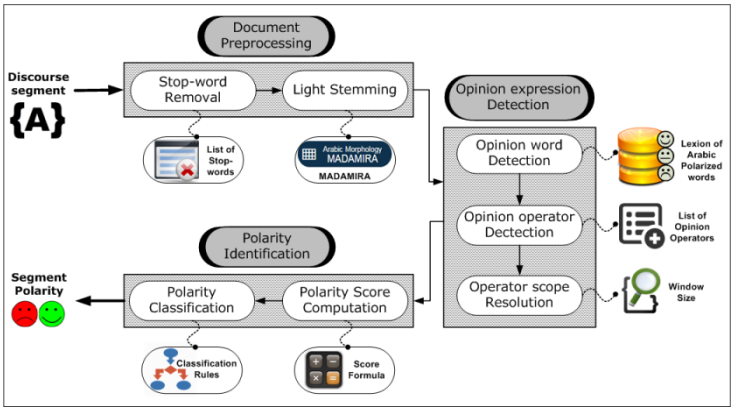


Figure 7: Steps of the proposed approach

5.1 Preprocessing steps

Preprocessing steps are required to accelerate and optimize the detection of opinions. They consist of two steps: stop-words removal and word stemming.

5.1.1 Stop-word removal

To accelerate the detection process of opinion words, we have profited from the stop-word list of Khoja stemmer tool (Khoja and Garside, 1999). In fact, this Stop-word list was widely used in Arabic processing community (Al-kabi, 2013) (Ababneh et al., 2012) (Sawalha and Atwell, 2008), but it was established to serve information retrieval applications. In sentiment classification task, a more reduced list is required, because many non-informative bearing words (such as negation operators and discourse markers) can serve as helpful cues in sentiment classification. Therefore, the stop-word list was revised to be tailored to sentiment classification constraints.

5.1.2 Stemming

Unlike Indo-European languages such as English and French, stemming in Arabic language is more difficult, mainly due to the fact that Arabic is an agglutinative and derivational language. Indeed, in Arabic, there are many more affixes than English and this leads to a large number of word forms. Besides, as a property of words in Semitic language, Arabic stem has an additional internal structure consisting of a two parts namely "root" and "pattern". The root is usually a sequence of three consonants and has some abstract meaning. The pattern is a template that defines the placement of each root letter among vowels and possibly other consonants. For example, in Figure 8, the word "AlkitabAn", meaning the two books, has a root composed of the letters k, t, and b; and his pattern is "Root1+i+Root2+a+Root3" (Heintz, 2010).

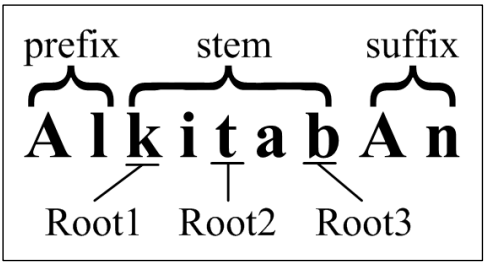


Figure 8: Example of Arabic stemming (Heintz, 2010)

Many Arabic tools, in particular morphological taggers, allow extracting roots from words. But, very few of them provide stems. To our knowledge, MADAMIRA (Pasha et al., 2014) is the only available light stemmer which performs morphological analysis and disambiguation of Arabic. Therefore, we used MADAMIRA to apply a light stemming on each document. Light stemming aims to reduce words to their lemma forms: for verbs, this is the 3rd person masculine singular perfective form and for nouns, this corresponds to the singular default form (Abdul-Mageed et al., 2014). In fact, stemming, which reduces words to their roots, is not convenient in Arabic language, because it may affect the word sense. Light stemming will be helpful to detect all morphological variations of the word.

5.2 Opinion expression detection

Opinion can be defined as a quadruplet Op=(w, t_w,h_w,$opers_w$), where:

- w is the opinion expression,
- t_w is the opinion target or topic,
- h_w is the opinion holder,
- $opers_w$ is the operator list affecting the opinion expression (Chardon, 2013).

In the same way, opinion expression is defined as "the minimal portion of text bearing an opinion". Hence, to identify the sentiment class of an opinion, it is necessary to identify the polarity of the words forming its opinion expression, and to analyze the effects of its operators.

5.2.1 Opinion bearing word detection

Once a sentiment lexicon is available, detecting opinion bearing words becomes a relatively simple task. In fact, preprocessing steps allow reducing the search scope by removing stop-words, and they allow also optimizing the detection process by mining the stems instead of the words themselves. Subsequently, the detection task becomes a naive comparison of two strings.

However, a more advanced treatment of this task may invoke the semantic disambiguation problem. Some word sense ambiguities are addressed by taking part of speech (POS) into account. For instance, *plot* is only negative when it is a verb, but should not be so in a noun dictionary; *novel* is a positive adjective, but a neutral noun (Taboada et al., 2011). Nevertheless, in Arabic language, this problem is much more challenging since most Arabic texts are non-vowelized. This leads to a high number of possible candidate solutions. For instance, "كرم" with POS=Noun can be vowelized as "كَرَمٌ" (generosity) which is positive, or as "كَرْمٌ" (vineyard) which is neutral.

In the current research, given the structure of the used sentiment lexicon LAP, opinion word detection was reduced to simple task especially that LAP is still under construction and his entries do not include POS information yet.

5.2.2 Opinion operator detection

Opinion operators or modifiers are linguistic elements which do not intrinsically bear opinions, but they are altering the characteristics of opinion words located in their scope (Chardon, 2013). In the course of our research, we consider only the two main opinion operators: intensifiers and negation operators. A limited list of each opinion operator category is prepared by a linguistic expert. Other operators such modality operators (Liu et al., 2014) and conditional operators (Narayanan et al., 2009) are left for future work.

- *Intensifiers*: they are operators altering the polarity or the intensity of the opinion expression. We distinguish two types of intensifiers: *(i)* amplifiers (i.e. very, much, extremely) which strengthen the intensity of the opinion expression, *(ii)* attenuators (i.e. little, less) which weaken the intensity of the opinion expression. It is notable that most intensifiers are adverbs and that many of them are term-compound such as "إلى حدّ كبير" (pretty much).

- *Negation operators*: Negation is a very common linguistic construction that affects polarity and, therefore, needs to be taken into consideration in sentiment analysis (Wiegand et al., 2010). Similarly to other language such as English (Taboada et al., 2011) and French (Benamara et al., 2012), negation can be introduced by different ways, through: *(i)* negators such as "not" and "without", *(ii)* quantifiers such as "never" and "nobody", *(iii)* lexical negation such as "absence" and "lack of".

In practice, according to their relative position to the opinion expression, we have classified negation operators into two categories: right operators and left operators. Right operators are the main negation words, while left operators can coexist in the same segment to play the role of quantifiers.

The detection of these operators follows the same technique described above concerning opinion bearing words. However, they are stored in specific separate lists since they do not

bear opinions or sentiment and subsequently have not prior polarity scores. Table 5 illustrates samples of Arabic opinion operators.

Table 5. Samples of opinion operators

Opinion operators	Samples
Amplifiers	جدا، كثيرا، (very, much)
Attenuators	قليلا، بعض الشيء، (little, slightly)
Right negation operators	لا، ما، لن، دون (not, less)
Left negation operators	أبدا، بتاتا، (never, at all)

5.2.3 Resolution of the opinion operator scope

While identifying intensifier scope is a simple task and can be performed by locating the closest opinion word to the intensifier, identifying the negation scope is among the challenging tasks in sentiment classification. In fact, negation scope and its effects have been a subject of interest for many researchers, not only in sentiment analysis domain, but also in many other fields such as philosophy, logic, and psycholinguistics (Morante and Sporleder, 2012). Basically, negation scope can be defined as the parts of a sentence whose meaning is inverted by a negation word. To resolve this problem, two major approaches were proposed in the literature: rule-based approach and machine learning approach. The rule-based approach relies upon linguistic rules which seek for negation words in the sentence and invert the meaning of certain surrounding parts based on different predefined window sizes (Prollochs et al., 2015). For instance, Hogenboom et al. (Hogenboom et al., 2011) have achieved a significant increase in overall sentiment classification accuracy when applying a two word window in a set of English movie review sentences. Concerning the machine learning approach, many techniques were applied to predict negation scope such conditional random fields (Councill et al., 2010) and Hidden Markov Models (Prollochs et al., 2015)

However, since machine learning approach, in particular supervised methods, requires an annotated training data which is unavailable and difficult to create, we have chosen to follow a window-based method to resolve negation scope. Hence, a set of experiments were carried out to determine the most effective window size. Obtained results are presented in the section 5.4.1.

5.3 Identification of the segment polarity

Identifying the polarity of the discourse segment depends, as we mentioned earlier, on the detected opinion bearing words and on the opinion operators affecting them. In this section, we explain the mapping process from the expression level to the discursive segment level; in other words, how can we exploit opinion words and operators to identify the polarity of a segment called the contextual polarity?

5.3.1 Prior Polarity

After detecting opinion bearing words, a polarity score P_w is assigned to each word according to its polarity and its intensity. This score, representing the prior polarity of the word or

the out of context polarity, is calculated according to the formula: $P_w = Pol_w * int_w$ where Pol_w is the polarity of the word and int_w is its intensity. Pol_w and int_w are respectively determined according to the lexicon sentiment class ant the intensity class into which the word belongs. Pol_w for the "Positive" class is 1 and for the "Negative" is -1. Int_w for the "Weak" class is 1 and for the "Strong" is 2. So, for example, the word "احتفل" (celebrate) which belongs to the "Positive" polarity class and the "Strong" intensity classe, P_w(احتفل)= 1*2=2.

5.3.2 Operator effect

Concerning negation operators, their effect on opinion expression is addressed at the local level by following one of three main strategies:

- Polarity reversal: called also switch negation. It is the classic approach for dealing with negation in sentiment analysis. It consists of changing the polarity sign of the opinion expression (Sauri 2008). For example, if P_w("good")=3 in a scale of [-5..5], then P_w("not good") will be -3.

- Polarity linear shift: first introduced by Taboada et al. (Taboada et al., 2011) who pointed out that polarity reversal works well in certain cases but fails drastically in others. For example, if P_w("Excellent")=5, we cannot say that P_w("not excellent")=-5. Therefore, they proposed treat negation by shifting the intensity towards the opposite polarity by a fixed amount. The amount used in implementation was 4. So, P_w("not excellent") will be 1.

- Polarity angular shift: first introduced by Chardon et al. (Chardon et al., 2013) who represent opinion expression by a point E of a parabola of focus F and summit O. The angle OḞE allows measuring the polarity score of the opinion expression. The negation effect is computed by adding/subtracting π to/from the angle OḞE. For example, if P_w("Excellent")=5π/6 in a scale of]-π..π[(5 in a scale of [-5..5]) , then P_w("not excellent") will be –π/6 (-1 in a scale of [-5..5]).

Concerning intensifiers, there are also three strategies in the literature to handle their effect on opinion expression:

- Addition and subtraction: It is the simplest way to deal with intensifiers. It consists in adding or subtracting a fixed value to/from the intensity of the opinion expression depending on the intensifier type. For example, if P_w("tired")=-3, then P_w("very tired") will be -4 and P_w("bit tired") will be -2 (Kennedy and Inkpen, 2006).

- Multiplicative factor: first introduced by Taboada et al. (Taboada et al., 2011) who considered that intensification should depend on the item being intensified. So, to each intensifying word, they have associated a percentage from -50 to 100 and created a separate dictionary for adjectival intensifiers. The polarity of an expression containing an intensifier operator is computed as $P_{exp} = P_w*(100\%+Perct_{int})$ where P_w is the polarity of the opinion word and $Perct_{int}$ is the percentage of the intensifier.

- Angle adjustment: introduced by Chardon et al. (Chardon et al., 2013) who treated intensification by increasing or decreasing the angle OḞE in their parabolic model.

Although Chardon et al. and Taboada et al. approaches have adopted different shift forms and values, they share the same strategy considering that negation is not always polarity reverser; It is basically polarity shifter and it affects also the intensity of the opinion expression. In the course of our research, we adopt the same strategy of Taboada et al. concerning

negation and intensification. As a matter of fact, we have applied polarity shift and multiplicative factor with different shift values from the one proposed by Taboada et al. since our intensity scale is different.

5.3.3 Computation of segment polarity score

After taking into account the different components affecting the sentiment classification of the segment, we have to put them together in order to compute the contextual polarity. In the literature, different heuristics are applied to perform the mapping from expression level to segment, sentence or document level. For instance, Yuan et al. (Yuan et al., 2013) proposed a simple sentiment word-count method to classify domain specific datasets. The method identifies the polarity of the text on the basis of the number of detected positive and negative opinion word. In other words, if the number of positive words bigger than the number of negative words, the text is positive; otherwise, it is negative.

Another research that addressed this issue is the work of Kim and Hovy who built and compared three models to assign a sentiment category to a given sentence (Kim and hovy, 2004). The first model computes the product of the signs of the sentiment polarities in the region (parts of the sentence in which sentiments would be considered). The second is the harmonic mean (average) of the sentiment strengths in the region, and the third is the geometric mean. The authors pointed out that the first model achieved the best overall performance.

However, the most used model to compute sentence polarity score is the sum model (Ding et al., 2008); (Oraby et al., 2013); (Tromp and Pechenizkiy, 2013); (Ghosh and Kar, 2013). It consists of summing up all opinion expression scores, upon which the result can be normalized depending on the used scale.

In the context of this research, we have used the sum model and normalized it by dividing it with the sum of the number of opinion words in the segment. Hence, $Pol_{seg} \in [-1..1]$ and it is computed according to the formula:

$Pol_{seg} = \frac{\sum_{i=1}^{n} Pol_{w_i}}{n}$ where Pol_{wi} is the polarity of the word w_i an n is the number of opinion words in the opinion expression.

5.3.4 Polarity identification

Given the polarity score of the discourse segment, a set of three rules are applied in to order to identify its polarity:

if $Pol_{seg} > 0$, the polarity of the segment is positive.

if $Pol_{seg} < 0$, the polarity of the segment is negative.

if $Pol_{seg} = 0$, the polarity of the segment is undetermined. Since, our classification scheme is binary and we do not take into account neutral segments, these segments are considered as misclassified segments when evaluating the approach.

5.4 Evaluation and discussion

In this section, we describe our performed experiments to evaluate negation scope resolution, negation effect, and the sentiment classification approach.

5.4.1 Negation scope resolution

To determine the most effective window size to use for negation scope, we conducted a set of experiments with different window sizes on OCA corpus. These experiments (Table 6) are performed with considering only negation operators (e.g. without intensifiers) and with applying polarity reversal as effect.

Table 6. Obtained results with different window sizes

Window size	Accuracy	Precision	Recall	F-score
1	70.63	67.38	89.69	76.95
2	71.39	68.33	88.85	77.25
3	71.96	69.17	87.85	77.40
4	71.83	69.30	87.04	77.16
5	71.50	69.23	86.14	76.77

Results show that, although the three-sized window has achieved the best F-score value, there is no significant difference in the classification between the different applied window sizes. This is similar to Dadvar et al. results (Dadvar et al., 2011) who have evaluated the effect of different window sizes in negation detection on 2000 movie reviews. Obtained accuracies were very close along the 5 windows. This may be explained by the fact that in movie reviews, many of sentiments are expressed implicitly. In addition, a more detailed approach about double negations and combined negation intensification has to be studied.

5.4.2 Negation effect

Two major strategies are proposed to deal with negation effect in sentiment analysis: polarity reversal and polarity shift. In these experiments (Table 7), we have evaluated these two strategies on OCA corpus by ignoring intensification and by adopting a three-sized window for negation scope as this was the size achieving the best performance in the previous experiments.

Table 7. Evaluation of the negation effects

Negation effect	Accuracy	Precision	Recall	F-score
Polarity reversal	71.96	69.17	87.85	77.40
Polarity shift	67.74	64.63	90.53	75.42

Although polarity shift strategy seems to be a more relevant strategy, it has achieved slightly worse F-score than the polarity reversal strategy. A possible explanation to this result may be the choice of the shift value. In reality, there is no rule which determines the fixed amount to shift from the intensity of the opinion expression when the negation word is encountered. This is can be a subject of an empirical study that investigates the shift value on the basis of the intensity scale.

5.4.3 Sentiment classification

In this section, we present our final classification results on OCA and COPARD2 with three-sized window as negation scope, polarity shift as negation effect, and by taking intensification into account.

Table 8. Obtained results with the proposed method

Corpus	Accuracy	Precision	Recall	F-score
OCA	70.48	67.91	87.18	76.35
COPARD2	71.41	67.79	83.58	74.86

Despite the fact that Arabic is a morphologically rich language that faces many challenging issues in sentiment analysis (Ibrahim et al., 2015), the proposed approach achieved relatively close results to the state of the art of English sentiment classification especially for OCA corpus. This proves that our build lexicon has a good coverage quality and that the implemented rules can constitute a good start for a high accurate classifier. Nevertheless, much more work is required to take into account more linguistic forms such as modal auxiliaries and conditional.

6 Conclusion and future work

In this paper, we have presented a lexicon-based approach for Arabic sentiment classification at sub-sentential level. First, we started by building a sentiment lexicon by following a semi-automatic approach. The lexicon entries were used to detect opinion words and assign to each one a sentiment class. Second, we proceeded to the annotation of new sentiment corpuses at discourse segment level. These corpuses are then used for the evaluation of the lexicon-based approach. This approach relies on an aggregation model taking into account advanced linguistic phenomena such negation and intensification. Evaluation results were considered good and not too far from state of the art results in English language.

As perspectives, we intend to enhance the lexicon quality by implementing a semantic disambiguation component based on POS information. This will improve the detected sentiment classes of the opinion words. In addition, we intend to improve existing strategies of treating negation and intensification by conducting more experiments especially on the effect of negation. Other opinion influencer cues like modalities and conditional sentences can be studied at this stage. Moreover, we intend to exploit the annotated corpuses to train a machine learning classifier for the sentence sentiment classification.

References

Abdulla et al. (2014). Nawaf A. Abdulla, Nizar A. Ahmedn Mohammed A. Shehab, Mahmoud Al-Ayyoub, Mohammed N. Al-Kabi,Saleh Al-rifai, Towards Improving the Lexicon-Based Approach for Arabic Sentiment Analysis, International Journal of Information Technology and Web Engineering, 9(3), 55-71, July-September 2014

Abdul-Mageed et al. (2012). Abdul-Mageed, M., & Diab, M. T. (2012). AWATIF: A multi-genre corpus for modern standard Arabic subjectivity and sentiment analysis (pp. 3907–3914). LREC.

Abdul-Mageed et al. (2014). M. Abdul-Mageed, M. Diab, and S. Kübler. Samar: Subjectivity and sentiment analysis for Arabic social media. Computer Speech & Language, 28(1):20–37, 2014.

Ababneh et al. (2012). Ababneh M., Al-Shalabi R., Kanaan G., Al-Nobani A.; Building an Effective Rule-Based Light Stemmer for Arabic Language to Improve Search Effectiveness, International Arab Journal of Information Technology (IAJIT) . Jul2012, Vol. 9 Issue 4, p368-372.

Aliwy. (2012). Ahmed H. Aliwy. Tokenization as Preprocessing for Arabic Tagging System. International Journal of Information and Education Technology, Vol. 2, No. 4, August 2012.

Al-Kabi (2013). Al-Kabi M.N.; Towards improving Khoja rule-based Arabic stemmer, Applied Electrical Engineering and Computing Technologies (AEECT), 2013 IEEE Jordan Conference,p1-6, 3-5 Dec. 2013, Amman, Jordan.

Al-Radaideh et al. (2014). Qasem A. Al-Radaideh, Laila M. Twaiq, Rough Set Theory for Arabic Sentiment Classification, 2014 International Conference on Future Internet of Things and Cloud.

Al-Subaihin et al. (2011). Al-Subaihin, A., Al-Khalifa, H., & Al-Salman, A. (2011). A proposed sentiment analysis tool for modern Arabic using human-based computing. the 13th International Conference on Information Integration and Web-based Applications and Services ACM.

Arafat et al. (2014). Arafat, H., Elawady, R., Baraka,S. and Elrashidy, N. Different Feature Selection for Sentiment Classification, International Journal of Information Science and Intelligent System.

Belguith et al. (2005).Lamia Hadrich Belguith, Leila Baccour et Ghassan Mourad, Segmentation de textes arabes basée sur l'analyse contextuelle des signes de ponctuations et de certaines particules, TALN 2005, Dourdan, 6-10 juin 2005.

Belguith et al. (2008). Lamia Hadrich Belguith, Chafik Aloulou et Abdelmajid Ben Hamadou, MASPAR : De la segmentation à l'analyse syntaxique de textes arabes. Revue Information Interaction Intelligence I3, vol 7, N2, mai 2008.

Benamara et al. (2012). Farah Benamara, Baptiste Chardon, Yannick Mathieu, Vladimir Popescu, Nicholas Asher, How do Negation and Modality Impact on Opinions?, Workshop on Extra-Propositional Aspects of Meaning in Computational Linguistics (ExProM-2012).

Boudabous et al. (2013). M.M. Boudabous, N. Chaâben Kammoun, N. Khedher, L. Hadrich Belguith, F. Sadat, "Arabic WordNet semantic relations enrichment through morpho-lexical patterns", ICCSPA'13, , February 12-14, Sharjah, UAE, 2013.

Chardon et al. (2013).Baptiste Chardon, Farah Benamara, Yannick Mathieu, Vladimir Popescu, Nicholas Asher.Sentiment Composition Using a Parabolic Model. In International Workshop on Computational Semantics (IWCS 2013), Potsdam Germany.

Chardon. (2013). B. Chardon, "Chaîne de traitement pour une approche discursive de l'analyse d'opinion", Phd dissertion, UPS, France, 2013

Councill et al. (2010).I. G. Councill, R. McDonald, and L. Velikovich, "What's Great and What's Not: Learning to Classify the Scope of Negation for Improved Sentiment Analysis," in Proceedings of the Workshop on Negation and Speculation in Natural Language Processing.

Dadvar et al. (2011). Dadvar, Maral and Hauff, Claudia and Jong de, Franciska (2011) Scope of negation detection in sentiment analysis. In: Dutch-Belgian Information Retrieval Workshop.

Daille et al. (2011). Daille, Béatrice, Estelle Dubreil, Laura Monceaux, and Matthieu Vernier. Annotating Opinion evaluation of Blogs: The Blogoscopy Corpus. Language Resources and Evaluation 45 (4) (June 29): 409–437.

Ding et al. (2008). X. Ding, B. Liu, and P. S. Yu. A holistic lexicon-based approach to opinion mining. In Proceedings of the 2008 International Conference on Web Search and Data Mining.

Elarnaoty et al. (2012). Elarnaoty, M., AbdelRahman, S., & Fahmy, A. (2012). A machine learning approach for opinion holder extraction in Arabic language.

ElSahar and El-Beltagy. (2015). H. ElSahar and S. R. El-Beltagy. Building large Arabic multidomain resources for sentiment analysis. In Computational Linguistics and Intelligent Text Processing.

Ghorbel and Jacot (2011). Ghorbel H., Jacot D. Sentiment Analysis of French Movie Reviews. In Studies in Computational Intelligence - Advances in Distributed Agent-Based Retrieval Tools. Ed. Springer, Vol 361. pp 97–108.

Ghosh and Kar (2013). Ghosh M., Kar A., Unsupervised Linguistic Approach for Sentiment Classification from Online Reviews Using SentiwordNet 3.0, International Journal of Engineering Research & Technology (IJERT), vol.2 Issue 9, September - 2013.

He and Zhou. (2011). He, Y., & Zhou, D. (2011). Self-training from labelled features for sentiment analysis. Information Processing & Management, 47(4), 606–616.

Heintz (2010). Heintz I., Arabic Language Modeling with stem-derived morphemes for automatic speech recognition, Phd dissertation, The Ohio State University, USA.

Hogenboom et al. (2011). A. Hogenboom, P. van Iterson, B. Heerschop, F. Frasincar, and U. Kaymak, Determining Negation Scope and Strength in Sentiment Analysis, in Systems, Man, and Cybernetics (SMC), 2011 IEEE International Conference on, 2011, pp. 2589–2594.

Ibrahim and Salim. (2013). Ibrahim, M. and Salim, N. opinion analysis for twitter and Arabic tweets: a systematic literature review. Journal of Theoretical and Applied Information Technology.

Ibrahim et al. (2015). Hossam S. Ibrahim , Sherif M. Abdou and Mervat Gheith, Sentiment analysis for modern stanadard Arabic and colloquial, International Journal on Natural Language Computing.

Kennedy and Inkpen. (2006). A. Kennedy and D. Inkpen. Sentiment classification of movie reviews using contextual valence shifters. Computational Intelligence, 22(2):110–125, 2006.

Keskes et al. (2014). Iskandar Keskes, Farah Benamara and Lamia Hadrich Belguith. Splitting Arabic Texts into Elementary Discourse Units. Journal ACM Transactions on Asian Language Information Processing. Volume 13, Issue 2, June 2014.

Keskes. (2015). Iskandar Keskes, Discourse Analysis of Arabic Documents and Application to Automatic Summarization, Phd dissertation, UPS, France, 2015

Khalifa et al. (2011). I. Khalifa, Z. Feki, A. Farawila, Arabic discourse segmentation based on rhetorical methods, In Electric Computer Sciences. 11, 1, 2011

Khoja and Garside (1999) Khoja S. and Garside R., Stemming Arabic Text, UK: Computing Department, Lancaster University.1999

Kim and hovy. (2004). Soo-Min Kim, Eduard H. Hovy: Determining the Sentiment of Opinions. COLING 2004.

Liu et al. (2014). Yang Liu, Xiaohui Yu, Bing Liu, and Zhongshuai Chen, Sentence-Level Sentiment Analysis in the Presence of Modalities, CICLing 2014, Part II, LNCS 8404, pp. 1–16, 2014.

Liu. (2012). Liu, B. Sentiment Analysis and Opinion Mining, Morgan and Claypool Publishers.

Morante and Sporleder. (2012). Roser Morante, Caroline Sporleder: Modality and Negation: An Introduction to the Special Issue. Computational Linguistics 38(2): 223-260 (2012)

Narayanan et al. (2009). Narayanan, R., Liu, B., Choudhary, A.: Sentiment analysis of conditional sentences. In: EMNLP, pp. 180–189. Association for Computational Linguistics (2009)

Oraby et al. (2013). Shereen Oraby, Yasser El-Sonbaty, Mohamad Abou El-Nasr:Finding Opinion Strength Using Rule-Based Parsing for Arabic Sentiment Analysis. MICAI (2).

Pak and Paroubek. (2010). Pak, A., & Paroubek, P. (2010). Twitter as a Corpus for Sentiment Analysis and Opinion Mining. in Proceedings of LREC'10, Valletta, Malta, 2010.

Pang et al. (2002). B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up? Sentiment classification using machine learning techniques," in Proceedings of EMNLP 2002.

Pasha et al. (2014). A. Pasha, M. Al-Badrashiny, M.T. Diab, A. El-Kholy, R. Eskander, N. Habash, M. Pooleery, O. Rambow, R. Roth, "MADAMIRA: A Fast, Comprehensive Tool for Morphological Analysis and Disambiguation of Arabic". LREC 2014.

Prollochs et al. (2015). Nicolas Prollochs,Stefan Feuerriegel, Dirk Neumann, Enhancing Sentiment Analysis of Financial News by Detecting Negation Scopes, 48th Hawaii International Conference on System Sciences, 2015 .

Read and Carroll. (2009). Read, J., & Carroll, J. (2009). Weakly supervised techniques for domain-independent sentiment classification. the 1st international CIKM workshop on Topic-sentiment analysis for mass opinion (pp. 45-52). ACM.

Rushdi-saleh et al., (2011) Rushdi-Saleh M., Martín-Valdivia M. T., Ureña-Ló L. A., Perea-Ortega J. M.. OCA: Opinion corpus for Arabic. Journal of the American Society for Information Science and Technology, 62(10), 2045–2054.

Shoukry and Rafea. (2012). Shoukry, A., & Rafea, A. (2012). Sentence-level Arabic sentiment analysis. Collaboration Technologies and Systems (CTS) (pp. 546–550). IEEE

Sawalha et al. (2008). Sawalha M. and Atwell E.S. Comparative evaluation of Arabic language morphological analysers and stemmers. In Proceedings of 22nd International Conference on Comptational Linguistics COLING 2008, 18-22 August, Manchester.

Taboada et al. (2011).Taboada, M., Brooke, J., Tofiloski, M., Voll, K., & Stede, M. (2011). Lexicon-based methods for sentiment analysis. Computational Linguistics, 37(2), 267–307.

Tang et al. (2014). Duyu Tang, Furu Wei, Bing Qin, Li Dong, Ting Liu, Ming Zhou, A Joint Segmentation and Classification Framework for Sentiment Analysis, Proceedings of (EMNLP), pages 477–487, October 25-29, 2014, Doha, Qatar.

Thelwall et al. (2010). Thelwall, M., Buckley, K., Paltoglou, G. Cai, D., & Kappas, A. (2010). Sentiment strength detection in short informal text. Journal of the American Society for Information Science and Technology, 61(12), 2544–2558.

Tromp and Pechenizkiy. (2013). Erik Tromp, Mykola Pechenizkiy: RBEM: a rule based approach to polarity detection. WISDOM 2013: 8.

Turney. (2002). Turney, P. D. (2002). Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. Proceedings of ACL 2002 (pp. 417-424).

Wiegand et al. (2010). Michael Wiegand, Alexandra Balahur, Benjamin Roth and Dietrich Klakow, Andrés Montoyo, A Survey on the Role of Negation in Sentiment Analysis, Proceedings of the Workshop on Negation and Speculation in Natural Language Processing, Uppsala, July 2010

Wilson et al. (2006). Wilson, Theresa, Janyce Wiebe, and Rebecca Hwa. 2006. “Recognizing Strong and Weak Opinion Clauses.” Computational Intelligence 22 (2): 73–99

Xia et al. (2011). R. Xia, C. Zong, S. Li, "Ensemble of feature sets and classification algorithms for sentiment classification", Information Sciences 181.

Yang and Cardie (2014). "Bishan Yang, Claire Cardie,Context-aware Learning for Sentence-level Sentiment Analysis with Posterior Regularization, In Proceedings of the ACL 2014.

Yuan et al. (2013). Yuan, Ying Liu and Hui Li, Sentiment Classification in Chinese Microblogs: Lexicon-based and Learning-based Approaches, International Proceedings of Economics Development and Research, V68. 1., 2013.

Alexander Magidow

A relational database model and prototype for storing diverse discrete linguistic data

1 Introduction

This article describes a model for storing multiple forms of linguistic data within a relational database as developed and tested through a prototype database for storing data from Arabic dialects. A challenge that typically confronts linguistic documentation projects is the need for a flexible data model that can be adapted to the growing needs of a project (Dimitriadis, 2006). Contributors to linguistic databases typically cannot predict exactly which attributes of their data they will need to store, and therefore the initial design of the database may need to change over time. Many projects take advantage of the flexibility of XML and RDF to allow for continuing revisions to the data model. For some projects, there may be a compelling need to use a relational database system, though some approaches to relational database design may not flexible enough to allow for adaptation over time (Dimitriadis, 2006). The goal of this article is to describe a relational database model which can adapt easily to storing new data types as a project evolves. It both describes a general data model and shows its implementation within a working project. The model is primarily intended for storing discrete linguistic elements (phonemes, morphemes including general lexical data, sentences) as opposed to text corpora, and would be expected to store data on the order of thousands to hundreds of thousands of rows.[1]

The relational model described in this paper is centered around the linguistic datum, encoded as a string of characters, associated in a many-to-many relationship with 'tags,' and in many-to-many named relationships with other datums.[2] For this reason, the model will be referred to as the 'tag-and-relationship' model. The combination of tags and relationships allows the database to store a wide variety of linguistic data.

This data model was developed in tandem with a project to encode linguistic data from Arabic dialects (the "Database of Arabic Dialects", DAD).[3] Arabic is an extremely diverse language group, with a dialects stretching from Mauritania to Afghanistan,

[1]The author would like to thank Yonatan Belinkov for his assistance with the initial phases of this project, and the two anonymous reviewers who provided extremely helpful feedback both for this article and for the associated web project, as well as Nicholas Coulombe for his insightful comments on early drafts. All remaining errors are the author's own.

[2]I will use the plural 'datums' as a plural of 'datum' for referring to a countable, individuated set of pieces of data, as opposed to 'data' which refers to a general, unindividuated collection. For example, one could discuss 'hundreds of datums', but write in general about the 'data' that needs to be inputted into a system.

[3]`http://database-of-arabic-dialects.org/`. The acronym is meant to evoke the Arabic letter *ḍād,* originally a pharyngealized voiced alveolar lateral fricative [ɮˤ], traditionally considered characteristic of Arabic, which is sometimes referred to as *luġat aḍ-ḍād,* 'language of *ḍād.*'

many of which are not immediately mutually intelligible with one another. Much of the diversity is lexical or morpholexical, so that closed-class words such as pronouns and open-class words such as the verb 'to go' function as shibboleths between dialects. Individual phonemes (realization of /*q/) and even phonological rules (raising of /a/ word finally) can also act as shibboleths. Since this information is scattered in a variety of different publications, the goal of the project is to develop a website which can act as a hub for researchers to input legacy and novel Arabic dialect data and visualize that data in a variety of ways (as lists, maps, paradigms, etc). It is also intended to allow for multiple researchers to use it as a research tool for inputting and analyzing their own data. Data can be made publicly available, or access may be restricted only to a researcher and selected collaborators.

2 Similar projects

The basic desiderata of the Database of Arabic Dialects project were as follows:

- Input and search of actual language data (i.e. words in Arabic dialects, not just typological meta-analysis)
- Ability to handle a wide range of linguistic data, from phonemes to short phrases, with the ability to easily add additional data types
- Ability to consider multiple relevant classifications for a given datum (e.g. a single word could be an interrogative, a pronoun, and a relative marker)
- Permission control and contributor attribution
- Intuitive and efficient interface for data input and analysis
- Publication of all project prototype source code

The initial phase of the project was to conduct a survey of existing projects with an eye to making use of their database structure or their code if they were open-source, provided they could meet the requirements outlined above.

The most similar existing project was the FIELD (Field Input Environment For Linguistic Data) tool,[4] a branch of the larger E-MELD (Electronic Metastructure for Endagered Language Data) initiative.[5] This website provided input for primarily lexical data, as well as interlinear glossed texts. However, the project appears to have been moribund since at least 2010, the last copyright date listed on the E-MELD website. The website claims that an improved version of this tool is forthcoming, but this seems not to have happened. Indeed, the database behind the project (a PostgreSQL database) appears to be broken and one can no longer access the website as of August 2015. When

[4] `http://emeld.org/tools/fieldinput.cfm`. Unless otherwise noted, all websites mentioned were accessed on August 26, 2015.

[5] `http://emeld.org/`

the project was accessible, in 2012, it featured very simple HTML (not Javascript) based input forms which proved to be extremely slow. The FIELD project is not open source, and thus little useful information can be gleaned from what remains today.

The FIELD tool was also reliant on the incomplete GOLD (General Ontology for Linguistic Description) ontology. All data had to be related to the categories in the GOLD ontology, but GOLD simply is not detailed enough to properly describe the Arabic data. For example, the most recent 2010 version of GOLD only includes demonstratives as pronominals, and makes no distinction between demonstratives which act pronominally and those which can only be determiners.[6] Arabic dialects often make this distinction (especially those in North African) and it is not an uncommon distinction cross-linguistically (Diessel, 1999).

Another project is the Vienna Corpus of Arabic Varieties (VICAV), "an international project aiming at the collection of digital language resources documenting varieties of spoken Arabic."[7] The VICAV project focuses on curated presentations of linguistic information, with attractive manually written profiles of dialects, curated lists of dialect isoglosses, dictionaries of single dialects, bibliographies and complete texts in several dialects. The project operates largely on a document-level of organization, with XML documents based on Text Encoding Initiative (TEI) standards being served largely intact as single documents. For example, the project has several small dictionaries, one for Cairene Arabic, one for Syrian, one for Tunisian and one for Modern Standard Arabic. While each dictionary can be searched individually, it does not appear to be possible via the current interface to search across multiple dictionaries or otherwise collate data between dialects.[8]

The VICAV project is similar to the DAD project, but is based on a different design philosophy. Whereas the goal of DAD is to present only raw data from a large number of dialects in a highly structured and searcheable format, VICAV is designed for presenting general information about a small number of dialects. A relational database model is more appropriate for the DAD model of data storage since it is based on the individual piece of linguistic data as the smallest datum, whereas the VICAV project is based on a document paradigm, even if individual documents contain smaller divisions.

An example of the different between these projects is how they illustrate isoglosses between Arabic dialects. There is a relatively small set of areas in which dialects tend to vary significantly. For example, most dialects have very similar interrogative systems, though the actual wordforms differ. On the VICAV website, there is a section which shows the distinctive linguistic features of two dialects (Cairene and Damascene), including interrogatives, demonstratives, pronouns, etc. These pages appear to be manually written rather than automatically generated, and show only those linguistic isoglosses which have been chosen by the editor of the pages. In contrast, the DAD

[6] `http://linguistics-ontology.org/gold/2010/Demonstrative`

[7] `http://minerva.arz.oeaw.ac.at/vicav2/`. Note in spite of the term 'corpus' in the title of the project, searchable textual data only represents a small percentage of its current resources.

[8] The project serves data from a MySQL database that is derived directly from the XML documents (Karlheinz Mörth, p.c.), so in theory it should be possible to perform cross-dialectal queries.

project allows a user to search for items with any tags, so that one could search for all words tagged with `interrogative`, providing greater comparative coverage and flexibility in the choice of variables to investigate. With automatically generated results, it is easier to include more dialects and more data. In the DAD dataset there are nearly 80 dialects which currently have comparative data, whereas it is not clear how the VICAV approach of manual curation could easily scale up to a larger number of dialects.

Another intriguing project is the Oto-Manguean Inflectional Class Database, a comparative database of verbal inflection data from twenty Oto-Manguean languages spoken in Mexico.[9] The project is of interest because these languages differ significantly in their verbal inflectional classes. They are so diverse that comparative searches are difficult to carry out as the categories between languages are not always equivalent. The challenges of storing this data could indeed inform the database design here, but the project does not appear to have documented their database structure nor have they seem to have released their source code. This is an excellent illustration of the need to document project design and to make source code available for other projects.

During the initial survey of existing projects, it appeared that the World Atlas of Language Structure[10] and the related Atlas of Pidgin and Creole Language Structures Online[11] were based on a very simple, flat database structure that was capable only of storing meta-linguistic information about languages based on scholarly analyses, e.g. typological categories, not actual linguistic data, e.g. lexemes and their meaning. Only while revising this article did it become clear that the underlying software, CLLD, is significantly more sophisticated than it first appeared to be and can indeed store full linguistic data (Forkel, 2014). Given the late addition of this information, it is impossible to integrate it completely into this article. When appropriate, reference will be made to the design decisions made by the CLLD team and how they compare with those made in developing the DAD project.

2.1 General Structural Desiderata for Linguistic Data

Though the FIELD tool itself is no longer functional, the E-MELD project produced a number of articles discussing the efficient storage of linguistic data. Farrar (2006) describes a model for a fundamental data type for storing linguistic information. His model is based on the notion of a 'linguistic sign,' consisting of a "3-tuple" with a form component, a meaning component, and a grammatical component. The form component is "any annotation entity that represents the phonetic, phonological, orthographic or otherwise physical manifestation of the sign" (p. 7). The meaning component can refer to the semantic content in the sense of the meaning of an item, but this is considered distinct from a translation in another language, which itself would be a linguistic sign. The meaning component could also encompass semantic features such as [+animate]. Finally the grammatical component includes information such as part of speech and

[9] `http://www.oto-manguean.surrey.ac.uk/`
[10] `http://wals.info/`
[11] `http://apics-online.info/`

morphosyntactic features. Only these three components are considered essential to a linguistic sign. Information such as annotations or even translations are to be modeled as relations between linguistic signs, rather than stored as components of a linguistic sign (p. 8). He also emphasizes that this should be a content-based model, rather than a display-oriented model, with display handled at the software level (in the case of their XML model, via XSL transformations).

Penton et al. (2004) discusses how to store paradigmatic information, a category into which much descriptive linguistic data falls, from phonemes to open-class lexemes. In exploring paradigmatic elements, they find that paradigms "simply represent an association between linguistic forms and lingustic categories " (p. 6). The tabular display of a paradigm is essentially an algorithm which places linguistic signs in the appropriate cell based on their linguistic properties, typically expressed in the labels above or beside it in the table. From the perspective of data storage, it is sufficient to store only the appropriate characteristics of the linguistic datum. A linguistic datum need not be 'aware' of its place within a paradigm. Only when it comes to displaying the data is there a need for the algorithm that will place that data into a table. In the terms of Farrar (2006), the meaning or grammatical components of the linguistic sign can store the necessary data for transforming multiple datums into any number of paradigms.

Good and Hendryx-Parker (2006) discuss a model for encoding the potentially contested relationships between the world's languages. Their database model consists of nodes (corresponding to "langoids") which have a primary key (a unique human readable identifier), basic metadata (human readable names) and a one-to-many relationship with digital documents and books. Each node is related to other nodes by way of relationships, which take the form of an RDF predicate, a relationship which links a subject to an object by way of a 'predicate,' an human readable expression of what kind of relationship exists between them. In this model, the subject and object would be nodes. One of their primary concerns is how to represent contested information (e.g. particular arguments about the structure of a language family) without compromising the integrity of the database. By using multiple RDF links between the same elements to encode competing relationship hypotheses, the nodes do not change (lingoids) and multiple hypotheses can be stored in the same database. For example, a hypothesis that groups languages B and C as daughters of language A would have RDF links between A (subject) and both B and C as objects, e.g. `A mother B`, and `A mother C`. On the other hand, if Language C is hypothesized to be a daughter of Language B, which in turn is a daughter of Language A, then there would be a RDF link of `A mother B` and `B mother C`, implying that A is the grandmother of language C. The database would store both sets of relationships. [12]

They use RDF links for linking to other contestable metadata. They link between a language and its 'language type' (language area, language family, language, dialect, etc) with an `is of language type` predicate. This allows for encoding whether a

[12] The two hypotheses would be marked in such a way that they are distinguishable, presumably through the use of reification to allow the relationship itself to be treated as a subject or object in another relationship.

researcher considers Chinese, for example, to be a language family (with Chinese varieties considered languages in their own right) or a language (with Chinese varieties to be considered dialects). The database is implemented in an Object-Oriented Database (Zope Object Database) which supports RDF relationships natively. Thus, in this data model, contradictory pieces of information in the same category (i.e. classification) are stored as overlapping relationships between nodes.

Lewis et al. (2006) present important considerations for citation, fair use and digital distribution of linguistics data. They suggest several important principles for data storage and attribution: full attribution (indication of the full citation for a datum), sheltering of data (providing tools to limit access to data) and acknowledgment of ownership (acknowledge additional ownership for data if it has changed from the original source). To follow these principles for a publicly accessible collection of descriptive linguistic data like DAD, the project should provide tools for fully citing data and controlling permissions for access to that data. There should also be acknowledgment for what they term "enrichment," adding to the data in a meaningful way. In the case of a crowd-sourced database, the very act of inputting the data (which often includes regularizing notation, adding annotations, geolocation, etc.) should be considered a form of enrichment and the person who performs this, even with legacy data found in published sources, should receive credit.

3 Tag-and-Relations Database Model

The data models from the E-MELD projects are mostly implemented in XML, which is common as a data storage and exchange format in digital humanities projects. XML is a very adaptable format which has the advantage of being relatively easy to change after the initial design phase of a project. However, there are a number of reasons why a project may prefer to use a relational database model, ranging from personnel expertise to software support to support within online communities.[13] For users of relational databases, a flexible database design is needed to adapt to changing requirements as a project evolves, preferably with a minimum of changes to the basic organization of the database's tables.

The models described by Farrar (2006) and Good and Hendryx-Parker (2006) both take a single, irreducible form as the central data element in the database. In the former, this is the linguistic sign, while in the latter it is the language node. Both models also allow for named relations between datums. These named relationships can be multiple and redundant — multiple relationships can express the same information with variations that represent different analytical interpretations of the data. Each of their models also allows for certain core metadata to be associated with each datum.

[13] In this particular project, which was unfunded, the sole developer's expertise was primarily in relational databases, while the software tools and hosting services that seemed best suited for the overall project worked with relational database software. The CLLD project made a similar decision to use relational database software rather than use an RDF graph database due to the latter's "non-standard requirements in terms of deployment, administration and maintenance" (Forkel, 2014, 3).

The data model describe here, therefore, builds on the central element of a transcribed linguistic datum, i.e. the actual linguistic signal, equivalent to the 'form component' of the linguistic sign in Farrar (2006). The linguistic datum, and its associated metadata, are stored as rows in a single table. Similar to those other models, every linguistic datum can also be related to other linguistic datums via a named relationship. In a relational database, this is easily operationalized as a through table with two foreign keys, both pointing to the table of linguistic datums, and with an additional field for naming the linguistic relationships.

An important concern is how to store the properties associated with each datum. All datums will have certain identical properties: a gloss or other indication of meaning, a bibliographical citation for where the information was actually contained, in order to give proper credit, the name of the language or dialect that the datum is drawn from, information about the transcription scheme or encoding used to transcribe the datum if that is relevant), a human readable annotation giving information about the datum for users, etc. Since this data is common to all datums, it can be included in the linguistic datum table as additional fields, some of which may constitute foreign keys to other tables (e.g. to a bibliographic reference table).

This contrasts with the approach recommended in Farrar (2006). In his model, much of the metadata is linked to a linguistic datum with a relational link. For example, a gloss is treated as a linguistic datum in and of itself, and a relational link connects the datum in the primary language of interest to the gloss datum in the language of translation. However, this is unnecessarily complex when a database is designed for storing data in a single language. For most purposes, it is much simpler to only store data in the language of interest, and to simply include the gloss (or glosses) within the same table as the linguistic datum, since they are functionally inseparable.

A greater difficulty comes in recording linguistic properties of datums. A single datum might belong to multiple classes of categories. For example, an interrogative may also function as a pronoun and a relativizer. Conversely, a single property may apply across many classes of datum. Pronouns, nouns and verbal agreement affixes all share properties of gender and number in many languages. A project may decide to add in classifications after the initial design and input of data that have not previously been encoded in the database. A synchronically oriented database may decide to include historical classifications, for example, and there might be multiple and contended classifications that need to be included. In an XML based approach, it is straightforward to add new attributes to each datum, but in a relational database it is difficult to organize the data in such a way that we can provide that level of flexibility.

A common approach to database design uses individual tables for each category of datum. In such a model, different parts of speech might each inhabit separate tables, with each table containing fields unique to that part of speech. A noun table could contain a field for gender, number, etc. For languages with unpredictable plurals, that too could be included in the same row as the singular as a separate field. However, it is difficult for such a model to handle datums that belong to different categories simultaneously, or to unite properties that are shared across tables. New categories

require alteration of the basic structure of the database, and a small number of members in a given table may require a property that is not shared by the majority of members of that table.

The approach that is used in the model here is instead to store every linguistic datum in a single table, and to mark each datum with an unlimited number of tags. Tags are simply text fields, and as such they are flexible enough to store almost any kind of data. One datum could be marked with the tags `noun, feminine, dative`, another with `demonstrative, pronoun, adjective, deictic, article`, all while being stored in the same table. The textual tag system can easily be extended to include values by way of a delimiter, so one could mark `gender=masculine` or `gender=feminine`, in a manner very similar to XML attributes.[14] This also allows for searches on whether or not an item has a gender at all. In a small number of Arabic dialects, the interrogatives for 'who' and 'which' can be marked for gender and searching for they keyword `gender` would allow a researcher to find those dialects, rather than having to search for both `masculine` and `feminine`. The tag system makes it straightforward to mark these rare forms with the appropriate properties, which would be more difficult in a table-per-POS approach. Note that the database structure itself does not specify how structured these tags must be. Depending on the requirements of a project, they could even be in XML-like form, e.g. `attribute = 'gender' value = 'masculine'` or any other system that could easily be parsed by the underlying software.

All of the models discussed in the previous section make use of named relationships between linguistic datums, and this model does so as well. A datum might be a variant of another datum, e.g. allophones, or a datum, e.g. a sentence, might exemplify another datum, e.g. a word. Named relationships between datums allow for the system to express an infinite variety of interrelations between the datums in the system.

The database system, then, is based on two simple mechanisms: Tags, which are applied to individual language datums, and relationships, which are named links between datums, so we can refer to this is a 'tag-and-relationship' database model. A schematic of the basic database structure is show in Figure 1. In the schematic, tags are used to mark both the predicates of relationships and the properties of datums — note that these are separate sets of tags. Only the linguistic datums are given relationships to one another in this current model, though a tag-and-relationship model could certainly be applied to other entities. For example, if storing hypothesized linguistic relationships was a priority, the tag-and-relationship model could easily be applied to a table that stores linguistic entities (i.e. languages or dialects). Both tags and relationships can be multiple, overlapping, and contradictory. This model therefore can support multiple analyses of the same material in a manner similar to the model described by Good and Hendryx-Parker (2006).

[14] For the DAD project, the period was chosen as a delimiter for different 'parts' of a tag. This proved to be problematic when performing searches with regular expressions, as the period is a metacharacter and must be escaped. The solution thus far has been to do searches as basic string searches instead, but the delimiter could easily be changed.

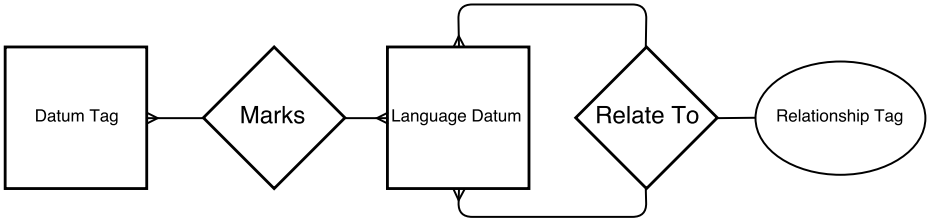


Abbildung 1: A diagram of the basic tag-and-relationship relational database model in entity-relation notation.

The content of tags is not limited to a particular area and this is the basis for the extremely flexibility of the tag-and-relationship model. Farrar (2006) argues for a separation of meaning and grammatical metadata, but there is no obvious reason to do so outside of a theoretical model of a linguistic sign. Tags can encode semantic data, such as `animacy=nonhuman`, and can also encode grammatical data such as `POS=noun`. If for some reason a separation between the two *is* necessary, it is simple to add that information into the tag along with a delimiter, for example `sem:animacy=nonhuman` or `gram:POS=noun`. Tags can mark other domains as well. A very inclusive project may contain complete datasets of a very specific nature. For Arabic, one dataset that may be included at a future date on the DAD website contains the babytalk (caregiver child-directed speech) equivalents of normal words in several Arabic dialects.[15] For someone who is not searching specifically for those items, that dataset may be of little use, so having all items from that dataset tagged in a particular way (e.g. `dataset=babytalk`) allows for users to easily include or exclude that data. Also, while we have earlier explored the issues with standardized ontologies such as GOLD, such ontologies can easily be used as the basis for the tag system while still being extensible beyond those standards if necessary.

The tag system also allows for inclusion of data which is underspecified, in contrast to a model which has a table-per-POS model. Manfred Woidich and Peter Behnstedt have granted the author access to the flat, spreadsheet style dataset that underlies their Wortatlases of Arabic dialects (Behnstedt and Woidich, 2010). This data is extremely messy, with almost no structured information on parts of speech. While part of speech can sometimes be inferred by data structure (e.g. some fields give the standard verbal citation forms separated by a comma), the majority of the data will necessarily be imported without any information on part of speech. This will make searching the data more difficult for scholars, but does not pose a serious problem for the database structure. Those datums which have POS information can be tagged accordingly, and those which do not simply will not be tagged.

With such an underspecified structure, the tag-and-relationship model requires each project to establish general principles of best practice. The primary criterion for how to

[15] See `http://babytalk.barefootlinguist.com/`

store data is whether it will be retrievable with a query. This is important both at the application level (computer-computer interaction) and the interface (human-computer interaction) level. For the data to be displayed, it must be sufficiently well tagged and interrelated that the frontend application can access it. For human interaction, the data must be accessible in a way that a human user can easily construct searches and read the results of those searches.

It is straightforward to store anything from a phoneme to affixes to open-class lexical items with this model, though a given project will have to decide exactly how to model some items. In essence, the tag-and-relationship model is fixed, but individual projects must design how they model their data within the system. In the DAD implementation, as we expect would be the case in most implementations, the gloss of a linguistic datum is a required field. For a phoneme, this gloss could be basically uninformative, e.g. `phoneme`, with the majority of the data stored in tags (e.g. `phoneme, bilabial, unvoiced, stop`). The gloss could be more elaborate, with the human readable and searchable, `unvoiced bilabial stop`. It would be best to encode those same properties as tags to allow for application level interactions with the data, as the application should be able to retrieve the entire class of stops without accidentally including lexemes glossed with, for example, `to stop`. Outside of phonemes, most items tend to be more straightforward in having a meaningful gloss.

Sentential or higher level data becomes more complex. An individual idiom or sentence can easily be stored, with the gloss operating as a free translation. More complex sentential data could simply be stored as XML, with tags aimed at the application to tell it that these datums need to displayed and searched in a manner different from data stored in plain text.[16] A more complex example would be interlinear glossing. Farrar (2006) discusses the issue of storing interlinear glosses in XML but it is not clear how they should be stored in this system. Interlinear glosses normally have three levels of information, the transcribed data from the language, a morpheme-by-morpheme gloss, and a free translation. Occasionally they have more levels depending on the complexity of the example. All three levels could be stored in the transcribed data field of an individual datum as XML. Alternatively, each level of the interlinear gloss could be its own datum, linked together with relationships such as `IL.morphemegloss` and `IL.freetrans`. The gloss fields could be empty or could have placeholder information.[17]

Relationships are best used when the application code must keep related datums together, though co-occurring linguistic forms such as paradigmatic data do not necessarily need to be explicitly linked. As Penton et al. (2004) have shown, paradigms are simply the intersection of different traits, which in this model would be stored as tags. The application would render the paradigm based on those tags, and there is no practical need for the members of that paradigm to be 'aware' of one another through

[16] It is not necessary that all data be stored in the same way, provided that the encoding is clearly marked. If the vast majority of transcribed data is in the form of a few characters of transcription from the language of research, it is better to not have redundant XML code used in every single entry. Instead, the exceptional entries could be marked as such.

[17] The CLLD project handles this issue by modeling sentential data as an entirely separate data type from morpholexical or typological data (Forkel, 2014).

relationships. On the other hand, if two members of a paradigm are stored as separate datums and co-vary, they should be linked so that they can be properly aligned during display. For example, in Arabic present-tense verbs, some conjugations of the verb take a circumfix, so in Syrian Arabic 'they write' is *yi-ktub-ū* 3M-write.present-PL. If a speaker is speaking more formally, they might say *ya-ktub-ūn,* with the same meaning but a change in the form of both the prefix and the suffix. In that case, the prefix and suffix datums should entered as separate datums, since a linguist may be interested in seeing only prefixes or suffixes, but they should be linked with relationships so they can be properly aligned upon display. While in principle it is best to provide too much marking for the data, rather than too little, there is a trade-off with programming complexity, as more complex queries are needed when both tags and relationships must be queried.

Relationships are useful for storing data such as allophones and allomorphs, since there is a clear base form and a clear variant form. It is also straightforward to mark the relationship between more and less common forms, since the more common form can act as the base form. Similarly, for the babytalk data, it is reasonable to consider the adult words to be the primary form, and to mark a relationship so that a babytalk form is the subject of a `baby talk variant of` predicate to an adult lexeme.

When there is a more equal relationship between datums, it is not always clear whether relationships should be marked in both directions (i.e. each datum has a relationship to the other for a total of two relationships), in only one direction (with queries tracing both sides of the relationship) or not at all. For example, many oblique pronoun suffixes in Arabic have allomorphs depending on the preceding phonological environment, usually with one form occurring in post-vocalic position, and one form after consonants. Neither form is clearly the base form. Ultimately in the DAD project it was found that since these allomorphs are inputted and displayed in paradigms, it was not necessary to link them with a relationship, as they are retrievable based on their tags alone.

4 Modeling Data in DAD

The model described in the previous section is a general model which can be instantiated in a variety of ways, depending on the needs of a project. In this section, we will use the Database of Arabic Dialects (DAD) website and database to illustrate the model and to discuss the details of some of the design decisions that are part of a large scale project of this nature. The goal of the DAD project is to provide a crowd-sourced website with comparative data for Arabic dialects. The web interface provides both the tools for data input and for data viewing and analysis. The website is implemented using the open-source, Python-based, model-view-controller Django web framework. Django provides the interface between a PostgreSQL database backend and the web interface, and specializes in allowing for database design and querying using Python code. Django is also valuable for an unfunded pilot project since it automatically provides a relatively

efficient data entry interface for all tables present in the database. The DAD project is open source, and the code is hosted on GitHub.[18]

The central table in the database represents the linguistic datum. This table utilizes the tag-and-relationship model described above.A simplified schematic of the database model from the DAD project is included in Figure 2. In the DAD implementation this table includes the following fields:

normalizedEntry A text field of unlimited length for storing transcribed linguistic data

normalizationStyle This field indicates which of the standard transcription styles are used for the `normalizedEntry` field

gloss An HSTORE field in the current implementation which stores key–value pairs. This allows for storing glosses from multiple languages

entryTags A foreign key, allowing a many-to-many relationship to tags

relationships A foreign key to an intermediate table which enables a many-to-many relationship with other linguistic datums. The intermediate table has a field which is a foreign key to a table of 'relationship tags' used to mark the nature of the relationship

dialect A foreign key to a dialect table

sourceDoc A foreign key to a bibliography table

sourceLoc A text field for storing information about where an item is located in the source document

contributor A foreign key, allowing a one-to-one relationship to a contributor

permissions A field which stores a permission string, marking data as "Private" (only the contributor and collaborators can see it), "Public" (any visitor to the website can see it) or "Public No Export" (any visitor can see the data, but it cannot be exported from the website)

originalOrthography An optional text field for storing the item in the original transcription in the source

annotation An optional text field for storing human readable additional information or reflections on the data

This table is described in detail since it forms the heart of the database, but also because it represents some important design decisions. First, note that it conforms to several of the recommendations from the literature discussed previously: It makes the actual linguistic data the primary piece of information, it contains information

[18] `http://github.com/amagidow/dialects`

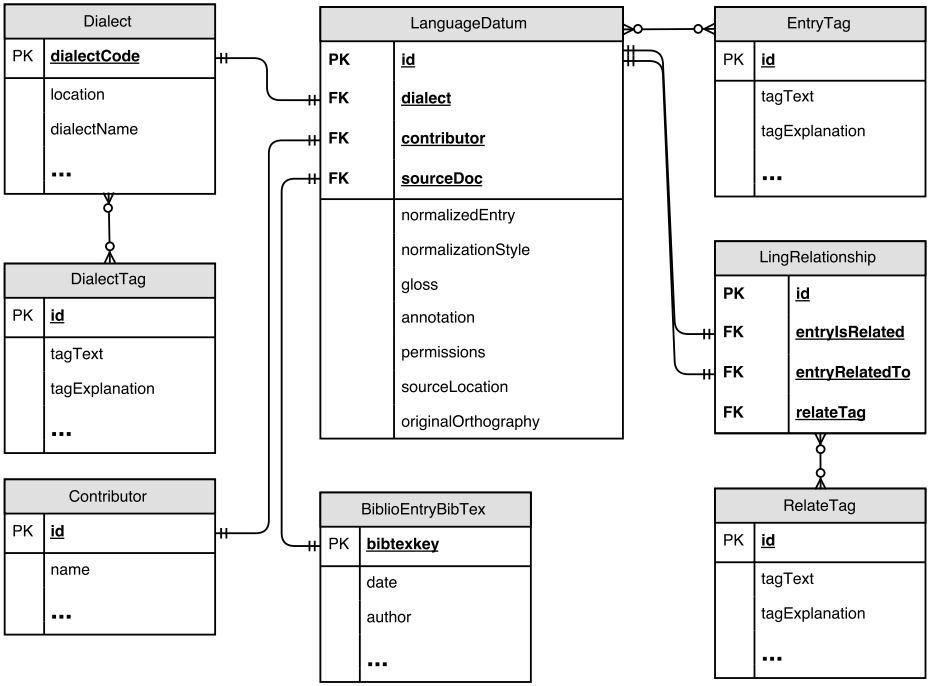


Abbildung 2: A simplified diagram of the DAD database structure. Ellipses indicate columns which exist in the database but are not shown explicitly on this chart.

about the original source and it assigns credit to the data contributor, who also has fine grained, changeable permission control. That is to say, it provides credit both to the original, published source of data, and to the contributor who provided what Lewis et al. (2006) termed the "enrichment" of digitizing that data on the website.

Second, it was necessary to make important decisions about normalization. In this table, the `sourceLoc` field represents a violation of third normal form, when "a non-key field is a fact about another non-key field." (Kent, 1983, 121). In theory and in practice, many datums come from the same source (a single page, or a single map in a volume) so normalizing this information into an intermediate table would be the most data-efficient way to store it. This would eliminate repetition and allow for more consistent update in case of errors. However, an intermediate table greatly increases the complexity of queries and of the database, and the citations are meant for human users, so some imprecision is acceptable. Leaving this unnormalized also does not represent a huge increase in duplicated data, since the `sourceLoc` field is rarely more than a few characters in length (e.g. `p. 15`) and the `sourceDoc` field is a very efficient integer foreign key field. As

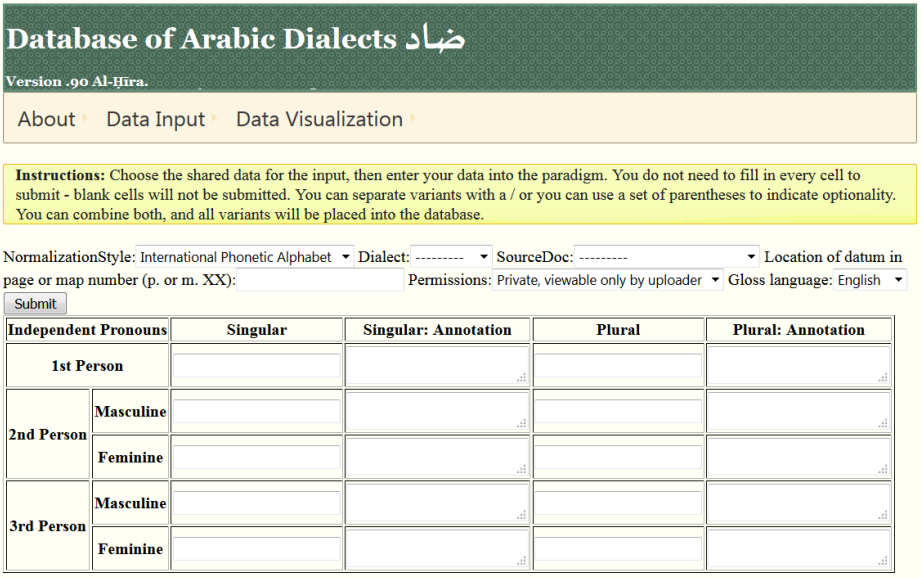
Dimitriadis (2006) notes, linguistic databases are tiny in comparison to most commercial databases, typically with under a million records, and therefore storage efficiency is not as important as it might be in a database with millions of rows.

The tags for linguistic datums are stored in a separate table from the linguistic datums, with an intermediary table to allow for a many-to-many relationship between tags and datums. Each tag has a both the text of the tag, as well as an explanation of the tag's use. This allows for more consistent application of tags. The same is true for the tags used to mark linguistic relationships. The `Dialect` table also links to tags with explanations, so that entire dialects can be tagged to indicate properties of the dialect, or hypothesized classifications. For example, Arabic dialects are traditionally classified as urban, rural or nomadic, so they are tagged accordingly, and a researcher could also tag dialects according to their own hypothesized classifications.

The bibliographic entry table is another area where some compromise was needed. Ideally we would be able to take advantage of existing software that has a clean interface for interacting with a database backend. We could link directly to a centrally stored bibliography database and avoid the need to implement our own. The widely used, open-source program Zotero seemed like a strong candidate, but its database structure has no persistent key analogous to a key in BibTex. Though each entry has an integer primary key in the database, if an item were accidentally deleted (a not uncommon occurrence when using the program's interface), any links to that primary key would be lost and the integer key would provide no clue as to the original source. Instead the built-in interface provided by Django (referred to as the "admin interface") is used for entering bibliographic items. All bibliographic entries must have a unique, human readable key. Should a bibliographic entry somehow be lost, the human readable key retained in the linguistic datum table would still provide information that would allow for reconstructing the bibliographical entry. A field in the `BiblioEntryBibTex` table also allows for entering a full BibTex entry, as this at least constitutes a defacto standard for storing bibliographical references. One limitation of this implementation is that the bibliographic entry table was designed around published works (with fields for author, title, publisher, etc.) and it is unclear how it should be modified to accommodate elicted field data, as the DAD project is intended to eventually accommodate field researchers. Going forward, the bibliographic entry table will probably need to be redesigned.

4.1 Flexibility and Data Integrity

Since most of the information about a linguistic datum is stored by way of tags and relationships, the system cannot always take advantage of the data integrity tools that are characteristic of relational databases, such as constraints or triggers. This is a major trade-off of the flexibility of the tag-and-relationship model. There is nothing keeping contributors from accidentally adding duplicate or synonymous tags, or to enforce the use of relationships in a consistent manner. If a more rigid structure had been used, such as a design in which each part of speech has its own table, there



Database of Arabic Dialects ضاد

Version .90 Al-Ḥīra.

About | Data Input | Data Visualization

Instructions: Choose the shared data for the input, then enter your data into the paradigm. You do not need to fill in every cell to submit - blank cells will not be submitted. You can separate variants with a / or you can use a set of parentheses to indicate optionality. You can combine both, and all variants will be placed into the database.

NormalizationStyle: International Phonetic Alphabet Dialect: --------- SourceDoc: --------- Location of datum in page or map number (p. or m. XX): Permissions: Private, viewable only by uploader Gloss language: English

Submit

Independent Pronouns		Singular	Singular: Annotation	Plural	Plural: Annotation
1st Person					
2nd Person	Masculine				
	Feminine				
3rd Person	Masculine				
	Feminine				

Abbildung 3: Screenshot of DAD paradigm input.

would be stronger built-in control over data validity, but this would bring with it the disadvantages described above.[19]

The solution for this issue in the DAD project has been to enforce uniformity at the interface level. Data input can be performed in several ways, but since most of the data currently being entered into DAD is paradigmatic, the data entry is itself in the forms of paradigms. Figure 3 shows the input interface for the independent (nominative) personal pronouns. From the user perspective, they are simply entering data into a table-like entity, not unlike how they would enter data into a table while writing an article in a word processor. The web application adds the appropriate tags as it saves the data into the database. The user does not directly adds the tags themselves. If necessary, a user can later go in and edit individual datums to modify the tags. For example, an input page might be missing a category that is found only very rarely, and so the user can go and add in that category data after first taking advantage of the paradigmatic input page. The user can still be restricted to only using the existing tags, helping maintain the integrity of the tag system. The addition of new tags can be restricted to a trusted set of users, or added by administrators upon user request.

[19] The model described by Dimitriadis (2006) uses a table of features and lists of values for each feature to strike a balance between flexibility and automatic constraint, though it is still possible for a user to accidentally add a redundant feature, just as it is possible for users to add redundant tags in the tag-and-relationship model.

The paradigms input and display pages are themselves based on Python code which specifies the vertical and horizontal headers for the paradigm, and the tags and relationships that should be applied to the datum in each cell. This means that creating a bespoke paradigm based on user demand is a simple process that can be accomplished very quickly and with relatively little technical skill. Strategically it is better and easier to build an interface that itself enforces data integrity than it is to allow a user to input messy data that must later be cleaned.

Other types of validation could be performed either by the SQL backend or the web interface. Duplicate entries could be disallowed by the web interfaced or restricted in the database using uniqeness constraints. Ultimately, of course, there is no foolproof method for ensuring clean data input, but if a given contributor often submits poor quality data, their contributions can be excluded from a search, or their user privileges revoked.

For data retrieval, the strategy has been to provide as much hinting as possible. There are a number of data views, but most of the views allow for searching against the Arabic datum itself, the gloss, the annotation, and the tags. For both the gloss and the tag fields, the website uses Javascript to provide auto-complete suggestions as the user types. This allows for the user to explore the system of tags, as well as common glosses, while they are in the midst of a search. Another page provides a complete list of tags and their explanations, but the auto-complete suggestions are intended to make it unnecessary for most users to visit that page.

4.2 Storing Diverse Data

It may be helpful to illustrate how a variety of different data types might be stored using the DAD implementation of the tags-and-relations model. The primary type of data that has been stored in DAD thus far is closed-class morpholexical items such as demonstratives, pronouns and interrogatives. In general, it has been straightforward to store this data within the tag-and-relationship model. For almost all of this data, only tags are necessary for input and retrieval, and so relationships have generally not been used.

The DAD project is flexible enough that it could also store data from other projects. The lexical data from the VICAV project is a good example. A sample dictionary entry is shown in Figure 4. The entry has a headword, basic grammatical information, the triconsonantal root of the word [ʕyn], multiple possible plural forms, multilingual definitions and two idiomatic phrases. In the DAD system, the singular and all three of its plural forms would each constitute a linguistic datum, as would each of the idioms, for a total of six linguistic datums. The appropriate properties of each of the forms would be marked with tags, e.g. `noun, feminine, root.`ʕyn The plural forms would be linked to the singular with the relationship `pluralOf`, and the idioms would be linked to the singular (and possibly to the plurals) with the relationship `idiomContaining`. The gloss fields are multilingual, and so could store the English and German glosses.

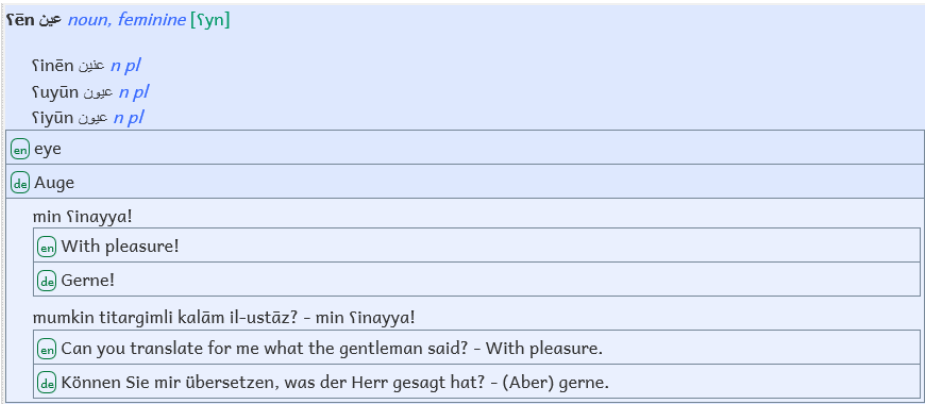


Abbildung 4: A screenshot of a typical entry from the VICAV website.

Note that the singular and plural *must* be linked with relationships since Arabic plurals are unpredictable from the singular.

The lexical data from the Babytalk project could also be stored in the DAD database. Figure 5 shows sample data from this website. The actual babytalk word would be stored as a datum, as would the adult equivalent, with part of speech data marked with tags and both would share similar glosses (though 'sheepie' would not be an appropriate gloss for the general adult term). The location would be linked in the same way as other datums. Each babytalk item would be linked to its adult equivalent with a relationship marked with e.g. `babytalkOf`. Entering this data would also have the positive side-effect of increasing the general use lexical data present in the database.

Finally, this system can store phonological data as well. One dataset available to the author is a listing of the realization of different proto-consonants in various Syrian dialects. For example, the proto-phoneme /*k/ is variously realized in Syria as [k], [tʃ], [ʃ], sometimes with phonological variation between a base variant (usually but not always [k]) and conditioned variant. Neither this database, nor the source that it is based upon (Behnstedt, 1997, map 15) specify the exact conditioning environment. To encode

Word	Language	Country	Area	Adult speech	English	PoS
daddaš	Arabic	Algeria	Dellys	dəddəš	toddle	Interjection, Verb (im…
nənni	Arabic	Algeria	Dellys	ərqüd, ərgüd	sleep	Verb
baɛɛa, baɛbaɛ	Arabic	Algeria	Dellys	kəbš	sheep, sheepie	Noun

Abbildung 5: A screenshot of a typical entry from the babytalk website (http://babytalk.barefootlinguist.com/)

this data, each modern realization of the proto-phoneme /*k/ (i.e. `k`, tʃ, etc) would be stored in the `normalizedEntry` field. The gloss field would simply read `phoneme` for the sake of this example. Each phoneme would be tagged with `reflexof.*k`. Conditioned variants would also be coded as datums, glossed as `allophones`, and linked to the base phoneme with a `variant.conditioned` relationship. The standard in DAD has been to express the conditioning environment of a datum with a tag, so the allophones would be tagged with `conditioning.unknown`. With this model, it should be straightforward to search for the realizations of this proto-phoneme.

5 Limitations of the model

The primary limitation of this database model is related to its flexibility. It has no inherent controls on the tagging or relational systems, and much of the validation of integrity and consistency must be performed at the level of the software. Prior to data entry, the project designers and stakeholders should design their ontology of tags and relations, and the software can be developed accordingly. The system does easily allow for growth or new requirements, since tags can easily be added and modified.

Unlike the RDF model used by Good and Hendryx-Parker (2006), the relational model here cannot easily treat relationships as objects that can themselves be parts of relationships, i.e. 'reification' (Good and Hendryx-Parker, 2006, pp. 18–20, illustrate this shortcoming of relational databases at length). For the purposes of their project, this could be a fatal flaw, but for projects such as DAD, there is no real need for such complex relational information. Any amount of potentially contradictory information can be stored both in the tags for datums and in relationships and their tags, but as that information gets more complex it would be better to modify the database structure.

Literatur

Behnstedt, P. (1997). *Sprachatlas von Syrien.* Harrassowitz, Weisbaden.

Behnstedt, P. and Woidich, M. (2010). *Wortatlas der arabischen Dialekte: Mensch, Natur, Fauna und Flora*, volume 1. Brill, Leiden.

Diessel, H. (1999). *Demonstratives : form, function, and grammaticalization.* John Benjamins Publishing Company, Amsterdam.

Dimitriadis, A. (2006). An extensible database design for cross-linguistic research. http://languagelink.let.uu.nl/burs/docs/burs-design.pdf.

Farrar, S. (2006). A universal data model for linguistic annotation tools. In *Proceedings of the EMELD'06 Workshop on Digital Language Documentation: Tools and Standards: The State of the Art*, Lansing, Michigan.

Forkel, R. (2014). The cross-linguistic linked data project. In *LREC 2014 (The International Conference on Language Resources and Evaluation)*, pages 60–66.

Good, J. and Hendryx-Parker, C. (2006). Modeling contested categorization in linguistic databases. In *Proceedings of the EMELD '06 Workshop on Digital Language Documentation: Tools and Standards: The State of the Art*, Lansing, Michigan.

Kent, W. (1983). A simple guide to five normal forms in relational database theory. *Communications of the ACM*, 26(2):120–125.

Lewis, W. D., Farrar, S., and Langendoen, D. T. (2006). Linguistics in the internet age: Tools and fair use. In *Proceedings of the EMELD'06 Workshop on Digital Language Documentation: Tools and Standards: The State of the Art*, Lansing, Michigan.

Penton, D., Bow, C., Bird, S., and Hughes, B. (2004). Towards a general model for linguistic paradigms. In *Proceedings of the EMELD'04 workshop on databases and best practice*.

Laurent Romary

TEI and LMF crosswalks[1]

Abstract

The present paper explores various arguments in favour of making the Text Encoding Initiative (TEI) guidelines an appropriate serialisation for ISO standard 24613:2008 (LMF, Lexical Mark-up Framework)[2]. It also identifies the issues that would have to be resolved in order to reach an appropriate implementation of these ideas, in particular in terms of informational coverage. We show how the customisation facilities offered by the TEI guidelines can provide an adequate background, not only to cover missing components within the current Dictionary chapter of the TEI guidelines, but also to allow specific lexical projects to deal with local constraints. We expect this proposal to be a basis for a future ISO project in the context of the on going revision of LMF.

Since this paper adopts the specific viewpoint of the TEI guidelines, no precise description of LMF is made here. For an introduction to LMF, see section 4 of (ROMARY 2013).

1 Towards a more intimate relationship between the TEI and the LMF standards

This chapter is about a simple thesis: the TEI framework could be the optimal serialisation[3] background for the LMF standard, since it provides both an ideal XML specification platform and a representation vocabulary that can be easily tuned (or *customized*) to cover the various LMF packages and components. This thesis does not come out of the blue but arises naturally when one observes the history of both initiatives, and their current impacts in various communities in the humanities and in computational linguistics, but also when one ponders on the relevance of having an LMF-specific serialisation when lexical data are in essence to be interconnected with various other types of linguistic resources.

As a matter of fact, the current XML serialisation of LMF suffers from both generic and specific problems that have prevented it from being widely used by the various communities interested in digital lexical resources. Right from the onset, the lack of consensus on the strategy to define a reliable and stable XML serialisation has forced the ISO working group on LMF to confine it to an informative annex, with the following main shortcomings:

Being carved in stone within the ISO standard, rather than being pointed to as an external and stable online resource, prevents it from being properly maintained, in order to either make corrections on identified weak points or bugs, or to add additional features;

It is only defined as a DTD, a vestigial XML schema language that hardly any XML developer currently uses anymore and which deeply limits its capacity to express constraints on types or to factorise global attributes. For the sake of simplicity (and this can be easily understood when one has to finalise a text for an ISO standard) no parallel definition of a RelaxNG or W3C schema was provided;

It does not reflect the intrinsic extensibility of LMF, as it does not contain any dedicated mechanism for customization, for instance when the developer of a new lexical model would like to discard some packages or add her own extensions;

A more intrinsic weakness of the suggested LMF serialisation is that it hardly takes up any existing vocabulary that could be reused to express either the macro- or micro-structure of a lexical entry. From a purely technical point of view, basic representation objects such as @xml:id or @xml:lang, which are standard practice in XML design, are redefined locally. At a low level, it misses using ISO 24610 for the representation of feature structures and redefines its own <feat> object[4]. As a whole, it suffers from a syndrome similar to that of the unfortunate ISO standard 1951[5]: it creates a specific silo that shows as little reuse of other initiatives as possible.

All in all, as we shall see in this paper, the TEI guidelines offer an appropriate answer to all the preceding issues. With a specification platform that allows the generation of multiple schema languages, a dynamic setting with short revision cycles, a proper integration of third party (ISO and W3C in particular) standards and of course the existence of a lexical representation basis with its *Dictionary* chapter, it provides the most flexible and reliable setting for deploying lexical applications that are meant to be compliant with the underlying LMF model.

Let us be clear: such infelicities as those we have notice above are usually the characteristics of standards that are in many other respects ahead of their time (think of ISO 8879:1986, SGML and its forerunner role for XML) and which require further years of ripening before they reach the best balance between comprehensiveness, simplicity and technical adequacy. The topic of our paper is indeed to contribute to improving LMF by considering bringing it closer to the TEI, an initiative that is well placed to demonstrate the importance of going through many years of fruitful iterations.

2 TEI as a data-modelling environment

Although the Text Encoding Initiative started nearly 3 decades ago in 1987, with its establishment as a consortium some 15 years ago, we will focus here on its current technical characteristics, knowing that the maintenance mechanisms we describe have contributed to its being the powerful infrastructure we know today.

The scope of the TEI mainly covers documents whose content can be seen as textual. This encompasses several possible object types such as manuscripts (BURGHART & REHBEIN 2012), scholarly papers (HOLMES & ROMARY 2010) or spoken data (SCHMIDT 2011). As we shall see lexical data are part of the covered domains but at this stage the most important feature to stress is that the almost 600 elements of the TEI guidelines are all defined in a specification language based on the TEI vocabulary itself. In a way, as was the case for Lisp[6] in the good old days, the TEI is expressed in its own language.

More fundamentally, the specification principles of the TEI infrastructure, reflected in the so-called ODD (One Document Does it all)[7] vocabulary, are based upon the concept of literate programming introduced by (KNUTH 1984), which advocates an integrated process through which technical specifications and prose descriptions are intimately linked with one another, so that one can easily work with one while having direct access to the equivalent object in the other. From the point of view of the TEI, this means that out of the ODD specification one can generate various schema formats (DTD, RelaxNG schemas, W3C schemas) as well as the documentation in any kind of possible format (pdf, docx, ePub, etc.).

Beyond the fact that the TEI is itself specified in ODD, the language is generic enough to be applicable to non-TEI environments. This has indeed been the case for several initiatives in the standardisation domain, the W3C using it for its ITS[8] recommendation, and ISO committee 37 using it for drafting several of its standards[9]. Moreover, ODD is well designed to combine heterogeneous vocabularies, like integrating CALS tables[10] or MathML[11] formulae within a TEI document. This is particularly important for the reuse of components (typically ISO-TEI feature structures) within a newly designed document model.

Without providing too many technical details here, we can describe the main aspects that give ODD its strength and flexibility:

The core declarative object is naturally the XML element, which can be associated with various descriptive properties (name, gloss, definition, examples and remarks) and technical information (content model based on RelaxNG snippets, further constraints (e.g. Schematron[12] rules), attribute declarations);

In complement to elements, the ODD language allows the definition of classes, which are grouping objects for elements having a similar semantics or occurring in the same syntactical context (for example all grammatical features). These are called *model classes*;

Attribute classes are also available to factorise attributes that are used uniformly by several elements (for instance all attributes providing additional temporal constraints to an element);

Elements may also be grouped together as *modules* (for instance: *drama*, *transcription of speech* and indeed *dictionaries*).

As described in (BURNARD & RAHTZ 2004) these various components provide a wealth of customization facilities, with for instance the possibility to add to or remove an element from a content model by changing its belonging to a given class in the TEI infrastructure. This specification and customization platform also paves the way to the description of coherent XML substructures (or *crystals*, ROMARY & WEGSTEIN 2012), that are essential for a component based data modelling and, as we shall see, correspond to the kind of granularity needed to implement LMF packages.

Finally, all these mechanisms are actually maintained and implemented as an open source portfolio of specifications[13] and tools[14] that facilitate their adoption by a wide range of users.

3 TEI as a quasi-LMF-compliant framework

Now that the motivations and general context for our approach have been set, we can focus on the actual representational tools that the TEI offers to deal with LMF compliant lexical structures. There are indeed two main approaches that one can consider here:

1. Considering lexical structures as **feature structures** and using the corresponding ISO-TEI joint vocabulary to this end;
2. Taking the XML vocabulary available from the **TEI chapter for dictionaries**.

3.1 The baseline – feature structures

The idea of representing lexical entries as feature structures has come to light in conjunction with the necessity of providing a structured representation of lexical data in the context of formal linguistic theories (POLLARD & SAG 1994; HADDAR et alii 2012 for an LMF proposal in this respect) but also to account for the deterministic representation and access to

legacy dictionary data (VÉRONIS & IDE, 1992). As a matter of fact, since the early days of the TEI guidelines (LANGENDOEN & SIMONS 1995; LEE et alii 2004), there existed a specific module[15] inspired by these two trends and extensively covering all aspects of typed feature structures, with mechanisms for declaring constraints on them[16]. In 2006, following an agreement between the TEI consortium and ISO, the module became an ISO standard (ISO 24610-1) and is now the reference XML representation for feature structures.

Applying the ISO-TEI feature structure format for representing data in a way compliant to the LMF meta-model can be achieved quite straightforwardly by mapping LMF concepts as follows:

Components are implemented as features whose value is a complex feature structure;
Elementary descriptors (i.e. which correspond to *complex data categories* in the sense of ISO 12620) are implemented as elementary features with a symbolic value (mapped onto a *simple data category*).

Mappings between features and feature values with data categories can be controlled either by eliciting the association within a feature system declaration, or even by describing a feature library to factorise the information expressed within lexical entries. These mechanisms, related to the use of the so-called DCR[17] attributes (WINDHOUWER and WRIGHT 2012), are based upon the technical description provided in (ARISTAR-DRY et alii 2012) and will not be elaborated further here.

To visualize what such an LMF compliant representation could look like, we provide below a verbatim representation of the "clergyman" example from the LMF standard (cf. figure 4) according to the principles stated above[18].

```
<fs type="Lexicon" xmlns="http://www.tei-c.org/ns/1.0">
   <f name="language">en</f>
   <f name="LexicalEntry">
      <fs>
         <f name="partOfSpeech">commonNoun</f>
         <f name="Lemma">
            <fs>
               <f name="writtenForm">clergyman</f>
            </fs>
         </f>
         <f name="WordForm">
            <fs>
               <f name="writtenForm">clergyman</f>
               <f name="grammaticalNumber">singular</f>
            </fs>
         </f>
```

```
         <f name="WordForm">
            <fs>
               <f name="writtenForm">clergymen</f>
               <f name="grammaticalNumber"/>plural</f>
            </fs>
         </f>
      </fs>
   </f>
</fs>
```

Example 1: Inflected forms of clergyman represented as full feature structures

Even if one does not want to go as far as using fully-fledged feature structures but limits oneself to keeping at least the general principles of the LMF serialisation skeleton (elements named according to their equivalent component in the meta model), it is still possible to use the ISO TEI feature syntax for the corresponding descriptors in an LMF representation[19]. One possible advantage, beyond a better convergence across standardisation initiatives is that it allows, as was alluded to before, a simple declaration of the corresponding feature in connection to a data category registry such as ISOcat (WINDHOUWER & WRIGHT 2012). The suggested mixed-approach is illustrated below with the same "clergyman" example:

```
<LexicalResource xmlns:tei="http://www.tei-c.org/ns/1.0">
   <GlobalInformation>
      <tei:f name="languageCoding">ISO 639-3</tei:f>
   </GlobalInformation>
   <Lexicon>
      <tei:f name="language">eng</tei:f>
      <LexicalEntry>
         <tei:f name="partOfSpeech">commonNoun</tei:f>
         <Lemma>
            <tei:f name="writtenForm"/>clergyman</tei:f>
         </Lemma>
         <WordForm>
            <tei:f name="writtenForm">clergyman</tei:f>
            <tei:f name="grammaticalNumber">singular</tei:f>
         </WordForm>
         <WordForm>
           <tei:f name="writtenForm">clergymen</tei:f>
           <tei:f name="grammaticalNumber">plural</tei:f>
```

```
        </WordForm>
      </LexicalEntry>
   </Lexicon>
</LexicalResource>
```

Example 2: The clergyman example represented as a combination of LMF informative DTD and feature structures

All in all, the feature structure module of the TEI offers several possibilities to work within an LMF friendly environment, with the advantage of being based on a strong formalism where data validation is actually built-in. On the weak side, the generic character of feature structures, which comes with some degree of verbosity, makes it more difficult to maintain by human lexicographers but also provides less off-the-shelf validation facilities[20]. When this becomes an issue, it is reasonable to turn to a format that is natively intended to represent lexical structures such as provided by the dictionary module from the TEI.

3.2 The TEI *Dictionaries* chapter

The TEI guidelines actually come with a quite elaborate XML vocabulary for the description of electronic dictionaries[21]. Conceived initially on the basis of an underlying formal model of the hierarchical nature of a lexical entry (IDE & VÉRONIS 1995), and based upon previous theoretical (VÉRONIS & IDE 1992) and descriptive (IDE et alii 1992) works anticipating the idea of a solid structural skeleton further decorated by means of a variety of descriptors, it is not a surprise that the TEI model matches the LMF core package so well[22]. Still, it is important to keep in mind that the original chapter of the TEI guidelines, then named "Print dictionaries", was strongly oriented towards the representation of digitized material rather than on the creation of born digital lexical data. This had basically two consequences: a) it contains many more constructs intended for the representation of human oriented features (typically the etymology of a word (SALMON-ALT 2006; SALMON-ALT et alii-b 2005)) and b) it offers specific "flat" representations intended to cover the early steps of the digitization process, and that are outside the scope of the structured view we consider in this paper.

Whereas we will provide concrete crosswalks examples between the LMF model and the TEI *Dictionaries* chapter in the following section, we focus here on the description of the main elements that form the basis of the TEI descriptive toolbox for dictionaries.

The main structural elements of the TEI *Dictionaries* chapter are presented below and schematised in Figure 1 to illustrate their structural relationships:

<entry> is the basic structuring element of a lexicon (in the LMF sense) and groups together form information, grammatical information (cf. comments in the following section), sense information and related entries;

<form> can be used to describe one or several forms associated with an entry;

<gramGrp> groups together all grammatical features that may be attached to the entry as a whole (by means of its belonging to the *model.entryPart.top* model class) , to a specific form (through the *model.formPart* model class) or even as constraint on one of the senses of a word (again thourgh *model.entryPart.top*);

<sense> brings together all sense related information, i.e. definitions, examples, usage information and additional notes.

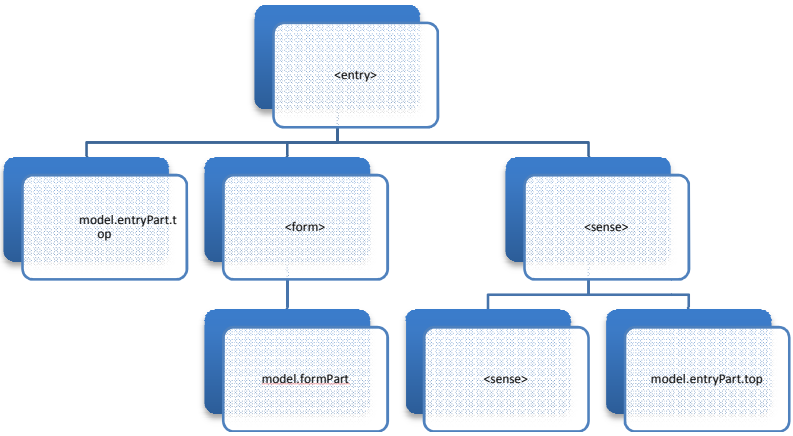


Figure 1: The simplified structure of an entry in the TEI *Dictionaries* chapter

The richness of the TEI descriptive toolbox has at times had the paradoxical effect that one could get deterred from using it simply because it does not come as a ready made module offering a single method of representing a given phenomenon. Although the same criticism could be addressed even more fiercely to the LMF standard itself, it is true that the experience gained over the years with the representation of lexical databases based on the TEI guidelines suggests that it is necessary to introduce more constraints, or at least some precise recommendation to make lexical representations more interoperable (cf. for instance ROMARY & WEGSTEIN 2012; BUDIN et alii 2012).

Among the core issues that sometimes make dictionary designers ponder upon which descriptive object to use is the variety of alternative elements that the TEI offers to <entry> proper. Apart from the possibility to group together homonyms (<hom>) or homographs (<superEntry>), the TEI has two specific elements for representing a lexical entry in a less structured manner: <entryFree> to allow any kind of combination and order of dictionary components, and <dictScrap>, which allows parts of a dictionary entry to be left un-encoded. These alternatives are indeed intended to deal with the specific scenarios of legacy human dictionaries, especially ancient ones, whose entries may not be straightforwardly organised (<entryFree>) or in the case of a multi-step scenario (<dictScrap>) whereby an initially OCRed dictionary is manually encoded step by step.

In the perspective of identifying the optimal customisation of the TEI guidelines that might implement the LMF model, we consider these various alternative constructs as transient objects that are part of specific workflows. For the purpose of disseminating LMF compliant data, we will thus from now onwards only consider <entry> as a proper implementation of the *LexicalEntry* component.

Another typical case of representational ambiguity results from the fact that the core sense-related sub-elements (<cit>, <def> or <usg>, with the ambivalent case of <gramGrp>)

can actually occur freely as children of the <entry> element. This was initially intended to simplify representations where only one sense is being recorded and the encoder wants to avoid the supposedly superfluous <sense> element around such information. But at the end of the day, the resulting representations are not interoperable with one another and, in the context of the arguments made here, some of them are not even compliant with the LMF model. It is thus essential for the TEI community (or the LMF standard in one of its further revisions) to identify which subset of the TEI guidelines can be set as the reference LMF compliant one. As elicited in (ROMARY & WEGSTEIN 2012), such a customization should make <sense> mandatory for the representation of semantic content in <entry>, even if there is indeed only one sense.

Finally, on a more positive note, it can be observed that the TEI brings a lot of potential elements, which, in complement to the basic lexical encoding mechanisms provided by LMF, can be useful for the encoding of deep textual features with text fields. Typically, names, dates, foreign expressions in definitions or examples are not part of the LMF ontology. Still, they are usually important for the proper traversal or cross-linking of lexical material. Whether they are manually or automatically detected, the corresponding TEI vocabulary can definitely be used even as an external resource to LMF compliant representations[23] that are not expressed using the TEI guidelines proper. Typically a location can be tagged within a definition as in the following example:

```
<def>Orchidée épiphyte, originaire d'<geogName>Amérique tropi-
cale</geogName>, et dont l'espèce la plus connue est très recherchée
pour l'élégance de ses fleurs mauves à grand labelle en cornet on-
duleux.</def>
```

Example 3: Inline annotation of TEI content.

Such a wealth of inline annotation mechanisms should not be neglected when one is actually building up lexical resources from heterogeneous sources, which may actually contain such annotations (see for instance ECKLE-KOHLER et alii, 2012).

4 A canonical match: form representation in TEI

As we mentioned earlier, the *TEI Dictionaries* chapter already contains most of the basic constructs needed to implement the various components of the LMF core package. In this section, we would like to focus more specifically on the Form component and identify, a) how the available TEI elements for form description can be matched to the LMF specification and b) what perspective it brings about for the representation of full-form dictionaries, which we will take as an typical example of the type of lexical objects that are needed in the language technology domain (SAGOT 2010).

From an LMF point of view, the description of form information within a lexical entry (see figure 3) consists of a very simple, yet extremely expressive, structure based upon two components:

a **Form** component, which can be iterated within a lexical entry and unites all descriptions associated to what is considered as a single and coherent morphological object associated to the entry;

a **Form Representation** component, which allows one to provide as many descriptive views as needed for a given form.

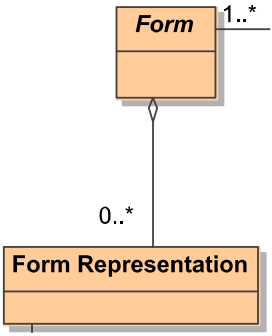


Figure 2: the Form and Form Representation components of the LMF core package

The two-level structure representation is an essential aspect to gain "form autonomy"[24] within a lexical entry. The canonical use of such a construct is typically when a word may occur in several written forms according to the script or transliteration mode being used. For instance, the Hangul representation of the verb "chida" (en: "to hit") can be associated with its Romanized transliteration as sketched below.

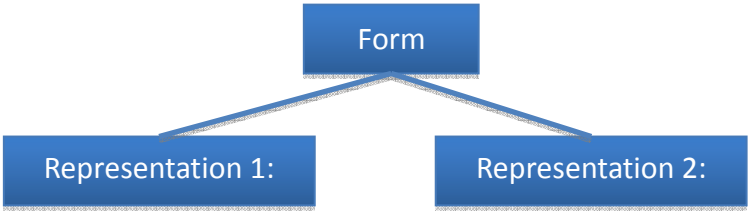


Figure 3: multiple scripting of the Korean verb "chida"

Given the canonical mapping that exists between the Form - Form Representation components in LMF and the <form> element - model.formPart model class in the TEI guidelines, this excerpt can be simply represented in TEI as follows, where the @xml:lang attribute is used to characterize the actual script (here, Hangul vs. Romanized) being used and the @type attribute provides some additional (e.g. project specific) categorisation of the corresponding linguistic segments.

```
<form>
   <orth type="standard" xml:lang="ko-Hang">치다</orth>
   <orth type="transliterated" xml:lang="ko-Latn">chida</orth>
</form>
```

Example 4: Multiple orthographic representations in TEI

If we now move to the slightly more elaborate “clergyman” example depicted in figure 4, the situation is hardly more complex and can be summarized by mean of the mapping table 1.

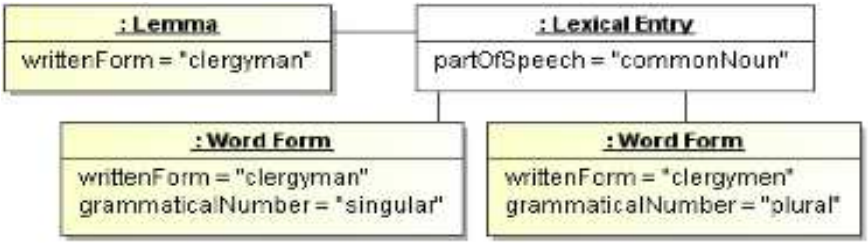


Figure 4: Schematic representation for the entry “Clergyman” (source: LMF standard)

LMF component	*TEI representation*
LexicalEntry	<entry>
Lemma	<form type=“lemma”>
Word Form	<form type=“inflected”>
writtenForm	<orth>
partOfSpeech	<pos>
grammaticalNumber	<number>

Table 1: Mapping between LMF components and corresponding TEI elements

The resulting representation, presented below, corresponds to a strict one-to-one mapping to the corresponding LMF model, which indeed can make it a strong basis for the implementation of any kind of full form lexica[25].

```
<entry>
   <form type="lemma">
      <orth>clergyman</orth>
      <gramGrp>
         <pos>commonNoun</pos>
      </gramGrp>
   </form>
   <form type="inflected">
      <orth>clergyman</orth>
      <gramGrp>
         <number>singular</number>
      </gramGrp>
   </form>
   <form type="inflected">
```

```
      <orth>clergymen</orth>
      <gramGrp>
         <number>plural</number>
      </gramGrp>
   </form>
</entry>
```

Example 5: The clergyman example represented in compliance to the TEI guidelines

As can be seen, the TEI guidelines provide quite a good coverage of the morpho-syntactic features typically needed for full form lexica. Still, there are several issues that have to be considered before one can systematically represent such lexica in an interoperable way for a variety of languages.

From a pure TEI point of view, we already tackled the issue of representational ambiguity, which can make encoders use different constructs to represent the same phenomenon. In the case of inflected forms, both the coherence of their representation and the necessity to remain compliant with LMF requires a systematic use of <form> and <gramGrp> to embed form and grammatical related information respectively, even if in both cases it may be seen as redundant. In the preceding example for instance, even if only a single grammatical feature (<number>) appears in the <gramGrp>, a coherent representation with other word categories (for instance verbs) or other languages, requires that the latter should not be omitted[26]. This allows for instance that a search for the various grammatical constraints used in a lexicon can be made with <gramGrp> as an entry point.

From a data model perspective, this also ensures, as demonstrated in the previous section, a coherent and strict equivalence of <gramGrp> with a feature structure in case one wants to use this generic representation means in place of <gramGrp> within <form>. For instance, the previous example can be reformulated as[27]:

```
<entry>
   <form type="lemma">
     <orth>clergyman</orth>
     <fs type="grammar">
       <f name="pos">commonNoun</f>
     </fs>
   </form>
   <form type="inflected">
     <orth>clergyman</orth>
     <fs type="grammar">
       <f name="number">singular</f>
     </fs>
   </form>
```

```
   <form type="inflected">
     <orth>clergymen</orth>
     <fs type="grammar">
       <f name="number">plural</f>
    </fs>
   </form>
 </entry>
```

Example 6: The clergyman example represented in compliance to the TEI guidelines with feature structures

Finally, we should address here the issue of linguistic coverage, with the possibility of constraining the semantics of the grammatical features used in such representations, and furthermore to add features that may not be part of the core grammatical elements of the TEI, but which are still necessary to describe morpho-syntactic constraints in other languages. For this purpose, the TEI provides a generic <gram> element, which, coupled with the appropriate value for its @type attribute, can theoretically mark any kind of grammatical feature. Still, it is strongly recommended, when one has such a representational need, to design an *ad hoc* element in one's ODD specification and relate this specification to ISOcat by means of either the <equiv> construct or the appropriate DCR attributes[28].

5 Adding components to the TEI framework: the syntactic case

Since the *TEI Dictionaries* chapter was initially conceived to account for the kind of information that appears in machine-readable dictionaries, it only sparsely covers features related to language processing and in particular does not propose any specific element for representing syntactic or semantic structures. When one looks at the various additional packages of LMF on the one hand and at the customisation facilities of the TEI infrastructure on the other, it appears to be relatively easy to define extensions that actually allow TEI based customisation to include the missing LMF constructs.

In this section we present the basic principles to be applied to create such a customization that extends the TEI guidelines by means of an ODD specification for the syntactic package of LMF. This presentation will be carried out by going through a specific example, namely the encoding of verbal structures in CoreNet, the Korean Wordnet.

CoreNet, the Korean Wordnet lexicon (also known as CoreNet, see (CHOI 2003) and (CHOI et alii 2004)) has been put together as a deep semantic and syntactic encoding of a selection of the 50 000 Korean most frequent words (mainly nouns and verbs). Looking at verbs proper, their representation is based upon a double filing system of a) *verb concepts*, associating a concept number (and therefore a Wordnet Synset, via a specific conceptual mapping) to the various senses and b) *verb frames*, associating each sense with one or several predicate-argument structure.

Figure 5: An entry from the verb concept section of CoreNet (senses are marked in red, sub-senses in green)

As illustrated in figure 5 for the verb "chida" (치다), the verb concept structure is organised in senses and sub-senses, to which are attached both a Wordnet reference and a gloss. This two-level semasiological representation is indeed entirely construable as a standard TEI <entry> structure as illustrated below:

```
<entry>
   <form>
      <orth type="한글">치다</orth>
      <orth type="Romanization">chida</orth>
   </form>
   …
   <sense n="3">
      <gramGrp>
         <subc>vt</subc>
      </gramGrp>
```

```
      <sense n="1">
         <ref type="wordnet">
            <idno>1221282691</idno>
            <gloss>치기</gloss>
         </ref>
      </sense>
      <sense n="2">…
      </sense>
   </sense>
</entry>
```

Example 7: Partial TEI representation of an entry from CoreNet (*chida*)

The verb-frame structure is in turn illustrated in figure 6, where one can see that a complementary semasiological structure is being used, grouping together senses from the verb concept structure (represented here by a combination of concept number and gloss) and associating such groups to one or several predicate-argument representations. An additional Japanese gloss is provided for each semantic group, on the basis of the actual semantic restriction introduced for the corresponding arguments.

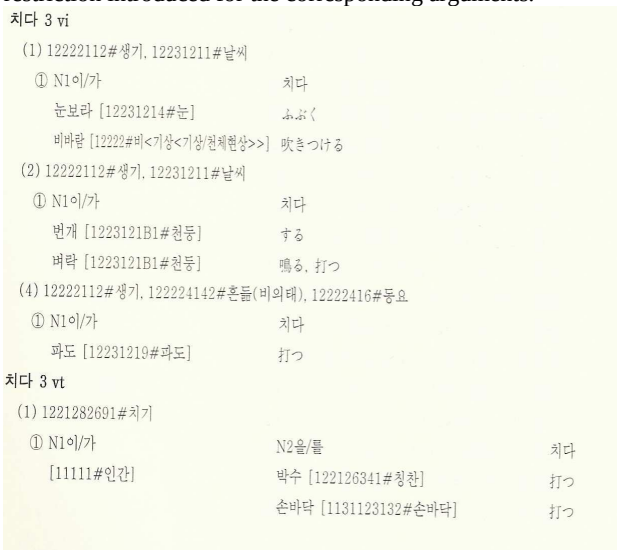


Figure 6: Two entries from the verb frame section of CoreNet

This predicate argument structure is indeed a good instance of the syntactic extension of LMF, which is based on the notion of a sub-categorisation frame (component: Sub-

categorisation Frame), which in turn is linked to various syntactic arguments (component: Syntactic Argument). Figure 7, which takes up an Italian example from the LMF standard, illustrates this core structure and shows how it is directly anchored on the Lexical Entry component.

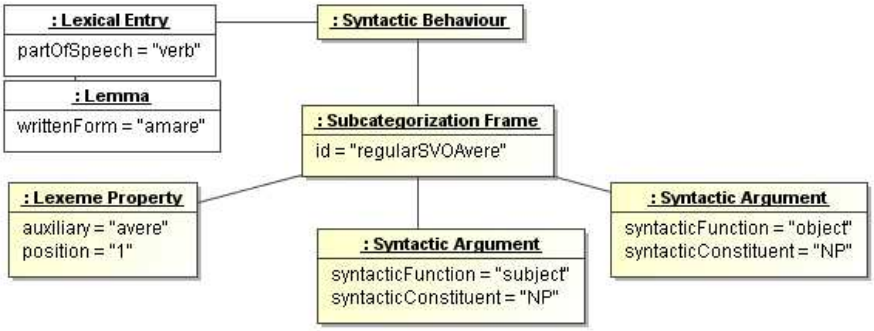


Figure 7: An instance of the LMF syntactic extension (source ISO 24613)

When transposing this model to our CoreNet example, we can actually embed the syntactic description within the sense level of the lexical entry[29]. This leads to a possible TEI extended construct that may look as follows:

```
<tei:sense>
   <tei:gloss xml:lang="ja">ふぶく</tei:gloss>
   <lmf:syntacticBehaviour>
      <lmf:subcategorizationFrame>
         <lmf:syntacticArgument>
            <lmf:syntacticFunction>N1</lmf:syntacticFunction>
            <tei:colloc type="particle" xml:lang="ko">
               이/가</tei:colloc>
            <tei:gloss xml:lang="ko">눈보라</tei:gloss>
            <tei:ref type="wordnet">
               <tei:idno>12231214</tei:idno>
               <tei:gloss xml:lang="ko">눈</tei:gloss>
            </tei:ref>
         </lmf:syntacticArgument>
      </lmf:subcategorizationFrame>
   <lmf:syntacticBehaviour>
</tei:sense>
```

Example 8: Inclusion of a syntactic construct in the TEI representation of an entry from CoreNet (*chida*).

In this representation, we applied the following core specification principles, which, to our view, should be systematically applied for any further TEI based LMF extension:

Limit the introduction of specific elements to those for which there are no equivalent constructs in the TEI infrastructure

Keep new elements within their own namespace. This is a general principle for TEI customization, but it allows here a clear management of the heterogeneous mix-up of elements that we suggest here at all levels of the representation

Avoid introducing new LMF elements within existing TEI constructs apart from the clear anchoring of the LMF syntax crystal within the <sense> element. This principle is essential to facilitate the future integration of our proposal as an official extension to the TEI guidelines, where unintended side effects should be avoided

As a side note, we can see the interesting case of the various usages of the TEI <gloss> element in this representation. Depending on the context, it can be applied in a systematic way to mark any kind of equivalent wording in the various object or working languages of the dictionary.

The actual implementation of such an extension is rather straightforward. Following the general principles outlined in (TBE 2010) for implementing a TEI customisation in ODD, we only give here the essential aspects of the proposed syntax extension to the TEI Dictionaries chapter[30].

The first step is to create a background customisation comprising the core modules of the TEI guidelines together with the Dictionaries module as follows:

```
<schemaSpec ident="LMFSyntax">
   <moduleRef key="core"/>
   <moduleRef key="tei"/>
   <moduleRef key="header"/>
   <moduleRef key="textstructure"/>
   <moduleRef key="dictionaries"/>
</schemaSpec>
```

Example 9: Outline of the ODD specification TEI customisation for dictionaries.

The second step is to create specifications for all new elements within a specific LMF namespace. When such elements have a complex content model, an associated element class is created so that the content model is easy to customise further. For instance, a simplified specification for the <syntacticArgument> element may look as follows:

```
<elementSpec ident="syntacticArgument" module="Syntax"
   ns="http://www.iso.org/ns/LMF">
   <classes>
      <memberOf key="model.subcategorizationFramePart"/>
```

```
   </classes>
   <content>
      <rng:oneOrMore>
         <rng:ref name="model.syntacticArgumentPart"/>
      </rng:oneOrMore>
   </content>
</elementSpec>
```

Example 10: ODD specification for the <syntacticArgument> element.

Finally, as seen also in the preceding example each element is made a member of the appropriate classes to appear in the intended content models.

The resulting specification is all in all quite simple and allows one to edit syntactic lexica right away, while remaining within the TEI realm. Moreover, it shows that implementing similar extensions for some additional packages would definitely be an easy tasks that would not take too much time for a minimally TEI minded person.

6 Contributing to the LMF packages: linguistic quotations

We now address the opposite case to the one we have just seen, namely when some existing constructs in the TEI infrastructure do not have any counterpart in the LMF standard and can thus contribute to defining additional packages. There are indeed several such interesting cases in the TEI guidelines (one may think in particular of all etymological related aspects), but in order to make the point clear we will focus on a simple yet essential type of information: *quotation structures*.

Quotations in a lexical database are linguistic segments that illustrate the use of the headword either as a constructed example, as the citation of an external source or through the embedding of excerpts that have been automatically extracted and selected from a corpus. In some lexicographic projects (cf. e.g. KILGARRIFF & TUGWELL 2001 or SINCLAIR 1987) such quotations have even been the organising principle of the whole lexical matter.

In their simplest form, quotations appear as a textual sequence embedded within other descriptive information of the word, for instance[31]:

> **ain't (eInt)** *Not standard. contraction of* am not, is not, are not, have not or has not: *I ain't seen it*.

When the quotation is actually taken from a known source, it is usually accompanied by an explicit (usually abbreviated) reference to it, as in[32]:

> **valeur** … n. f. … 2. Vx. Vaillance, bravoure (spécial., au combat). 'La valeur n'attend pas le nombre des années' (Corneille).

In the case of multilingual dictionaries, we can extend the notion of quotations to the provision of a translation, possibly accompanied by additional contextualising information. This falls indeed within our earlier definition of a quotation, since such translations actually illustrate the intended meaning in the target language. In the following example we see for instance how such a translation can in turn be refined by an explicit gloss for the corresponding meaning:

> **rémoulade [Remulad]** nf remoulade, rémoulade (*dressing containing mustard and herbs*).

Further types of quotation refinements can be observed in existing dictionaries and indeed, any kind of morpho-syntactic, syntactic or semantic information may be associated with quotations, as long as it provides a qualification for the corresponding usage. Taking again the case of multilingual dictionaries, it is indeed standard practice to refine a translation by means of gender information as in the following excerpt:

> **dresser** … (a) (Theat) habilleur m, -euse f; (Comm: window ~) étalagiste mf. she's a stylish ~ elle s'habille avec chic; V hair. (b) (tool) (for wood) raboteuse f; (for stone) rabotin m.

In this example, we see various types of refinements, with a simple marking of gender for the translation (*habilleur m*), to a combination of morpho-syntactic and semantic constraints (*(for wood) raboteuse f*).

As can be seen, quotation structures are a strong component of the organisation of lexical entries in senses. We are used to observing these in traditional print dictionaries, but indeed, it is easy to foresee a generic mechanism that applies to any lexical database where illustrative text (examples or translations) are to be integrated.

In this respect, the TEI has taken this issue very seriously by introducing in its recent editions (from P5 onwards), a single construct based on the <cit> element[33] that merged the various specific constructs that existed for examples (the <eg> element in the P4 edition of the TEI guidelines) or translations (the <tr> element in P4). This construct can be characterised as follows:

it is based upon a very generic two-level structure where the <cit> element is the entry point and comprises a language excerpt expressed by means of a <quote> (occasionally a <q>) element;

the <cit> element may have a @type attribute to further constrain the nature of the quotation construct, for instance “example” or “translation”.

In the simplest case, when no further constraint or bibliographic reference is needed, the <cit> construct boils down to something as simple as the following example representing a translation[34]:

```
<cit type="translation" xml:lang=”fr”>
   <quote>horrifier</quote>
</cit>
```

Example 11: Simple example for <cit>.

When further refinements are expressed in relation to the quotation, these are added to the actual quoted sequence, using the usual descriptive vocabulary available from the TEI guidelines. For instance, the provision of the gender for the French equivalent to the headword “dresser” in English would be expressed as follows:

```
<cit type="translation" xml:lang="fr">
   <quote>habilleur</quote>
   <gramGrp>
      <gen>m</gen>
```

```
   </gramGrp>
</cit>
```

Example 12: <cit> construct with grammatical constraints.

Finally, an important feature of the <cit> element is its recursivity where for instance the actual translation for a example is also provided, as in the following example:

```
<cit type="example”>
   <quote>she was horrified at the expense.</quote>
   <cit type="translation" xml:lang=”fr”>
      <quote>elle était horrifiée par la dépense.</quote>
   </cit>
</cit>
```

Example 13: <cit> construct with a translation of the main example.

The LMF standard does not have a real equivalent to the <cit> crystal and the only similar structure that appears in LMF may be the possibility to associate a statement in a definition[35]. We thus propose to define an optional extension to the LMF core package, anchored on the sense component and schematized in figure 8.

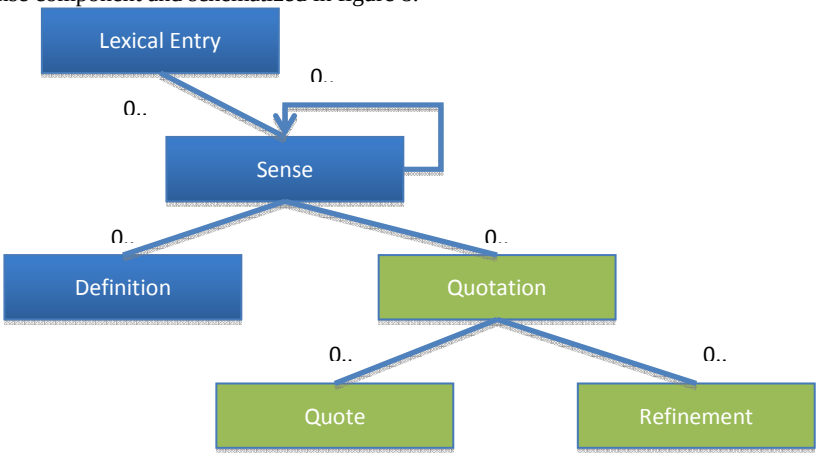


Figure 8: A sketch for a possible Quotation package in LMF

As we can see, the package is directly part of the Sense component aggregation and further defined as a combination of a Quote (an instance of the Text Representation component in LMF) and a Refinement component.

A further specification process, which should be carried out in consultation with the community of lexical databases developers and users, should clarify what should pertain to the Refinement component in this model. As we have seen, we have here a wide spectrum of possibilities, ranging from authorship or bibliographical information to morpho-syntactic constraints and comprising various alternative forms (pronunciation, variants, translations)

or usage information (subject, definition, gloss). Of course, a possible instance of a Requirement may also be a Quotation.

7 Towards more convergence between initiatives: a roadmap

One of the underlying aims of this paper is to demonstrate that there are some good possibilities to work towards a better convergence between the LMF and the TEI initiatives in the domain of lexical structures, and in particular take full benefit of each side's strengths. Indeed, whereas the ISO perspective brings stability and an international validation, it should not be neglected how large the current TEI community is. With this perspective in mind, the project of having an LMF serialisation entirely expressed as a TEI customisation can be seen as a most important endeavour to offer a common and strong basis for any kind of lexical work both in the language technology and the digital humanities domains. This will also provide LMF with a real customisation platform that will facilitate the work of defining project specific subset within a coherent framework that guaranties compliance to the underlying reference standard.

There is indeed a good window of opportunity to go in this direction. ISO committee TC 37/SC 4 has issued a plan in 2015 to revise ISO standard 24613 so that it becomes a multipart standard reflecting the variety of domains addressed so far within one single document. In this context, it would probably be appropriate to submit a specific part dedicated to the serialisation of LMF by means of the TEI guidelines on the basis of the principles expressed in this paper. Even if we cannot anticipate, at the time of publication of this paper, the possible success of such an endeavour, the various positive signs received already by the author of this paper are encouraging to carry this out as far as possible.

Acknowledgements

I want to address here my deep and friendly thanks to the colleagues and friends who provided such a valuable feedback on early draft of the paper, in particular JUDITH ECKLE-KOHLER, MARTIN HOLMES, JOHN MCCRAE, CHARLY MÖRTH, DANIEL STOEKL, TOMA TASOVAC, WERNER WEGSTEIN, MENZO WINDHOUWER as well as the two anonymous reviewers of JLCL.

Abbreviations

DCR	Data Category Registry
FSD	Feature System Declarations (ISO 24610-2:2011)
ISO	International Organisation for Standards
LMF	Lexical Markup Framework (ISO standard 24613:2008)
ODD	One Document Does it all, the specification subset of the TEI guidelines
RelaxNG	Regular Language for XML Next Generation
SGML	Standard Generalized Markup Language (ISO 8879:1986)
TEI	Text Encoding Initiative
W3C	World Wide Web Consortium
XML	Extensible Markup Language (W3C recommendation)

References

ARISTAR-DRY, H., DRUDE, S., WINDHOUWER, M., GIPPERT, J., and NEVSKAYA, I. (2012). "Rendering Endangered Lexicons Interoperable through Standards Harmonization: The RELISH Project." In: Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC 2012), Istanbul, May 23rd-25th, 2012 (pp. 766-770).

BUDIN G., S. MAJEWSKI and K. MÖRTH (2012). "Creating Lexical Resources in TEI P5", Journal of the Text Encoding Initiative, Issue 3.

BURGHART M. and M. REHBEIN (2012). "The Present and Future of the TEI Community for Manuscript Encoding", Journal of the Text Encoding Initiative, Issue 2, February 2012. URL : http://jtei.revues.org/372 ; DOI : 10.4000/jtei.372

BURNARD L. and S. RAHTZ (2004). "RelaxNG with Son of ODD". In: Proceedings of Extreme Markup Languages conference.

CHOI K.-S. (2003). "CoreNet: Chinese-Japanese-Korean wordnet with shared semantic hierarchy", In: Proc. IEEE International Conference on Natural Language Processing and Knowledge Engineering - NLP-KE (pp. 767-770).

CHOI, K.-S. , H.-S. BAE, W. KANG, J. LEE, E. KIM, H. KIM, D. KIM, Y. SONG and H. Shin (2004). "Korean-Chinese-Japanese Multilingual Wordnet with Shared Semantic Hierarchy." In: Proceedings of LREC 2004, Lisbon, Portugal, 26 May - 28 May 2004

ECKLE-KOHLER J., I. GUREVYCH, S. HARTMANN, M. MATUSCHEK and C. M. MEYER (2012). "UBY-LMF – A Uniform Model for Standardizing Heterogeneous Lexical-Semantic Resources in ISO-LMF." In: Proceedings LREC 2012, May 2012. Istanbul, Turkey (pp. 275–282).

HADDAR K., H. FEHRI and L. ROMARY (2012). "A prototype for projecting HPSG syntactic lexica towards LMF", Journal of Language Technology and Computational Linguistics, 27(1), (pp. 21-46) — *http://hal.inria.fr/hal-00719954*.

IDE N. and J. VÉRONIS, (1995). Encoding dictionaries. In Ide, N., Veronis, J. (Eds.) *The Text Encoding Initiative: Background and Context*. Dordrecht: Kluwer Academic Publishers, 167-80.

IDE N., J. VERONIS, S. WARWICK-ARMSTRONG and N. CALZOLARI (1992). "Principles for encoding machine readable dictionaries." In: Proceedings of EURALEX'92, H. Tommola, K. Varantola, T. Salmi-Tolonen, Y. Schopp, eds., in *Studia Translatologica*, Ser. a, 2, Tampere, Finland, (pp. 239-246) — *http://www.cs.vassar.edu/~ide/papers/Euralex92.pdf*

ISO 1951:2007 Presentation/representation of entries in dictionaries -- Requirements, recommendations and information

ISO 24610-1:2006 Language resource management -- Feature structures -- Part 1: Feature structure representation

ISO 24613:2008 Language resource management - Lexical markup framework (LMF)

KILGARRIFF, A. and D. TUGWELL (2001). "WORD SKETCH: Extraction and display of significant collocations for lexicography." In: Proceedings Collocations workshop, ACL 2001.

KNUTH, D. (1984) "Literate Programming." In: Literate Programming. CSLI, 1992, pg. 99.

HOLMES, M. and L. ROMARY (2010). "Encoding models for scholarly literature", in *Publishing and digital libraries: Legal and organizational issues*, Ioannis Iglezakis, Tatiana-Eleni Synodinou, Sarantos Kapidakis (Ed.) (pp. 88-110) - *http://hal.archives-ouvertes.fr/hal-00390966*.

LANGENDOEN, D. TERENCE and G. F. SIMONS, (1995). "A rationale for the TEI recommendations for feature-structure markup." Computers and the Humanities 29 (pp. 191-209).

LEE K., L. BURNARD, L. ROMARY, E. DE LA CLERGERIE, T. DECLERCK, S. BAUMAN, H. BUNT, L. CLEMENT, T. ERJAVEC, A. ROUSSANALY, C. ROUX (2004). “Towards an international standard on feature structures representation.” In: Proceedings of LREC 2004 (pp. 373-376) — *http://hal.inria.fr/inria-00099855*.

LEMNITZER, L., L. ROMARY, and A. WITT (2013). “Representing Human and Machine Dictionaries in Markup Languages.” In : HSK - Dictionaries. An International Encyclopedia of Lexicography. Supplementary Volume: Recent Developments with Special Focus on Computational Lexicography, December 2013 — *https://hal.inria.fr/inria-00441215*.

POLLARD C. and I. A. SAG (1994): Head-Driven Phrase Structure Grammar. Chicago: University of Chicago Press.

ROMARY L. (2013). “Standardization of the Formal Representation of Lexical Information for NLP.” In: Dictionaries. An International Encyclopedia of Lexicography. Supplementary Volume: Recent Developments with Special Focus on Computational Lexicography, December 2013 — *https://hal.archives-ouvertes.fr/hal-00436328*.

ROMARY L., SALMON-ALT S., FRANCOPOULO G. (2004). “Standards going concrete: from LMF to Morphalou”, In: Proceedings of Workshop on Electronic Dictionaries, Coling 2004, Geneva, Switzerland — *http://hal.inria.fr/inria-00121489*

ROMARY L. and W. WEGSTEIN (2012). “Consistent modelling of heterogeneous lexical structures”, *Journal of the Text Encoding Initiative*, Issue 3 | November 2012 URL : http://jtei.revues.org/540 ; DOI : 10.4000/jtei.540 — *http://hal.inria.fr/hal-00704511*

SAGOT B. (2010). “The Lefff, a freely available and large-coverage morphological and syntactic lexicon for French.” In: Proceedings of LREC 2010, Valletta : Malte — *http://hal.archives-ouvertes.fr/inria-00521242/*

SALMON-ALT S., A. AKROUT and L. ROMARY (2005). “Proposals for a normalized representation of Standard Arabic full form lexica.” In: Proceedings of Second International Conference on Machine Intelligence (ACIDCA-ICMI 2005), Tozeur, Tunisia — *http://halshs.archives-ouvertes.fr/halshs-00004541*

SALMON-ALT S. (2006) “Data structures for etymology: towards an etymological lexical network.” BULAG 31 1-12 — *http://hal.archives-ouvertes.fr/hal-00110971*

SALMON-ALT S., L. ROMARY, E. BUCHI (2005). “Modeling Diachrony in Dictionaries”. ACH-ALLC 2005, Vancouver, Canada.

SCHMIDT T. (2011). “A TEI-based Approach to Standardising Spoken Language Transcription”, *Journal of the Text Encoding Initiative*, Issue 1 | June 2011, URL : http://jtei.revues.org/142 ; DOI : 10.4000/jtei.142

SINCLAIR J. M. (ed.) (1987). Looking Up: An Account of the COBUILD Project in Lexical Computing. Collins, London.

TBE (2010) “Customising TEI, ODD, Roma”, in *TEI by Example*, *http://tbe.kantl.be/TBE/modules/TBED08v00.htm*

VÉRONIS, J. and N. IDE (1992). A feature-based model for lexical databases. *Proceedings of the 14th International Conference on Computational Linguistics, COLING'92,*Nantes, France, 588-594.

WINDHOUWER, M., and WRIGHT, S. E. (2012). “Linking to linguistic data categories in ISOcat”. In C. Chiarcos, S. Nordhoff, & S. Hellmann (Eds.), *Linked data in linguistics: Representing and connecting language data and language metadata* (pp. 99-107). Berlin: Springer.

[1] Most abbreviations are elicited when they appear for the first time in the text. A complete abbreviation section is available at the end of this paper, right before the bibliographical references.
[2] We will henceforth refer to the ISO document as simply *LMF*.
[3] "Serialisation" means a concrete data representation on computers for the sake of storage or interchange. A serialisation, for instance an XML format, is often conceived in compliance with a reference model (in the case of our paper, LMF).
[4] The LMF <feat> object is not even compliant with ISO standard 16642 (TMF) which defined such an element before ISO 24610 was in place.
[5] See (LEMNITZER et alii 2013) for a more precise analysis of the difficulties related to ISO 1951.
[6] See *https://en.wikipedia.org/wiki/Lisp_(programming_language)*
[7] See a technical introduction in *http://www.tei-c.org/Guidelines/Customization/odds.xml*
[8] Internationalization Tag Set; *http://www.w3.org/TR/its/*
[9] ISO 24611, ISO 24616, ISO 24617-1, and on going revision of ISO 16642
[10] Maintained by OASIS, see *https://www.oasis-open.org/specs/tablemodels.php*
[11] Maintained by W3C, see *http://www.w3.org/Math/*
[12] *http://www.schematron.com*
[13] *https://github.com/TEIC/TEI*
[14] For instance, Roma (*http://www.tei-c.org/Roma/startroma.php*) for the online design of customization, or Oxgarage (*http://www.tei-c.org/oxgarage/*) for the transformation of TEI documents from and to various possible formats or schema languages.
[15] Chapter 18 in TEI P5 - *http://www.tei-c.org/release/doc/tei-p5-doc/en/html/FS.html*
[16] FSD – Feature System Declarations
[17] Data Category Registry
[18] In all our examples, we will use the simplified (untyped) form for feature values as plain text content of the <f> element. More elaborate implementations should distinguish specific subtypes as specified in the ISO-TEI specification.
[19] A very similar approach has indeed been developed by MENZO WINDHOUWER in the context of the RELISH project, see *http://tla.mpi.nl/relish/lmf/* and (ARISTAR-DRY et alii 2012)
[20] Note that the same criticism applies to RDF based representations, which should only be contemplated for some specific end-user delivery scenarios.
[21] see *http://www.tei-c.org/release/doc/tei-p5-doc/en/html/DI.html*
[22] It is even less surprising given that the TEI principles informed the first ISO meeting in Korea (February 2004) where the first LMF consensus was put together (ROMARY et alii 2004)
[23] see for instance the chapter "Names, Dates, People, and Places" (*http://www.tei-c.org/release/doc/tei-p5-doc/en/html/ND.html*) for the encoding of basic name entities.
[24] Like we have the term autonomy principle in terminology

[25] See also the first experiments done on the Morphalou dictionary (ROMARY et alii, 2004) or for the Arabic language (SALMON-ALT et alii, 2005-a)
[26] In the case where there is no grammatical information available, the <gramGrp> element should be of course omitted. Indeed, it is important to keep to the general encoding rule of avoiding the insertion of useless void elements (With thanks to MARTIN HOLMES for pointing this out to me).
[27] CHARLY MÖRTH rightly mentions that when implementing such a solution on a large scale it may be appropriate to move all <fs> elements into <fLib> elements and use an @ana attribute on form to refer to them.
[28] Namely: dcr:datcat and dcr:valueDatcat
[29] The full LMF package for syntax is (rightly) intended to allow the factorisation of syntactic constructs across several entries. We simplify the representation here to make our point clearer. The full ODD specification should indeed implement both views.
[30] The complete customisation is available under *http://hal.inria.fr/hal-00762664*
[31] Source: TEI P5, chapter "Dictionaries", *http://www.tei-c.org/release/doc/tei-p5-doc/en/html/DI.html* (original source: *Collins English Dictionary*. London: Collins)
[32] *ibid.* (original source: GUERARD, FRANÇOISE (1990). *Le Dictionnaire de Notre Temps*, Hachette, Paris)
[33] *http://www.tei-c.org/release/doc/tei-p5-doc/en/html/ref-cit.html*
[34] We recommend this construct rather than the simpler: <gloss xml:lang="en">horrifier</gloss>, with the assumption that it is better to use the same structure (<cit>) for both glosses and illustrative quotations. Thanks to Martin Holmes for pointing to this.
[35] cf. ISO 24613 "*Statement* is a class representing a narrative description and refines or complements *Definition*."

Uladzimir Sidarenka, Andreas Peldszus, Manfred Stede

Discourse Segmentation of German Texts

This paper addresses the problem of segmenting German texts into minimal discourse units, as they are needed, for example, in RST-based discourse parsing. We discuss relevant variants of the problem, introduce the design of our annotation guidelines, and provide the results of an extensive inter-annotator agreement study of the corpus. Afterwards, we report on our experiments with three automatic classifiers that rely on the output of state-of-the-art parsers and use different amounts and kinds of syntactic knowledge: constituent parsing versus dependency parsing; tree-structure classification versus sequence labeling. Finally, we compare our approaches with the recent discourse segmentation methods proposed for English.

1 Introduction

'Discourse parsing' nowadays typically refers to the task of assigning a structure to a monologue text, where this structure is driven by an underlying theory of coherence relations and their composition. One popular such theory is Rhetorical Structure Theory (RST, Mann and Thompson (1988)), which holds that a tree can be obtained by recursively relating adjacent 'elementary discourse units' (EDUs). Early work on deriving RST trees automatically was done by Marcu (2000), and following the advent of a large data set, the RST Discourse Treebank (Carlson and Marcu, 2001), a variety of machine learning approaches have been proposed to tackle the problem, for example those of Hernault et al. (2010b) and Ji and Eisenstein (2014).

An alternative theory is Segmented Discourse Representation Theory (SDRT, Asher and Lascarides (2003)) which does not enforce the tree-structure constraint and is also applicable to dialogue. Corpora are available in English and French, and one of the SDRT parsers that have been presented recently is the one by Muller et al. (2012).

For RST and SDRT (and similar approaches), any discourse parser relies on a segmentation of the text into EDUs, which in all present frameworks amounts to a sequence of non-overlapping units that completely covers the text. While the original RST paper by Mann and Thompson remained relatively vague on the issue of defining EDUs (the essential characterization was "typically clauses"), work on discourse parsing relies on an operationalization and thus on a specific definition. We will discuss the issues involved in this step below in Section 2.

RST and SDRT are in contrast with the Penn Discourse Treebank approach to discourse structure (Prasad et al., 2008), which does not assign any complete structure to the text, but marks up individual coherence relations (which may or may not be explicitly signaled by a connective or other lexical means) and their argument spans.

Each relation is annotated individually, without considering any surrounding structure. And since annotators do not receive specific instructions on what may constitute an argument of a connective, there is no need for a definition of EDU in this particular approach. However, it is empirically interesting to compare the arguments that were selected intuitively by the annotators to a formal notion of EDU and determine where they match and where they do not.

Discourse parsing is not the only application that needs EDU segmentation. Work on speech act assignment, which originated in the dialogue community but has spread to the annotation of social media contributions and sometimes also to "standard" monologue texts (e.g., Stede and Peldszus (2012)) also relies on the notion of a minimal unit of text that can be ascribed a speech act. One extension of this is the increasingly popular area of argumentation mining, where argumentative moves and relations among them are being identified. Another research field which also might significantly benefit from a high-quality discourse segmentation is that of anaphora resolution. For any implementation of Centering Theory (Grosz et al., 1995), a discourse segmentation is the prerequisite for computing centering transitions, which in turn influence the assignment of pronoun antecedents. In an interesting study, Taboada and Zabala (2008) demonstrated the effects of different EDU definitions on pronoun resolution performance. Likewise, approaches in the "constraints and preferences" tradition (Lappin and Leass, 1994), which compute salience rankings over sequences of minimal discourse units, rely on a definition of EDU.

While a lot of work has been done along the lines described above for English, very few studies have been presented on German, and we will mention them later in the paper. With this paper, we hope to close this gap by detailing our guidelines for the manual annotation of discourse segments in a German corpus (Potsdam Commentary Corpus, Stede and Neumann (2014)), conducting an extensive inter-annotator agreement study of the resulting annotation, and offering the first generally-available discourse segmentation module that was specifically designed and implemented for German.

The rest of this work is structured as follows. Section 2 introduces the problem of discourse segmentation in more depth and discusses different options for defining the notion of EDU. Then, in Section 3, we describe our annotation guidelines and present the results of an annotator agreement study as well as a brief description of the resulting annotated corpus. After summarizing related work on discourse segmentation for English and German in Section 4, we turn our attention to automatic segmentation methods and describe three different approaches involving classifiers that operate on the output of state-of-the-art German syntax parsers. In particular, our interest is in comparing the performances of a dependency parser and a constituency-based parser for our particular task; this is then supplemented by the analysis of a sequence labeling approach. The performance of the three methods is evaluated on our manually-annotated corpus. Finally, Section 5 discusses our results and draws conclusions.

2 Discourse Segmentation: The Problem

As Mosegaard-Hanse (1998) observed, the task of segmentation can in principle be performed on the basis of

- *form*, by providing structural criteria for boundary identification; or
- *meaning*, by identifying stretches of text that express complete propositions, and assigning segment boundaries accordingly; or
- *action*, by identifying stretches of text that represent complete speech acts, and assigning their boundaries.

From a computational perspective, the first option is the "ideal" one: If it were possible to exploit formal signals only, any task involving language unit *interpretation* could be clearly split off from a preparatory step of purely form-based unit *identification*. This would certainly work if human language users communicated solely in terms of sequences of main clauses. But obviously matters are more complicated, mainly because

- complex sentences for many purposes need to be split into individual clauses, and
- fragmentary material of various kinds needs to be accounted for.

In some approaches, the problem is "solved" by generally taking the complete sentence, including all minor clauses and adverbials, as unit of analysis. In local-coherence analyses based on Centering, this was done regularly (Tetreault, 2001; Miltsakaki, 2002). But this was hardly a decision on the grounds of theoretical adequacy or empirical evidence, but on the grounds of practical convenience. Taboada and Zabala (2008) discussed this in detail and proposed a more fine-grained segmentation for the application of Centering.

Similarly, when the goal is to annotate coherence relations, as with RST, defining smaller units is inevitable, since such relations often hold intra-sententially. Again, the importance of high-quality segmentation can be demonstrated by its effect on the task, here on RST parsing: Soricut and Marcu (2003) in their study determined that the availability of perfect (gold standard) segmentation would reduce the number of discourse parser errors by 29%. Therefore, finding good solutions to the EDU demarcation task is of great importance.

Nonetheless, in empirical analyses, it proved rather difficult to make boundary decisions solely based on formal criteria. For example, Marcu (2000) in his pioneering work gave a few structural criteria for RST segmentation, but then added that further boundaries are to be introduced "when a coherence relation is present", thus effectively combining 'form' and 'meaning' criteria. As will become clear in the next section, in our project we also in general follow this approach, but the meaning-aspect of boundary assignment is accounted for by a more general "interpretative" criterion.

Once the basic decision is made to break sentences into smaller units, the target level of granularity needs to be determined. One extreme position is advocated by Schmitt

(2000), who, for the purposes of illocution analysis, accepts some individual adverbs as complete units, since they can express a separate illocution (most often an evaluation) supplementing that of the clause. The vast majority of approaches, however, adopts a much less radical position and takes the presence of a verb as a central condition. We also follow this line in our own approach and take the clause as a basis for the analysis, with exceptions made for fragmentary material that occurs with sentence-final punctuation (see the next section).

Finally, when defining a segmentation task, it needs to be decided whether the output should be a 'flat' sequence of units (of equal status), or whether it should mirror syntactic structure to some extent. Again, this of course depends on the purpose: A *topic-based* segmentation of a text, e.g. in the 'text tiling' tradition (Hearst, 1997), is flat in the vast majority of approaches. Also, the EDUs in RST parsing, in the end, constitute a flat partitioning, but here, the segmentation step can benefit from hierarchical information when determining embedded EDUs. Paying attention to embedding is also of great importance in illocution-oriented analyses, where independent speech acts need to be identified. Embedding can be represented to different degrees and in different ways, viz., by explicitly providing the bracketing structure or by adding syntactic type labels to EDUs that otherwise would form a flat sequence.

For our corpus, we want to be open to various tasks and chose to annotate clause hierarchy (i.e., to provide a very coarse-grained syntactic analysis) and to label the units with their structural types. The different annotation tasks that build upon the segmentation can then either peruse or ignore the structural embedding and the labels.

3 Human Annotation Study

The data for our study comes from the Potsdam Commentary Corpus (PCC, Stede and Neumann (2014)), a freely-available collection of 176 newspaper editorials from a German regional newspaper. The PCC is being distributed with various layers of manually-produced annotations: sentence syntax, nominal coreference, connectives and their arguments, and rhetorical structure. The work reported in the present paper adds a new layer of discourse segments to the corpus. To this end, we devised annotation guidelines and conducted an annotator agreement study, which we present in this section.

3.1 Annotation guidelines

The idea behind our guidelines (Stede et al., 2015) is to provide a base segmentation that can be utilized by other layers of text annotation, such as analyses of illocutionary force, rhetorical structure, coreference, or argumentation. These different purposes can make use of a segmentation in slightly different ways; therefore we aim at a layer of EDUs that is relatively fine-grained, provides type information for the units, and can be thus systematically mapped to reduced, more coarse-grained versions.

A central design decision was that our guidelines for manual annotation take both structural and subjective-interpretation features into consideration. Annotators are asked to identify complete 'sense units', chunks of information that convey a sense of completeness. This clearly subjective criterion is intertwined with the guideline to have most decisions revolve around sentence-final punctuation symbols (full stop, colon, exclamation mark, question mark; henceforth: SFPS). And besides finding boundaries, the annotators are asked to assign a syntactic type label to each unit.

Specifically, our annotators proceed in three phases, with each one being applied to the complete text:

1. For each SFPS, check whether it finishes off a complete sense unit; i.e., make sure that the current sense unit does not stretch beyond this SFPS. If the sense unit does stop at the SFPS, mark it as a sense unit boundary.

2. For each sense unit, check whether it contains more than one structural unit, i.e., one that ends with a SFPS. This can be a full sentence or a fragment; assign appropriate syntactic type labels to each such unit.

3. For each full sentence, check whether it is structurally complex, i.e., it contains several clauses or parenthetical material. If it does, provide markup for the structure.

The result of the procedure is a tree-like structure spanning the complete text: a sequence of contiguous sense units, each of which may consist of recursively embedded structural units. In the following, we provide some details about the three phases of annotation and illustrate them with examples.

3.1.1 Step 1: Identify sense units

The idea of this step is to break the text into interpretable units. We constrain the possible positions of unit boundaries: They can occur only at the punctuation symbols. This largely leads to standard sentence segmentation, but it also takes care of possible fragmentary material that does not correspond to a full sense unit but is to be amalgamated with the preceding or subsequent sentence. Our criterion of "complete sense unit" is to be tested after removing any connectives and after (mentally) replacing anaphoric material with their antecedents.

Here are a few examples of material that contains sentence-final punctuation but is to be fused with a neighbor unit in order to form a complete sense unit:

(1) Most important is: Always keep your eyes open.

(2) The boy bought himself an ice cream. And another one.

(3) There was only one thing that could save me. A good book.

3.1.2 Step 2: Subdivide sense units

As the examples given above illustrate, fragments can be attached either to their left neighbor (because they provide an extension) or to their right neighbor (because they introduce it). One purpose of step 2 is to make this decision explicit by assigning them different types (initiating fragment, FRE; versus finalizing fragment, FRB).[1]

The other purpose is to give types to the non-fragments, i.e., the "ordinary" material. For the most part, this will be main clauses, which are marked as 'HS' *(Hauptsatz)*. However, we distinguish the full main clauses from reduced ones, where the 'fragment' is not to be adjoined with one of the neighbors. These incomplete main clauses (HSF, *Hauptsatzfragment*) are assigned when the clause is elliptical (but can be easily filled from the preceding context) or when it constitutes a complete illocution.

(4) [*Most important is:*]$_{HSF}$ [*Always keep your eyes open.*]$_{HS}$

(5) [*I bought a new laptop.*]$_{HS}$ [*And a camera.*]$_{HSF}$

Thus at the end of step 2, the text is broken into a sequence of labeled units: (two types of) fragments, main clauses, and incomplete main clauses.

3.1.3 Step 3: Subdivide complex sentences

This step is in charge of recursive embedding, where three cases are to be distinguished.

Parataxis: main clauses that appear in the same sentence, possibly linked by a coordinating conjunction. Each receives the label *HS*.

Hypotaxis: Largely following the inventory presented by Bußmann (2002), we distinguish nine different kinds of minor clause, among them subject clause, object clause, adverbial clause, predicative clause, relative clause. Annotators need to determine the clause type and assign the appropriate label. Minor clauses can also be conjoined, in which case we mark them individually.

Parentheticals are units that "interrupt" the clause and are marked by commas, hyphens, or parentheses. However, we mark only those that correspond to a complete proposition or a clearly identifiable illocution.

The following examples[2] illustrate our usage of categories for different types of embedded units.

(6) [*Gestern hat der Lehrer* [*– ganz schön lächerlich! –*]$_{HSF}$ *mit blauen Briefen für verspätete Schüler gedroht.*]$_{HS}$
('Yesterday the teacher – quite ridiculously! – threatened late-coming pupils with sending letters to their parents.')

(7) [*Heute war,* [*so soll es ja auch sein,*]$_{HS}$ *das Kind pünktlich in der Schule.*]$_{HS}$
('Today, as it should be, the child arrived at school on time.')

[1] The abbreviations of all our types are compiled in Table 2 below.

[2] Since the categories do not straightforwardly map to English, we give German examples here.

(8) [*Heute war das Kind* [*(das öfters mal Probleme mit dem Aufstehen hat)*]$_{ANR}$ *pünktlich in der Schule.*]$_{HS}$
('Today the child (who often has problems with getting out of bed) arrived at school on time.')

3.2 Agreement study

From our corpus, we selected 10 texts, each of which is approximately 180 words or 1100 characters long. Two annotators (one Ph.D. student and one post-doc, both with linguistic background) annotated these texts separately, after studying the 15-page guideline document.

3.2.1 Methodological issues

Choosing the right metric for evaluating the results of our agreement study is not trivial. Three desiderata are important for an ideal metric: The metric should be chance-corrected to compensate for expected chance agreement; furthermore, it should be appropriate for the annotation task, i.e., be able to account for all aspects of the annotated structure; finally, it should be well understood, so that there is a consensus on how to interpret the results and what constitutes "good" agreement.

Flat segmentation metrics: There exist different metrics to evaluate flat text segmentation (as for example for the task of topic segmentation, argumentative zoning, or discourse segmentation). Prominent ones are P_k (Beeferman et al., 1999) and WinDiff (Pevzner and Hearst, 2002). Both metrics measure the boundary agreement by moving a window of fixed size over two segmentations of the same sequence and checking whether both agree that the window's edges are in the same segment or not. WinDiff was proposed to overcome insensitivities for certain error types that P_k exhibited. However, both metrics are not chance-corrected. Therefore, to access the reliability of segmentation annotation more accurately, Krippendorff (1995) presented α_U, an agreement measure for unitizing in the family of alpha coefficients. An alternative, a Fleiss multi-π agreement coefficient based on boundary edit distance π^*_{BED} has been presented by Fournier (2013).

Even though our annotations are hierarchical, we will present results in these metrics in order to allow for comparisons with past segmentation studies. For this purpose, we flattened the annotated tree structures to a fine-grained partitioning, with a boundary inserted at every constituent border: *(a (b (c d)))* $\mapsto$ *\| a \| b \| c d \|* . It is to be noted, however, that flat segmentation will only be able to represent embeddings above a depth of 1 when the units are at the left or right border of the superordinate segment, because discontinuous segments (as they would result from center embedding) cannot be captured.

Category agreement metric: All of the above metrics are untyped, i.e., they capture the distinction between boundary or no boundary, but do not consider segments of different categories. If the extensions of spans of two annotators match, agreement metrics for categorical data such as Cohen's κ (Cohen, 1960) can be used to assess

the agreement of category assignment. With this metric, the category assignment for embedded structures can be evaluated without the need of flattening. However, whenever segmentations are different, this simple form of categorial agreement cannot systematically be applied. A generalization of α_U for segments with different categories has been given by Krippendorff (2004). It is not only able to assess the agreement in segmentation but also in segment categorization. In this paper, we will use the symbol α^c_U for this metric.[3]

For using the α^c_U metric on our segmentation trees, we again have to flatten the trees, this time with category labels: *(X a (Y b (Z c) d) e)* $\mapsto$ $|_X\ a\ |_Y\ b\ |_Z\ c\ |_Y\ d\ |_X\ e\ |$. As before, the mapping splits nodes with center embedding into an opening, an embedded, and a closing segment. The opening and the closing segment will be of the same type.

Tree metrics: More appropriate for the comparison of our annotations are tree metrics. In parser evaluation, phrase-structure trees are typically compared with the parseval metric (Black et al., 1991), i.e., labeled and unlabeled precision, recall, and $F1$. The unlabeled scores will reflect the structural agreement of segmentation, the labeled ones will furthermore reflect the category assignment. However, parseval is also not directly suited for accessing inter-annotator agreement, first because it assumes one representation to be the correct one, and second because it is not chance-corrected.

3.2.2 Results

The results of the agreement study are presented in Table 1. The first four rows represent scores for flat segmentation: the uncorrected scores metrics P_k and WinDiff, and the corrected metrics α_U and π^*_{BED}. Note that P_k and WinDiff are error measures, where small numbers are desired. The next rows present α^c_U for typed, flat segmentations and unlabeled and labeled $F1$ scores. The last two rows of the table represent the ratio of exact boundary matches and the κ agreement on the categories of those matches. Except for these two rows, all other metrics are reported as average percentages with standard deviation over the evaluated texts. We also performed similar calculations on the whole corpus, but this yielded very similar scores.

The error in flat segmentation measured with P_k and WinDiff is remarkably low. According to these metrics, seven of ten texts have a perfect agreement, and only one text stands out with an error rate of 11-12%. The chance-corrected agreement measured with π^*_{BED} and α_U is consequently very high. However, note that the flat segmentation neither accounts for the actual embedding nor for the correctness of the segments' types. The α^c_U metric takes these categories into account. It is on average on a very high level. Most texts yield nearly perfect results, only three texts fall out with agreement scores around 60. For the tree metrics, unlabeled and labeled F-score are both around 90%. Three of the texts reach perfect agreement, another three attain nearly perfect F-scores of 95-99%. For the labels, there were only a few disagreements about the categories of subordinate clauses. Here, the most frequent confusion was between

[3] For the calculation of all coefficients of the alpha family, we use the implementation of Meyer et al. (2014).

Metric	Anno_1 vs Anno_2	
	Mean	Std.Dev.
P_k	01.56	±3.55
WinDiff	01.77	±4.04
π^*_{BED}	96.92	±6.64
α_{U}	98.46	±4.07
α^c_{U}	85.95	±17.03
parseval unlab. $F1$	90.18	±7.72
parseval lab. $F1$	89.13	±7.75
matching spans	89.91	
categories κ	88.56	

Table 1: Results of the agreement study between the two annotators in different metrics. For details about the metrics, see Section 3.2.1.

subject and object clauses, typically occurring in the context of expletive constructions, where annotators found it hard to correctly identify the grammatical role of the minor sentence. Structural differences are minimal and mostly occur when one annotator decides for a more fine-grained sub-clausal segmentation. The strongest drop in terms of F-score was observed when one annotator repeatedly split up a conjunction of main sentences, which are supposed to be enclosed by one large main sentence node, into independent main sentences without an enclosing sentence node. The high structural agreement is also reflected by the high ratio of matching spans: Nearly 90% of the spans were exact matches and yielded a substantial agreement of 0.88 κ for segment categories.

It is worth pointing out that the results in Table 1 do not consider the level of sense units (which is step 1 of the annotation procedure, see Section 3.1.1). Identifying sense units is a task that requires a deeper understanding of the text, something that is much easier to achieve for human annotators than for computational models of text processing. We decided to exclude nodes of the sense unit level, in order to facilitate the comparison with the automatic segmenters that we will present in Section 4. Nevertheless, we want to report the annotator agreement for the full annotation task including sense unit identification: All flat segmentation metrics are unaffected by an additional level of nodes in the segmentation trees and thus yield equal results. Only the tree metric reflects the increased structural complexity: The annotators achieve an unlabeled $F1$-score of 89.72 ±11.88, and a labeled $F1$-score of 88.58 ±11.53. These figures are on average only slightly lower than those presented in Table 1, but show a higher variation.

To sum up, the agreement between the annotators is very high, allowing us to conclude that discourse segmentation can be reliably annotated with our scheme. In the light of the above discussion, it is not straightforward to compare the results with others given in the literature, but we want to mention that Tofiloski et al. (2009) measured the annotator agreement for their flat, untyped segmentation task on 10 English texts, and reported a κ of 0.85. For German, the only result we are aware of is the experiment by Versley and Gastel (2012), which lead to a $\kappa > 0.9$, likewise for flat, untyped segmentation.

3.3 Corpus

Having obtained the promising agreement results, we proceeded to annotate the full PCC data set (mentioned at the beginning of this section). The annotation was done by a trained annotator using the EXMARaLDA annotation tool (Schmidt et al., 2011) and corrected in a later consolidation phase. The full corpus contains 176 texts, with 2,180 sentences and about 32,000 tokens. An overview of the frequency of the different segment types that resulted from the annotation is given in Table 2. For illustration, we show an original sentence from the corpus with its annotation:

(9) [*Zu einer Zeit ,* [*in der alles Denkbare auch machbar erscheint ,*]$_{ARR}$ *ist es beruhigend* [*zu wissen ,* [*dass die Rettungskräfte sich nicht erst seit gestern damit befassen ,* [*wie sie die Bürger vor Katastrophen schützen können .*]$_{OBJ}$]$_{OBJ}$]$_{SUB}$]$_{HS}$

Segment Type			Symbol	Count
Main Sentence	complete		HS	2133
	fragment		HSF	285
Minor Sentence	clause constituent	subject clause	SUB	222
		object clause	OBJ	281
		adverbial clause	ADV	346
		predicative clause	PRD	28
	attributive clause	restrictive relative clause	ARR	209
		non-restrictive relative clause	ANR	51
		participle construct	APK	8
		other	ATT	74
		expansive minor clause	WEI	8
	unclear		UNS	5
Fragment	sentence-initial		FRE	55
	sentence-final		FRB	36
				3742

Table 2: Segment types annotated in the corpus

We also analyzed the resulting hierarchical structures of the annotated segments and present the results of this analysis in Table 3. The first column of this table specifies different depths of segment embeddings. These values range from one (a simple,

uncoordinated main sentence without minor clauses) to five. The second column shows the total number of segments annotated at the given depth. Finally, the number of texts exhibiting this maximal segment nestedness is given in the third column. Notice that a clear majority of texts has a maximum embedding depth of three.

For comparison, Afantenos et al. (2010), who work with a French corpus annotated in accordance with SDRT, report that almost 10% of EDUs in their corpus are part of an embedded structure.

Depth of Embedding	Segments	Texts
1	2180	2
2	1335	63
3	206	93
4	20	17
5	1	1
total	3742	176

Table 3: Depth of embedding: The number of segments annotated at this depth (second column) and the number of texts with this maximum depth (third column).

4 Automatic Segmentation

A natural question that arises after measuring human agreement and annotating the complete corpus is how well automatic methods can perform as compared to simple baseline techniques and to the human level of expertise. In this context, we first need to know whether nested or flat segmentation would be more amenable to the automatic processing, and how much the results of the two approaches would differ. To this end, for predicting nested segments, we also have to look into what kind of syntactic information and which form of syntactic structure (syntactic constituents or dependency trees) are suited to make correct predictions about the scope and embedding level of discourse units.

We try to address these and other questions in this section. After summarizing related work on the automatic discourse segmentation of English and German,[4] we establish a straightforward comparison baseline, in which we consider every sentence as a single atomic discourse unit. We use this baseline to compare two more advanced segmentation methods that recursively apply automatic classifiers in order to predict which syntactic constituents or dependents initiate discourse segments. In the final step, we present the results of the flat state-of-the-art segmenter of Feng and Hirst (2014), which we adjusted to the peculiarities of the German language and applied to our corpus. To ease the comparison, we test all three classifiers on our original dataset but

[4] More comprehensive summaries can be found in Stede (2011, pp. 87-97) and Webber et al. (2012, pp. 448-455).

do not make a distinction between the boundaries of sense units (step 1 in the human annotation procedure) and the boundaries of other segments. Finally, we summarize our results and draw conclusions in Section 5, which also includes some suggestions for future research.

4.1 Related Work

As noted by Grosz and Sidner (1986), discourse segments serve as fundamental building blocks in virtually every discourse theory. Even if these theories might disagree on the mechanisms and final results of assembling separate segments into bigger structures, the mere necessity of defining and automatically detecting elementary discourse units has hardly ever been questioned. Accordingly, discourse segmentation plays a crucial role and has attracted much attention in the discourse research community, with by far the most work being done on English.

One of the the earliest attempts at discourse segmentation for RST was made by Le Thanh et al. (2004). In their primarily rule-based approach, the authors consecutively applied a cascade of processing steps: first reading input syntactic trees into a pushdown automaton, storing the non-terminal nodes of these trees on the automaton's stack, and then analyzing these non-terminals with a set of hand-crafted rules, once the system came across a constituency boundary. After applying special heuristics for disambiguating the placement of adverbs and determining the satellite/nucleus status of detected units, the last stage of this system extracted EDUs from clauses and strong cue noun phrases found in text. This system achieved an F-score of 80.35 % on a test corpus of 166 sentences.

Following this line of research, Tofiloski et al. (2009) proposed an automatic segmentation system called `SLSeg` which relied on 12 syntax-based rules and a set of lexical and part-of-speech constraints. The syntactic rules identified potential EDU boundaries between tree nodes, and the constraints removed spurious boundaries surrounding idiomatic phrases (for example, *as it stands*) or inserted new boundaries around units which could not be captured by the syntactic context only (e.g., phrases introduced by *in order to*).

One of the first supervised learning approaches to segmentation was developed by Soricut and Marcu (2003) as a part of the SPADE system. This system took lexicalized syntactic trees as input. The authors computed the probability of inserting a discourse boundary between a child and parent node by estimating the number of lexicalized child-parent pairs with an EDU boundary between them in their training corpus and dividing this count by the total number of all child-parent pairs found in the training set. This system attained an F-score of 83.1% when tested on a set of 38 journal articles.

A different technique was proposed by Sporleder and Lapata (2005). In their work, the authors considered discourse segmentation as a sequence labeling task and tried to solve it using the supervised Boosting approach (Schapire and Singer, 2000). Since this approach aimed not only at segmentation but also at determining the (RST-style) satellite/nucleus status of the detected units, the authors experimented both with

a joint and two-stage approach for solving these two tasks. For segmentation, the two-pass method performed significantly better than the joint technique and gave an improvement in F-score by $\approx 1.5\%$ over the method proposed by Soricut and Marcu (2003).

Fisher and Roark (2007), who obtained an F-score of almost 5% over the SPADE system, used a binary log-linear classifier for recognizing EDU boundaries. The authors experimented with three sets of features, including: *a)* basic finite-state, *b)* context-free, and *c)* a finite-state approximation of context-free features. The former two sets largely coincided with the features used by Sporleder and Lapata (2005), and Soricut and Marcu (2003). As a finite-state approximation, Fisher and Roark (2007) took the output of a shallow syntactic parser and partially lexicalized its chunks. Experiments showed that the best performance could be achieved by using all three sets of features together, thus supporting the claim that full syntax parsing does contribute favorably to discourse segmentation.

Finally, current state-of-the-art results for discourse segmentation of English were obtained by the two-pass system of Feng and Hirst (2014). This system also relies on a sequence labeling approach. Similarly to Soricut and Marcu (2003), the method makes its predictions over pairs of tokens rather than single words. But instead of operating on syntactic trees, this approach expects plain token sequences as input and only incorporates syntax information in the form of features associated with these token pairs. In the first stage, a supervised CRF-classifier makes initial guesses about the potential segment boundaries, which are subsequently corrected by another CRF-model. As shown by Feng and Hirst (2014), both of these strategies (predicting over token pairs and making two-pass predictions) have a crucial positive effect on the net segmentation results, achieving a F-score of 92.6% on the recognition of in-sentence boundaries.

To the best of our knowledge, the only reported attempt at discourse segmentation of German was made by Lüngen et al. (2006). In the initial guessing phase, this approach introduces a potential segment boundary for every comma, conjunction, or parenthesis found in a sentence. In the next step, a special rule-based filter removes margins surrounding enumerations, relative clauses, clausal and infinitival complements, as well as proportional clauses (*the more A, the B*), since these elements do not form independent discourse segments according to the authors' guidelines. The resulting flat segmentation was tested on a corpus of four scientific and two web-published articles, showing an average F-score of 75.57% for the recognition of in-sentence boundaries.

4.2 Baseline

In order to compare our system with these and other approaches, we establish a simple baseline to see how different techniques perform with respect to this rather simplistic method. For this purpose, we have implemented a simple segmentation module which creates a single discourse unit for each sentence identified by a customary sentence splitter (specifically, we are using the one from the OpenNLP tool suite[5]). This method

[5] `https://opennlp.apache.org/`

is expected to work correctly for most of the sentences except for the cases when the annotators identified sub-clause EDUs or considered incomplete main clauses (see the first two steps of the annotation procedure) as separate discourse units.

4.3 Hierarchical Segmentation

4.3.1 Constituency Parser Model

The first segmentation system that we are going to compare against the baseline is called `BitParSegmenter`. As suggested by the name, this system makes its predictions over syntactic constituents that are obtained from the output of the BitPar constituency parser (Schmid, 2004). In all our experiments, we used the parsing model trained on the TIGER corpus (Brants et al., 2004).

In order to train our segmentation model, we automatically processed raw sentences from our corpus with BitPar. This gave us a set of 1,911 constituency trees with a total of 50,402 non-terminal and 32,872 terminal nodes (tokens).[6] Since the results of the built-in BitPar tokenizer disagreed with the gold tokenization from our segmentation data, we next applied the Needleman-Wunsch algorithm (Needleman and Wunsch, 1970) in order to unify both splittings. In a concluding step, we consecutively aligned each constituent of every parse tree with a corresponding discourse segment (if there was one). To find such correspondences, we represented each constituent in question as a set of uniquely numbered tokens that belonged to that node, translated this token set to a respective set of discourse tokens using the previously computed alignment, and eventually checked whether there was a segment in our gold data that fully agreed with the tokens belonging to the constituency node under scrutiny.

By applying this procedure, we were able to align 2,941 out of 50,320 non-terminals (5.84%) with at least one discourse segment (we skipped those non-terminals which consisted solely of punctuation marks or whose aligned tokens resulted in an empty list for the discourse tokenization). A detailed breakdown of 10 most frequently matching constituent and segment types is given in Table 4. Conversely, 78.76% of all discourse segments had a corresponding constituent in the parse trees. This figure also gives us an upper bound on the classification results for our segmenter (i.e., even with a perfect recognition of which constituents initiated which types of segments, we still would be able to correctly reconstruct only $\approx 80\%$ of all EDUs).

After aligning syntactic constituents with their respective discourse counterparts, we constructed the training set by extracting features from every constituency (sub-)tree and taking the type of the discourse segment aligned with its top constituent as the gold label for our prediction. (Sub-)trees which did not have a corresponding discourse segment were assigned the gold class NONE.

We used the following types of attributes as features:

- the set of all terminals (tokens) comprised by the top constituent;

[6] Due to the automatic splitting with BitPar's scripts, the number of tokens and constituency trees in this set differs from the number of words and sentences in the manually labeled corpus.

Constituent Type	Segment Type	Count
TOP	HS	1,432
S-TOP	HS	253
S-MO	ADV	153
S-RC	ARR	140
TOP	HSF	100
S-OC	OBJ	86
S-SB	SUB	84
S-OC	HS	44
S-RC	ANR	40
VP-OC/zu	OBJ	39

Table 4: Most frequently matching constituent and segment types.

- the first and the last token of the head constituent as two separate features;
- the syntactic label of the head node and the syntactic label of its right-most descendant;
- the syntactic type of the parent (sub-)tree, if there was one;
- the syntactic type, the first, and the last tokens of the immediate left and right siblings of the (sub-)tree in question;
- and, finally, the height of the tree as a numeric feature.[7]

After comparing several different machine learning approaches including the random forest classifier (Liaw and Wiener, 2002), the decision tree method (Breiman et al., 1984), and the k-nearest neighbors algorithm (Fix and Hodges, 1989), we chose the linear support vector classification system (Fan et al., 2008) with the Crammer-Singer multi-class strategy (Crammer and Singer, 2002) due to both its superior performance and much faster training times as compared to the other methods.

Once the classifier was trained, obtaining the final automatic segmentation was relatively straightforward: We simply traversed each node of the input syntactic trees in the depth-first-search order and let the trained model predict the segment class of the processed nodes. Whenever the classifier made a prediction other than NONE (i.e., it decided that the constituent in question was in fact giving rise to a segment), we constructed an EDU of the predicted class for this constituent and recursively processed the children of that syntactic node, storing the results of this recursion as leaves of the newly created EDU (these results in turn could be either plain tokens or further

[7] Notice that, since BitPar does not include POS tags in its output, our features do not make use of them in this model.

discourse segments). A sample constituency tree with the classifier's predictions (shown in brackets next to the node names) and resulting segmentation is shown in Figure 1.

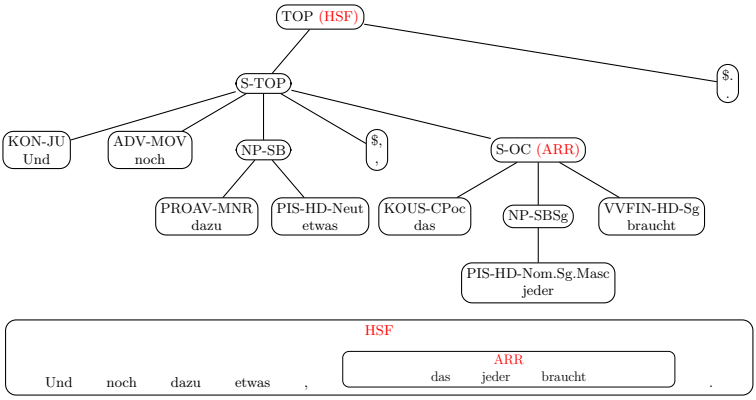


Figure 1: Example of a constituency tree with resulting discourse segmentation.

4.3.2 Dependency Parser Model

The second segmentation system that we are going to compare against the baseline is called `MateSegmenter`. This system makes its predictions over the sub-graphs of a dependency parse derived by the mate-parser (Bohnet, 2010).

In order to generate a training set for this approach, we first parsed the raw corpus with the mate-parser, getting 2,013 dependency graphs with a total of 32,838 tokens.[8] Similarly to the training procedure for the previous model, we then aligned the gold tokenization with the automatic token splitting using the Needleman-Wunsch algorithm. In the next step, all dependency sub-graphs were aligned with the annotated discourse segments by matching spans of uniquely numbered tokens. Of all dependency sub-graphs, 2,983 directly corresponded to a discourse segment (9.7% of all dependency sub-graphs). Conversely, 2,989 of the annotated discourse segments had a corresponding dependency sub-graph (79.8% of all discourse segments). This gave us a similar upper bound for the automatic classification as with the constituency parser approach.

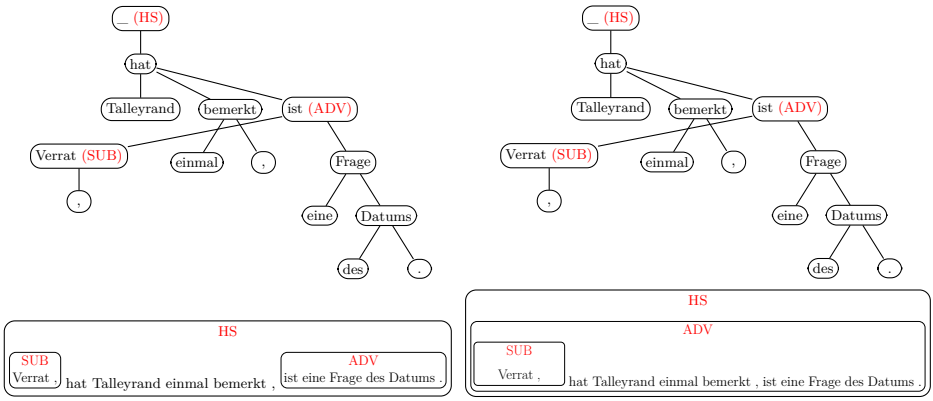
The classification items were constructed by extracting features from every dependency sub-graph. The target class for each item was the type of the aligned discourse segment or NONE, if no alignment could be established. We used the following types of attributes of the sub-graph as features:

- the token and the part-of-speech of the sub-graph's root;

[8] As with the previous parser model, the number of tokens and dependency trees differs from the number of words and sentences in the manually labeled corpus due to the automatic splitting.

- the token and the part-of-speech of the head of the sub-graph's root;
- the dependency relation between the sub-graph's root and its head;
- pairwise and triple combinations of the above three features;
- the first and the last token of the sub-graph's span;
- the token to the left and to the right of the sub-graph's span;
- unigram features for all tokens in the sub-graph's span;
- the length of the sub-graph's span measured in tokens;
- the number of punctuation tokens in the sub-graph's span.

As with the previous method, we compared several machine learning approaches and chose the linear support vector classifier. However, the construction of the final segmentation with the trained model was not as easy as for the `BitParSegmenter`. This was mainly due to the presence of non-projective edges in the input dependency trees. Once an ancestor node of such an edge was predicted to initiate a segment, simply putting all descendants of that node into a single discourse segment resulted in intertwined discontinuous discourse units, which was not a valid segment structure according to our guidelines.



(a) Greedy approach. **(b)** Generous approach.

Figure 2: Examples of dependency trees with resulting discourse segmentation.

To overcome this issue, we devised two different approaches: *greedy* and *generous*. With the former technique, we constructed a new discourse unit only of those tokens that formed a continuous span around the node-token that gave rise to the segment

Classifier	Macro-$F1$		Micro-$F1$		$F1_{tp,fp}$
	Mean	Std.Dev.	Mean	Std.Dev.	
`BitParSegmenter`	39.38%	±6.33%	97.49%	±0.21%	77%
`MateSegmenter`	47.8%	±6.13%	96.6%	±0.34%	80.05%

Table 5: Intrinsic evaluation of syntax-based classifiers.

(i.e., we neglected the part that was connected to this node via a non-projective edge). With the latter method, we collected all child tokens of the segment-generating node, and added the tokens that disrupted these children (i.e., those that occurred between the projective and the non-projective descendants) to the discourse segment as well. Examples of both approaches are shown in Figures 2a and 2b.

After testing both methods, we opted for the greedy technique, since it achieved slightly better results than the generous method, though the difference between the two approaches was not very big (the difference in P_k only amounted to 0.1 and the difference in π^*_{BED} only ran up to 0.2%).

4.3.3 Results

In order to train both classifiers and obtain segments based on their predictions, we applied 10-fold cross validation over the whole training corpus by successively splitting it into ten parts and subsequently training the classifier on nine of ten fractions, then applying the resulting system to the remaining part. We performed both an intrinsic and an extrinsic evaluation of the results.

In the intrinsic evaluation, we assessed how good each classifier was at predicting the correct segment classes (including NONE) for syntactic constituents and dependency nodes respectively. To do so, we estimated the mean and the standard deviation of the micro- and macro-averaged F-scores obtained in all 10 folds. The results of this evaluation are shown in Table 5.[9]

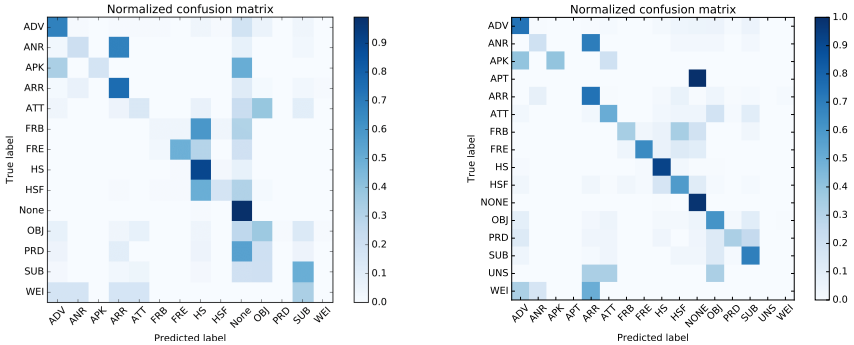
As one can see from the table, the dependency-based segmenter clearly outperforms the constituency-based one, even though our preliminary estimations of the respective upper bounds of these approaches suggested equal results. Furthermore, we also can observe a dramatic difference between the macro- and micro-averaged F-scores for both segmenters. This discrepancy can be explained by a skewed distribution of the segment classes in our corpus and different susceptibility of the two metrics to such

[9] All evaluations were carried out using cross validation with the released versions of the segmenters (v.0.0.1.dev1). To ease the alignment, we have also removed the backslash escapes of quotation marks and slashes that were introduced by BitPar. Note, however, that the pre-trained model delivered with the module was trained on the whole corpus and not on the nine cross-validation folds, so it might therefore produce slightly different results.

unbalanced data: while the micro-averaged F-score counts the total number of correct and wrong decisions, the macro-metric takes the average of the F-scores for predicting each particular segment class. The majority of the items in our data, however, have the gold class NONE which is also correctly predicted in most of the cases (as suggested by the micro-score). But since there are also many less frequent segment classes in the data set (some of which only occur a few times in our corpus), making even a few wrong predictions for these items leads to considerably lower macro-averaged results.

Another source of bias, which might significantly affect the evaluation, can be due to a skewed distribution of different gold classes across multiple folds. This problem was brought to our attention by one of the reviewers and particularized in the work of Forman and Scholz (2010). As a remedy for this issue, the authors suggest using an alternative average metric which they call $F1_{tp,fp}$ and which is calculated as a ratio $F1_{tp,fp} = \frac{2*TP}{2*TP+FP+FN}$, where TP, FP, and FN denote the total counts of true positive, false positive, and false negative predictions in all test folds of cross validation. As can be seen from the last column in Table 5, the results of this metric still correlate with other measurements and are situated between the micro- and macro-estimations.

To get a better intuition about the particular kinds of errors made by each segmenter, we also generated a confusion matrix of their wrong decisions (see Figure 3). Our goal was to see whether these two syntax-based approaches had complementary strengths and weaknesses or rather showed approximately the same behavior. As can be seen from the figure, the `BitParSegmenter` clearly tends to under-segment the input sentences. While this tendency is generally also observed for the `MateSegmenter`, it is much less acute there, and the confusion classes are spread more uniformly.



(a) `BitParSegmenter` **(b)** `MateSegmenter`

Figure 3: Confusion matrices for constituency- and dependency-based segmenters.

The classification results on their own are not very informative about the structural properties of the resulting segments, though. Indeed, the fact that some syntactic node will be correctly recognized as a segment-initiating item does not necessarily imply that

the recognized segment will be correctly integrated into the overall segment structure (if, for example, the levels of syntactic dependencies are confused in the syntax tree).

To check whether such discrepancies were present in our data, we performed an extrinsic evaluation of the resulting segmentation by applying the same agreement tests to the output of the automatic systems as we did for estimating the inter-annotator agreement between the human experts. In contrast to the previous IAA study, however, where we estimated the percentage of matching spans as a ratio between the spans annotated by both experts and the total number of spans annotated by just one annotator (whom we considered as the gold reference), this time, we had to introduce two metrics: matching $\text{spans}_{\text{pred}}$ and matching $\text{spans}_{\text{gold}}$. With the former benchmark, we computed the ratio of matching spans with respect to all *predicted* spans. With the latter metric, we estimated the percentage of matching segments with respect to the total number of *gold* segments as we did in the previous estimations.

The results of this evaluation are presented in Table 6. As can be seen from the table, the extrinsic figures still strongly correlate with the intrinsic macro-$F1$ scores. Furthermore, we can also observe that both automatic segmenters significantly outperform the baseline results and that the dependency-based approach generally yields better scores from both intrinsic and extrinsic perspectives.

Metric	Baseline		`BitParSegmenter`		`MateSegmenter`	
	Mean	Std.Dev.	Mean	Std.Dev.	Mean	Std.Dev.
P_k	30.99	±10.10	11.47	±4.39	5.37	±4.64
WinDiff	41.62	±15.23	18.61	±5.45	6.49	±5.42
π^*_{BED}	51.72	±11.49	64.43	±8.49	87.42	±9.77
α_{U}	47.54	±19.95	66.77	±13.66	89.12	±11.69
α^c_{U}	49.33	±17.56	59.98	±17.19	66.98	±19.53
parseval unlab. $F1$	25.12	±11.48	64.53	±17.39	78.52	±14.63
parseval lab. $F1$	25.12	±11.48	60.66	±17.53	72.07	±16.62
matching $\text{spans}_{\text{pred}}$	48.38		34.43		68.69	
matching $\text{spans}_{\text{gold}}$	50.87		48.16		78.96	
categories κ	65.24		72.07		74.91	

Table 6: Extrinsic evaluation of syntax-based segmenters.

4.4 Flat Segmentation

Instead of taking syntactic trees as input and trying to reconstruct discourse segments based on the predictions for their nodes, another viable alternative is to rely on plain sequences of tokens.

A clear advantage of this approach is that, in contrast to syntax-based systems, the underlying input data structure (token sequence), which serves as a basis for constructing the segments, is trivially guaranteed to be flawless and hence more amenable to a correct segmentation. At the same time, an incorrectly built syntactic tree (which not

infrequently happens in parsing) will almost inevitably lead to wrong segments even with correct classifiers.

A disadvantage of this method, however, is that plain token chains are much less suitable for hierarchical segmentation. Almost all approaches relying on token sequences therefore produce only a flat segmentation of the input sentences. Examples of such systems include the works proposed by Hernault et al. (2010a) and Bach et al. (2012). The system of Hernault et al. was, to the best of our knowledge, the first attempt to tackle the segmentation problem as a sequence labeling task based on Conditional Random Fields (Lafferty et al., 2001). This system operated on strings of tokens, but extensively utilized syntactic features obtained from parsers. Bach et al. (2012) later refined this method by first obtaining N-best sequences from a base CRF-classifier and then re-ranking those sequences judging by the properties of syntactic parse trees that bound or split potential segments.

The state-of-the art results for this type of processing were obtained by the method proposed by Feng and Hirst (2014). In their approach, the authors devised a two-pass system which first made initial guesses over a sequence of token pairs, predicting 'B' if there was a segment boundary in between two adjacent tokens and 'C' otherwise (see Figure 4). These guesses were subsequently corrected by another CRF-classifier (to be explained below).

For the purpose of our experiments, we adjusted the system of Feng and Hirst (2014) to the specifics of German text processing and tested it on our corpus in the same cross-validation fashion as we did for the hierarchical systems explained above. For this purpose, we first converted hierarchical discourse structures annotated in our data to a flat segmentation, as explained earlier and as illustrated in Figure 4. We then applied the method of Feng and Hirst (2014) and separately tested its one- and two-pass variants.

Input sentence: `(HS Doch das Angebot , (ATT nahe der Haustür den Personennahverkehr nutzen zu können , ) wollen nicht alle . )`

Label sequence: `C C C B C C C C C C C C B C C C`

Figure 4: Flattening of a hierarchical segment structure.

4.4.1 One-pass Model

The one-pass variant corresponds to a plain CRF classifier that takes a sequence of feature representations for token pairs and returns the most probable assignment of segment boundaries for this sequence.

Following the original approach, we used the same set of feature attributes for each pair of adjacent tokens:

- the part-of-speech tags and the lemmas of both tokens;
- Boolean features indicating whether the first or the last token of the pair were located at the beginning or the end of a sentence;
- The part-of-speech tags of the top dependency nodes for which the first and the last token of the pair were the left- and the right-most children respectively;[10]
- the height of the sub-trees whose left- and right-most children were the first and the last token of the pair respectively;
- the top production rule of the largest syntactic constituents starting from the first or ending with the last token of the pair;
- the same set of features for the left and right neighbor token pairs.

4.4.2 Two-pass Model

After the first stage is complete, a second pass of the algorithm corrects the hypothesized segment boundaries by effectively applying the same CRF-method and the same set of token-pair features as in the first pass, plus taking into account the global properties of the potential segments such as:

- the part-of-speech tags and the lemmas of the tokens located at the left / right boundary of the enclosing discourse segment;
- the distance to the nearest left / right segment boundary;
- the number of syntactic nodes between the token pair in question and its nearest discourse segment boundaries;
- the part-of-speech tag of the top node of the lowest sub-tree that encompasses all tokens between the respective token pair and its left / right neighboring segment boundary.

4.4.3 Results

To evaluate the one- and two-pass variants of this approach, we applied the same 10-fold cross validation strategy as we did for the syntax-based hierarchical methods. The results of the macro- and micro-averaged F-scores for this two-class classification task are shown in Table 7. We also performed an extrinsic evaluation of the resulting segment structures whose results are presented in Table 8.

[10] In this regard, our features slightly differ from the ones adopted by Feng and Hirst (2014). Since syntactic features used in their work were obtained from the output of the Stanford Parser (Klein and Manning, 2003), the authors used syntactic labels of tree constituents at this point. We, however, operate on the output of the Mate parser and use the part-of-speech tags of the top nodes instead, since this parser does not provide constituents.

Classifier	Macro-$F1$		Micro-$F1$		$F1_{tp,fp}$
	Mean	Std.Dev.	Mean	Std.Dev.	
One-pass model	87%	±1%	97.67%	±0.13%	76.33
Two-pass model	93.19%	±1.5%	98.66%	±0.26%	88.26

Table 7: Intrinsic evaluation of CRF-based classifiers.

Metric	Baseline		One-pass Model		Two-pass Model	
	Mean	Std.Dev.	Mean	Std.Dev.	Mean	Std.Dev.
P_k	30.99	±10.10	9.24	±5.80	4.29	±4.03
WinDiff	41.62	±15.23	10.91	±6.69	5.14	±4.61
π^*_{BED}	51.72	±11.49	81.03	±10.08	89.96	±8.73
α_{U}	47.54	±19.95	81.69	±14.32	91.18	±10.03
α^c_{U}	49.33	±17.56	46.67	±18.89	46.39	±19.88
parseval unlab. $F1$	25.12	±11.48	74.25	±14.37	77.62	±15.17
parseval lab. $F1$	25.12	±11.48	70.46	±16.47	73.23	±16.86
matching $\text{spans}_{\text{pred}}$	48.38		53.47		59.99	
matching $\text{spans}_{\text{gold}}$	50.87		74.08		84.92	
categories κ	65.24		23.67		18.57	

Table 8: Extrinsic evaluation of CRF-based segmenters.

As can be seen from the table, even the one-pass model outperforms the baseline method by a factor of three in terms of the P_k measure. This improvement is even bigger when measured with WinDiff, where the error drop is quadruple. Estimations with other metrics yield consistent results: π^*_{BED} is improved by 29.31%, whereas α_{U} is increased by 34.15%. The most drastic quality boost, however, can be observed for the parseval measurements: here the unlabeled F-score changes from 25.12% to almost 75%, and the labeled metric surpasses the considerably high 70% landmark.

These results are further improved by the two-pass classifier, which not only outperforms the one-pass approach but also yields better scores for P_k, WinDiff, π^*_{BED}, α_{U}, and the labeled parseval $F1$ than the tree-based MateSegmenter. Two-pass CRFs are still approximately on par with the mate system in terms of the unlabeled parseval $F1$, but perform significantly worse than that when measured with α^c_{U}. An obvious explanation for this is that the latter metric puts much weight on the correct prediction of segment categories – a part which we deliberately sacrificed when flattening the segment structures. Nevertheless, we consider it as an interesting finding that a plain sequence-based segmentation method forms a viable alternative to the tree-based approaches in many other regards.

5 Summary

The central contribution of this paper is an implemented, comparative approach to discourse segmentation of German texts. We provide a thorough discussion of the evaluation problem for flat and hierarchical segmentation, and measure our inter-annotator agreement and the performance of the automatic approaches by various means. For applications where a hierarchical and possibly also labeled segmentation is advantageous, we offer classifiers operating on the output of state-of-the-art German syntax parsers (mate and BitPar). Our results show an advantage for mate as the basis of a segmenter, but this could be due to the absence of POS tags in the BitPar output, which thus did not enter our feature set. We leave it to future work to determine whether a POS-enhanced version of the feature set would improve the results (or, conversely, whether the mate results would deteriorate if the POS features were left out). To our knowledge, this is the first comparison of the two linguistic parsing strategies regarding their suitability for a subsequent discourse segmentation step, and our error analysis indicates that the difference in the results stems from a tendency to under-segmentation on the side of BitPar. The general technique we use resembles that of Soricut and Marcu (2003). We do not comment on the relationship between the evaluation results, because syntactic parsing of English and German are clearly not of the same difficulty, so there is little point in comparing our numbers to those of SPADE.

When hierarchy and labels are not needed, a CRF model yields very good results. Our implementation followed that of Feng and Hirst (2014), but we made a number of adaptations that were necessary for applying this technique to German. Again, we do not compare the German versus English results, but we notice that for the German data, the CRF approach yields better performance than the tree classification techniques; but that is probably not surprising, because the task of flat segmentation is somewhat easier.

Our three implementations are freely available online[11] and thus constitute – to the best of our knowledge – the first re-usable modules for this discourse processing task for German. Likewise, our corpus of annotated hierarchical and labeled discourse segments as well as the accompanying guidelines for this corpus are also released[12] and can be freely used for research purposes.

References

Afantenos, S. D., Denis, P., Muller, P., and Danlos, L. (2010). Learning recursive segments for discourse parsing. In Calzolari, N., Choukri, K., Maegaard, B., Mariani, J., Odijk, J., Piperidis, S., Rosner, M., and Tapias, D., editors, *Proceedings of the International Conference on Language Resources and Evaluation, LREC 2010, 17-23 May 2010, Valletta, Malta*. European Language Resources Association.

Asher, N. and Lascarides, A. (2003). *Logics of Conversation*. Cambridge University Press, Cambridge.

[11] https://github.com/discourse-lab/DiscourseSegmenter
[12] http://angcl.ling.uni-potsdam.de/resources/pcc.html

Bach, N. X., Nguyen, M. L., and Shimazu, A. (2012). A reranking model for discourse segmentation using subtree features. In *Proceedings of the SIGDIAL 2012 Conference, The 13th Annual Meeting of the Special Interest Group on Discourse and Dialogue, 5-6 July 2012, Seoul National University, Seoul, South Korea*, pages 160–168. The Association for Computer Linguistics.

Beeferman, D., Berger, A., and Lafferty, J. (1999). Statistical models for text segmentation. *Mach. Learn.*, 34(1-3):177–210.

Black, E., Abney, S. P., Flickenger, D., Gdaniec, C., Grishman, R., Harrison, P., Hindle, D., Ingria, R., Jelinek, F., Klavans, J. L., Liberman, M., Marcus, M. P., Roukos, S., Santorini, B., and Strzalkowski, T. (1991). A procedure for quantitatively comparing the syntactic coverage of english grammars. In *Speech and Natural Language, Proceedings of a Workshop held at Pacific Grove, California, USA, February 19-22. 1991*. Morgan Kaufmann.

Bohnet, B. (2010). Very high accuracy and fast dependency parsing is not a contradiction. In *Proceedings of the 23rd International Conference on Computational Linguistics*, COLING '10, pages 89–97, Stroudsburg, PA, USA. Association for Computational Linguistics.

Brants, S., Dipper, S., Eisenberg, P., Hansen, S., König, E., Lezius, W., Rohrer, C., Smith, G., and Uszkoreit, H. (2004). Tiger: Linguistic interpretation of a german corpus. *Research on Language and Computation*, 2(4):597–620.

Breiman, L. et al. (1984). *Classification and Regression Trees*. Chapman & Hall, New York.

Bußmann, H. (2002). *Lexikon der Sprachwissenschaft*. Kröner, Stuttgart.

Carlson, L. and Marcu, D. (2001). *Discourse tagging manual. Technical report*. Univ. of Southern California/ISI.

Cohen, J. (1960). A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement*, 20(1):37–46.

Crammer, K. and Singer, Y. (2002). On the algorithmic implementation of multiclass kernel-based vector machines. *J. Mach. Learn. Res.*, 2:265–292.

Fan, R.-E., Chang, K.-W., Hsieh, C.-J., Wang, X.-R., and Lin, C.-J. (2008). LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, 9:1871–1874.

Feng, V. W. and Hirst, G. (2014). Two-pass discourse segmentation with pairing and global features. *CoRR*, abs/1407.8215.

Fisher, S. and Roark, B. (2007). The utility of parse-derived features for automatic discourse segmentation. In *Proc. of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 488–495, Prague, Czech Republic. Association for Computational Linguistics.

Fix, E. and Hodges, J. L. J. (1989). Discriminatory analysis. Nonparametric discrimination: Consistency properties. *International Statistical Review*, 57(3):238–247.

Forman, G. and Scholz, M. (2010). Apples-to-apples in cross-validation studies: pitfalls in classifier performance measurement. *SIGKDD Explorations*, 12(1):49–57.

Fournier, C. (2013). Evaluating text segmentation using boundary edit distance. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, ACL 2013, 4-9 August 2013, Sofia, Bulgaria, Volume 1: Long Papers*, pages 1702–1712. The Association for Computer Linguistics.

Grosz, B., Joshi, A., and Weinstein, S. (1995). Centering: A framework for modelling the local coherence of discourse. *Computational Linguistics*, 21(2):203–226.

Grosz, B. J. and Sidner, C. L. (1986). Attention, intentions, and the structure of discourse. *Computational Linguistics*, 12(3):175–204.

Hearst, M. A. (1997). TextTiling: Segmenting text into multi-paragraph subtopic passages. *Computational Linguistics*, 23(1):33–64.

Hernault, H., Bollegala, D., and Ishizuka, M. (2010a). A sequential model for discourse segmentation. In Gelbukh, A. F., editor, *Computational Linguistics and Intelligent Text Processing, 11th International Conference, CICLing 2010, Iasi, Romania, March 21-27, 2010. Proceedings*, volume 6008 of *Lecture Notes in Computer Science*, pages 315–326. Springer.

Hernault, H., Prendinger, H., duVerle, D., and Ishizuka, M. (2010b). Hilda: A discourse parser using support vector machine classification. *Dialogue and Discourse*, 1(3):1–33.

Ji, Y. and Eisenstein, J. (2014). Representation learning for text-level discourse parsing. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014, Baltimore/MD*, pages 13–24.

Klein, D. and Manning, C. D. (2003). Accurate unlexicalized parsing. In Hinrichs, E. W. and Roth, D., editors, *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics, 7-12 July 2003, Sapporo Convention Center, Sapporo, Japan.*, pages 423–430. ACL.

Krippendorff, K. (1995). On the reliability of unitizing continuous data. *Sociological Methodology*, 25:47–76.

Krippendorff, K. (2004). Measuring the reliability of qualitative text analysis data. *Quality & Quantity*, 38:787–800.

Lafferty, J. D., McCallum, A., and Pereira, F. C. N. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In Brodley, C. E. and Danyluk, A. P., editors, *Proceedings of the Eighteenth International Conference on Machine Learning (ICML 2001), Williams College, Williamstown, MA, USA, June 28 - July 1, 2001*, pages 282–289. Morgan Kaufmann.

Lappin, S. and Leass, H. J. (1994). An algorithm for pronominal anaphora resolution. *Computational Linguistics*, 20(4):535–561.

Le Thanh, H., Abeysinghe, G., and Huyck, C. (2004). Generating discourse structures for written text. In *Proc. of the 20th International Conference on Computational Linguistics (COLING)*, pages 329–335, Geneva/Switzerland.

Liaw, A. and Wiener, M. (2002). Classification and regression by randomforest. *R News*, 2(3):18–22.

Lüngen, H., Puskás, C., Bärenfänger, M., Hilbert, M., and Lobin, H. (2006). Discourse segmentation of german written texts. In Salakoski, T., Ginter, F., Pyysalo, S., and Pahikkala, T., editors, *Advances in Natural Language Processing, 5th International Conference on NLP, FinTAL 2006, Turku, Finland, August 23-25, 2006, Proceedings*, volume 4139 of *Lecture Notes in Computer Science*, pages 245–256. Springer.

Mann, W. and Thompson, S. (1988). Rhetorical structure theory: Towards a functional theory of text organization. *TEXT*, 8:243–281.

Marcu, D. (2000). *The theory and practice of discourse parsing and summarization*. MIT Press, Cambridge/MA.

Meyer, C. M., Mieskes, M., Stab, C., and Gurevych, I. (2014). Dkpro agreement: An open-source java library for measuring inter-rater agreement. In Tounsi, L. and Rak, R., editors, *Proceedings of the 25th International Conference on Computational Linguistics: System Demonstrations (COLING)*, pages 105–109, Dublin, Ireland. Dublin City University and Association for Computational Linguistics.

Miltsakaki, E. (2002). Toward an aposynthesis of topic continuity and intrasentential anaphora. *Computational Linguistics*, 28(3):319–355.

Mosegaard-Hanse, M.-B. (1998). *The Function of Discourse Particles: A Study with Special Reference to Spoken Standard French*. John Benjamins, Amsterdam and Philadelphia.

Muller, P., Afantenos, S., Denis, P., and Asher, N. (2012). Constrained decoding for text-level discourse parsing. In *Proc. of the International Conference on Computational Linguistics (COLING)*, Mumbai, India.

Needleman, S. and Wunsch, C. (1970). A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of Molecular Biology*, 48(3):443–453.

Pevzner, L. and Hearst, M. A. (2002). A critique and improvement of an evaluation metric for text segmentation. *Computational Linguistics*, 28(1):19–36.

Prasad, R., Dinesh, N., Lee, A., Miltsakaki, E., Robaldo, L., Joshi, A. K., and Webber, B. L. (2008). The penn discourse treebank 2.0. In *Proceedings of the International Conference on Language Resources and Evaluation, LREC 2008, 26 May - 1 June 2008, Marrakech, Morocco*. European Language Resources Association.

Schapire, R. E. and Singer, Y. (2000). Boostexter: A boosting-based system for text categorization. *Machine Learning*, 39(2/3):135–168.

Schmid, H. (2004). Efficient parsing of highly ambiguous context-free grammars with bit vectors. In *COLING 2004, 20th International Conference on Computational Linguistics, Proceedings of the Conference, 23-27 August 2004, Geneva, Switzerland*.

Schmidt, T., Wörner, K., Hedeland, H., and Lehmberg, T. (2011). New and future developments in EXMARaLDA. In Schmidt, T. and Wörner, K., editors, *Multilingual Resources and Multilingual Applications. Proceedings of GSCL Conference 2011 Hamburg*.

Schmitt, H. (2000). *Zur Illokutionsanalyse monologischer Texte*. Peter Lang, Frankfurt.

Soricut, R. and Marcu, D. (2003). Sentence-level discourse parsing using syntactic and lexical information. In *Proc. of the Human Language Technology Conference of the North American Chapter of the ACL*, pages 149–156, Edmonton/Canada.

Sporleder, C. and Lapata, M. (2005). Discourse chunking and its application to sentence compression. In *Proc. of the HLT/EMNLP Conference*, pages 257–264, Vancouver.

Stede, M. (2011). *Discourse Processing*. Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers.

Stede, M., Mamprin, S., and Peldszus, A. (2015). Diskurssegmentierung. In Stede, M., editor, *Handbuch Textannotation: Potsdamer Kommentarkorpus 2.0*, volume 8 of *Potsdam Cognitive Science Series*, pages 23–44. Universitätsverlag Potsdam.

Stede, M. and Neumann, A. (2014). Potsdam Commentary Corpus 2.0: Annotation for discourse research. In *Proc. of the International Conference on Language Resources and Evaluation (LREC)*, Reykjavik.

Stede, M. and Peldszus, A. (2012). The role of illocutionary status in the usage conditions of causal connectives and in coherence relations. *Journal of Pragmatics*, 44(2):214–229.

Taboada, M. and Zabala, L. H. (2008). Deciding on units of analysis within centering theory. *Corpus Linguistics and Linguistic Theory*, 4(1):63–108.

Tetreault, J. (2001). A corpus-based evaluation of centering and pronoun resolution. *Computational Linguistics*, 27(4):507–520.

Tofiloski, M., Brooke, J., and Taboada, M. (2009). A syntactic and lexical-based discourse segmenter. In *Proc. of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP (Short Papers)*, pages 77–80, Suntec, Singapore. Association for Computational Linguistics.

Versley, Y. and Gastel, A. (2012). Linguistic tests for discourse relations in the TüBa-D/Z corpus of written German. *Dialogue & Discourse*, 4 (2).

Webber, B. L., Egg, M., and Kordoni, V. (2012). Discourse structure and language technology. *Natural Language Engineering*, 18(4):437–490.

Lucie Flekova, Tahir Sousa, Margot Mieskes, Iryna Gurevych

Document-level school lesson quality classification based on German transcripts

Analyzing large-bodies of audiovisual information with respect to discourse-pragmatic categories is a time-consuming, manual activity, yet of growing importance in a wide variety of domains. Given the transcription of the audiovisual recordings, we propose to model the task of assigning discourse-pragmatic categories as supervised machine learning task. By analyzing the effects of a wide variety of feature classes, we can trace back the discourse-pragmatic ratings to low-level language phenomena and better understand their dependency. The major contribution of this article is thus a rich feature set to analyze the relationship between the language and the discourse-pragmatic categories assigned to an analyzed audiovisual unit. As one particular application of our methodology, we focus on modelling the quality of lessons according to a set of discourse-pragmatic dimensions. We examine multiple lesson quality dimensions relevant for educational researchers, e.g. to which extent teachers provide objective feedback, encourage cooperation and pursue thinking pathways of students. Using the transcripts of real classroom interactions recorded in Germany and Switzerland, we identify a wide range of lexical, stylistic and discourse-pragmatic phenomena, which affect the perception of lesson quality, and we interpret our findings together with the educational experts. Our results show that especially features focusing on discourse and cognitive processes are beneficial for this novel classification task, and that this task has a high potential for automated assistance.

1 Introduction

Analyzing large-bodies of audiovisual information with respect to discourse-pragmatic categories is a time-consuming, manual activity. This task is of great importance in a wide variety of domains, including education, psychology, criminal forensics, sociology, and human resources management, since the volume of audiovisual information is steadily growing. At the same time, the tools for automatic speech recognition are getting ever more mature to be applied under realistic conditions.

Given the transcription of the audiovisual recordings, we propose to model the task of assigning discourse-pragmatic categories as supervised machine learning task. The major contribution of this article is thus a rich feature set to analyze the relationship between the language and the discourse-pragmatic categories assigned to an analyzed audiovisual unit. By analyzing the effects of a wide variety of feature classes, we can

trace back the discourse-pragmatic ratings to low-level language phenomena and better understand their dependency.

As one particular application of our methodology, we focus on modelling the quality of lessons according to a set of discourse-pragmatic dimensions. Educational researchers extensively analyze the interaction between teachers and students in all age groups in order to find components of teaching effectiveness. A commonly employed method is based on videography, in which acoustical and visual elements of lessons are recorded. Researchers analyze these recordings and assess the quality of interaction both between the teacher and the students, and among the students, on a variety of levels, in order to design teacher trainings and educational interventions. This assessment is a very tedious process, as multiple trained experts have to evaluate the quality independently. During this process, each annotator has to go over the complete material several times in order to assign quality scores on the various dimensions and levels of depth relevant for educational researchers. For details on the procedure see e.g. Rakoczy (2006). Currently, this task is carried out completely manually, which does not allow for scaling such studies to a wider scope. Supporting it with automatic methods would help to reduce this impediment.

We show that our feature set can reveal important insights into the nature of language associated with the studies of quality dimensions of the lessons. These findings can be utilized in a number of ways: i) to automate the task of analyzing the quality dimensions manually, and thus scale the technology to monitor huge amounts of data, and ii) to provide feedback to educational specialists about the lessons and help to conceptualize the appropriate interventions. The feature set itself can be re-used in the future for similar tasks.

In close cooperation with educational researchers, we processed a data set of transcripts of German mathematics lessons and created multiple machine learning models to classify the lessons on a range of quality dimensions. Thanks to the manual expert annotations, provided by educational researchers, we were able to train and test each model on a highly reliable gold standard. We make both the data[1] and the software[2] available to the research community.

Our key contribution is two-fold: Firstly, we determine which aspects of the verbal behaviour expressed in the transcripts (such as sentiment polarity or discourse structure) are predictive for the ratings of lesson quality dimensions (such as constructive feedback, student cooperation and reasoning path development), and we offer interpretation of our findings. Secondly, by modeling the problem as a text classification task with NLP features, we also demonstrate that this educational research task has a high potential for automation. Our initial results show that especially features related to e.g. discourse, sentiment and cognitive processes allow for good lesson classification within the quality dimensions studied here. The data set of transcripts is publicly available. To our

[1] `https://www.ukp.tu-darmstadt.de/index.php?id=11716`

[2] `https://github.com/UKPLab/pythagoras` - Kindly keep in mind that the aim of publishing the experimental code is mainly to provide a reference for feature implementation details rather than to distribute a fully documented and tested piece of software.

knowledge, we are the first to use this type of data for an automated natural language analysis with this focus, especially in German.

The paper is organized as follows: Section 2 presents related research, mainly in the area of educational NLP, and Section 3 provides a description of the data set we used in general, while in Section 4 we describe and motivate the subset we used in the current study. In Section 5, we describe our text classification approach. Section 6 presents the results along with the suggested interpretation of our findings and their discussion. Section 7 concludes our work and addresses future research directions.

2 Related Work

As the task we present here has not yet been covered, there is little previous work in direct relation. Our work fits best into the growing field of *Educational NLP*. The series of workshops on *Innovative Uses of NLP for Building Educational Applications* give an overview over the current trends. Previously addressed tasks are largely varied, including, but not limited to, the assessment of student tests (spoken and written) (Cheng et al., 2014; Kharkwal and Muresan, 2014; Loukina et al., 2014, 2015; Farra et al., 2015; Napoles and Callison-Burch, 2015), computer-assisted translation (Ahrenberg and Tarvi, 2014) and vocabulary-constrained natural language generation (Swanson et al., 2014).

Group Cooperation as expressed in the interaction among the students is an important phenomenon in learning. Machine-learning techniques were used to automatically classify elements of group interaction in written German conversations (Rosé et al., 2008), based on manual annotation of the available data. The authors gathered the data using a chat system. Students who participated in the study had to type their conversations. The resulting dataset contained 250 conversations and a total of 1,250 German text segments. It was annotated for argumentative knowledge construction based on an elaborate coding scheme. The authors then derived a variety of features such as punctuation, token uni- and bigrams, POS bigrams and presence of non-stop words and rare words. The results indicate that phenomena of group interaction can be reliably detected based on textual information. This is important for our work, as we also only rely on textual information (see Section 5.2). Gweon et al. (2009) automatically predict student activity levels in group meetings using only average student talk time and overlap. With these basic features, they can already reliably differentiate between the students taking lead in conversation and the ones back-channeling. Others, such as Kersey et al. (2009) focus on computationally measuring the shifts of initiative as a predictor of knowledge co-construction, a high-level concept explaining the effectiveness of peer learning (Hausmann et al., 2004). In one of the tasks, the authors found a significant correlation between the post-test score and the number of shifts in dialog initiative between speakers.

Tutor/Teacher-Student Interaction and Feedback has been studied extensively, as an important mechanism in teaching (see for example Hattie and Timperley (2007) and Hattie (2009)). Studies on the impact of feedback on mistake vs. feedback

when correct found that feedback types were not predictive of post-test results (Di Eugenio et al., 2005). Other studies (Chen et al., 2011) on the correlation between dialog acts and learning gain in informatics found "several dialog act sequences that significantly correlate with learning gain"(Chen et al., 2011, p.73). Examples are a prompt followed by an instruction and feedback. However, the effective dialogue act sequences varied per topic studied in the class, not being conclusive for a generic application. Additionally, it was observed that there is a "tendency of dialog partners to adjust various features of their speech to be more similar to one another" (Ward and Litman, 2007, p.57). The authors hypothesize that the convergence towards the tutor might be associated with learning, and show that lexical overlap in consecutive utterances can discriminate well between a tutoring dialog and randomly ordered text, which we translated into features to use in our work (see Section 5.2).

Student Emotion Detection has been the focus of multiple studies aiming at automatic emotion classification under various conditions. Researchers studied the emotion of students in human-human tutoring dialogues as opposed to human-computer ones (Litman and Forbes-Riley, 2006). The authors compare results on positive, negative and neutral utterances both based on lexical and surface features from transcripts and on acoustic-prosodic features. Their findings indicate that, based on the transcripts alone, it is possible to achieve comparable emotion prediction results to using transcripts *and* recordings. This is important for our research, as the speaker emotion detection plays a role in several lesson quality dimensions we study. Other researchers demonstrated that student uncertainty negatively correlates with learning success (Forbes-Riley and Litman, 2011). In another work, the authors additionally found (Forbes-Riley et al., 2012) that student disengagement negatively affects learning success. Reasons for disengagement were found to be: Presenting a problem for too long and presenting a too hard problem. Additionally, a short time interval between the question of an automated tutoring agent and the answer of a student was a strong predictor for student disengagement.

The following two sections (3 and 4) present the data and the ratings created in the study by educational researchers. A subset of these is then used in our NLP analysis as described from section 5 onwards.

3 The Pythagoras Data set

The data used in this paper originates from a bi-national (Germany and Switzerland) study presented by Lipowsky et al. (2009), in which 40 classes of 8^{th} (Germany) and 9^{th} (Switzerland) grade students were video-taped during five of their mathematics lessons. During three of the lessons each class was introduced to the Pythagorean Theorem (*Theory*) and during two lessons each class dealt with mathematical problem solving (*ProbSol*) in general (i.e. not related to the Pythagorean Theorem). The whole study thus contains 200 videos, each lesson being 40-50 minutes long.

3.1 Manual Transcription

The videographic studies are very sensitive to privacy issues, therefore the recordings are available only with a well-founded request and under tight restrictions. To facilitate the studies and the accessibility of the material, 193 videos were manually transcribed and anonymized. These anonymized transcripts are now available to research communities of various fields. Table 1 shows a snippet of the transcript.

The transcripts include elements such as laughter, coughing and door slams. Pauses were not marked specifically, beyond using "..." for short pauses and splitting a segment[3] into two segments for the same speaker if he/she paused longer. Dialectal elements were translated to Standard German and the utterances were anonymized (e.g. Schueler #F).

The recordings were carried out in the actual classrooms, using only one head-mounted microphone for the teacher and a microphone attached to the camera. Therefore, the recording quality is occasionally quite low, which also accounts for inaudible passages. Additionally, the transcribers were asked to not transcribe a range of phenomena such as hesitations, in order to keep the transcription efforts low. Only the beginning time of each utterance was given, therefore any rhetoric pauses cannot be determined.

In addition to the transcripts, manual summaries for each lesson and fine-grained lesson segment annotations of the situations observed in the classroom were produced and made available. These represent in detail, for example, whether homework is being discussed or a proof is being shown.

Time	Speaker	Dialog
00:12:41:01	S	Wenn es um den Pythagoras geht, dann ist ja klar, dass das ()! *If this is about Pythagoras, then it is obvious that ()!*
00:12:49:28	SN	Ja, doch! *Yes, of course!*
00:12:51:00	T	Klar, SCHUELER#F., wie lautet der denn? *Sure, student#F., what is it then?*

Table 1: Snapshot of a part of a transcript. Time stamps and speakers (Teacher (T), Student (S) or New Student (SN)) are marked. The transcribed parentheses () in the first utterance indicate that part of the conversation was not audible to the transcriber and was therefore, not transcribed. Anonymized student names are indicated by "SCHUELER#F". The translations to English below each segment are our own.

3.2 Lesson Quality Assessment

In order to rate the interaction between teachers and students and among students, an annotation scheme was developed by Rakoczy and Pauli (2006). In this scheme each aspect of the interaction (we further refer to these aspects as 'dimensions') is described as a basic idea and a list of indicators such as "Students do not mock each other". For each of the dimensions defined, each lesson was rated on a 4-point Likert scale by

[3] A *segment* in our wording is a stretch of speech uttered uninterrupted by a single speaker, as shown in Table 1. Other words used in the literature for this phenomenon are *turns* or *utterances*.

2-3 expert annotators. The average score of all annotators is taken as a final lesson score in each of the dimensions. These scores are also called "high-inference ratings" in educational research literature. The annotators were encouraged to consider: frequency or duration of the specific behavior during a lesson, intensity of this behavior and distribution across students. The general impression was based on all three parameters. For all dimensions, the inter-rater reliability was assessed to ensure the quality of the annotations. The inter-rater agreement was measured using the generalization coefficient, as detailed in Clausen et al. (2003). This coefficient "expresses the relative amount of true variance in the variance observed." (Lipowsky et al., 2009, p. 532), hence accounting for chance agreement. The reliability threshold recommended by educational researchers is 0.66.

Based on these quality ratings, educational researchers performed further analysis. For example, Rakoczy (2006) looked into the correlation between classroom conditions and motivational and cognitive development of the students, but found no significant correlation. Lipowsky et al. (2009) analyzed the student performance based on a post-test. Their findings include, but are not limited to, that disturbances in the classroom correlate with the performance, whereas the feedback type does not affect it. In our work we use a reliable selection of these ratings, as explained below.

4 Dimensions Analyzed in Our Study

The lesson transcripts and the high-inference ratings presented above are the basis for our computational study. Out of the 193 only 187 transcripts (115 Theory and 72 ProbSol) could be used. The data in the remaining transcripts was corrupted beyond repair, i.e., the transcription did not follow the recommended format and could not be correctly segmented by any automated methods.

Our final data set thus contains 78,242 transcribed conversation segments from a total of 140 hours of recordings, as detailed in Table 3. Each conversation segment, further referred to as "utterance", is a set of sentences which has its own speaker and time annotation, as previously shown in Table 1. Further analysis of the data set, such as rating correlations, is provided in Section 5, in order to facilitate direct comparison with the results.

As the transcribers for the manual transcriptions also transcribed unspoken elements, such as laughter (see Section 3.1 above), we could take these phenomena into account.

But, they also had to translate dialectal elements (for example from the Swiss German dialect) to Standard German. Hence, we could not take into account the effect of dialectal phenomena on the lesson quality.

All 28 lesson quality dimensions were rated on a four-point Likert scale (see the annotation guidelines of Rakoczy and Pauli (2006) for details). Educational researchers have averaged the annotation values from raters per lesson. These averages serve as the gold-standard values for our evaluation. For the purpose of achieving reliability in this study, we selected only those dimensions out of all available ones, that had the Generalization Coefficient value (for details see Section 3.2 above) of inter-rater

agreement > 65%, as calculated by Rakoczy and Pauli (2006) (see Table 2). As the two different lesson types (ProbSol and Theory) had two different sets and numbers of annotators, they are listed separately here.

Given the relatively small data set, we have approached the problem as a binary classification task and have divided the lessons into high- and low-rated lessons. A score of [1.0-2.0] indicates a low rating, while a score of [3.0-4.0] indicates a high rating. Lessons with an average score between 2.0 and 3.0 were ignored for a given quality dimension.

Based on the quality criteria stated above and after determining highly correlated dimensions, we selected six dimensions with sufficiently wide rating distribution (i.e. more than 35 lessons in each of the rating intervals 1-2 and 3-4) to conduct our machine learning experiments. Most of these dimensions relate to how the teacher treats the students, as explained below in this section.

Dimension	ProbSol	Theory
Relevance of lesson content (RELEV)	.83	.89
Recognition of the student (RECOG)	.85	.74
Objective and constructive feedback (FEED)	.89	.70
Learning community (LEARN)	.84	.70
Cooperation (COOP)	.99	.80
Exploration ofhinking processes (THINK)	.95	.65

Table 2: Values for Inter-Annotator Agreement using the Generalization Coefficient as calculated by Rakoczy and Pauli (2006) for the dimensions we analyzed.

Figure 1 shows the dimensions this work focuses on, along with the number of high and low rated lessons they each contain. Note that the majority class is different for individual dimensions (high for FEED and low for THINK) and some dimensions show stronger imbalance than others (THINK vs. COOP). Table 3 displays basic length-based statistics of the data in the dimensions used.

No. of	$\sum$(T)	$\sum$(S)	$\sum$(all)	avg(T)	avg(S)	avg(all)
Utterances	45 546	32 696	78 242	243.56	174.84	209.20
Sentences	73 509	27 380	100 889	393.10	146.42	269.76
Tokens	705 467	253 213	958 680	3772.55	1354.08	2563.22

Table 3: Dataset statistics in terms of total and average numbers of teacher(T), student(S) and combined(all) utterances, sentences and tokens in the data.

Below we give a brief description of each dimension used in a study. A more detailed description of these, the remaining dimensions and underlying scientific motivation can be found in the original annotation guidelines by Rakoczy and Pauli (2006).

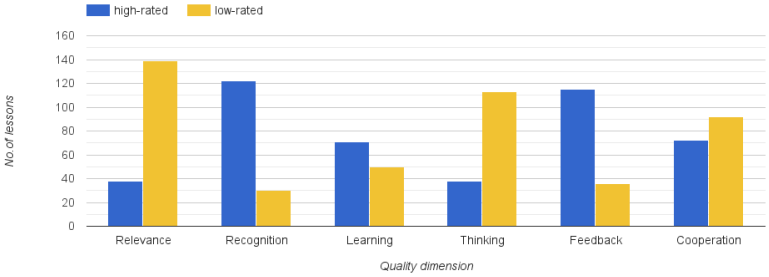


Figure 1: Distribution of lessons in the dimensions under analysis.

4.1 Objective and constructive Feedback (Feed)

This dimension rates the amount and quality of feedback, for example the teacher should be benevolent, provide guidance through the improvement path and show no sarcasm. Constructive feedback shall motivate students and improve their methodology by correcting the student but not discouraging him to try again. Objective means that the feedback only focuses on the topic and not on the person.

4.2 Exploration of thinking processes of students (Think)

This dimension rates the aptitude of the teacher to request detailed explanations. The teacher shall actively encourage students to justify their answers. This allows for an easier understanding of the material by the students. Additionally, this dimension rates whether the teacher attempts to learn the background of student's answers. Suggested indicators for this dimension are *why-* and *how-* questions. The teacher encourages the students to explain phenomena in his/her own words rather than repeating what the teacher has said.

4.3 Cooperation (Coop)

This dimension relates to how well the students support each other during work in smaller groups. The teacher shall show appreciation for team work, students shall appear accustomed to work together.

4.4 Relevance of lesson content (Relev)

This dimension rates how much the teacher tries to present the students the relevance of what he/she is teaching. Suggested indicators for this dimension are examples taken directly or indirectly from every day life of the students, they are allowed to work with every day items or the phenomenon relates to something the students know from their

every day life. But the historical context of what is being presented should not be forgotten. The quality of the examples chosen by the teacher for this purpose is more important than the quantity.

4.5 Recognition of the student (Recog)

Ratings in this dimension show how respectfully a teacher treats the students, whether he/she takes them seriously or mocks them using cynicism and sarcasm. This is especially obvious when students give wrong answers. Suggested indicators for this dimension are the confidence and the interest which the teacher shows towards the students, when they describe their perspectives and opinions. Presence of stinging jokes results in a lower score.

4.6 Learning Community (Learn)

Here, the working atmosphere in the class is rated, which is supposed to be supportive. The teacher is not supposed to take the role of a superior, but is part of the group, which tries to achieve a learning goal. Suggested indicators for this include the teacher using the first person plural ("wir" in German), teacher and students listening to each other and interacting without mocking each other.

5 Experimental Setup

We first build one classification model for each of the quality dimensions separately, i.e. for feedback quality, cooperation support etc., and later combine these into a global model. Since our goal is to identify features particularly characteristic for each dimension, we assume that a model which successfully learns useful information from the data shall perform better on predicting ratings for the dimension in question than on predicting the other ratings. To verify this assumption, we perform cross-dimensional tests for each model, i.e. examining how an all-features classification model trained on one dimension performs on another dimension.

Due to the small data set size, we use a Leave One Out Cross-Validation approach (LOOCV). In order to prevent learning phenomena specific to a certain teacher-class-combination, we modify the LOOCV-approach by excluding lessons of the same teacher-class-combination from the training set. We hereafter call this approach Leave One *Classroom* Out Cross-Validation (LOCOCV). We compare our results to the majority class baseline (Table 1) with respect to the accuracy.

Relations between the dimensions are displayed in Table 4. The first value of each cell shows the Pearson's correlation for the ratings. There is a strong correlation between the dimensions Feedback quality and Learning environment, Thinking processes and Receptive understanding, and Feedback quality and Receptive understanding.

The second value in each cell shows the percentage overlap of lessons used for each dimension, based on the average rating score in the dimension. It shows that for example

about 46% of the data from the Feedback dimension is shared with the dimension Cooperation.

	Feed	Coop	Think	Relev	Learn	RecUnd
Feed	1 / 100%	.34 / 46%	.42 / 40%	.22 / 41%	.66 / 54%	.61 / 43%
Coop	.34 / 43%	1 / 100%	.29 / 54%	.09 / 55%	.39 / 42%	.20 / 35%
Think	.42 / 40%	.29 / 58%	1 / 100%	.08 / 62%	.48 / 50%	.77 28%
Relev	.22 / 35%	.09 / 51%	.08 / 53%	1 / 100%	.35 / 38%	.21 / 35%
Learn	.66 / 67%	.39 / 57%	.48 / 62%	.35 / 56%	1 / 100%	.36 / 37%
RecUnd	.61 / 40%	.20 / 35%	.77 / 26%	.21 / 38%	.36 / 28%	1 / 100%

Table 4: First value in the table represents the Pearson's pair correlation coefficient for the dimension ratings. Second value shows the percentage share of lessons from one dimensions of experimental data (row) in the experimental data of another dimension (column), i.e. overlap of identical lessons with rating 1.0-2.0, resp. 3.0-4.0 between dimensions.

5.1 System Architecture

Our experimental setup is based on the Darmstadt Knowledge Processing (DKPro) (Eckart de Castilho and Gurevych, 2014) software repository, an open-source Natural Language Processing (NLP) toolkit building upon the Unstructured Information Management Architecture (UIMA) (Ferrucci and Lally, 2004). We publish the Java code used in this experiment as open-source on our website (see Section 1).

We preprocess the data using the TreeTagger (Schmid, 1994) components for German lemmatization, Part of Speech (POS) and chunk tagging. Our machine learning configuration is based on the Waikato Environment for Knowledge Analysis (WEKA) toolkit (Witten and Frank, 2005) and consists of a Support Vector Machine (SVM-SMO) classifier with polynomial (quadratic) kernel and default parameters, wrapped in a cost sensitive meta classifier with error costs empirically set to account for the class imbalances.

We examine the contribution of individual features, described in Section 5.2, using the ablative analysis, the Information Gain scores and the WEKA visual analytics capabilities, and summarize our findings in Section 6.

5.2 Features Used

Our features are clustered into seven groups, described below. Details, including references and examples, are provided in the individual sections.

We empirically selected the strategy of normalizing count-based features per utterance (as opposed to normalization per time or sentence), averaged per lesson.

For every feature, we additionally measure values for teacher and student utterances in the lessons individually, as well as the ratios between a feature value for the teacher and the student.

5.2.1 Ngram features (Ng)

This group of features consists of the 500 most frequent word uni-, bi- and trigrams from each fold of the training set after stopword cleaning[4]. We also derived phrase triplets based on constituency parsing, i.e., the 500 most frequent word trigrams extracted from the NP-VP-NP triples in an utterance (Levy et al., 2013), as these appeared to represent the content of short questions and answers well in automated question-answering tasks.

5.2.2 Surface features (Su)

These features are commonly used in text classification tasks. We measure text length ratios, expressed in terms of length of a word, sentence or utterance, information-based values such as the average $tf * idf$ score per utterance, and the type-token ratio. Additionally, this group includes surface features based on meta data taken from the transcripts, such as the time taken per utterance and per speaker, and the number of speaker changes. We hypothesize that longer monologues of the teacher might lead to student disengagement (Forbes-Riley et al., 2012), thus lower rating on LEARN. Unfortunately, the transcriptions did not allow us to capture pauses between speakers, which were helpful for Forbes-Riley et al. (2012).

5.2.3 Stylistic features (Sty)

These features capture aspects such as the level of contextuality (CF) in the wording of an utterance, measured based on POS tags used (Heylighen and Dewaele, 2002) :

CF = (noun frequency + adjective freq. + preposition freq. + article freq. - pronoun freq. - verb freq. - adverb freq. - interjection freq. + 100) * 0.5

We expect it to influence student engagement, i.e., our hypothesis is that more casually speaking teachers could be more engaging, and that an increased usage of pronouns, interjections etc. at the expense of nouns can serve as a proxy to capture this casual way of speaking (Heylighen and Dewaele, 2002). We measure also the usage of modals and conditionals, which we assume to indicate student uncertainty (Forbes-Riley and Litman, 2011), and the proportion of each Part-of-Speech tag to other ones in teacher's and students' utterances. We excluded syntactic features based on dependency parsing and Part-of-Speech bi- and trigrams, as they negatively impacted our classification results on the development data. We hypothesize that this is due to spoken language transcription, which has specific properties leading to low parsing accuracy (see for example work by Bechet et al. (2014) for experiments on parsing spontaneous speech).

5.2.4 Sentiment and other lexicon-based features (Se)

These features are mainly based on the Linguistic Inquiry and Word Count (LIWC) utility (Pennebaker et al., 2001) in its German adaptation (Wolf et al., 2008). The

[4] `http://snowball.tartarus.org/algorithms/german/stop.txt`

88 word lists in LIWC contain valuable information not only on emotion (e.g. words expressing anger, sadness or fear), but also social processes (e.g. friends, family, communication) and cognitive processes (e.g. certainty, insight or discrepancy), validated by expert judges. LIWC additionally counts several syntactic aspects, e.g. pronoun type or verb tense.

We hypothesize that teachers using harsh words score consistently lower in the quality dimensions, in comparison to teachers using a lot of encouraging words. Therefore, we employ also the SentimentWortschatz lexicon for an additional, more fine-grained sentiment polarity measure of the lessons, on top of LIWC. SentimentWortschatz (Remus et al., 2010) lists polarity bearing German words weighted within an interval of [-1; 1] plus their Part-of-Speech tag, and if applicable, their inflections. In addition, we also use amplifying and down-toning intensifiers (such as *very*) translated by a German native speaker from their English form (Taboada et al., 2011). We also add polarity changers, such as *not* (Steinberger et al., 2012), and compute alternative sentiment polarity measures with negations and intensifiers included, in the same way as the authors of these intensifiers do (Steinberger et al., 2012; Taboada et al., 2011). However, we do not observe any difference in performance compared to using the plain lexicons without the polarity changers and intensifiers.

Next, we add three features based on grammatical mood (Eisenberg et al., 1998). A grammatical mood of verbs (subjunctive, imperative, indicative) allows speakers to express their attitude to the statement (for example desire, doubt or command). We calculate the proportions of these utterances in the lessons based on syntactic annotations of verbs, combined with the sentiment polarity of the utterance, i.e., counting for example negative and positive imperative as two separate features.

Finally, we monitor the occurrence of several custom-defined words indicating politeness, such as *Danke* (Thank you) and *Bitte* (Please).

A brief overview of lexicons used in this section is provided in Table‘5

Resource	Example
German LIWC	*Anger: hate, kill, annoyed, Insight: think, know, consider*
SentimentWortschatz	*harmonic: +0.5243 ADJ, crisis: 0.3631 NN*
168 intensifiers	*very, slightly, somewhat, extraordinarily*
15 polarity changers	*not, none, any*
grammatical mood	doubt vs. command
politeness words	*thank you, please*

Table 5: Lexicon-based features coming from various German and English resources, of which the latter have been translated for our purpose.

5.2.5 Features based on discourse indicators (Di)

These features capture the discourse-relevant information. We model the Boolean presence and the normalized count of individual discourse markers in the utterances as features. To detect these, we utilize two lexicons of discourse markers - the German

DimLEx (Stede and Umbach, 1998) and the 15 most frequent discourse markers of the Penn Discourse Treebank (PDTB) (Prasad et al., 2008) translated by a German native speaker. Discourse markers are lexical items, annotated in their lexicon with a particular discourse relation they tentatively express, such as `Cause`, `Reason` or `Opposition`. Each of the 173 Discourse Markers are grouped according to their discourse relations and are one word-count-based features in our model. We also count discourse marker bigrams, i.e., the occurrence of pairs of two consecutive discourse relations appearing in the same utterance.

Additionally, we calculate the ratio of nouns repeated by one speaker from the previous speaker in proportion to the total number of nouns used by the speaker in that utterance, assuming that higher overlap demonstrates better understanding, based on the work by Ward and Litman (2007) on lexical convergence. This number is then averaged per lesson, i.e. how many nouns on average do the students repeat after previous speaker and how many nouns on average does the teacher repeat after the previous speaker.

We also measure the frequency and the pattern type of speaker changes as an indicator of student initiative turns, demonstrated as predictors of the knowledge co-construction (Kersey et al., 2009). The pattern is defined as a bigram of speaker annotations (column Speaker in Table 1), for example S-T, T-S or S-SN (student followed by new student). We further monitor individual transcribed non-verbal expressions and attentive back-channeling. These include sighs, laughter, but also so-called social noise, such as *mhm*. Back-channeling is a way of showing a speaker that you follow and understand their contribution, often through interjections, such as "I see" or "ok" (Gweon et al., 2009).

5.2.6 Topic Features

We hypothesize that due to certain expressions (e.g. expert mathematical terms) students may perceive the content as harder, causing disengagement (Forbes-Riley et al., 2012) and impacting the rating of the lesson. Hence, we crafted custom word lists to measure the frequency of terms related to topics discussed in the lessons. Among others, these were on *mathematics*, *school* and the *Pythagorean theorem*. A separate word list was used to capture expressions unrelated to the lesson content and possibly indicating creativity of the teacher – this contained words such as *princess, castle, river* and so on. Additionally, we built topic models on the teacher's and students' speech in high and low rated lessons using Mallet topic modelling tools (McCallum, 2002) empirically set to 50 topics. See Table 6 for an example of topics used in this feature group.

5.2.7 Phonetic (Ph)

Phonetic features have been used in text processing areas such as machine translation (Vogel et al., 2003) or normalization (Han et al., 2012), however they remain unexplored for more abstract tasks, such as the prediction of lesson quality. Our intuition behind this group of features is that certain phoneme combinations may be difficult to understand or certain phoneme occurrences may point to the sentiment of a speaker (Nastase et al.,

Topic	Example
mathematics	*comma, squared, surface*
school	*homework, pupil, lesson*
pythagorean theorem	*triangle, right angle, theorem*
storytelling	*farmer, river, house, tree*
Mallet topic 1:	*five, two, fourteen, seven*
Mallet topic 2:	*mhm, ah, aha, hm, okay, oh*

Table 6: Topical features based on the transcripts and the discussed topics.

2007). We phonetize the transcripts using a German text-to-speech tool (Schröder and Trouvain, 2003) and analyze the frequency of each type of phonemes (e.g. plosives, fricatives, glottal stops, etc. - see Table 7 for examples).

Phonetic	Example (in SAMPA notation)
frequency of plosives	p, t, k, ...
frequency of fricatives	f, S, Z, x, ...
frequency of vowels	I, E, a, Y, ...
...	

Table 7: Phonetic Features in SAMPA notation, taken from the automatically phonetized transcripts.

6 Results

In the following, we present our findings for the six dimensions examined (Objective and constructive feedback, Exploration of thinking processes, Cooperation, Relevance, Recognition of a student, Learning community) as well as the analysis of their relations both on the gold-standard and classification-model levels. In order to illustrate the usefulness of the presented methods in supporting the analysis of classroom interaction through educational researchers, we discuss in detail the results on the three dimensions with the highly informative features beyond lesson ngrams. These three dimensions are: Exploration of thinking processes, Objective and constructive feedback and Cooperation. Other dimensions were omitted since their generalizable features are related to the ones in selection (e.g. Recognition of a student shows similarities to constructive feedback) and their unique features are of less interest (e.g. the Relevance dimension is specific to the lesson content).

6.1 Summary of classification results

Table 8 shows the comparison of the outcome of the classification system (Support Vector Machines, detailed in 4.1) to human annotators using percentage agreement (SysAA). Our system performs comparably to a human annotator on every dimension, suggesting, that these highly abstract tasks include computationally measurable clues.

	Theory			Problem Solving			Combined	
Dim	IAA	SysAA	Acc	IAA	SysAA	Acc	BaseF_1	BestF_1
Think	.66	.84 (.80-.87)	.92	.95	.73 (.72-.75)	.78	.647	.855
Relev	.87	.86 (.85-.88)	.87	.99	.75 (.74-.75)	.89	.691	.873
Recog	.70	.71 (.64-.77)	.72	.95	.84 (.67-.87)	.73	.505	.716
Feed	.69	.79 (.63-.89)	.88	.83	.90 (.90-.90)	.90	.609	.883
Learn	.65	.82 (.79-.86)	.83	.78	.88 (.88-.88)	.88	.434	.851
Coop	.77	.82 (.79-.84)	.87	.99	.83 (.83-.83)	.86	.403	.866

Table 8: Results comparing our system to human performance.
IAA: percentage agreement on Theory and Problem Solving within the three and two human annotators respectively.
SysAA: percentage agreement when considering the system as an additional annotator (in brackets is the minimal and maximal pair agreement with individual annotators).
Acc: percentage agreement between the system results and the average human annotator rating (taken as our Gold standard).
F_1: Results for the majority baseline and the best F_1 score achieved through the machine learning setup for each studied dimension.

Our best results for binary classification of high- and low-rated lessons (see F_1 in column Combined in Table 8) differ from the weighted majority baseline (BaseF_1) significantly ($p < 0.05$) in all dimensions. Statistical significance of differences was computed using an approximate randomization approach (Noreen, 1989). For human annotators (IAA), the Problem Solving lessons were not as challenging to rate as the Theory lessons, possibly due to a more straightforward student-teacher interaction, more explicit feedback etc. For the system (Acc), this issue does not arise – performance on both lesson content types is comparable.

6.2 Global Model and comparison between dimensions

Using the ratings in the six quality dimensions above, we also trained a global model. We selected only lessons rated high [3-4] in at least 3 dimensions and low in at most two (70 lessons), resp. rated low [1-2] in at least 3 dimensions and high in at most two (61 lessons). We then perform binary classification to discriminate such overall high-rated from the overall low-rated lessons and examine feature rankings.

Distribution of individual dimensions in the global model is very similar to the one described in Figure 1. The most frequently appearing high ratings in the overall high-rated lessons are in the dimensions Feed and Learn, while the low-rated lessons are most frequently rated low for the dimensions Think and Relev.

We achieve the best global result ($F_1 = 0.885$) using the combination of discourse, surface and phonetic features and topics. Discourse features are the most predictive ones based on an ablation test. In the detailed analysis, we found that in the global model high rated lessons were characterized by:

- increased reasoning and elaboration discourse markers from the student side (as in LEARN, COOP and THINK)

Feature		Dimension					
Ngrams, topics (Ng/To)	Relev	Recog	Feed	Learn	Coop	Think	Global
wie, warum				•			
besprechen, resultat						•	
mhm, ja, hm, genau		•	•				
Storytelling: felder, wasser...	•						
Number topic: komma, null...	•						•
Surface (Su)	Relev	Recog	Feed	Learn	Coop	Think	Global
(S)No.of dialog turns				•			•
Talk ratio T/S				•			•
(S)Answers <5 words		•		•	•	•	•
(S)Answers >25 words		•		•	•		•
Syntax (Sy)	Relev	Recog	Feed	Learn	Coop	Think	Global
(T)Interjection ratio		•	•				
(S)Verb ratio		•	•		•		
(S)Adverb ratio		•	•				
(S)Pronoun ratio		•	•		•		
(S)Conjunction ratio			•				
LIWC & Sentiment (Se)	Relev	Recog	Feed	Learn	Coop	Think	Global
(T,S)Positive		•	•				
(T)Communication					•	•	
(T,S)Cognitive				•	•		
(T,S)Cause				•		•	
(S)Question words		•	•			•	
(S)We, Self					•		
(T,S)Assent					•		
(T)You					•		
(S)Discrepancy		•					
(S)Negate		•					
Discourse (Di)	Relev	Recog	Feed	Learn	Coop	Think	Global
(T)Attentive signs (*ja*,nods)		•	•	•			•
(S)Attentive signs			•				•
(S)Cause				•	•	•	•
(S)Reason				•		•	
(T)Reason	•	•				•	
(S)Compare	•	•		•		•	•
(T,S)Elaboration	•			•			•
(T)Circumstance	•						
Dialog sequence S-S		•			•		
Phonetic (Ph)	Relev	Recog	Feed	Learn	Coop	Think	Global
(T)Syllables per word		•	•		•		

Table 9: Detailed presentation of features with the highest information gain (top 20) for the best classification model in each dimension. Teacher (T) or Student (S) indicates features measured for the respective utterances only.

- more back-channels from both sides (similarly to FEED and LEARN)
- longer student utterances (similarly to COOP and THINK)
- relatively more student utterances in comparison to the teacher (similarly to LEARN)

To validate that our models learn aspects relevant to the dimension in question, we measured the performance of classifiers trained on one dimension (row in Table 10)

and tested on another (column in Table 10). Expectably, results for dimensions with a higher rate of correlation and data overlap (see Table 4) are better than for dimensions with lower rate of correlation and data overlap, however, all cross-dimensional train-test results (see Table 10) were still significantly different (worse) than a cross-validation on each single dimension itself (see Table 8). This suggests that each of our three models learned features predominantly relevant for its dimension of interest.

	Cross Relev	Recog	Feed	Learn	Coop	Think
Relevance	1	.36	.38	**.62**	**.61**	**.67**
Recognition	.40	1	**.92**	**.69**	.46	.37
Feedback	.44	**.87**	1	**.68**	.49	.42
Learning	.59	**.69**	**.74**	1	**.62**	**.64**
Cooperation	.57	.47	.51	.60	1	**.62**
Thinking	**.65**	.37	.43	**.69**	**.65**	1

Table 10: Percentage accuracy for the best classification model from each dimension (row) tested on another dimension (column). Accuracy is chosen here over F-score for better comparison with the correlation and data overlap values presented in Table 4. Accuracy scores $\geq$0.6 are highlighted.

Across all dimensions, students in high rated lessons are given a chance to express themselves in more elaborated and argumentative manner, while teachers extensively demonstrate their attention and stimulate the communication. Already the simple discourse features and word categories related to sentiment and cognitive processes show high predictive power in our task, which is a promising path for an automated assistance in qualitative evaluation of teaching.

6.3 Detailed feature analysis on selected dimensions

In previous work, Rosé et al. (2008) (see Section 2) analyzed the possibility to employ NLP features for the task of classification of cooperation in learning environments. The authors note that in the future it would be beneficial to evaluate "which features provide the greatest predictive power for the classification" (Rosé et al., 2008, p.266). To explore this question, we discuss below in detail the results on the three dimensions with the highest potential of being generalized to other studies. These are: the Exploration of thinking processes, Objective and constructive feedback and Cooperation.

A detailed examination of ablative tests with different feature groups for each dimension revealed that features from different groups are in many cases mutually substitutive, indirectly representing the same phenomenon (see Table 9). For example, the length of sentences, captured in surface features is also apparent through a larger variety of POS tags and discourse markers present in the utterance. Similarly, back-channels are reflected in ngrams and word length etc. Therefore, we further examine the ranking of individual features based on information gain in order to understand the underlying phenomena. For each dimension, the features consistently scoring high across classification folds are listed in the following, together with our suggested interpretation.

6.3.1 Exploration of thinking processes of students (THINK)

Features	Ngram	Surface	Syntax	Sentiment	Discourse	Phonetic	All	Base
F-score	.809	.767	.678	.640	.855	.647	.779	.647

Table 11: Performance (F-score) of individual feature groups for the dimension *Exploration of thinking processes*. Baseline is a majority baseline weighed accordingly to the lesson proportion in the high and low class.

Table 11 reveals that especially the discourse features, ngrams and surface features were important for this dimension. Table 12 sheds additional light on this classification task by highlighting individual features with the highest information gain for this dimension. We found that high rated lessons are characterized through student features such as Reason and Cause (*weil (because), so dass (so that)*), question words (*wie (how), wo (where), woher (where from), warum (why)*) and long utterance (more than 25 words). On the teacher side these are characterized through features such as Comparison and Elaboration (*insbesondere (especially), das heisst (that means)*) and Communication (*sagen (say), fragen (ask), meinen (mean), beschreiben (describe)*).

Feature	Examples
For speaker: Teacher	
$\text{Communication}_{LIWC}$	*sagen (say), fragen (ask), meinen (mean), beschreiben (describe)*
Compare_{PDTB}	*insbesondere (especially)*
$\text{Elaboration}_{DiMLex}$	*das heisst (that means)*
For speaker: Student	
Cause_{PDTB}	*weil (because)*
Reason_{DiMLex}	*so dass (so that)*
Utterances >25 words	
Questions_{LIWC}	*wie (how), wo (where), woher (where from), warum (why)*

Table 12: Features predictive for high rating in THINK.

We conclude that these features approximate the behaviour observed by educational researchers: In highly rated lessons the teacher encourages the students to communicate and students ask more questions. Both students and teachers use more reasoning, especially students have longer utterances and compare and explain concepts.

6.3.2 Objective and constructive feedback (FEED)

In the ablative tests (Table 13), this dimension is best predicted by discourse-based, sentiment- and cognition-oriented lexicon-based features, and ngrams. The most informative individual features are listed in Table 14. Teachers use positive words and both teachers and students give back-channels, which is reflected also through ngrams, interjection frequency and phonetic features. Students use question words, but

Features	Ngram	Surface	Syntax	Sentiment	Discourse	Phonetic	All	Base
F-score	.872	.739	.735	.757	.831	.739	.883	.659

Table 13: Performance (F-score) of individual feature groups for the dimension *Objective and constructive feedback*. Baseline is a majority baseline weighed accordingly to the lesson proportion in the high and low class.

also words from the group Discrepancy and negation (e.g. *Das wollen wir aber nicht, oder? (We don't want this, do we?)*), Comparison and Specification (e.g. *oder (or) ... beispielsweise (for example)*). Similar to THINK, long sentences and a high variation in POS are very predictive for this dimension.

Feature	Examples
For speaker: Teacher	
Positive$_{SentiWS,LIWC}$	*gut (good), perfekt (perfect), wunderbar (wonderful)*
Interjection rate (high)	
For speaker: Student	
Questions$_{LIWC}$	*wie (how), wo (where), woher (where from), warum (why)*
Discrepancy$_{LIWC}$	*aber (but), hoffe (hope), sollte (should)*
Negation$_{LIWC}$	*nicht (not), keine (none)*
Comparison+ Specification$_{DiMLex}$	*oder (or)...beispielsweise (for example)*
Pronoun rate (high)	
Verb rate (high)	
For all speakers	
Back-channels	*hm, ja (yes), mhm, genau (exactly)*
One-syllable words	

Table 14: Features predictive for high rating in FEED.

Lessons rated **high** on FEED are also characterized by higher frequency of indicative verbs, demonstrative pronouns and subordinate conjunctions, hinting towards an increased number of complex factual sentences. Together with the higher question word ratio observed in students' turns, this suggests, that an environment with constructive feedback may encourage students to pursue the problems further and discuss them with the teacher. Lessons scored **low** in this dimension show on average lower frequencies of positive sentiment words and in several cases even some negative tone, e.g. *Das ist falsch* (That is wrong).

We hypothesize that these features are indicative of the behaviour observed by educational researchers: Both students and teachers actively listen to each other. The teacher encourages the students to proceed and the students express opinions and voice questions. Additionally, they do not hesitate to ask even when they are unsure.

Tentatively, an environment with constructive feedback appears to support students to pursue the problems with more confidence and discuss them with the teacher.

6.3.3 Cooperation (COOP)

Features	Ngram	Surface	Syntax	Sentiment	Discourse	Phonetic	All	Base
F-score	.662	.739	.735	.754	.712	.564	.725	.403

Table 15: Performance (F-score) of individual feature groups for the dimension *Cooperation*. Baseline is a majority baseline weighed accordingly to the lesson proportion in the high and low class.

This dimension benefits from the entire range of features. Among the best performing useful feature groups are the syntactic, surface, and sentiment and other lexicon-based features. Table 16 lists individual features with the highest information gain for this dimension.

The use of back-channels on the teacher side was also useful in FEED, whereas the use of communication words on the teacher side was useful in THINK. The other features that best predicted this dimension are: The specific speaker pattern of student speaker followed by a new student speaker (S-SN), indicating that students discuss things amongst themselves. This is supported by the use of *we* rather than *I* on the student side. The teacher uses *you* (German second person plural, referring to the group). Also, the students use words from Alternative and Comparison (*oder (or) ... obwohl (although)*, Alternative and Elaboration (*oder (or) ... beispielsweise (for example)*), Contrast and Elaboration (e.g. *anderseits (on the other hand) ... und (and)*) in their utterances. Also, the students use a range of cognition words, such as *erkennen (recognize), konstruieren (construct), wissen (know)*. Also, the students use long utterances.

In the lessons rated **low** for COOPERATION, students often use very short sentences, as in most of the previously discussed dimensions.

Feature	Examples
For speaker: Teacher	
Communication$_{LIWC}$	*sagen (say), fragen (ask), meinen (mean), beschreiben (describe)*
Back-channels	*hm, ja (yes), mhm, genau (exactly)*
Pronoun You$_{LIWC}$	
For speaker: Student	
Cognition$_{LIWC}$	*erkennen (recognize), konstruieren (construct), wissen (know)*
Alternative+Comparison$_{DiMLex}$	*oder (or) ... obwohl (although)*
Alternative+Elaboration$_{DiMLex}$	*oder (or) ... beispielsweise (for example)*
Contrast+Elaboration$_{DiMLex}$	*anderseits (on the other hand) ... und (and)*
Speaker pattern S-SN	student - new student
Pronoun We$_{LIWC}$	
Utterances >25 words	

Table 16: Features predictive for high rating in COOP.

These results capture the following aspects relevant for educational researchers: Students communicate among each other and perceive themselves as a team, using *we* rather than *I*. They speak more and make their own suggestions. The teacher encourages this behavior by showing attention, while letting the students provide the explanations.

6.4 Discussion

In general, it can be concluded that a **high**-rated lesson in our data set is characterized by the following: Firstly, increased **reasoning activity** of the students, visible through the discourse markers. Secondly, increased **engagement of the students**, apparent through length-based features. And thirdly, increased **encouragement and attention** from both sides, detected through specific interjections, adjectives and adverbs and through non-verbal signs such as nodding. Particularly, the reasoning activity analysis could benefit from an extended, focused NLP research on argumentation, which is beyond the scope of this paper.

In accordance with Kersey et al. (2009), we find speaker changes to be a good predictor of cooperation. Also, back-channeling appears important for indicating collaborative thinking in our data set, in line with Gweon et al. (2009) inter alia. Usage of plurals (*we* vs. *I*) and words indicating cognition and communication were also prevalent in high-rated lessons along multiple dimensions. Short student utterances in general predict a lesson of low quality, which may indicate disengagement as described by Forbes-Riley and Litman (2012).

Previous research in educational NLP (see Sections 2 and 3) indicated that feedback does not influence the performance in post tests. Despite this, our results suggest that it may influence the way students express themselves – lessons, where students obtain more and better feedback, correlate with lessons where students speak more often. However, the direction of the causation needs to be further studied.

Our study is of relatively small scale, partly due to the necessity of manual transcription of video texts. Future research could overcome these limitation by using automated speech recognition tools or focusing on multimodal features. Additionally, an extension to teaching domains other than mathematics would be welcome to verify the generalizability of our findings.

While the identification of specific misclassification sources on the document level is challenging in our data, due to the high level of abstraction in ratings, we identified two areas of improvement.

Lexical ambiguity Lexical resources such as LIWC are lacking additional annotations to support sense disambiguation. Hence, particular words were occasionally misrepresented, as they were often used in a sense irrelevant to the category. For example the word *(S|s)chau* ([the] show|looking at) was grouped in the LIWC category `TV` even though it never occurred in the lessons in relation to television. Also, the discourse markers are known to be highly ambiguous (Stede and Umbach, 1998), e.g. the word *while* can represent a contrast as well as temporal co-occurrence.

Loss of audio/video information Restricting ourselves to the analysis of text deprives our experiments of information from the acoustic and visual parts of the data, which was available to the expert raters. We hypothesize that the maximum attainable performance is lower than for a multimodal system. For example, sarcasm in speech, which was often present in our data, is more easily discernible through prosodic and facial gesture features (see for example Rakov and Rosenberg (2013)), and so is user disengagement (Forbes-Riley and Litman, 2012) or convergence between conversants (Ward and Litman, 2008). Furthermore, we found that several misclassified documents contained markers of dialog instances inaudible to the transcriptionists.

7 Conclusion and Future Work

Predicting the quality of classroom lessons and analyzing the interaction between teachers and students and among students is an educational research topic of great importance. In this paper, we present initial experiments on the task of assessing several quality dimensions of classroom interaction, employing an existing data set of this kind for the first time. We model this as a text classification task, demonstrating the high potential of automated quality prediction systems to assist educational researchers. We present a freely available data set of German classroom transcripts and expert ratings on quality dimensions such as constructive feedback, thinking process and cooperation.

We defined a broad range of features from diverse NLP areas, reflecting the analysis of the verbal behaviour of the teachers and students, such as discourse analysis, phonetics and emotion detection. We applied machine learning techniques to classify lessons according to the dimensions highly relevant for educational researchers.

We carefully examined the relation between each of the measured phenomena and the quality dimensions, and suggested an interpretation of the most remarkable findings. We successfully built classifiers comparable to human annotators on this data set.

Our findings on the relevance of various feature groups offer room for extension both on the NLP and the educational researchers' side. On the latter, it would be worthwhile to analyze the correlation between the students' performance and the features which possibly influence the quality of a lesson, e.g. the back-channeling of a teacher. In continuation of our collaboration, it would be interesting to examine the benefit for the educational researchers of using a semi-automatic approach based on this work in the annotation of future data sets.

We hypothesize that the maximum attainable performance of our approach is lower than for a multimodal system. For example, sarcasm in speech, which was often present in our data, is more easily discernible through prosodic and facial gesture features (see for example Rakov and Rosenberg (2013)), which require signal analysis both on the visual and the acoustical part of the data. Our approach can be applied to further data sets of similar kind [5]. Automatic speech recognition (ASR) could be used in order to see how stable our results are in light of noisy, ASR-output.

[5] such as `http://www.timssvideo.com/timss-video-study`

Acknowledgement

This work has been supported by the Volkswagen Foundation as part of the Lichtenberg Professorship Program under grant No. I/82806 and by the German Research Foundation under grant No. GU 798/14-1. The authors thank Prof. Klieme, Dr. Katrin Rakoczy and Petra Pinger from the German Institute for Educational Research for their support with the data and educational research questions.

References

Ahrenberg, L. and Tarvi, L. (2014). Translation Class Instruction as Collaboration in the Act of Translation. In *Proceedings of the Ninth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 34–42, Baltimore, Maryland.

Bechet, F., Nasr, A., and Favre, B. (2014). Adapting dependency parsing to spontaneous speech for open domain spoken language understanding. In *Proceedings of the 15th Annual Conference of the International Speech Communication Association (Interspeech), 14.–18. September 2014, Singapore*.

Chen, L., Di Eugenio, B., Fossati, D., Ohlsson, S., and Cosejo, D. (2011). Exploring Effective Dialogue Act Sequences in One-on-one Computer Science Tutoring Dialogues. In *Proceedings of the Sixth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 65–75, Portland, Oregon.

Cheng, J., Zhao D'Antilio, Y., Chen, X., and Bernstein, J. (2014). Automatic Assessment of the Speech of Young English Learners. In *Proceedings of the Ninth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 12–21, Baltimore, Maryland.

Clausen, M., Reusser, K., and Klieme, E. (2003). Unterrichtsqualität auf der Basis hoch-inferenter Unterrichtsbeurteilungen. Ein Vergleich zwischen Deutschland und der deutschsprachigen Schweiz. *Unterrichtswissenschaft*, 31(2):122–141.

Di Eugenio, B., Lu, X., Kershaw, T. C., Corrigan-Halpern, A., and Ohlsson, S. (2005). Positive and Negative Verbal Feedback for Intelligent Tutoring Systems. In *Proceedings of the 2005 Conference on Artificial Intelligence in Education: Supporting Learning Through Intelligent and Socially Informed Technology*, pages 798–800, Amsterdam, The Netherlands. IOS Press.

Eckart de Castilho, R. and Gurevych, I. (2014). A broad-coverage collection of portable nlp components for building shareable analysis pipelines. In Ide, N. and Grivolla, J., editors, *Proceedings of the Workshop on Open Infrastructures and Analysis Frameworks for HLT (OIAF4HLT) at COLING 2014*, pages 1–11, Dublin, Ireland. Association for Computational Linguistics and Dublin City University.

Eisenberg, P., Gelhaus, H., Wellmann, H., Henne, H., and Sitta, H. (1998). *Duden, Grammatik der deutschen Gegenwartssprache*, volume 4. Bibliographisches Institut & F. A. Brockhaus AG, Mannheim, 6 edition.

Farra, N., Somasundaran, S., and Burstein, J. (2015). Scoring persuasive essays using opinions and their targets. In *Proceedings of the Tenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 64–74, Denver, Colorado.

Ferrucci, D. and Lally, A. (2004). UIMA: An Architectural Approach to Unstructured Information Processing in the Corporate Research Environment. *Natural Language Engineering*, 10(3-4):327–348.

Forbes-Riley, K. and Litman, D. (2011). Benefits and challenges of real-time uncertainty detection and adaptation in a spoken dialogue computer tutor. *Speech Commununications*, 53(9-10):1115–1136.

Forbes-Riley, K. and Litman, D. (2012). Adapting to Multiple Affective States in Spoken Dialogue. In *Proceedings of the SIGdial 2012 Conference: The 13th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 217–226, Seoul, Korea.

Forbes-Riley, K., Litman, D., Friedberg, H., and Drummond, J. (2012). Intrinsic and extrinsic evaluation of an automatic user disengagement detector for an uncertainty-adaptive spoken dialogue system. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 91–102, Montréal, Canada.

Gweon, G., Kumar, R., and Rosé, C. P. (2009). Grasp: The group learning assessment platform. In *Proceedings of the 9th International Conference on Computer Supported Collaborative Learning – Volume 2*, CSCL'09, pages 186–188. International Society of the Learning Sciences.

Han, B., Cook, P., and Baldwin, T. (2012). Automatically Constructing a Normalisation Dictionary for Microblogs. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 421–432.

Hattie, J. and Timperley, H. (2007). The Power of Feedback. *Review of Educational Research*, 77(1):81–112.

Hattie, J. A. C. (2009). *Visible Learning: A synthesis of over 800 Meta-Analyses Relating to Achievement*. Routledge, New York, USA.

Hausmann, R. G. M., Chi, M. T. H., and Roy, M. (2004). Learning from collaborative problem solving: An analysis of three hypothesized mechanisms. *26th Annual Conference of the Cognitive Science society*, pages 547–552.

Heylighen, F. and Dewaele, J.-M. (2002). Variation in the Contextuality of Language: An Empirical Measure. *Foundations of Science*, 7(3):293–340.

Kersey, C., Di Eugenio, B., Jordan, P., and Katz, S. (2009). Knowledge co-construction and initiative in peer learning interactions. In *Proceedings of the 2009 Conference on Artificial Intelligence in Education: Building Learning Systems That Care: From Knowledge Representation to Affective Modelling*, pages 325–332, Amsterdam, The Netherlands. IOS Press.

Kharkwal, G. and Muresan, S. (2014). Surprisal as a Predictor of Essay Quality. In *Proceedings of the Ninth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 54–60, Baltimore, Maryland.

Levy, O., Zesch, T., Dagan, I., and Gurevych, I. (2013). UKP-BIU: Similarity and Entailment Metrics for Student Response Analysis. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 285–289, Atlanta, Georgia, USA.

Lipowsky, F., Rakozcy, K., Pauli, C., Drollinger-Vetter, B., Klieme, E., and Reusser, K. (2009). Quality of Geometry Instruction and its Short-Term Impact on Students' Understanding of the Pythagorean Theorem. *Learning and Instruction*, 19(6):527–537.

Litman, D. J. and Forbes-Riley, K. (2006). Recognizing Student Emotions and Attitudes on the Basis of Utterances in Spoken Tutoring Dialogues with both Human and Computer Tutors. *Speech Communication*, 48(5):559–590.

Loukina, A., Zechner, K., and Chen, L. (2014). Automatic Evaluation of Spoken Summaries: The Case of Language Assessment. In *Proceedings of the Ninth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 68–78, Baltimore, Maryland.

Loukina, A., Zechner, K., Chen, L., and Heilman, M. (2015). Feature selection for automated speech scoring. In *Proceedings of the Tenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 12–19, Denver, Colorado.

McCallum, A. K. (2002). MALLET: A Machine Learning for Language Toolkit.

Napoles, C. and Callison-Burch, C. (2015). Automatically scoring freshman writing: A preliminary investigation. In *Proceedings of the Tenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 254–263, Denver, Colorado.

Nastase, V., Sokolova, M., and Shirabad, J. S. (2007). Do happy words sound happy? A Study of the Relation between Form and Meaning for English Words Expressing Emotions. In *Proceedings of Recent Advances in Natural Language Processing (RANLP 2007),* Borovets, Bulgaria, pages 406–410.

Noreen, E. W. (1989). Computer intensive methods for hypothesis testing: An introduction.

Pennebaker, J. W., Francis, M. E., and Booth, R. J. (2001). *Linguistic Inquiry and Word Count.*

Prasad, R., Dinesh, N., Lee, A., Miltsakaki, E., Robaldo, L., Joshi, A. K., and Webber, B. L. (2008). The Penn Discourse TreeBank 2.0. In *Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC),* Marrakech, Morocco, 26 May – 1 June 2008.

Rakoczy, K. (2006). *Motivationsunterstützung im Mathematikunterricht – Unterricht aus der Perspektive von Lernenden und Beobachtern.* PhD thesis, Johann Wolfgang Goethe-Universität, Frankfurt, Germany.

Rakoczy, K. and Pauli, C. (2006). *Hoch inferentes Rating: Beurteilung der Qualität unterrichtlicher Prozesse*, chapter 13, pages 206–233. Number 15 in Materialien zur Bildungsforschung. Gesellschaft zur Förderung Pädagogischer Forschung (GFPF)/Deutsches Institut für Internationale Pädagogische Forschung (DIPF).

Rakov, R. and Rosenberg, A. (2013). "Sure, I Did The Right Thing": A System for Sarcasm Detection in Speech. In *Proceedings of the 14th Annual Conference of the International Speech Communcation Association,* Lyon, France, 25–29 August 2013.

Remus, R., Quasthoff, U., and Heyer, G. (2010). SentiWS – a Publicly Available German-language Resource for Sentiment Analysis. In *Proceedings of the 7th International Language Resources and Evaluation (LREC'10)*, pages 1168–1171.

Rosé, C. P., Wang, Y.-C., Arguello, J., Stegmann, K., Weinberger, A., and Fischer, F. (2008). Analyzing Collaborative Learning Processes Automatically: Exploiting the Advances of Computational Linguistics in Computer-Supported Collaborative Learning. *Computer Supported Collaborative Learning*, 3(3):237–271.

Schmid, H. (1994). Probabilistic Part-of-Speech Tagging Using Decision Trees. In *Proceedings of the International Conference on New Methods in Language Processing*, Manchester, UK.

Schröder, M. and Trouvain, J. (2003). The German Text-to-Speech Synthesis System MARY: A Tool for Research, Development and Teaching. *Journal of Speech Technology*, 6:365–377.

Stede, M. and Umbach, C. (1998). DiMLex: A Lexicon of Discourse Markers for Text Generation and Understanding. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics – Volume 2*, pages 1238–1242, Stroudsburg, PA, USA.

Steinberger, J., Ebrahim, M., Ehrmann, M., Hurriyetoglu, A., Kabadjov, M., Lenkova, P., Steinberger, R., Tanev, H., Vázquez, S., and Zavarella, V. (2012). Creating Sentiment Dictionaries via Triangulation. *Decision Support Systems*, 53(4):689–694.

Swanson, B., Yamangil, E., and Charniak, E. (2014). Natural Language Generation with Vocabulary Constraints. In *Proceedings of the Ninth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 124–133, Baltimore, Maryland.

Taboada, M., Brooke, J., Tofiloski, M., Voll, K., and Stede, M. (2011). Lexicon-based Methods for Sentiment Analysis. *Computational Linguistics*, 37(2):267–307.

Vogel, S., Zhang, Y., Huang, F., Tribble, A., Venugopal, A., Zhao, B., and Waibel, A. (2003). The CMU Statistical Machine Translation System. In *Proceedings of MT Summit IX,* New Orleans, USA, volume 9, pages 54–63.

Ward, A. and Litman, D. (2007). Automatically measuring lexical and acoustic/prosodic convergence in tutorial dialog corpora. In *Proceedings of the SLaTE Workshop on Speech and Language Technology in Education,* Farmington, PA, USA, October 1-3, 2007.

Ward, A. and Litman, D. (2008). Semantic Cohesion and Learning. In *Proceedings of the 9th International Conference on Intelligent Tutoring Systems, ITS* Montreal, Canada, June 23-27, pages 459–469.

Witten, I. H. and Frank, E. (2005). *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann, 2 edition.

Wolf, M., Horn, A. B., Mehl, M. R., Haug, S., Pennebaker, J. W., and Kordy, H. (2008). Computergestützte quantitative Textanalyse. *Diagnostica*, 54(2):85–98.

Author Index

Amine Bayoudhi
ANLP Group, MIRACL laboratory
FSEGS, University of Sfax
B.P. 1088, 3018, Sfax, TUNISIA
bayoudhi.amine@gmail.com

Lamia Hadrich Belguith
ANLP Group, MIRACL laboratory
FSEGS, University of Sfax
B.P. 1088, 3018, Sfax, TUNISIA
l.belguith@fsegs.rnu.tn

Lucie Flekova
Ubiquitous Knowledge Processing Lab
Department of Computer Science
Technische Universität Darmstadt
Hochschulstraße 10, 64289 Darmstadt, Germany
www.ukp.tu-darmstadt.de

Hatem Ghorbel
ISIC Lab, HE-Arc Ingnierie
University of Applied Sciences

CH-2610 St-Imier, Switzerland
hatem.ghorbel@he-arc.ch

Iryna Gurevych
Ubiquitous Knowledge Processing Lab
Department of Computer Science
Technische Universität Darmstadt
Hochschulstraße 10, 64289 Darmstadt, Germany
www.ukp.tu-darmstadt.de
Deutsches Institut für Internationale Pädagogische Forschung (DIPF)
Germany

Houssem Koubaa
ANLP Group, MIRACL laboratory
FSEGS, University of Sfax
B.P. 1088, 3018, Sfax, TUNISIA
houssemkoubaa90@gmail.com

Alexander Magidow
University of Rhode Island
160 Swan Hall, 60 Upper College Road, Kingston
amagidow@uri.edu

Margot Mieskes
Hochschule Darmstadt - University of Applied Sciences
Fachbereich Media
Max-Planck-Str. 2, 64807 Dieburg, Germany
margot.mieskes@h-da.de

Andreas Peldszus
Applied Computational Linguistics/FSP Cognitive Science
University of Potsdam
Karl-Liebknecht Straße 24-25
14476 Potsdam, Germany
peldszus@uni-potsdam.de

Laurent Romary
INRIA, France

Uladzimir Sidarenka
Applied Computational Linguistics/FSP Cognitive Science
University of Potsdam
Karl-Liebknecht Straße 24-25
14476 Potsdam, Germany
sidarenk@uni-potsdam.de

Manfred Stede
Applied Computational Linguistics/FSP Cognitive Science
University of Potsdam
Karl-Liebknecht Straße 24-25
14476 Potsdam, Germany
stede@uni-potsdam.de

Tahir Sousa
University of Minnesota - Twin Cities
55455 Minneapolis, Minnesota, USA
tahirsousa@gmail.com